• (now C.1.19.1.1, now C.1.19.1.5) added an optional subfield for V.25bis destination dial address and calling identification number

• (now C.1.19.1.3, now C.1.19.1.5) changed FRPVC DSAP and SSAP from canonical to noncanonical format

• (now C.1.19.1.7, now C.1.19.1.8) corrected the wording of the address length field in subfields X′46A5′ and X′46A6′ for X.25

* The following revisions were made based on review comments to Document Number SV40-1010-02/DLUR-9 (APPN DLUR Architecture Reference Version 1.3). These changes are indicated by an asterisk (*) throughout the text.

* • (8.1.3, 8.1.4, 12.1.2, 12.1.4, new 13.1.3, new C.1.4, new C.1.5, new C.1.12, new C.1.30.1, new C.1.30.2) updated the DLUR name substitution processing

indicator. This support includes all aspects of session awareness, i.e., SESSEND and CV X′2A′ as well as SESSST (see 10.1.2, “DLUR/S Dependent LU Session Awareness” on page 10-2 for more information on session awareness).

- In the CV X′0C′ are two indicators as to whether or not the dependent/subarea LU supports extended BINDs (byte 7, bits 4 and 6). The DLUR will remember these indicators′ settings with its other information about the dependent LU. Both bits off will indicate that the LU does not support
extended BINDs, while either bit on will indicate the LU does support extended BINDs. The DLUR
will also set the peripheral node extended BIND support to 0 and the subarea node extended BIND
support to 1.
- In the CV X′0C′ is an indicator as to whether or not the dependent LU supports network-qualified names (byte 7, bit 5). The DLUR will remember this indicator′s setting with its other information about the dependent LU.

If there already is an active LU-LU session when the +RSP(ACTLU) is received, the active LU-LU session awareness information will be sent to the SSCP in the DLUS by way of the CV X′2A′ added to the +RSP(ACTLU) by the DLUR (see 10.1.2, “DLUR/S Dependent LU Session Awareness” on page 10-2 for more information on session awareness).

8.1.3 INIT-SELF

When the DLUR node receives an INIT-SELF from a DLUR-attached dependent LU,

* • if the INIT-SELF format is Format 0, the DLUR will create a User Request Correlation (URC) (X′0A′) control vector, include the control vector in the RU, and remember that the URC is DLUR-generated;
* • if the INIT-SELF format is Format 1,
  * – if the URC field is not present, the DLUR will create a URC, include it in the RU, and remember that the URC is DLUR-generated.

* For all INIT-SELF RUs received from a DLUR-attached dependent LU the DLUR will save the PLU name and the User Request Correlation (URC) field from the RU and associate this information with the LU.

8.1.4 BIND

When the DLUR node receives a BIND whose RSCV terminates at the DLUR, the DLUR must check its LU table to determine the location of the SLU name specified in the RU. Once located,

- ***SLU session count processing -*** the LU′s session count will be checked:
  - if the count is zero, it will be incremented to one, and processing will continue;
  - if the count is already one, the BIND will be rejected with sense data X′0805 000A′;
- ***extended BIND processing -*** the BIND will be unextended if necessary, based on the extended
BIND support indicators in the CV X′0C′ in the +RSP(ACTLU) - this involves removing all keyed
control vectors, turning off the CVs included indicator, truncating trailing zeroes and the SLU name
(for non-negotiable BINDs), and updating the new length of the BIND RU;
- ***network-qualified name processing -*** the NETIDs (if any) will be stripped from the LU names in the BIND if necessary, based on the network-qualified names support indicator in the CV X′0C′ in the +RSP(ACTLU);
- ***name substitution processing -*** the PLU name will be replaced if necessary, based on whether or not the saved and BIND′s URCs match:

- if the URCs do not match, BIND processing will continue;
- if the URCs match,

* - the BIND PLU name will be replaced with the saved PLU name,
* - the DLUR will discard the saved URC and the saved PLU name, and
* - if the URC was DLUR-generated, the URC will be removed from the BIND;

