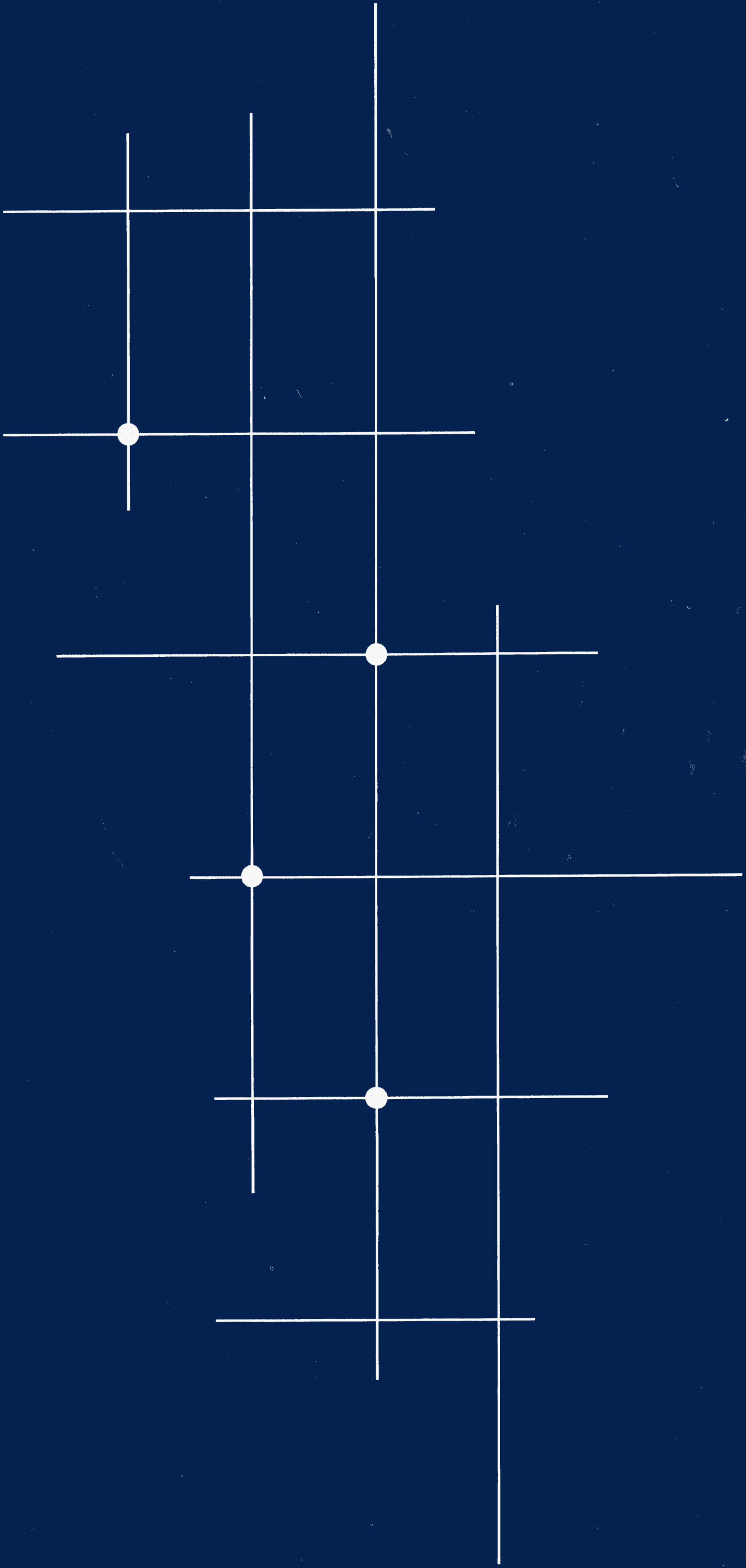
Systems
Network
Architecture



Formats

IBM

GA27-3136-7

Systems Network Architecture

Formats

GA27-3136-7
File No. GENL-30 (SNA)

Eighth Edition (March 1987)

The title of this publication has been changed from "SNA Reference Summary" to "SNA Formats."

This is a major revision of GA27-3136-6, which is now obsolete. The book should be reread in its entirety.

PREFACE

This book describes the Systems Network Architecture (SNA) formats used between subarea nodes and peripheral nodes, and between type 2.1 nodes using peer-to-peer protocols.

This book is intended for product developers, system programmers, and others who need detailed information about Systems Network Architecture in order to maintain, develop, or adapt a product or program for attachment to an SNA network.

Related publications, providing overview and protocol information, are:

- Systems Network Architecture Concepts and Products (GC30-3072)
- Systems Network Architecture Technical Overview (GC30-3073)
- IBM Synchronous Data Link Control Concepts (GA27-3093)
- Systems Network Architecture Format and Protocol Reference Manual: Architectural Logic (SC30-3112)
- Systems Network Architecture Format and Protocol Reference Manual: Architecture Logic for Type 2.1 Nodes (SC30-3422)
- Systems Network Architecture: Sessions Between Logical Units (GC20-1868)
- Systems Network Architecture Format and Protocol Reference Manual: Architecture Logic for LU Type 6.2 (SC30-3269)
- Systems Network Architecture Format and Protocol Reference Manual: Distribution Services (SC30-3098)
- Systems Network Architecture Format and Protocol Reference Manual: Management Services (SC30-3346)
- Token-Ring Network Architecture Reference (6165877)
- Document Interchange Architecture: Technical Reference (SC23-0781)

HOW THIS BOOK IS ORGANIZED

This book identifies the formats and meanings of the bytes that a basic link unit (BLU) contains. A BLU is the basic unit of transmission at the data link and link station level. The figure on page v illustrates the organization of this book.

Chapter 1 identifies the formats and meanings of the bytes in a link header and a link trailer.

Chapter 2 identifies the formats and meanings of the bytes in an SDLC station identification command.

Chapter 3 identifies the formats and meanings of the bytes in a tranmission header.

Chapter 4 identifies the formats and meanings of the bytes in a request or response header.

Chapter 5.1 identifies the formats and meanings of the bytes in a request unit.

Chapter 5.2 identifies the formats and meanings of the bytes in a response unit.

Chapter 6 explains the transmission services and function management profiles that SNA defines to describe session characteristics.

Chapter 7 identifies the formats and meanings of the bytes in user-structured subfields that appear in a request or response unit.

Chapter 8 identifies the formats and meanings of the control vectors, session keys, and management services vectors that appear in a request or response unit.

Chapter 9 explains the meanings of the SNA-defined sense data that appears in a negative response unit.

Chapter 10 presents the descriptions and formats of the different function management headers.

Chapter 11 identifies the formats and meanings of the bytes in a presentation services header.

Chapter 12 identifies the formats and meanings of the general data stream (GDS) variables that are specific to SNA service transaction programs.

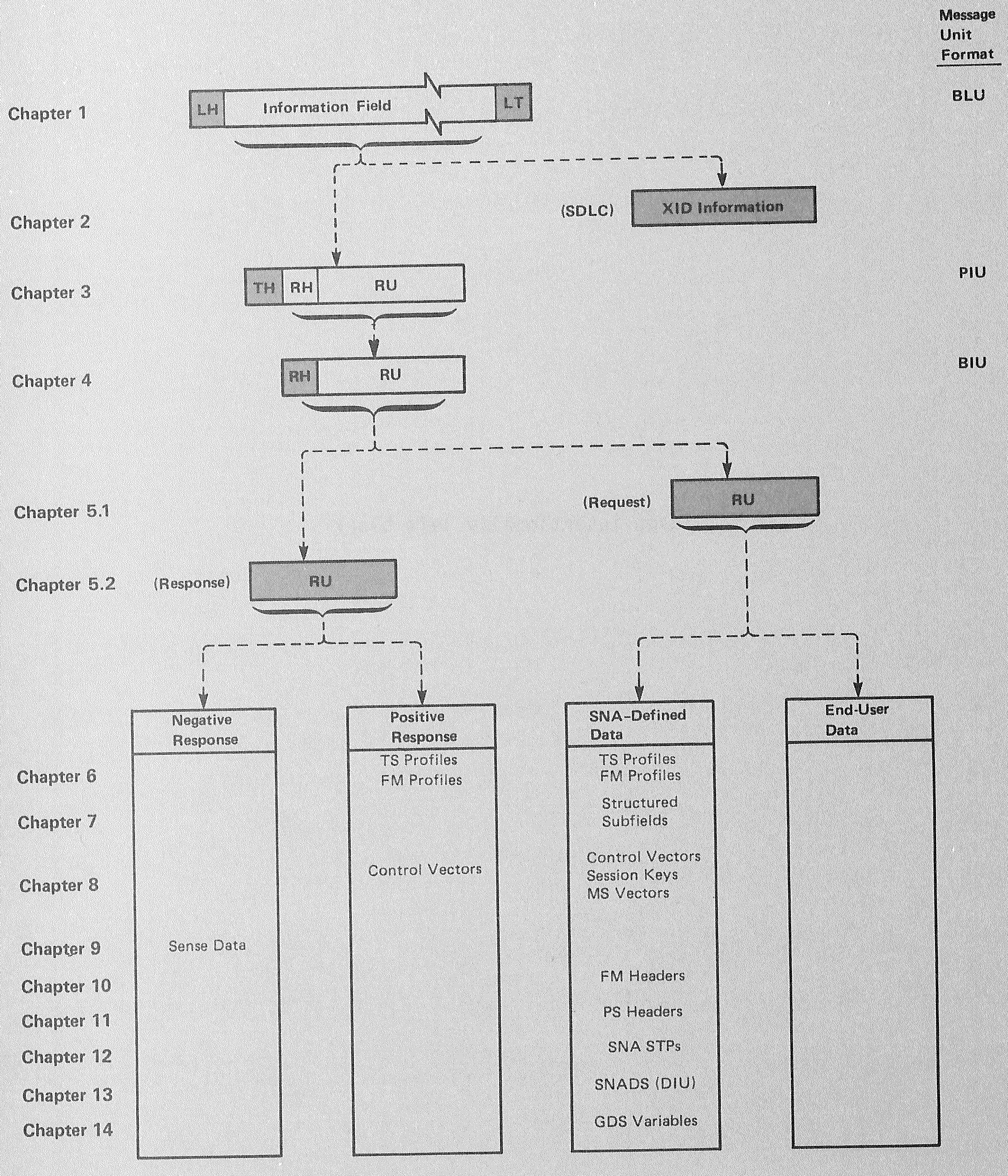
Chapter 13 identifies the formats and meanings of the message units that SNA Distribution Services transaction programs use.

Chapter 14 identifies the general data stream (GDS) variables that are for general use.

Appendix A provides a summary of SNA character sets and symbol-string types.

Appendix B provides a summary of general data stream identifier (GDS ID) value assignments.

Appendix C lists the abbreviations and symbols that are used in this book.



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

This page intentionally left blank

CONTENTS

Chapter 8. Common Fields . . . 8-1

LIST OF ILLUSTRATIONS

Chapter 1. DLC Links

Chapter 2. Exchange Identification (XID) Information Fields

Chapter 3. Transmission Headers

Chapter 4. Request/Response Headers

Chapter 5.1. Request Units

Chapter 5.2. Response Units

Chapter 6. Profiles

Chapter 7. User Data

Chapter 8. Common Fields

Chapter 9. Sense Data

Chapter 10. Function Management Headers

Chapter 11. Presentation Services Headers

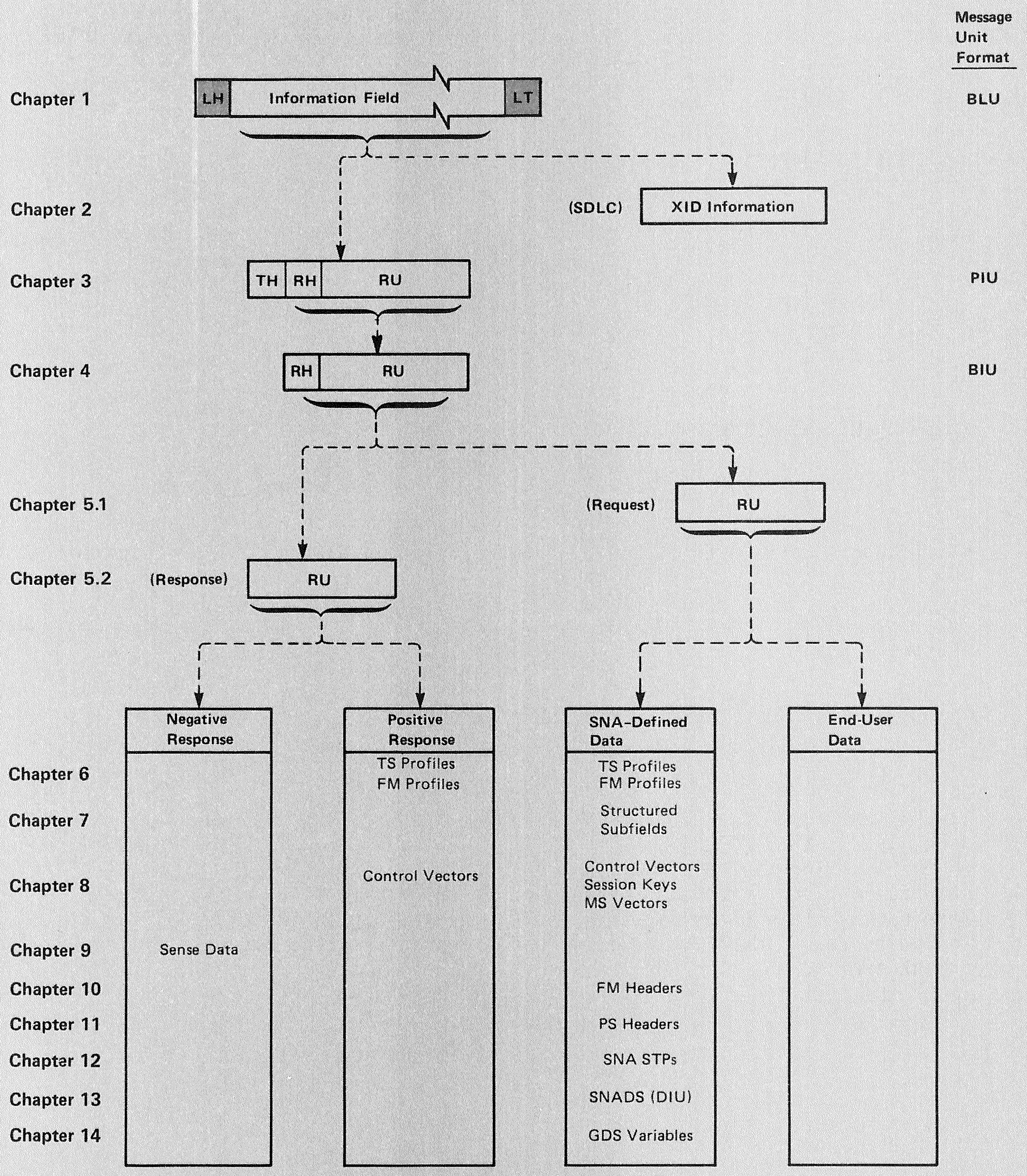
Chapter 12. GDS Variables for SNA Service Transaction Programs

Chapter 13. SNA Distribution Services

Chapter 14. GDS Variables for General Use

Chapter 1

DLC Links



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 1. DLC LINKS

Two data link controls are described in this chapter: "Synchronous Data Link Control (SDLC)", beginning on this page, and the "Token-Ring Network DLC" on page 1-12.

SYNCHRONOUS DATA LINK CONTROL (SDLC)

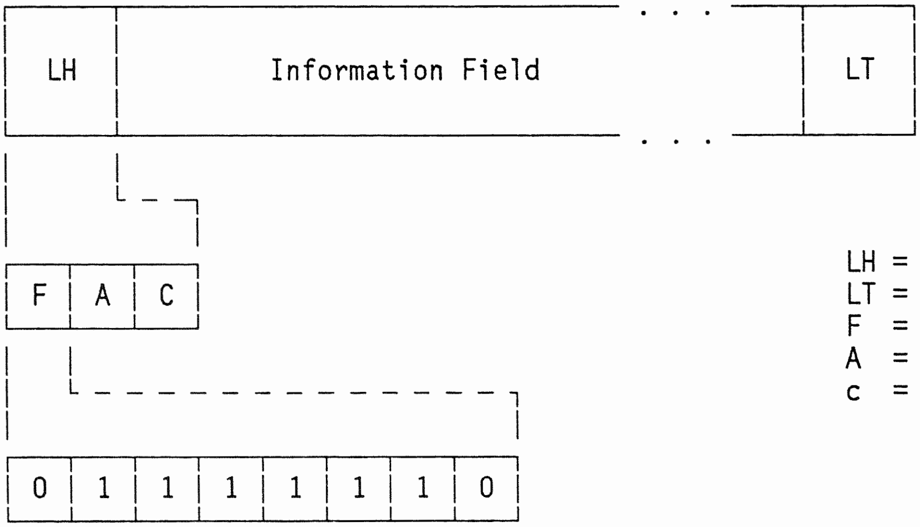
All transmissions on an SDLC link are organized in a specific format called a frame:

```
  Frame = BLU = LH [,I-field], LT

   where:        BLU = Basic Link Unit
                  LH = Link Header
             I-field = Information field
                  LT = Link Trailer
```

Link headers and link trailers contain data link control information for synchronous data link control (SDLC) links. An SDLC frame begins with the link header (LH), which has three fields: the Flag, Address, and Control fields. The link trailer (LT) follows the Information field and is three bytes long. The first two bytes make up the Frame Check Sequence field; the last byte, the closing Flag field. The following pages identify the formats and meanings of the bytes in a link header and a link trailer.

LINK HEADER
(FLAG)



LH = Link Header
LT = Link Trailer
F = Flag
A = Address
c = Control

Figure 1-1. Flag field of link header: always X'7E', B'01111110'.

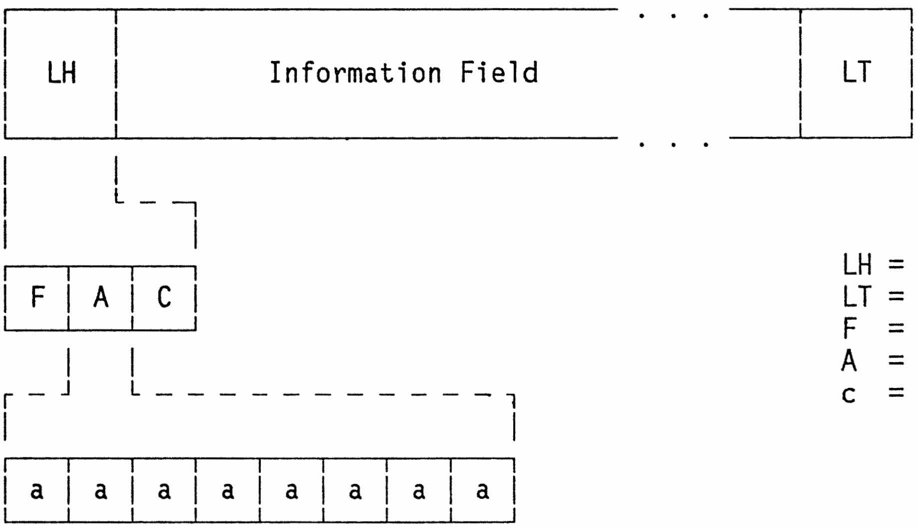
All frames begin with a Flag field. The configuration of the flag is always 01111110 (X'7E'). Because frames also end with flags (see link trailer), the trailing flag of one frame may serve as the leading flag of the next frame. When receiving, the last 0 in the trailing flag may also be the first 0 in the next leading flag, as Figure 1-2 illustrates.

```
              |--leading flag-|
0 1 1 1 1 1 1 0 1 1 1 1 1 1 0
|-trailing flag-|
```

Figure 1-2. Shared Trailing/Leading 0 in SDLC Flags

Note: Zero bit insertion between the beginning and ending flags prevents a flag pattern from occurring anywhere else in the frame.

LINK HEADER
(ADDRESS)



LH = Link Header
LT = Link Trailer
F = Flag
A = Address
c = Control

Figure 1-3. Address field of link header: B'aaaaaaaa'

The second byte of the link header is the Address field. This address can be:

- a specific link station address -- to only one link station
- a group address -- to one or more link stations
- a broadcast address (X'FF', B'11111111') -- to all link stations
- a "no stations" address (X'00').

The "no stations" address is reserved and should not be used for any link station or group of link stations.

Note: The specific link station address of the secondary is used when the transmission is going from primary to secondary or from secondary to primary.

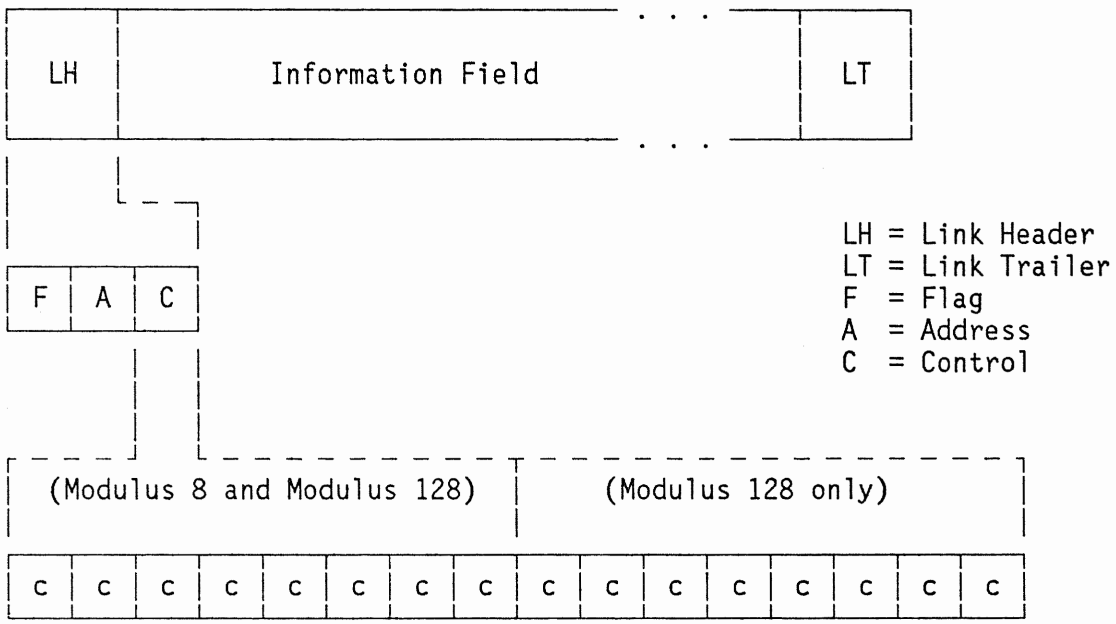


Figure 1-4. Control field of link header: B'cccccccc' for modulus 8; B'cccccccc cccccccc' for modulus 128.

The third byte (or third and fourth bytes) of the link header is the Control field. The Control field contains either an SDLC command or a response. All frames transmitted by a primary station are commands, while frames transmitted by a secondary station are responses. There are three categories of SDLC commands and responses:

- Unnumbered Format
- Supervisory Format
- Information Format

Unnumbered Format: These commands and responses have a poll/final (P/F) bit that is set to 1 to solicit a response (P bit) or when it is the last SDLC frame of a transmission (F bit). This bit is a poll bit for commands and a final bit for responses. Each of the Unnumbered Format commands and responses have two possible hex values: one value for when the poll/final bit is 0 and another value for when the poll/final bit is 1.

Supervisory Format: These commands and responses have a varying number of possible hex values. The number of possible hex values corresponds to the receive sequence numbers assigned to this frame and the setting of the P/F bit. To increase the sequence number modulus from 8 to 128, a two-byte extended Control field is used.

Information Format: These commands and responses also vary in the number of possible hex values. The number of possible hex values correspond to the send and receive sequence numbers assigned to this frame and the setting of the P/F bit. To increase the sequence number modulus from 8 to 128, a two-byte extended Control field is used.

The Information Format is identified by a 0 in the low-order bit of the first or only byte of the Control field. In an Information Format SDLC command or response, the Information field contains a PIU (Path Information Unit). The remaining chapters of this book, with the exception of Chapter 2, discuss the contents of the PIU.

Figure 1-5 lists the SDLC commands and responses for modulus 8 (one-byte) Control fields; Figure 1-6 lists them for modulus 128 (two-byte) Control fields.

Figure 1-7 describes the Information field of the Frame Reject (FRMR) response frame, which is one of the unnumbered formats listed in Figure 1-5.

FORMAT	BINARY CONFIGURATION	HEX EQUIVALENT P/F off,P/F on	COMMAND NAME	ACRO- NYM
Unnumbered Format	000 P/F 0011	X'03', X'13'	Unnumbered Information	UI
	000 F 0111	X'07', X'17'	Request Initialization Mode	RIM
	000 P 0111	X'07', X'17'	Set Initialization Mode	SIM
	000 F 1111	X'0F', X'1F'	Disconnect Mode	DM
	001 P 0011	X'23', X'33'	Unnumbered Poll	UP
	010 F 0011	X'43', X'53'	Request Disconnect	RD
	010 P 0011	X'43', X'53'	Disconnect	DISC
	011 F 0011	X'63', X'73'	Unnumbered Acknowledgment	UA
	100 P 0011	X'83', X'93'	Set Normal Response Mode	SNRM
	100 F 0111	X'87', X'97'	Frame Reject	FRMR
	101 P/F 1111	X'AF', X'BF'	Exchange Identification	XID
	110 P/F 0111	X'C7', X'D7'	Configure	CFGR
	110 P 1111	X'CF', X'DF'	Set Normal Response Mode Extended	SNRME
	111 P/F 0011	X'E3', X'F3'	Test	TEST
	111 F 1111	X'EF', X'FF'	Beacon	BCN
Supervisory Format	RRR P/F 0001	X'x1', X'x1'	Receive Ready	RR
	RRR P/F 0101	X'x5', X'x5'	Receive Not Ready	RNR
	RRR P/F 1001	X'x9', X'x9'	Reject	REJ
Information Format	RRR P/F SSS0	X'xx', X'xx'	Numbered Information Present	

Notes: P = Poll bit (sent to secondary station)
F = Final bit (sent to primary station)
RRR = Nr (receive count)
SSS = Ns (send count)

Figure 1-5. Control fields for SDLC Commands and Responses—Modulus 8

FORMAT	BINARY CONFIGURATION	HEX EQUIVALENT	COMMAND NAME	ACRO-NYM
Unnumbered Format	same as modulus 8 (one-byte), as in Figure 1-5.			
Supervisory Format	0000 0001 RRRR RRR P/F	X'01xx'	Receive Ready	RR
	0000 0101 RRRR RRR P/F	X'05xx'	Receive Not Ready	RNR
	0000 1001 RRRR RRR P/F	X'09xx'	Reject	REJ
Information Format	SSSS SSS0 RRRR RRR P/F	X'xxxx'	Numbered Information Present	

Notes: P = Poll bit (sent to secondary station)
F = Final bit (sent to primary station)
RRR = Nr (receive count)
SSS = Ns (send count)

Figure 1-6. Control fields for SDLC Commands and Responses—Modulus 128

Information Field of the FRMR Response Frame

Modulus 8:

```
┌─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┐
| Control Field  |  Nr  |0|  Ns  |0|0 0 0 0|z y x w|
└─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┘
|                |               |               |
|     Byte 0     |    Byte 1     |    Byte 2     |
└ ─ ─ ─ ─ ─ ─ ─ ─┴ ─ ─ ─ ─ ─ ─ ─ ┴ ─ ─ ─ ─ ─ ─ ─ ┘
```

Modulus 128:

```
┌─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┬─┐
|           Control Field           |      Ns     |0|      Nr     |0|0 0 0 0|z y x w|
└─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┘
|                |                |               |               |               |
|     Byte 0     |     Byte 1     |     Byte 2    |     Byte 3    |     Byte 4    |
└ ─ ─ ─ ─ ─ ─ ─ ─┴ ─ ─ ─ ─ ─ ─ ─ ─┴ ─ ─ ─ ─ ─ ─ ─┴ ─ ─ ─ ─ ─ ─ ─┴ ─ ─ ─ ─ ─ ─ ─┘
```

Note: For modulus 128, if control field causing FRMR is an unnumbered format (one-byte), it is placed in byte 0 and byte 1 is set to all 0's.

Field	Description	Explanation/Usage
C	Control Field	Control field of the rejected command, as received
Nr	Receive Count	This station's present receiver frame count (the existing count prior to FRMR)
Ns	Send Count	This station's present transmitter frame count (the existing count prior to FRMR)
	Rejection Indicators:	
z	Count	0 = no error 1 = Received Nr disagrees with transmitted Ns
y	Buffer	0 = no error 1 = Buffer overrun (I-field is too long)
x	I-field	0 = no error 1 = Prohibited I-field received
w	Command	0 = no error 1 = Invalid or nonimplemented command received

Figure 1-7. Information field of the FRMR Response Frame: modulus 8 and modulus 128. In each byte, the low-order bit is sent first and the high-order bit is sent last.

LINK TRAILER (FRAME CHECK SEQUENCE)

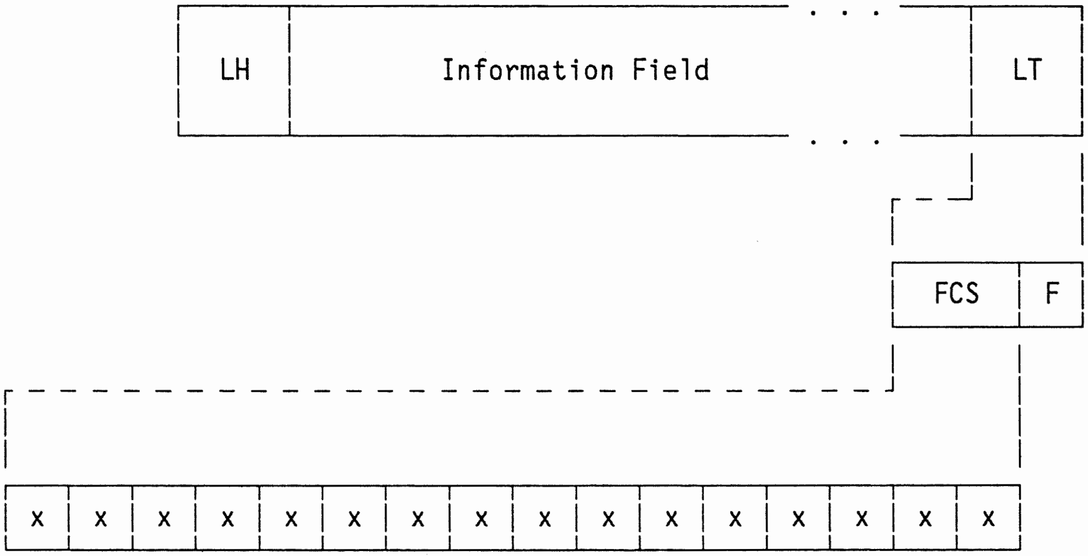


Figure 1–8. Frame Check Sequence field of link trailer

The Frame Check Sequence field carries information that the receiver uses to check the received frame for errors that may have been introduced by the communication channel. This field contains a 16-bit check sequence that is the result of a computation on the contents of both the LH (with the exception of the flag) and the Information field at the transmitter. Cyclic redundancy checking (CRC) is used to perform this calculation.

The receiver performs a similar computation and checks its results.

LINK TRAILER
(FLAG)

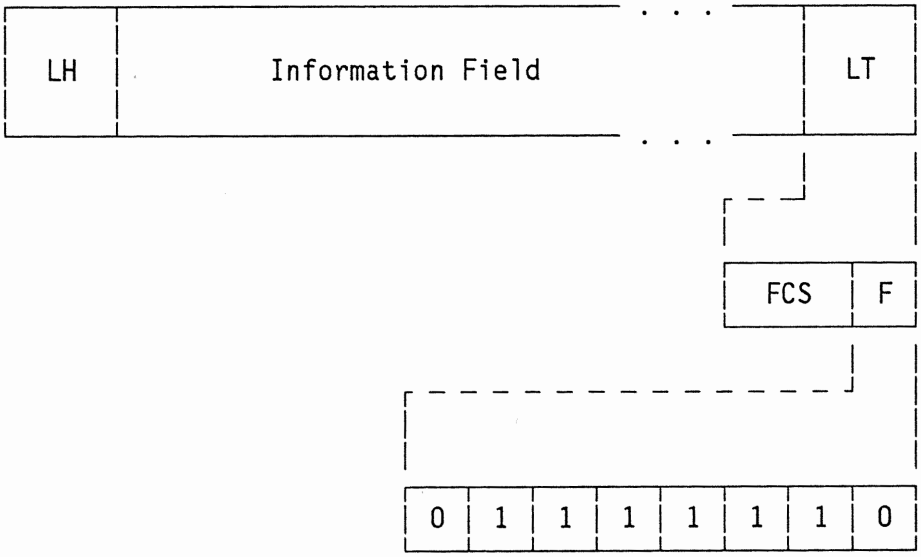


Figure 1-9. Flag field of link trailer: always X'7E', B'01111110'.

All frames end with a Flag field. The configuration of the ending (trailing) flag is the same as that of the beginning (leading) flag that is present in the link header: 01111110 (X'7E').

The token-ring network DLC consists of two sublayers: the medium access control and the logical link control. The medium access control (MAC) sublayer controls the routing of information between the physical layer and the logical link control sublayer. It provides the following functions: address recognition, frame copying, frame delimiting, and 32-bit frame check sequence generation and verification. The logical link control (LLC) sublayer provides sequential, connection-oriented data transfer.

The following commands and responses, a subset of those shown in Figure 1-6, are used by the LLC sublayer in the token-ring network:

Format	Command/Response Name
Unnumbered Format	DM Response
	DISC Command
	UA Response
	SABME Command
	FRMR Response
	XID Command or Response
	Test Command or Response
Supervisory Format	Receive Ready
	Receive Not Ready
	Reject
Information Format	Numbered Information Present

Figure 1-10. LLC Commands and Responses

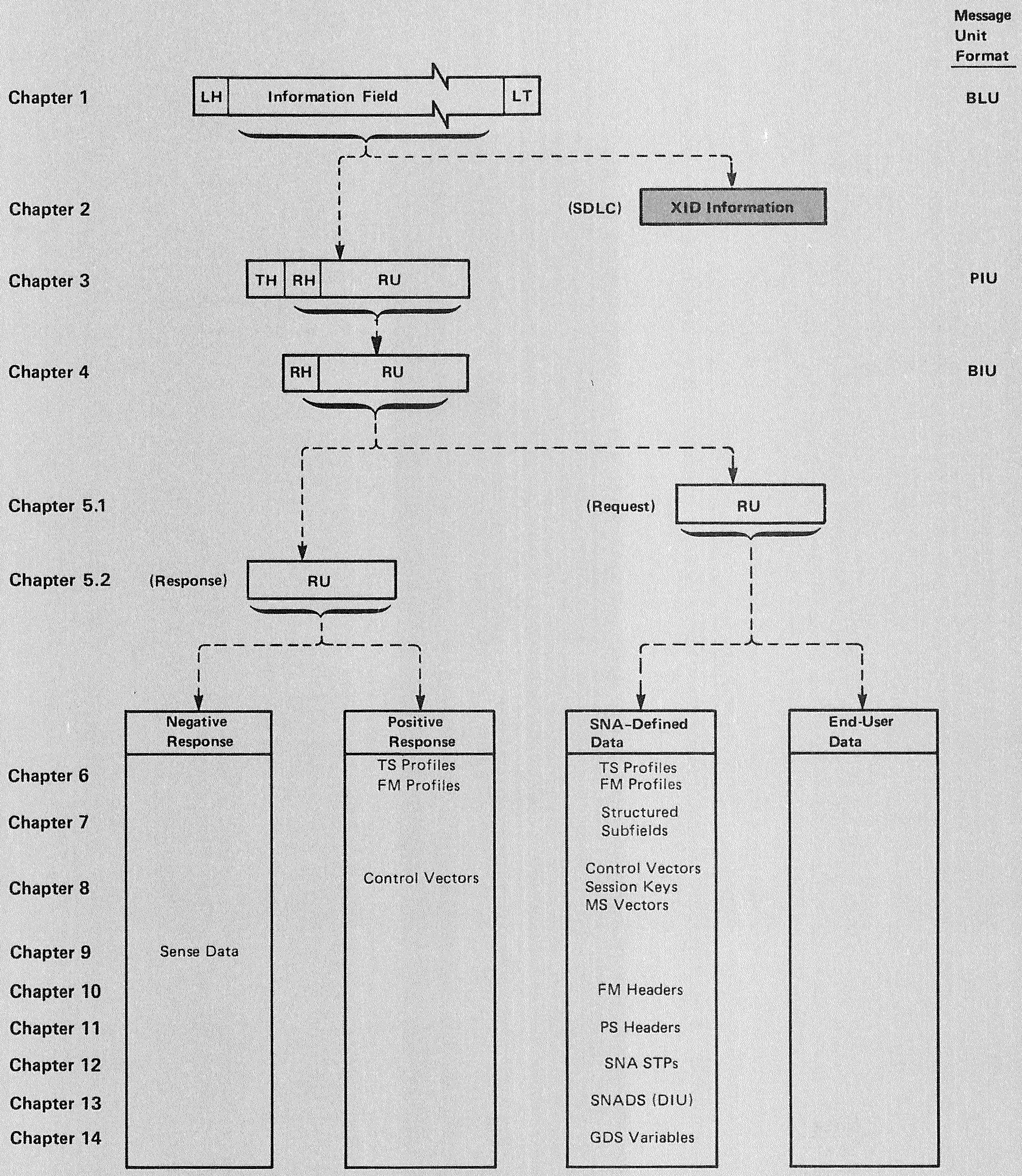
The code points associated with these commands and responses are the same as those shown in Figure 1-6.

The token-ring network DLC, in contrast to SDLC, transmits the high-order bit first and the low-order bit last within each byte. Also, zero bit insertion is required on the token-ring network, since the differential Manchester encoding technique is used.

Additional information about the token-ring network DLC architecture is contained in the Token-Ring Network Architecture Reference.

Chapter 2

Exchange Identification (XID) Information Fields



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 2. EXCHANGE IDENTIFICATION (XID) INFORMATION FIELDS

This chapter describes the formats of the information field of the DLC XID command and response.

DLC XID INFORMATION-FIELD FORMATS

0	bits 0-3, format of XID I-field: X'0' fixed format: only bytes 0-5 are included X'1' variable format (for T1|2.0 to T4|5 node exchanges): bytes 0-p are included X'2' reserved X'3' variable format (for T2.1 to T2.1 and T2.0 to T4|5 node exchanges): bytes 0-p are included X'8'-X'F' defined for external standards organizations bits 4-7, type of the XID-sending node: X'1' T1 X'2' T2 X'3' reserved X'4' subarea node (T4 or T5)
1	Length, in binary, of variable-format XID I-field (bytes 0-p); reserved for fixed-format XID I-field
2-5|7	Node Identification
2-5	bits 0-11, block number: an IBM product specific number; see the individual product specifications for the specific values used Note: The values all 0's and all 1's indicate that bytes 2-5 do not contain a unique node identifier. bits 12-31, ID number: a binary value that, together with the block number, identifies a specific station uniquely within a customer network installation; the ID number can be assigned in various ways, depending on the product; see the individual product specifications for details Note: When the Block Number field does not contain all 0's or all 1's, a value of all 0's in the ID number indicates that no ID number has been assigned. Note: For XID format 3, the contents of bytes 2-5 of the node identification field are used in some instances as a role-negotiation-value to resolve contention in protocol roles of nodes, e.g., primary/secondary DLC roles or the ODAI value to be appended to the (OAF', DAF') values assigned at a node. When a role-negotiation value is needed and the node does not supply a unique node identification value, it supplies a random value in the ID number field.

- End of Format 0

- 6-p Format 1 Continuation
- 6-7 Reserved
- 8 Link Station and Connection Protocol Flags
- 8 bits 0-1, reserved
  - bit 2, link-station role of XID sender:
    - 0 sender is a secondary link station (nonnegotiable)
    - 1 sender is a primary link station (nonnegotiable)
  - bit 3, reserved.
  - bits 4-7, link-station transmit-receive capability:
    - X'0' two-way alternating
    - X'1' two-way simultaneous
- 9 Characteristics of the node of the XID sender:
  - bits 0-1, reserved
  - bits 2-3, segment assembly capability of the path control element of the node:
    - 00 the Mapping field is ignored and PIUs are forwarded unchanged
    - 01 segments are assembled on a link-station basis
    - 10 segments are assembled on a session basis
    - 11 only whole BIUs are allowed
  - bits 4-5, reserved
  - bit 6, short-hold status (reserved if byte 9, bit 7 is set to 0):
    - 0 sender not already engaged in a logical connection using short-hold mode
    - 1 sender already engaged in a logical connection using short-hold mode
  - bit 7, short-hold capability of the XID sender:
    - 0 short-hold mode not supported
    - 1 short-hold mode supported
- 10-11 Maximum I-field length that the XID sender can receive:
  - bit 0, format flag:
    - 0 bits 1-15 contain the maximum I-field length (only value defined)
  - bits 1-15, maximum I-field length, in binary
- 12 bits 0-3, reserved
  - bits 4-7, SDLC command/response profile:
    - X'0' SNA link profile (only value defined)

      Note: These profiles refer to the mandatory command/response support on an SDLC link, as follows:

      - For an SDLC link in normal response mode (NRM/NRME), having a point-to-point or multipoint configuration (determined from system definition), the support required is:

Commands	Responses
I-frames	I-frames
RR	RR
RNR	RNR
Test	Test
XID	XID
SNRM/SNRME	UA
Disconnect	DM
-	RD
-	Frame Reject
Reject	Reject

Note 1: The RD response is sent by the secondary station if and only if the PU in its node receives a DISCONTACT request from its CP.

Note 2: Reject is required only if both sender and receiver have two-way simultaneous transmit-receive capability.

- For an SDLC link in normal response mode (NRM), having a loop configuration (determined from system definition), the support required is:

Commands	Responses
I-frames	I-frames
RR	RR
RNR	RNR
Test	Test
XID	XID
SNRM	UA
Disconnect	DM
UP	-
-	Frame Reject
Configure	Configure
-	Beacon
-	RD

Note: The RD response is sent by the secondary station if and only if the PU in its node receives a DISCONTACT request from its CP.

13	bits 0-1, reserved bit 2, SDLC initialization mode options: 0 SIM and RIM not supported 1 SIM and RIM supported bits 3-7, reserved
14-15	Reserved

16	bit 0, reserved
	bits 1-7, maximum number of I-frames that can be received by the XID sender before an acknowledgment is sent, with an implied modulus for the send and receive sequence counts—less than 8 implies a modulus of 8, 8 or greater implies a modulus of 128
17	Reserved
	• For byte 9, bit 7 = 0 (short-hold mode not supported)
18-p	SDLC Address Assignment Field
18	Length (p minus 18), in binary, of the SDLC address to be assigned
19-p	Secondary station address to be assigned
	• For byte 9, bit 7 = 1 (short-hold mode supported)
18-p	Short-Hold Mode Dependent Parameters
18	Reserved
19-n	Dial Digits of XID Sender
19	Number, in binary, of dial digits
20-n	Dial digits: a string of digits, each having the form X'Fn' (0≤n≤9)
n+1-p	Dial digits of an available short-hold mode port
	Note: This field is included only in an XID from a T4 or T5 node and only for an incoming call on an already logically busy (byte 9, bit 6 = 1) short-hold mode port. If this field is not included, then p = n.
n+1	Number, in binary, of dial digits of an available short-hold mode port, if one exists
n+2-p	Dial digits of an available short-hold mode port: a string of digits, each having the form X'Fn' (0≤n≤9)
	Note: Byte n+1 is set to the value X'00' and the n+2-p field is not included if no free alternate port is found. In this case the station may retry later on the same port used for the current XID.
	• End of Format 1

6-p	Format 3 Continuation
6-7	Reserved
8-9	Characteristics of the node of the XID sender:
	bit 0, INIT-SELF support:
	0 INIT-SELF may be sent to the XID sender
	Note: If the XID sender does not contain an SSCP, it forwards any INIT-SELF received to the proper node for processing, which returns the response to the originator of the request.
	1 INIT-SELF (and character-coded logon) cannot be sent to the XID sender
	Note: For bits 0-1, the value 11 is reserved.
	bit 1, stand-alone BIND support:
	0 BIND may be sent to the XID sender without a prior INITIATE sequence
	1 BIND may not be sent to the XID sender
	Note: For bits 0-1, the value 11 is reserved.
	bits 2-3, retired, set to 11
	bits 4-7, reserved
	bit 8, retired
	bits 9-11, reserved
	bits 12-13, XID exchange state:
	00 exchange state indicators not supported (set only by implementations not at the current level of SNA)
	01 negotiation proceeding

	10 pre-negotiation exchange
	11 non-activation exchange
	bits 14-15, reserved
10-16	Reserved
17	DLC type:
	X'01' SDLC (only value defined)
18-n	DLC Dependent Section
18	Length, in binary, of the DLC Dependent Section field (Length field includes itself in the length specified.)
19	Link Station and Connection Protocol Flags
19	bit 0, reserved
	bit 1, ABM support indicator:
	0 XID sender cannot be an ABM combined station
	1 XID sender can be an ABM combined station
	bits 2-3, link-station role of XID sender:
	00 sender is a secondary link station (nonnegotiable)
	01 sender is a primary link station (nonnegotiable)
	10 reserved
	11 negotiable (primary or secondary capability)
	Note: For ABM stations, the value of bits 2-3 is used only for the purposes of OAF'-DAF' assignment and deciding which node sends the Set Mode command.
	bits 4-5, reserved
	bits 6-7, link-station transmit-receive capability:
	00 two-way alternating
	01 two-way simultaneous
20	Reserved
21-22	Maximum BTU length that the XID sender can receive:
	bit 0, format flag:
	0 bits 1-15 contain the maximum BTU length (only value defined)
	bits 1-15, maximum BTU length, in binary
23	bits 0-3, reserved
	bits 4-7, SDLC command/response profile:
	X'0' SNA link profile (only value defined)
	Note: These profiles refer to the mandatory command/response support on an SDLC link, as follows:
	• For an SDLC link in normal response mode (NRM/NRME), having a point-to-point or multipoint configuration (determined from system definition), the support required is:

Commands	Responses
I-frames	I-frames
RR	RR
RNR	RNR
Test	Test
XID	XID
SNRM/SNRME	UA
Disconnect	DM
-	RD
-	Frame Reject
Reject	Reject

Note 1: The RD response is sent by the secondary station if and only if the PU in its node receives a DISCONTACT request from its CP.

Note 2: Reject is required only if both sender and receiver have two-way simultaneous transmit-receive capability.

- For an SDLC link in normal response mode (NRM), having a loop configuration (determined from system definition), the support required is:

Commands	Responses
I-frames	I-frames
RR	RR
RNR	RNR
Test	Test
XID	XID
SNRM	UA
Disconnect	DM
UP	-
-	Frame Reject
Configure	Configure
-	Beacon
-	RD

Note: The RD response is sent by the secondary station if and only if the PU in its node receives a DISCONTACT request from its CP.

- For an SDLC link in asynchronous balanced mode (ABM) (determined from the Link-Station Role of XID Sender field), having a point-to-point configuration, the support required is:

Commands	Responses
I-frames	-
RR	RR
RNR	RNR
Reject	Reject
SABME	UA
Disconnect	DM
Test	Test
XID	XID
-	Frame Reject

Note 1: All commands and responses are transmitted and received in two-octet format (extended control field).

Note 2: Frame Reject is not required to be transmitted; receive capability is required.

24 bits 0-1, reserved
bit 2, SDLC initialization mode options:
  0 SIM and RIM not supported
  1 SIM and RIM supported
bits 3-7, reserved

25-26 Reserved

27 bit 0, reserved
bits 1-7, maximum number of I-frames that can be received by the XID sender before an acknowledgment is sent, with an implied modulus for the send and receive sequence counts--less than 8 implies a modulus of 8, 8 or greater implies a modulus of 128

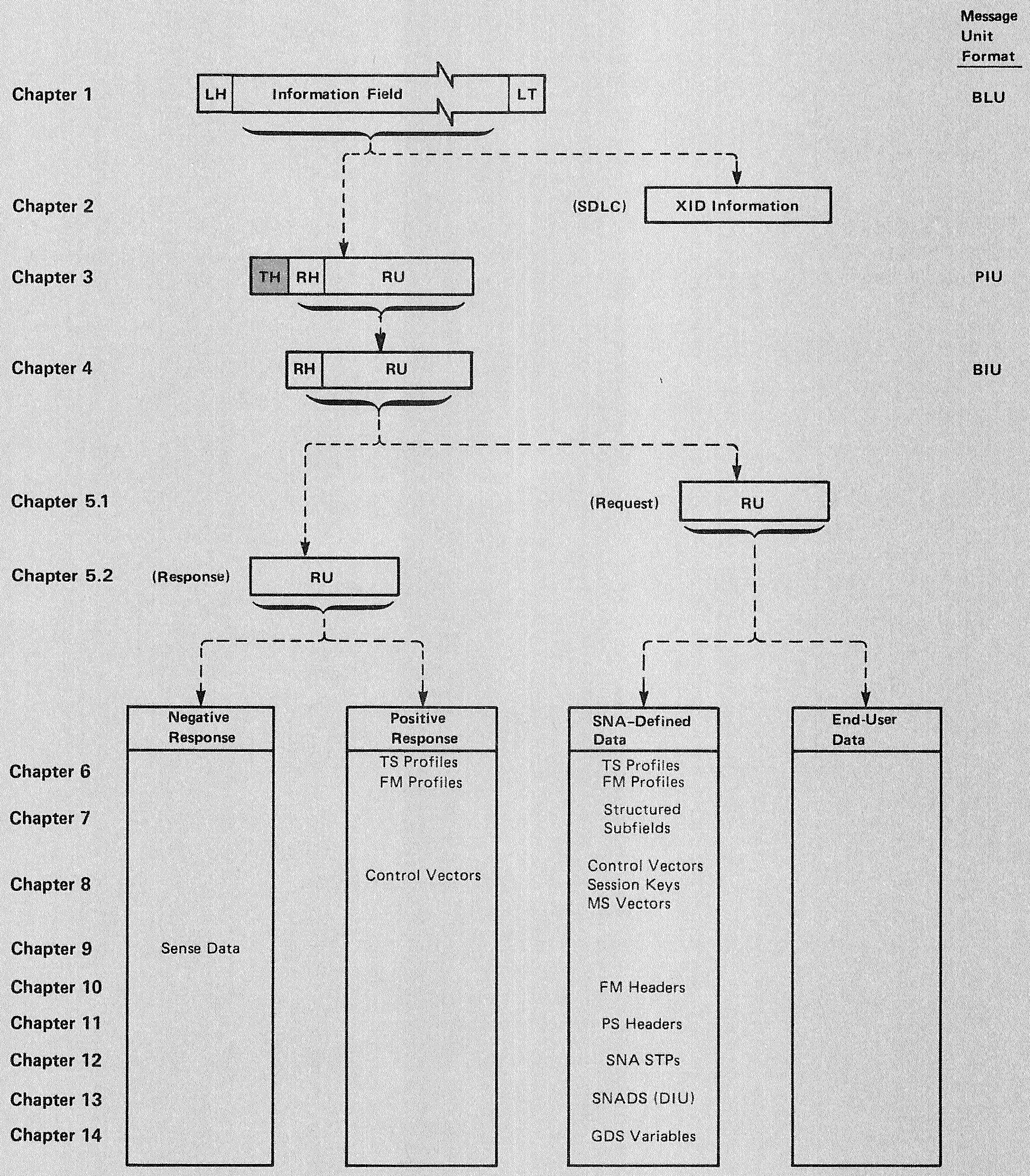
28(=n) Reserved

n+1-p Control vectors, as described in "Control Vectors" in Chapter 8
Note: The following control vectors may be included; they are parsed according to subfield parsing rule KL:
X'0E' Network Name control vector: type X'F4', network-qualified CP name (always present; the network identifier is always used, i.e., valid lengths of the CP name are 3 to 17 bytes with an imbedded period)
X'0E' Network Name control vector: type X'F7', local name of the ALS at the XID sender (always present)
X'10' Product Set ID control vector (always present)
  Note: When included in XID, the product set ID is limited to 60 bytes or less in length.
X'22' XID Negotiation Error control vector (conditionally present: present when an error during XID negotiation is detected; more than one may be present)

This page intentionally left blank

Chapter 3

Transmission Headers



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 3. TRANSMISSION HEADERS

INTRODUCTION

A transmission header (TH) is the leading, or only, field of every PIU. The first half-byte of any TH is the Format Identifier (FID) field. FID2 corresponds to hexadecimal value 2 in the FID field. The FID2 TH is described below.

FID 2 LAYOUT

Byte		
0	FID2—Format Identification MPF—Mapping Field ODAI—OAF'-DAF' Assignor Indicator EFI—Expedited Flow Ind.	Reserved Byte
2	DAF'—Destination Address	OAF'—Origin Address
4	SNF—Sequence Number Field	

Figure 3-1. Transmission Header for FID Type 2

FID 2 Field Descriptions

> FID2 is the format used between a T4 or T5 node and an adjacent T2 node, or between adjacent T2.1 nodes.

0 bits 0-3, FID2--Format Identification: 0010

bits 4-5, MPF--Mapping Field. The MPF consists of bit 4, the Begin-BIU (BBIU) bit, and bit 5, the End-BIU (EBIU) bit. It specifies whether the information field associated with the TH is a complete or partial BIU, and, if a partial BIU, whether it is the first, a middle, or the last segment.

10 first segment of a BIU (BBIU, ¬EBIU)
00 middle segment of a BIU (¬BBIU, ¬EBIU)
01 last segment of a BIU (¬BBIU, EBIU)
11 whole BIU (BBIU, EBIU)

bit 6, ODAI--OAF'-DAF' Assignor Indicator (used for T2.1 - T2.1 flows; otherwise, reserved). The ODAI indicates which node assigned (at session-activation time) the OAF'-DAF' values carried in the TH (see <u>SNA Format and Protocol Reference Manual: Architecture Logic for Type 2.1 Nodes</u> for details). Together with the DAF' and OAF' values, the ODAI value forms a 17-bit local-form session identifier (LFSID); the DAF' and OAF' values used in the TH in one direction are reversed in the other direction.

bit 7, EFI--Expedited Flow Indicator. The EFI designates whether the PIU belongs to the normal or expedited flow. Normal-flow PIUs are kept in order on a session basis by PC; so are expedited-flow PIUs. Expedited-flow PIUs can pass normal-flow PIUs flowing in the same direction at queuing points in TC within half-sessions and boundary function session connectors. It has the following meaning:

0 normal flow

1 expedited flow

1 Reserved

2 DAF'--Destination Address Field. See discussion above for ODAI.

3 OAF'--Origin Address Field. See discussion above for ODAI.

Note: The PU_T2 is always assigned the local address value of 0. Therefore, BIUs *to* the physical unit always have the associated DAF' = 0; BIUs *from* the physical unit always have the associated OAF' = 0. The OAF' is also 0 for BIUs *from* the SSCP, and DAF' is 0 for BIUs *to* the SSCP. A PU_T4|5 adjacent to the T2 node has the local address X'FF'.

4-5 SNF--Sequence Number Field. The Sequence Number Field contains a numerical identifier for the associated BIU; path control, when segmenting, puts the same SNF value in each segment derived from the same BIU. The numerical identifier used depends on a number of factors. If the TS profile indicates sequence numbers are not used, the SNF value is a 16-bit identifier that distinguishes a request being sent or responded to from any other outstanding request on the same flow. If the TS profile indicates sequence numbers are used, the flow is a factor. Expedited-flow requests (other than SIG for LU 6.2) carry 16-bit identifiers; expedited-flow responses echo the SNF values of their corresponding requests. Normal-flow requests, other than between LU 6.2's, carry 16-bit numerical values ranging in value from 1-65,535 (incremented by 1 for each request) and wrapping through 0 thereafter; the corresponding responses echo their SNF values. The table below defines the SIG and normal-flow SNF usage between LU 6.2's.

	Request	Response
FMD|LUSTAT with BB	A	C
FMD|LUSTAT with ¬BB	A	B
BIS	A	D
RTR	A	E
SIG	B	E

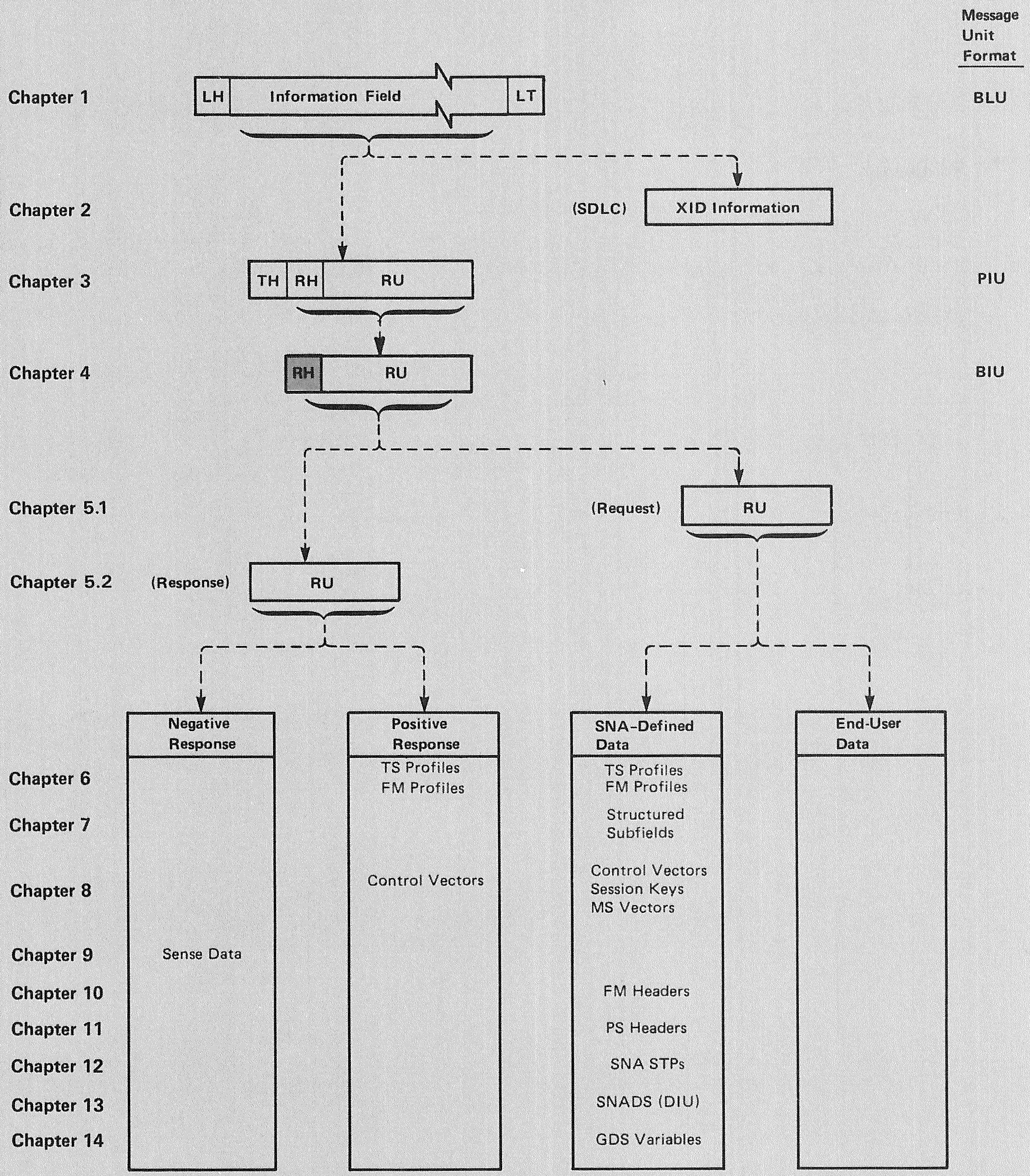
A: A 16-bit number (1-65,535) incremented by 1 for each request and wrapping through 0 thereafter
B: Low-order 15 bits of the SNF in the request that carried the last successful BB; the high-order bit identifies the half-session that started the bracket (0 = secondary, 1 = primary); in the case of the first bracket of a session, where the BB is implied, not sent, the low-order 15 bits are 0 and the high-order bit is 1.
C: Low-order 15 bits of the SNF in the BB request being responded to; the high-order bit identifies the sender of the BB request (0 = secondary, 1 = primary).
D: The half-session does not respond to BIS.
E: Same value as the corresponding request.

Note: For additional details of LU 6.2 processing, see *SNA Format and Protocol Reference Manual: Architecture Logic for LU Type 6.2*.

This page intentionally left blank

Chapter 4

Request/Response Headers



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 4. REQUEST/RESPONSE HEADERS

INTRODUCTION

This chapter identifies the formats and meanings of the request and response headers (RH); Chapter 5.1 and Chapter 5.2 describe the request and response units (RU).

To distinguish between a request and a response, examine bit 0 in byte 0 of the RH:

If bit 0 = 0: the RH is a request header and the associated RU is a request unit.

If bit 0 = 1: the RH is a response header and any associated RU is a response unit.

Figure 4-1 on page 4-2 provides a summary of the bytes and field names in the RH.

Two message units, IPR and EXR, which make use of the RH for special purposes, are described at the end of this chapter.

Request Header

RRI =0	RU Category	r	FI	SDI	BCI	ECI	Request

Byte 0

RRI =1	RU Category	r	FI	SDI	1	1	Response

Request	DR1I	r	DR2I	ERI	r	r	QRI	PI

Byte 1

Response	DR1I	r	DR2I	RTI	r	r	QRI	PI

Request	BBI	EBI	CDI	r	CSI	EDI	PDI	CEBI

Byte 2

Response	r	r	r	r	r	r	r	r

r = Reserved

Field	Description	Explanation/Usage
RRI	Request/Response indicator	0 = request (RQ); 1 = response (RSP)
RU Category	Request/Response Unit Category	00 = FM data (FMD) 01 = network control (NC) 10 = data flow control (DFC) 11 = session control (SC)
FI	Format indicator	0 = no FM header (¬FMH), for LU-LU sessions; or character-coded without an NS header (¬NSH), for network services (NS) 1 = FM header (FMH) follows, for LU-LU sessions; or field-formatted with an NS header (NSH), for NS
SDI	Sense Data Included indicator	0 = not included (¬SD); 1 = included (SD)
BCI	Begin Chain indicator	0 = not first in chain (¬BC); 1 = first in chain (BC)

Figure 4-1. RH Formats (continued next page)

Field	Description	Explanation/Usage
ECI	End Chain indicator	0 = not last in chain (¬EC); 1 = last in chain (EC)
DR1I	Definite Response 1 indicator	0 = ¬DR1; 1 = DR1
DR2I	Definite Response 2 indicator	0 = ¬DR2; 1 = DR2
ERI	Exception Response indicator	Used in conjunction with DR1I and DR2I to indicate, in a request, the form of response requested. Values and meanings of DRI1I, DR2I, ERI are: 000 = no-response requested 100|010|110 = definite-response requested 101|011|111 = exception-response requested
RTI	Response Type indicator	0 = positive (+); 1 = negative (-)
QRI	Queued Response indicator	0 = response bypasses TC queues (¬QR); 1 = enqueue response in TC queues (QR)
PI	Pacing indicator	0 = ¬PAC; 1 = PAC
BBI	Begin Bracket indicator	0 = ¬BB; 1 = BB
EBI	End Bracket indicator	0 = ¬EB; 1 = EB (reserved for LU type 6.2)
CDI	Change Direction indicator	0 = do not change direction (¬CD); 1 = change direction (CD)
CSI	Code Selection indicator	0 = code 0; 1 = code 1
EDI	Enciphered Data indicator	0 = RU is not enciphered (¬ED); 1 = RU is enciphered (ED)
PDI	Padded Data indicator	0 = RU is not padded (¬PD); 1 = RU is padded (PD)
CEBI	Conditional End Bracket indicator	0 = not conditional end bracket (¬CEB); 1 = conditional end bracket (CEB) (used for LU type 6.2; else, reserved)

Figure 4-1. RH Formats

RH FORMATS

The request/response header (RH) is a 3-byte field; it may be a request header or a response header. The RH control fields shown in Figure 4-1 on page 4-2 are described below.

Request/Response Indicator (RRI): Denotes whether this is a request or a response.

RU Category: Denotes that the BIU belongs to one of four categories: session control (SC), network control (NC), data flow control (DFC), or function management data (FMD). (The NC category is not supported by T2.1 nodes.)

Format Indicator: Indicates which of two formats (denoted Format 1 and Format 0) is used within the associated RU (but not including the sense data field, if any; see Sense Data Included indicator, below).

For SC, NC, and DFC RUs, this indicator is always set to Format 1.

On FMD requests for SSCP-SSCP, SSCP-PU, and SSCP-LU sessions, Format 1 indicates that the request RU includes a network services (NS) header and is field-formatted (with various encodings, such as binary data or bit-significant data, in the individual fields). Format 0 indicates that no NS header is contained in the request RU and the RU is character-coded. The Format indicator value on a response is the same as on the corresponding request.

For LU-LU sessions that support FM headers on FMD requests, Format 1 indicates that an FM header is present; Format 0 implies there is no FM header. The Format indicator is always set to 0 on positive responses; negative responses are implementation dependent.

For LU-LU sessions that do not support FM headers, the meaning of this indicator on requests, positive responses, and negative responses is implementation dependent. (A BIND session parameter indicates whether FM headers are supported by the session. For further information, see Chapter 5.1 for details on BIND.)

Sense Data Included Indicator (SDI): Indicates that a 4-byte sense data field is included in the associated RU. The sense data field (when present) always immediately follows the RH and has the format and meaning described in Chapter 9. Any other data contained in the RU follows the sense data field. Sense data is included on negative responses and on EXRs, where it indicates the type of condition causing the exception.

(The Format indicator does not describe or affect the sense data, which is always in the 4-byte format shown in Chapter 9.)

Chaining Control: Indicates that a sequence of contiguous transmitted requests is being grouped in a chain. Two indicators, Begin Chain indicator (BCI) and End Chain indicator (ECI), together denote the relative position of the associated RU within a chain. The 1 values of these indicators (BCI = 1 and ECI = 1) are referred to as BC and EC, respectively.

(BC, ¬EC) = first RU in chain

(¬BC, ¬EC) = middle RU in chain

(¬BC, EC) = last RU in chain

(BC, EC) = only RU in chain

Responses are always marked "only RU in chain."

Form of Response Requested: In a request header, defines the response protocol to be executed by the request receiver.

Three bits in a request header specify the form of response that is desired. They are: Definite Response 1 indicator (DR1I), Definite Response 2 indicator (DR2I), and the Exception Response indicator (ERI). They can be coded to request:

1. No-response, which means that a response will not be issued by the half-session receiving the request. (DR1I,DR2I) = (0,0) = (¬DR1,¬DR2) and ERI=0 is the only coding possible; the abbreviation RQN refers to a request with this coding. (A special response, ISOLATED PACING RESPONSE [IPR], does set [DR1I,DR2I,ERI] = [0,0,0], but it is used independently of the other responses listed. IPR is sent in connection with session-level pacing; the sequence number in its associated TH does not correlate it to any given request.)

2. Exception response, which means that a negative response will be issued by the half-session receiving the request only in the event of a detected exception (a positive response will not be issued). (DR1I, DR2I) = (1,0)|(0,1)|(1,1) and ERI=1 are the possible codings; RQE1, RQE2, and RQE3 are the abbreviations, respectively; the abbreviation RQE or RQE* refers to a request with any of these codings.

3. Definite response, which means that a response will always be issued by the half-session receiving the request, whether the response is positive or negative. (DR1I, DR2I) = (1,0)|(0,1)|(1,1) and ERI=0 are the possible codings; RQD1, RQD2, and RQD3 are the abbreviations, respectively; the abbreviation RQD or RQD* refers to a request with any of these codings.

A request that asks for an exception response or a definite response has one or both of the DR1I and DR2I bits set to 1 (three combinations); a response to a request returns the same (DR1I, DR2I) bit combination (see Figure 4-2 on page 4-6).

The setting of the DR1I, DR2I, and ERI bits varies by RU category. In the case of LU-LU sessions (e.g., LU 6.2), BIND parameters specify the form of response to be requested during the session; Figure 4-2 on page 4-6 shows the values in table form.

For sessions that use sync point protocols with TS profile 4 (LU 6.1), RQD2 or RQE2 asks for the commitment of a unit of work that is to be shared between the session partners; RQD1 is used to request a response when the current unit of work is not to be committed. The table for this set of values is given in Figure 4-3 on page 4-8.

For nonzero, non-LU 6.2, LU types that do not use sync point protocols, the specific meanings of the DR1I and DR2I bits are defined in SNA—Sessions Between Logical Units; for LU type 0, the interpretations of the DR1I and DR2I bits (and distinctions among the three settings) are implementation-dependent.

The (DR1I, DR2I, ERI) = (0, 0, 1) combination is reserved.

REQUEST	VALID RESPONSE	MEANING OF RESPONSE
RQD1=(1,0,0) (Used by DFC)	+RSP1=(1,0,0) -RSP1=(1,0,1)	positive response negative response
RQE1=(1,0,1) (Used by DFC and PS)	implied +RSP1 -RSP1=(1,0,1)	reply received with no intervening response negative response
RQD2=(0,1,0)	+RSP2=(0,1,0) -RSP2=(0,1,1)	CONFIRMED verb issued SEND_ERROR verb issued
RQE2=(0,1,1) (Used by PS)	implied +RSP2 -RSP2=(0,1,1)	reply received with no intervening response no CONFIRMED verb issued
RQD3=(1,1,0)	+RSP3=(1,1,0) -RSP3=(1,1,1)	CONFIRMED verb issued SEND_ERROR verb issued
RQE3=(1,1,1) (Used by PS)	implied +RSP3 -RSP3=(0,1,1)	reply received with no intervening response no CONFIRMED verb issued

Notes:

1. Values displayed in this table are in the order (DR1I,DR2I,ERI) for requests and (DR1I,DR2I,RTI) for responses.
2. All ¬EC requests are sent as RQE1.
3. RQN=(0,0,0) is not used.

Figure 4-2. FMD Request/Response Combinations for Sessions between Two LU 6.2s

Queued Response Indicator (QRI): In a response header for a normal-flow RU, the Queued Response indicator denotes whether the response is to be enqueued in TC queues (QRI=QR), or whether it is to bypass these queues (QRI=¬QR). In a request header for a normal-flow RU, it indicates what the setting of the QRI should be on the response, if any, to this request (i.e., the values on the request and response are the same).

For expedited-flow RUs, this bit is reserved.

The setting of the QRI bit is the same for all RUs in a chain.

Response Type: In a response header, two basic response types can be indicated: positive response or negative response. For negative responses, the RH is always immediately followed by four bytes of sense data in the RU. Thus, RTI=NEG and RTI=POS occur jointly with SDI=SD and SDI=¬SD, respectively.

REQUEST	VALID RESPONSE	MEANING OF RESPONSE
RQD1=(1,0,0)	+RSP1=(1,0,0) -RSP1=(1,0,1)	positive response negative response
RQE1=(1,0,1)	-RSP1=(1,0,1)	negative response
RQD2=(0,1,0)	+RSP2=(0,1,0) -RSP2=(0,1,1)	positive sync point response negative sync point response
RQE2=(0,1,1)	-RSP2=(0,1,1)	negative sync point response
RQD3=(1,1,0)	+RSP3=(1,1,0) -RSP3=(1,1,1)	positive sync point response negative sync point response
RQE3=(1,1,1)	-RSP3=(1,1,1)	negative sync point response

Notes:

1. Values displayed in this table are in the order (DR1I,DR2I,ERI) for requests and (DR1I,DR2I,RTI) for responses.

2. Each definite- or exception-response chain has the same setting of (DR1I,DR2I)--either (1,0) or (0,1)--on all requests with ECI = ¬EC. When DR1I = 1 on these requests, the End-Chain request can carry (DR1I,DR2I) = (1,0)|(1,1). When DR2I = 1 on these requests, the End-Chain request can carry only (DR1I,DR2I) = (0,1). ERI is 0 only for definite-response chains and when ECI = EC.

3. RQN=(0,0,0) is not used.

Figure 4-3. Request/Response Combinations For TS Profile 4 Sync Points

Three kinds of positive and negative responses correspond to the three valid (DR1I, DR2I) combinations allowed on requests. The settings of the DR1I and DR2I bits in a response always equal the settings of the DR1I and DR2I bits of the form-of-response-requested field of the corresponding request header.

Pacing: In a request header, the Pacing Request indicator denotes that the sender can accept a Pacing Response indicator.

The Pacing Response indicator in a response header is used to indicate to the receiver that additional requests may be sent on the normal flow. The Pacing

Response indicator may be _on_ in an RH that is attached to a response RU on the normal flow; or, if desired, a separate, or isolated, response header may be used, to which no RU is attached. This latter RH signals only the pacing response; it is called an ISOLATED PACING RESPONSE (IPR); isolated and non-isolated pacing responses are functionally equivalent. IPR is discussed further in a later section of this chapter.

Bracket Control: Used to indicate the beginning or end of a group of exchanged requests and responses called a bracket. Bracket protocols are used only on LU-LU sessions. When used, BB appears on the first request in the first chain of a bracket and denotes the beginning of the bracket; the end of the bracket is indicated in one of two ways, depending on LU type.

- For LU 6.2, CEB appears on the last request of the last chain of a bracket. (When bracket usage is specified in BIND, the BIND request carries an implied BB.) The bracket indicators are set only on LUSTAT and FMD requests, and are thus sent normal-flow.

- For other LU types, the end of bracket is delimited by setting EBI to EB in the first request of the last chain in the bracket.

Change Direction Indicator (CDI): Used when there is half-duplex (HDX) control of the normal flows within a session (not to be confused with link-level HDX protocols). It permits a sending half-session to direct the receiving half-session to send. The HDX protocol is useful to half-sessions with limited input/output capabilities that cannot simultaneously send and receive user data. When used, CD appears only on the last request in a chain; it is set only on LUSTAT and FMD requests.

Code Selection Indicator (CSI): Specifies the encoding used for the associated FMD RU. When a session is activated, the half-sessions can choose to allow use of two codes in their FMD RUs (e.g., EBCDIC and ASCII), which they designate as Code 0 and Code 1. FM headers and request and response codes are not affected by the Code Selection indicator.

For SC, NC, and DFC RUs, this bit is reserved.

Enciphered Data Indicator (EDI): Indicates that information in the associated RU is enciphered under session-level cryptography protocols.

Padded Data Indicator (PDI): Indicates that the RU was padded at the end, before encipherment, to the next integral multiple of 8 bytes in length; the last byte of such padding is the count of pad bytes added, the count being a number (1-7 inclusive) in unsigned 8-bit binary representation.

IPR AND EXR

Two special message units exist in SNA: ISOLATED PACING RESPONSE (IPR) and EXCEPTION REQUEST (EXR). These are explained below.

ISOLATED PACING RESPONSE (IPR)

An IPR is used on a session if BIND specifies session-level pacing is used; it indicates a pacing response, and can be used even when operating under no-response protocols.

The following fields of the TH and RH are set for an IPR:

TH: Either the normal or expedited flow may be indicated. The sequence number is undefined (it may be set to any value, and is not checked by the receiver).

RH: An IPR is coded all 0's except for the Request-Response indicator, the Pacing indicator, and the Chain indicators, which are set to 1's; thus, the IPR RH is coded X'830100' by the sender; the receiver identifies an IPR by detecting that (RRI, DR1I, DR2I, PI) = (1, 0, 0, 1) and ignoring the remaining bits

EXCEPTION REQUEST (EXR)

Two EXR types are defined: those replacing requests, and those replacing too-long path information units (PIUs) received by transmission group control (TGC) from an upper layer (e.g., ERC in an intermediate routing node).

EXRs replacing requests are generated by some component between the origin and intended destination of a request found to be in error. The following fields are set in the TH, RH, and RU.

TH: The sequence number remains the same as in the request being replaced. The data count is altered to properly record the new BIU size. The Mapping field is set to (BBIU, EBIU); an EXR replaces a complete BIU, not just one segment of a segmented BIU. All other fields are left as received.

RH: The Sense Data Included bit is set to 1. All other fields are unchanged.

RU: Bytes 0-3 contain sense data defining the last error detected, and in the same format as returned in negative responses. The sense data is followed by the original RU, truncated to no more than three bytes, as described for negative responses.

EXRs replacing too-long PIUs are formatted as follows.

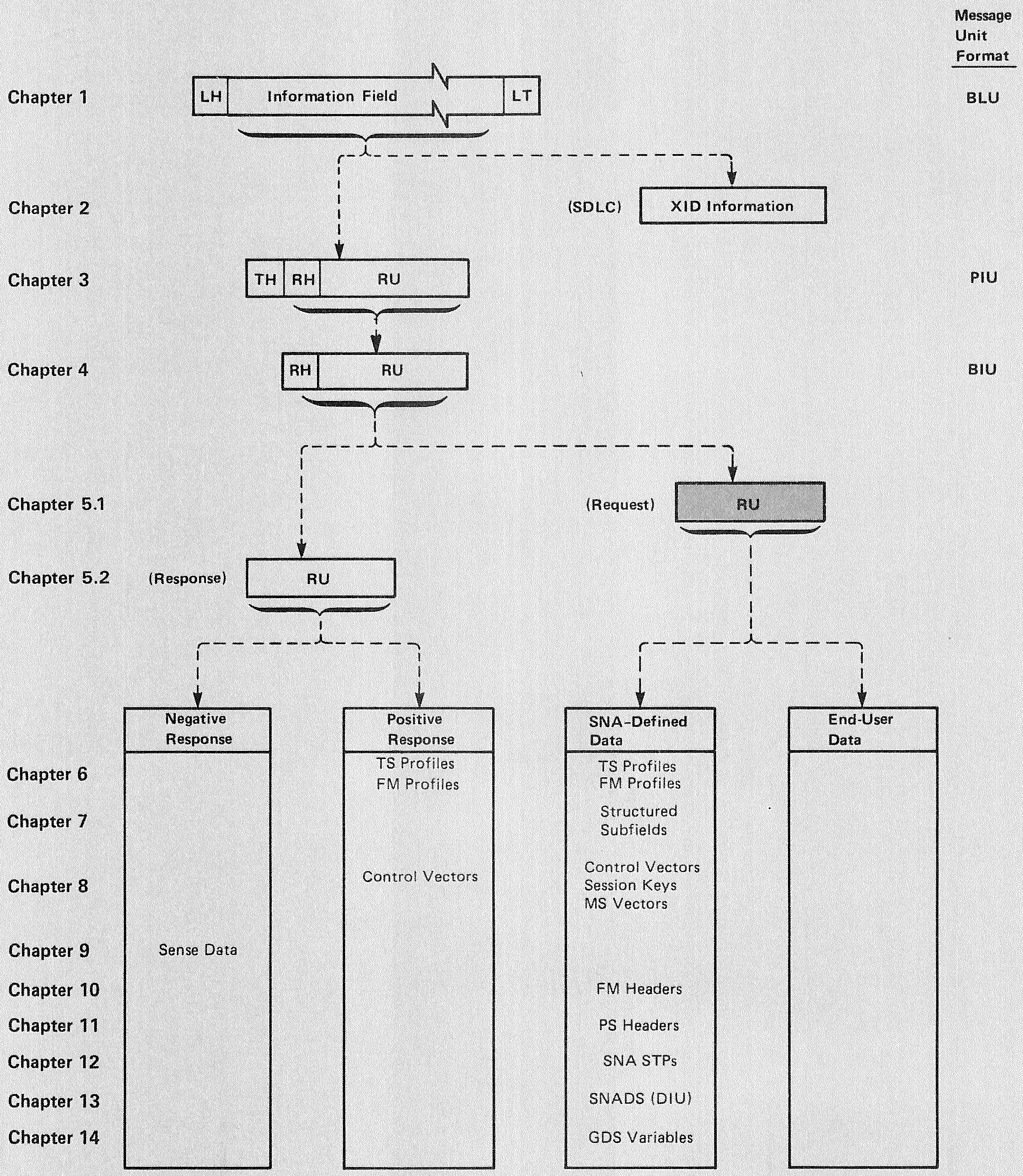
TH: Like EXRs replacing requests, EXRs replacing too-long PIUs change only the Mapping field (to 1's) and the data count (to 7 in this case).

RH: The RH is set to X'07B000', no matter what the replaced RH (if any) was.

RU: Bytes 0-3 contain the sense data X'800A'; no other bytes are included.

Chapter 5.1

Request Units



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 5.1. REQUEST UNITS

INTRODUCTION TO REQUEST UNITS

This section contains detailed formats of the request units, arranged in alphabetical order. Each format description begins with the following heading:

"ABBREVIATED RU NAME; Origin-NAU-->Destination-NAU, Normal (Norm) or Expedited (Exp) Flow; RU Category (RU NAME)"

<u>Notes:</u>

1. "RU Category" is abbreviated as follows:

DFC	data flow control
SC	session control
NC	network control
FMD NS(ma)	function management data, network services, management services (note: formerly maintenence services)
FMD NS(s)	function management data, network services, session services

2. The formats of character-coded FMD NS requests are implementation dependent. LU-->LU FMD requests (e.g., FM headers) are described in Chapter 10 .

3. All values for field-formatted requests that are not defined in this section are reserved.

4. The request-code value X'FF' and the NS-header values X'(3|7|B|F)F****' and X'**(3|7|B|F)F**' are set aside for implementation internal use, and will not be otherwise defined in SNA.

5. Throughout the format descriptions, <u>reserved</u> is used as follows: reserved bits, or fields, are ones that currently are set to 0's (unless explicitly stated otherwise); reserved values are those that currently are invalid. Correct usage of reserved fields is enforced by the sender; no receive checks are made on these fields.

6. Throughout the format descriptions, <u>retired</u> fields and values are those that were once defined in SNA but are no longer defined. To accommodate implementations of back-level SNA, current implementations of SNA treat retired fields as follows: send checks enforce the setting of retired fields to all 0's except where other unique values are required (described individually); no receive checks are made on these fields, thereby accepting back-level settings

of these fields. Special handling of retired fields, such as echoing or passing on retired fields as received, is discussed where appropriate.

7. User data, control vectors, and session keys referred to in the format descriptions are described in Chapter 7 and Chapter 8.

REQUEST UNIT SUMMARY INFORMATION

The following is a categorized list of RU abbreviations, followed by a list of RUs indexed by NS headers and request codes.

SUMMARY OF REQUEST RUS BY CATEGORY

Request RUs prefixed by an asterisk (*) require response RUs that, if positive, have an extended format containing data in addition to the NS header or request code.

SC Requests

*ACTLU
*ACTPU
*BIND
CLEAR
CRV
DACTLU
DACTPU
RQR
SDT
*STSN
UNBIND

DFC Requests

BID
BIS
CANCEL
CHASE
LUSTAT
QC
QEC
RELQ
RSHUTD
RTR
SBI
SHUTC
SHUTD
SIG

FMD NS(ma) Requests

NMVT

FMD NS(s) Requests

INIT-SELF
NOTIFY
TERM-SELF

INDEX OF RU'S BY NS HEADERS AND REQUEST CODES

Within DFC, NC, SC, or any specific FMD NS category, the request code is unique. However, while a request code has only one meaning in a specific category, a given code (e.g., X'05') can represent different requests in separate categories (e.g., DFC, NC, and configuration services).

FMD NS Headers (third byte is the request code)

X'010681'	INIT-SELF (Format 0)
X'010683'	TERM-SELF (Format 0)
X'41038D'	NMVT
X'810620'	NOTIFY
X'810681'	INIT-SELF (Format 1)
X'810683'	TERM-SELF (Format 1)

DFC, NC, and SC Request Codes

X'04'	LUSTAT (DFC)
X'05'	RTR (DFC)
X'0D'	ACTLU (SC)
X'0E'	DACTLU (SC)
X'11'	ACTPU (SC)
X'12'	DACTPU (SC)
X'31'	BIND (SC)
X'32'	UNBIND (SC)
X'70'	BIS (DFC)
X'71'	SBI (DFC)
X'80'	QEC (DFC)
X'81'	QC (DFC)
X'82'	RELQ (DFC)
X'83'	CANCEL (DFC)
X'84'	CHASE (DFC)
X'A0'	SDT (SC)
X'A1'	CLEAR (SC)
X'A2'	STSN (SC)
X'A3'	RQR (SC)
X'C0'	SHUTD (DFC)
X'C0'	CRV (SC)
X'C1'	SHUTC (DFC)
X'C2'	RSHUTD (DFC)
X'C8'	BID (DFC)
X'C9'	SIG (DFC)

DESCRIPTIONS OF REQUEST UNITS

ACTLU; SSCP-->LU, Exp; SC (ACTIVATE LOGICAL UNIT)

> ACTLU is sent from an SSCP to an LU to activate a session between the SSCP and the LU and to establish common session parameters.

0 X'0D' request code

1 Type activation requested:
- X'01' cold
- X'02' ERP

2 bits 0-3, FM profile:
  - X'0' FM profile 0
  - X'6' FM profile 6

bits 4-7, TS profile:
  - X'1' TS profile 1 (only value defined)

ACTPU; SSCP|PUCP-->PU, Exp; SC (ACTIVATE PHYSICAL UNIT)

> ACTPU is sent by the SSCP to activate a session with the PU, and to obtain certain information about the PU.

0 X'11' request code

1 bits 0-3, format:
  - X'0' Format 0

bits 4-7, type activation requested:
  - X'1' cold
  - X'2' ERP

2 bits 0-3, FM profile:
  - X'0' FM profile 0

bits 4-7, TS profile:
  - X'1' TS profile 1

3-8 A six-byte field that specifies the ID of the SSCP issuing ACTPU; the first four bits specify the format for the remaining bits:
bits 0-3, format: 0000 (only value defined)
bits 4-7, PU type of the node containing the CP
bits 8-47, implementation and installation dependent binary identification

BID; LU-->LU, Norm; DFC (BID)

> BID is used by the bidder to request permission to initiate a bracket, and is used only when using brackets. This RU is not used for LU 6.2.

0 X'C8' request code

BIND; PLU-->SLU, Exp; SC (BIND SESSION)

BIND is sent from a primary LU to a secondary LU to activate a session between the LUs. The secondary LU uses the BIND parameters to help determine whether it will respond positively or negatively to BIND.

0 X'31' request code

1 bits 0-3, format: 0000 (only value defined)

bits 4-7, type:
- 0000 negotiable (only value defined for LU 6.2)
- 0001 nonnegotiable

2 FM profile:
- X'02' FM profile 2
- X'03' FM profile 3
- X'04' FM profile 4
- X'07' FM profile 7
- X'12' FM profile 18
- X'13' FM profile 19 (only value defined for LU 6.2)

3 TS profile:
- X'02' TS profile 2
- X'03' TS profile 3
- X'04' TS profile 4
- X'07' TS profile 7 (only value defined for LU 6.2)

FM Usage—Primary LU Protocols for FM Data

4 bit 0, chaining use selection:
- 0 only single-RU chains allowed from primary LU half-session
- 1 multiple-RU chains allowed from primary LU half-session (only value defined for LU 6.2)

bit 1, request control mode selection:
- 0 immediate request mode (only value defined for LU 6.2)
- 1 delayed request mode

bits 2-3, chain response protocol used by primary LU half-session for FMD requests; chains from primary will ask for:
- 00 no response
- 01 exception response
- 10 definite response
- 11 definite or exception response (only value defined for LU 6.2)

bit 4, 2-phase commit for sync point (reserved if any TS profile other than 4):
- 0 2-phase commit not supported
- 1 2-phase commit supported

bit 5, reserved

bit 6, compression indicator (reserved for LU 6.2):
- 0 compression will not be used on requests from primary
- 1 compression may be used

bit 7, send End Bracket indicator:
- 0 primary will not send EB (only value defined for LU 6.2)
- 1 primary may send EB

FM Usage—Secondary LU Protocols for FM Data

5 bit 0, chaining use selection:
- 0 only single-RU chains allowed from secondary LU half-session

- 1 multiple-RU chains allowed from secondary LU half-session (only value defined for LU 6.2)

bit 1, request control mode selection:
- 0 immediate request mode (only value defined for LU 6.2)
- 1 delayed request mode

bits 2-3, chain response protocol used by secondary LU half-session for FMD requests; chains from secondary will ask for:
- 00 no response
- 01 exception response
- 10 definite response
- 11 definite or exception response (only value defined for LU 6.2)

bit 4, 2-phase commit for sync point (reserved if any TS profile other than 4):
- 0 2-phase commit not supported
- 1 2-phase commit supported

bit 5, reserved

bit 6, compression indicator (reserved for LU 6.2):
- 0 compression will not be used on requests from secondary
- 1 compression may be used

bit 7, send End Bracket indicator:
- 0 secondary will not send EB (only value defined for LU 6.2)
- 1 secondary may send EB

<u>FM Usage—Common LU Protocols</u>

6 bit 0, whole-BIUs required indicator:
- 0 the sending node supports receipt of segments on this session
- 1 the sending node does not support receipt of segments on this session; the maximum sent-RU size specified in bytes 10 and 11 of BIND and RSP(BIND) are negotiated so that BIUs on this session are not segmented when sent to a node requiring whole BIUs

bit 1, FM header usage:
- 0 FM headers not allowed
- 1 FM headers allowed (only value defined for LU 6.2)

bit 2, brackets usage and reset state:
- 0 brackets not used if neither primary nor secondary will send EB, i.e., if byte 4, bit 7 = 0 and byte 5, bit 7 = 0; brackets are used and bracket state managers' reset states are INB (1) if either primary or secondary, or both, may send EB, i.e., if byte 4, bit 7 = 1 or byte 5, bit 7 = 1; or (2) if FM profile 19 is specified (only value defined for LU 6.2)
- 0 brackets are used and bracket state managers' reset states are INB
- 1 brackets are used and bracket state managers' reset states are BETB

bit 3, bracket termination rule selection (reserved if brackets not used, i.e., if byte 6, bit 2 = 0, byte 4, bit 7 = 0, and byte 5, bit 7 = 0; and if FM profile is not 19):
- 0 Rule 2 (unconditional termination) will be used during this session
- 1 Rule 1 (conditional termination) will be used during this session (only value defined for LU 6.2)

bit 4, alternate code set allowed indicator:
- 0 alternate code set will not be used

1 alternate code set may be used

bit 5, sequence number availability for sync point resynchronization (reserved if any TS profile other than 4 is used):
- 0 sequence numbers not available
- 1 sequence numbers available

  Note: Sequence numbers are transaction processing program sequence numbers from the previous activation of the session with the same session name; they are associated with the last acknowledged requests and any pending requests to commit a unit of work. If no previous activation existed, the numbers are 0, and this bit is set to 0.

bit 6, BIS sent (reserved for TS profiles other than 4):
- 0 BIS not sent
- 1 BIS sent

bit 7, BIND queuing indicator:
- 0 BIND cannot be queued (held, pending resource availability, thus delaying the BIND response)
- 1 BIND sender allows the BIND receiver to queue the BIND for an indefinite period, thus delaying the sending of the BIND response

  Note: BIND sender may provide a timer or operator interface to send UNBIND if session-activation time exceeds BIND sender's implementation-defined limits. BIND queuing is terminated by sending UNBIND to the BIND receiver.

7 bits 0-1, normal-flow send/receive mode selection:
- 00 full-duplex
- 01 half-duplex contention
- 10 half-duplex flip-flop (only value defined for LU 6.2)
- 11 reserved

bit 2, recovery responsibility (reserved if normal flow send/receive mode is FDX, i.e., if byte 7, bits 0-1 = 00):
- 0 contention loser responsible for recovery (see byte 7, bit 3 for specification of which half-session is the contention loser)
- 1 symmetric responsibility for recovery (only value defined for LU 6.2)

bit 3, contention winner/loser (reserved if normal flow send/receive mode is FDX, i.e., if byte 7, bits 0-1 = 00; or if the normal flow send/receive mode is HDX-FF, brackets are not used, FM profile is not 19, and symmetric responsibility for recovery is used, i.e., if byte 7, bits 0-1 = 10, byte 4, bit 7 = 0, byte 5, bit 7 = 0, byte 6, bit 2 = 0, and byte 7, bit 2 = 1):
- 0 secondary is contention winner and primary is contention loser
- 1 primary is contention winner and secondary is contention loser

Note: Contention winner is also brackets first speaker if brackets are used.

Note: Contention winner is also brackets first speaker.

bits 4-5, alternate code processing identifier (reserved unless Alternate Code Set Allowed indicator (byte 6, bit 4) is 1):
- 00 process alternate code FMD RUs as ASCII-7
- 01 process alternate code FMD RUs as ASCII-8 (only value defined for LU 6.2)

  Note: When the Alternate Code Processing Identifier indicator is set to the value 01, the entire FMD request RU is to be translated using the transforms defined by the ANSI X3.26 Hollerith Card Code.

bit 6, control vectors included indicator:
- 0 control vectors are not included after the SLU name (bytes r+1-s)
- 1 control vectors are included after the SLU name (bytes r+1-s)

bit 7, half-duplex flip-flop reset states (reserved unless (1) normal-flow send/receive mode is half-duplex flip-flop (byte 7, bits 0-1 = 10) and (2) brackets are not used or bracket state manager's reset state is INB (byte 6, bit 2 = 0)):
- 0 HDX-FF reset state is RECEIVE for the primary and SEND for the secondary (e.g., the secondary sends normal-flow requests first after session activation)
- 1 HDX-FF reset state is SEND for the primary and RECEIVE for the secondary (e.g., the primary sends normal-flow requests first after session activation) (only value defined for LU 6.2)

<u>TS Usage</u>

8 bit 0, staging indicator for session-level pacing of the secondary-to-primary normal flow:
- 0 the secondary send window size (byte 8, bits 2-7) and the primary receive window size (byte 13, bits 2-7) are for one-stage pacing (The secondary send window size is always equal to the primary receive window size.)
- 1 the secondary send window size (byte 8, bits 2-7) and the primary receive window size (byte 13, bits 2-7) are for two-stage pacing

<u>Note:</u> The meanings of 0 and 1 are reversed from the corresponding staging indicator for the primary-to-secondary normal flow.

bit 1, reserved

bits 2-7, secondary send window size, in binary, for session-level pacing

9 bits 0-1, reserved

bits 2-7, secondary receive window size, in binary, for session-level pacing

10 Maximum RU size sent on the normal flow by the secondary half-session: if bit 0 is set to 0, no maximum is specified and the remaining bits 1-7 are ignored; if bit 0 is set to 1, bit 0 is set to 1, and the byte is interpreted as X'ab' = a•2**b (Notice that, by definition, a≥8 and therefore X'ab' is a normalized floating point representation.) See Figure 5.1-1 on page 5.1-13 for all possible values.

11 Maximum RU size sent on the normal flow by the primary half-session: identical encoding as described for byte 10

12 bit 0, staging indicator for session-level pacing of the primary-to-secondary normal flow:
- 0 the primary send window size (byte 12, bits 2-7) and the secondary receive window size (byte 9, bits 2-7) are for two-stage pacing
- 1 the primary send window size (byte 12, bits 2-7) and the secondary receive window size (byte 9, bits 2-7) are for one-stage pacing (The primary send window size is always equal to the secondary receive window size.)

<u>Note:</u> The meanings of 0 and 1 are reversed from the corresponding staging indicator for the secondary-to-primary normal flow (byte 8, bit 0).

bit 1, reserved

	bits 2-7, primary send window size, in binary, for session-level pacing
13	bits 0-1, reserved bits 2-7, primary receive window size, in binary, for session-level pacing
	PS Profile
14	bit 0, PS Usage field format: 0 basic format (only value defined) bits 1-7, LU type: 0000000 LU type 0 0000001 LU type 1 0000010 LU type 2 0000011 LU type 3 0000100 LU type 4 0000110 LU type 6 0000111 LU type 7
	PS Usage characteristics Note: The following format for bytes 15-25 applies only to LU 6.2; for information on PS usage bytes 15-25 for other than LU 6.2 (indicated by byte 14, bits 1-7 = 0000110 and byte 15 = 00000010), see SNA—Sessions Between Logical Units.
15	LU-6 level: X'02' Level 2 (i.e., LU 6.2)
16-22	Reserved
23	bits 0-2, retired bit 3, conversation-level security support: 0 Access Security Information field will not be accepted on incoming FMH-5s 1 Access Security Information field will be accepted on incoming FMH-5s bits 4-5, reserved bit 6, already-verified function support: 0 Already Verified indicator will not be accepted on incoming FMH-5s 1 Already Verified indicator will be accepted on incoming FMH-5s bit 7, reserved
24	bit 0, reserved bits 1-2, synchronization level: 01 confirm is supported 10 confirm, sync point, and backout are supported bit 3, reserved bits 4-5, responsibility for session reinitiation (reserved unless bit 6 of this byte is set to 0): 00 operator controlled 01 primary half-session will reinitiate 10 secondary half-session will reinitiate 11 either may reinitiate bit 6, parallel session support for LU-LU pair: 0 not supported 1 supported bit 7, Change Number of Sessions GDS variable flow support (set to 1 if byte 24, bit 6 = 1): 0 not supported 1 supported
25	Reserved End of PS Usage Field

26-k Cryptography Options

26 bits 0-1, private cryptography options (reserved for LU 6.2):
  - 00 no private cryptography supported
  - 01 private cryptography supported: the session cryptography key and cryptography protocols are privately supplied by the end user

bits 2-3, session-level cryptography options:
  - 00 no session-level cryptography supported
  - 01 session-level selective cryptography supported; all cryptography key management is supported by the SSCP and LU; exchange (via +RSP(BIND)) and verification (via CRV) of the cryptography session-seed value is supported by the LUs for the session; all FMD requests carrying ED are enciphered/deciphered by the TCs
  - 10 reserved
  - 11 session-level mandatory cryptography supported; all cryptography key management is supported by the SSCP and LU; exchange (via +RSP(BIND)) and verification (via CRV) of the cryptography session-seed value is supported by the LUs for the session; all FMD requests are enciphered/deciphered by TC

Note: Only values 00 and 11 are defined for LU 6.2.

bits 4-7, session-level cryptography options field length:
  - X'0' no session-level cryptography specified; following additional cryptography options fields (bytes 27-k) omitted
  - X'9' session-level cryptography specified; additional options follow in next nine bytes

27 bits 0-1, session cryptography key encipherment method:
  - 00 session cryptography key enciphered under SLU master cryptography key using a seed value of 0 (only value defined)

bits 2-4, reserved

bits 5-7, cryptography cipher method:
  - 000 block chaining with seed and cipher text feedback, using the Data Encryption Standard (DES) algorithm (only value defined)

28-k Session cryptography key enciphered under secondary LU master cryptography key; an eight-byte value that, when deciphered, yields the session cryptography key used for enciphering and deciphering FMD requests

k+1-m Primary LU Name Field (always present)

k+1 Length of primary LU name (values 1 to 17 are valid)

Note: Value 0 is retired.

k+2-m Primary LU name or, if the secondary LU issued the INIT-SELF (or INIT-OTHER), INIT-SELF, the uninterpreted name as carried in that RU (and also in CDINIT for a cross-domain session)

Note: The PLU name is network-qualified (i.e., has a network identifier present) if the SLU name in bytes p+2-r is network-qualified.

m+1-n User Data Field

m+1 Length of user data

Note: X'00' = no User Data field present; if unstructured user data present, values 1 to 65 are valid.

m+2-n User data

m+2 User data key:

X'00' structured subfields follow (only value defined for LU 6.2)

  Note: Individual structured subfields may be omitted entirely. When present, they appear in ascending subfield-number order.

	¬X'00' first byte of unstructured user data
	• For unstructured user data:
m+3-n	Remainder of unstructured user data
	• For structured user data:
m+3-n	Structured subfields (For detailed definitions, see "Chapter 7. User Data" in Chapter 7.)
n+1-p	User Request Correlation Field (present only if carried in INIT from SLU, or if Secondary LU name field or control vectors are included)
n+1	Length of user request correlation (URC) field (values 0 to 12 are valid) Note: X'00' = no URC present.
n+2-p	URC: LU-defined identifier (present only if carried in INIT from SLU)
p+1-r	Secondary LU Name Field (present only for negotiable BINDs and for non-negotiable BINDs that include control vectors)
p+1	Length of secondary LU name (values 1 to 17 are valid) Note: Value 0 is retired.
p+2-r	Secondary LU name

Note 1: The length of the BIND RU cannot exceed 256.

Note 2: If the last byte of a format 0 request not having control vectors is a length field and that field is 0, that byte may be omitted from the BIND request.

	Mantissa (a)							
Exponent (b)	8	9	A (10)	B (11)	C (12)	D (13)	E (14)	F (15)
0	8	9	10	11	12	13	14	15
1	16	18	20	22	24	26	28	30
2	32	36	40	44	48	52	56	60
3	64	72	80	88	96	104	112	120
4	128	144	160	176	192	208	224	240
5	256	288	320	352	384	416	448	480
6	512	576	640	704	768	832	896	960
7	1024	1152	1280	1408	1536	1664	1792	1920
8	2048	2304	2560	2816	3072	3328	3584	3840
9	4096	4608	5120	5632	6144	6656	7168	7680
A (10)	8192	9216	10240	11264	12288	13312	14336	15360
B (11)	16384	18432	20480	22528	24576	26624	28672	30720
C (12)	32768	36864	40960	45056	49152	53248	57344	61440
D (13)	65536	73728	81920	90112	98304	106496	114688	122880
E (14)	131072	147456	163840	180224	196608	212992	229376	245760
F (15)	262144	294912	327680	360448	393216	425984	458752	491520

Note: A value of X'ab' in byte 10 or byte 11 of BIND represents a•2**b. For example, X'C5' represents (in decimal) 12•2**5 = 384.

Figure 5.1-1. RU Sizes Corresponding to Values X'ab' in BIND

BIS; LU-->LU, Norm; DFC (BRACKET INITIATION STOPPED)

> BIS is sent by a half-session to indicate that it will not attempt to begin any more brackets.

0 X'70' request code

CANCEL; LU-->LU, Norm; DFC (CANCEL)

> CANCEL may be sent by a half-session to terminate a partially sent chain of FMD requests. CANCEL may be sent only when a chain is in process. The sending half-session may send CANCEL to end a partially sent chain if a negative response is received for a request in the chain, or for some other reason. This RU is not used for LU 6.2.

0 X'83' request code

CHASE; LU-->LU, Norm; DFC (CHASE)

> CHASE is sent by a half-session to request the receiving half-session to return all outstanding normal-flow responses to requests previously received from the issuer of CHASE. The receiver of CHASE sends the response to CHASE after processing (and sending any necessary responses to) all requests received before the CHASE. This RU is not used for LU 6.2.

0 X'84' request code

CLEAR; PLU-->SLU, Exp; SC (CLEAR)

> CLEAR is sent by primary session control to reset the data traffic FSMs and subtrees (for example, brackets, pacing, sequence numbers) in the primary and secondary half-sessions (and boundary function, if any). This RU is not used for LU 6.2.

0 X'A1' request code

CRV; PLU-->SLU, Exp; SC (CRYPTOGRAPHY VERIFICATION)

> CRV, a valid request only when session-level cryptography was selected in BIND, is sent by the primary LU session control to verify cryptography security and thereby enable sending and receiving of FMD requests by both half-sessions.

0 X'C0' request code

1-8 A transform of the (deciphered) cryptography session-seed value received (enciphered) in bytes 28-k of +RSP(BIND), re-enciphered under the session cryptography key using a seed value of 0; the transform is the cryptography session-seed value with the first four bytes inverted

Note: The cryptography session-seed is used as the seed for all session-level cryptography encipherment and decipherment provided for FMD RUs.

DACTLU; SSCP-->LU, Exp; SC (DEACTIVATE LOGICAL UNIT)

> DACTLU is sent to deactivate the session between the SSCP and the LU.

0 X'0E' request code
Note: End of short (one-byte) request

1 Type of deactivation requested:
X'01' normal deactivation
X'03' session outage notification (SON)
2 Cause (reserved if byte 1 ≠ X'03'):
X'07' virtual route inoperative: the virtual route serving the SSCP-LU session has become inoperative, thus forcing the deactivation of the session
X'08' route extension inoperative: the route extension serving the SSCP-LU session has become inoperative, thus forcing the deactivation of the session
X'09' hierarchical reset: the identified session is being deactivated because of a +RSP(ACTPU, Cold)
X'0B' virtual route deactivated: the SSCP-LU session is being deactivated because of a forced deactivation of the virtual route being used by the session
X'0C' SSCP or LU failure—unrecoverable: the SSCP-LU session had to be reset because of an abnormal termination; recovery from the failure was not possible
X'0D' session override: the SSCP-LU session has to be deactivated because of a more recent session activation request for the SSCP to subarea PU session over a different virtual route
X'0E' SSCP or LU failure—recoverable: the SSCP-LU session had to be deactivated because of an abnormal termination of the SSCP or LU of the session; recovery from the failure may be possible
X'0F' cleanup: the SSCP is resetting its half-session before receiving the response from the LU being deactivated

DACTPU; SSCP|PUCP-->PU, PU-->SSCP, Exp; SC (DEACTIVATE PHYSICAL UNIT)

> DACTPU is sent to deactivate the session between the SSCP and the PU.

0 X'12' request code
1 Type deactivation requested:
X'01' final use, physical connection may be broken
X'02' not final use, physical connection should not be broken
X'03' session outage notification (SON)
2 Cause (not present if byte 1 ≠ X'03'):
X'07' virtual route inoperative: the virtual route for the SSCP-PU session has become inoperative, thus forcing the deactivation of the SSCP-PU session

X'08' route extension inoperative: the route extension serving the SSCP-PU session has become inoperative, thus forcing the deactivation of the SSCP-PU session

X'09' hierarchical reset: the identified session is being deactivated because of a +RSP(ACTPU, Cold)

X'0B' virtual route deactivated: the identified SSCP-PU session is being deactivated because of a forced deactivation of the virtual route being used by the session

X'0C' SSCP or PU failure—unrecoverable: the identified SSCP-PU session had to be deactivated because of an abnormal termination of the SSCP or PU of the session; recovery from the failure was not possible

X'0D' session override: the SSCP-PU session has to be deactivated because of a more recent session activation request for the SSCP to subarea PU session over a different virtual route

X'0E' SSCP or PU failure—recoverable: the identified SSCP-PU session had to be deactivated because of an abnormal termination of the SSCP or PU of the session; recovery from the failure may be possible

X'0F' cleanup: the SSCP is resetting its half-session before receiving the response from the PU that is being deactivated.

INIT-SELF Format 0; ILU-->SSCP, Norm; FMD NS(s) (INITIATE-SELF)

> INIT-SELF from the ILU requests that the SSCP authorize and assist in the initiation of a session between the LU sending the request (that is, the ILU, which also becomes the OLU) and the LU named in the request (the DLU). This RU is not used for LU 6.2; refer to INIT-SELF Format 1.

0-2 X'010681' NS header

3 bits 0-3, format:
- 0000 Format 0: specifies a subset of the parameters shown in Format 1 of INIT-SELF (described separately, because the NS header differs in the first byte), with the receiver supplying default values

bit 4, reserved

bits 5-6,
- 00 DLU is PLU
- 01 DLU is SLU

bit 7,
- 0 initiate only (I): do not enqueue.
- 1 initiate/enqueue (I/Q): enqueue the request if it cannot be satisfied immediately

4-11 Mode name: an eight-character symbolic name (implementation and installation dependent) that identifies the set of rules and protocols to be used for the session; used by the SSCP(SLU) to select the BIND image that will be used by the SSCP(PLU) to build the CINIT request

12-m Uninterpreted Name of DLU

12 Type: X'F3' logical unit

13 Length, in binary, of DLU name

14-m EBCDIC character string

m+1-m+2 Retired

m+3-n User Field

m+3 Length, in binary, of user data

Note: X'00' = no user data is present.

m+4-n User data: user-specific data that is passed to the primary LU on the CINIT request

m+4 User data key
X'00' structured subfields follow
¬X'00' first byte of unstructured user data
Note: Individual structured subfields may be omitted entirely. When present, they appear in ascending field number order.

- For unstructured user data

m+5-n Remainder of unstructured user data

- For structured user data

m+5-n Structured subfields (For detailed definitions, see "Chapter 7. User Data" in Chapter 7.)

Note: The following default values are supplied by the SSCP(ILU) receiving the Format 0 INIT-SELF request:

- Queuing conditions (if queuing is specified):
  - Enqueue if session limit exceeded.
  - Enqueue this request FIFO.
- Initiate origin: network user is the initiator.
- NOTIFY: do not notify

INIT-SELF Format 1 and 2; ILU-->SSCP, Norm; FMD NS(s) (INITIATE-SELF)

> INIT-SELF from the ILU requests that the SSCP authorize and assist in the initiation of a session between the LU sending the request (that is, the ILU, which also becomes the OLU) and the LU named in the request (the DLU).

0-2 X'810681' NS header
3 bits 0-3, format:
0001 Format 1: specifies queuing, initiate origin, NOTIFY, and URC in addition to the parameters in Format 0 (only value defined for LU 6.2)
0010 Format 2: specifies the COS name field in addition to the parameters in Format 1
bits 4-7, reserved
4 Type:
bits 0-1, 00 dequeue (DQ) a previously enqueued initiate request (See bits 2-3 for further specification of setup actions.)
Note: Value 00 is defined only for Format 1.
01 initiate only (I): do not enqueue
10 enqueue only (Q) (See byte 5 for further specification of queuing conditions.)
11 initiate/enqueue (I/Q): enqueue the request if it cannot be satisfied immediately
Note: Only values 01 and 11 are defined for LU 6.2.
bits 2-3, (used for DQ; otherwise, reserved)
00 leave on queue if setup attempt is unsuccessful
01 remove from queue if setup attempt is unsuccessful
10 remove from queue; do not attempt setup
11 reserved
bit 4, reserved

bits 5-6, PLU/SLU specification:
00 DLU is PLU
01 DLU is SLU
bit 7, reserved

5 Queuing conditions for DLU (reserved when Type = DQ):
bit 0, 0 do not enqueue if session limit exceeded
1 enqueue if session limit exceeded
bit 1, 0 do not enqueue if DLU is not currently able to comply with the PLU/SLU specification (as given in byte 4, bits 5-6)
1 enqueue if DLU is not currently able to comply with the PLU/SLU specification
bit 2, reserved
bit 3, reserved for LU 6.2; otherwise:
0 do not enqueue if no SSCP(DLU)-DLU path
1 enqueue if no SSCP(DLU)-DLU path
bit 4, reserved
bits 5-6, queuing position/service:
00 put this request at the bottom of the queue (the request is put at the bottom of the queue and serviced last)
01 enqueue this request FIFO (only value defined for LU 6.2)
10 enqueue this request LIFO
11 reserved
bit 7, reserved

Note: Since queuing conditions are specified for the DLU only, the following default values are used by SSCP(OLU) for the OLU:

- Enqueue if session limit exceeded.
- Enqueue this request at the foot of the queue (FIFO).

6 Reserved for LU 6.2; otherwise:
bits 0-2, reserved
bit 3, initiator (reserved when Type = DQ):
0 network user is the initiator
1 network manager is the initiator
bits 4-7, reserved

7 NOTIFY specifications (reserved when Type = DQ and for LU 6.2):
bits 0-1, NOTIFY (Resource Requested) conditions:
00 do not notify LUs in session with DLU
01 reserved
10 notify LUs in session with DLU only if request is queued
11 reserved
bits 2-7, reserved

8-15 Mode name: an eight-character symbolic name (implementation and installation dependent) that identifies the set of rules and protocols to be used for the session; used by the SSCP(SLU) to select the BIND image that will be used by the SSCP(PLU) to build the CINIT request (reserved when Type = DQ)

16-n Uninterpreted Name of DLU
16 Type: X'F3' logical unit
17 Length, in binary, of DLU name
18-n DLU name EBCDIC character string
n+1-n+2 Retired
n+3-r User Field (reserved when Type = DQ and for LU 6.2)
n+3 Length, in binary, of user data

Note: X'00' = no user data is present.

n+4-r User data: user-specific data that is passed to the primary LU on the CINIT request

n+4 User data key

X'00' structured subfields follow

¬X'00' first byte of unstructured user data

Note: Individual structured subfields may be omitted entirely. When present, they appear in ascending field number order.