- ***adaptive session pacing processing -*** the Adaptive Session-Level Pacing support indicator (byte 9, bit 0 in the BIND) value for the APPN stage (from the DLUR into the APPN network) will be remembered by the DLUR, which will then set the indicator for the REX stage (from the DLUR to the downstream PU) as follows - note that this processing is performed on non-negotiable BINDs as well as negotiable BINDs:

- if the BIND is unextended set the ASP indicator off;
- if the BIND is extended set the ASP indicator to the same value as the REX stage pacing parameter (byte 4, bit 0) in the Assign LU Characteristics (X′30′) control vector included in the X′1500′ GDS variable encapsulating the ACTLU (note that the supported/not supported settings are defined opposite to each other in the BIND and in the CV X′30′);

- ***pacing window size processing -*** the DLUR will put the REX stage pacing window size into the BIND only if the following three conditions are all true:
  1. the DLUR changed the BIND′s ASP indicator from 1 (adaptive) to 0 (fixed), and
  2. the BIND is negotiable, and
  3. the BIND pacing window size is zero;

and the BIND (if not rejected) will be forwarded to the appropriate internal or external component.

When a DLUR receives a BIND for a dependent LU and there is no active CP-SVR pipe servicing that dependent LU, the DLUR as a product option can forward the BIND to the LU.

8.1.5 +RSP(BIND)

When the DLUR node receives a +RSP(BIND),

& - ***short RSP(BIND) processing -*** the short +RSP(BIND) is converted into an extended short form
& RSP(BIND) by:

& - zeroing 31 (X′1F′) bytes for the extended short form RSP(BIND)
& - setting the RU code to BIND (X′31′)
& - setting the BIND type to nonnegotiable
& - including the SLU max send size and the PLU max send size from the BIND request
% - setting the pacing fields in the RSP(BIND) for the REX stage to the BIND request values
& - setting the limited resource bit if the link station is a limited resource link (LU 6.2 RSP(BIND)s
& only)

& and the pacing values for the SLU stage are saved so that fixed session pacing will be used with
& the window sizes specified in the BIND request;

- ***extended BIND processing -*** the +RSP(BIND) will be reextended if necessary, based on the extended BIND support indicators in the CV X′0C′ in the +RSP(ACTLU) - this involves reinserting

Chapter 12. DLUR Option Set Content

12.1 DLUR Tower Requirements

Any product that implements the DLUR tower must implement the following functions as described in this document:

12.1.1 CP-SVR Pipe

The DLUR node will have to be able to establish the CP-SVR pipes in conjunction with the activation or reactivation of a given SSCP-PU session as described in Chapter 5, “CP-SVR Pipe” on page 5-1. This will specifically include the ability to:

1. Support the new CPSVRMGR mode and COS.
2. Know the DLUS node to be contacted when a PU requires ACTPU.
3. De-activate CP-SVR pipes when idle.
4. Bring down a conwinner session when the conloser fails.

12.1.2 SSCP-PU & SSCP-LU Session Activation/Encapsulation

The DLUR node will also be required to activate SSCP-PU and SSCP-LU sessions over the CP-SVR pipes. This will include both logic to establish the sessions as well as logic to encapsulate and de-encapsulate the associated flows. Specific requirements include the ability to:

1. Generate an FQPCID for a PU requiring activation.
2. Correlate SSCP-PU and SSCP-LU sessions to PU images on the basis of the FQPCID.
3. Generate the REQACTPU RU and understand related responses and sense codes.
4. Interrogate ACTLU commands to build a table that relates SLU names to the PU image that supports them.
5. Maintain an active session count for each dependent LU.
6. Examine +RSP(ACTLU) for extended BIND and network-qualified name support information.
7. Examine INIT-SELF to acquire PLU name and URC information, and include a URC in the RU if one is not already present.
8. Optionally provide SDDLU NMVT support to allow for dynamic dependent LU definition in the DLUS (required only if dynamic definition is supported).
9. Detect ACTPU race conditions and resolve them appropriately.
10. Generate the REQDACTPU RU and understand related responses and sense codes.
11. Encapsulate and de-encapsulate SSCP-PU and SSCP-LU FID2 PIUs.
12. Send and receive multiple FID2 Encapsulation (X′1500′) GDS variables in a single BIU.
13. Support network name forwarding.