- For unstructured user data

n+5-r Remainder of unstructured user data

- For structured user data

n+5-r Structured subfields (For detailed definitions, see "Chapter 7. User Data" in Chapter 7.)

r+1-s User Request Correlation (URC) Field (When Type = DQ, the URC must be the same as in the original INIT-SELF request.)

r+1 Length, in binary, of URC

Note: X'00' = no URC. (The length field is always present.)

r+2-s URC: LU-defined identifier; this value can be returned by the SSCP in a subsequent NOTIFY to correlate a given session to this initiating request

Note: End of Format 1; Format 2 Continues

s+1-s+8 COS name: symbolic name of class of service in EBCDIC characters (A value of eight space characters may be specified; in this case, the COS name is derived from the mode name table using the mode name received in bytes 8-15.)

LUSTAT; LU-->LU|SSCP, Norm; DFC (LOGICAL UNIT STATUS)

> LUSTAT is used by one half-session to send up to four bytes of status information to its paired half-session. The RU format allows the sending of either end-user information or LU status information. If the high-order two bytes of the status information are 0, the low-order two bytes carry end-user information and may be set to any value. In general, LUSTAT is used to report about failures and error recovery conditions for a local device of an LU.

0 X'04' request code

1-4 Status value + status extension field (two bytes each):

X'0000'+'uuuu' user status (no system-defined status) + user-defined field

X'0001'+'ccdd' component now available + component identification (see Note)

X'0002'+'rrrr' sender will have no (more) FMD requests to transmit during the time that this session remains active + reserved field

X'0003'+'ccdd' component entering attended mode of operation + component identification (see Note)

X'0004'+'ccdd' component entering unattended mode of operation + component identification (see Note)

X'0005'+'iiii' prepare to commit all resources required for the unit of work + information field:

X'0001' request End Bracket be sent on next chain (only value defined)

X'0006'+'rrrr'	no-op (used to allow an RH to be sent when no other request is available or allowed) + reserved field (only value defined for LU 6.2)
X'0007'+'rrrr'	sender currently has no FMD requests to transmit (but may have later during the time that this session remains active) + reserved field
X'0801'+'ccdd'	component not available (e.g., not configured) + component identification (see Note)
X'0802'+'ccdd'	component failure (intervention required) + component identification (see Note)
X'081C'+'ccdd'	component failure (permanent error) + component identification (see Note)
X'0824'+'rrrr'	function canceled + reserved field
X'082B'+'ccdd'	component available, but presentation space integrity lost + component identification (see Note)
X'0831'+'ccdd'	component disconnected (power off or some other disconnecting condition) + component identification (see Note)
X'0848'+'rrrr'	cryptography component failure + reserved field
X'400A'+'ssss'	no-response mode not allowed + sequence number of the request specifying no-response

Note: Values for cc byte are:

X'00'	LU itself rather than a specific LU component (For this cc value, dd=X'00'.)
X'FF'	The dd byte specifies the LU component medium class and device address. (See *SNA—Sessions Between Logical Units* for definitions of these terms and usage of the values according to LU type.)
¬X'(00|FF)'	LU component medium class and device address (For these cc values, dd=X'00'.)

NMVT; SSCP<-->PU_T2 Norm; FMD NS(ma) (NETWORK MANAGEMENT VECTOR TRANSPORT)

> NMVT carries management services (MS) requests and replies between a CP and a PU.

0-2	X'41038D' NS header
3-4	Retired: set to network address by subarea node sender; set to 0, the PU local address, by peripheral node sender; ignored by receivers implementing the current level of SNA
5-6	bits 0-1, reserved bits 2-3, retired: set to 01 by subarea PU sender; set to 00 by peripheral node sender; ignored by receivers implementing the current level of SNA bits 4-15, procedure related identifier (PRID) Note: For unsolicited replies (byte 7, bit 0 = 0), the PRID field contains X'000'. For solicited replies (byte 7, bit 0 = 1), the PRID field echoes the PRID from the NMVT RU request. For requests that need no replies, this field contains X'000'.
7	Flags: bit 0, solicitation indicator: used only for PU-to-SSCP|SNCP flow (reserved for SSCP|SNCP-to-PU flow): 0 unsolicited NMVT 1 solicited NMVT

bits 1-2, sequence field--used only for PU-to-SSCP|SNCP flow (reserved for SSCP|SNCP-to-PU flow):
- 00 only NMVT for this PRID
- 01 last NMVT for this PRID
- 10 first NMVT for this PRID
- 11 middle NMVT for this PRID

bit 3, SNA Address List subvector indicator:
- 0 For the SSCP-to-PU flow: MS major vector in this NMVT does not contain an SNA Address List subvector

  For the PU-to-SSCP flow: MS major vector in this NMVT does not contain an SNA Address List subvector, or it contains an SNA Address List subvector that does not require address-to-name translation by the SSCP
- 1 For the SSCP-to-PU flow: MS major vector in this NMVT contains an SNA Address List subvector

  For the PU-to-SSCP flow: MS major vector in this NMVT contains an SNA Address List subvector that requires address-to-name translation by the SSCP

bits 4-7, reserved

8-m One MS major vector, as described (using zero-origin indexing) in the table in "MS Major Vectors and Unique Subvectors" in Chapter 8.

NOTIFY; LU-->SSCP, Norm; FMD NS(s) (NOTIFY)

NOTIFY is used to send information from an LU to an SSCP.

0-2 X'810620' NS header

3-p NOTIFY vector X'0C' as described in detail below

X'0C' LU-LU Session Services Capabilities: used to inform the SSCP having an active session with the sending LU of the current LU-LU session services capability of that LU

NOTIFY vectors (described zero-origin)

LU-LU Session Services Capabilities NOTIFY Vector

Note: This NOTIFY vector should not be confused with control vector X'0C' which carries similar information.

0 Key: X'0C'

1 Length of Vector Data field, encoded in binary

2-m Vector Data

2 LU-LU session capability:

bits 0-3, (reserved)

bits 4-7, secondary LU capability:
- 0000 SLU capability is inhibited, sessions can neither be queued nor started
- 0001 SLU capability is disabled, sessions can be queued but not started
- 0010 reserved
- 0011 SLU capability is enabled, sessions can be queued or started

3-4	Retired, (set to X'0001')
5-7	Retired
8-15(=m)	Retired, (set to X'4040404040404040' or omitted)

QC; LU-->LU, Norm; DFC (QUIESCE COMPLETE)

> QC is sent by a half-session after receiving QEC, to indicate that it has quiesced. This RU is not used for LU 6.2

0	X'81' request code

QEC; LU-->LU, Exp; DFC (QUIESCE AT END OF CHAIN)

> QEC is sent by a half-session to quiesce its partner half-session after it (the partner) finishes sending the current chain (if any). This RU is not used for LU 6.2.

0	X'80' request code

RELQ; LU-->LU, Exp; DFC (RELEASE QUIESCE)

> RELQ is used to release a half-session from a quiesced state. This RU is not used for LU 6.2

0	X'82' request code

RQR; SLU-->PLU, Exp; SC (REQUEST RECOVERY)

> RQR is sent by the secondary to request the primary to initiate recovery for the session by sending CLEAR or to deactivate the session. This RU is not used for LU 6.2.

0	X'A3' request code

RSHUTD; SLU-->PLU, Exp; DFC (REQUEST SHUTDOWN)

> RSHUTD is sent from the secondary to the primary to indicate that the secondary is ready to have the session deactivated. RSHUTD does not request a shutdown; therefore, SHUTD is not a proper reply; RSHUTD requests an UNBIND. This RU is not used for LU 6.2.

0	X'C2' request code

RTR; LU-->LU, Norm; DFC (READY TO RECEIVE)

> RTR indicates to the bidder that it is now allowed to initiate a bracket. RTR is sent only by the first speaker.

0	X'05' request code

SBI; LU-->LU, Exp; DFC (STOP BRACKET INITIATION)

> SBI is sent by either half-session to request that the receiving half-session stop initiating brackets by continued sending of BB and the BID request. This RU is not used for LU 6.2.

0	X'71' request code

SDT; PLU-->SLU, SSCP-->PU|SSCP, Exp; SC (START DATA TRAFFIC)

> SDT is sent by the primary session control to the secondary session control to enable the sending and receiving of FMD and DFC requests and responses by both half-sessions. This RU is not used for LU 6.2.

0	X'A0' request code

SHUTC; SLU-->PLU, Exp; DFC (SHUTDOWN COMPLETE)

> SHUTC is sent by a secondary to indicate that it is in the shutdown (quiesced) state. This RU is not used for LU 6.2.

0	X'C1' request code

SHUTD; PLU-->SLU, Exp; DFC (SHUTDOWN)

> SHUTD is sent by the primary to request that the secondary shut down (quiesce) as soon as convenient. This RU is not used for LU 6.2.

0	X'C0' request code

SIG; LU-->LU, Exp; DFC (SIGNAL)

> SIG is an expedited request that can be sent between half-sessions, regardless of the status of the normal flows. It carries a four-byte value, of which the first two bytes are the signal code and the last two bytes are the signal extension value.

0	X'C9' request code
1-2	Signal code: X'0000' no-op (no system-defined code) X'0001' request to send (only value defined for LU 6.2) X'0002' assistance requested X'0003' intervention required (no data loss)
3-4	Signal extension: set by the sending end user or NAU services manager, or set to X'0001' for LU 6.2 by data flow control

STSN; PLU-->SLU, Exp; SC (SET AND TEST SEQUENCE NUMBERS)

STSN is sent by the primary half-session sync point manager to resynchronize the values of the half-session sequence numbers, for one or both of the normal flows at both ends of the session. This RU is not used for LU 6.2.

0 X'A2' request code
1 bits 0-1, action code for S-->P flow (related data in bytes 2-3)
bits 2-3, action code for P-->S flow (related data in bytes 4-5)
Note: Each action code is set and processed independently. Values for either action code are:
00 ignore; this flow not affected by this STSN
01 set; the half-session value is set to the value in bytes 2-3 or 4-5, as appropriate
10 sense; secondary half-session's sync point manager returns the transaction processing program's sequence number for this flow in the response RU
11 set and test; the half-session value is set to the value in appropriate bytes 2-3 or 4-5, and the secondary half-session's sync point manager compares that value against the transaction processing program's number and responds accordingly
bits 4-7, reserved
2-3 Secondary-to-primary sequence number data to support S-->P action code
4-5 Primary-to-secondary sequence number data to support P-->S action code
Note: For action codes 01 and 11, the appropriate bytes 2-3 or 4-5 contain the value to which the half-session value is set and against which the secondary half-session's sync point manager tests the transaction processing program's value for the respective flow. For action codes 00 and 10, the appropriate bytes 2-3 or 4-5 are reserved.

TERM-SELF Format 0; TLU-->SSCP, Norm; FMD NS(s) (TERMINATE-SELF)

TERM-SELF from the TLU requests that the SSCP assist in the termination of one or more sessions between the sender of the request (TLU = OLU) and the DLU. This RU is not used for LU 6.2; refer to TERM-SELF Format 1.

0-2 X'010683' NS header
3 Type:
bits 0-1, 00 the request applies to active and pending-active sessions
01 the request applies to active, pending-active, and queued sessions
10 the request applies to queued only sessions
11 reserved
bit 2, reserved if byte 3, bit 4 = 1; otherwise:
0 forced termination—session to be deactivated immediately and unconditionally
1 orderly termination—permitting an end-of-session procedure to be executed at the PLU before the session is deactivated
bit 3, 0 do not send DACTLU to OLU; another session initiation request will be sent for OLU

1 send DACTLU to OLU when appropriate; no further session initiation request will be sent (from this sender) for OLU
bit 4, 0 orderly or forced (see byte 3, bit 2)
1 clean up
bits 5-6, 00 select session(s) for which DLU is PLU
01 select session(s) for which DLU is SLU
10 select session(s) regardless of whether DLU is SLU or PLU
11 reserved
bit 7, 0 indicates that the format of the RU is Format 0 and that byte 3 is the Type byte.

4-m Uninterpreted Name of DLU
4 Type: X'F3' logical unit
5 Length, in binary, of DLU name
Note: If the length value of the DLU name is 0, then the TERM-SELF applies to all sessions, as specified in the Type byte, where the TLU is a partner.
6-m EBCDIC character string

Note: The following defaults are supplied by the SSCP receiving a Format 0 TERM-SELF:

- Reason: network user, normal
- Notify: do not notify
- URC is not used in mapping to subsequent requests.

TERM-SELF Format 1; TLU-->SSCP, Norm; FMD NS(s) (TERMINATE-SELF)

> TERM-SELF from the TLU requests that the SSCP assist in the termination of one or more sessions between the sender of the request (TLU = OLU) and the DLU.

0-2 X'810683' NS header
3 bits 0-3, format:
0001 Format 1 (only value defined)
bits 4-6, reserved
bit 7, 1 indicates that byte 3, bits 0-3, contain the format value
4 Type:
bits 0-1, 00 the request applies to active and pending-active sessions
01 the request applies to active, pending-active, and queued sessions (only value defined for LU 6.2)
10 the request applies to queued sessions only
11 reserved
bit 2, reserved if byte 4, bit 7 = 1; otherwise:
0 forced termination—session to be deactivated immediately and unconditionally
1 orderly termination—permitting an end-of-session procedure to be executed at the PLU before the session is deactivated
bit 3, 0 do not send DACTLU to OLU; another session initiation request will be sent for OLU
1 send DACTLU to OLU when appropriate; no further session initiation request will be sent (from this sender) for OLU (only value defined for LU 6.2)
bit 4, reserved
bits 5-6, 00 select session(s) for which DLU is PLU
01 select session(s) for which DLU is SLU

10 select session(s) regardless of whether DLU is SLU or PLU
11 reserved
bit 7, 0 orderly or forced (see byte 4, bit 2)
1 clean up

5 Reason:
bit 0, 0 network user (only value defined for LU 6.2)
1 network manager
bit 1, 0 normal termination
1 abnormal termination
bits 2-7, reserved

6 NOTIFY specifications (reserved for LU 6.2):
bits 0-5, reserved
bit 6, 0 do not notify TLU when the session takedown procedure is complete
1 notify the TLU when the session takedown procedure is complete
bit 7, reserved

7 Reserved

8-n Session key, as described in the section "Session Key" in Chapter 8
Note: One of the following session keys is used:
X'01' uninterpreted name of DLU
Note: If the length value is 0, then the TERM-SELF applies to all sessions specified in the Type byte where the TLU is a partner.
X'0A' URC
Note: This URC is the one carried in the INIT issued previously by the same LU (i.e., ILU = TLU), and differs from the one in bytes n+4 through p.

n+1-n+2 Retired: set to X'0000'

n+3-p User Request Correlation (URC) Field

n+3 Length, in binary, of URC field
Note: X'00' = no URC.

n+4-p URC: LU-defined identifier; this value can be returned by the SSCP in a subsequent NOTIFY to correlate the NOTIFY to this terminating request

UNBIND; LU-->LU, Exp; SC (UNBIND SESSION)

> UNBIND is sent to deactivate an active session between the two LUs.

0 X'32' request code

1 UNBIND type (for UNBIND types X'00' through X'06' and X'80' through X'FF', the session is ended when the response is received; for UNBIND types X'07' through X'7F', the session is ended immediately):
X'01' normal end of session
X'02' BIND forthcoming; retain the node resources allocated to this session, if possible
X'06' invalid session parameters: the BIND negotiation has failed because the primary half-session cannot support parameters specified by the secondary
X'07' virtual route inoperative: the virtual route used by the LU-LU session has become inoperative, thus forcing the deactivation of the identified LU-LU session
X'08' route extension inoperative: the route extension used by the LU-LU session has become inoperative, thus forcing the deactivation of the identified LU-LU session

X'09' hierarchical reset: the identified LU-LU session is being deactivated because of a +RSP((ACTPU | ACTLU), Cold)
X'0A' SSCP gone: the identified LU-LU session had to be deactivated because of a forced deactivation of the SSCP-PU or SSCP-LU session (e.g., DACTPU, DACTLU, or DISCONTACT was received)
X'0B' virtual route deactivated: the identified LU-LU session had to be deactivated because of a forced deactivation of the virtual route being used by the LU-LU session
X'0C' LU failure--unrecoverable: the identified LU-LU session had to be deactivated because of an abnormal termination of the PLU or SLU; recovery from the failure was not possible
X'0E' LU failure--recoverable: the identified LU-LU session had to be deactivated because of an abnormal termination of one of the LUs of the session; recovery from the failure may be possible
X'0F' cleanup: the node sending UNBIND is resetting its half-session before receiving the response from the partner node
X'11' gateway node cleanup: a gateway node is cleaning up the session because a gateway SSCP has directed the gateway node (via NOTIFY) to deactivate the session (e.g., a session setup error or session takedown failure has occurred)
X'FE' session failure: the session has failed for a reason specified by the associated sense data

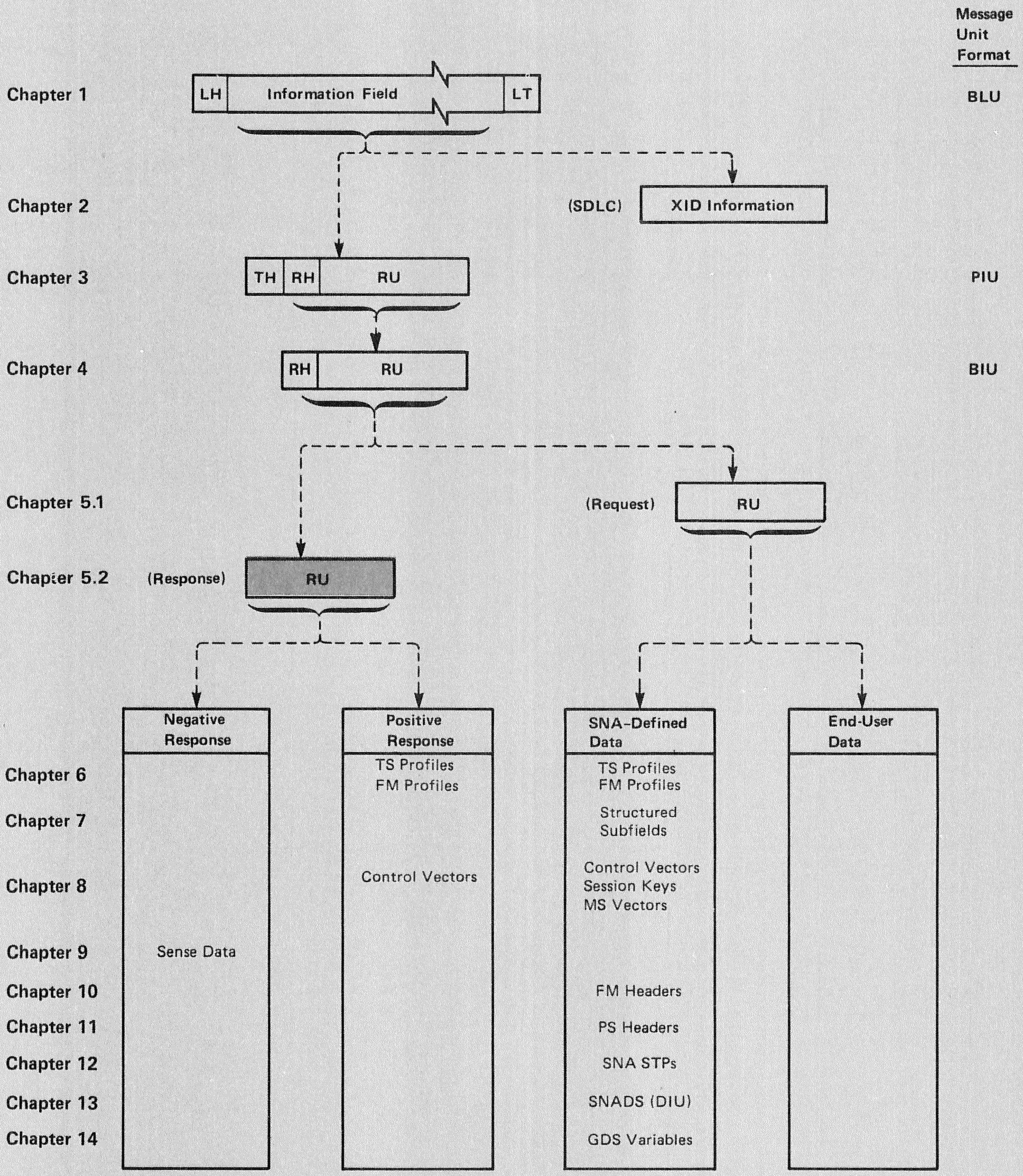
- For UNBIND Type=X'FE', the Sense Data field is included; otherwise, it is omitted.

2-5 Sense data: same value as generated at the time the error was originally detected (e.g., for a negative response, receive check, or EXR)

Note 1: Extensions to the UNBIND (unrecognized data after byte 5 for UNBIND type=X'FE'; unrecognized data after byte 1 for all other types) are ignored by receivers at the current level of SNA.

Chapter 5.2

Response Units



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 5.2. RESPONSE UNITS

INTRODUCTION

Apart from the exceptions cited below, response units return the number of bytes specified in the following table; only enough of the request unit is returned to include the field-formatted request code or NS header.

RU Category of Response	Number of Bytes
DFC	1
SC	1
NC	1
FMD NS (FI=1) (field-formatted)	3
FMD NS (FI=0) (character-coded)	0
FMD (LU-LU)	0

All negative responses return four bytes of sense data in the RU, followed by either:

1. The number of bytes specified in the table above, or

2. Three bytes (or the entire request unit, if shorter than three bytes).

The second option applies where a sensitivity to SSCP-based sessions versus LU-LU sessions does not necessarily exist and can be chosen for implementation simplicity. Refer to Chapter 9 for sense data values and their corresponding meanings.

Some positive response units return the request code or NS header followed by additional data. "Positive Response Units with Extended Formats" on page 5.2-3 contains detailed formats of these response units, arranged in alphabetical order. Each format description begins with the following heading:

"RSP(ABBREVIATED RU NAME); Origin-NAU-->Destination-NAU, Normal (Norm) or Expedited (Exp) Flow; RU Category"

Notes:

1. "RU Category" is abbreviated as follows:

DFC	data flow control
SC	session control
NC	network control

FMD NS(ma)	function management data, network services, management services (note: formerly maintenance services)
FMD NS(s)	function management data, network services, session services

2. Throughout the format descriptions, reserved is used as follows: reserved bits, or fields, are ones that currently are set to 0's (unless explicitly stated otherwise); reserved values are those that currently are invalid. Correct usage of reserved fields is enforced by the sender; no receive checks are made on these fields.

3. Throughout the format descriptions, retired fields and values are those that were once defined in SNA but are no longer defined. To accommodate implementations of back-level SNA, current implementations of SNA treat retired fields as follows: send checks enforce the setting of retired fields to all 0's except where other unique values are required (described individually); no receive checks are made on these fields, thereby accepting back-level settings of these fields. Special handling of retired fields, such as echoing or passing on retired fields as received, is discussed where appropriate.

4. User data, control vectors, and control lists referred to in the format descriptions are described in Chapter 7 and Chapter 8.

POSITIVE RESPONSE UNITS WITH EXTENDED FORMATS

RSP(ACTLU); LU-->SSCP, Exp; SC

0	X'0D' request code
1	Type of activation selected: X'01' cold X'02' ERP
2	bits 0-3, FM profile: X'0' FM Profile 0 X'6' FM Profile 6 bits 4-7, TS profile: same as the corresponding request

Note: Two versions of this RU are defined.

- A full response can be sent in which bytes 0-m are present.

3-m	Control vectors as described in the section "Control Vectors" in Chapter 8 Note: The following control vectors may be included; they are parsed according to subfield parsing rule KL. When present, they appear in the order specified. X'00' SSCP-LU session capabilities (always present, always first) X'0C' LU-LU session services capabilities (always present, always second)

- A two-byte response may be received; it means maximum RU size = 256 bytes, LU-LU session limit = 1, the LU can act as a secondary LU, and all other fields in control vectors X'00' and X'0C' are defaulted to 0's.

RSP(ACTPU); PU-->SSCP|PUCP, Exp; SC

0	X'11' request code
1	bits 0-1, reserved bits 2-3, format of response: 00 format 0 bits 4-7, type activation selected: X'2' ERP
2-9	Contents ID: eight-character EBCDIC symbolic name of the load module currently operating in the node; eight space (X'40') characters is the default value

RSP(BIND); SLU-->PLU, Exp; SC

A +RSP(BIND) carries the session parameters as indicated by the SLU or by intermediate nodes along the session path.

- A short (1-byte) response may be sent for a nonnegotiable BIND request that specifies no session-level cryptography.
- A cryptography response (bytes 0-k) may be sent for a nonnegotiable BIND request that specifies session-level cryptography.
- A negotiable response (bytes 0-r) may be sent for a negotiable BIND request.

0	X'31' request code
1	bits 0-3, format: 0000 (only value defined)
	bits 4-7, type: 0000 negotiable (only value defined for LU 6.2) 0001 nonnegotiable
2-25	Bytes 2-25 of the BIND request: for a negotiable response, the negotiated values may differ; otherwise, the values are the same as those received in the BIND request
26-k	Cryptography Options (see Note 3)
26	bits 0-1, private cryptography options: for a nonnegotiable response, same value as received in the request, if present
	bits 2-3, session-level cryptography options: for a nonnegotiable response and an LU 6.2 response, same value returned as received in the request, if present
	bits 4-7, session-level cryptography options field length: same value (Bytes 27-k are omitted if this length field is omitted or set to 0.)
27	bits 0-1, session cryptography key encipherment method: same value returned as received in the request, if present
	bits 2-4, reserved
	bits 5-7, cryptography cipher method: same value returned as received
28-k	An eight-byte implementation-chosen, nonzero, pseudo random session-seed cryptography value enciphered under the session cryptography key, if session-level cryptography is specified; otherwise, omitted
k+1(=m)	Retired: set to 0 by implementations at the current level of SNA
m+1	Length of user data
m+2-n	User data: for a negotiable response, the user data may differ from that received on the BIND request
n+1	Length of URC
n+2-p	URC as received on the BIND request
p+1(=r)	Retired: set to 0 by implementations at the current level of SNA

Note 1: On a response, if the last byte of a response is a length field and that field is 0, that byte may be dropped from the response. This applies also to byte 26 (where the count occupies only bits 4-7) if bits 0-3 are also 0—the entire byte may be dropped if no bytes follow.

Note 2: In negotiable BIND responses, reserved fields in the BIND are set by the SLU to binary 0's in the RSP(BIND); any fields at the end of the BIND (after byte r) that are not recognized by the SLU are discarded and not returned in the RSP(BIND).

Note 3: The first byte of the Cryptography Options field (byte 26) is returned on the response for a nonnegotiable BIND only when session-level cryptography was specified in the BIND. Byte 26 is always present in any negotiable response if not truncated as allowed in Note 1. In all cases, however, the remaining bytes of the Cryptography Options field (bytes 27-k) are present only if session-level cryptography was specified in the BIND.

RSP(STSN); SLU-->PLU, Exp; SC

0 X'A2' request code

1 bits 0-1, result code for S-->P action code in the request (related data in bytes 2-3)

bits 2-3, result code for P-->S action code in the request (related data in bytes 4-5)

Note: Values for either result code are:

- For set or ignore action code:

01 ignore (other values reserved); appropriate bytes 2-3 or 4-5 reserved

- For sense action code:

00 for LU-LU session type 0: user-defined meaning; for all other LU-LU session types: reserved (appropriate bytes 2-3 or 4-5 reserved)

01 reserved

10 secondary half-session's sync point manager does not maintain or cannot return a valid transaction processing program sequence number (appropriate bytes 2-3 or 4-5 reserved)

11 transaction processing program sequence number, as known at the secondary, is returned in bytes 2-3 or 4-5, as appropriate

- For set and test action code:

00 for LU-LU session type 0: user-defined meaning; for all other LU-LU session types: invalid sequence numbers have been detected by the secondary (appropriate bytes 2-3 or 4-5 return the secondary transaction processing program sequence number)

Note: An invalid determination results when the sequence number indicated could not have occurred. For example, the mounting of an incorrect sync point log tape by the operator at one of the LUs would cause this condition.

01 value received in STSN request equals the transaction processing program sequence number value as known at the secondary (appropriate bytes 2-3 or 4-5 return the secondary's value for the transaction processing program sequence number)

10 secondary half-session's sync point manager does not maintain or cannot return a valid transaction processing program sequence number (appropriate bytes 2-3 or 4-5 reserved)

11 value received in STSN request does not equal the transaction processing program sequence number value as known at the secondary (appropriate bytes 2-3 or 4-5 return the

secondary's value for the transaction processing program sequence number)
bits 4-7, reserved

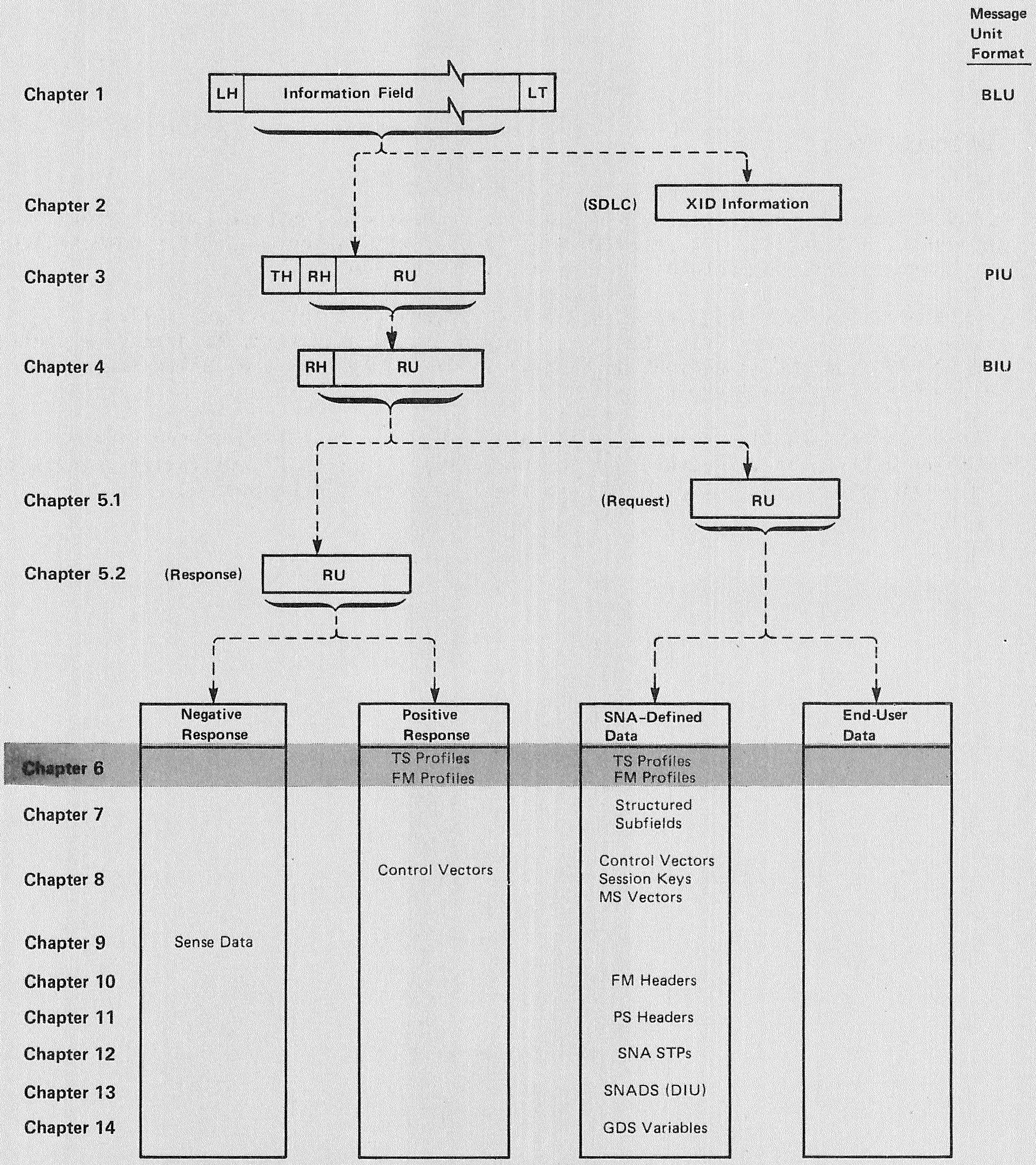
2-3 Secondary-to-primary normal-flow sequence number data to support S-->P result code, or reserved (see Note 1 above)

4-5 Primary-to-secondary normal-flow sequence number data to support P-->S result code or reserved (see Note 1 above)

Note: Where the STSN request specified as action codes two "sets," two "ignores," or a combination of "set" and "ignore," the positive response RU optionally may consist of one byte—X'A2' (the STSN request code)—rather than all six bytes.

Chapter 6

Profiles



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 6. PROFILES

INTRODUCTION

Some of the session protocols (such as for request and response control modes, brackets, and pacing) are selectable at session activation. Specific combinations of these selectable protocol options are known as profiles.

Those profiles that refer to transmission control (TC) options are called transmission services (TS) profiles; those profiles that refer to data flow control (DFC) and function management data services (FMDS) options are called function management (FM) profiles.

The TS and FM profiles to be used in any session are specified at the time of session activation via parameters in the appropriate session activation request and response (see ACTPU, ACTLU, BIND, and their responses in Chapter 5).

TRANSMISSION SERVICES (TS) PROFILES

This section describes the transmission services (TS) profiles and their use for sessions defined in SNA. Profile numbers not shown are reserved.

Note: If the TS Usage field in BIND specifies a value for a parameter, that value is used unless it conflicts with a value specified by the TS profile. The TS profile overrides the TS Usage field.

Figure 6-1 identifies the different sessions and logical unit (LU) types that use each TS profile.

TS Profile	Session Types	LU Types
1	SSCP-PU(T1|2),[1] SSCP-LU	-
2	LU-LU	0
3	LU-LU	0, 1, 2, 3
4	LU-LU	0, 1, 6.1
7	LU-LU	0, 4, 6.2, 7

---

[1] The boundary function serves in place of the PU type 1 (e.g., to process ACTPU).

Figure 6-1. TS Profiles and Their Usage

TS PROFILE 1

Profile 1 (used on SSCP-PU and SSCP-LU sessions) specifies the following session rules:

- No pacing.
- Identifiers rather than sequence numbers are used on the normal flows (whenever the TH format used includes a sequence number field).
- SDT, CLEAR, RQR, STSN, and CRV are not supported.
- Maximum RU size on the normal flow between an SSCP and a peripheral PU or peripheral LU is 256, unless a different value is specified in RSP(ACTLU) in control vector X'00'.

There is no TS Usage field associated with this profile.

TS PROFILE 2

Profile 2 (used on LU-LU sessions) specifies the following session rules:

- Primary-to-secondary and secondary-to-primary normal flows are paced.
- Sequence numbers are used on the normal flows (whenever the TH format used includes a sequence number field).
- CLEAR is supported.
- SDT, RQR, STSN, and CRV are not supported.

The TS Usage subfields defining the options for this profile are:

- Pacing window counts
- Maximum RU sizes on the normal flows

TS PROFILE 3

Profile 3 (used on LU-LU sessions) specifies the following session rules:

- Primary-to-secondary and secondary-to-primary normal flows are paced.
- Sequence numbers are used on the normal flows (whenever the TH format used includes a sequence number field).
- CLEAR and SDT are supported.
- RQR and STSN are not supported.
- CRV is supported when session-level cryptography is selected (via a BIND parameter).

The TS Usage subfields defining the options for this profile are:

- Pacing window counts
- Maximum RU sizes on the normal flows

TS PROFILE 4

Profile 4 (used on LU-LU sessions) specifies the following session rules:

- Primary-to-secondary and secondary-to-primary normal flows are paced.
- Sequence numbers are used on the normal flows (whenever the TH format used includes a sequence number field).
- SDT, CLEAR, RQR, and STSN are supported.
- CRV is supported when session-level cryptography is selected (via a BIND parameter).

The TS Usage subfields defining the options for this profile are:

- Pacing window counts
- Maximum RU sizes on the normal flows

TS PROFILE 7

Profile 7 (used on LU-LU sessions) specifies the following session rules:

- Primary-to-secondary and secondary-to-primary normal flows are optionally paced.
- Sequence numbers are used on the normal flows (whenever the TH format used includes a sequence number field).
- SDT, CLEAR, RQR, and STSN are not supported.
- CRV is supported when session-level cryptography is selected (via a BIND parameter).

The TS Usage subfields in BIND defining the options for this profile are:

- Pacing window counts
- Maximum RU sizes on the normal flows

FUNCTION MANAGEMENT (FM) PROFILES

This section describes the function management (FM) profiles and their use for sessions defined in SNA. Profile numbers not shown are reserved.

Note: If the FM Usage field in BIND specifies a value for a parameter, that value is used unless it conflicts with a value specified by the FM profile. The FM profile overrides the FM Usage field.

Figure 6-2 identifies the different sessions and logical unit (LU) types that use each FM profile.

FM Profile	Session Types	LU Types
0	SSCP-PU(T1|2),[1] SSCP-LU	-
2	LU-LU	0
3	LU-LU	0, 1, 2, 3
4	LU-LU	0, 1
6	SSCP-LU	-
7	LU-LU	0, 4, 7
18	LU-LU	0, 6.1
19	LU-LU	6.2

---

[1] The boundary function serves in place of the PU type 1 (e.g., to process ACTPU).

Figure 6-2. FM Profiles and Their Usage

FM PROFILE 0

Profile 0 (used on SSCP-PU and SSCP-LU sessions) specifies the following session rules:

- Primary and secondary half-sessions use immediate request mode and immediate response mode.
- Only single-RU chains allowed.
- Primary and secondary half-session chains indicate definite response. Half-session chains generated by a boundary function on behalf of the LU may indicate no-response or definite response.
- No compression.
- Primary half-session sends no DFC RUs.
- Secondary LU half-session may send LUSTAT.
- No FM headers.
- No brackets.
- No alternate code.
- Normal-flow send/receive mode is full-duplex.

FM PROFILE 2

Profile 2 (used on LU-LU sessions) specifies the following session rules:

- Secondary LU half-session uses delayed request mode.
- Secondary LU half-session uses immediate response mode.
- Only single-RU chains allowed.
- Secondary LU half-session requests indicate no-response.
- No compression.
- No DFC RUs.
- No FM headers.
- Secondary LU half-session is first speaker if brackets are used.
- Bracket termination rule 2 is used if brackets are used.
- Primary LU half-session will send EB.
- Secondary LU half-session will not send EB.
- Normal-flow send/receive mode is FDX.
- Primary LU half-session is responsible for recovery.

The FM Usage fields defining the options for Profile 2 are:

- Primary request control mode selection
- Primary chain response protocol (no-response may not be used)
- Brackets usage and reset state
- Alternate code

FM PROFILE 3

Profile 3 (used on LU-LU sessions) specifies the following session rules:

- Primary LU half-session and secondary LU half-session use immediate response mode.
- Primary LU half-session and secondary LU half-session support the following DFC functions:
  - CANCEL
  - SIGNAL
  - LUSTAT (allowed secondary-to-primary only)
  - CHASE
  - SHUTD
  - SHUTC
  - RSHUTD
  - BID and RTR (allowed only if brackets are used)

The FM usage fields defining the options for Profile 3 are:

- Chaining use (primary and secondary)
- Request control mode selection (primary and secondary)
- Chain response protocol (primary and secondary)
- Compression indicator (primary and secondary)
- Send EB indicator (primary and secondary)
- FM header usage
- Brackets usage and reset state
- Bracket termination rule
- Alternate Code Set Allowed indicator
- Normal-flow send/receive mode
- Recovery responsibility
- Contention winner/loser
- Half-duplex flip-flop reset states

FM PROFILE 4

Profile 4 (used on LU-LU sessions) specifies the following session rules:

- Primary LU half-session and secondary LU half-session use immediate response mode.
- Primary LU half-session and secondary LU half-session support the following DFC functions:

  - CANCEL
  - SIGNAL
  - LUSTAT
  - QEC
  - QC
  - RELQ
  - SHUTD
  - SHUTC
  - RSHUTD
  - CHASE
  - BID and RTR (allowed only if brackets are used)

The FM Usage fields defining the options for Profile 4 are:

- Chaining use (primary and secondary)
- Request control mode selection (primary and secondary)
- Chain response protocol (primary and secondary)
- Compression indicator (primary and secondary)
- Send EB indicator (primary and secondary)
- FM header usage
- Brackets usage and reset state
- Bracket termination rule
- Alternate Code Set Allowed indicator
- Normal-flow send/receive mode
- Recovery responsibility
- Contention winner/loser
- Half-duplex flip-flop reset states

Profile 6 (used on SSCP-LU sessions) specifies the following session rules:

- Only single-RU chains allowed.
- Primary and secondary half-sessions use delayed request mode and delayed response mode.
- Primary and secondary half-session chains may indicate definite response, exception response, or no response.
- Primary half-session sends no DFC RUs.
- Secondary half-session may send LUSTAT.
- No FM headers.
- No compression.
- No brackets.
- No alternate code.
- Normal-flow send/receive mode is full-duplex.

FM PROFILE 7

Profile 7 (used on LU-LU sessions) specifies the following session rules:

- Primary LU half-session and secondary LU half-session use immediate response mode.
- Primary LU half-session and secondary LU half-session support the following DFC functions:
    - CANCEL
    - SIGNAL
    - LUSTAT
    - RSHUTD

The FM Usage fields defining the options for Profile 7 are:

- Chaining use (primary and secondary)
- Request control mode selection (primary and secondary)
- Chain response protocol (primary and secondary)
- Compression indicator (primary and secondary)
- Send EB indicator (primary and secondary)
- FM header usage
- Brackets usage and reset state
- Bracket termination rule
- Alternate Code Set Allowed indicator
- Normal-flow send/receive mode
- Recovery responsibility
- Contention winner/loser
- Half-duplex flip-flop reset states

FM PROFILE 18

Profile 18 (used on LU-LU sessions) specifies the following session rules:

- Primary LU half-session and secondary LU half-session use immediate response mode.
- Primary LU half-session and secondary LU half-session support the following DFC functions:
  - CANCEL
  - SIGNAL
  - LUSTAT
  - BIS and SBI (allowed only if brackets are used)
  - CHASE
  - BID and RTR (allowed only if brackets are used)

The FM Usage fields defining the options for Profile 18 are:

- Chaining use (primary and secondary)
- Request control mode selection (primary and secondary)
- Chain response protocol (primary and secondary)
- Compression indicator (primary and secondary)
- Send EB indicator (primary and secondary)
- FM header usage
- Brackets usage and reset state
- Bracket termination rule
- Alternate Code Set Allowed indicator
- Normal-flow send/receive mode
- Recovery responsibility
- Contention winner/loser
- Half-duplex flip-flop reset states

Profile 19 (used on LU-LU sessions) specifies the following session rules:

- Primary LU half-session and secondary LU half-session use immediate request and immediate response mode.
- Multiple RU chains allowed.
- Primary LU half-session and secondary LU half-session chains indicate definite or exception response.
- No compression.
- Brackets are used.
- FM headers (types 5, 7, and 12 only) are allowed.
- Conditional termination for brackets (specified by CEB) will be used--primary and secondary half-sessions may send CEB.
- Normal-flow send/receive mode is half-duplex flip-flop.
- Half-duplex flip-flop reset state is <u>send</u> for the primary LU half-session and <u>receive</u> for the secondary LU half-session after RSP(BIND).
- Symmetric responsibility for recovery.
- Contention winner/loser polarity is negotiated at BIND time; the contention winner is the first speaker and the contention loser is the bidder.
- Primary and secondary half-sessions support the following DFC functions:

  - SIGNAL
  - LUSTAT
  - BIS
  - RTR

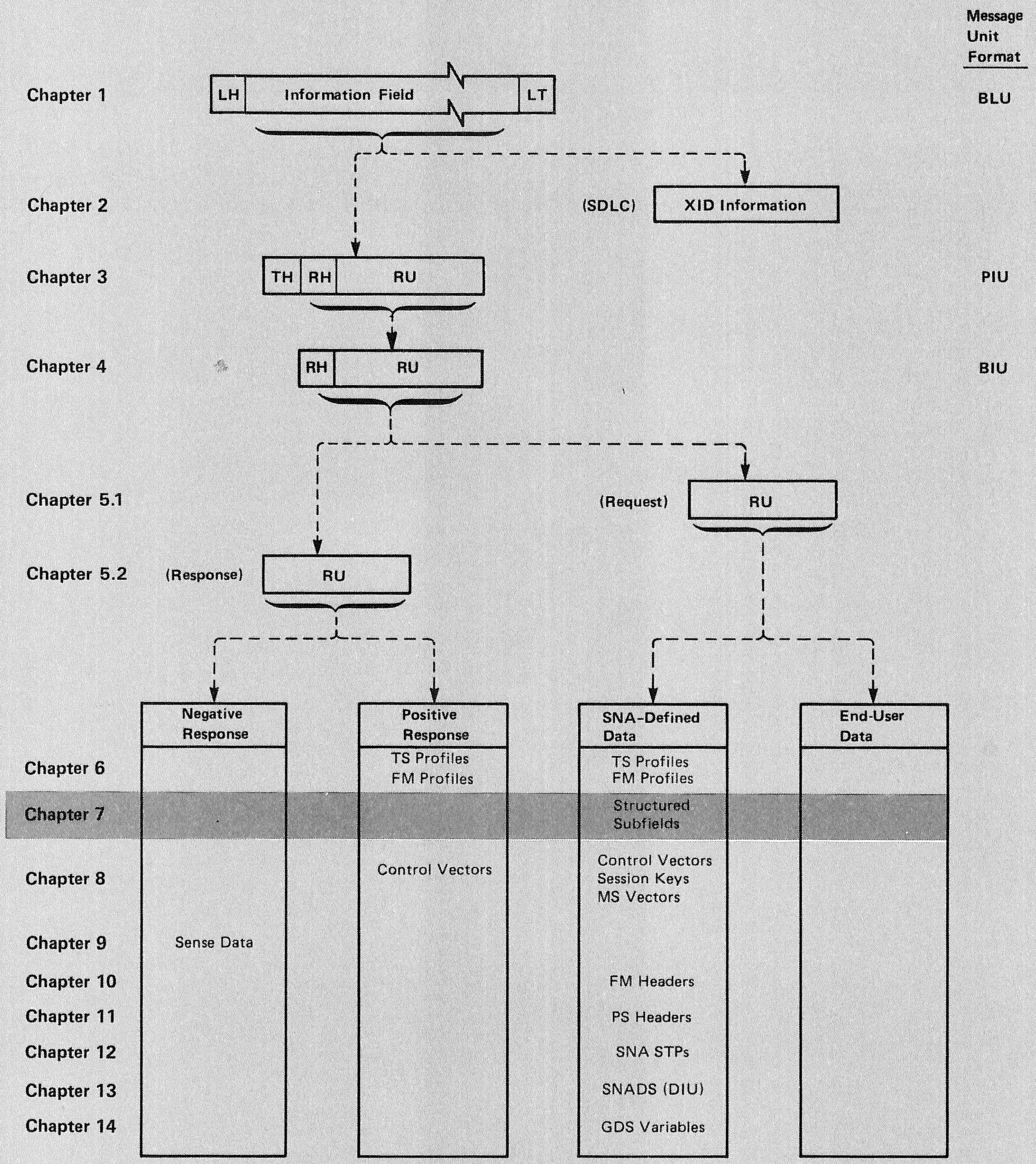
- The following combinations of RQE, RQD, CEB, and CD are allowed on end-chain RUs:

  - RQE*, CD, ¬CEB
  - RQD2, CD, ¬CEB
  - RQD3, CD, ¬CEB
  - RQE1, ¬CD, CEB
  - RQD*, ¬CD, CEB
  - RQD*, ¬CD, ¬CEB

The only FM Usage field defining options for Profile 19 is Contention Winner/Loser.

Chapter 7

User Data



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 7. USER DATA

The structured subfields of the User Data field are defined as follows (shown with zero-origin indexing of the subfield bytes—see the individual RU description for the actual displacement within the RU). Each subfield starts with a one-byte binary Length field and is identified by a subfield number in the following byte. The length does not include the Length byte itself. When more than one subfield is included, they appear in ascending order by subfield number.

For LU type 6.2, the Structured User Data field of BIND and RSP(BIND) may contain the Unformatted Data, Mode Name, Network-Qualified PLU Network Name, Network-Qualified SLU Network Name, and Session Instance Identifier subfields. Any subfields received in the Structured User Data field of BIND that are not recognized by the SLU are discarded and not returned as part of the Structured User Data field of the RSP(BIND).

Unformatted Data

> The Unformatted Data subfield may optionally be sent in BIND, RSP(BIND), or any of the INITIATE RUs. The content is implementation-defined.

0	Length of the remainder of the Unformatted Data subfield: values 1 to 17 (X'11') are valid for LU 6.2; otherwise, values 1 to 65 (X'41') are valid
1	X'00'
2-n	Unformatted data: a type-G symbol string

Session Qualifier

> The Session Qualifier subfield is used for LU 6.1. It may be carried in BIND, RSP(BIND), or any of the INITIATE RUs.

0	Length of the remainder of the Session Qualifier subfield (If Session Qualifier subfield is present, values 3 to 19 (X'13') are valid.)
1	X'01'
2	Length of primary resource qualifier: values 0 to 8 are valid (X'00' means no primary resource qualifier is present)
3-m	Primary resource qualifier
m+1	Length of secondary resource qualifier: values 0 to 8 are valid (X'00' means no secondary resource qualifier is present)
m+2-n	Secondary resource qualifier

Mode Name

> The Mode Name subfield is present in both BIND and RSP(BIND) if the PLU knows the mode name being used by the session.

0	Length of the remainder of the Mode Name subfield: values 1 to 9 are valid
1	X'02'
2-n	Mode name: 0 to 8 type-A symbol string characters with optional (but not significant) trailing blanks

Session Instance Identifier

> The Session Instance Identifier subfield may be present in both BIND and RSP(BIND).

0	Length of the remainder of the Session Instance Identifier subfield: values 3 to 9 are valid
1	X'03'
2-n	Session instance identifier: a type-G symbol string Note: In BIND, the PLU sets a unique session instance identifier of length 1 to 7 and appends it to X'00'. If known, the SLU compares its network-qualified name with that of the PLU; if the PLU name > SLU name then the SLU changes the first byte of the Session Instance Identifier subfield in the response from X'00' to X'F0', if the PLU name < SLU name then the subfield is simply echoed.

Network-Qualified PLU Network Name

> BIND contains the Network-Qualified PLU Network Name subfield (if the name is known by the PLU).

0	Length of the remainder of the Network-Qualified PLU Network Name subfield: values 2 to 18 (X'12') are valid
1	X'04'
2-n	Network-Qualified PLU network name Note: The network-qualified PLU network name is 1 to 17 bytes in length, consisting of an optional 1- to 8-byte network ID and a 1- to 8-byte LU name, both of which are type-A symbol strings. When present, the network ID is concatenated to the left of the LU name, using a separating period and having the form "NWID.NAME"; when the network ID is omitted, the period is also omitted.

Network-Qualified SLU Network Name

> The RSP(BIND) contains the Network-Qualified SLU Network Name subfield (if the name is known by the SLU).

0	Length of the remainder of the Network-Qualified SLU Network Name subfield: values 2 to 18 (X'12') are valid
1	X'05'
2-n	Network-Qualified SLU network name Note: The network-qualified SLU network name is 1 to 17 bytes in length, consisting of an optional 1- to 8-byte network ID and a 1- to 8-byte LU name, both of which are type-A symbol strings. When present, the network ID is concatenated to the left of the LU name, using a separating period and having the form "NWID.NAME"; when the network ID is omitted, the period is also omitted.

Random Data

> The Random Data subfield contains the random data used in session-level security verification. When session-level security verification is in effect, this subfield is present in both BIND and RSP(BIND).

0	Length of the remainder of the Random Data subfield: 10 is the only valid value
1	X'11'
2	Reserved
3-10	Random data: a type-G random value generated for subsequent checking in RSP(BIND) or FMH-12

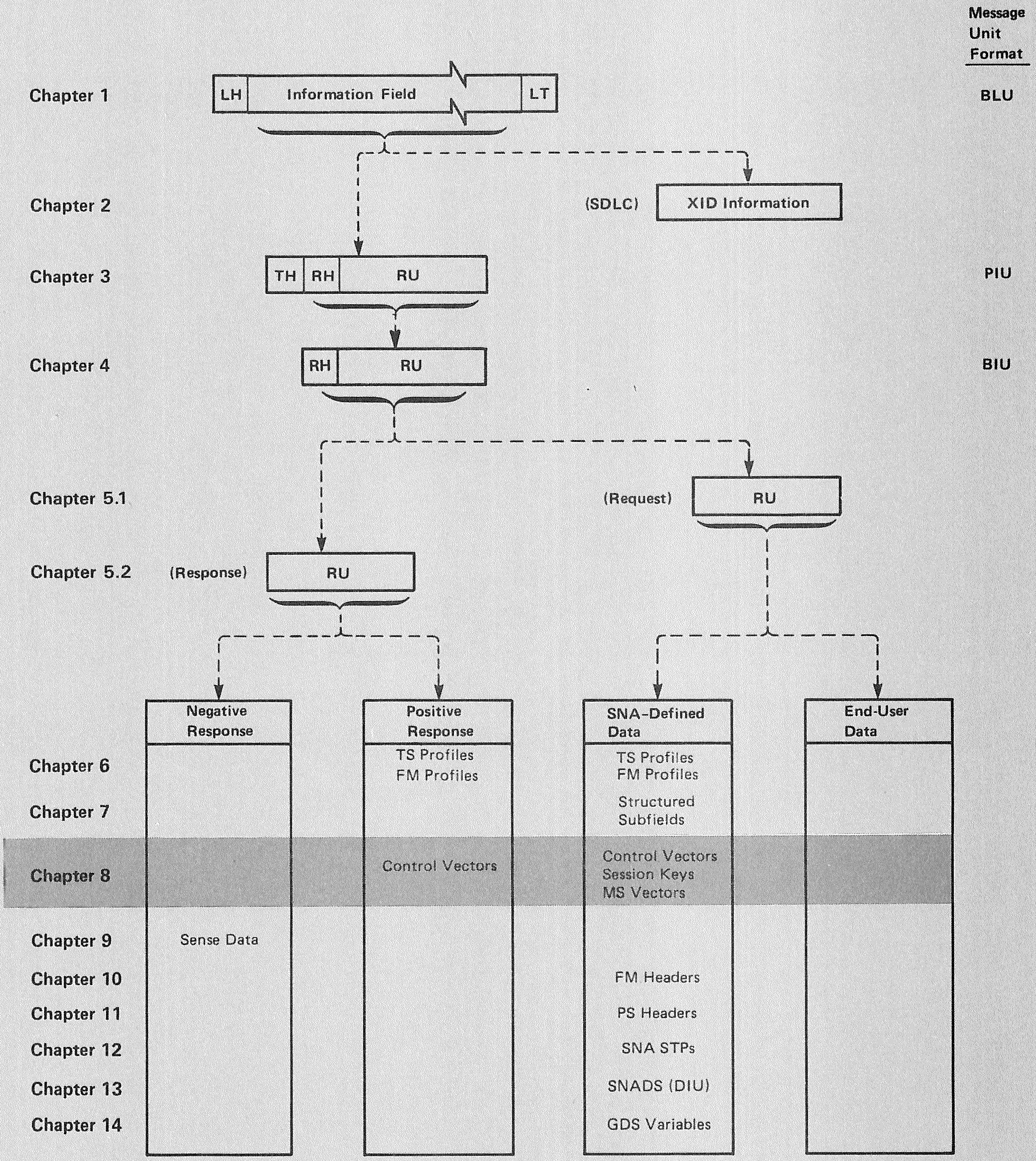
Enciphered Data

> The Enciphered Data subfield is present in the RSP(BIND) when session-level security verification is in effect. This subfield contains the enciphered version of the clear data received in BIND.

0	Length of the remainder of the Enciphered Data subfield: 9 is the only valid value
1	X'12'
2-9	Enciphered version of the Clear Data field carried in BIND (using the DES algorithm and the installation-defined LU-LU password as the cryptographic key)

Chapter 8

Common Fields



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 8. COMMON FIELDS

INTRODUCTION

This chapter contains detailed formats of the following common fields used in message units:

- Control vectors
- Session keys
- Management services major vectors and subvectors

SUBFIELD ENCODING/PARSING RULES

RULES FOR COMMON SUBSTRUCTURES

The following rules apply to encodings defined in this chapter; they govern the encoding of SNA-defined RU substructures, i.e., structures that are carried within some enclosing structure and that have one-byte keys identifying the substructures. The terms key and type are used interchangeably here, since both terms are used in the substructures to which the following rules apply.

Partitioning of Key/Type Values

The use of one-byte keys means that 256 values are available for defining substructures. The available values are partitioned as follows.

ARCHITECTURE-DEPENDENT KEYS: Within the category of control vectors, keys in the range X'00' to X'5F' are unique; within the independent category of management services (MS) subvectors, they are also unique.

UNIQUE KEYS: Keys in the range X'60' to X'7D' are unique across the composite grouping of control vectors and MS subvectors, i.e., such a key has the same meaning in both groups.

CONTEXT-SENSITIVE KEYS: Keys in the range X'80' to X'FD' are context-sensitive. These are unique only within the enclosing structure (e.g., control vector, GDS variable). Thus a subfield key X'80' may be defined for use within control vector X'30' and also within control vector X'31', and the subfields may be different. The only exception to this rule is found in the management services subfields. Keys in the range X'00' to X'7F' are unique within the enclosing subvector. Keys in the range X'80' to X'FF' are unique within the group of unique subvectors defined for a given management services major vector.

Parsing Rules

Common substructures with variable length formats, such as control vectors may be parsed in one of two ways. The parsing rule used is format specific--see the individual format description for the parsing rule used:

KL The Key field precedes the Length field and the length is the number of bytes, in binary, of the substructure's Data field (e.g., Vector Data field). The Length field value does **not** include the length of the substructure Header field.

LT	The Length field precedes the Key field (also called the "type" field--hence "LT") and the length is the number of bytes, in binary, of the substructure including **both** the Header field and the Data field.

The general format of a control vector, for example, is:

0-1	Vector header; Key=X'45' (see "Subfield Encoding/Parsing Rules" on page 8-2)
2-n	Vector Data

When the enclosing structure indicates use of subfield parsing rule KL, the first two bytes are interpreted as:

0	Key
1	Length (n-1), in binary, of the Vector Data field (i.e., excluding the length of the Vector Header field)

When the enclosing structure indicates use of subfield parsing rule LT, the first two bytes are interpreted as:

0	Length (n+1), in binary, of the control vector (i.e, including the Vector Header and Vector Data fields)
1	Type

Enclosing Rule for Substructures

All substructures that are enclosed by other than an RU (e.g., another substructure or a GDS variable) are constructed and parsed LT. This is the case even when, for example, an enclosing control vector is parsed KL. This rule holds true for all levels of nesting.

Consider the Product Set ID (X'10') control vector as an example of this rule. Imbedded within this substructure are other substructures, specifically Product Set Identifier (X'11') MS common subvectors.

When the Product Set ID (X'10') is present in XID format 3, it is parsed KL, whereas when it is present in NMVT, it is parsed LT. In both cases the Product Set Identifier (X'11') subvectors are parsed LT.

CONTROL VECTORS

INTRODUCTION

The following table shows, by key value, the control vector and the message-unit structures that can carry the control vector.

Key	Control Vector	Applicable Message-Unit Structures
X'00'	SSCP-LU Session Capabilities	RSP(ACTLU)
X'0C'	LU Session Services Capabilities	RSP(ACTLU)
X'0E'	Network Name	XID
X'10'	Product Set ID	XID
X'22'	XID Negotiation Error	XID

CONTROL VECTOR FORMATS

The control vectors are defined as follows (with zero-origin indexing of the vector bytes—see the individual RU description for the actual displacement within the RU):

Note: When more than one control vector may appear in an RU, unless otherwise stated, the vectors may appear in any order.

SSCP-LU Session Capabilities (X'00') Control Vector

0 Key: X'00'

1 Maximum RU size sent on the normal flow by either half-session: if bit 0 is set to 0, then no maximum is specified and the remaining bits 1-7 are ignored; if bit 0 is set to 1, then the byte is interpreted as X'ab' = a•2**b (Notice that, by definition, a≥8 and therefore X'ab' is a normalized floating point representation.) See Figure 5.1-1 on page 5.1-13 for all possible values.

2-3 LU Capabilities

bit 0, character-coded capability:
- 0 the SSCP may not send unsolicited character-coded requests; a solicited request is a reply request or a request that carries additional error information to supplement a previously sent negative response or error information after a positive response has already been sent
- 1 the SSCP may send unsolicited character-coded requests

bit 1, field-formatted capability:
- 0 the SSCP may not send unsolicited field-formatted requests
- 1 the SSCP may send unsolicited field-formatted requests

bits 2-15, reserved

4 Reserved

LU-LU Session Services Capabilities (X'0C') Control Vector

Note: Do not confuse control vector X'0C' with NOTIFY vector X'0C' which carries similar information.

0-1	Vector header; Key=X'0C' (see "Subfield Encoding/Parsing Rules" on page 8-2)
2-m	Vector Data
2	bits 0-3, (reserved) bits 4-7, secondary LU capability: 0000 SLU capability is inhibited, sessions can neither be queued nor started 0001 SLU capability is disabled, sessions can be queued but not started 0010 reserved 0011 SLU capability is enabled, sessions can be queued or started
3-4	LU-LU session limit: X'0001' session limit of 1 (only value allowed for peripheral LUs)
5-6	LU-LU session count: the number of LU-LU sessions that are not reset for this LU, and for which SESSEND will be sent to the SSCP
7	(reserved)
8-15(=m)	Retired, (set to X'4040404040404040' or omitted)

Network Name (X'0E') Control Vector

0-1	Vector header; Key=X'0E' (see "Subfield Encoding/Parsing Rules" on page 8-2) Note: A null X'0E' control vector consists of a vector header with no vector data. The length field is set appropriately.
2-n	Vector Data
2	Network name type: X'F1' PU name X'F3' LU name X'F4' CP name X'F7' link station name (not network-qualified)
3-n	Network-qualified name: a 1- to 17-byte name consisting of an optional qualifier concatenated to a 1- to 8-byte type-A symbol-string name; when present, the qualifier contains a 1- to 8-byte type-A symbol-string network identifier concatenated with a period (when the qualifier is not present, the period is omitted). The network-qualified name appears, for example, as follows: NETID.NAME, with no imbedded blanks and with optional (but not significant) trailing blanks.

Product Set ID (X'10') Control Vector

0-1	Vector Header; Key=X'10' (see "Subfield Encoding/Parsing Rules" on page 8-2)
2-n	Vector Data
2	Retired
3-n	Network product identifier: one or two Product Identifier (X'11') MS common subvectors, as described in "MS Common Subvectors" on page 8-23 , one for each hardware product and software product in the implementation of the PU

XID Negotiation Error (X'22') Control Vector

0-1	Vector header; Key=X'22' (see "Subfield Encoding/Parsing Rules" on page 8-2)
2-n	<u>Vector Data</u>
2-3	Error byte offset: the binary offset (zero-origin in the XID information field) of the first byte of the field in error
4(=n)	Error bit offset: the binary offset (zero-origin in the byte pointed at in the Error Byte Offset field) of the first bit of the field in error

SESSION KEY

The following table shows, by key value, the session key and the message-unit structures that can carry the session key.

Key	Session Key	Applicable Message-Unit Structures
X'01'	Network or Uninterpreted name	TERM-SELF
X'0A'	URC	TERM-SELF

The session keys are defined as follows, with zero-origin indexing of the key bytes—see the individual RU description for the actual displacement within the RU.

Network or Uninterpreted Name (X'01')

0	Key: X'01'
1	Type: X'F3' logical unit
2	Length, in binary, of name
3-n	Network or Uninterpreted Name Note: For a Network Name session key, the name is a symbolic name; for an Uninterpreted Name session key, the name is any EBCDIC character string.

URC (X'0A')

0	Key: X'0A'
1	Length, in binary, of the URC
2-n	URC: LU-defined identifier

MS MAJOR VECTORS AND UNIQUE SUBVECTORS

INTRODUCTION

The following table shows, by key value, the MS major vectors that NMVT can carry.

Key	MS Major Vector
X'0000'	Alert
X'0080'	RTM
X'8080'	Request RTM

Note: The major vectors are defined as follows (using zero-origin indexing):

The description of each major vector includes a matrix indicating the subvectors that may be included within it.

Subvectors with keys X'80' through X'FE' have a meaning that is unique to the major vector in which they are used. They are defined following each major vector.

Subvectors with keys X'00' through X'7F' are referred to as common subvectors. Their meaning is independent of the major vector in which they are used. They are defined in "MS Common Subvectors" on page 8-23.

Subvectors may appear in any order within a major vector unless otherwise stated.

MS Major Vector Formats

Alert (X'0000') MS Major Vector

PU T2-->SSCP

This major vector provides unsolicited notification of a problem or impending problem, type of problem, identification of the cause, and identification of the component that caused the problem.

0-1	Length (n+1), in binary, of this MS major vector
2-3	Key: X'0000'
4-n	MS subvectors, as described (using zero-origin indexing) in "MS Common Subvectors" on page 8-23 for subvector keys X'00' - X'7F', and in "Alert MS Subvectors" on page 8-11 for subvector keys X'80' - X'FE'.

Note: The following subvector keys may be used as indicated:

Subvector	Presence in NMVT Alert (X'0000') Major Vector	
Text Message (X'00')	O	
Date/Time (X'01)	CP	Note 1
Hierarchy Name List (X'03')	CP	Note 2
SNA Address List (X'04')	CP	Note 3
Product Set ID (X'10')	P(n)	Note 4
Relative Time (X'42')	CP	Note 5
Correlation (X'43')	CP	Note 6
LAN Link Connection Subsystem Data (X'51')	CP	Note 7
Basic Alert (X'91')	P	
Detail Qualifier (X'A0' or X'A1')	O(n)	

Key:

P	Present one time
P(n)	Present one or more times
CP	Conditionally present one time (See Notes for conditions.)
O	Optionally present one time
O(n)	Optionally present one or more times

Notes:

1. If the PU sending the Alert major vector has the capability of providing it, it places this subvector in the NMVT. See Note 5.

2. This subvector is present in the NMVT instead of, or in addition to, the SNA Address List subvector if the origin of the Alert condition cannot be represented in the SNA Address List subvector.

3. This subvector is present when it is necessary to identify, with an SNA address, the origin of the Alert condition. If the origin of the Alert condition is the PU sending the Alert, this subvector is not present.

4. An instance of this subvector describing the PU sending the Alert is always present. A second instance is present if the origin of the

Alert condition is a hardware or software product, and is not the PU sending the Alert. If present, this subvector is placed immediately after the first instance of the X'10' subvector.

5. If the PU sending the Alert cannot provide a Date/Time subvector, it places this subvector in the NMVT instead.

6. This subvector is present if the Alert sender has preserved detailed data, e.g., a storage dump, to which the Alert must be correlated.

7. This subvector is present when the Alert reports an error on a LAN, and the node sending the Alert is attached to the LAN.

Alert MS Subvectors

Basic Alert (X'91') Alert MS Subvector

This subvector transports Alert information, including an index to predefined screens.

0 Length (p+1), in binary, of the Basic Alert subvector

1 Key: X'91'

2 Flags:
bit 0, initiation indicator:
    0 Alert was not directly initiated by an operator action
    1 Alert was initiated by an operator action
bits 1-3, reserved
bits 4-7, retired

3 Alert type:
X'01' permanent loss of availability: a loss of availability to the end user that is not recovered from without intervention external to the reporting product
X'02' temporary loss of availability: a momentary loss of availability that will probably be noticed by the end user, yet is recovered from without intervention external to the reporting product
X'03' performance: a recognized measurement of response time has exceeded a predetermined threshold
X'04' operator intervention required: the intervention of an operator is required to restore proper operational capability to the resource
X'05'-X'09' retired
X'0A' notification: a loss of availability to the end user is impending but has not yet happened
X'0B'-X'0E' retired
X'0F' delayed: the sender is reporting a previously detected alertable condition that prevented reporting when detected

4 General cause code: indicates the general classification and cause of the exception condition:
X'01' hardware or microcode (not distinguished): the Alert condition was caused by either a hardware (machine or equipment) failure, or a microcode failure, but the specific cause cannot be determined.
X'02' software: the Alert condition was caused by a software (programming) failure or malfunction.
X'03' retired

X'04'-X'05' reserved

X'06' media (e.g., tape, disk, diskette, paper): a failure, imperfection, or defect in the media.
Note: This code is used for cases where a particular area of a tape, disk or diskette cannot be read or written but other areas are operational. It is also used for torn or jammed forms or paper. It is not used for cases where the medium is not present or the wrong medium, e.g., the wrong size forms, are present; these cases are indicated by X'17' (operator intervention required).

X'07' hardware or software (not distinguished): the Alert condition was caused by either a hardware (machine or equipment) failure, or a software (programming) failure but the specific cause cannot be determined.

X'08'-X'09' retired

X'0A' media or hardware (not distinguished): the Alert condition was caused by either a hardware (machine or equipment) failure, or a failure, imperfection, or defect in the media, but the specific cause cannot be determined.

X'0B' hardware: the Alert condition was caused by a hardware (machine or equipment) failure or malfunction.

X'0C' microcode: the Alert condition was caused by a microcode failure or malfunction.
Note: This code is not used for ROS chips that are packaged in field replaceable units (FRUs) or customer replaceable units (CRUs) and are serviced in the same manner as hardware logic is serviced. X'0B' (hardware) is used in those cases.

X'0D' protocol above link level: the Alert condition was caused by an SNA protocol error.
Note: This code point reports protocol errors that are caused by incorrect programming, for example, failure to include a BB bit on the first RU when in BETB state on a session that must use BRACKET protocol.

X'0E' link-level protocol: the Alert condition was caused by a link level protocol error.
Note: Errors such as send/receive count errors that can be caused by missing a message due to line hits do not fall into this category; they are indicated by X'0B' (hardware).

X'0F' undetermined: the cause of the Alert condition cannot be determined.

X'10' external facilities change or restriction: the number called is temporarily unobtainable
Note: This code point is used by X.21 networks.

X'11' user: the Alert condition was caused by an incorrect action taken by a user.
Note: Unavailability due to a device being varied offline does not fall into this category; it is indicated by X'13' (component offline).

X'12' system generation, customization, or installation consistency problem: the Alert condition was caused by an invalid system definition or customizing parameter, or by a mismatch between a system definition or customizing parameter and the hardware.
Note: This code is used only in those cases that typically are not corrected by the action of the local operator.

X'13' component offline: the Alert condition was caused by a component being offline.
X'14' component busy: the Alert condition was caused by a component being busy.
X'15' external power failure: the Alert condition was caused by an external power failure.
X'16' thermal problem: the Alert condition was caused by temperature that is not within recommended specifications.
X'17' operator intervention required: the Alert condition was caused because action is required by an operator.
Note: Unattended devices will always Alert when operator intervention is required. Attended devices will not Alert until the local operator has time to perform the required action. After the device-allocated time has expired for attended devices, the device has the option of sending an Alert.
X'18' microcode or software (not distinguished): the Alert condition was caused by either a software (programming) failure or malfunction or a microcode failure but the specific cause cannot be determined.

5-6 Specific component code: indicates the generic type of component, subcomponent, or logical resource that can be most closely related to the exception condition. The component indicated may be the generic type of the "target" or it may be a subcomponent of the target. The terms "local" and "remote" used below, refer to the perspective of the Alert originator. Defined codes are:

X'0001' base processor
X'0002' service processor
X'0003' reserved
X'0004' main storage
X'0005' disk device
X'0006' printer
X'0007' card reader and/or punch
X'0008' tape device
X'0009' keyboard
X'000A' selector pen
X'000B' magnetic stripe reader
X'000C' display/printer
X'000D' display device
X'000E' remote product: used when a product to which the Alert generator is linked (in any form) has caused an Alert condition and the generic product type cannot be determined
X'000F' power supply internal to this product
X'0010' I/O attached controller
X'0011' communication controller scanner
X'0012' communication link adapter
X'0013' reserved
X'0014' channel adapter
X'0015' loop adapter
X'0016' adapter for directly attaching devices
X'0017' reserved
X'0018' channel (direct memory access channel)
X'0019' link: used only when common-carrier equipment cannot be distinguished from customer equipment
X'001A' link: common-carrier equipment
X'001B' link: customer equipment

X'001C'	loop: used only when common-carrier equipment cannot be distinguished from customer equipment
X'001D'	loop: common-carrier equipment
X'001E'	loop: customer equipment
X'001F'	X.21 link connection external to this product
X'0020'	X.25 network connection external to this product
X'0021'	local X.21 interface (DTE-DCE)
X'0022'	local X.25 interface (DTE-DCE)
X'0023'	local modem
X'0024'	remote modem
X'0025'	local modem interface (DTE-DCE)
X'0026'	remote modem interface (DTE-DCE)
X'0027'	local modem link monitor
X'0028'	remote modem link monitor
X'0029'	local modem link monitor interface
X'002A'	remote modem link monitor interface
X'002B'-X'0031'	reserved
X'0032'	remote modem or modem interface or remote product
X'0033'	transmission medium or remote modem
X'0034'	SDLC data link control component
X'0035'	BSC data link control component
X'0036'	start/stop data link control component
X'0037'-X'0043'	reserved
X'0044'	cluster controller or device
X'0045'	local link monitor or modem interface
X'0046'	reserved
X'0047'	card reader/punch or display/printer
X'0048'	controller application program
X'0049'	keyboard or display
X'004A'	storage control unit
X'004B'	storage control unit or storage control unit channel
X'004C'	storage control unit or controller
X'004D'	control unit (other than storage control unit)
X'004E'-X'0051'	reserved
X'0052'	maintenance device
X'0053'	maintenance device interface
X'0054'	reserved
X'0055'	control program
X'0056'	application subsystem on top of control program
X'0057'	telecommunication access method
X'0058'	application program (other than application subsystem)
X'0059'	communication controller program
X'005A'-X'005F'	reserved
X'0060'	X.25 network interface: DCE to first interface node in X.25 network
X'0061'	disk device with nonremovable media
X'0062'	disk device with removable media
X'0063'	control tailed modem
X'0064'	reserved
X'0065'	remote tailed modem
X'0066'	remote tailed modem interface
X'0067'	sensor I/O unit
X'0068'	magnetic stripe reader/encoder
X'0069'	check (bank) reader

X'006A'	document feed mechanism
X'006B'	coin feed mechanism
X'006C'	envelope depository
X'006D'	timer adapter
X'006E'	encryption/decryption adapter
X'006F'	outboard, user programmable processor
X'0070'	cable connecting local device to local adapter
X'0071'-X'007F'	reserved
X'0080'	Token-ring LAN error
X'0081'	Carrier Sense Multiple Access (CSMA/CD) LAN error
X'0082'-X'00FE'	reserved
X'00FF'	undetermined (the problem cannot be isolated to one of the above generic component types)

7-8	Alert description code: a code that provides an index to predefined text that explains the condition that caused the Alert Note: This field is product dependent.
9-10	User Action Code: a code that provides an index to predefined screens that can include predefined text and variable fields for MS User Action Qualifier subvectors Note: This field is product dependent.
11-12	Detail text reference code: a code that provides an index to predefined screens that can include predefined text and variable fields for MS Detail Qualifier subvectors Note: This field is product dependent.
13(=p)	Retired

Detail Qualifier (EBCDIC) (X'A0') Alert MS Subvector

This subvector supplies variables for the Alert function in EBCDIC form that can be inserted on the Alert Detail screens. This subvector and the Detail Qualifier (hexadecimal) subvector (X'A1') are identical in function and format except that this subvector contains EBCDIC codes. Note: The detail qualifier (X'A0'-X'A1') subvectors are displayed in the order that they appear in the Alert major vector.

0	Length (p+1), in binary, of the Detail Qualifier subvector
1	Key: X'A0'
2-p	Detail qualifier: a type-AE symbol-string that qualifies a reference on the Alert Detail screen Note: Each qualifier is p-1 bytes in length, but only one qualifier is used per Detail Qualifier subvector. All qualifiers include only codes, numbers, or internationally recognized terms that do not require translation. The coding is not interpreted by the Alert display mechanism.

Detail Qualifier (Hexadecimal) (X'A1') Alert MS Subvector

This subvector supplies variables for the Alert function in hexadecimal form that can be inserted on the Alert Detail screens. This subvector and the Detail Qualifier (EBCDIC) subvector (X'A0') are identical in function and format except that this subvector contains codes in hexadecimal. Note: The detail qualifier (X'A0'-X'A1') subvectors are displayed in the order that they appear in the Alert major vector.

0	Length (p+1), in binary, of the Detail Qualifier subvector
1	Key: X'A1'
2-p	Detail qualifier: a type-G symbol-string

Request Response Time Monitor (X'8080') MS Major Vector

SSCP-->PU T2

This major vector enables or disables response time monitoring, transports RTM parameters, and transports a request for RTM data and status from a device.

0-1	Length (n+1), in binary, of this MS major vector
2-3	Key: X'8080'
4-n	MS subvectors, as described (using zero-origin indexing) in "MS Common Subvectors" on page 8-23 for subvector keys X'00' - X'7F', and in "Request Response Time Monitor Subvectors" on page 8-17 for subvector keys X'80' - X'FE'

Note: The following subvector keys may be used as indicated:

Subvector	Presence in NMVT Request RTM (X'8080') Major Vector	
SNA Address List (X'04')	CP	Note 1
*RTM Request (X'92')	P	
RTM Control (X'94')	CP	Note 2

Key:

*	Command Subvector (for PU parsing)
P	Present one time
CP	Conditionally present one time (See Notes for conditions.)

Notes:

1. This subvector is present in the NMVT containing an X'8080' major vector when the request is for a specific LU (i.e., identified in the X'04' subvector) associated with the PU processing the request. This subvector is not present when the request is to apply to all LUs associated with the PU processing the request.

2. This subvector is present when RTM parameters are being set. If present, it immediately follows the RTM Request (X'92' subvector).

Request Response Time Monitor Subvectors

RTM Request (X'92') Request RTM MS Subvector

> This subvector requests RTM data and status or accompanies an RTM control subvector.

0	Length (p+1), in binary, of this subvector
1	Key: X'92'
2	Request indicators (bit is set to 1 to request that the function be performed): bit 0, reset RTM data for the target LU upon reply transmission or immediately if no reply is requested bit 1, Retrieve data and status for all LUs with accumulated RTM data. bit 2, retired bit 3, Retrieve data and status for the LU specified in the SNA Address List (X'04') MS common subvector also included in this major vector bit 4, apply the RTM Control (X'94') MS subvector also included in this major vector to all LUs Note: If this bit is set to 1, the RTM Control (X'94') MS subvector will be present. If this bit is set to 0 and the RTM Control (X'94') subvector is present, the SNA Address List (X'04') MS common subvector will be present. bits 5-6, retired
3(=p)	Reserved

Request Type	Subvectors present in the Request RTM (X'8080') major vector	Bits B1	B3
Retrieve data for all LUs with accumulated data	92	1	0
Retrieve for specified LU	92, 04	0	1
Set parameters for all LUs	92, 94	0	0
Set parameters for specified LU	92, 94, 04	0	0

Figure 8-1. Setting of Bits 1 and 3 of Byte 2 of the RTM Request (X'92') Subvector

RTM Control (X'94') Request RTM MS Subvector

This subvector controls RTM data accumulation.

```
0       Length (p+1), in binary, of this subvector
1       Key:  X'94'
2-3     RTM status and control change mask (bit is set to 1 if the setting
        specified by the corresponding RTM status and control indicator in bytes
        4-5 should be used):
        bits 0-8, mask bits corresponding respectively to indicator bits 0-8 in
                  bytes 4-5
        bits 9-15, reserved
4-5     RTM status and control indicators (bit is set to 1 to activate the
        function or 0 to deactivate it):
        bit  0, RTM measurement active
        bit  1, return data unsolicited on session deactivation
        bit  2, return data unsolicited on counter overflow
        bit  3, retired
        bit  4, set the RTM measurement definition using byte 8
        bit  5, set the RTM response time measurement boundaries using bytes 9 and
                16-m
        bit  6, retired
        bit  7, local display of RTM data
        bit  8, retired
        bits 9-15, reserved
6       Reserved
7       Retired
8       RTM measurement definition--defines when the response-time measurement
        will begin and end for each exchange between session partners:
        X'01' measured from the attention or action key depression to the arrival
              back at the LU of the first character that can alter the
              presentation space
        X'02' measured from the attention or action key depression until the LU is
              ready to accept input from its end user
```

	X'03' measured from the attention or action key depression to the receipt and processing back at the LU of Change Direction (CD) or End Bracket (EB)
9	Response-time unit of measure: X'00' 100 milliseconds X'01'-X'7F' retired X'90' retired X'A0' retired X'C0' retired X'D0' retired
10-15	Reserved
16	RTM data collection parameters: bits 0-3, reserved bits 4-7, binary number of 2-byte boundaries in bytes 17-p
17-p	A set of response-time measurement boundaries, specified in binary (as units of response-time units of measure described by byte 9) and increasing in order of magnitude; thus, response-time data is collected for intervals (0 < r1 <= b1*u), (b1*u < r2 <= b2*u), ... up to (b4 < r5), where bi is the value of the boundary i, ri is the response-time being measured for interval i, and u is the unit of measure described by byte 9.
17-18	Boundary 1
19-20	Boundary 2
21-p	Additional boundaries as required (the total number is defined by byte 16, bits 4-7), up to a maximum of 4

Response Time Monitor (X'0080') MS Major Vector

PU T2-->SSCP

This major vector transports RTM data. This data includes the collected response time data and current RTM status.

0-1	Length (n+1), in binary, of this MS major vector
2-3	Key: X'0080'
4-n	MS subvectors, as described (using zero-origin indexing) in "MS Common Subvectors" on page 8-23 for subvector keys X'00' - X'7F', and in "Response Time Monitor Subvectors" on page 8-21 for subvector keys X'80' - X'FE'

Note: The following subvector keys may be used as indicated:

Subvector	Presence in NMVT RTM (X'0080') Major Vector	
Date/Time (X'01)	CP	Note 1
SNA Address List (X'04')	CP	Note 2
Relative Time (X'42')	CP	Note 3
Data Reset Flag (X'45')	CP	Note 4
Sense Data (X'7D')	CP	Note 5
RTM Status Reply (X'91')	CP	Note 6
RTM Data (X'93')	CP	Note 7

Key:

P	Present one time
CP	Conditionally present one time (See Notes for conditions.)

Notes:

1. If the PU sending the X'0080' major vector has the capability of providing it, it places this subvector in the NMVT.

2. This subvector is present when positively replying to a request for RTM data and status if RTM data has been accumulated.

3. If the PU sending the X'0080' cannot provide a Date/Time subvector, it places this subvector in the NMVT instead.

4. This subvector is present in an X'0080' major vector when a set of counters has been reset, either as a result of a request or when sent unsolicited.

5. This subvector is present when a Request RTM major vector cannot be processed, or when requested data cannot be gathered and the PU sending this major vector has elected to send sense data in a reply instead of a negative response.

6. This subvector is present when replying to a request for RTM data and status when data is present or when the reply is for the last LU (sent as a result of a request to retrive data for all LUs with accumulated data) and no data is present.

7. This subvector is present when positively replying to a request for RTM data and status if RTM data has been accumulated.

Response Time Monitor Subvectors

RTM Status Reply (X'91') RTM MS Subvector

> This subvector transports the current status of RTM function for a device.

0 Length (p+1), in binary, of this subvector

1 Key: X'91'

2 Reply indicators (bit is set to 1 to indicate that the assertion is true):
- bit 0, reserved
- bit 1, data not included
- bit 2, an RTM data request has been issued for an LU that has its RTM function disabled
- bit 3, this is the first RTM reply since session activation (used to initiate a recording of the session partner names and the correlation value in bytes 7-8 of this subvector while there is reasonable assurance that the session is active); on subsequent replies the correlation value will be used to associate data from the same LU-LU session
- bit 4, an LU-LU session activation or deactivation has occurred at least once while the included RTM data was being accumulated
- bits 5-7, reserved

3 Reason for unsolicited reply, if any (bit is set to 1 to indicate the appropriate reason):
- bit 0, the session for this resource has ended and is enabled unsolicited-reply-on-session-deactivation
- bit 1, a counter for this LU has overflowed and unsolicited-reply-on-counter-overflow is enabled
- bit 2, retired
- bits 3-5, reserved
- bit 6, retired
- bit 7, reserved

4 Reason for potential loss of RTM data, if any (bit is set to 1 to indicate the reason):
- bit 0, reserved
- bit 1, an overflow has occurred on at least one counter and updating for all of this LU's counters has been stopped to retain the relative significance of the data
- bit 2, this LU has been reset since the last reply was sent
- bit 3, a new session was activated before data for the previous session could be transmitted: loss of data for the new session may have occurred; updating for all of this LU's counters has been stopped to retain the relative significance of the data
- bit 4, the RTM definition or response time measurement boundaries have been changed by a request that did not solicit the RTM data and RTM accumulation was active for this LU: any data collected since the last data request has been lost
- bits 5-7, reserved

5-6 RTM status when this subvector was constructed (a bit set to 1 indicates that the function was active):

bit 0, RTM measurement active
bit 1, data to be sent unsolicited on session deactivation
bit 2, data to be sent unsolicited on counter overflow
bit 3, retired
bit 4, RTM definition was set by the control point
bit 5, RTM boundaries were set by the control point
bit 6, retired
bit 7, local display of RTM data
bit 8, retired
bits 9-15, reserved

7-8(=p) Correlation value: a unique 2-byte value, generated by the PU, that is retained and used in all RTM replies dealing with the same LU-LU session from session activation through the subsequent session deactivation

RTM Data (X'93') RTM MS Subvector

This subvector transports solicited or unsolicited RTM data.

0 Length (q+9), in binary, of this subvector

1 Key: X'93'

2 RTM measurement definition in effect:

X'01' measured from the attention or action key depression to the arrival back at the LU of the first character that can alter the presentation space
X'02' measured from the attention or action key depression until the LU is ready to accept input from its end user
X'03' measured from the attention or action key depression to the receipt and processing back at the LU of Change Direction (CD) or End Bracket (EB)

3 Response time unit of measure:

X'00' 100 milliseconds
X'01'-X'7F' retired
X'90' retired
X'A0' retired
X'C0' retired
X'D0' retired

4-5 Reserved

6-7 Retired

8 RTM data collection parameters:

bits 0-3, the number, in binary, of response time measurement boundaries returned; all boundaries that were set previously will be returned in this subvector
bits 4-7, the number, in binary, of boundary sets for which valid data was collected (overflow data--a count of response times exceeding the maximum boundary--is not included in this number but is always present)

9-p A set of response-time measurement boundaries as previously set at the LU or by the RTM Control (X'94') MS subvector (specified in binary as units of response-time units of measure described by byte 9) and increasing in order of magnitude; thus, response-time data is collected for intervals (0 < r1 <= b1*u), (b1*u < r2 <= b2*u), ... up to (b4 < r5), where bi is the value of the boundary i, ri is the response-time being measured for interval i, and u is the unit of measure described by byte 9.

9-10	Boundary 1
11-12	Boundary 2
13-p	Additional boundaries as required to equal the number of boundaries set previously and specified by byte 8, bits 0-3
p+1-q	The number of measured exchanges for each response-time interval: the number of exchanges whose duration was within an interval's boundaries is reported in binary separately for each interval.
p+1-p+2	Number of exchanges in the (0,b1) range
p+3-p+4	Number of exchanges in the (b1+1,b2) range
p+5-q	Additional exchange counts to satisfy the number of boundaries defined , up to a maximum of 4
q+1-q+2	Overflow: the number of exchange durations greater than the maximum boundary specified
q+3-q+6	Total of all individual times for all exchanges measured and reported by this record, including overflow, in the measurement units defined by byte 3
q+7-q+8	Last measured exchange duration in the measurement units defined by byte 3

MS Common Subvectors

The common MS subvectors are defined as follows (using zero-origin indexing):

Text Message (X'00') MS Common Subvector

> This MS common subvector transports EBCDIC data.

0	Length (p+1), in binary, of the Text Message subvector
1	Key: X'00'
2-p	Text message in EBCDIC Note: The coded character set that may be transported in this field is dependent on the implementation that provided the text or allowed an operator to input the text, as well as the output device used by the Alert processor. The installation management ensures the compatibility of these products.

Date/Time (X'01') MS Common Subvector

> This MS common subvector is used by the PU for time-stamping the NMVT in which it is carried.

0	Length (p+1), in binary, of the Date/Time subvector
1	Key: X'01'
2-p	One or more of the following subfields:

X'10' Local Date/Time (required subfield)
X'20' Greenwich Mean Time Offset

Local Date/Time (X'10') Date/Time Subfield

> This subfield transports the local date and time of the creation of the major vector.

0	Length (q+1), in binary, of the Local Date/Time subfield
1	Key: X'10'
2-4	Local date
2	Year, in binary, consisting of the last two digits of the year
3	Month, in binary (X'01'-X'0C')
4	Day, in binary (X'01'-X'1F')
5-q	Local time
5	Hours, in binary (X'00'-X'17')
6	Minutes, in binary (X'00'-X'3B')
7	Seconds, in binary (X'00'-X'3B')
8-q	Optional extension of time: a binary value to provide finer granularity than seconds

Greenwich Mean Time Offset (X'20') Date/Time Subfield

> This subfield transports the Greenwich Mean Time (GMT) offset of the node that originated the management services RU (i.e., the origin node). It is optionally included in a major vector by the origin node or by the control point in whose domain the origin node resides.

0	Length (q+1), in binary, of the Greenwich Mean Time Offset subfield
1	Key: X'20'
2-3(=q)	Time zone adjustment to Greenwich Mean Time: an interval of time to be added to, or subtracted from, the local time given in the Local Date/Time (X'10') subfield to adjust that time to Greenwich Mean Time bit 0, positive or negative adjustment indicator: 0 adjustment to be added to the local time (i.e., all time zones westward, between the Greenwich time zone and the International Date Line) 1 adjustment to be subtracted from the local time (i.e., all time zones eastward, between the Greenwich time zone and the International Date Line) bits 1-3, reserved bits 4-7, number of hours of adjustment, in binary (X'0'-X'C') bits 8-15, number of minutes of adjustment, in binary (X'00'-X'3B')

Hierarchy Name List (X'03') MS Common Subvector

> This MS common subvector identifies target resources, other than the reporting PU, that are within the same domain as the origin PU, but cannot be represented in the SNA Address List subvector.

0	Length (p+1), in binary, of the Hierarchy Name List subvector
1	Key: X'03'
2	Reserved
3	Number, in binary, of name entries in the hierarchy name list.
4-p	Hierarchy Name List Entries (1 to 5 entries may be present)

Note: Each entry contains a Name field and a Resource Type field, and has the following form (shown zero-origin):

0	Length (q+1), in binary, of the following name plus this Length field
1-q	Name of resource in upper-case alphanumeric EBCDIC characters Note: Resource name never exceeds eight characters.
q+1-q+4	Resource type identifier: category in which the resource (named in bytes 1-q) belongs: X'C1C4C1D7' adapter X'C3C2E740' computerized branch exchange X'C3C2E4E2' carrier sense multiple access with collision detection (CSMA/CD) bus X'D3C9D5C5' communication link X'C3E3D9D3' controller X'C4C9E2D2' disk X'C4E2D2E3' diskette

X'D2E8C2C4'	keyboard
X'D3C1D540'	local area network (LAN)
X'D3D6D6D7'	loop
X'D7C2D440'	personal banking machine
X'D7D6E240'	point-of-sale unit
X'D7C2E740'	private branch exchange
X'D7D3E3D9'	plotter
X'D7D9E3D9'	printer
X'D9C9D6C7'	token ring
X'E2D74040'	service point
X'E3F1D9D4'	T1 resource manager
X'E3C1D7C5'	tape
X'E3C1E440'	teller assist unit
X'E3C2E4E2'	token bus
X'E3C5D9D4'	terminal
X'C4C5E540'	unspecified device

SNA Address List (X'04') MS Common Subvector

> This MS common subvector is used in both request and data NMVTs. In a request NMVT, it identifies one or more destinations of the MS request when the destination is not the PU addressed in the Transmission Header (TH). In a data NMVT, it identifies the resource associated with the data when the resource is not the PU addressed in the TH.
>
> If present, this subvector is the first subvector within the MS major vector.

0 Length (p+1), in binary, of the SNA Address List subvector

1 Key: X'04'

2 Address Count: For address entity format types X'00', X'40', and X'80', a binary number indicating the number of individual addresses present in the X'04' subvector. This field is set to X'00' for all other address entity format types.
Note: This field provides a count of individual addresses; thus for format X'40', each pair of addresses counts as two.

3 Address entity format type:
X'00' address format is one or more single local addresses
X'40' address format is one or more pairs of session-partner local addresses, each pair identifying a session
X'80' address format is one or more single network addresses
X'A0' address format is one or more network-qualified address pairs, each pair identifying a session
X'C0' address format is one or more pairs of session partner network addresses, each pair identifying a session

4-p Address entities: one or more address entities, each having one of the formats defined below (zero-origin):

- For a single local address (byte 3 = X'00'):

0-4 Reserved
5 Local address

- For a pair of session-partner local addresses (byte 3 = X'40'):

0-4	Reserved
5	Local address of SLU
6	Retired
7-11	Reserved
12	Session index (local address of PLU)

- For a single network address (byte 3 = X'80'):

0-5	Network address

- For a network-qualified address pair (byte 3 = X'A0'):

0-5	Network address of NAU1
6-11	Network address of NAU2
12-19	Network ID of the subnetwork in which the above addresses are valid

- For a pair of session-partner network addresses (byte 3 = X'C0'):

0-5	Network address 1
6	Retired
7-12	Network address 2

Product Set ID (X'10') MS Common Subvector

> This MS common subvector identifies one or more products that implement a network component.

0	Length (p+1), in binary, of the Product Set ID subvector
1	Key: X'10'
2	Retired
3-p	Network product identifier consisting of one or more Product ID (X'11') MS common subvectors, as described below (using zero-origin indexing). Each Product ID (X'11') MS Common Subvector uniquely identifies a product. Products fall into two categories: hardware (with or without microcode) and software.

Product Identifier (X'11') MS Common Subvector

> This MS common subvector uniquely identifies a single product. A product may consist of electronic circuitry (hardware), executable instructions (software), or both (in the case of hardware containing microcode).

0	Length (q+1), in binary, of the Product Identifier subvector
1	Key: X'11'
2	bits 0-3, Reserved bits 4-7, Product classification: X'1' IBM hardware X'3' IBM or non-IBM hardware (not distinguished) X'4' IBM software X'9' non-IBM hardware X'C' non-IBM software X'E' IBM or non-IBM software (not distinguished)
3-q	One or more subfields containing product- and installation-specific information on hardware, microcode, and software. Note: The subfields may be used as indicated in the table on the following page.

Sub-field	HW or SW X'11' (Note 1)	X'0000' Alert (Note 2)		XID3 (Note 3)	
		Sender	Resource		
X'00'	HW	P	P	P	
X'01'	HW	CP	CP	CP	Note 4
X'02'	SW	CP	CP	CP	Note 5
X'04'	SW	CP	CP	CP	Note 6
X'06'	SW	P	P	O	
X'07'	SW	—	CP	O	Note 7
X'08'	SW	CP	CP	CP	Note 6
X'09'	SW	—	CP	O	Note 7

Conditions of Subfield Presence in Product Identifier Subvector

Key:

—	Not present
P	Present one time
CP	Conditionally present one time
O	Optionally present one time

Subfield Names:

X'00' — Hardware Product Identifier
X'01' — Emulated Product Identifier
X'02' — Software Product Serviceable Component Identifier
X'04' — Software Product Common Level
X'06' — Software Product Common Name
X'07' — Software Product Customization Identifier
X'08' — Software Product Program Number
X'09' — Software Product Customization Date and Time

Notes:

1. The hardware X'11' Product Identifier subvector is present when the Product classification nibble (byte 2, bits 4-7) is X'1', X'3', or X'9'. The software X'11' Product Identifier subvector is present when this nibble is X'4', X'C', or X'E'.

2. If a PU is sending an Alert for itself, a single Product Set ID (X'10') subvector is present. This is the "Indicated Resource" for purposes of reading this matrix. If the PU is reporting on an Alert for an attached device, two X'10' subvectors are present, in the following order:

   a. "Alert Sender"--identifies the PU sending the Alert

   b. "Indicated Resource"--identifies the resource that is being Alerted upon

3. In XID3, the Hardware and Software X'11' subvectors are carried in the X'10' Control Vector rather than the X'10' MS Common subvector.

4. This subfield is present in the hardware X'11' when a product is emulating another hardware product.

5. This subfield is present in the software X'11' for products assigned a component ID by the National Service Division. For products not assigned a component ID, the X'04' and X'08' subfields must be present. (See note 6)

6. The X'04' and X'08' subfields are required in the software X'11' if the X'02' subfield is not present. They are optional when the X'02' is present. (See note 5.) If, however, the software identified is a customer-written application, only the X'08' subfield is present.

7. One of the X'07' and X'09' subfields is required in the software X'11' for software products modifiable by the customer.

Hardware Product Identifier (X'00') Product Identifier Subfield

> This subfield uniquely identifies an instance of a hardware product.

0 Length (r+1), in binary, of the Hardware Product Identifier subfield

1 Key: X'00'

2 Format type:

X'10' product instance is identified by a serial number (i.e., IBM plant of manufacture and sequence number) unique by machine type

X'11' product instance is identified by a serial number (i.e., IBM plant of manufacture and sequence number) unique by machine type and model number

X'12' product instance is identified by a serial number (i.e., IBM plant of manufacture and sequence number) unique by machine type (as in Format X'10' above). This format provides the model number not to uniquely identify a product instance but, for the purpose of additional information only.

X'13' retired

X'40' retired

X'41' retired

3-r	Product identification Note: The originator of a message unit (e.g., NMVT, XID) reporting for another product that does not supply information required for the Hardware Product Identifier subfield inserts binary 0's into the appropriate fields (except for the Machine Type field where EBCDIC 0's [X'F0'] are inserted) of the Product Identification field to indicate that no identification information is available.
	• Format X'10'
3-6	Machine type: four numeric EBCDIC characters
7-8	IBM plant of manufacture: two numeric EBCDIC characters
9-15(=r)	Sequence number: seven upper-case alphanumeric EBCDIC characters, right-justified, with EBCDIC 0's (X'F0') fill on the left
	• Format X'11'
3-6	Machine type: four numeric EBCDIC characters
7-9	Machine model number: three upper-case alphanumeric EBCDIC characters
10-11	IBM plant of manufacture: two numeric EBCDIC characters
12-18(=r)	Sequence number: seven upper-case alphanumeric EBCDIC characters, right-justified, with EBCDIC 0's (X'F0') fill on the left
	• Format X'12'
3-6	Machine type: four numeric EBCDIC characters
7-9	Machine model number: three upper-case alphanumeric EBCDIC characters
10-11	IBM plant of manufacture: two numeric EBCDIC characters
12-18(=r)	Sequence number: seven upper-case alphanumeric EBCDIC characters, right-justified, with EBCDIC 0's (X'F0') fill on the left

Emulated Product Identifier (X'01') Product Identifier Subfield

> This subfield identifies the hardware of the product being emulated in sufficient detail to allow problem determination

0	Length (r+1), in binary, of the Emulated Product Identifier subfield
1	Key: X'01'
2-5	Machine type of product being emulated: four numeric EBCDIC characters
6-8(=r)	Model number of product being emulated: three upper-case alphanumeric EBCDIC characters

Software Product Serviceable Component Identifier (X'02') Product Identifier Subfield

> This subfield transports the serviceable component identifier and release level as assigned by service personnel.

0	Length (r+1), in binary, of the Software Product Serviceable Component Identifier subfield
1	Key: X'02'

2-10	Serviceable component identifier: nine upper-case alphanumeric EBCDIC characters
11-13(=r)	Serviceable component release level: three numeric EBCDIC characters

Software Product Common Level (X'04') Product Identifier Subfield

> This subfield transports the common version, release, and modification level numbers as given in the product announcement documentation.

0	Length (r+1), in binary, of the Software Product Common Level subfield
1	Key: X'04'
2-3	Common version identifier (numeric EBCDIC characters, right-justified with X'F0' fill on left)
4-5	Common release identifier (numeric EBCDIC characters, right-justified with X'F0' fill on left)
6-7(=r)	Common modification identifier (numeric EBCDIC characters, right-justified with X'F0' fill on left)

Software Product Common Name (X'06') Product Identifier Subfield

> This subfield transports the software common name as given in the product announcement documentation.

0	Length (r+1), in binary, of the Software Product Common Name subfield
1	Key: X'06'
2-r	Up to thirty upper-case alphanumeric EBCDIC characters identifying the software product common name

Software Product Customization Identifier (X'07') Product Identifier Subfield

> This subfield identifies a set of executable instructions, customized to the user's environment.

0	Length (r+1), in binary, of the Software Product Customization Identifier subfield
1	Key: X'07'
2-r	Customization identifier: up to eight alphanumeric EBCDIC characters

Software Product Program Number (X'08') Product Identifier Subfield

> This subfield transports either the program product number as assigned by distribution personnel, or a substitute value supplied by a user-written software program.

0	Length (r+1), in binary, of the Software Product Program Number subfield
1	Key: X'08'
2-8(=r)	Program product number: seven upper-case alphanumeric EBCDIC characters Note: A user-written application program does not send a program product number in this field. Instead it sends one of 16 substitute values

comprised of seven upper-case alphanumeric EBCDIC characters having the following form: characters 1-4 are the letters USER; character 5 is one of the characters 0-9, or A-F; characters 6-7 are space (X'40') characters. Installation managers have the sole responsibility for managing the usage of these substitute values within their networks.

Software Product Customization Date and Time (X'09') Product Identifier Subfield

> This subfield identifies the date and time that a set of executable instructions was customized to the user's environment.

0	Length (r+1), in binary, of the Software Product Customization Date and Time subfield.
1	Key: X'09'
2	Year in unsigned packed decimal
3-4	Julian day in unsigned packed decimal, right-justified with 0's as fill
5	Hour in unsigned packed decimal (24-hour clock)
6(=r)	Minute in unsigned packed decimal

Relative Time (X'42') MS Common Subvector

> This MS common subvector indicates when a record was created relative to other records created by the originating component.

0	Length (p+1), in binary, of the Relative Time subvector
1	Key: X'42'
2	Time increment of measure: X'00' tenths of a second X'01'-X'7F' a number that, when divided into the timer data (in bytes 3-6), converts the value to seconds X'90' microseconds X'A0' milliseconds X'C0' minutes (not used in Alerts) X'D0' hours (not used in Alerts) X'EF' indicates time value is purely a sequence indicator showing relative order only
3-6(=p)	Time, in binary, having the measure defined by byte 2

Correlation (X'43') MS Common Subvector

> This MS common subvector is used to correlate an Alert with trace, storage dump, and log data records originating from the same source for a single incident.

0	Length (p+1), in binary, of the Correlation subvector
1	Key: X'43'
2	Correlator identification: bit 0, data type: 0 hexadecimal: correlation data is specified as a type-G symbol-string

	1 EBCDIC: correlation data is specified as a type-A symbol-string without the first-character restriction; i.e., 0-9 may be used as the first character
	bits 1-3, reserved
	bits 4-7, correlation type:
	X'1' correlator is related to trace data
	X'2' correlator is related to a storage dump
	X'3' correlator is related to a log
3-p	Correlation data (as specified by data type in byte 2): implementation-defined data.

Data Reset Flag (X'45') MS Common Subvector

This MS common subvector acknowledges that the reset function has been performed.

0	Length (p+1), in binary, of the Data Reset Flag subvector
1(=p)	Key: X'45'

LAN Link Connection Subsystem Data (X'51') MS Common Subvector

This MS common subvector transports data on the elements of the LAN link connection.

0	Length (p+1), in binary, of the LAN Link Connection Subsystem Data subvector
1	Key: X'51'
2-p	One or more subfields containing data specific to the link connection elements (listed by Key value below and described in detail following):

X'02'	Ring or Bus Identifier
X'03'	Local Individual MAC Address
X'04'	Remote Individual MAC Address
X'05'	LAN Routing Information
X'06'	Ring Fault Domain Description
X'07'	Beaconing Data
X'08'	Single MAC Address
X'09'	Fault Domain Error Weight Pair
X'23'	Local Individual MAC Name
X'24'	Remote Individual MAC Name
X'26'	Fault Domain Names
X'28'	Single MAC Name

Ring or Bus Identifier (X'02') LAN Link Connection Subsystem Data Subfield

This subfield transports the ring number (for a token-ring LAN) or the bus number (for a CSMA or token-bus LAN).

0	Length (q+1), in binary, of the ring or bus identifier subfield
1	Key: X'02'
2-3(=q)	Ring or bus number, in hexadecimal

Local Individual MAC Address (X'03') LAN Link Connection Subsystem Data Subfield

This subfield transports the address of the MAC within the node sending the MS major vector.

0	Length (q+1), in binary, of the local individual MAC address subfield
1	Key: X'03'
2-7(=q)	Local individual MAC address, in hexadecimal

Remote Individual MAC Address (X'04') LAN Link Connection Subsystem Data Subfield

This subfield transports the address of the MAC, part of the link connection, within the adjacent node.

0	Length (q+1), in binary, of the remote individual MAC address subfield
1	Key: X'04'
2-7(=q)	Remote individual MAC address, in hexadecimal

LAN Routing Information (X'05') LAN Link Connection Subsystem Data Subfield

This subfield transports the routing information used by a link.

0	Length (q+1), in binary, of the LAN routing information subfield
1	Key: X'05'
2-q	Routing information, not to exceed 18 bytes, in hexadecimal

Fault Domain Description (X'06') LAN Link Connection Subsystem Data Subfield

This subfield identifies a pair of LAN ring stations as a fault domain, i.e., the upstream and the downstream LAN ring stations and the cable between them.

0	Length (q+1), in binary, of the Ring Fault Domain Description subfield
1	Key: X'06'
2-7	Individual MAC address of downstream station, in hexadecimal
8-13(=q)	Individual MAC address of upstream station, in hexadecimal

Beaconing Data (X'07') LAN Link Connection Subsystem Data Subfield

This subfield specifies the type of beacon detected by the LAN adapter.

0	Length (q+1), in binary, of the Beaconing Data subfield
1	Key: X'07'
2(=q)	Beaconing type:
	X'01' type 1, recovery mode set
	X'02' type 2, signal loss
	X'03' type 3, streaming signal

Single MAC Address (X'08') LAN Link Connection Subsystem Data Subfield

This subfield transports the address of the MAC element associated with the failure.

0	Length (q+1), in binary, of the Single MAC Address subfield
1	Key: X'08'
2-7(=q)	Single MAC address, in hexadecimal

Fault Domain Error Weight Pair (X'09') LAN Link Connection Subsystem Data Subfield

This subfield indicates the severity of the problems reported by two MAC elements (LAN stations) belonging to a fault domain.

0	Length (q+1), in binary, of the Fault Domain Error Weight Pair subfield
1	Key: X'09'
2-3	in binary, severity weight for the downstream MAC element (LAN station) problems
4-5(=q)	in binary, severity weight for the upstream MAC element (LAN station) problems

Local Individual MAC Name (X'23') LAN Link Connection Subsystem Data Subfield

This subfield transports the name of the MAC element within the sending node.

0	Length (q+1), in binary, of the Local Individual MAC Name subfield
1	Key: X'23'
2-q	string of no more than 16 uppercase alphanumeric EBCDIC characters, local individual MAC name

Remote Individual MAC Name (X'24') LAN Link Connection Subsystem Data Subfield

This subfield transports the name of the MAC element, part of the link connection, within the adjacent node.

0	Length (q+1), in binary, of the remote individual MAC Name subfield
1	Key: X'24'
2-q	string of no more than 16 uppercase alphanumeric EBCDIC characters, remote individual MAC name

Fault Domain Names (X'26') LAN Link Connection Subsystem Data Subfield

This subfield transports the names of the upstream and the downstream LAN ring stations belonging to a fault domain.

0	Length (q+1), in binary, of the Ring Fault Domain Names subfield
1	Key: X'26'

2-q Pair of Entries

Note: Each entry contains a Length field and a Name field, first entry is for the downstream MAC element, second entry is for the upstream MAC element, and has the following form (shown zero-origin)

0 Length (r+1), in binary, of the following name plus this length field
1-r string of no more than 16 uppercase alphanumeric EBCDIC characters, individual MAC Name

Single MAC Name (X'28') LAN Link Connection Subsystem Data Subfield

> This subfield transports the name of the MAC related to the failure.

0 Length (q+1), in binary, of the Single MAC Name subfield
1 Key: X'28'
2-q string of no more than 16 uppercase alphanumeric EBCDIC characters, Single MAC name

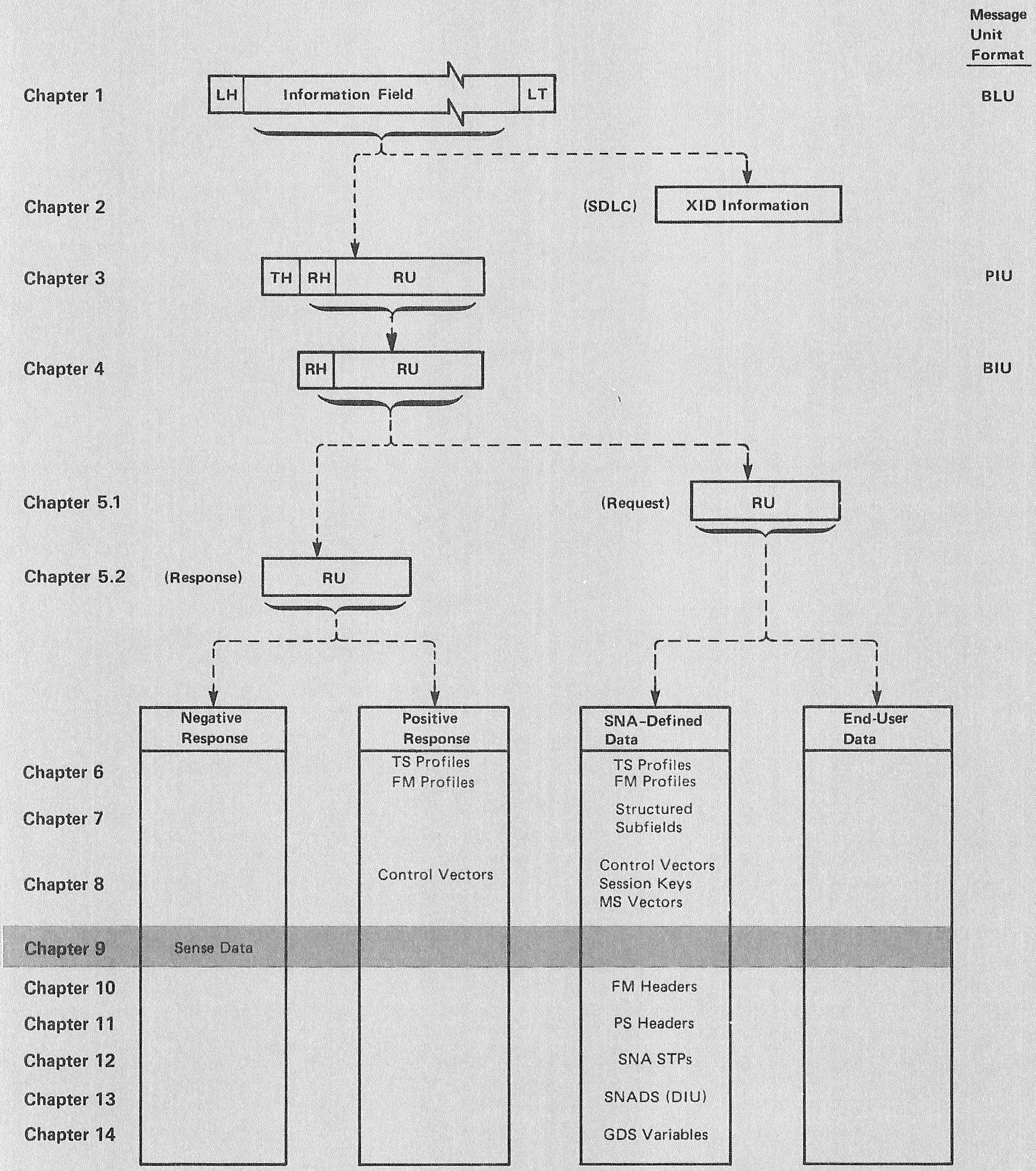
Sense Data (X'7D') MS Common Subvector

> This MS common subvector transports error information back to the control point that initiated an MS request. The subvector contains a 4-byte field for sense data.