© Copyright IBM Corp. 1993, 1998

12.1.3 EN TGV Reporting

DLUR ENs will be required to report their TGVs up to the DLUS node only if the DLUR and the DLUS are in the same subnet. These TDUs will be sent as events that effect them occur. In the case where the DLUR and the DLUS are non-adjacent (or no CPSVCMG session exists between them), these TDUs will flow over the CP-SVR pipe. In the case where the DLUR and DLUS are also connected via a
% CPSVCMG session, these TDUs may optionally flow over the CP-CP session or over the CP-SVR pipe.

% 12.1.4 LU-LU BIND/UNBIND Processing

% 1. Provide reformating services:

% • Unextend extended BINDs destined for nodes that do not support receipt of such BINDs.

& • Reextend unextended +RSP(BIND)s when the corresponding BIND request was unextended.

& • Convert short +RSP(BIND)s into extended short form +RSP(BIND)s.

% • Unextend extended UNBINDs destined for nodes that do not support receipt of extended BINDs
% - no reextending of the +RSP(UNBIND) is required.

2. Remove NETIDs from LU names in BINDs destined for nodes that do not support network-qualified names.

3. Replace PLU names in BINDs destined for nodes which specified a PLU name in the INIT-SELF
* different from the PLU name received in the BIND, removing DLUR-generated URCs from the BIND.

4. Reject BINDs received for already active dependent LUs.

% 5. Properly set the Whole BIUs Required Indicator (WBRI) in the RSP(BIND) dependent on the
% segment reassembly capabilities of the DLUR BF.

6. Properly set the adaptive session pacing indicator in the BIND and +RSP(BIND) for the REX and
& APPN stages and set the REX stage pacing window size in the BIND when required.

7. Route BINDs to the destination PU/LU based upon the SLU name specified in the NS field of the BIND and information maintained in the SLU name table.

8. Perform intermediate routing functions when the LU is located on a downstream PU. This includes:

   • Intermediate routing function

   • Session outage notification

   • Maintaining pacing counts/indicators in both directions

12.1.5 Session Awareness Reporting

In order to assist the DLUS node in providing session limit management, the DLUR node must provide the DLUS node with session awareness data. The DLUR node will be required to:

1. Upon receipt of a positive RSP(ACTLU) from the attached dependent LU, if the dependent LU has an active LU-LU session, the DLUR must build a Session Information (X'2A') control vector, append it to the RSP(ACTLU), and send the RU to the SSCP by encapsulating the RU and sending it on the CP-SVR pipe to the DLUS.

2. Upon receipt of a positive RSP(BIND) from the attached dependent LU, the DLUR must build a SESSST RU and send it to the SSCP by encapsulating the RU and sending it on the CP-SVR pipe to the DLUS.

Table 13-1. GDS X'1500' Control Vectors		
	term used in flows	**RUs CV appears with**
% X'0E'	PU TG	REQACTPU
X'25'		RSP(ACTPU), REQACTPU
X'30'	LU CHAR	ACTLU
X'43'	PU CHAR	ACTPU
X'46'	PU TG	ACTPU, REQACTPU
X'51'	DL CAP	ACTPU, RSP(ACTPU), REQACTPU, RSP(REQACTPU)
X'60'	FQPCID	all RUs
X'81'	XID	REQACTPU, RSP(ACTPU)

13.1.1.1 XID Image (X'81') FID2 Encapsulation Control Vector

This is a new control vector, locally defined to this GDS variable, which contains either:

% • the I-field image of the XID0, XID1 or XID3 SDLC response of the external PU attached to the DLUR, or

• a DLUR-generated XID0 image for an internal PU or for an external PU which does not support XID.