0 Length (p+1), in binary, of the Sense Data subvector
1 Key: X'7D'
2-5(=p) Sense data, as defined in "Chapter 9. Sense Data"

Chapter 9

Sense Data



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

The sense data included with an EXCEPTION REQUEST (EXR), a negative response, an UNBIND request, a Sense Data (X'7D') MS common subvector, a function management header type 7 (FMH-7), or an extended sense data control vector (X'35') is a four-byte field (see Figure 9-1) that includes a one-byte category value, a one-byte modifier value, and two bytes of sense code specific information, whose format is defined along with the sense code definition, below.

Byte 0	Byte 1	Byte 2	Byte 3
Category	Modifier	Sense-code specific information	

|<——Sense Code——>|

|<——————Sense Data——————>|

Figure 9-1. Sense Data Format

Together, the category byte 0, the modifier byte 1, and the sense code specific bytes 2 and 3 hold the sense data defined for the exception condition that has occurred.

The following categories are defined; all others are reserved:

VALUE	CATEGORY
X'00'	User Sense Data Only
X'08'	Request Reject
X'10'	Request Error
X'20'	State Error
X'40'	Request Header (RH) Usage Error
X'80'	Path Error

The category User Sense Data Only (X'00') allows the end users to exchange sense data in bytes 2-3 for conditions not defined by SNA within the other categories (and perhaps unique to the end users involved). The modifier value is also X'00'.

In earlier versions of SNA, user data (as well as implementation-specific data) generally could be carried in bytes 2-3 for all categories. This is no longer the case. Bytes 2-3 are used generally only for SNA-defined conditions for nonzero categories; exceptions for implementation-specific use are documented in the appropriate product publications.

The sense codes for the other categories are discussed below.

REQUEST REJECT (CATEGORY CODE = X'08')

This category indicates that the request was delivered to the intended component and was understood and supported, but not executed.

Category and modifier (in hexadecimal):

0801	Resource Not Available: The LU, PU, link station, or link specified in an RU is not available.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000	No specific code applies.
0003	Name aliasing cannot be performed because the name alias function is not available.
0009	The LU is not available because it is not able to comply with the PLU-SLU role specification.
000A	The LU is not available because it is being taken down, and is therefore not accepting new sessions. The initiation request should not be retried.
4001-4002	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0802	Intervention Required: Forms or cards are required at an output device, or a device is temporarily in local mode, or other conditions require intervention.
0803	Missing Password: The required password was not supplied.
0804	Invalid Password: Password was not valid.

0805 Session Limit Exceeded: The requested session cannot be activated, as one of the NAUs is at its session limit, for example, the LU-LU session limit or the (LU, mode) session limit. This sense code applies to ACTCDRM, INIT, BIND, and CINIT requests.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 If accepted, the BIND request would prevent either the receiving LU or the sending LU from activating the number of contention winner sessions to the partner LU that were agreed upon during a change-number-of-sessions procedure.

0002 If accepted, the BIND request would cause the XRF-backup session limit to be exceeded.

0003 If accepted, the BIND request would cause the XRF-active session limit to be exceeded.

Note: The session limit for XRF-active sessions is one. An XRF-active BIND is valid only if there are no XRF-active or XRF-backup sessions with the receiving SLU.

0009 If accepted, the request would cause the PLU session limit to be exceeded.

000A If accepted, the request would cause the SLU session limit to be exceeded.

000B The request was rejected because a session already exists between the same LU pair, and at least one of the LUs does not support parallel sessions.

0806 Resource Unknown: For example, the request contained a name or address not identifying a PU, LU, SSCP, link, or link station known to the receiver or the sender.

Note: In an interconnected network environment, this sense code may be set by an SSCP in whose subnetwork and domain the LU was expected to reside; it is not set by an SSCP that is only an intermediary on the session-setup path. A gateway SSCP examines the Resource Identifier control vector in a session setup request (for example, CDINIT), to determine whether the LU is in the SSCP's subnetwork and domain.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The resources identified in an SNA Address List (X'04') MS common subvector are unknown to the PU receiving the request.

	Note: When this sense data flows in a -RSP to an NMVT, the referenced X'04' subvector is the one that was present in the request NMVT to which the -RSP corresponds. When this sense data flows in a Sense Data (X'7D') MS common subvector, the referenced X'04' subvector is present with the X'7D' subvector in the same major vector.
0002	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.
000A	The configuration identifier specified in a management services command is not recognized by the DLC manager at the receiving node.
0011	An unknown OLU name was specified in the request.
0012	An unknown DLU name was specified in the request.
0013	An unknown SLU name was specified in the request.
0014	An unknown PLU name was specified in the request.
0015	An unknown OLU address was specified in the request.
0016	An unknown DLU address was specified in the request.
0017	An unknown SLU address was specified in the request.
0018	An unknown PLU address was specified in the request.
0021	The session-initiation request specified that the receiving SSCP is the SSCP having the DLU in its domain, but the DLU is unknown to the receiving SSCP.
0022	The originator of the request is unknown to the receiver.
0023	The destination of the request or response is unknown to the sender.
0024	An unknown LU1 name was specified in the request.
0025	An unknown LU2 name was specified in the request.
0026	The SSCP does not have a session with the boundary function PU of an independent LU.
0027	The PU associated with a switched SLU is unknown. Session setup processing for the switched SLU cannot proceed.
0028	NAU1 network address is unknown.
0029	NAU2 network address is unknown.

0807 Resource Not Available—LUSTAT Forthcoming: A subsidiary device will be unavailable for an indeterminate period of time. LUSTAT will be sent when the device becomes available.

0808 Invalid Contents ID: The contents ID contained on the ACTCDRM request was found to be invalid.

0809 Mode Inconsistency: The requested function cannot be performed in the present state of the receiver.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000	No specific code applies.
0001-001B	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.
001C	The RNAA request contains a network ID that is not known to the gateway PU.
001D	An address pair session key in a Network-Qualified Address Pair control vector (X'15') is not known to the gateway PU.
001E	A gateway PU received an RNAA request for a cross-network session and all possible address transforms for the named resource are allocated.
001F	(Retired) An SSCP has detected a specification of gateway responsibility in the CDINIT request that is not consistent with its own definition. For example, an SSCP that has predesignated responsibility to control a gateway node specified in the CDINIT request sends this sense data when it receives the CDINIT from a session partner and the CDINIT indicates that the session partner also has predesignated responsibility for the gateway node; in this situation, a mismatch exists in the responsibilities of the SSCPs, because both cannot simultaneously have predesignated responsibility for the gateway node.
0020	The gateway node receiving an RNAA request cannot support another session between the named resource pair.
0021-0023	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.
0024	A PU received an ACTPU request with the SSCP-PU Session Capabilities control vector (X'0B') indicating that the sending SSCP does not support ENA, but the PU does not know the SSCP's maximum subarea address value.

0025-0026 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0027 A request for a function was received by a component but the function was not enabled or activated.

0028 Cleanup termination of an LU-LU session has been converted to a forced termination by the LU. The SSCP must wait for session ended signals before deleting its session awareness records of the session.

0031 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0034 A termination request has been received for a resource that has been taken over by an SSCP. The termination type is not strong enough to apply to the resources. The termination type needs to be Forced or Cleanup.

0035 A cross-domain resource, which was expected to be active, is inactive.

080A Permission Rejected: The receiver has denied an implicit or explicit request of the sender; when sent in response to BIND, it implies either that the secondary LU will not notify the SSCP when a BIND can be accepted, or that the SSCP does not recognize the NOTIFY vector key X'0C'. (See the X'0845' sense code for a contrasting response.)

080B Bracket Race Error: Loss of contention within the bracket protocol. This error can arise when bracket initiation/termination by both NAUs is allowed

080C Procedure Not Supported: A procedure (Test, Trace, IPL, REQMS type, MS major vector key) specified in an RU is not supported by the receiver.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001-0003 Set aside for implementation specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0005 The MS major vector key is not supported by the receiver.

0006 The MS major vector is identified as one that contains a command, but the receiver does not recognize or support the command subvector. (See the X'086C' sense code for the case in which the command subvector is identified, but an additional required subvector is missing.)

0007 A request for a function is supported by the receiver, but the resource identified in the request does not support that function (no function is specifically indicated).

0009 A request for session information retrieval for an independent LU was received in an REQMS; such requests are permitted only in an NMVT.

000A A request was received containing an address list MS subvector with multiple entries, but the receiver supports only a single entry in such a subvector.

4001, 4003 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

080D NAU Contention: A request to activate a session was received while the receiving half-session was awaiting a response to a previously sent activation request for the same session; for example, the SSCP receives an ACTCDRM from the other SSCP before it receives the response for an ACTCDRM that it sent to the other SSCP and the SSCP ID in the received ACTCDRM was less than or equal to the SSCP ID in the ACTCDRM previously sent.

080E NAU Not Authorized: The requesting NAU does not have access to the requested resource.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The PU, according to its system definition, does not accept an ACTPU from any SSCP having the network ID of the sending SSCP.

080F End User Not Authorized: The requesting end user does not have access to the requested resource.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

6051 Access Security Information Invalid: The request specifies an Access Security Information field that is unacceptable to the receiver; for security reasons, no further detail on the error is provided. This sense data is sent in FMH-7 or UNBIND.

0810 Missing Requester ID: The required requester ID was missing.

0811 Break: Asks the receiver of this sense code to terminate the present chain with CANCEL or with an FMD request carrying EC. The half-session sending the Break sense code enters chain-purge state when Break is sent; the half-session receiving the Break sense code discards the terminated chain without ever retransmitting it.

0812 Insufficient Resource: Receiver cannot act on the request because of a temporary lack of resources.

Bytes 2 and 3 may contain the following sense code specific information:

0000 No specific code applies.

0001 More PUs or LUs requested by RNAA than are present in the pool.

0003 Resources are not currently available to support an XRF session.

0004 The RNAA request indicates that the requested address must be pre-ENA compatible, but no pre-ENA compatible address is available.

0007 Insufficient resources are available for LU address allocation.

000D Insufficient buffers exist to activate a session.

0011 There is insufficient storage available to the SNA component to satisfy the request at this time.

0012 No network address is available to assign to a parallel session.

0813 Bracket Bid Reject—No RTR Forthcoming: BID (or BB) was received while the first speaker was in the in-bracket state, or while the first speaker was in the between-brackets state and the first speaker denied permission. RTR will not be sent.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Bracket Bid Reject: The component was in the in-bracket state when a bracket request was received.

0002 Bracket Bid Reject: The component was in the between-bracket state when a bracket request was received.

0814 Bracket Bid Reject—RTR Forthcoming: BID (or BB) was received while the first speaker was in the in-bracket state, or while the first speaker was in the between-brackets state and the first speaker denied permission. RTR will be sent.

0815 Function Active: A request to activate a network element or procedure was received, but the element or procedure was already active.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 A session activation request was received by a boundary function to activate a session that was already active.

0002 A session activation request was received by a gateway function to activate a cross-network session that was already active.

0003 Processing for another management services request in progress. Sender should retry the request.

Note: This sense data is sent only by a type 2 node, which may lack sufficient queuing space.

0004 A BIND was received from an T2.1 node when the session is already active; i.e., the LFSID is in use. The receiver rejects the BIND.

0816 Function Inactive: A request to deactivate a network element or procedure was received, but the element or procedure was not active.

0817 Link or Link Resource Inactive: A request requires the use of a link or link resource that is not active.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Link inactive.

0002 Link station inactive.

0003 Switched link connection inactive.

4001 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0818 Link Procedure in Process: CONTACT, DISCONTACT, IPL, or other link procedure in progress when a conflicting request was received.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001,0002 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0004 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0005 Link problem determination test for a modem in progress.

0006 Online terminal test in progress.

0007 SDLC link test, level 2, in progress.

0009 Test initiated from the modem panel is in progress.

0819 RTR Not Required: Receiver of READY TO RECEIVE has nothing to send.

081A Request Sequence Error: Invalid sequence of requests.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0002 An IPL or DUMP RU sequence error has occurred.

0004 An NC_ER_TEST was to be sent as a result of receiving a ROUTE_TEST request. The ROUTE_TEST was sent in one subnetwork, the NC_ER_TEST was to be sent in another. The SSCP sending the ROUTE_TEST did not have a required alias address within the subnetwork where the NC_ER_TEST was to be sent. (Before sending ROUTE_TEST, the SSCP sends RNAA, or the installation predefines the alias address, so that an origin SSCP address is available within the subnetwork of the route being tested. This address is then specified in the NC_ER_TEST RU.)

081B Receiver in Transmit Mode: A race condition: normal-flow request received while the half-duplex contention state was not-receive, (*S,¬R), or while resources (such as buffers) necessary for handling normal-flow data were unavailable. (Contrast this sense code with X'2004', which signals a protocol violation.)

081C Request Not Executable: The requested function cannot be executed, because of a permanent error condition in the receiver.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0002 The receiver has an error resulting from a software problem that prevents execution of the request.

00B1 An SDLC error was detected during link problem determination for a modem.

00B2 A modem error (for example, modem check) was detected during link problem determination.

00B3 A timeout threshold was exceeded for a link problem determination aid modem response.

00B4 An overrun or underrun occurred in the node using the link connection during link problem determination for a modem.

00B5 Data Check was signaled during LPDA-2 test.

00B6 Format exception was signaled during LPDA-2 test.

00B7 LPDA-2 modem test was attempted and failed because of a communication controller equipment (for example, scanner) error.

0n0m An error was detected by the DLC manager of the receiving node during the execution of a management services request. If n=X'A', the link connection status has not changed from the state previous to the execution; if n=X'B', the link connection status was modified from the state existing previous to the execution. The error is specified as follows: m=X'1' for volatile storage error, m=X'2' for non-volatile storage (e.g., file access error), m=X'3' for link connection component (e.g., modem) interface error, and m=X'4' for unspecified software error conditions.

Sense code specific information settings 0004, 0008, 000C, 0010, 0014, 0018, 0020, 0028, 0030, 0034, 0038, 003C, 0040, 0072, 0098, 00AB, 0100-0109, 0120-0125, 0149, 0189-0191, 0200-0209, 0220-0225, 0290, 0291, 07**, and 08** are all set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

081D Invalid Station/SSCP ID: The station ID or SSCP ID in the request was found to be invalid.

081E Session Reference Error: The request contained reference to a half-session that either could not be found or was not in the expected state (generally applies to network services requests).

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0003 No session was found during the processing of a session services request.

0004 The appropriate session was found during processing of a session services request, but the session is not in the expected state.

081F Reserved

0820 Control Vector Error: Invalid data for the control vector specified by the target network address and key.

0821 Invalid Session Parameters: Session parameters were not valid or not supported by the half-session whose activation was requested.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0003 The primary half-session requires cryptography, but the secondary half-session does not support cryptography.

0004 The secondary half-session requires cryptography, but the primary half-session does not support cryptography.

0005 Selective or required cryptography is specified, but no SLU cryptographic data key is provided.

0822 Link Procedure Failure: A link-level procedure has failed due to link equipment failure, loss of contact with a link station, or an invalid response to a link command. (This is not a path error, since the request being rejected was delivered to its destination.)

0823 Unknown Control Vector: The control vector specified by a network address and key is not known to the receiver.

0824 Logical Unit of Work Aborted: The current unit of work has been aborted; when sync point protocols are in use, both sync point managers are to revert to the previously committed sync point.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 For LU 6.2, Backout Initiated: A transaction program or its LU has initiated backout. The protected resources for the distributed logical unit of work are to be restored to the previously committed sync point. This sense data is sent only in FMH-7.

For non-LU 6.2, no specific code applies.

0825 Component Not Available: The LU component (a device indicated by an FM header) is not available.

0826 FM function not supported: A function requested in an FMD RU is not supported by the receiver.

0827 Intermittent Error—Retry Requested: An error at the receiver caused an RU to be lost. The error is not permanent, and retry of the RU (or chain) is requested.

0828 Reply Not Allowed: A request requires a normal-flow reply, but the outbound data flow for this half-session is quiesced or shut down, and there is no delayed reply capability.

0829 Change Direction Required: A request requires a normal-flow reply, but the half-duplex flip-flop state (of the receiver of the request) is not-send, and CD was not set on the request. Therefore, there is no delayed reply capability.

082A Presentation Space Alteration: Presentation space altered by the end user while the half-duplex state was not-send, (¬S,*R); request executed.

082B Presentation Space Integrity Lost: Presentation space integrity lost (for example, cleared or changed) because of a transient condition—for example, because of a transient hardware error or an end user action such as allowing presentation services to be used by the SSCP. (Note: The end-user action described under X'082A' and X'084A' is excluded here.)

082C Resource-Sharing Limit Reached: The request received from an SSCP was to activate a half-session, a link, or a procedure, when that resource was at its share limit.

082D LU Busy: The LU resources needed to process the request are being used; for example, the LU resources needed to process the request received from the SSCP are being used for the LU-LU session.

082E Intervention Required at LU Subsidiary Device: A condition requiring intervention, such as out of paper, or power-off, or cover interlock open, exists at a subsidiary device.

082F Request Not Executable because of LU Subsidiary Device: The requested function cannot be executed, due to a permanent error condition in one or more of the receiver's subsidiary devices.

0830 Session-Related Identifier Not Found: The receiver could not find a session-related identifier for a specified session.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 PCID not found for the specified resources.

0002 LSID not found for the specified session.

0831 LU Component Disconnected: An LU component is not available because of power off or some other disconnecting condition.

0832 Invalid Count Field: A count field contained in the request indicates a value too long or too short to be interpreted by the receiver, or the count field is inconsistent with the length of the remaining fields.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnnn Bytes 2 and 3 contain a binary count that indexes (zero-origin) the first byte of the invalid count field.

Note: This sense code is not used for a BIND error because the displacement of fields within the BIND may not be the same at both ends of a session when the BIND was affected by name transformations—for example, after the BIND has passed through a gateway. Sense code X'0835' is used to specify a displacement for a BIND error.

0833 Invalid Parameter (with Pointer and Complemented Byte): One or more parameters contained in fixed- or variable-length fields of the request are invalid or not supported by the NAU that received the request.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnmm Byte 2 contains a binary value that indexes (zero-origin) the first byte that contained an invalid parameter.

Byte 3 contains a transform of the first byte that contained an invalid parameter: the bits that constitute the one or more invalid parameters are complemented, and all other bits are copied.

Note: This sense code is not used for a BIND error because the displacement of fields within the BIND may not be the same at both ends of a session when the BIND was affected by name transformations—for example, after the BIND has passed through a gateway. Sense code X'0835' is used to specify a displacement for a BIND error.

0834 RPO Not Initiated: A power-off procedure for the specified node was not initiated because one or more other SSCPs have contacted the node, or because a CONTACT, DUMP, IPL, or DISCONTACT procedure is in progress for that node.

0835 Invalid Parameter (with Pointer Only): The request contained a fixed- or variable-length field whose contents are invalid or not supported by the NAU that received the request.

nnnn Bytes 2 and 3 contain a two-byte binary count that indexes (zero-origin) the first byte of the fixed- or variable-length field having invalid contents.

Note: This sense code is not used to report an invalid value in an MS major vector. If the invalid value occurs in a formatted MS subvector, sense code X'086B' is used. If it occurs in an unformatted subvector, sense code X'0870' is used.

0836 PLU/SLU Specification Mismatch: For a specified LU-LU session, both the origin LU (OLU) and the destination LU (DLU) have only the primary capability or have only the secondary capability.

0837 Queuing Limit Exceeded: For an LU-LU session initiation request (INIT, CDINIT, or INIT-OTHER-CD) specifying (1) Initiate or Queue (if Initiate not possible) or (2) Queue Only, the queuing limit of either the OLU or the DLU, or both, was exceeded.

0838 Reserved

0839 LU-LU Session Being Taken Down or LU being Deactivated.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 During session-initiation processing, a session-termination request has caused the LU-LU session to be taken down.

083A LU Not Enabled: At the time an LU-LU session initiation request is received at the SSCP, at least one of the two LUs, although having an active session with its SSCP, is not ready to accept CINIT or BIND requests.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The PLU is not enabled.

0002 The SLU is not enabled.

083B Invalid PCID: the received PCID for a new session duplicated the PCID assigned to another session, or the received PCID intended as an identifier for an existing session could not be associated with such an existing session.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The PCID contained in CDINIT(Initiate or Queue), INIT-OTHER-CD, or CDTAKED duplicates a PCID received previously in one of these requests.

083C Domain Takedown Contention: While waiting for a response to a CDTAKED, a CDTAKED request is received by the SSCP containing the SSCP-SSCP primary half-session. Contention is resolved by giving preference to the CDTAKED sent by the primary half-session.

083D Dequeue Retry Unsuccessful—Removed from Queue: The SSCP cannot successfully honor a CDINIT(Dequeue) request (which specifies "leave on queue if dequeue-retry is unsuccessful") to dequeue and process a previously queued CDINIT request (for example, because the LU in its domain is still not available for the specified session), and removes the queued CDINIT request from its queue.

083E Reserved

083F Terminate Contention: While waiting for a response to a CDTERM, a CDTERM is received by the SSCP of the SLU. Contention is resolved by giving preference to the CDTERM sent by the SSCP of the SLU.

0840 Procedure Invalid for Resource: The named RU is not supported in the receiver for this type of resource (for example, (1) SETCV specifies boundary function support for a type 1 node but the capability is not supported by the receiving node, or (2) the NCP PU receiving an EXECTEST or TESTMODE is not the primary NCP PU for the target link.)

0841 Duplicate Network Address: In an LU-LU session initiation request, one of the specified LUs has a duplicate network address already in use.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 The SSCP of the DLU determines that the OLU network address specified in the CDINIT request is a duplicate of an LU network address assigned to a different LU name.

0001 The SSCP of the SLU determines that the switched SLU-network address is a duplicate.

0002 A duplicate PLU network address is found during initiation of a parallel session.

0003 An SSCP finds a duplicate network address for the DLU on the OLU side of the gateway.

0004 An SSCP finds a duplicate network address for the DLU on the DLU side of the gateway.

0005 An SSCP finds a duplicate network address for the OLU on the OLU side of the gateway.

0006 An SSCP finds a duplicate network address for the OLU on the DLU side of the gateway.

0008 A duplicate SLU network address is found during initiation of a parallel session.

0842 Session Not Active.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 (Retired) SSCP-SSCP Session Not Active: The SSCP-SSCP session, which is required for the processing of a network services request, is not active; for example, at the time an LU-LU session initiation or termination request is received, at least one of the following conditions exists:

- The SSCP of the ILU and the SSCP of the OLU do not have an active session with each other, and therefore INIT-OTHER-CD cannot flow.

- The SSCP of the OLU and the SSCP of the DLU do not have an active session with each other, and therefore CDINIT or CDTERM cannot flow.

0002 For a session-initiation request, an SSCP does not have an SSCP-SSCP session with an SSCP in the direction of the DLU.

0003 For a session-initiation request, an SSCP does not have an SSCP-SSCP session with an SSCP in the direction of the OLU.

0843 Required Synchronization Not Supplied: For example, a secondary LU (LU type 2 or 3) received a request with Write Control Code = Start Print, along with RQE and ¬CD.

0844 Initiation Dequeue Contention: While waiting for a response to a CDINIT(Dequeue), a CDINIT(Dequeue) is received by the SSCP of the SLU. Contention is resolved by giving preference to the CDINIT(Dequeue) sent by the SSCP of the SLU.

0845 Permission Rejected—SSCP Will Be Notified: The receiver has denied an implicit or explicit request of the sender; when sent in response to BIND, it implies that the secondary LU will notify the SSCP (via NOTIFY vector key X'0C') when a BIND can be accepted, and the SSCP of the SLU supports the notification. (See the X'080A' sense code for a contrasting response.)

0846 ERP Message Forthcoming: The received request was rejected for a reason to be specified in a forthcoming request.

0847 Restart Mismatch: Sent in response to STSN, SDT, or BIND to indicate that the secondary half-session is trying to execute a resynchronizing restart but has received insufficient or incorrect information.

0848 Cryptography Function Inoperative: The receiver of a request was not able to decipher the request because of a malfunction in its cryptography facility.

0849 Reserved

084A Presentation Space Alteration: The presentation space was altered by the end user while the half-duplex state was not-send, (¬S,*R); request not executed.

084B Requested Resources Not Available: Resources named in the request, and required to honor it, are not currently available. It is not known when the resources will be made available.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0003 The application transaction program specified in the request is not available.

6002 The resource identified by the destination program name (DPN) is not supported.

6003 The resource identified by the primary resource name (PRN) is not supported.

6031 Transaction Program Not Available—Retry Allowed: The FMH-5 Attach command specifies a transaction program that the receiver is unable to start. Either the program is not authorized to run or the resources to run it are not available at this time. The condition is temporary. The sender is responsible for subsequent retry. This sense data is sent only in FMH-7.

084C Permanent Insufficient Resource: Receiver cannot act on the request because resources required to honor the request are permanently unavailable. The sender should not retry immediately because the situation is not transient.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 For LU 6.2, Transaction Program Not Available—No Retry: The FMH-5 Attach command specifies a transaction program that the receiver is unable to start. The condition is not temporary. The sender should not retry immediately. This sense data is sent only in FMH-7.

For non-LU 6.2, no additional information is specified.

0001 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

hnnn where h≥8, i.e., the high-order bit in byte 2 is set to 1. The 15 low-order bits of bytes 2 and 3 contain a binary count that indexes (zero-origin) the first byte of the field found to be in error.

084D Retired

084E Invalid Session Parameters—PRI: A positive response to an activation request (for example, BIND) was received and was changed to a negative response because of invalid session parameters carried in the response. The services manager receiving the response will send a deactivation request for the corresponding session.

084F Reserved

0850 Link-Level Operation Cannot Be Performed: An IPL, dump, or RPO cannot be performed through the addressed link station because the system definition or current state of the hardware configuration does not allow it.

0851 Session Busy: Another session that is needed to complete the function being requested on this session is temporarily unavailable.

0852 Duplicative Session Activation Request: Two session activation requests have been received with related identifiers. The relationship of the identifiers and the resultant action varies by request.

If the RU is an ACTPU or ACTCDRM, it means that a session has already been activated for the subject destination-origin pair by a session activation request that carried a larger activation request identifier than the current request; the current request is refused.

If the RU is a BIND, it means that the BIND request was received with the same session instance identifier (in the structured subfield X'03' of the User Data field) as an active session's; the current request is refused.

0853 TERMINATE(Cleanup) Required: The SSCP cannot process the termination request, as it requires cross-domain SSCP-SSCP services that are not available. (The corresponding SSCP-SSCP session is not active.) TERMINATE(Cleanup) is required.

0854 Retired, formerly used for product-specific information.

0855 Reserved

0856 SSCP-SSCP Session Lost: Carried in the Sense Data field in a NOTIFY (Third-Party Notification vector, X'03') or -RSP(INIT_OTHER) sent to an ILU to indicate that the activation of the LU-LU session is uncertain because the SSCP(ILU)-SSCP(OLU) session has been lost. (Another sense code, X'0842', is used when it is known that the LU-LU session activation cannot be completed.)

0857 SSCP-LU Session Not Active: The SSCP-LU session, required for the processing of a request, is not active; for example, in processing REQECHO, the SSCP did not have an active session with the target LU named in the REQECHO RU.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The SSCP-SLU session is in the process of being reactivated.

0002 The SSCP-PLU session is inactive.

0003 The SSCP-SLU session is inactive.

0004 The SSCP-PLU session is in the process of being reactivated.

0858 Reserved

0859 REQECHO Data Length Error: The specified length of data to be echoed (in REQECHO) violates the maximum RU size limit for the target LU.

085A
through
085F

Reserved

0860

Function Not Supported—Continue Session: The function requested is not supported; the function may have been specified by a request code or some other field, control character, or graphic character in an RU.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnnn Bytes 2 and 3 contain a two-byte binary count that indexes (zero-origin) the first byte in which an error was detected. This sense code is used to request that the session continue, thereby ignoring the error.

0861

Invalid COS Name: The class of service (COS) name, either specified by the ILU or generated by the SSCP of the SLU from the mode table is not in the "COS name to VR identifier list" table used by the SSCP of the PLU.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 COS name was generated by the SSCP.

0001 COS name was generated by the ILU.

0003 The CDINIT request or response contains a Session Initiation control vector that has Class of Service (COS) Name fields that have not been properly specified.

0862

Medium Presentation Space Recovery: An error has occurred on the current presentation space. Recovery consists of restarting at the top of the current presentation space. The sequence number returned is of the RU in effect at the top of the current presentation space.

nnnn Bytes 2 and 3 following the sense code contain the byte offset from the beginning of the RU to the first byte of the RU that is displayed at the top of the current presentation space.

0863

Referenced Local Character Set Identifier (LCID) Not Found: A referenced character set does not exist.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code appplies.

hnnn where h≥8, i.e., the high-order bit in byte 2 is set to 1. The 15 low-order bits of bytes 2 and 3 contain a binary count that indexes (zero-origin) the first byte of the field found to be in error.

0864 Function Abort: The conversation was terminated abnormally. Other terminations may occur after repeated reexecutions; the request sender is responsible to detect such a loop.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 For LU 6.2, Premature Conversation Termination: The conversation is terminated abnormally; for example, the transaction program may have issued a DEALLOCATE_ABEND verb, or the program may have terminated (normally or abnormally) without explicitly terminating the conversation. This sense data is sent only in FMH-7.

For non-LU 6.2, no additional information is specified.

0001 System Logic Error—No Retry: A system logic error has been detected. No retry of the conversation should be attempted. This sense data is sent only in FMH-7.

0002 Excessive Elapsed Time—No Retry: Excessive time has elapsed while waiting for a required action or event. For example, a transaction program has failed to issue a conversation-related protocol boundary verb. No retry of the conversation should be attempted. This sense data is sent in UNBIND when there is no chain to respond to; otherwise, it is sent in FMH-7.

0865 Retired

0866 Retired

0867 Sync Event Response: Indicates a required negative response to an (RQE,CD) synchronizing request.

0868 No Panels Loaded: Referenced format not found because no panels are loaded for the display.

0869 Panel Not Loaded: The referenced panel is not loaded for the display.

086A Subfield Key Invalid: A subfield key in an MS subvector was not valid in the conditions under which it was processed.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnmm Byte 2 following the sense code contains the subvector key (nn) of the subvector containing the unrecognized subfield, and byte 3 contains the unidentified subfield key (mm).

086B Subfield Value Invalid: A value in a subfield within an MS major vector is invalid for the receiver.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnmm Byte 2 following the sense code contains the subvector key (nn) of the subvector containing the subfield with the invalid value, and byte 3 contains the subfield key (mm) of the subfield with the invalid value.

Note: See sense code X'0870' for the case in which the invalid value occurs in an unformatted subvector, that is, one not containing subfields with keys and lengths, or in the unformatted portion of a partially formatted subvector.

086C Required Subvector Missing: One or more MS subvectors that are required by the receiver to perform some function are missing from the received list of subvectors, or are not present in the required position.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nn00 Byte 2 following the sense code contains the subvector key (nn) of one of the subvectors that is missing, or improperly positioned. Byte 3 is reserved (00).

Note: See the X'080C0006' sense data for the case in which the major vector key is recognized but a subvector representing the function to be performed cannot be identified.

086D Required Subfield Missing: An MS subvector lacks one or more subfield keys that are required by the receiver to perform the function requested.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnmm Byte 2 following the sense code contains the subvector key (nn) of the subvector lacking a required subfield, and byte 3 contains the subfield key (mm) of a missing subfield.

086E Invalid Subvector Combination: Two or more subvectors, each permissible by itself, are present in a combination that is not allowed.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnmm Bytes 2 and 3 following the sense code contain the subvector keys (nn) and (mm) of two of the subvectors that should not be jointly present.

086F Length Error: A length field within an MS major vector is invalid, or two or more length fields are incompatible.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The MS major vector length is incompatible with the RU length.

0002 The sum of the MS subvector lengths is incompatible with the MS major vector length.

nn03 The sum of the subfield lengths in a MS subvector is incompatible with the subvector length. Byte 2 following the sense code contains the subvector key (nn).

nn05 MS subvector length invalid. Byte 2 following the sense code contains the relevant subvector key (nn). (This is specified only if the sum of the subvector lengths is compatible with the major vector length.)

nn06 Subfield length invalid. Byte 2 following the sense code contains the subvector key (nn) of the MS subvector containing the invalid subfield length. (This is specified only if the sum of the subfield lengths is compatible with the subvector length.)

0870 Unformatted Subvector Value Invalid: A value in an unformatted MS subvector, or in an unformatted portion of a partially formatted MS subvector, is invalid.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nnxx Byte 2 following the sense code contains the subvector key (nn) of the MS subvector containing the invalid value. Byte 3 contains a one-byte binary count that indexes the first byte in which the invalid value falls. The indexing is zero-origin, from the beginning of the subvector.

Note: See sense code X'086B' for the case in which the invalid value occurs in a formatted MS subvector, that is, one containing subfields with keys and lengths, or in the formatted portion of a partially formatted subvector.

0871 Read Partition State Error: A Read Partition structured field was received while the display was in the retry state.

0872 Orderly Deactivation Refused: An NC_DACTVR(Orderly) request has been received, but sessions are assigned to the VR and it will not be deactivated.

0873 Virtual Route Not Defined: No ERN is designated to support this VRN.

0874 ER Not in a Valid State: The ER supporting the requested VR is not in a state allowing VR activation.

0875 Incorrect or Undefined Explicit Route Requested: The reverse ERNs specified in the NC_ACTVR do not contain the ERN defined to be used for the VR requested, or the ERN designated to be used for the VR is not defined.

0876 Nonreversible Explicit Route Requested: The ERN used by the NC_ACTVR does not use the same sequence of transmission groups (in reverse order) as the ERN that should be used for the RSP(NC_ACTVR).

0877 Resource Mismatch: The receiver of a request has detected a mismatch between two of the following: (1) its definition of an affected resource, (2) the actual configuration, and (3) the definition of the resource as implied in the request.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Link Defined as Switched Is Nonswitched: A link defined to an ACTLINK receiver as being switched was found to be nonswitched during the activation attempt.

0002 Link Defined as SDLC Is Non-SDLC: A link defined to an ACTLINK receiver as being SDLC was found to be non-SDLC during the activation attempt.

0003 Link Defined as Having Automatic Connect-Out Capability Does Not: A link defined to an ACTLINK receiver as having automatic connect-out capability was found to lack it during the activation attempt.

0004 ACTLINK Received for a Resource Other Than a Link: An ACTLINK was received that resolved to a local device address representing a device other than a link.

0005 Link defined as X.21 is not X.21.

0006 Link defined as LPDA-capable is configured in NRZI mode.

0007 A request that is allowed only for a primary link station was received for a link station that is defined to the receiver as secondary.

0008 A request for link problem determination for modems was received for a link that is defined to the receiver as not supporting link problem determination for modems.

0009 A request for link problem determination for modems was received for a link that is defined to the receiver as supporting link problem determination for modems, but no link station supporting link problem determination for modems was found on the link.

000A A request that is allowed only for a nonswitched link was received for a link that is defined to the receiver as switched.

000B A request that is allowed only for a link with a modem not using the multiplexed links feature was received for a link that is defined to the receiver as having a modem using the multiplexed links feature.

000C Resource Definition Mismatch for Modems: A request that is allowed only for a link with a non-tailed modem was received for a link that is defined to the receiver as having a tailed modem.

0878 Insufficient Storage: The storage resource required for a data format is not available.

0879 Storage Medium Error: A permanent error has occurred involving a storage medium.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Disk I/O error.

087A Format Processing Error: A processing error occurred during data formatting.

087B Resource Unknown: The request contains a session key that does not identify a session known to some gateway node; for example, a session activation request arrives at a gateway node after it has released the address transform for the intended session.

087C SSCP-PU Session Not Active: A gateway SSCP-PU session that is needed to establish an address transform for the intended cross-network LU-LU session was not active.

087D Session Services Path Error: A session services request cannot be rerouted along a path of SSCP-SSCP sessions. This capability is required, for example, to set up a cross-network LU-LU session.

Bytes 2 and 3 contain sense code specific information that indicates the specific reason for not rerouting the request. Settings allowed are:

0000 No specific code applies.

0001 An SSCP has attempted unsuccessfully to reroute a session services request to its destination via one or more adjacent SSCPs; this value is sent by a gateway SSCP when it has exhausted trial-and-error rerouting.

Note: This code is used when SSCP rerouting fails completely. The remaining codes are used for failures to reroute to a particular SSCP. For example, they are associated with specific SSCPs when information about a rerouting failure is displayed in the node that was trying to reroute.

0002 An SSCP is unable to reroute a session services request because a necessary routing table is not available, that is, there is no adjacent SSCP table corresponding to the rerouting key in the Resource Identifier control vector. The receiver of this value will, if possible, try rerouting to another SSCP.

0003 This SSCP has no predefinition for an LU, but an adjacent SSCP does not support dynamic definition in partner SSCPs. As a result, this SSCP cannot both dynamically define the LU and reroute to that adjacent SSCP.

0005 (Retired) An SSCP is unable to use the gateway node specified in CDINIT because that gateway node cannot allocate an address transform for the intended cross-network LU-LU session.

0006 (Retired) An SSCP is able to use only a subset of the alternate gateway nodes available to it. However, for the subset that it can use, none can provide the needed alias address pair.

0008 The adjacent SSCP does not support the requested CDINIT function (for example, notification of resource availability or XRF).

000A An SSCP is unable to reroute a session services request because the request has been routed through the same SSCP twice.

000B The DLU specified in the CDINIT is unknown to the receiving SSCP, and the receiving SSCP cannot reroute the CDINIT.

087E SSCP Visit Count Exceeds Limit: The SSCP visit count specified in the session services request—CDINIT, INIT_OTHER_CD, or DSRLST—has been decremented to 0. The session services request has been routed through an excessive number of SSCPs. (The SSCPs are not necessarily distinct.)

087F Reserved

0880 Reserved

0881 ACTCDRM Failure—REQACTCDRM Sent: An SSCP-SSCP session-activation request, ACTCDRM, cannot be rerouted to a gateway SSCP because, at some gateway PU, the necessary transform is not complete and the gateway PU has sent REQACTCDRM to the gateway SSCP.

0882 Reserved

0883 Reserved

0884 ACTCDRM Failure—No REQACTCDRM Sent: An SSCP-SSCP session activation request, ACTCDRM, cannot be rerouted to the destination SSCP because, at some gateway node PU, the necessary transform is not complete and REQACTCDRM cannot be sent to the destination SSCP because the gateway SSCP-PU session is not active or the intended SSCP session partner does not provide gateway services.

0885 Reserved

0886 Subnetwork Rerouting Not Supported: An SSCP received a session services request—CDINIT, INIT_OTHER_CD, NOTIFY(Vector Key=X'01'), or DSRLST—from an SSCP in its subnetwork that, if rerouted, would not cross a subnetwork boundary. The SSCP does not support rerouting within a subnetwork.

0887 Dequeue Retry Unsuccessful—Session Remains Queued: The SSCP cannot successfully honor a CDINIT(Dequeue) request. The request specifies "leave on queue if dequeue-retry is unsuccessful." The SSCP has left the queued session on its queue.

0888 Name Conflict: A name specified in an RU is unknown, or is known and does not have the required capabilities, or is a duplicate resource for the specified resource type. When a name conflict is detected, further name checking ceases; multiple name conflicts are not reported or detected.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000	No specific code applies.
0001	The specified DLU real network name is known, but identifies a resource that is not LU-LU session capable.
0002	The specified DLU alias network name is known, but identifies a resource that is not LU-LU session capable.
0003	The specified OLU real network name is known, but identifies a resource that is not LU-LU session capable.
0004	The specified OLU alias network name is known, but identifies a resource that is not LU-LU session capable.
0005	Name translation was invalid; that is, a different LU name was returned with the same network ID as the original LU name.
0006	The specified DLU real network name is known, but is a duplicate resource.
0007	The specified DLU alias network name is known, but is a duplicate resource.
0008	The specified OLU real network name is known, but is a duplicate resource.
0009	The specified OLU alias network name is known, but is a duplicate resource.
000A	A predefined real resource name and a predefined alias resource name were found for the same resource.
000B	A cross-network DLU name is defined as a shadow resource, but shadow resources are not supported for cross-network sessions.
000C	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0889 Transaction Program Error: The transaction program has detected an error.

This sense code is sent only in FMH-7.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 Program Error—No Data Truncation: The transaction program *sending* data detected an error but did not truncate a logical record.

Program Error—Purging: The transaction program *receiving* data detected an error. All remaining information, if any, that the receiving program had not yet received, and that the sending program had sent prior to being notified of the error, is discarded.

0001 Program Error—Data Truncation: The transaction program *sending* data detected an error and truncated the logical record it was sending.

0100 Service Transaction Program Error—No Data Truncation: The service transaction program *sending* data detected an error and did not truncate a logical record.

Service Transaction Program Error—Purging: The service transaction program *receiving* data detected an error. All remaining information, if any, that the receiving service transaction program had not yet received, and that the sending service transaction program had sent prior to being notified of the error, is discarded.

0101 Service Transaction Program Error—Data Truncation: The service transaction program *sending* data detected an error and truncated the logical record it was sending.

088A Resource Unavailable—NOTIFY Forthcoming: The SSCP cannot satisfy the request because a required resource is temporarily unavailable. When the required resource becomes available, NOTIFY NS(s) key X'07' or X'08' will be sent.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 SSCP-SSCP Session Not Active: A SSCP-SSCP session required to reroute the cross-network request was not active.

0003 SSCP-LU session not active: The SSCP(DLU) is currently not in session with the DLU.

0004 LU session limit exceeded: The DLU is currently at its session limit and the requested session would cause the limit to be exceeded.

088B BB Not Accepted—BIS Reply Requested: Sent in response to a BB (either an LUSTAT bid or an Attach) to indicate that the receiver has sent a BIS request and wishes to terminate the session without processing any more conversations, but without sending an UNBIND. A BIS reply is requested so that the negative response sender may send a normal UNBIND. This sense code is sent only by LUs not supporting change-number-of-session protocols.

088C Missing Control Vector: The RU did not contain a control vector that was expected to appear.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

nn00 A required control vector is missing. Byte 2 contains the key (nn) of the required control vector that is missing. If more than one control vector is missing, only the first omission is reported. The second byte of the sense code specific field is set to X'00'.

088D Duplicate Network Name: An SSCP has detected a violation of the requirement that network names used across multiple domains be unique within the multiple-domain network. For example, the SSCP(DLU) has detected that the OLU name received in CDINIT is currently also defined in the domain of the SSCP(DLU).

088E ENA Address Mismatch: An SSCP detected that an ENA LU has an address too large for one of the pre-ENA components (LU or SSCP) involved in the session to support.

Bytes 2 and 3 following the sense code contains sense code specific information. Settings allowed are:

0000 A resource encountered during LU-LU session initiation is not ENA-capable; the session initiation request may be rerouted.

0001 A resource encountered during LU-LU session initiation is not ENA-capable; the session initiation request should not be rerouted.

0002 An SSCP has requested a "pre-ENA compatible" SLU address for an SLU that already has an ENA address.

0003 The gateway node selected by the gateway SSCP from the gateway node list is not ENA-capable when an ENA-capable gateway node is required. Another gateway node may be tried.

088F XRF Procedure Error: A request was received for an XRF-active or XRF-backup session and was not acted on.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0004 A SWITCH request was received that was invalid.

0006 An INITIATE request for an XRF-backup session was received that allowed queuing (XRF-backup and session queuing are mutually exclusive functions.)

0007 CDINIT was received requesting an XRF-backup session, and the DLU does not support XRF sessions.

0008 An XRF-active BIND was received with a session correlation identifier that duplicates a session correlation identifier associated with an existing XRF session.

0009 An XRF-backup BIND was received for an LU that currently does not have an XRF session.

0010 An XRF-backup BIND was received with a session correlation identifier that does not match the session correlation identifier associated with the existing XRF session with that LU.

000A Cryptography Not Supported: An XRF BIND was received indicating cryptography.

0890 through 0896 Reserved

0897 System Definition Mismatch: The requested function is not supported by the receiver, or there is a mismatch between the sending and receiving system definitions.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0006 The SSCP has no predefinition for an LU and does not support dynamic resource definition.

0007 The receiving SSCP has a different system-defined name for the SSCP(DLU) than the SSCP(DLU) name in the session initiation request.

0008 In a gateway with three gateway SSCPs, a gateway SSCP on the OLU side of the gateway was specified as having predesignated control in the CDINIT. In this configuration, only the middle gateway SSCP may have predesignated control.

0009 In a gateway with three gateway SSCPs, none of which is predesignated, the gateway node believes that one is predesignated. As a result, the gateway node receives gateway control RUs such as RNAA from a different SSCP than the one it expects.

000B An SSCP has detected a specification of gateway responsibility in the CDINIT request that is not consistent with its own definition. For example, two gateway SSCPs in the same gateway are both predefined to be predesignated.

000C The receiver is unable to interpret the DLU name.

0898 Reserved

0899 Reserved

089A Invalid File or File Not Found: The requested file was not found, or was found to be an invalid file.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Requested file not found.

089B Reserved

089C Reserved

089D Gateway Node Error Detected during Cross-Network Session Initiation.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The gateway node list used to select a gateway node to cross a network boundary is exhausted.

0003 RNAA has failed; another gateway node should be tried.

0004 Address conversion based on the subarea/element address split was unsuccessful.

0005 The gateway node selected by one gateway SSCP is not known to another gateway SSCP in the same gateway. This can be a system definition error in the gateway SSCP that does not recognize the gateway node.

0006 A gateway SSCP has found that a gateway node has assigned duplicate addresses.

REQUEST ERROR (CATEGORY CODE = X'10')

This category indicates that the RU was delivered to the intended NAU component, but could not be interpreted or processed. This condition represents a mismatch of NAU capabilities.

Category and modifier (in hexadecimal):

1001 RU Data Error: Data in the request RU is not acceptable to the receiving component; for example, a character code is not in the set supported, a formatted data field is not acceptable to presentation services, or a value specified in the length field (LL) of a structured field is invalid.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000	No specific code applies.
0001	The request contains a subarea address of 0 or a subarea address greater than the maximum subarea value within the specified or implied network.
0002	The network ID specified in the ACTPU is unknown, or is not valid on the link over which the ACTPU was received.
0008	An invalid character code was found.
0009	The formatted data field is unacceptable to presentation services.
000A	An invalid length field for a structured field was found.
000B	The value in the name length field is too great.
000C	The value in the cryptography key length field is too great.
000D	The URC field length is invalid.
000E	The control vector length field is inconsistent with the control vector data.
000F	A PLU or SLU role specification encoding is invalid.
0020	Too many session keys are present.
0021	A control vector or session key data is invalid.
0022	A BIND image in a session services RU is invalid.
0023	A device characteristics field is invalid.
hnnn	where h≥8, i.e., the high-order bit in byte 2 is set to 1. The 15 low-order bits of bytes 2 and 3 contain a binary count that indexes (zero-origin) the first byte of the field found to be in error.

1002 RU Length Error: The request RU was too long or too short.

1003 Function Not Supported: The function requested is not supported. The function may have been specified by a formatted request code, a field in an RU, or a control character.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The half-session receiving the request did not perform the function because it is not capable of doing so. The requesting half-session requested a function that the receiver does not support and the receiver did not specify that it was capable of supporting the function at session activation; consequently, there is an apparent mismatch of half-session capabilities.

Note: This is to cover a system error. For example, if the PU receiving a SETCV(Vector Key=X'15') is not a gateway PU, that is, the PU did not indicate in the ACTPU response that it is a gateway PU, the PU reports to the SSCP that sent the SETCV that there is an apparent mismatch of half-session capabilities.

0002 The half-session receiving the request did not perform the function, though it is capable of doing so. The requesting half-session did not specify at session activation that it was capable of supporting the function; consequently, there is an apparent mismatch of half-session capabilities.

Note: This is to cover a system error. For example, if the SSCP sending a SETCV(Vector Key=X'15') is not known to the receiving PU as a gateway SSCP, that is, the SSCP did not indicate in ACTPU that it is a gateway SSCP, the PU reports a mismatch of capabilities.

Note: 0001 and 0002 are also assigned for implementation-specific use; see implementation documentation for details of usage.

0003 The component received an unsupported normal-flow DFC command.

0004 The component received an unsupported expedited-flow DFC command.

0005 The component received a network control command during an LU-SSCP session.

0006 The component received an unsupported session control command during an LU-SSCP session.

0007 The component received an unsupported data flow control command with LU-SSCP session specified.

000D The function identified in the request is not supported by the processing application transaction program.

0010 The RU is not known to session services.

0011 A session key is not supported.

0012 A control vector is not supported.

0014 Cryptography is not supported but a nonzero length was specified for the cryptography key.

0015 Queuing not supported for a controller session.

0020 A session initiation request specified an OLU and DLU that are the same LU. An LU cannot establish a session with itself.

6002 The resource identified by the destination program name (DPN) is not supported.

6003 The resource identified by the primary resource name (PRN) is not supported.

Note: This sense code can also be used instead of sense code X'0826'.

1004 Reserved

1005 Parameter Error: A parameter modifying a control function is invalid, or outside the range allowed by the receiver.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 For NMVT, the address type field in an SNA Address List subvector does not match the address type required by the command subvector.

0002 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

0004 Invalid display type was requested.

0005 Invalid storage length for display type requested.

0006 Invalid storage address; out of specified range.

0008 and 0121-0229 Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.

1006 Required field or parameter is missing.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 One or more required COS names were omitted.

0002 A required name was omitted.

0003	A required network identifier was omitted.
0004	A required session key was omitted.
0005	A required control vector was omitted.

1007 Category Not Supported: DFC, SC, NC, or FMD request was received by a half-session not supporting any requests in that category; or an NS request byte 0 was not set to a defined value, or byte 1 was not set to an NS category supported by the receiver.

1008 Invalid FM Header: The FM header was not understood or translatable by the receiver, or an FM header was expected but not present. For LU 6.2, this sense code is sent in FMH-7 or UNBIND.

Bytes 2 and 3 following the sense code contain sense code specific information. Figure 9-2 on page 9-41 shows the usage of the allowed values by LU type. Settings allowed are:

0000	No specific code applies.
0801	The function code parameters are invalid.
0803	The forms functions cannot be performed.
0805	The copy function cannot be performed.
0806	Compaction table outside the supported set: The number of master characters is not within the valid range.
0807	The PDIR (peripheral data information record) identifier is invalid.
0808	The printer train function cannot be performed.
0809	The FCB (forms control block) load function cannot be performed.
080A	The FCB (forms control block) load function is not supported.
080B	The compaction table name is invalid.
080C	The ACCESS is invalid.
080D	The RECLEN is invalid.
080E	The NUMRECS is invalid.
080F	The data set is in use.
0810	The data set cannot be found.
0811	The password is invalid.

0812	The function is not allowed for the destination or for the data set.
0813	The record is too long.
0814	The data set is full.
0815	The RECID is invalid.
0816	Reserved
0817	The VOLID format is invalid.
0818	The maximum number of logical records per chain is exceeded.
0819	The data set exists.
081A	No space is available.
081B	The VOLID is invalid.
081C	The DSACCESS is invalid.
081D	The RECTYPE is invalid or the data set cannot be found.
081E	The resolution space is insufficient.
081F	The key technique is invalid.
0820	The key displacement is invalid.
0821	The key is invalid.
0822	There is an Invalid N (number of records.)
0823	The KEYIND is invalid.
0824	The SERID is invalid.
0825	Disk Error: An error was detected while reading from, or writing on, the disk.
0826	The RECID format is invalid.
0827	The password has not been supplied.
0828	The record ID has not been supplied.
0829	The Volume ID has not been supplied.
082A	The PGMNAME is invalid.

1204	Set aside for implementation-specific use, and will not be otherwise defined in SNA; see implementation documentation for details of usage.
2001	The destination (active) is invalid.
2002	The destination (inactive) is invalid.
2003	The destination (suspended) is invalid.
2004	The suspend-resume sequence is invalid.
2005	There has been an interruption level violation.
2006	The resume properties are invalid.
2007	The destination is not available.
2008	The end sequence is invalid.
2009	The FM header length is invalid.
200A	Invalid field setting: The reserved field is set to 1 or the setting is not defined.
200B	Invalid destination: The destination does not exist.
200C	The ERCL is invalid.
200D	The DST is invalid.
200E	Invalid Concatenation Indicator: The concatenation indicator is on, but concatenation is not allowed.
200F	FM data is not allowed for the header.
2010	The FM header set specified in the BIND has been violated.
2011-2013	Reserved
2014	The FM header was not sent concatenated.
2015-2018	Reserved
2019	The stack reference indicator (SRI) is invalid.
201A	The CMI modification could not be accepted.
201B	The CPI modification could not be accepted.
201C	The ECRL modification could not be accepted.
201D	FM Header and Associated Data Mismatch: The FM header indicated associated data would or would not follow (for

example, FM header 7 followed by log data, or FM header 5 followed by program initialization parameters), but this indication was in error; or a previously received RU (for example, -RSP(X'0846')) implied that an FM header would follow, but none was received.

4001	Invalid FM Header Type for this LU: The type of the FM header is other than 5, 7, or 12.
4002	The FMH code is invalid.
4003	Compression is not supported.
4004	Compaction is not supported.
4005	Basic exchange is not supported.
4006	Only basic exchange is supported.
4007	The medium is not supported.
4008	There has been a code selection compression violation.
4009	FMHC is not supported.
400A	Demand select is not supported.
400B	DSNAME is not supported.
400C	The media subaddress field is invalid.
400D	There are insufficient resources to perform the requested function.
400E	DSP select is not supported.
6000	FM Header Length Not Correct: The value in the FM header Length field differs from the sum of the lengths of the subfields of the FM header.
6001	The deblocking algorithm (DBA) is invalid.
6004	The queue name length is invalid.
6005	Access Security Information Length Field Not Correct: The value in the Access Security Information Length field differs from the sum of the lengths of the Access Security Information subfields.
6006	The data stream profile (DSP) is invalid.
6007	The FMH-7 is not preceded by a negative response carrying the X'0846' sense code.

6008	The Attach access code is invalid.
6009	Invalid Parameter Length: The field that specifies the length of fixed-length parameters has an invalid setting.
600A	This is not the first FMH-5, the interchange unit type is not the same as the old, and the interchange unit end indicator is off.
600B	Unrecogized FM Header Command Code: The partner LU received an FM header command code that it does not recognize. For LU 6.2 this sense data is sent only in FMH-7.
600C	A null sequence field is required.
600D	User to user program transition is not allowed.
600E	User to non-SNA defined program transition is not allowed.
600F	The FMH-5 reset attached program (RAP) was not sent properly.
6010	The FMH-5 reset attached program (RAP) was sent with an inactive Attach register.
6011	Invalid Logical Unit of Work (LUW): The LUW Length field (in a Compare States GDS variable or an FMH-5) is incorrect, or the length field is invalid, or a LUW ID is not present but is required by the setting of the synchronization level field.
6021	Transaction Program Name Not Recognized: The FMH-5 Attach command specifies a transaction program name that the receiver does not recognize. This sense data is sent only in FMH-7.
6031	PIP Not Allowed: The FMH-5 Attach command specifies program initialization parameter (PIP) data is present, but the receiver does not support PIP data for the specified transaction program. This sense data is sent only in FMH-7.
6032	PIP Not Specified Correctly: The FMH-5 Attach command specifies a transaction program name that requires program initialization parameter (PIP) data, and either the FMH-5 specifies PIP data is not present or the number of PIP subfields present does not agree with the number required for the program. This sense data is sent only in FMH-7.
6034	Conversation Type Mismatch: The FMH-5 Attach command specifies a conversation type that the receiver does not support for the specified transaction program. This sense data is sent only in FMH-7.
6040	Invalid Attach Parameter: A parameter in the FMH-5 Attach command conflicts with the statement of LU capability previously provided in the BIND negotiation.

6041 Synchronization Level Not Supported: The FMH-5 Attach command specifies a synchronization level that the receiver does not support for the specified transaction program. This sense data is sent only in FMH-7.

C000 The Header is not supported.

C001 The header length is invalid.

C002 There has been a logical message services block-level error.

C003 There is a version ID mismatch.

1009 Format Group Not Selected: No format group was selected before issuing a Present Absolute or Present Relative Format structured field to a display.

STATE ERROR (CATEGORY CODE = X'20')

This category indicates a sequence number error, or an RH or RU that is not allowed for the receiver's current session control or data flow control state. These errors prevent delivery of the request to the intended component.

Category and modifier (in hexadecimal):

2001 Sequence Number: Sequence number received on normal-flow request was not 1 greater than the last.

2002 Chaining: Error in the sequence of the chain indicator settings (BCI, ECI), such as first, middle, first.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The receiver received a middle or end-chain request when in the in-chain state.

0002 The receiver received a begin-chain request when in the in-chain state.

2003 Bracket: Error resulting from failure of sender to enforce bracket rules for session. (This error does not apply to contention or race conditions.)

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

Range	LU 1	LU 4	LU 6.1	LU 6.2
0801–0824	X	X		
0825	X			
0826–082A	X	X		
2001–200D	X	X		
200E	X	X	X	
200F–201C	X	X		
201D				X
4001–400E	X	X		
6000				X
6001,6004			X	
6005			X	X
6006–6008			X	
6009			X	X
600A			X	
600B			X	X
600C–6010			X	
6011–6034				X
6040			X	X
6041				X
C000–C003			X	

Figure 9-2. Usage of X'1008' Sense Code Specific Information by LU Type

0001 The receiver received a begin-bracket request before receiving a response to its own previously sent begin-bracket request.

0002 The receiver received a begin-bracket request not specifying begin-bracket when in the between-bracket state.

	0003 The receiver received an out-of-sequence LUSTAT command.
2004	Direction: Error resulting from a normal-flow request received while the half-duplex flip-flop state was not Receive.
2005	Data Traffic Reset: An FMD or normal-flow DFC request received by a half-session whose session activation state was active, but whose data traffic state was not active.
2006	Data Traffic Quiesced: An FMD or DFC request received from a half-session that has sent QUIESCE COMPLETE or SHUTDOWN COMPLETE and has not responded to RELEASE QUIESCE.
2007	Data Traffic Not Reset: A session control request (for example, STSN), allowed only while the data traffic state is reset, was received while the data traffic state was not reset.
2008	No Begin Bracket: An FMD request specifying BBI=BB was received after the receiver had previously received a BRACKET INITIATION STOPPED request.
2009	Session Control Protocol Violation: An SC protocol has been violated; a request, allowed only after a successful exchange of an SC request and its associated positive response, has been received before such successful exchange has occurred (for example, an FMD request has preceded a required CRYPTOGRAPHY VERIFICATION request). The request code of the particular SC request or response required, or X'00' if undetermined, appears in the fourth byte of the sense data.
200A	Immediate Request Mode Error: The immediate request mode protocol has been violated by the request.
200B	Queued Response Error: The Queued Response protocol has been violated by a request, i.e., QRI=¬QR when an outstanding request had QRI=QR.
200C	ERP Sync Event Error: The ERP sync event protocol in DFC has been violated; for example, after receiving a negative response to a chain, a request other than a request soliciting a synchronization event response was sent to DFC_SEND and rejected.
200D	Response Owed Before Sending Request: An attempt has been made in half-duplex (flip-flop or contention) send/receive mode to send a normal-flow request when a response to a previously received request has not yet been sent.
200E	Response Correlation Error: A response was received that cannot be correlated to a previously sent request.
200F	Response Protocol Error: A violation has occurred in the response protocol; e.g., a +RSP to an RQE chain was generated.
2010	BIS Protocol Error: A BIS protocol error was detected; for example, a BIS request was received after a previous BIS was received and processed.

2011	Pacing Error: A normal-flow request is received by a half-session after the pacing count has been reduced to 0 and before a pacing response has been sent.
2012	Invalid Sense Code Received: A negative response was received that contains an SNA-defined sense code that cannot be used for the sent request.

RH USAGE ERROR (CATEGORY CODE = X'40')

This category indicates that the value of a field or combination of fields in the RH violates architectural rules or previously selected BIND options. These errors prevent delivery of the request to the intended component and are independent of the current states of the session. They may result from the failure of the sender to enforce session rules. Detection by the receiver of each of these errors is optional.

Category and modifier (in hexadecimal):

4001	Invalid SC or NC RH: The RH of a session control (SC) or network control (NC) request was invalid. For example, an SC RH with pacing request indicator set to 1 is invalid.
4002	Reserved
4003	BB Not Allowed: The Begin Bracket indicator (BBI) was specified incorrectly, for example, BBI=BB with BCI=¬BC.
4004	CEB or EB Not Allowed: The Conditional End Bracket indicator (CEBI) or End Bracket indicator (EBI) was specified incorrectly, for example, CEBI=CEB when ECI=¬EC or EBI=EB with BCI=¬BC, or by the primary half-session when only the secondary may send EB, or by the secondary when only the primary may send EB.
4005	Incomplete RH: Transmission shorter than full TH-RH.
4006	Exception Response Not Allowed: Exception response was requested when not permitted.
4007	Definite Response Not Allowed: Definite response was requested when not permitted.
4008	Pacing Not Supported: The Pacing indicator was set on a request, but the receiving half-session or boundary function half-session does not support pacing for this session.
4009	CD Not Allowed: The Change Direction indicator (CDI) was specified incorrectly, for example, CDI=CD with ECI=¬EC, or CDI=CD with EBI=EB.
400A	No-Response Not Allowed: No-response was specified on a request when not permitted. (Used only on EXR.)

400B	Chaining Not Supported: The chaining indicators (BCI and ECI) were specified incorrectly, for example, chaining bits indicated other than (BC,EC), but multiple-request chains are not supported for the session or for the category specified in the request header.
400C	Brackets Not Supported: The bracket indicators (BBI, CEBI, and EBI) were specified incorrectly, e.g., a bracket indicator was set (BBI=BB, CEBI=CEB, or EBI=EB), but brackets are not used for the session.
400D	CD Not Supported: The Change-Direction indicator was set, but is not supported.
400E	Reserved
400F	Incorrect Use of Format Indicator: The Format indicator (FI) was specified incorrectly, for example, FI was set with BCI=¬BC, or FI was not set on a DFC request.
4010	Alternate Code Not Supported: The Code Selection indicator (CSI) was set when not supported for the session.
4011	Incorrect Specification of RU Category: The RU Category indicator was specified incorrectly, for example, an expedited-flow request or response was specified with RU Category indicator = FMD.
4012	Incorrect Specification of Request Code: The request code on a response does not match the request code on its corresponding request.
4013	Incorrect Specification of (SDI, RTI): The Sense Data Included indicator (SDI) and the Response Type indicator (RTI) were not specified properly on a response. The proper value pairs are (SDI=SD, RTI=negative) and (SDI=¬SD, RTI=positive).
4014	Incorrect Use of (DR1I, DR2I, ERI): The Definite Response 1 indicator (DR1I), Definite Response 2 indicator (DR2I), and Exception Response indicator (ERI) were specified incorrectly, for example, a SIGNAL request was not specified with DR1I=DR1, DR2I=¬DR2, and ERI=¬ER.
4015	Incorrect Use of QRI: The Queued Response indicator (QRI) was specified incorrectly, for example, QRI=QR on an expedited-flow request.
4016	Incorrect Use of EDI: The Enciphered Data indicator (EDI) was specified incorrectly, for example, EDI=ED on a DFC request.
4017	Incorrect Use of PDI: The Padded Data indicator (PDI) was specified incorrectly, for example, PDI=PD on a DFC request.
4018	Incorrect Setting of QRI with Bidder's BB: The first speaker half-session received a BB chain requesting use of a session (via LUSTAT(X'0C06')), but the QRI was specified incorrectly, that is, QRI = ¬QR.
4019	Incorrect Indicators with Last-In-Chain Request: A last-in-chain request has specified incompatible RH settings, for example, RQE*, CEBI=¬CEB, and CDI=¬CD.

401A through 4020 Reserved

4021 QRI Setting in Response Different From That in Request: The QRI setting in the response differs from the QRI setting in the corresponding request.

PATH ERROR (CATEGORY CODE = X'80')

This category indicates that the request could not be delivered to the intended receiver, because of a path outage, an invalid sequence of activation requests, or one of the listed path information unit (PIU) errors. Some PIU errors fall into other categories; for example, sequence number errors are sense code category X'20'. A path error received while the session is active generally indicates that the path to the session partner has been lost.

Category and modifier (in hexadecimal):

8001 Intermediate Node Failure: Machine or program check in a node providing intermediate routing function. A response may or may not be possible.

8002 Link Failure: Data link failure.

8003 NAU Inoperative: The NAU is unable to process requests or responses; for example, the NAU has been disrupted by an abnormal termination.

8004 Unrecognized Destination: A node in the path has no routing information for the destination specified either by the SLU name in a BIND request or by the TH.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 A request was received by a gateway function that could not be rerouted because of invalid or incomplete routing information.

8005 No Session: No half-session is active in the receiving end node for the indicated origination-destination pair, or no boundary function session connector is active for the origin-destination pair in a node providing the boundary function. A session activation request is needed.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The receiver received a request other than session control request when no LU-LU session was active.

0002 The receiver received a request other than session control request when no LU-SSCP session was active.

0003 The receiver received a session control request other than BIND/UNBIND when no LU-LU session was active.

0004 The receiver received an UNBIND when no LU-LU session was active.

0005 The receiver received a session control request other than ACTLU/DACTLU for the LU-SSCP session when no LU-SSCP session was active.

0006 The receiver received DACTLU when no LU-SSCP session was active.

8006 Invalid FID: Invalid FID for the receiving node. (Note 1)

8007 Segmenting Error: First BIU segment had less than 10 bytes; or mapping field sequencing error, such as first, last, middle; or segmenting not supported and MPF not set to 11. (See note 2.)

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The node does not support receipt of segments, and a mapping field value other than OIS was received.

8008 PU Not Active: The SSCP-PU secondary half-session in the receiving node has not been activated and the request was not ACTPU for this half-session; for example, the request was ACTLU from an SSCP that does not have an active SSCP-PU session with the PU associated with the addressed LU.

8009 LU Not Active: The destination address specifies an LU for which the SSCP-LU secondary half-session has not been activated and the request was not ACTLU.

800A Too-Long PIU: Transmission was truncated by a receiving node because the PIU exceeded a maximum length or sufficient buffering was not available.

800B Incomplete TH: Transmission received was shorter than a TH. (see Note 1)

800C DCF Error: Data Count field inconsistent with transmission length.

800D Lost Contact: Contact with the link station for which the transmission was intended has been lost, but the link has not failed. If the difference between link failure and loss of contact is not detectable, link failure (X'8002') is sent.

800E Unrecognized Origin: The origin address specified in the TH was not recognized.

800F The address combination is invalid.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 The (DAF',OAF') (FID2) combination or the LSID (FID3) specified an invalid type of session, for example, a PU-LU combination.

0001 The FID2 ODAI setting in a received BIND is incorrect; the BIND is rejected.

8010 Segmented RU Length Error: An RU was found to exceed a maximum length, or required buffer allocation that might cause future buffer depletion.

8011 ER Inoperative or Undefined: A PIU was received from a subarea node that does not support ER and VR protocols, and the explicit route to the destination is inoperative or undefined.

8012 Subarea PU Not Active or Invalid Virtual Route: A session-activation request for a peripheral PU or LU cannot be satisfied because there is no active SSCP-PU session for the subarea node providing boundary function support, or the virtual route for the specified SSCP-PU (type 1 or type 2 nodes) or SSCP-LU session is not the same as that used for the SSCP-PU session of the type 1 or type 2 node's PU or the LU's subarea PU.

8013 COS Not Available: A session activation request cannot be satisfied because none of the virtual routes requested for the session is available.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

Byte 2 indicates the environment in which the failure was detected:

00 Single network

01 Interconnected network: Failure was detected at a node in a subnetwork other than that of the NAU sending the activation request.

Byte 3 indicates the reason for the session-activation failure:

00 No Specific Code Applies: This means an error occurred, but none of the conditions listed below applies.

01 No Mapping Specified: A session-activation request cannot be satisfied because for each VR in the VR identifier list for the session, no VR to ER mapping is specified.

02 No Explicit Routes Defined: A session-activation request cannot be satisfied because each VR in the VR identifier list for the session maps to a corresponding ER that is not defined.

03 No VR Resource Available: A session-activation request cannot be satisfied because each VR specified in the VR identifier list for the session requires a node resource that is not available.

04 No Explicit Routes Operative: A session-activation request cannot be satisfied because no underlying ER is operative for any VR specified in the VR identifier list for the session.

05 No Explicit Route Can Be Activated: A session-activation request cannot be satisfied because no VR specified in the VR identifier list for the session mapped to a defined and operative ER that could be activated.

06 No Virtual Route Can Be Activated: A session-activation request cannot be satisfied because no VR specified in the VR identifier list for the session can be activated by the PU, though for at least one VR an underlying ER is defined, operative, and activated.

07 No Virtual Route Identifier List Available: A session-activation request cannot be satisfied because a VR identifier list is not available.

Note: If none of the virtual routes specified in the VR identifier list for the session is active or can be activated, the reported reason is set based on a hierarchy of failure events. The "highest" of the failures that occurred within the set of virtual routes is returned on the response. For example, if the VR manager receives a negative response to an NC_ACTVR request for a VR specified in the VR identifier list and for all other VRs in the list no VR to ER mapping is specified, then reason X'06' is reported. The hierarchy of the failure reasons is in ascending numeric order, that is, reason X'02' is higher than reason X'01'.

8014 through 8016 Reserved

8017 PIU from Adjacent Pre-ER-VR Subarea Node Rejected: A PIU that requires intermediate path-control routing was received by a subarea node from an adjacent subarea node that does not support ER-VR protocols, but the receiving subarea node does not support intermediate path-control routing for adjacent subarea nodes that do not support ER-VR protocols.

8018 Management Services component is unable to find or recognize the name of the application transaction program specified in the request.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The application transaction program specified in the request is not recognized by PUMS.

Notes:

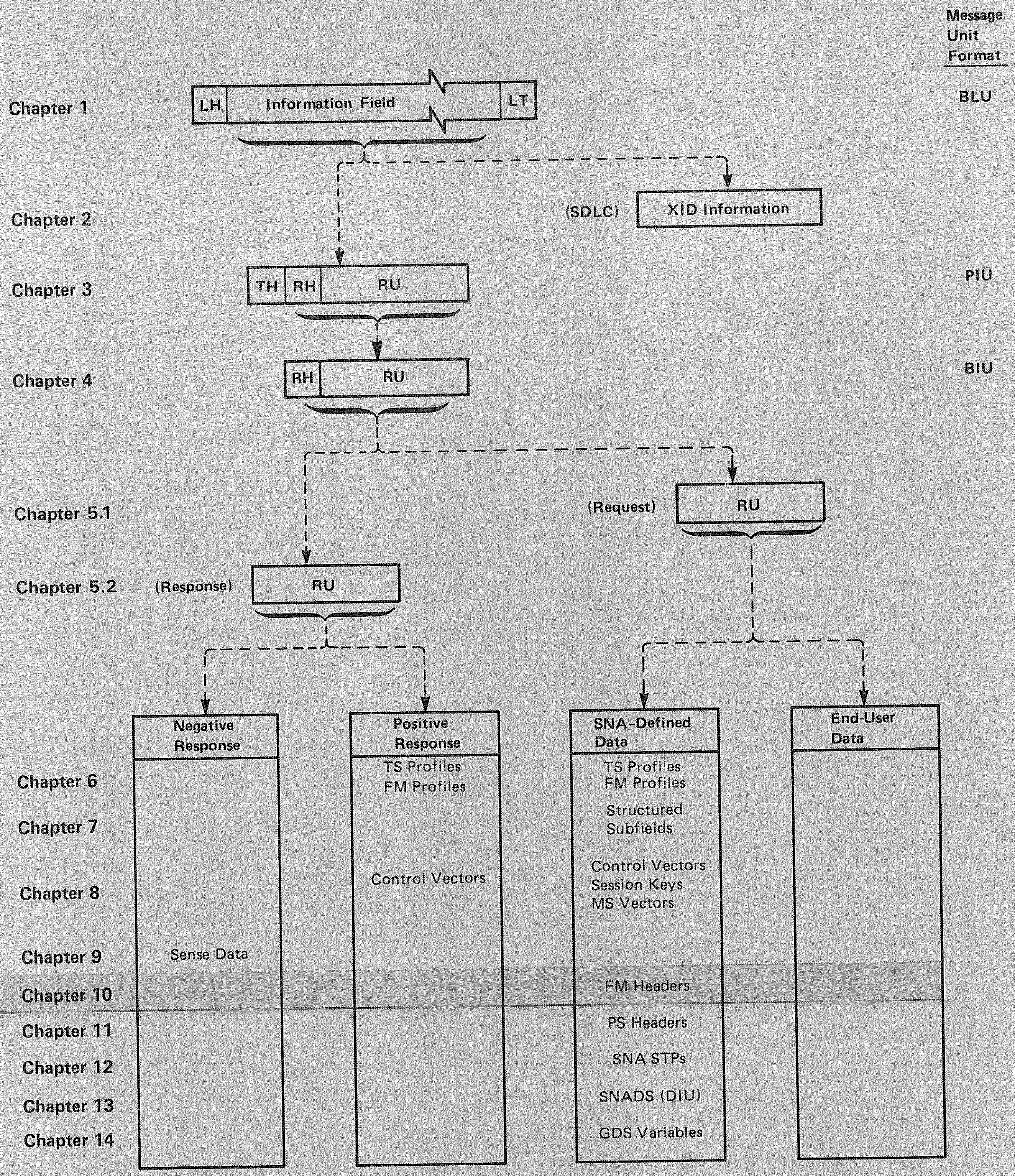
1. It is generally not possible to send a response for this exception condition, since information (FID, addresses) required to generate a response is not available. It is logged as an error if this capability exists in the receiver.

2. If segmenting is not supported, a negative response is returned for the first segment only, since this contains the RH. Subsequent segments are discarded.

This page intentionally left blank

Chapter 10

Function Management Headers



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 10. FUNCTION MANAGEMENT HEADERS

The request header (RH) contains a format indicator (FI) that, when on, indicates that an FM header is at the beginning of the request unit (RU).

For LU 6.2, FM headers may appear only singly at the beginning of an RU. The RU containing the FM header may appear anywhere within a chain. When the FM header is longer than one RU will hold, the header is continued in as many additional RUs as needed to hold it. Figure 10-1 and Figure 10-2 show the placement of FM headers within an RU:

RH: FMH, *BC,*EC	FM header	Data

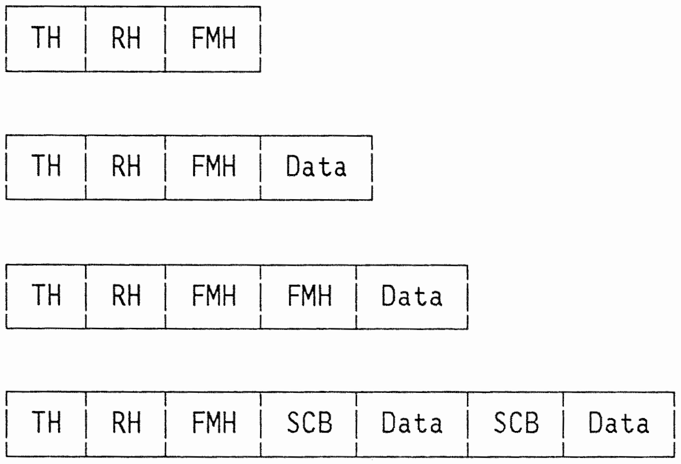
Figure 10-1. FM Header Contained in One RU

RH: FMH, *BC,¬EC	First of FM header

RH: ¬FMH,¬BC,*EC	Rest of FM header	Data

Figure 10-2. FM Header Contained in Two Contiguous RUs of a Chain

Figure 10-3 shows some instances where FM headers are used and Figure 10-4 identifies the logical unit (LU) types that use each FM Header.



FMH -- Function Management (FM) Header
SCB -- String Control Byte
TH -- Transmission Header
RH -- Request/Response Header

Figure 10-3. Usage of FM Headers

LU Type	FM Header Type
0	None required, but may use any header
1	1, 2, and/or 3
2	None
3	None
4	1, 2, and/or 3
6.1	4, 5, 6, 7, 8, and/or 10
6.2	5, 7, 12
7	None

Figure 10-4. LU Types That Support FM Headers

FM header 1

This header is used to select a destination within a logical unit (LU). A destination may be represented by a device, a data set residing on a device, or merely a data stream. The LU initiates, interrupts, resumes, and concludes data traffic for the half-session using the FMH-1.

- 0 Length, in binary, of FMH-1, including this Length byte
- 1
  - bit 0, FMH concatenation:
    - 0 no FMH follows this FMH-1
    - 1 another FMH follows this FMH-1
  - bits 1-7, type: 0000001
- 2
  - bits 0-3, select desired medium for data (see notes 1 and 2):
    - 0000 console
    - 0001 exchange
    - 0010 card
    - 0011 document
    - 0100 nonexchange disk
    - 0101 extended document
    - 0110 extended card
    - 0111 data set name select destination (see note 3)
    - 1000 word processing (WP) media 1
    - 1001 WP media 2
    - 1010 WP media 3
    - 1011 reserved
    - 1100 WP media 4
    - 1101 reserved
    - 1110 reserved
    - 1111 reserved
  - bits 4-7, logical subaddress (see note 2):
    - 0000-1110 specific device in medium class
    - 1111 any device in medium class (see note 3)
- 3
  - bit 0, SRI: stack reference indicator:
    - 0 stack to be used is the sender's send stack
    - 1 stack to be used is the receiver's send stack
  - bit 1, demand select:
    - 0 receiver may direct data to alternate medium/subaddress
    - 1 receiver must direct data to specified medium/subaddress (spooling is prohibited)
  - bits 2-3, reserved
  - bits 4-7, DSPs: data stream profiles:
    - 0000 default (the DSP is implied by the Medium Select field)
    - 0001 base
    - 0010 general
    - 0011 job
    - 0100 WP raw-form text
    - 0101 WP exchange diskette
    - 0110 reserved
    - 0111 Office Information Interchange level 2
    - 1000 reserved
    - 1001 reserved
    - 1010 document interchange
    - 1011 structured field
    - 1100 reserved

1101 reserved
1110 reserved
1111 reserved
4 FMH-1 properties
bits 0-2, DSSEL: destination selection:
000 resume
001 end
010 begin
011 begin/end
100 suspend
101 end-abort
110 continue
111 reserved
bit 3, DST: data set transmission (see note 6):
0 transmission exchange format
1 basic exchange format
bit 4, reserved
bit 5, CMI: compression indicator (see notes 4 and 5):
0 no compression
1 compression (the first byte following FMH(s) is a string control byte)
bit 6, CPI: compaction indicator (see notes 4 and 5):
0 no compaction
1 compaction (the first byte following the FMH(s) is a string control byte)
bit 7, reserved
5 bits 0-7, ECRL: exchange record length if medium select = exchange or card; otherwise reserved. For medium select = card, a hexadecimal value indicates maximum card length:
00000000 80-column length
6-7 Reserved (optional)
8 DSLEN: length of destination name (optional)
9-n DSNAME: destination name (optional; reserved when DSSEL = continue)

Note 1: The data stream profile (DSP) defaults for the Medium Select Field are:

FMH-1 MEDIUM SELECT	DEFAULT DSP
Console, X'0'	Base
Exchange, X'1'	DST field of FMH-1
Card, X'2'	SCS (IRS, TRN)
Document, X'3'	Subset 2 (RJE)
Nonexchange Disk, X'4'	DST field of FMH-1
Extended Document,X'5'	Subset 2 (RJE)
Extended Card, X'6'	SCS (IRS, TRN)
WP Medium 1, X'8'	WP Raw Form
WP Medium 2, X'9'	WP Raw Form
WP Medium 3, X'A'	WP Raw Form
WP Medium 4, X'C'	WP Raw Form

An LU requiring any other DSP value associated with Medium Select must do so by specifying the desired DSP in byte 3, bits 4-7 of the FMH-1. This selection must adhere to those DSPs allowed on the session as specified in the BIND parameters.

Note 2: Medium Select and Logical Subaddress fields are reserved when the Destination Selection (DSSEL) field is set to 110 (continue), 001 (end), 100 (suspend), or 101 (end-abort).

Note 3: Medium Select = X'7' and Logical Subaddress = X'F', the Destination Name (DSNAME) field is used to select destination.

Note 4: CMI, CPI, and ERCL indicators are meaningful and valid only when specified in a Begin, Begin/end, or Continue FMH-1.

Note 5: CMI, CPI, and ERCL information received when DSSEL = Continue overlays the settings of the Begin FMH-1 or the last-received Continue FMH-1.

Note 6: When Medium Select is not equal to Exchange, this field is reserved. Receiver may do spooling and exchange-medium creation locally. When Medium Select = Exchange, specifying 0 preserves chain boundaries while spooling, but nonsequential allocation techniques may be used. Specifying 1 does not preserve chain boundaries, but uses sequential medium allocation.

FM header 2

Once a destination has been selected using a FMH-1, this header handles the data management tasks for that destination.

0	Length, in binary, of FMH-2, including this Length byte
1	bit 0, FMH concatenation: 0 no FMH follows this FMH-2 1 another FMH follows this FMH-2
	bits 1-7, type: 0000010
2	bit 0, SRI: stack reference indicator (see note below): 0 FMH-2 pertains to the active destination of the sending half-session's send stack and the receiving half-session's receive stack 1 FMH-2 pertains to the active destination of the receiving half-session's send stack and the sending half-session's receive stack
	bits 1-7, FMH-2 function to be performed (see note): nnnnnnn identifies the function that this FMH-2 is to perform
3-n	Parameter fields -- These fields provide the information needed to perform the selected function. They are different for each FMH-2 function, and are described in *Sessions Between Logical Units*.

Note: Byte 2 of the FMH-2 contains the Stack Reference indicator (SRI) and defines the function to be performed. The valid combinations of SRI and function codes are:

Function Code	Function
X'01'	Peripheral data information record (PDIR)
X'02'	Compaction table
X'04'	Prime compression character
X'07'	Execute program offline
X'20'	Create data set
X'21'	Scratch data set
X'22'	Erase data set
X'23'	Password
X'24'	Add
X'25'	Replace
X'26'	Add replicate
X'27'	Replace replicate
X'28'	Query for data set
X'29'	Note
X'2B'	Record ID
X'2C'	Erase record
X'2D'	Scratch all data sets
X'2E'	Volume ID
X'AA'	Note reply (SRI is always on)

FM header 3

This header handles data management tasks that are common to all destinations in the LU-LU session.

The FMH-3 format is identical to the FMH-2 format except that an FMH-3 does not have a Stack Reference indicator (SRI) in byte 2. An FMH-3 is used when information is needed or used by all destinations managed by the half-session. By contrast, an FMH-2 is used for a specific destination.

Two functions, the Compaction Table FMH and the Prime Compression Character FMH, can be sent as an FMH-2 or FMH-3. They should be sent as an FMH-2 when they apply to a specific destination at the half-session and as an FMH-3 when they apply to all destinations at the half-session.

The FMH-3 functions are as follows:

Function Code	Function
X'02'	Compaction table
X'03'	Query for compaction table
X'04'	Prime compression character
X'05'	Status
X'06'	Series ID

FM header 4

This header carries a logical block command and its parameters that, together with information, applies to a logical block within a logical message as defined for Logical Message Service.

0	Length, in binary, of FMH-4, including this Length byte
1	bit 0, FMH concatenation (must be 0) bits 1-7, type: 0000100
2	FMH4FXCT: Length of fixed length parameters excluding the length of FMH4FXCT. The first nonfixed parameter position is FMH4LBN. The minimum value of FMH4FXCT is 3, the maximum is 4.
3	FMH4TT1: block transmission type: X'00' inherit code (from MM-TT register) X'01'-X'3F' reserved X'40' FFR-FNI record X'41' FFR-FS record X'42' FFR-FS2 record X'43'-X'4F' reserved X'50'-X'FE' reserved X'FF' reserved <u>Note:</u> FFR=field formatted record, FNI=fixed fields without field separators, FS=fixed fields with field separators, FS2=fixed fields with or without field separators.
4	FMH4TT2: block transmission type qualifier: reserved except for FMH4TT1=X'41' or X'42', in which case it holds the separator value
5	FMH4CMD: Command: X'00' CRT-NU-BLK X'02' CRT-SU-BLK X'03' CRT-SN-BLK X'10' CONT-NU-BLK X'12' CONT-SU-BLK X'13' CONT-SN-BLK X'23' DEL-SN-BLK X'32' UPD-SU-BLK X'33' UPD-SN-BLK X'42' RPL-SU-BLK X'43' RPL-SN-BLK Other reserved <u>Note:</u> NU=nonshared, unnamed; SU=shared, unnamed; SN=shared, named; NN=nonshared, named
6	FMH4FLAG: flags (if omitted, X'00' is assumed): bits 0-1, reserved bits 2-3, F4RDESCR: record descriptor flag: 00 no logical record headers (LRHs) in transmission block 01 LRHs present, with implicit lengths 10 reserved 11 reserved bits 4-5, reserved bit 6, FMH4BDTF: block data transform flag: 0 FMH4BDT absent 1 FMH4BDT present bit 7, FMH4RDTF: reserved
m	FMH4LBN: Length of FMH4BN (X'00', or omitted, if unnamed block)

m+1	FMH4BN: Name of block
n	FMH4LBDT: Length of FMH4BDT (X'00 if FMH4BDTF is 0)
n+1	FMH4BDT: Block data transform
p	FMH4LVID: Length of FMH4VID
p+1	FMH4VID: Version identifier

FM header 5: Attach

LU type 6.2 uses this header to carry a request for a conversation to be established between two transaction programs. This header identifies the transaction program that is to be put into execution and connected to the receiving half-session.

When a transaction program issues an ALLOCATE verb naming a transaction program to be run at the other end of the conversation, an Attach FMH-5 carries the transaction program name (TPN) to the receiving LU.

0	Length, in binary, of FMH-5, including this Length byte
1	bit 0, reserved bits 1-7, type: 0000101
2-3	Command code: X'02FF' (Attach)
4	bit 0, security indicator: 0 user ID is not already verified 1 user ID is already verified bits 1-3, reserved bit 4, program initialization parameter (PIP) presence: 0 PIP not present following this FMH-5 1 PIP present following this FMH-5 (see "PIP Variable " on page 10-11 for format) bits 5-7, reserved
5	Length (j-5), in binary, of Fixed Length Parameters field (currently 3--future expansion possible)
6-j	Fixed Length Parameters
6	Resource type: X'D0' basic conversation X'D1' mapped conversation
7	Reserved
8(=j)	bits 0-1, synchronization level: 00 none 01 confirm 10 confirm, sync point, and backout 11 reserved bits 2-7, reserved
j+1-p	Variable Length Parameters
j+1-k	Transaction Program Name Field:
j+1	Length (values 1 to 64 are valid), in binary, of transaction program name
j+2-k	Transaction program name: a symbol string identifying a transaction program name known at the receiver; receivers may constrain such names to be type A, AE, GR, or DB, depending on the implementation
k+1-m	Access Security Information Field:
k+1	Length (0 or m-k-1), in binary, of Access Security Information subfields
k+2-m	Zero or more Access Security Information subfields (see "Access Security Information Subfields " on page 10-11 for format)
m+1-n	Logical-Unit-of-Work Identifier Field:
m+1	Length (values 0 and 10 to 26 are valid), in binary, of Logical-Unit-of-Work Identifier field
m+2-n	Logical-Unit-of-Work Identifier
m+2	Length (values 1 to 17 are valid), in binary, of fully qualified LU network name
m+3-w	Fully qualified LU network name (format described in "User Data Structured Subfield Formats" in "Chapter 7. User Data Structured Subfields")

w+1-w+6	Logical-unit-of-work instance number, in binary
w+7- w+8(=n)	Logical-unit-of-work sequence number, in binary
n+1-p	Conversation Correlator Field:
n+1	Length (values 0 to 8 are valid), in binary, of conversation correlator of sender
n+2-p	Conversation correlator of the sending transaction: a 1- to 8-byte symbol-string type G identifier (unique between partner LUs) of the conversation being allocated via FMH-5 (an example construction of this field would be the composition of a transaction program instance identifier and a resource identifier)

Note: Trailing Length fields (bytes n+1, m+1, and k+1) that have value X'00' can be omitted.

Access Security Information Subfields

The Access Security Information subfields in FMH-5 have the following formats:

0	Length (valid values are 1 to 11), in binary, of remainder of subfield--does not include this Length byte
1	Subfield type: X'00' profile X'01' password X'02' user ID
2-i	Data: a symbol string identifying access security information known at the receiver; receivers may constrain such information to be type A, AE, GR, or DB, depending on the implementation.

Note: The Access Security Information subfields may appear in any order in the Access Security Information field of the FMH-5.

PIP Variable

The PIP variable following FMH-5 Attach has the following format:

0-1	Length (4 or n+1), in binary, of PIP variable, including this Length field
2-3	GDS indicator: X'12F5'
4-n	Zero or more PIP subfields, each of which has the following format (shown in "PIP Subfield " using zero-origin)

PIP Subfield

Zero or more of these subfields are contained in a PIP variable (see "PIP Variable ").

0-1	Length, in binary, of PIP subfield, including this Length field
2-3	GDS indicator: X'12E2'
4-m	PIP subfield data: type G symbol string is valid

FM header 5 (Not LU Type 6.2)

This header flows from the program using the sending half-session to the attach manager of the receiving half-session. This header identifies the program at the receiving LU that it wishes to have attached. An FMH-5 can be followed by other FMHs (for example, FMH-6, FMH-8, and FMH-4), a logical record header (LRH), and FM data. Optionally, it can be sent with CD or EB.

0	Length, in binary, of FMH-5, including this Length byte
1	bit 0, FMH concatenation: 0 no FMH follows this FMH-5 1 another FMH follows this FMH-5
	bits 1-7, type: 0000101
2-3	FMH5CMD: command code: X'0202' attach transaction program X'0204' reset attached process X'0206' data descriptor
4	FMH5MOD: modifier
5	FMH5FXCT: fixed-length parameters: X'00' reset attached process X'02' attach transaction program, data descriptor
6	ATTDSP
7	ATTDBA
8-n	Resource Names

FM header 6

This header flows from a currently active transaction program using a sending half-session to a currently active transaction program using a receiving half-session.

0	Length, in binary, of FMH-6, including this Length byte
1	bit 0, FMH concatenation: 0 no FMH follows this FMH-6 1 another FMH follows this FMH-6 bits 1-7, type: 0000110
2-3	Command code (CC2): For service transaction programs, the first byte of the command code identifies a transaction program and the second byte identifies a function within a transaction program.
4	FMH6MOD: modifier bit 0, FMH6LNSZ: length of parameter length fields: 0 1-byte field 1 2-byte field bits 1-7, reserved
5-n	Fixed: total length of fixed length parameters (LF): This field contains the sum of the lengths of all fixed length parameters that are mandatory for the particular command code located in bytes 2 and 3. This field is either one byte or two bytes in length based on the setting of FMH6LNSZ (0 = one byte; 1 = two bytes).
n+1-m	Fixed length parameters (FDy): the fixed length parameters are positional by command code
m+1-p	Variable: length field of first, positional variable-length parameter (LV1): This field is either one byte or two bytes in length based on the setting of FMH6LNSZ (0 = one byte; 1 = two bytes). If the Length field (LVx) is equal to 0, then the variable parameter is omitted. The next positional variable-length parameter length (LV2) occurs in byte q+1.
p+1-q	Variable-Length Positional Parameter (VD). The LV and VD fields are replicated to represent x number of variable-length parameters according to command code.

FM header 7: Error Description

LU type 6.2 uses this header, following negative response (0846), to carry information that relates to an error on the session or conversation. For example, an FMH-7 and additional error information are sent when an FMH-5 (Attach) specifies a nonexistent transaction program name.

0	Length (7), in binary, of FMH-7, including this Length byte
1	bit 0, reserved
	bits 1-7, type: 0000111
2-5	SNA-defined sense data (see below)
6	bit 0, error log variable presence:
	0 no error log variable follows this FMH-7
	1 error log GDS variable follows this FMH-7
	bits 1-7, reserved

Note: Only the following sense data (in hexadecimal) can be sent in an LU 6.2 FMH-7. Sense data carried in non-LU 6.2 FMH-7 varies by implementation. See "Chapter 9. Sense Data" for additional details on sense data. The phrases following the sense data are the symbolic return codes provided to the application program in LU 6.2 verbs (see SNA Transaction Programmer's Reference Manual for LU Type 6.2) when the sense data is received.

Sense Data	Return Code
1008600B	RESOURCE_FAILURE_NO_RETRY
10086021	ALLOCATION_ERROR--TPN_NOT_RECOGNIZED
10086031	ALLOCATION_ERROR--PIP_NOT_ALLOWED
10086032	ALLOCATION_ERROR--PIP_NOT_SPECIFIED_CORRECTLY
10086034	ALLOCATION_ERROR--CONVERSATION_TYPE_MISMATCH
10086041	ALLOCATION_ERROR--SYNC_LEVEL_NOT_SUPPORTED_BY_PGM
080F6051	ALLOCATION_ERROR--SECURITY_NOT_VALID
08240000	BACKED_OUT
084B6031	ALLOCATION_ERROR--TRANS_PGM_NOT_AVAIL_RETRY
084C0000	ALLOCATION_ERROR--TRANS_PGM_NOT_AVAIL_NO_RETRY
08640000	DEALLOCATE_ABEND_PROG
08640001	DEALLOCATE_ABEND_SVC
08640002	DEALLOCATE_ABEND_TIMER
08890000	PROG_ERROR_NO_TRUNC or PROG_ERROR_PURGING
08890001	PROG_ERROR_TRUNC
08890100	SVC_ERROR_NO_TRUNC or SVC_ERROR_PURGING
08890101	SVC_ERROR_TRUNC

FM header 7: Error Description (not LU 6.2)

This header is sent after a negative response (0846) to provide further information about an error.

0	Length, in binary, of FMH-7, including this Length byte
1	bit 0, FMH concatenation: 0 no FMH follows this FMH-7 1 reserved bits 1-7, type: 0000111
2-5	ERPSENSE: SNA-defined sense data, which would appear on error response (see Chapter 9, Sense Data)
6-7	ERPSEQ: sequence number of RU chain in which error was detected

FM HEADER 8

This header is used only with IMS/VS logical message services that utilize LU type 6.1 protocols. Refer to the IMS publications for the formats and meanings of the bytes in this header.

FM header 10

This header is sent to prepare the session for a sync point. It may be sent with data. The RU chain must have CD set on so that the receiver may, on the next flow, request a sync point or abort the unit of work.

0	Length, in binary, of FMH-10, including this length byte
1	bit 0, FMH concatenation: 0 no FMH follows this FMH-10 1 another FMH follows this FMH-10 bits 1-7, type: 0001010
2-3	SPCCMD: sync point command: X'0202' prepare command
4-5	SPCMOD: sync point modifier For a prepare command (FMH-10), the modifier indicates RH settings to be returned on the first RU chain sent by the FMH-10 receiver. X'0000' *CD, *EB. The sender of FMH-10 does not care what RH settings are returned on the reply. X'0001' EB. The sender of FMH-10 requires an EB on the reply. X'0002' CD, ¬EB. The sender of FMH-10 requires a CD on the reply.

FM header 12: Security

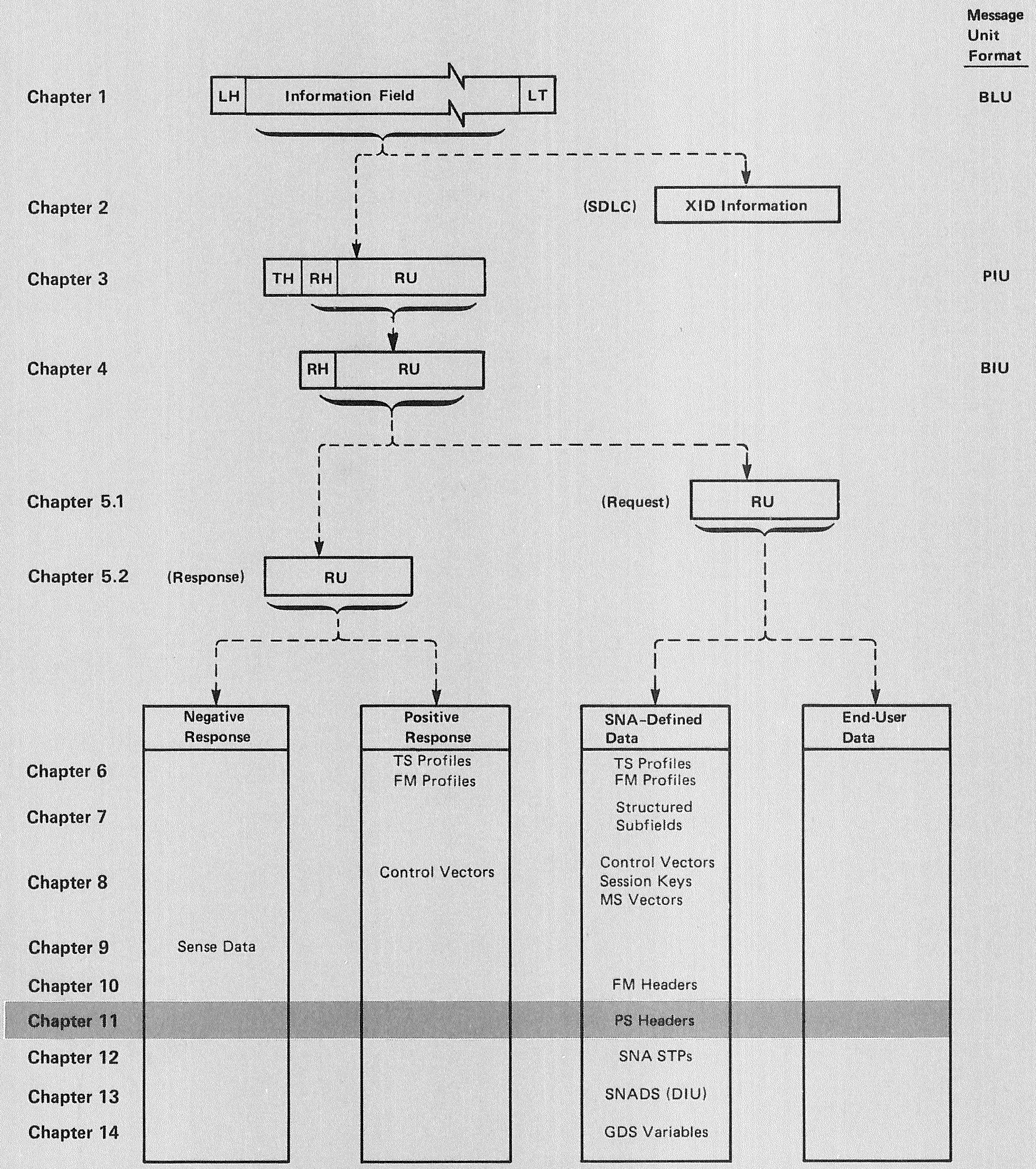
LU type 6.2 uses this header to carry enciphered data that it received in the clear in +RSP(BIND) for LU-LU session password verification. An FMH-12 is the third flow of LU-LU session password verification information. (BIND and +RSP(BIND) are the other two.)

The function management header 12 (FMH-12) has the following format:

0	Length (10), in binary, of FMH-12, including this Length byte.
1	bit 0, reserved bits 1-7, type: 0001100
2-9	Enciphered version of the random data received in RSP(BIND)

Chapter 11

Presentation Services Headers



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 11. PRESENTATION SERVICES HEADERS

PRESENTATION SERVICES (PS) HEADERS

Presentation services (PS) headers convey information between PS component sync point managers when the conversation using the session is allocated with the sync-point synchronization level. These headers are only used by LU type 6.2. *SNA Format and Protocol Reference Manual: Architecture Logic for LU Type 6.2* describes the use of these PS headers.

Transaction program data is delimited using a 2-byte length field called an LL, containing a value that is the number of bytes contained in the transaction program data plus 2 (the length of the LL field itself).

LL	transaction program data

All PS headers are identified by an LL of X'0001' immediately preceding the header. X'0001' is an invalid LL value for use by transaction programs because the LL's value must include the length of itself, which is 2 bytes. Therefore, all LLs indicating a length of less than 2 are reserved for use by the LU. The format of PS headers is shown below.

PS header 10: Sync Point Control

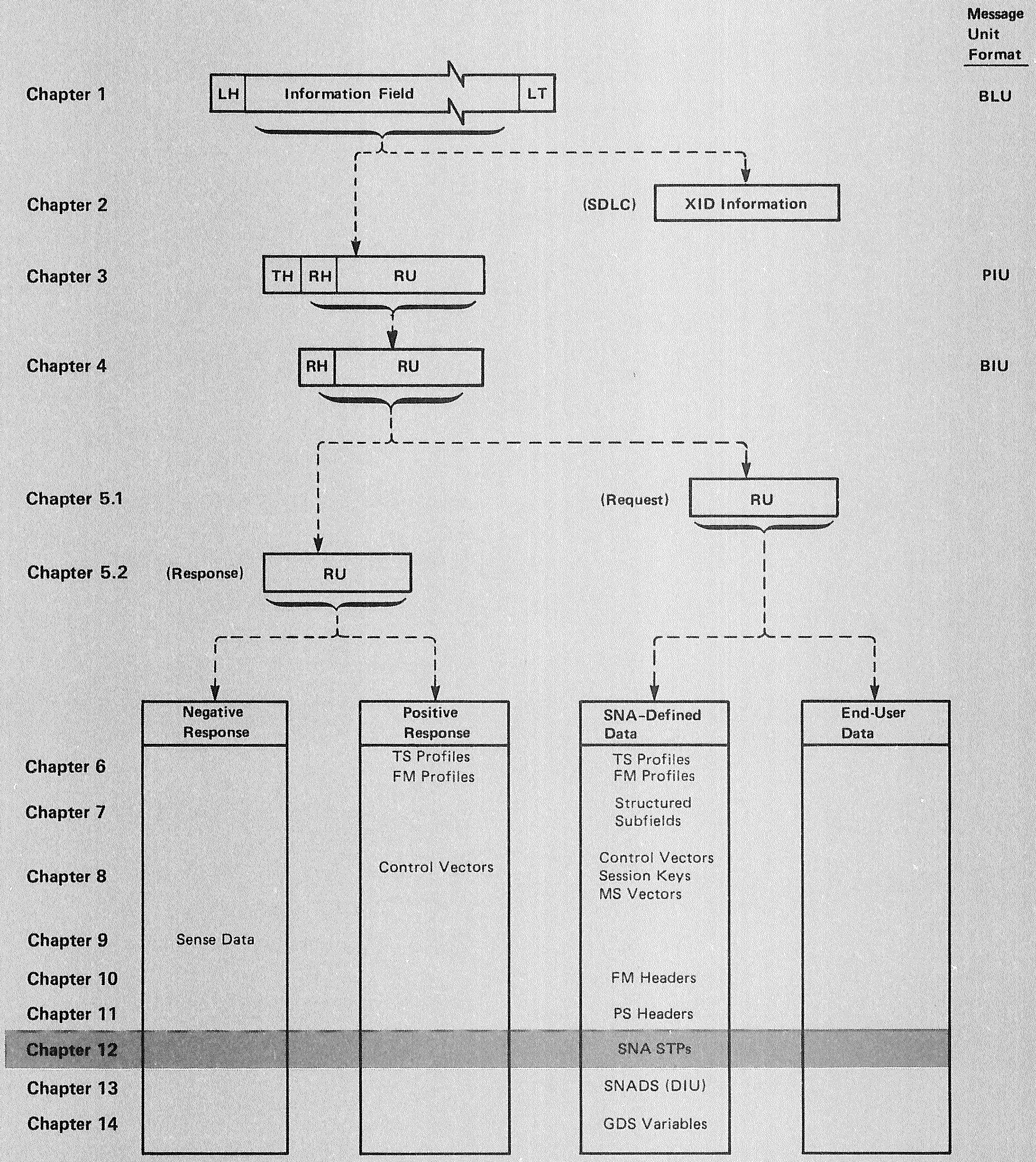
Presentation services header 10 (Sync Point Control) has the following format:

0	Length, in binary, of PS header, including this length field
1	bit 0, reserved bits 1-7, type: 0001010 sync point control (only value defined)
2-3	Sync point command type: X'0005' Prepare X'0006' Request Commit X'0007' Committed X'0008' Forget X'0009' Heuristic Mixed
4-5	Modifier specifying next flow (present only if bytes 2-3 = X'0005' or X'0006'; reserved when bytes 2-3 = X'0006' and 2-phase sync point being used): X'0000' request RECEIVE X'0001' request DEALLOCATE X'0002' request SEND *Note:* Bytes 4-5 affect the change direction indicator (CDI) and conditional end bracket indicator (CEBI) settings of the RH for the last PS header in the sync point sequence. For example, Forget command type when Prepare was the first PS header received, and Committed command type when Request Commit was the first PS header received. (See *SNA Format and Protocol Reference Manual: Architecture Logic for LU Type 6.2* for details.)

Chapter 12

General Data Stream (GDS) Variables

For SNA Service Transaction Programs



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 12. GDS VARIABLES FOR SNA SERVICE TRANSACTION PROGRAMS

LIST OF SNA SERVICE TRANSACTION PROGRAMS

Logical Unit type 6.2 service transaction programs are identified by a transaction program name (TPN) that begins with a value of X'06'. Other SNA service transaction programs are identified similarly. Figure 12-1 identifies the transaction program names that SNA currently defines. These TPNs are specified in an FM header type 5 (FMH5 Attach).

TP Name	TP Description
X'06F1****'	LU 6.2 Change Number of Sessions
X'06F2****'	LU 6.2 Sync Point Resynchronization
X'07F0F0F1'	DDM Synchronous Conversation
X'20F0F0F0'	DIA PROCESS Destination TP
X'20F0F0F1'	DIA SERVER TP
X'21F0F0F1'	DS SEND TP
X'21F0F0F2'	DS RECEIVE TP
X'21F0F0F3'	DS ROUTER DIRECTOR TP
X'21F0F0F6'	DS General Server TP

* = any hexadecimal digit

Figure 12-1. SNA-defined Service Transaction Programs

Refer to "Chapter 13. SNA Distribution Services" for additional SNADS information and refer to "Chapter 14. GDS Variables for General Use" for information about GDS variables that are not specific to SNA service transaction programs.

DESCRIPTIONS OF GDS VARIABLES FOR SNA STPS

CHANGE NUMBER OF SESSIONS (X'1210') GDS Variable

0-1 Length (17 or n+1), in binary, of Change Number of Sessions GDS variable, including this Length field

2-3 GDS ID: X'1210'

4 Service flag:
bits 0-3, reserved
bits 4-7, request/reply indicator:
- 0010 request
- 1000 reply, function completed abnormal
- 1010 reply, function accepted but not yet completed

5 Reply modifier (reserved if byte 4, bits 4-7 = 0010):
- X'00' normal--no negotiation performed
- X'01' abnormal--command race detected
- X'02' abnormal--mode name not recognized
- X'03' reserved
- X'04' normal--negotiated reply
- X'05' abnormal--(LU,mode) session limit is 0

6 Action:
- X'00' set (LU,mode) session limits
- X'01' reserved
- X'02' close

7 Drain immediacy:
bits 0-2, reserved
bit 3, source LU drain (reserved if byte 6 ¬= 02):
- 0 no (send BIS at next opportunity)
- 1 yes

bits 4-6, reserved
bit 7, target LU drain (reserved if byte 6 ¬= 02):
- 0 no (send BIS at next opportunity)
- 1 yes

8 Action flags:
bits 0-6, reserved
bit 7, session deactivation responsibility:
- 0 sender of Change Number of Sessions request (source LU)
- 1 receiver of Change Number of Sessions request (target LU)

Note: Bytes 9-14 are reserved if byte 6 ¬= 0.

9-10 (LU,mode) session limit:
bit 0, reserved
bits 1-15, maximum (LU,mode) session count, in binary

11-12 Source LU contention winners:
bit 0, reserved
bits 1-15, guaranteed minimum number of contention winner sessions at source LU, in binary

13-14 Target LU contention winners:
bit 0, reserved
bits 1-15, guaranteed minimum number of contention winner sessions at target LU, in binary

15 Mode name selection:
bits 0-6, reserved
bit 7, mode names affected by this command:

	0 a single mode name is affected 1 all mode names are affected
16	Length (values 0 to 8 are valid; reserved if byte 15, bit 7 = 1), in binary, of mode name
17-n	Mode name (omitted if byte 16 = X'00')

EXCHANGE LOG NAME (X'1211') GDS Variable

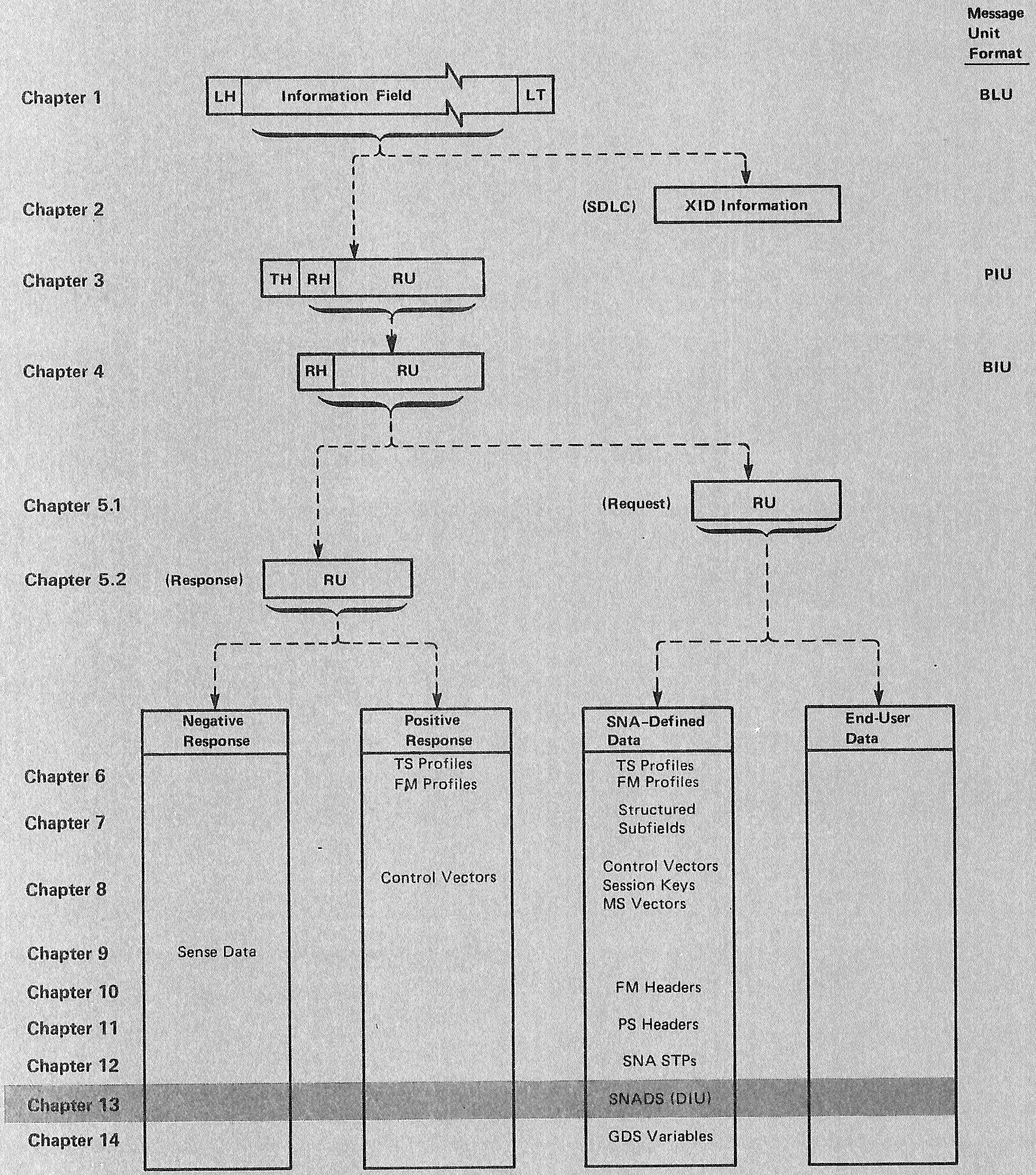
0-1	Length (p+1), in binary, of Exchange Log Name GDS variable, including this Length field
2-3	GDS ID: X'1211'
4	Service flag:
	bits 0-3, reserved
	bits 4-7, request/reply indicator:
	0010 request
	1000 reply, function completed abnormally
	1001 reply, function completed normally
5	Sync point manager flags:
	bits 0-6, reserved
	bit 7, log status:
	0 cold
	1 warm
6	Length (values 1 to 17 are valid), in binary, of fully qualified LU network name
7-n	Fully qualified LU network name (format described in "Chapter 7. User Data Structured Subfields")
n+1	Length (values 1 to 64 are valid), in binary, of log name
n+2-p	Log name (symbol-string type-AE)

COMPARE STATES (X'1213') GDS Variable

0-1	Length (p+1), in binary, of Compare States GDS variable, including this Length field
2-3	GDS ID: X'1213'
4	Service flag: bits 0-3, reserved bits 4-7, request/reply indicator: 0010 request 1000 reply, function completed abnormally 1001 reply, function completed normally
5	Sync point manager state: X'01' RESET X'02' SYNC_POINT_MANAGER_PENDING X'03' IN_DOUBT X'04' COMMITTED X'05' HEURISTIC_RESET X'06' HEURISTIC_COMMITTED X'07' HEURISTIC_MIXED
6	Reserved
7	Length, in binary, of Logical-Unit-of-Work Identifier field (values 10 to 26 are valid)
8-n	<u>Logical-Unit-of-Work Identifier</u>
8	Length, in binary, of fully qualified LU network name (values 1 to 17 are valid)
8-w	Fully qualified LU network name (format described in "Chapter 7. User Data Structured Subfields")
w+1-w+6	Logical-unit-of-work instance number, in binary
w+7- w+8(=n)	Logical-unit-of-work sequence number, in binary
n+1	Length (values 0 to 8 are valid), in binary, of conversation correlator
n+2-q	Conversation correlator of transaction program that allocated the conversation that failed: see FMH-5 for format of this correlator
q+1	Length (values 2 to 8 are valid), of session instance identifier
q+2-p	Session instance identifier of session being used by conversation at time of failure (See "Chapter 7. User Data Structured Subfields" for the format of this identifier.)

Chapter 13

SNA Distribution Services



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 13. SNA DISTRIBUTION SERVICES

INTRODUCTION

This chapter presents the encodings of the SNA Distribution Services (SNADS) interchange units (IUs) used to transport information and control between the distribution service units (DSUs). It has two parts:

- "Interchange Unit Description" on page 13-2 presents the format and the semantics of the IUs

- "Names and Code Points" on page 13-41 presents all the IU code points, transaction program names, and the server names

The syntax of the IUs follows the Distribution Interchange Architecture: IUs contain a prefix, command, object (optionally), and a suffix. The IUs are sent between DSUs on LU 6.2 conversations. Some of the SNADS IUs may be segmented.

DISTRIBUTE INTERCHANGE UNIT

DISTIU; DSU ---> DSU; (DISTRIBUTE INTERCHANGE UNIT)

DISTIU carries data and status for distribution to one or more users.

DISTIU consists of a sequence of GDS variables as follows:

- A prefix
- A command
- A distribution object (optional)
- A suffix

See Figure 13-1 on page 13-3.

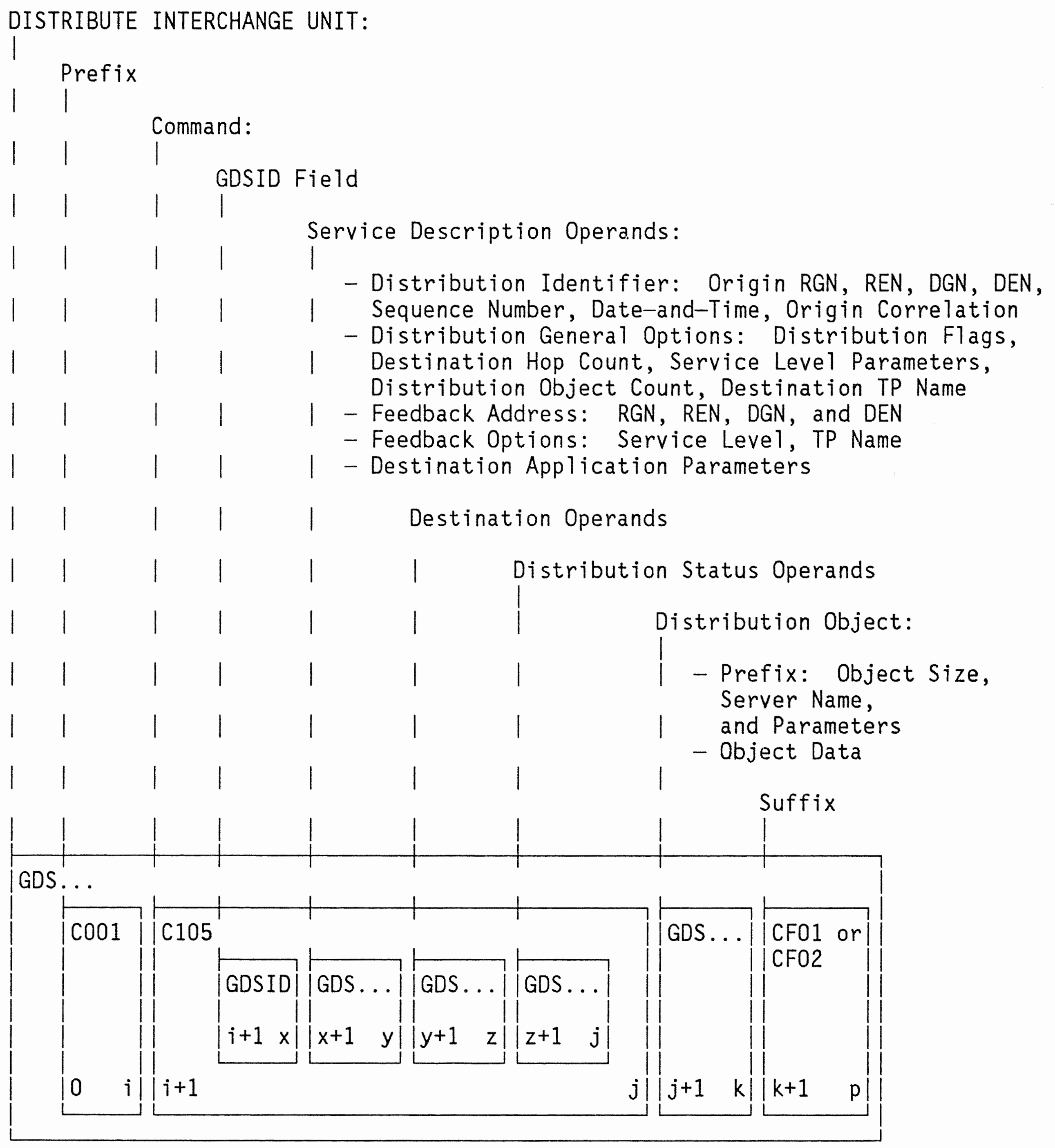


Figure 13-1. Structure of the Distribute IU

Note: The identifiers (e.g., C001, C105) above are shown to identify portions of the data stream. The two identifier bytes are always preceded by two length bytes in the encoded data stream.

0-i Prefix

Required GDS variable, identifies the beginning of an interchange unit (IU)

0-1 Length (i+1), in binary, of the prefix, including this Length field (values from 5 to 21 are valid)

2-3 Identifier: X'C001'

4 Format:
X'02' no segmentation descriptor follows

5-i Interchange unit identifier, optional subfield, a 1- to 16-byte correlator of the sending DSU. If an error occurs, the receiving DSU returns this value in an Acknowledge IU, see "Acknowledge Interchange Unit" on page 13-34.
Note: A particular correlator value is optional and is used by a send-receive DSU pair; it has no meaning through the entire SNADS network.

i+1-j Command

Required sequence of one or more GDS variables that contains:

- A sequence of Service Description operands
- A sequence of Destination operands
- A sequence of Distribution Status operands

See Figure 13-1 on page 13-3.

i+1-x GDSID field

i+1-i+2 Length (j-i), in binary, of the GDS variable, including this Length field
Note: The maximum length of the command, including all segments, is 32511.

i+3-i+4 Identifier: X'C105'

i+5 Format:
X'02' no segmentation descriptor follows
X'82' segmentation descriptor follows

i+6-i+8 Segmentation Descriptor
(present only if Format=X'82')

i+6 Position in the command:
X'00' last or only segment (this is the last or only command segment sent)
X'20' not the last segment of the command (at least one more command segment follows contiguously on this conversation[1])

i+7-i+8 Segment sequence number:
X'0000' segment sequence numbering not used

x=(i+5)|(i+8) for i+5 containing X'02'|X'82', respectively

x+1-y Service Description Operands

Required set of GDS variables containing:

- Distribution Identifier

---

[1] SNADS command and distribution object data may be divided into smaller segments to send across the conversation. The last segment is sent before any other different GDS variable is sent.

- Distribution General Options
- Feedback Address (optional)
- Feedback Options (optional)
- Destination Application parameters (optional)

The above components can be in any order. See Figure 13-1 on page 13-3.
Note: Throughout the DISTIU and ACKIU presentation, the offset of the first byte of a subfield is reset to 0 if the subfield can appear in positions (within the field) different from the position shown in the offset-stream field description.

0-n Distribution Identifier

Required GDS variable. It contains the following LT-subfields:

- Origin RGN (optional)
- Origin REN
- Origin DGN
- Origin DEN
- Origin Sequence number
- Origin Date and time
- Origin Correlation (optional)

See Figure 13-1 on page 13-3. The LT-subfields can be in any order.

0-1 Length (n+1), in binary, of the GDS variable, including this Length field (values from 28 to 107 are valid)
Note: The quadruple (origin DGN, origin DEN, Sequence number, Date and time) is the unique distribution identifier (UDI) in the whole SNADS network.
2-3 Identifier: X'C340'
4 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

0-n1 Origin RGN

Optional LT-subfield
0 Length (n1+1), in binary, of the subfield including this Length field (values from 3 to 10 are valid)
1 Type: X'01'
2-n1 RGN, filled in at the origin DSU, 1- to 8-character string (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), first part of the name of the DSU where DISTIU originated, obtained from the origin DSU directory

0-n2 Origin REN

Required LT-subfield
0 Length (n2+1), in binary, of this subfield including this Length field (values from 3 to 10 are valid)
1 Type: X'02'
2-n2 REN, 1- to 8-character string (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used),

second part of the DSU name where DISTIU originated, obtained from the origin DSU directory

0-n3 Origin DGN

Required LT-subfield
0 Length (n3+1), in binary, of this subfield including this Length field (values from 2 to 10 are valid)
1 Type: X'03'
2-n3 DGN, 0- to 8-character string (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), first part of the distribution user name (DUN) that initiated DISTIU, passed by DISTRIBUTE_DATA or DISTRIBUTE_STATUS verbs
Note: The name is null (i.e., the length of the LT-subfield is 2) if DISTIU is a status DISTIU generated by a DSU.

0-n4 Origin DEN

Required LT-subfield
0 Length (n4+1), in binary, of the subfield including this Length field (values from 2 to 10 are valid)
1 Type: X'04'
2-n4 DEN, 0- to 8-character string, (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), second part of the, distribution user name (DUN) that initiated DISTIU, passed by DISTRIBUTE_DATA or DISTRIBUTE_STATUS verbs
Note: The name is null (i.e., Length of the LT-subfield is 2) if DISTIU is a status DISTIU generated by a DSU.

0-5 Origin Sequence Number

Required LT-subfield
0 Length (6), in binary, of the subfield, including this Length field
1 Type: X'05'
2-5 Integer in base 10, 4-numeric EBCDIC encoded character string (values from 'F0F0F0F1' to 'F9F9F9F9' generated by modulo-10000 incrementing counter [with 1, 2, 3, ..., 9999 as states] for DISTIUs of type=DATA, and 'F0F0F0F0' for DISTIUs of type=STATUS), when the distribution verb is invoked or for STATUS DISTIU, when the feedback is generated

0-9 Origin Date and Time

Required LT-subfield, contains the date and time generated by SNADS when the distribution verb was issued.
0 Length (10), in binary, of the subfield including this Length field
1 Type: X'06'
2-5 Date:
2-3 Year, in binary (e.g., year 1983 is encoded as X'07BF')
4 Month of the year, in binary (values from 1 to 12 are valid)
5 Day of the month, in binary (values from 1 to 31 are valid)
6-9 Time:
6 Hour of the day, in binary (values from 0 to 23 are valid)
7 Minute of the hour, in binary (values from 0 to 59 are valid)
8 Second of the minute, in binary (values from 0 to 59 are valid)

9 Hundredth of the second, in binary (values from 0 to 99 are valid)

0-n5 Origin Correlation

Optional LT-subfield
0 Length (n5+1), in binary, of the subfield including this Length field (values from 3 to 46 are defined)
1 Type: X'07'
2-n5 String supplied by application transactions, not used by SNADS

0-n Distribution General Options

Required GDS variable, describes the processing SNADS performs upon the DISTIU. It contains the following LT-subfields:

- Distribution Flags
- Destination Hop Count
- Service Level parameters
- Distribution Object Count
- Destination TP Name

See Figure 13-1 on page 13-3. The LT-subfields can be in any order.

0-1 Length (n+1), in binary, of the GDS variable, including this Length field (values from 30 to 114 are valid)
2-3 Identifier: X'C33D'
4 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

0-2 Distribution Flags

Required LT-subfield, services indicated specified by the application transaction
0 Length (3), in binary, of the LT-subfield, including this Length field
1 Type: X'01'
2 Flags:
bit 0, no-feedback request bit:
  0 SNADS is requested to generate feedback in case of error
  1 no SNADS feedback for this DISTIU is requested
bit 1, DISTIU type bit:
  0 DISTIU is a data distribution
  1 DISTIU is a status distribution
bit 2, reserved, DISTIU not rejected whatever its value
bits 3-7, reserved[2]
Note: The functions encoded by the Flags field are described in Figure 13-2 on page 13-8.

---

[2] Reserved fields are to be used for later releases; unless specified otherwise, their values are passed unchanged by the intermediate nodes, and, if not 0, rejected by the destination nodes.

Flag (hex)	Interpretation of the Flag
00	Data distribution with feedback
80	Data distribution with no feedback
C0	Status distribution with no feedback

Figure 13-2. Interpretation of the Distribution Flags

0-3 Destination Hop Count

Required LT-subfield
0 Length (4), in binary, of the subfield including this Length field
1 Type: X'02'
2-3 Number, in binary, of hops that may not be exceeded by DISTIU in its way toward its destination DSUs, set by the DSU where DISTIU originated, decremented by one every time the DISTIU is sent to another DSU. When its value is 0 and this DSU is not the final destination of the DISTIU, the DISTIU has a "hop count" error.

0-n1 Service Level Parameters

Required LT-subfield, provided by the application transaction to describe the requested functions for the DISTIU
0 Length (n1+1), in binary, of Service Level parameters, including this Length field (values 11, 14, 17, ... 32 are valid)
1 Type: X'03'
2-4 Priority parameter, priority level to be given to the distribution, in decreasing priority order:
X'01C0F0' FAST priority
X'01C0D0' STATUS priority
X'01C080',..., X'01C0xx',..., and X'01C008' data priority levels, where 'xx' is the value of the priority level DATA_i, and 'xx'= [(8 * i) in hexadecimal]; e.g., 'xx'='80' for i=16, and 'xx'='08' for i=1
Note: If only two data priorities are honored then the high data priority would map to priorities DATA_16 to DATA_9, and the low data priority to priorities DATA_8 to DATA_1.
5-7 Protection parameter, requirement for minimum protection against distribution losses in the data store:
X'02C010' no protection required, distribution object can be stored on volatile storage
X'02C030' protection required, distribution object must be stored on non-volatile storage
8-10 Capacity parameter, requirement for a route which can provide the storage of distribution data of specified size:
X'03C000' NONE capacity, used for DISTIUs without distribution objects
X'03C00C' 4K capacity, used for DISTIUs with distribution objects whose sizes are smaller than 4096 bytes
X'03E0FF' INDEF capacity, used for DISTIUs with distribution objects whose sizes are relatively large, for example larger than 4096 bytes

Notes:

1. The capacity requirement is for the distribution object, and does not include the capacity needed to store and handle the control subfields of the DISTIU. For example, the 3K required to handle one distribution object of 3K byte size is above the storage needed to store the associated command.

2. The defined values for (priority, protection, capacity) parameters are shown in Figure 13-3.

3. Implementations may accept other service levels as long as they can route IUs responsibly.

Service Level Parameter			Comments
Priority	Protection	Capacity	
01C0F0	02C010	03C000	Data or Status DISTIU with priority=FAST, no protection, and no capacity requested
01C0F0	02C010	03C00C	Data or Status DISTIU with priority=FAST, no protection, and 4K capacity requested
01C0D0	02C030	03C00C	Status DISTIU with priority=STATUS, protected, and 4K capacity requested
01C0D0	02C030	03C000	Status DISTIU with priority=STATUS, protected, and no capacity requested
01C0xx	02C030	03E0FF	Data DISTIU with priority=DATA-i (see Note) protected, and indefinite capacity
Note: 'xx'=[(8*i)], in hexadecimal, is the priority level corresponding to DATA_i, where i=1, 2,..., 15, 16.			

Figure 13-3. Priority, Protection, and Capacity Subfield Values

11-n1 Optional Service Level Parameters

An optional stream of up to 7 3-byte fields with the following structure:

0-2 Optional parameter

0 Parameter identifier: any value larger than X'03'

1 Comparison operator:
X'00'-'0F' optional request, is ignored by a receiving product in the intermediate role that allows the transfer of the DISTIU
≥ X'10' mandatory request, if not recognized by a product results in the "function not supported" error condition, the DISTIU is rejected

2 Parameter value: defined by product, refer to its specifications for the definition of the field

0-3	Distribution Object Count
	Required LT-subfield
0	Length (4), in binary, of the Distribution Object Count including this Length field
1	Type: X='04'
2-3	Number, in binary, of distribution objects present in this DISTIU (values 0 and 1 defined)

0-n2	Destination TP Name
	Required LT-subfield
0	Length (n2+1), in binary, of the subfield including this Length field (values from 3 to 66 are valid)
1	Type: X'05'
2-n2	TP Name, character string, (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), started by the destination DSU after the DISTIU is enqueued for delivery

0-n	Feedback Address
	Optional GDS variable, valid only if DISTIU type is DATA, contains: • Feedback RGN (optional) • Feedback REN • Feedback DGN • Feedback DEN The LT-subfields can be in any order. See Figure 13-1 on page 13-3.
0-1	Length (n+1), in binary, of the GDS variable including this Length field (values 14 to 45 are valid)
2-3	Identifier: X'C360'
4	Format: X'41' no segmentation descriptor follows, contains LT-subfields

0-n1	Feedback RGN
	Optional LT-subfield
0	Length, in binary, of the subfield including this Length field (values from 3 to 10 are valid)
1	Type: X'01'
2-n1	Name, character string, first part of the name of the DSU where the feedback is requested to be sent

0-n2	Feedback REN
	Required LT-subfield
0	Length (n2+1), in binary, of the LT-subfield including this Length field (values from 3 to 10 are valid)
1	Type: X'02'
2-n1	Name, character string, second part of the DSU name where feedback is requested to be sent

0-n3	Feedback DGN

Required LT-subfield
0 Length (n3+1), in binary, of the subfield, including this Length (values from 3 to 10 are valid)
1 Type: X'03'
2-n3 Name, character string, first part of the distribution user name (DUN) to which feedback is to be sent

0-n4 Feedback DEN

Required LT-subfield
0 Length (n4+1), in binary, of the subfield including this Length (values from 3 to 10 are valid)
1 Type: X'04'
2-n4 Name, character string, second part of the distribution user name (DUN) to which feedback is requested to be sent

0-n Feedback Options

Optional GDS variable, valid only if the distribution is of type DATA, defined by the application transaction to specify the functions needed for the feedback DISTIU, containing:

- Feedback Service Level (optional)
- Feedback TP Name (optional)

See Figure 13-1 on page 13-3. The LT-subfields can be in any order.
0-1 Length (n+1), in binary, of the GDS variable, including this Length field (values 8 to 103 are valid)
2-3 Identifier: X'C343'
4 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

0-n1 Feedback Service Level

Optional LT-subfield. If omitted, the service level of the feedback DISTIU is calculated as follows: The priority is FAST if this DISTIU has priority FAST, the priority is STATUS for other priorities, the protection is the same as the protection of this DISTIU, and the capacity is 4K or NONE.
0 Length (n1+1), in binary, of Feedback Service Level, including this Length field (values 11, 14, 17, ... 32 are valid)
1 Type: X'01'
2-4 Feedback Priority Parameter, priority level to be given to the feedback DISTIU, in decreasing priority order:
X'01C0F0' FAST priority
X'01C0D0' STATUS priority
5-7 Feedback Protection Parameter, requirement for minimum protection against feedback distribution object losses in the data store:
X'02C010' no protection required, feedback distribution object can be stored on volatile memory
X'02C030' protection required, feedback distribution object must be stored on non-volatile memory
8-10 Feedback Capacity Parameter, requirement for a route which can provide the storage of feedback distribution data of specified size:

X'03C000' NONE capacity, used for DISTIUs without feedback distribution objects
X'03C00C' 4K capacity, used for DISTIUs with feedback distribution objects whose sizes are smaller than 4096 bytes

Note: The feedback capacity requirement is for the distribution object and does not include the capacity needed to store and handle the control subfields of the feedback DISTIU.
The defined values for (feedback priority, feedback protection, feedback capacity) parameters are shown in Figure 13-4.

Feedback Service Level Parameter			Comments
Priority	Protection	Capacity	
01C0F0	02C010	03C000	Feedback with priority=FAST, no protection and no capacity requested
01C0F0	02C010	03C00C	Feedback with priority=FAST, no protection and 4K capacity requested
01C0D0	02C030	03C00C	Feedback with priority=STATUS, protected, and 4K capacity requested
01C0D0	02C030	03C000	Feedback with priority=STATUS, protected, and no capacity requested

Figure 13-4. Feedback Priority, Protection, and Capacity Subfield Values

11-n1 Feedback Optional Service Level Parameter

An optional stream of up to seven 3-byte fields with the following structure:
0-2 Optional parameter
0 Parameter identifier: any value larger than X'03'
1 Comparison operator:
X'00'-'0F' optional request, is ignored by a receiving product in the intermediate role that allows the transfer of the DISTIU
≥ X'10' mandatory request, if not recognized by an implementation, results in the "function not supported" error condition, the DISTIU is rejected
2 Parameter value: defined by product, refer to its specifications for the definition of the field

0-n2 Feedback TP Name

Optional LT-subfield, if omitted the transaction application program to be started after the feedback is queued for delivery is given by the Destination TP Name within Distribution General Options GDS variable
0 Length (n2+1), in binary, of the LT-subfield including this Length field (values from 3 to 66 are valid)

1	Type: X'02'
2	Name, character string, (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), name of the transaction application program to be started after the feedback is queued for delivery
0-n	Destination Application Parameters (see Figure 13-1 on page 13-3)
	Optional GDS variable
0-1	Length (n+1), in binary, of the GDS variable including this Length field (values from 6 to 517 are valid)
2-3	Identifier: X'C32D'
4	Format: X'01' no segmentation descriptor follows
5-n	Parameters, to 512-byte stream, supplied by the application transaction that invoked the distribution verb and passed as a returned parameter when RECEIVE_DISTRIBUTION is issued.
y+1	replaced by b1

Notes:

1. Offset y+1 will be replaced by b1 in order to simplify the stream description.

2. Offsets bi and ei+1 are used for the first byte of the beginning and the end of a list, as it follows: b1, e1+1 for Destination operands, b2, e2+1 for REN List, b3, e3+1 for DGN List, and b4, e4 for DEN List.

b1-e1+5 Destination Operands

List of destinations of the DISTIU, no larger than 256 destinations if DISTIU is of type DATA and exactly one destination if the DISTIU is of type STATUS, encoded as a list of factored lists of DUNs and DSUNs. The identical RGNs, RENs, and DGNs are factored out of the Destination operands, REN, and DGN lists. The Destination operands list may be fully factored, partially factored, or unfactored (see the following example).

Example: Following is a list of destinations (qualified by RGN.REN.DGN.DEN):

A.K.DA.U1, A.K.DA.U2, A.K.DB.U3, A.K.DB.U4,
A.L.DC.U5, A.L.DC.U6, A.L.DD.U7, A.L.DD.U8,
B.M.DE.U9, B.M.DE.U10, B.M.DF.U11, B.M.DF.U12,
B.N.DG.U13, B.N.DG.U14, B.N.DH.U15, and B.N.DH.U16.

The list may appear factored in Destination operands as follows:

- completely factored:
  ```
  (A(K(DA(U1 U2) DB(U3 U4))
    L(DC(U5 U6) DD(U7 U8)))
   B(M(DE(U9 U10) DF(U11 U12))
    N(DG(U13 U14) DH(U15 U16))))
  ```

- partially factored:

```
(A(K(DA(U1) DA(U2) DB(U3 U4))
 L(DC(U5 U6)) L(DD(U7 U8)))
 B(M(DE(U9 U10) DF(U11 U12))
  N(DG(U13)) N(DG(U14)) N(DH(U15 U16))))
```

- not factored, equivalent to the initial list:
  (A(K(DA(U1))) A(K(DA(U2))) A(K(DB(U3))) A(K(DB(U4))) A(L(DC(U5)))
  A(L(DC(U6))) A(L(DD(U7))) A(L(DD(U8))) B(M(DE(U9))) B(M(DE(U10)))
  B(M(DF(U11))) B(M(DF(U12))) B(N(DG(U13))) B(N(DG(U14))) B(N(DH(U15)))
  B(N(DH(U16))))

In the above lists, '(' and ')' represent the beginning and the end of a list respectively. (Inner parentheses have precedence over outer parentheses.)

<u>End of Example</u>

The Destination operands are a required sequence of:

- Beginning of Destination operands field
- Sequence of pairs of RGN and REN List (see Figure 13-6 on page 13-16)
- End of Destination operands field

See Figure 13-1 on page 13-3 and Figure 13-5.

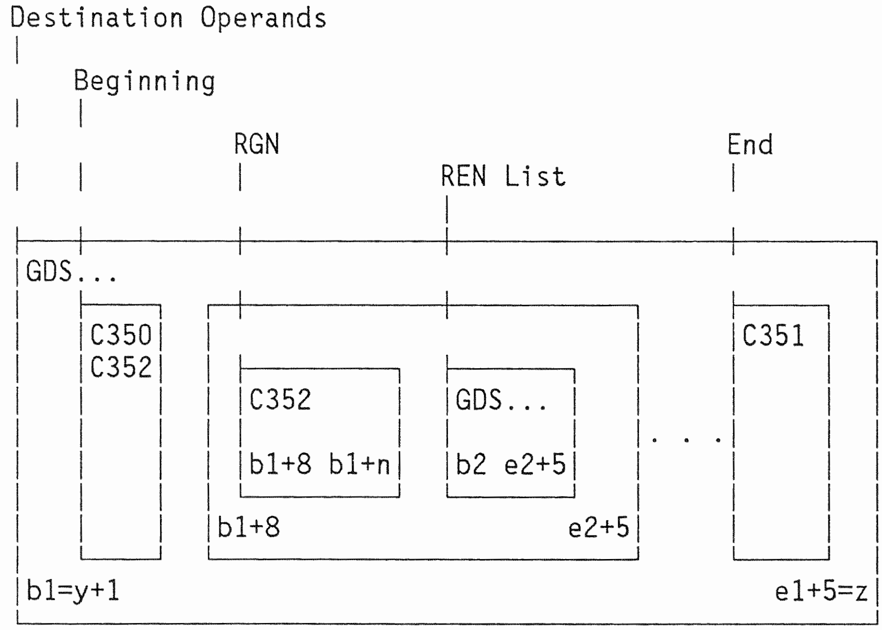


Figure 13-5. Structure of the Destination Operands

b1-b1+7 <u>Beginning of Destination Operands</u>

Required GDS variable
b1-b1+1 Length (8), in binary, of the GDS variable including this Length field
b1+2-b1+3 Beginning of Destination operands identifier: X'C350'
b1+4 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b1+5-b1+6 RGN identifier: X'C352'
b1+7 Format:
X'01' no segmentation descriptor follows

Note: RGN and REN List (the stream from offset b1+8 through e2+5) may be repeated.

b1+8-b1+n RGN

Required GDS variable, its content may be null
b1+8-b1+9 Length (n+1), in binary, of the GDS variable including this Length field (values from 5 to 13 are valid)
b1+10-b1+11 Identifier: X'C352'
b1+12 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b1+13-b1+n Name, 0- to 8-character length string, obtained from the origin DSU directory

b2-e2+5 REN List

Required sequence of:

- Beginning of REN List
- Sequence of pairs of REN and DGN list (see Figure 13-7 on page 13-17)
- End of REN List

See Figure 13-6 on page 13-16 and Figure 13-5 on page 13-14.

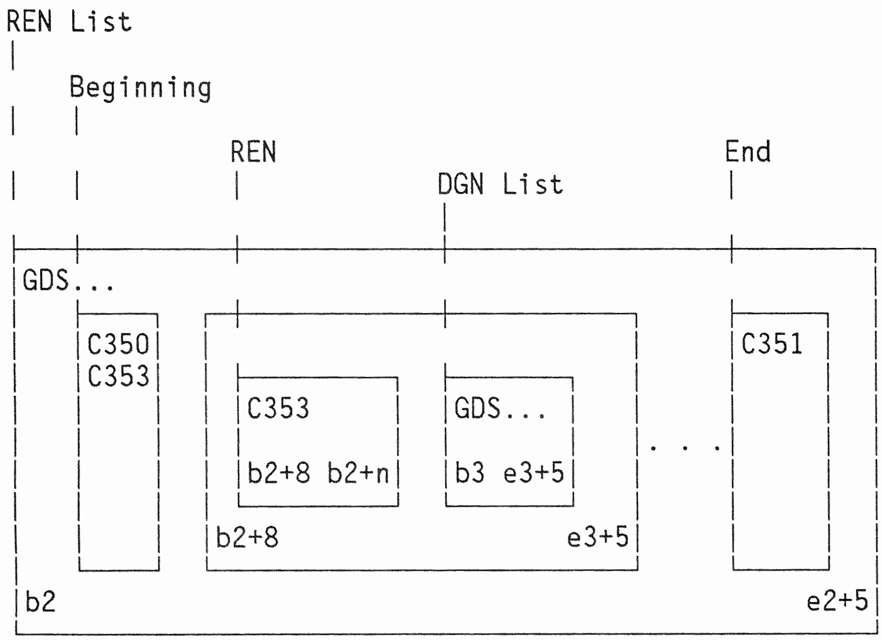


Figure 13–6. Structure of the REN List

b2-b2+7 <u>Beginning of REN List</u>

	Required GDS variable
b2-b2+1	Length (8), in binary, of the GDS variable, including the Length field
b2+2-b2+3	Beginning of a list identifier: X'C350'
b2+4	Format: X'01' no segmentation descriptor follows, does not contain LT-subfields
b2+5-b2+6	REN identifier: X'C353', with X'C350' represents the identifier of the Beginning of REN List
b2+7	Format: X'01' no segmentation descriptor follows

<u>Note:</u> REN and DGN List (the stream from offset b2+8 through e3+5) may be repeated.

b2+8-b2+n <u>REN</u>

	Required GDS variable
b2+8-b2+9	Length (n+1), in binary, of the GDS variable including this Length field (values from 6 to 13 are valid)
b2+10-b2+11	Identifier: X'C353'
b2+12	Format: X'01' no segmentation descriptor follows, does not contain LT-subfields
b2+13-b2+n	Name, 1- to 8-character string, obtained from the origin DSU directory

b2+n+1

<u>Note:</u> Offset b2+n+1 will be replaced by b3 in order to simplify the stream description

b3-e3+5 DGN List

Required sequence of:

- Beginning of DGN List
- Sequence of pairs of DGN and DEN List (see Figure 13-8 on page 13-18)
- End of DGN List

See Figure 13-6 on page 13-16 and Figure 13-7.

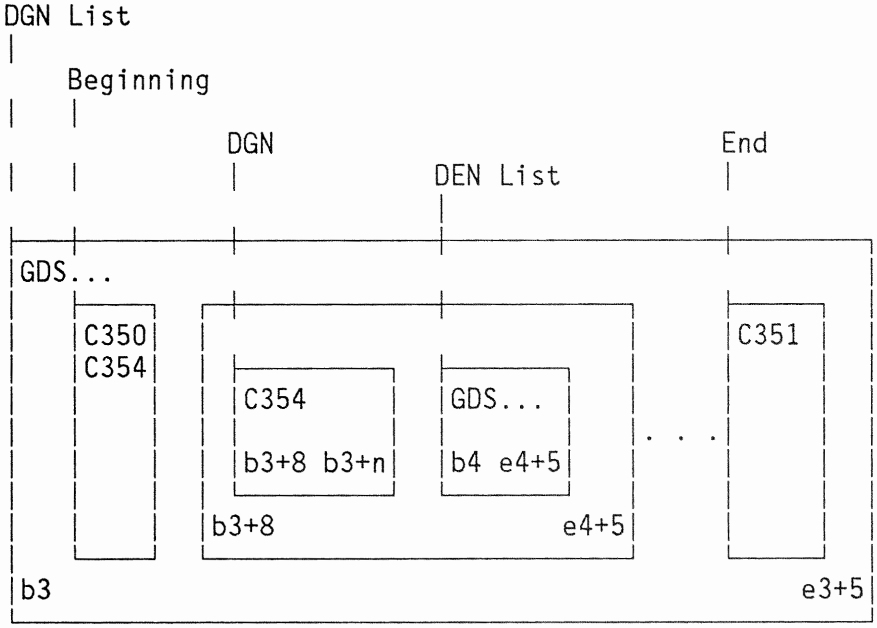


Figure 13-7. Structure of the DGN List

b3-b3+7 Beginning of DGN List

	Required GDS variable
b3-b3+1	Length (8), in binary, of the GDS variable including this Length field
b3+2-b3+3	List Beginning Identifier: X'C350'
b3+4	Format: X'01' no segmentation descriptor follows, does not contain LT-subfields
b3+5-b3+6	DGN identifier: X'C354', with X'C350' represents the identifier of the Beginning of the DGN List
b3+7	Format: X'01' no segmentation descriptor follows

Note: DGN and DEN list (the stream from offset b3+8 through e4+5) may be repeated

b3+8-b3+n DGN

Required GDS variable
b3+8-b3+9 Length, in binary, of the GDS variable including this Length field (values from 6 to 13 are valid)
b3+10-b3+11 Identifier: X'C354'
b3+12 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b3+13-b3+n Name, 1- to 8-character length string. It is provided either in the DESTINATION_DUN parameter of the DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb, if DISTIU was generated by the issue of the DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb, or in the Feedback DGN subfield (or the Origin DGN subfield if Feedback Address is missing) of the DISTIU for which the feedback was generated, if this DISTIU is a feedback.

b3+n+1

Note: Offset b3+n+1 will be replaced by b4 in order to simplify the stream description.

b4-e4+5 DEN List

Required sequence of:

- Beginning of DEN List
- Sequence of DENs
- End of DEN List

See Figure 13-7 on page 13-17 and Figure 13-8.

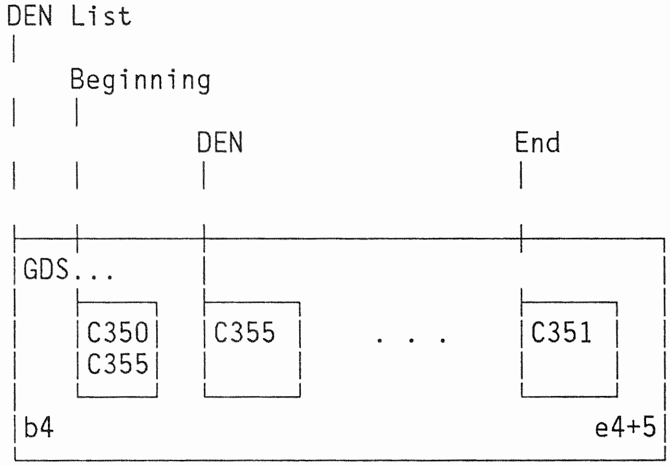


Figure 13-8. Structure of the DEN List

b4-b4+7 Beginning of DEN List

Required GDS variable
b4-b4+1 Length (8), in binary, of the GDS variable including this Length field
b4+2-b4+3 List Beginning code: X'C350'

b4+4 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b4+5-b4+6 DEN identifier: X'C355', with X'C350' represents the identifier of the beginning of the DEN List
b4+7 Format:
X'01' no segmentation descriptor follows

Note: DEN GDS variable (the stream from offset b4+8 through e4) may be repeated.

b4+8-b4+n DEN

Required GDS variable
b4+8-b4+9 Length, in binary, of the GDS variable including this Length field (values from 6 to 13 are valid)
b4+10-b4+11 Identifier: X'C355'
b4+12 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b4+13-b4+n Name, 1- to 8-character length string. It is provided either in the DESTINATION_DUN parameter of DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb, if DISTIU was generated by the issue of the DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb, or in the Feedback DEN subfield (or the Origin DEN subfield if Feedback Address is missing) of the DISTIU for which the feedback was generated, if this DISTIU is a feedback.

e4+1-e4+5 End of DEN List

Required GDS variable
e4+1-e4+2 Length (5), in binary, of the GDS variable including this Length field
e4+3-e4+4 Identifier: X'C351'
e4+5 Format:
X'01' no segmentation descriptor follows, it does not contain LT-subfields

e3+1-e3+5 End of DGN List

Required GDS variable
e3+1-e3+2 Length (5), in binary, of the GDS variable including this Length field
e3+3-e3+4 Identifier: X'C351'
e3+5 Format:
X'01' no segmentation descriptor follows, it does not contain LT-subfields

e2+1-e2+5 End of REN List

Required GDS variable
e2+1-e2+2 Length (5), in binary, of the GDS variable including this Length field
e2+3-e2+4 Identifier: X'C351'
e2+5 Format:
X'01' no segmentation descriptor follows, it does not contain LT-subfields

e1+1-e1+5 End of Destination Operands

Required GDS variable
e1+1-e1+2 Length (5), in binary, of the GDS variable including this Length field
e1+3-e1+4 Identifier: X'C351'

e1+5 Format:
X'01' no segmentation descriptor follows, it does not contain LT-subfields

z

Note: Offset z will replace e1+5 in order to simplify the stream description.

z+1-j Distribution Status Operands

Required and allowed if and only if DISTIU is of type STATUS (i.e., the value of bit 1 of Distribution Flags of Distribution General Options is X'1'.). It may include SNADS status, application status, or both, along with fields to identify DISTIU for which the status is being reported and transported to the originator or its designate, sequence of:

- Status operands
- General SNADS Status (optional)
- General Application Status (optional)
- Specific Status

See Figure 13-1 on page 13-3 and Figure 13-9 on page 13-21.

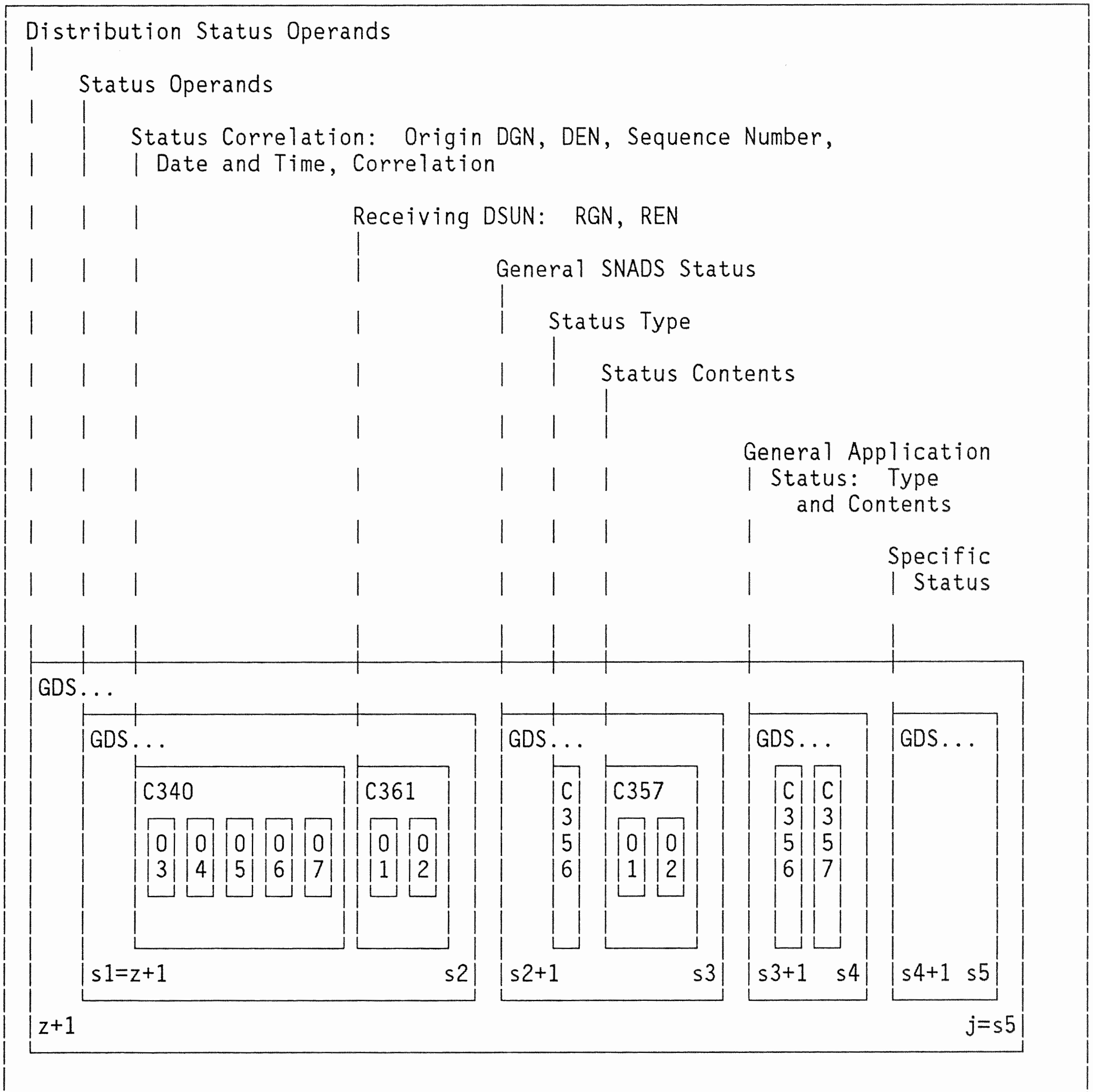


Figure 13-9. Distribution Status Operands Structure

z+1

<u>Note:</u> Offset z+1 will be replaced by s1 in order to simplify the stream description. Offsets s1, s2, s3, s4, and s5 are used to describe the Distribution Status operands stream.

s1-s2 <u>Status Operands</u>

Required set of:

- Status Correlation
- Receiving DSUN (optional)

The above components can be in any order. See Figure 13-1 on page 13-3 and Figure 13-9 on page 13-21.

0-n Status Correlation

Required GDS variable, contains LT-subfields with information to identify the DISTIU for which status information (provided either by SNADS if this DISTIU is generated by SNADS, or by the application transaction if DISTIU is generated through the issue of DISTRIBUTE_STATUS) is reported:

- Origin DGN
- Origin DEN
- Origin Sequence Number
- Origin Date and Time
- Origin Correlation (optional)

See Figure 13-9 on page 13-21. The LT-subfields can be in any order.

0-1 Length (n+1), in binary, of the GDS variable, including this Length field (values from 27 to 87 are valid)
2-3 Identifier: X'C340'
4 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

0-n1 Origin DGN

Required LT-subfield
0 Length (n1+1), in binary, of the LT-subfield including this Length field (values from 3 to 10 are valid)
1 Type: X'03'
2-n1 Name DGN, 1- to 8-character string, equal to the corresponding subfield in the Distribution Identifier of the data-DISTIU for which this status-DISTIU is returned

0-n2 Origin DEN

Required LT-subfield
0 Length (n2+1), in binary, of the LT-subfield including this Length field (values from 3 to 10 are valid)
1 Type: X'04'
2-n2 Name, 1- to 8-character string, equal to the corresponding subfield in the Distribution Identifier of the data-DISTIU for which this status-DISTIU is returned

0-5 Origin Sequence Number

Required LT-subfield
0 Length (6), in binary, of the LT-subfield including this Length field
1 Type: X'05'
2-5 Integer in base 10, 4-numeric EBCDIC encoded character string (values from 'F0F0F0F1' to 'F9F9F9F9'), equal to the corresponding subfield in the

Distribution Identifier of the data distribution for which this DISTIU is returned

0-9	Origin Date and Time
	Required LT-subfield, equal to the corresponding subfield in the Distribution Identifier of the data-DISTIU for which this DISTIU is returned (see Figure 13-1 on page 13-3)
0	Length (10), in binary, of the subfield including this Length field
1	Type: X'06'
2-5	Date:
2-3	Year, in binary (e.g., year 1983 is encoded as X'07BF')
4	Month of the year, in binary (values from 1 to 12 are valid)
5	Day of the month, in binary (values from 1 to 31 are valid)
6-9	Time:
6	Hour of the day, in binary (values from 0 to 23 are valid)
7	Minute of the hour, in binary (values from 0 to 59 are valid)
8	Second of the minute, in binary (values from 0 to 59 are valid)
9	Hundredth of the second, in binary (values from 0 to 99 are valid)
0-n3	Origin Correlation
	Optional LT-subfield, equal to the corresponding subfield in the Distribution identifier of the data-DISTIU for which this status-DISTIU is returned.
0	Length (n3+1), in binary, of this LT-subfield including this Length field (values from 3 to 46 defined)
1	Type: X'07'
2-n3	Origin Correlation, byte string, used by application transactions
0-n	Receiving DSUN
	Optional GDS variable if this status-DISTIU is returned because of receive time errors. It contains the name of the receiving DSU if the transmission error was detected by the receiving DSU. • Receiving RGN (optional) • Receiving REN See Figure 13-9 on page 13-21. The positions of its two LT-subfields are arbitrary.
0-1	Length (n+1), in binary, of the GDS variable, including this Length field (values from 8 to 25 are valid)
2-3	Identifier: X'C361'
4	Format: X'41' no segmentation descriptor follows, contains LT-subfields
0-n1	Receiving RGN
	Optional LT-subfield
0	Length (n1+1), in binary, of the subfield, including this Length field (values from 3 to 10 are valid)
1	Type: X='01'

2-n1	Name, 1- to 8-character string, first part of the name of the DSU that detected the receive-time error
0-n2	Receiving REN
	Required LT-subfield
0	Length (n2+1), in binary, of the subfield, including the Length field (values from 3 to 10 are valid)
1	Type: X='02'
2-n2	Name, 1- to 8-character string, second part of the name of the DSU that detected the receive-time error
s2+1-s4	General Status Note: New SNADS DSUs will not generate both General SNADS Status and General Application Status in a single IU. However, older SNADS DSUs may generate both SNADS and Application General Status fields in a single IU, and all SNADS DSUs must receive such IUs without error. DSUs may ignore the General Application Status field if General SNADS Status is present.
s2+1-s3	General SNADS Status
	Optional pair (elements in required order) of GDS variables, used if Specific SNADS status in Specific Status is missing, describes the SNADS status applicable to all DUNs in the Specific Status when no SNADS status is indicated along with them (i.e., Specific SNADS Status elements [described in the stream from offset s2+1 up to s3 below within Specific Status subfield] are missing): • Status Type • Status Contents See Figure 13-9 on page 13-21.
s2-s2+6	Status Type
	Required GDS variable
s2-s2+1	Length (7), in binary, of the GDS variable, including this Length field
s2+2-s2+3	Identifier: X'C356'
s2+4	Format: X'01' no segmentation descriptor follows, does not contain LT-subfields
s2+5-s2+6	Status Type: X'0001' for SNADS Status
s2+7-s3	Status Contents
	Required GDS variable; it contains the SNADS Condition Code
s2+7-s2+8	Length, in binary, of the GDS variable, including this Length field (value is 9)
s2+9-s2+10	Identifier: X'C357'
s2+11	Format: X'41' no segmentation descriptor follows, contains LT-subfields
s2+12-s2+15	SNADS Condition Code

Required LT-subfield
s2+12 Length (4), in binary, of the subfield, including this Length field
s2+13 Type: X'01'
s2+14-s2+15 Values are listed in Figure 13-10.

Code (hex)	Description	allowed in DISTIU	allowed in ACKIU
0001	Routing error	yes	yes
0002	Invalid DUN	yes	yes
0003	Hop Count exceeded	yes	yes
0004	Syntax error	yes	yes
0005	Function not supported	yes	yes
0006	Permanent server error	yes	yes
0007	Unknown server name	yes	yes
0008	Invalid server parameters	yes	yes
0009	Invalid Destination TP name	yes	yes
000C	Request purged	yes	no
000D	Lost DUNs	yes	no
000E	Resource not available	yes	yes
000F	System error	yes	yes
0010	Temporary server error	yes	yes
0011	I/O system error	yes	yes
0012	ACKIU error	yes	no
0013	Distribution Object Size incompatible with DSL	yes	yes
0000, 000A, 000B, and 0014 up to FFFF are reserved			

Figure 13-10. SNADS Status Condition Codes

s3+1-s4 General Application Status

Optional pair (elements in required order) of GDS variables, describes the application status applicable to all DUNs in the Specific Status (the interpretation of both general and specific application status information is defined by the application):

- Status Type
- Status Contents

s3+1-s3+7 Status Type

Required GDS variable
s3+1-s3+2 Length (7), in binary, of the GDS variable, including this Length field
s3+3-s3+4 Identifier: X'C356'
s3+5 Format:
X'01' no segmentation descriptor follows, without LT-subfields
s3+6-s3+7 Status Type:
not equal to X'0001'; see "Status Type Codes" on page 13-44

s3+7-s4 Status Contents

Required GDS variable

s3+7-s3+8 Length, in binary, of the GDS variable, including this Length field (values from 7 to 69 are valid)

Note: Older SNADS DSUs may generate IUs with Lengths of up to 517. All SNADS DSUs MUST receive such IUs without error. However, DSUs may modify such IUs to force the Length to be 69 or less.

s3+9-s3+10 Identifier: X'C357'

s3+11 Format:

X'41' no segmentation descriptor follows, contains LT-subfields

s3+12-s4 Status content, application defined bit string

s4+5

Note: The Specific Status has a list structure similar to DGN List of Destination operands; the offset s4+5 and j are replaced by b3 and e3+5 respectively. Offsets b3, e3+1, and b4, e4+1 point to the corresponding DGN and DEN lists beginnings and ends.

b3-e3+5 Specific Status

Required sequence of GDS variables (see Figure 13-9 on page 13-21 and Figure 13-11 on page 13-27), lists all the names of the recipient users of the distribution for whom the DISTIU reports the status, along with SNADS and/or application status information. It is structured as a DGN List:

- DGN List Beginning
- DGN
- DEN List (see Figure 13-12 on page 13-28)
- End List

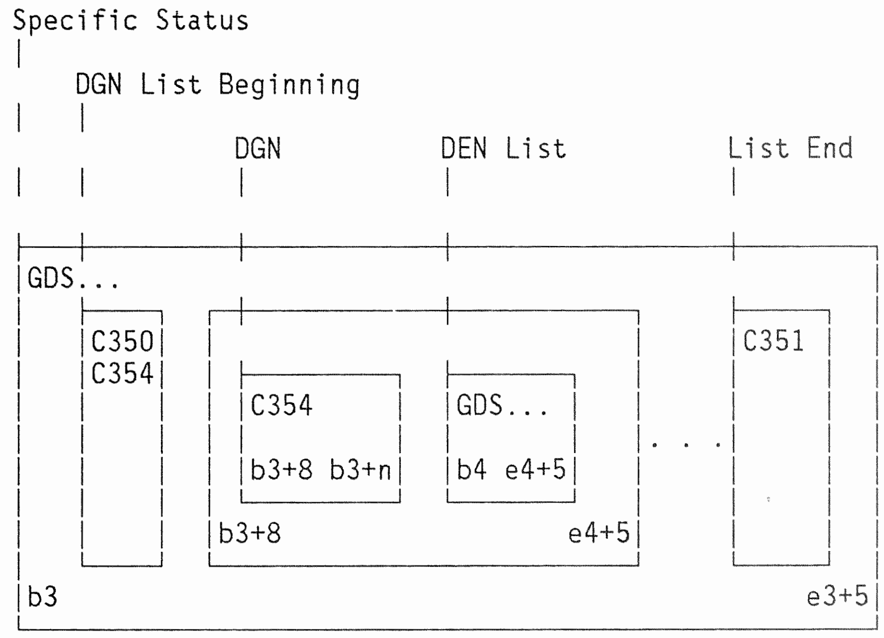


Figure 13-11. Structure of the DGN List of Specific Status

b3-b3+7 <u>DGN List Beginning</u>

Required GDS variable
b3-b3+1 Length (8), in binary, of the GDS variable including this Length field
b3+2-b3+3 List Beginning Identifier: X'C350'
b3+4 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b3+5-b3+6 DGN identifier: X'C354'
b3+7 Format:
X'01' no segmentation descriptor follows

<u>Note:</u> DGN and DEN list (the stream from offset b3+8 through e4+5) can be repeated.

b3+8-b3+n <u>DGN</u>

Required GDS variable
b3+8-b3+9 Length (n+1), in binary, of the GDS variable including this Length field (values from 5 to 13 are valid)
b3+10-b3+11 Identifier: X'C354'
b3+12 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b3+13-b3+n Name, 0- to 8-character length string

b3+n+1

<u>Note:</u> Offset b3+n+1 will be replaced by b4 in order to simplify the stream description.

b4-e4+5 DEN List

Required sequence of:

- DEN List Beginning
- DEN
- Specific SNADS Status (optional)
- Specific Application Status (optional)
- List End

See Figure 13-11 on page 13-27 and Figure 13-12.

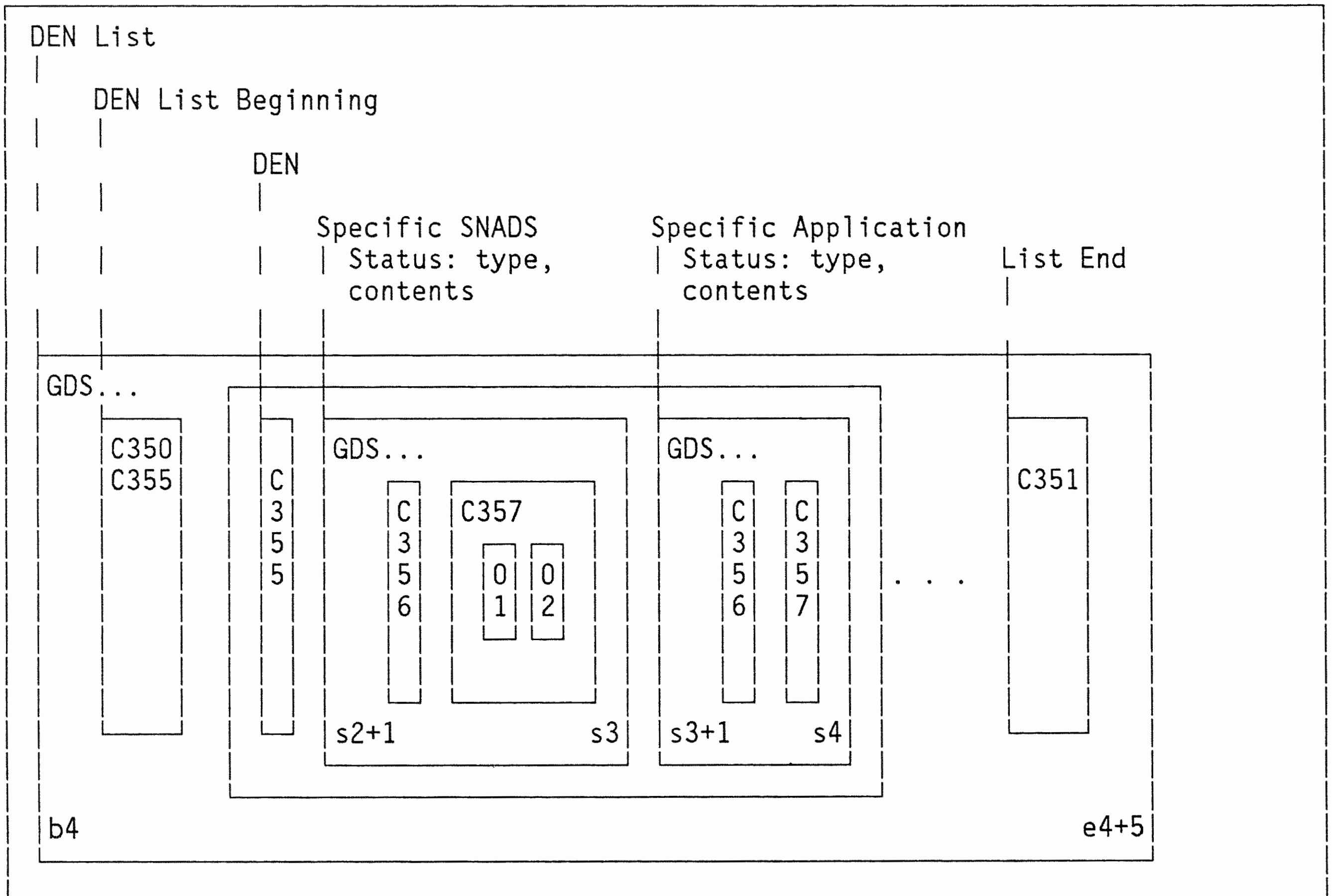


Figure 13–12. Structure of the DEN List of Specific Status

b4-b4+7 DEN List Beginning

	Required GDS variable
b4-b4+1	Length (8), in binary, of the GDS variable including this Length field
b4+2-b4+3	List Beginning code: X'C350'
b4+4	Format:
	X'01' no segmentation descriptor follows, does not contain LT-subfields
b4+5-b4+6	DEN identifier: X'C355'

b4+7 Format:
X'01' no segmentation descriptor follows

Note: DEN GDS variable (the stream from offset b4+8 through e4) may be repeated.

b4+8-b4+n DEN

Required GDS variable
b4+8-b4+9 Length (n+1), in binary, of the GDS variable including this Length field (values from 5 to 13 are valid)
b4+10-b4+11 Identifier: X'C355'
b4+12 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
b4+13-b4+n Name, 0- to 8-character length string
Note: DGN is null if and only if DEN is null. In this case, this null addressee (i.e., DGN and DEN null) is the only addressee permitted. DGN and DEN are null if either the General or Specific SNADS Status subfields are present with the SNADS Condition Code X'000D', or if either the General or Specific Application Status of type X'FEFF' are present.

b4+n
Note: Offset b4+n is replaced by s2.

s2+1-s4 Specific Status
Note: New SNADS DSUs will not generate both Specific SNADS Status and Specific Application Status for a single destination. However, older SNADS DSUs may generate both SNADS and Application Specific Status fields for a single destination, and all SNADS DSUs must receive such IUs without error. DSUs may ignore the Specific Application Status field if Specific SNADS Status is present.

s2+1-s3 Specific SNADS Status

Optional pair (elements in required order) of GDS variables, used if General SNADS Status information does not apply to this DUN:

- Status Type
- Status Contents

s2-s2+6 Status Type

Required GDS variable
s2-s2+1 Length (7), in binary, of the GDS variable, including this Length field
s2+2-s2+3 Identifier: X'C356'
s2+4 Format:
X'01' no segmentation descriptor follows, does not contain LT-subfields
s2+5-s2+6 Status Type:
X'0001' for SNADS Status

s2+7-s3 Status Contents

Required GDS variable; it contains the SNADS Condition Code

s2+7-s2+8 Length, in binary, of the GDS variable, including this Length field (value is 9)
s2+9-s2+10 Identifier: X'C357'
s2+11 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

s2+12-s2+15 SNADS Condition Code

Required LT-subfield
s2+12 Length (4), in binary, of the subfield, including this Length field
s2+13 Type: X'01'
s2+14-s2+15 Values are listed in Figure 13-10 on page 13-25.

s3+1-s4 Specific Application Status

Optional pair (elements in required order) of GDS variables, (the interpretation of both general and specific application status information is defined by the application):

- Status Type
- Status Contents

s3+1-s3+7 Status Type

Required GDS variable
s3+1-s3+2 Length (7), in binary, of the GDS variable, including this Length field
s3+3-s3+4 Identifier: X'C356'
s3+5 Format:
X'01' no segmentation descriptor follows, without LT-subfields
s3+6-s3+7 Status Type:
not equal to X'0001', see "Status Type Codes" on page 13-44

s3+7-s4 Status Contents

Required GDS variable
s3+7-s3+8 Length, in binary, of the GDS variable, including this Length field (values from 7 to 69 are valid)
Note: Older SNADS DSUs may generate IUs with Lengths of up to 517. All SNADS DSUs MUST receive such IUs without error. However, DSUs may modify such IUs to force the Length to be 69 or less.

s3+9-s3+10 Identifier: X'C357'
s3+11 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

s3+12-s4 Status content, application defined bit string

e4+1-e4+5 DEN List End

Required GDS variable
e4+1-e4+2 Length (5), in binary, of the GDS variable including this Length field
e4+3-e4+4 Identifier: X'C351'
e4+5 Format:
X'01' no segmentation descriptor follows, it does not contain LT-subfields

e3+1-e3+5	DGN List End
	Required GDS variable
e3+1-e3+2	Length (5), in binary, of the GDS variable including this Length field
e3+3-e3+4	Identifier: X'C351'
e3+5	Format: X'01' no segmentation descriptor follows, it does not contain LT-subfields
j	Note: Offset e3+5 is replaced by j.
j+1-k	Distribution Object
	Optional sequence of GDS variables (required order), containing: • Distribution Object Prefix • Data
	See Figure 13-1 on page 13-3.
j+1-j+n	Distribution Object Prefix
	Required GDS variable containing: • Object Size (optional) • Server Name • Server Parameters (optional)
j+1-j+2	Length (n), in binary, of the subfield, including this Length field (values from 8 to 336 are valid)
j+3-j+4	Identifier: X'C90A'
j+5	Format: X'41' no segmentation descriptor follows, contains LT-subfields
0-9	Object Size
	Optional LT-subfield
0	Length (10), in binary, of this LT-subfield including this Length field
1	Type: X'01'
2-9	Unsigned number, in binary, number of bytes of all the segments of the distribution object, need not be accurate
0-n1	Server Name
	Required LT-subfield
0	Length, in binary, of this LT-subfield, including the Length field (values from 3 to 66 are valid)
1	Type: X'02'
2-n1	Name, 1- to 64-character string (see "Specification of the Character-String Fields" on page 13-41 for the encoding of the graphic characters used), of the server to be used at the destination to store the Data GDS variable, supplied as a parameter of the DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb

0-n2	Server Parameters
	Optional LT-subfield
0	Length, in binary, of this LT-subfield, including the Length field (values from 3 to 255 are valid)
1	Type: X'03'
2-n2	Stream of bytes, the values of the parameters to be used by the destination server, supplied as a parameter of the DISTRIBUTE_DATA or DISTRIBUTE_STATUS verb
j+n+1	Replaced by b Note: The stream from offset b through e may be repeated.
b-e	Data
	Required sequence of one or more GDS variables
b-b+1	Length, in binary of this GDS variable including this Length field; values from 6 to 32511 are valid if there is no segmentation descriptor. Values from 9 to 32511 are valid if there is a segmentation descriptor and if this is not the last segment, or if this segment is the only segment. Values from 8 to 32511 are valid if there is a segmentation descriptor, and if this is the last and not the only segment.
b+2-b+3	Identifier: X'C908'
b+4	Format: X'01' no segmentation descriptor follows, and no LT-subfields included in the GDS variable X'81' segmentation descriptor follows, and no LT-subfields included in the GDS variable
b+5-b+7	Segmentation Descriptor
b+5	Position of this GDS variable in the Data: X'00' last or only segment (this is the last or only segment sent) X'20' not the last segment of the Data (at least one more segment follows contiguously on this conversation)
b+6-b+7	Segment sequence number: X'0000' segment sequence numbering not used
q	= (b+5)|(b+8) for b+4 containing X'01'|X'81', respectively
q-e	data stream
e	= k for b+4 and b+5 containing X'81' and X'00', respectively, or b+4 containing X'01'. e < k for b+4 and b+5 containing X'81' and X'20', respectively.
k+1-p	Suffix
	Required GDS variable, either Type 1 or Type 2
k+1-k+2	Length, in binary, of the GDS variable, including this Length field (value 5 is valid if Suffix is Type 1, value 8 valid if Suffix is Type 2)
k+3-k+4	Type identifier: X'CF01' Suffix Type 1: indicates that no exception condition has occurred sending the DISTIU X'CF02' Suffix Type 2: indicates that an exception condition was detected by the sender of the DISTIU (The Suffix Type 2 may follow after any subfield within DISTIU after the exception condition occurs.) Note: FMH-7 (see Chapter 10) is sent over the conversation before the Suffix Type 2 is sent. This is the result of issuing the SEND_ERROR verb

(see *SNA Transaction Programmer's Reference Manual for LU Type 6.2*) by the DS_SEND transaction program of the sending DSU.

k+5 Format:
X'00' no segmentation descriptor follows, for Suffix Type 1
X'01' no segmentation descriptor follows, for Suffix Type 2

k+6-k+8 *Exception Code*

Required subfield for a Type 2 suffix (otherwise, omitted), describing the type of error detected:

k+6 Exception Class:
bits 0-1, severity of error:
  11 catastrophic error, request not processed
bits 2-7, class of error:
  000101 sender error

k+7 Exception Condition Code, indicating the reason for the exception: values are defined in Figure 13-14 on page 13-37

k+8 Exception Object, indicating the syntactical entity in error: values are defined in Figure 13-15 on page 13-38

p = k+5 or k+8 as the Suffix is Type 1 or Type 2 respectively.

ACKIU; DSU ---> DSU; (ACKNOWLEDGE INTERCHANGE UNIT)

ACKIU flows between pairs of DSUs, to acknowledge or to notify the DSU that sent a DISTIU that the receiving DSU found errors while processing the DISTIU. It is not forwarded through the SNADS network. It contains:

- Prefix
- Command
- Suffix

See Figure 13-13 on page 13-35.

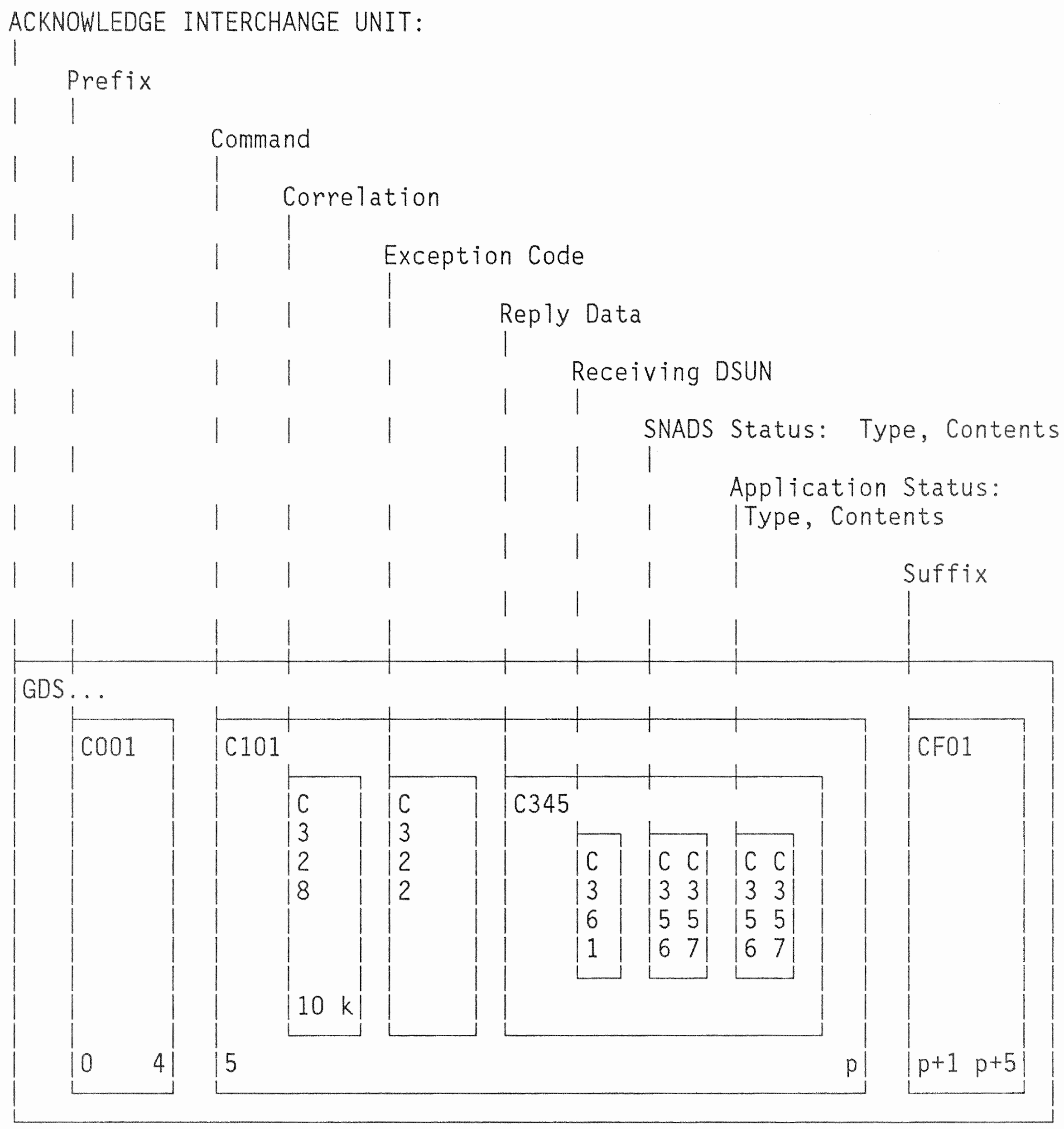


Figure 13-13. Structure of the Acknowledge IU

0-4 <u>Prefix</u>

Required GDS variable
0-1 Length (5), in binary, of the prefix, including this Length field
2-3 Identifier: X'C001'
4 Format:
X'02' no segmentation descriptor follows

5-p Command

Required GDS variable of:

- Correlation
- Exception Code
- Reply Data

See Figure 13-13 on page 13-35. Reply Data may appear after or before the Exception Code.

5-6 Length, in binary, of the GDS variable, including this Length field (values from 49 to 853 are valid)

7-8 Identifier: X'C101', identifies this IU as an ACKIU

9 Format:
X'01' no segmentation descriptor follows

10-k Correlation

Required GDS variable (see Figure 13-13 on page 13-35)

10-11 Length, in binary, of the GDS variable, including this Length field (values from 7 to 23 are valid)

12-13 Identifier: X'C328'

14 Format:
X'01' no segmentation descriptor follows

15 Reply indicator:
X'00' ACKIU is the last reply to the referenced DISTIU, see "Distribute Interchange Unit" on page 13-2

16 Command position, in binary, in the IU command sequence:
X'01' for SNADS

17-k Interchange unit identifier, optional field, a 0- to 16-byte identifier provided by the prefix of the DISTIU for which this ACKIU is a notification equal to the IU identifier (see "Distribute Interchange Unit" on page 13-2)

0-m Exception Code

Required GDS variable, contains error classification information and, optionally, the DISTIU subfield found in error:

- Exception Class
- Exception Condition Code
- Exception Object
- Exception Data (optional)

0-1 Length (m+1), in binary, of the GDS variable including this Length field (values from 8 to 255 are valid)

2-3 Identifier: X'C322'

4 Format:
X'01' no segmentation descriptor follows, contains no LT-subfields

5 Exception Class

Contains severity and error class:
bits 0-1, severity
'11' catastrophic error, request not processed

bits 2-7, class
'000011' semantic error
'000010' syntactic error
'000100' process error

6 Exception Condition Code, indicates reason for exception, values defined in Figure 13-14.

Code (hex)	Description	Allowed in DISTIU Suffix	Allowed in ACKIU
01	Function not supported		yes
02	Data not supported		yes
04	Resource not available		yes
06	Execution terminated	yes	yes
07	Data not found		yes
08	Segmentation		yes
0A	Sequence		yes
0B	I/O error	yes	yes
0C	ID invalid		yes
0E	Format invalid		yes
0F	Length invalid	yes	yes
10	Indicator invalid		yes
11	Range exceeded		yes
15	Subfield length invalid		yes
16	Subfield type invalid		yes
17	Invalid parameters		yes
18	Content error	yes	yes
All other values are reserved			

Figure 13-14. Exception Condition Codes

7 Exception Object, indicates the syntactical entity in error. Values are defined in Figure 13-15 on page 13-38.

Code (hex)	Exception Object	allowed in DISTIU Suffix	allowed in ACKIU
01	IU prefix	yes	yes
02	IU identifier		yes
07	Command	yes	yes
08	Command operand		yes
09	Operand value		yes
0C	Document unit	yes	yes
0D	Document unit identifier		yes
0E	Document profile		yes
0F	Document profile parameter		yes
10	Document content introducer		yes
11	Document content control		yes
12	Document content data		yes
13	IU suffix	yes	yes
14	Segment		yes
16	Unsupported subfield		yes
17	Unknown subfield	yes	yes
1A	Distribution object prefix	yes	yes
1B	Distribution object data	yes	yes
All other values are reserved.			

Figure 13-15. Exception Object Codes

8-m Exception Data, optional, contains the DISTIU subfield in error

0-p Reply Data

Required GDS variable, information to be used by the DSU to create the feedback:

- Receiving DSUN
- SNADS Status
- Application Status (optional)

See Figure 13-13 on page 13-35.
Note: Unless otherwise specified, the receiving DSU is the DSU that receives the DISTIU; likewise, the sending DSU is the DSU that sends the DISTIU.

0-1 Length (p+1), in binary, of the GDS variable including this Length field (values from 29 to 570 are valid)

2-3 Identifier: X'C345'

4 Format:
X'01' no segmentation descriptor follows, contains no LT-subfields

5-n2 Receiving DSUN

	Required GDS variable, the name of the receiving the DSU that detected the error, and the name of the sender of the ACKIU, contains:
	• Receiving RGN (optional) • Receiving REN
	The above LT-subfields can be in any order.
5-6	Length, in binary, of the GDS variable including this Length field (values from 8 to 25 are valid)
7-8	Identifier: X'C361'
9	Format: X'41' no segmentation descriptor follows, contains LT-subfields
0-n1	Receiving RGN
	Optional LT-subfield
0	Length (n1+1), in binary, of the subfield including this Length field (values from 3 to 10 are valid)
1	Type: X'01'
2-n1	Name, 1- to 8-character length string, first part of the name of the detecting DSU
0-n2	Receiving REN
	Required LT-subfield
0	Length (n2+1), in binary, of the subfield including the Length field (values from 3 to 10 are valid)
1	Type: X'02'
2-n2	Name, 1- to 8-character string, second part of the name of the detecting DSU
n2+1-n3	SNADS Status
	Required pair of GDS variables:
	• Status Type • Status Contents
	See Figure 13-13 on page 13-35:
n2+1-n2+7	Status Type
	Required GDS variable
n2+1-n2+2	Length (7), in binary, of the GDS variable including this Length field
n2+3-n2+4	Identifier: X'C356'
n2+5	Format: X'01' no segmentation descriptor follows, does not contain LT-subfields
n2+6-n2+7	Status Type: X'0001' for SNADS Status
n2+8-n2+16	Status Contents
	Required GDS variable
n2+8-n2+9	Length (9), in binary, of the GDS variable including this Length field

n2+10-n2+11 Identifier: X'C357'
n2+12 Format:
X'41' no segmentation descriptor follows, contains LT-subfields

n2+13-n2+16 SNADS Condition Code

Required LT-subfield
n2+13 Length (4), in binary, of the subfield including this Length field
n2+14 Type: X'01'
n2+15-n2+16 Values are listed in Figure 13-10 on page 13-25.

n2+17-p Application Status

Optional pair of GDS variables:

- Status Type
- Status Contents

Note: When generating a DIST_IU with Status information supplied by an ACKIU, the Reporting DSU may ignore Application Status.

n2+17-n2+18 Status Type

Required GDS variable
n2+17-n2+18 Length (7), in binary, of the GDS variable including this Length field
n2+19-n2+20 Identifier: X'C356'
n2+21 Format:
X'01' no segmentation descriptor follows, without LT-subfields
n2+22-n2+23 Status Type:
X'0200'-X'06FF' are valid for applications

n2+24-p Status Contents

Required GDS variable
n2+24-n2+25 Length, in binary, of the GDS variable including this Length field
(values from 7 up to 517 are valid)
n2+26-n2+27 Identifier: X'C357'
n2+28 Format:
X'41' no segmentation descriptor follows, contains LT-subfields
n2+29-p Status content, application defined bit string

p+1-p+5 Suffix

Required GDS variable
p+1-p+2 Length (5), of the GDS variable, including this field
p+3-p+4 Identifier: X'CF01'
p+5 Format:
X'00' Only valid value

SPECIFICATION OF THE CHARACTER-STRING FIELDS

Figure 13-16 defines the character sets, string rules, and string lengths used to encode the SNADS interchange units. For more details, see the syntactic and the semantic descriptions.

The character strings are specified for two cases: the base and the enhanced character string (ECS) option subset.

Field Name	Character Set		String Rules		String Length	
	Base	ECS Opt	Base	ECS Opt	Min	Max
DGN	A	930	none	(Note 1)	1 (Note 1)	8
DEN	A	930	none	(Note 1)	1 (Note 1)	8
Origin RGN	A	A	none	none	1	8
Destination RGN	A	930	none	(Note 1)	1	8
Origin REN	A	A	none	none	1	8
Destination REN	A	930	none	(Note 1)	1	8
Destination TP Name, Feedback TP Name, and Server Name	AE (Note 2)		none		1	64
Origin Sequence Number	Numerics of A: (F0, F1, .. F9)		n/a		4	4

Notes:

1. - DGN and DEN minimum length is 0 if the DISTIU is of type STATUS and is generated by SNADS.
   - Leading space (X'40') characters are not used, trailing space (X'40') characters are not significant, and imbedded space (X'40') characters are significant.
2. The first character of an SNA service transaction program name is a byte ranging in value from X'00' through X'3F'.

Figure 13-16. Character-String Specification

Note: Appendix A defines the codes of the graphic character sets A, AE, and 930.

CODE POINTS USED BY SNADS

The values of the ID component of the LLIDF field as used for SNADS GDS variables are shown below:[3]

ID	Subfield Name
C001*	In DIA, IU PREFIX; in SNADS, Prefix for Distribute IU and Acknowledge IU
C101*	in DIA, IU CMD NO REPLY ACKNOWLEDGE; in SNADS, Command of the Acknowledge IU
C105	Command, Distribute IU
C322*	in DIA, IU OPERAND IMM DATA EXCEPTION-CODE; in SNADS, Exception Code, within Acknowledge IU
C328*	in DIA, IU OPERAND IMM DATA DATA CORRELATION; in SNADS, Correlation, within Acknowledge IU
C32D*	in DIA, IU OPERAND IMM DATA USER-DATA; in SNADS, Destination Application Parameters, within Distribute IU
C33D*	in DIA, IU OPERAND IMM DATA STATUS-INFORMATION; in SNADS, Distribution General Options, within Distribute IU
C340*	in DIA, IU OPERAND IMM DATA DISTRIBUTION-IDENTIFIER; in SNADS, Distribution Identifier, within Distribute IU
C343*	in DIA, IU OPERAND IMM DATA GENERAL-ROUTING-DATA; in SNADS, Feedback Options, within Distribute IU
C345*	in DIA, IU OPERAND IMM DATA REPLY DATA; in SNADS, Reply Data, within Acknowledge IU
C350	Beginning of Destination Operand Lists, of the Specific Status Lists, within Distribute IU
C351	End of Destination Operands Lists, of the Specific Status Lists, within Distribute IU
C352	Routing Group Name (RGN) of Destination Operands, within Distribute IU
C353	Routing Element Name (REN) of REN List, within Distribute IU
C354	Distribution Group Name (DGN) of DGN List, within Distribute IU
C355	Distribution Element Name (DEN) of DEN List, within Distribute IU
C356	Status Type, within Distribute IU

---

[3] The asterisk following the ID indicates that that identifier is used by both DIA (Document Interchange Architecture) and SNADS.

C357	Status Contents, within Distribute IU
C360	Feedback Address, within Distribute IU
C361	Receiving Distribution Service Unit Name, within Distribute IU
C908	Data, Distribution Object, within Distribute IU
C90A	Distribution Object Prefix, Distribute IU
CF01*	in DIA, IU SUFFIX NORMAL-TERMINATION; in SNADS, Suffix Type 1
CF02*	in DIA, IU SUFFIX ABNORMAL-TERMINATION; in SNADS, Suffix Type 2

STATUS TYPE CODES

SNADS uses the following status type code as a subfield of the General and Specific Status subfields of the DISTIUs:

Code	Meaning
X'0000'	Reserved
X'0001'	SNADS errors
X'0002'-X'01FF'	Reserved
X'0200'	DIA application errors
X'0201'-X'FFFF'	Reserved

TRANSACTION PROGRAM AND SERVER NAMES

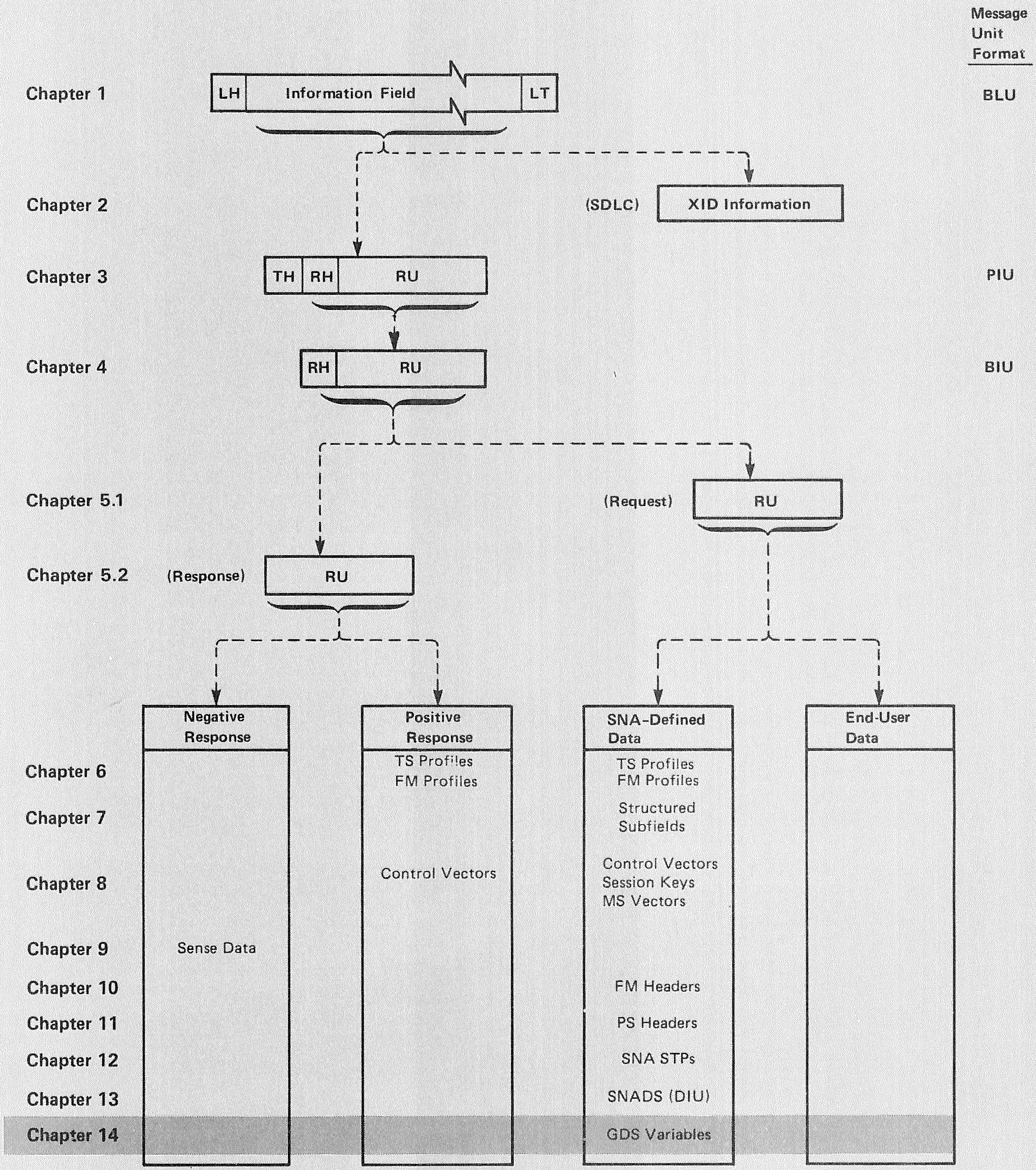
Following is a list of all transaction program and server names defined for SNADS, in the FM header 5 (Attach), in the Distribute IU, or used internally in the distribution service unit (DSU).

Code	Meaning
X'20F0F0F0'	DIA PROCESS Destination Transaction Program Name
X'20F0F0F1'	DIA SERVER Name
X'21F0F0F1'	DS_SEND Transaction Program Name
X'21F0F0F2'	DS_RECEIVE Transaction Program Name
X'21F0F0F3'	DS_ROUTER_DIRECTOR Transaction Program Name
X'21F0F0F6'	SNADS General Server Name

Chapter 14

General Data Stream (GDS) Variables

For General Use



LH = Link Header
LT = Link Trailer
XID = Exchange Identification
TH = Transmission Header
RH = Request/Response Header
RU = Request/Response Unit
TS = Transmission Services
FM = Function Management
MS = Management Services

PS = Presentation Services
STP = Service Transaction Program
SNADS = SNA Distribution Services
DIU = Distribution Interchange Unit
GDS = General Data Stream
BLU = Basic Link Unit
PIU = Path Information Unit
BIU = Basic Information Unit

CHAPTER 14. GDS VARIABLES FOR GENERAL USE

The following chart indicates (using an "X") each GDS variable code point (with first byte = X'12') used by LU 6.2.

First hexadecimal digit
Second hexadecimal digit

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1	X	X		X												
2																
3																
4																
5																
6																
7																
8																
9																
A	X															
B																
C																
D																
E		X	X													
F		X	X	X	X	X										X

Figure 14-1. LU type 6.2 GDS Variable Code Points

The code points used by LU 6.2 are:

X'1210'	Change Number of Sessions (see note 1)
X'1211'	Exchange Log Name (see note 1)
X'1213'	Compare States (see note 1)
X'12A0'	Workstation Display Passthrough
X'12E1'	Error Log
X'12E2'	PIP Subfield Data (see note 2)
X'12F1'	Null Data
X'12F2'	User Control Data
X'12F3'	Map Name
X'12F4'	Error Data
X'12F5'	PIP Data (see note 2)
X'12FF'	Application Data

Notes:

1. See "Chapter 12. GDS Variables for SNA Service Transaction Programs" for the formats and meanings of these GDS variables.

2. See "Chapter 10. Function Management Headers" for the formats and meanings of these GDS variables.

FORMAT OF APPLICATION DATA GDS VARIABLE

The Application Data GDS variable, ID X'12FF', contains application data. The application transaction program's data as specified in the MC_SEND_DATA verb is (optionally) mapped and then sent as X'12FF' variables.

FORMAT OF NULL DATA VARIABLE

The Null Data GDS variable, ID X'12F1', contains no application data. This variable may optionally be generated to carry certain control information (e.g., Confirm) when no application data is available.

FORMAT OF USER CONTROL DATA GDS VARIABLE

The User Control Data GDS variable, ID X'12F2', contains user control data. The meaning of this data is known only to the LU Services Component Programs or the transaction programs and their mapping programs. This data can be used, for example, as prefix control information for an Application Data GDS variable that follows it or to carry FM Header Data for a mapped conversation transaction.

FORMAT OF MAP NAME GDS VARIABLE

The Map Name GDS variable, ID X'12F3', is followed by a 0- to 64-byte map name.

Format of an Error Data GDS variable

The Error Data GDS variable, ID X'12F4', is used to convey information about mapping errors. It is sent using the SEND_DATA verb following a SEND_ERROR verb. Its format is:

0-1	Length (n+1), in binary, of Error Data GDS variable, including this Length field
2-3	GDS ID: X'12F4'
4-7	Error code: X'00010000' Invalid GDS ID: The mapped conversation verb component encountered a GDS ID that it did not recognize. X'00030001' Map Not Found: The specified map was not available at the target, or access to the referenced map could not be completed. X'00030002' Map Execution Failure: The map program was not able to process the data stream.
8	Length (n-8), in binary, of error parameter
9-n	Error parameter: for a mapping failure, the map name carried in the GDS variable for which the error occurred; for an invalid GDS ID, the 2-byte GDS ID that was not recognized

Format of Error Log GDS Variable

The Error Log GDS variable, ID X'12E1', following an FMH-7 conveys implementation-specific error information to an LU, where it is added to the system error log for use in debugging and error recovery. It is not used by SNA-defined service transaction programs (other than to log it) since it contains implementation-specific data. The Error Log variable is sent as a consequence of issuing the SEND_ERROR verb, but is not passed to the receiving transaction program. Its format is:

0-1	Length (n+1), in binary, of Error Log GDS variable, including this Length field
2-3	GDS ID: X'12E1'
4-m	*Product Set ID*
4-5	Length, in binary, of Product Set ID, including this Length field (values 2 to 32,767 are valid) *Note:* The Length field is always present; a value of 2 indicates no Product Set ID subvector follows.
6-m	Product Set ID (X'10') subvector (format described in "Chapter 8. Common Fields")
m+1-n	*Message Text*
m+1-m+2	Length, in binary, of message text, including this Length field (values 2 to 32,767 are valid) *Note:* The Length field is always present; a value of 2 indicates no message text follows.
m+3-n	Message text data: implementation-specific data

APPENDIX A. SNA CHARACTER SETS AND SYMBOL-STRING TYPES

This appendix describes the character sets and symbol-string types used for various fields, such as:

- LU name
- Network-qualified LU name
- Mode name
- Transaction program name
- Access security information subfields
- Program initialization parameters (PIP) subfields
- Map name
- SNADS server, user (DGN, DEN), and service unit (RGN, REN) names

The detailed syntax of these and other such strings is described in other chapters where their usage within individual message units is defined.

SYMBOL-STRING TYPE

The symbol-string type specifies the set of code points and corresponding characters from which the strings listed above are composed, as follows:

- Type A (Assembler oriented): a character string consisting of one or more characters from character set A. The first character of a type A symbol string is not a numeric; i.e., it is different from X'F0', X'F1', ..., or X'F9'.
- Type AE (A extended): a character string consisting of one or more characters from character set AE, with no restriction on the first character.
- Type 930 (distribution services oriented): a character string consisting of one or more characters from character set 930, with the following rules:
  - No leading space (X'40') characters are used, but no other restrictions exist on the first character.
  - Imbedded space (X'40') characters are significant.
  - Trailing space (X'40') characters are not significant.

- Type USS (unformatted system services oriented, used for character-coded requests): a character string consisting of one or more characters from character set USS, with no restriction on the first character.
- Type GR (EBCDIC graphics): a byte string consisting of one or more bytes of value within the range X'41' through X'FE', with no restriction on the first character.
- Type DB (double byte): a byte string consisting of an even number of four or more bytes beginning with a byte of X'0E', followed by bytes in the range X'41' through X'FE', and ending with a byte of X'0F'.
- Symbol-string type G (general): a byte string consisting of one or more bytes of value within the range X'00' through X'FF', with no restriction on the first byte.

SNA CHARACTER SETS

Figure A-1 defines the character sets A, AE, 930, and USS. The code points that do not belong to any of these sets are not shown.

Hex Code	Graphic	Description	Set			
			A	AE	930	USS
15		Line Feed				X
40		Space			X	X
4B	.	Period		X	X	X
4D	(	Left Parenthesis				X
4E	+	Plus Sign				X
50	&	Ampersand			X	X
59	ß	Sharp S			X	
5B	$	Dollar Sign	X	X	X	X
5C	*	Asterisk				X
5D	)	Right Parenthesis				X
60	–	Minus Sign			X	X
61	/	Slash			X	X
62	Â	A Circumflex, Capital			X	
63	Ä	A Dieresis, Capital			X	
64	À	A Grave, Capital			X	
65	Á	A Acute, Capital			X	
66	Ã	A Tilde, Capital			X	
67	Å	A Overcircle, Capital			X	
68	Ç	C Cedilla, Capital			X	
69	Ñ	N Tilde, Capital			X	
6B	,	Comma			X	X
71	É	E Acute, Capital			X	
72	Ê	E Circumflex, Capital			X	
73	Ë	E Dieresis, Capital			X	
74	È	E Grave, Capital			X	
75	Í	I Acute, Capital			X	
76	Î	I Circumflex, Capital			X	
77	Ï	I Dieresis, Capital			X	
78	Ì	I Grave, Capital			X	
7B	#	Number Sign	X	X	X	X
7C	@	At Sign	X	X	X	X
7D		Apostrophe			X	X
7E	=	Equal Sign				X
80	Ø	O Slash, Capital			X	
81	a	a, Small		X		
82	b	b, Small		X		
83	c	c, Small		X		
84	d	d, Small		X		
85	e	e, Small		X		
86	f	f, Small		X		
87	g	g, Small		X		
88	h	h, Small		X		
89	i	i, Small		X		
91	j	j, Small		X		
92	k	k, Small		X		
93	l	l, Small		X		
94	m	m, Small		X		
95	n	n, Small		X		
96	o	o, Small		X		
97	p	p, Small		X		
98	q	q, Small		X		
99	r	r, Small		X		
9A	a̲	a Underscore, Small			X	
9B	o̲	o Underscore, Small			X	
[illegible]	¸	Cedilla			X	
9E	Æ	AE Diphthong, Capital			X	
A0	μ	Micro, Mu			X	
A2	s	s, Small		X		
A3	t	t, Small		X		
A4	u	u, Small		X		
A5	v	v, Small		X		
A6	w	w, Small		X		
A7	x	x, Small		X		
A8	y	y, Small		X		
A9	z	z, Small		X		
AC	Đ	D Stroke, Capital			X	
AD	Ý	Y Acute, Capital			X	
AE	Þ	Thorn, Capital			X	
C1	A	A, Capital	X	X	X	X
C2	B	B, Capital	X	X	X	X
C3	C	C, Capital	X	X	X	X
C4	D	D, Capital	X	X	X	X
C5	E	E, Capital	X	X	X	X
C6	F	F, Capital	X	X	X	X
C7	G	G, Capital	X	X	X	X
C8	H	H, Capital	X	X	X	X
C9	I	I, Capital	X	X	X	X
D1	J	J, Capital	X	X	X	X
D2	K	K, Capital	X	X	X	X
D3	L	L, Capital	X	X	X	X
D4	M	M, Capital	X	X	X	X
D5	N	N, Capital	X	X	X	X
D6	O	O, Capital	X	X	X	X
D7	P	P, Capital	X	X	X	X
D8	Q	Q, Capital	X	X	X	X
D9	R	R, Capital	X	X	X	X
DF	ÿ	y, Dieresis, Small			X	
E2	S	S, Capital	X	X	X	X
E3	T	T, Capital	X	X	X	X
E4	U	U, Capital	X	X	X	X
E5	V	V, Capital	X	X	X	X
E6	W	W, Capital	X	X	X	X
E7	X	X, Capital	X	X	X	X
E8	Y	Y, Capital	X	X	X	X
E9	Z	Z, Capital	X	X	X	X
EB	Ô	O Circumflex, Capital			X	
EC	Ö	O Dieresis, Capital			X	
ED	Ò	O Grave, Capital			X	
EE	Ó	O Acute, Capital			X	
EF	Õ	O Tilde, Capital			X	
F0	0	Zero	X	X	X	X
F1	1	One	X	X	X	X
F2	2	Two	X	X	X	X
F3	3	Three	X	X	X	X
F4	4	Four	X	X	X	X
F5	5	Five	X	X	X	X
F6	6	Six	X	X	X	X
F7	7	Seven	X	X	X	X
F8	8	Eight	X	X	X	X
F9	9	Nine	X	X	X	X
FB	Û	U Circumflex, Capital			X	
FC	Ü	U Dieresis, Capital			X	
FD	Ù	U Grave, Capital			X	
FE	Ú	U Acute, Capital			X	

Figure A-1. Character Sets A, AE, 930, and USS.

APPENDIX B. GDS ID DESCRIPTION AND ASSIGNMENTS

This appendix defines the general data stream (GDS), which is used in a variety of ways in SNA. For instance, it is used to encode the Document Interchange Architecture (DIA) message units. The basic structural unit in GDS is the structured field, a string of bytes preceded by a length and beginning with a GDS identifier (ID) that defines the structure of the remainder of the field. Some structured fields are used by components of SNA that are defined in this book; these uses are defined in "Chapter 12. GDS Variables for SNA Service Transaction Programs", "Chapter 14. GDS Variables for General Use" , "Chapter 5.1. Request Units", "Chapter 5.2. Response Units", and "Chapter 10. Function Management Headers". GDS IDs are assigned, generally in blocks of consecutive values, to different layers and components of SNA and to other interconnection architectures. For a complete listing of these block assignments, see below.

The general data stream applies to data exchanged between nodes over SNA links, over non-SNA links, and to data exchanged via removable storage media or shared storage facilities.

STRUCTURED FIELDS

Each structured field has the format shown in Figure B-1.

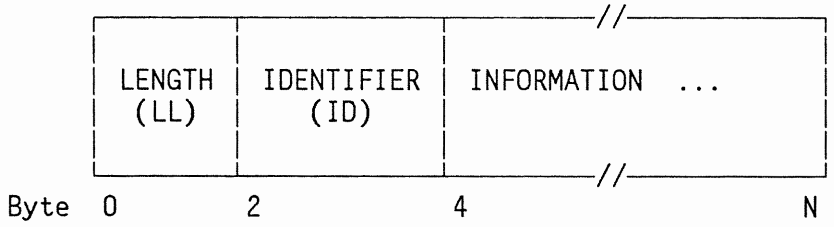


Figure B-1. GDS Structured Field

LENGTH (LL) DESCRIPTION

The LLID is a 4-byte field in which the two LL bytes are used to indicate the length of the LLID field itself (4 bytes) plus the data following the LLID; up to 32,763 bytes of data may follow the LLID. Values 0 and 1 of the LL are reserved for use as escape sequences; values 2 and 3 are not used. For example, a value of X'0001' indicates a presentation services header, which is used for sync point management.

Bit 0 (high-order bit) of byte 0 is used as a length continuation (or not-last segment) indicator. If that bit is set to 1, the logical record is continued by a 2-byte LL; the ID occurs only following the first LL. The continuing LL is located immediately following the information bytes encompassed by the first LL. The continuing LL might itself be continued. In other words, the length specified by the continuing LL might not be the entire remainder of the logical record; it might be followed by yet another LL. The amount of data spanned by each continuing LL can be any size convenient to the sender (including 0). Eventually, the chain of continuing LLs is ended by a final LL, i.e., one with the high-order (not-last) bit set to 0. The final LL may indicate a null information field follows (length = 2).

When an LLID encompasses a string of logical records identified by full LLIDs, the length of the string, determined by summing the (nested) encompassed LLs, equals the length definer of the (outer) encompassing LLID less 4 (this applies at each level of nesting). If the encompassing LLID is continued by segmenting, the length of the string of segments equals the sum of the initial LL and all continuing LLs of the encompassing ID less 4 for the initial LLID and 2 for each continuing LL.

The 2-byte ID values, irrespective of the level of nesting at which they occur, are defined uniquely across all levels of nesting, with the following exception. The ID values X'FF00' through X'FFFF' are used only within an encompassing LLID (which is not necessarily the immediate parent structure); their meaning is defined by the architecture that owns the higher-level ID and it applies only within the context of that ID. In other words, ID values in the X'FF**' range are context dependent. All other ID values are context independent.

IDENTIFIER (ID) DESCRIPTION

The 2-byte identifier that follows the length field indicates the format and meaning of the data that follows. Sometimes additional values appearing in the information field are needed to completely specify the information field's content. The uniqueness of the identifier (with the exceptions noted above) makes it easy to decode structured fields in line traces, and also to make it easier to create composite data streams by including elements of several architectures. DIA carried by SNADS is an example of such a use.

IDENTIFIER REGISTRY

The identifiers that have been assigned for specific use are listed below. Identifiers are assigned in blocks; not all identifiers in a block are necessarily currently used by the owner.

GDS ID	Structured Field Owner
00**	3270
01**	3270

03**	3270
06**	3270
09**	3270
0B**	3270
0C**	3270
0D**	3270
0E**	3270
0F00-0FFF	3270
1010-101F	3270
1030-1034	Print Job Restart
1100-1104	SNA Character String
1200-12FF	LU 6.2 (APPC)
1300-13FF	Management Services
1400-140F	3820 Page Printer
1570-157F	SNA Distribution Services
40**	3270
41**	3270
4A**	3270
4B**	3270
4C**	3270
7100-71FF	3250
80**	3270
8100-81FF	3270
C000-C00F	Document Interchange Architecture
C100-C104	Document Interchange Architecture
C105	SNA Distribution Services
C10A-C122	Document Interchange Architecture
C123-C124	SNA Distribution Services
C219	Document Interchange Architecture
C300-C345	Document Interchange Architecture
C350-C359	SNA Distribution Services
C366-C3FF	Document Interchange Architecture

C400-C46F	Document Interchange Architecture
C500-C56F	Document Interchange Architecture
C600-C66F	Document Interchange Architecture
C700-C7FF	Graphical Display Data Manager
C800-C87F	Document Interchange Architecture
C900-C97F	Document Interchange Architecture
C980-C9FF	Document Interchange Architecture
CA00-CA7F	Document Interchange Architecture
CA80-CAFF	Document Interchange Architecture
CB00-CB0F	Document Interchange Architecture
CC00-CC3F	Document Interchange Architecture
CD00-CD3F	Document Interchange Architecture
CF00-CF0F	Document Interchange Architecture
D000-D0FF	Distributed Data Management
D300-D3FF	Document Content Architecture
D600-D6FF	Intelligent Printer Data Stream
D780-D7BF	Facsimile Architecture
E100-E10F	Level-3 Document Content Architecture
E200-E20F	Level-3 Document Content Architecture
E300-E30F	Level-3 Document Content Architecture
E400-E40F	Level-3 Document Content Architecture
E500-E50F	Level-3 Document Content Architecture
E600-E60F	Level-3 Document Content Architecture
E700-E70F	Level-3 Document Content Architecture
E800-E80F	Level-3 Document Content Architecture
E900-E90F	Level-3 Document Content Architecture
EA00-EA0F	Level-3 Document Content Architecture

EFFF	IBM Token Ring Network PC Adapter
F000-FEFF	Non-IBM Reserved Block
FF00-FFFF	Context-Dependent Block

APPENDIX C. LIST OF ABBREVIATIONS AND SYMBOLS

A	address (SDLC)
ACT	active, activate

B'nnnn'	binary digits
BB	begin bracket
BBI	begin bracket indicator
BC	begin chain
BCI	begin chain indicator
BETB	between brackets
BF	boundary function
BIU	basic information unit
BLU	basic link unit
BSC	Binary Synchronous Communication
BTU	basic transmission unit

(c)	configuration services
C	control (SDLC)
CCA	communication controller adapter
CCITT	International Telegraph and Telephone Consultative Committee
CD	cross-domain, change direction
CDI	change direction indicator
CDRM	cross-domain resource manager
CEB	conditional end bracket
CEBI	conditional end bracket indicator
CICS/VS	Customer Information Control System/Virtual Storage
CMI	compression indicator
CNOS	change number of sessions
CONT	contention
COS	class of service
CP	control point
CPI	compaction indicator
CRC	cyclic redundancy checking
CRV	cryptography verification
CSI	code selection indicator
CSP	control sequence prefix

DAF	destination address field
DCE	data circuit-terminating equipment
DCF	data count field
DD	day of month
ddd	day of year

DEF	destination element address field
DEN	distribution user element name (SNADS)
DES	Date Encryption Standard
DFC	data flow control
DGN	distribution user group name (SNADS)
DISC	Disconnect (SDLC)
DISOSS	Distributed Office Support System
DISTIU	distribution interchange unit (SNADS)
DLC	data link control
DLU	destination logic unit
DM	Disconnected Mode (SDLC)
DPN	destination program name
DQ	dequeue
DR1I	definite response 1 indicator
DR2I	definite response 2 indicator
DSAF	Destination Subarea Address Field
DSP	data stream profile
DST	data services task or device service task
DSU	distribution service unit (SNADS)
DTE	data terminal equipment

EB	end bracket
EBCDIC	extended binary coded decimal interchange code
EBI	end bracket indicator
EC	end chain
ECI	end chain indicator
ED	enciphered data
EDI	enciphered data indicator
EFI	expedited flow indicator
ENA	extended network addressing
ENP	Enable Presentation
ER	explicit route
ERP	error recovery procedures
ERCL	exchange record length
ERI	exception response indicator
ERN	explicit route number
ERP	error recovery procedures
Exp	expedited flow
EXR	exception request

F	flag (SDLC)
FCB	forms control block
FCS	frame check sequence (SDLC)
FDX	full-duplex data flow
FF	flip-flop direction control
FFR	field-formatted record
FI	format indicator
FID	format identification
FIFO	first-in, first-out
FM	function management

FMD	function management data
FMDS	function management data services
FMH	function management header
FMHC	function management header concatenation
FNI	fixed fields without field separators
FRMR	Frame Reject (SDLC)
FS	fixed fields with field separators
FS2	fixed fields with or without field separators

GDS	general data stream

HDX	half-duplex data flow
hex	hexadecimal
HH	hours
HPCA	High-Performance Communication Adapter
HSID	half-session identification

I	information (SDLC), initiate only
ID	identification
IERN	initial explicit route number
ILU	initiating logical unit
IMS/VS	Information Management Systems/Virtual Storage
INB	in bracket
INP	Inhibit Presentation
IPL	initial program load
I/Q	initiate or queue
IRS	interchange record separator
ISO	International Organization for Standardization
IU	interchange unit (SNADS)

KEYIND	key indicator

LAN	local-area network
LCID	local coded graphic character set identifier
LH	link header
LIFO	last-in, first-out
LL	logical record length (prefix)
LMS	logical messages services
LRH	logical record header
LT	link trailer
LSID	local session identification
LU	logical unit
LVx	variable length parameter
LV1	variable length parameter, first position

(ma)	maintenance services
MGR	manager
MM	month, minutes
MPC	maximum presentation column
MPF	mapping field (BIU segments)
MPL	maximum presentation line
NA	network address
NAU	network addressable unit
NC	network control
Norm	normal flow
NS	network services
NUMRECS	number of records
OAF	origin address field
OEF	origin element field
OII	office information interchange
OLU	originating logical unit
OSAF	origin subarea field
P	primary
PC	path control
PCID	procedure correlation identifier
PD	padded data
PDI	padded data indicator
PDIR	peripheral data information record
PEND	pending
PGMNAME	program name
PI	pacing indicator
PIP	program initilization parameter
PIU	path information unit
PLU	primary logical unit
PNCP	peripheral node control point
POC	Program Operator Communication
PPU	primary physical unit
PRI	primary
PRID	procedure related identifier
PRN	primary resource name
PRTY	priority
PS	presentation services
PSH	presentation services header
PU	physical unit
PUCP	physical unit control point
P/F	poll/final (SDLC)

Q	queue
QC	quiesce complete
QEC	quiesce at end of chain
QR	queued response
QRI	queued response indicator

RCV	receive
RD	Request Disconnect (SDLC)
REC	receive
RECLEN	record length
RECID	record identification
RECTYPE	record type
REJ	Reject (SDLC)
RELQ	release quiesce
REN	routing element name (SNADS)
REQECHO	Request Echo Test
RES	resource
RH	request/response header
RIM	Request Initialization Mode (SDLC)
RJE	remote job entry
RLSD	released
RNR	Receive Not-Ready (SDLC)
RQ	request
RQD	definite-response request
RQE	exception request
RQR	request recovery
RR	Receive Ready (SDLC)
RRI	request/response indicator
RSP	response
RTI	response type indicator (+/-)
RTR	Ready To Receive (SDLC)
RU	request/response unit

S	secondary
(s)	session services
SC	session control
SCB	string control byte
SCS	SNA character string
SDI	sense data included indicator
SDLC	Synchronous Data Link Control
SEC	secondary
SESS	session
SIM	Set Initialization Mode (SDLC)
SLU	secondary logical unit
SNA	Systems Network Architecture
SNC	sense code
SNF	sequence number field

SNI	SNA network interconnection
SNADS	SNA distribution services
SNRM	Set Normal Response Mode (SDLC)
SPC	sync point command
SPU	secondary physical unit
SQN	sequence number
SRI	stack reference indicator
SS	seconds
SSCP	system services control point
STP	service transaction program
SU	shared; unnamed
SVC	services

T1	type-1 (node)
T2.0	type-2.0 (node)
T2.1	type-2.1 (node)
T4	type-4 (node)
T5	type-5 (node)
TC	transmission control
TERM	terminate
TEST	Test (SDLC)
TG	transmission group
TGN	transmission group number
TH	transmission header
TLU	terminating logical unit
TPF	transmission priority field
TPN	transaction program name
TRN	transparent
TS	transmission services
TWX	teletypewriter exchange service

UA	Unnumbered Acknowledgment (SDLC)
UI	Unnumbered Information (SDLC)
UNAVL	unavailable
UP	Unnumbered Poll (SDLC)
URC	user request correlation

VD	variable-length positional parameter
VOLID	volume identification
VR	virtual route
VRID	virtual route identifier
VRN	virtual route number
VRPRQ	virtual route pacing request
VRPRS	virtual route pacing response
VT	vertical tab

WP	word processing
XID	Exchange Identification (SDLC)
X'n...n'	hexadecimal digits
XMIT	transmit
YY	year
|	(vertical stroke) exclusive or
*	(asterisk) any value
¬	(not sign) logical not
**	exponential operator
_	(underscore) separates multiple terms, or qualifiers, in a phrase

INDEX

Systems Network Architecture
Formats

Publication No. GA27-3136-7

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. 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.

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.)

GA27-3136-7

Reader's Comment Form

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

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Dept. E95
P.O. Box 12195
Research Triangle Park, N.C. 27709-9990

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

IBM®

TAB DIVIDERS FOR SNA FORMATS

This package contains 18 tab dividers for you to insert in the accompanying publication, SNA Formats, GA27-3136-7. When placing the formats in a three-ring binder, insert these dividers as follows:

	Before...
Links	Chapter 1 divider page
XID Field	Chapter 2 divider page
TH	Chapter 3 divider page
RH	Chapter 4 divider page
Requests (RU)	Chapter 5.1 divider page
Responses (RU)	Chapter 5.2 divider page
Profiles	Chapter 6 divider page
User Data	Chapter 7 divider page
Common Fields	Chapter 8 divider page
Sense Data	Chapter 9 divider page
FM Headers	Chapter 10 divider page
PS Headers	Chapter 11 divider page
STP Records	Chapter 12 divider page
SNADS	Chapter 13 divider page
GDS Variables	Chapter 14 divider page
Appdx. A, B	page A-1
Abbreviations	page C-1
Index	page X-1

File No. GENL-30 (SNA)
SLSS No. 5743-SNA

GA27-3136-7

Printed in USA.

GA27-3136-07