13.1.2 ACTPU (Activate Physical Unit)

The Network Name (X'0E') control vector (type X'F1' - PU name) is included with the ACTPU when sent by a DLUS to a DLUR-serviced PU.

There is a network name forwarding request indicator in the PU Capabilities (X'80') ACTPU control
vector which, when set, requests the DLUR to leave the Network Name (X'0E') control vector in the
ACTPU RU when the DLUR forwards the RU to the PU. Also, if the indicator is set, the DLUR is
requested to leave the Network Name (X'0E') control vector in the ACTLU RU for all of the PU's LUs
when the DLUR forwards these RUs to the LUs (for more details about PU/LU network name for-
warding, see 6.2.4, "PU/LU Network Name Forwarding" on page 6-25).

* 13.1.3 INIT-SELF Format 0 (INITIATE-SELF)

* When the DLUR node receives an INIT-SELF Format 0 from a DLUR-attached dependent LU, it will
* create a User Request Correlation (URC) (X'0A') control vector and include the control vector in the
* RU as part of name substitution processing (for more details about name substitution processing, see
* 8.1.3, "INIT-SELF" on page 8-3 and 8.1.4, "BIND" on page 8-3).

13.1.4 REQACTPU (Request ACTPU)

This is a new RU built by the DLUR and sent to the SSCP in the DLUS requesting that an ACTPU be sent to establish an SSCP-PU session with a DLUR-attached PU. This PU could be predefined or undefined to the SSCP ; it could be internal to the DLUR or downstream of the DLUR.

The REQACTPU has two formats:

• Format X'0' - external PU

C.1.4 BIND

C.1.4.1 BIND (BIND SESSION)

PLU→SLU, Exp; SC

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.

BIND (BIND SESSION)

Byte	Bit	Content
0		X′31′ request code
1	0– 3	Format: 0000 (only value defined)
	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	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)
	1	Request control mode selection: 0 immediate request mode (only value defined for LU 6.2) 1 delayed request mode
	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)
	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
	5	Reserved
	6	FMH-1 SCB compression indicator (reserved for LU 6.2): 0 FMH-1 SCB compression will not be used on requests from primary 1 FMH-1 SCB compression may be used

BIND (BIND SESSION)

Byte	Bit	Content
	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	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)
	1	Request control mode selection: 0 immediate request mode (only value defined for LU 6.2) 1 delayed request mode
	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)
	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
	5	Reserved
	6	FMH-1 SCB compression indicator (reserved for LU 6.2): 0 FMH-1 SCB compression will not be used on requests from secondary 1 FMH-1 SCB compression may be used
	7	Send End Bracket indicator: 0 secondary will not send EB (only value defined for LU 6.2) 1 secondary may send EB
		FM Usage—Common LU Protocols
6	0	Whole-BIUs required indicator (reserved in nonextended, non-LU 6.2 BINDs, i.e., when control vector X′60′ is not present): 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 send-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
	1	FM header usage: 0 FM headers not allowed 1 FM headers allowed (only value defined for LU 6.2)
	2	Brackets usage and reset state: 0 The value of this bit should be 0 if either condition (1) or condition (2) is true. 1. Brackets are not used if neither primary nor secondary will send EB (byte 4, bit 7 = 0 and byte 5, bit 7 = 0). 2. Brackets are used and the bracket state managers′ reset states are INB if: • either primary or secondary, or both, may send EB (byte 4, bit 7 = 1 or byte 5, bit 7 = 1). • FM profile 19 is specified (byte 2 = X′13′). (only value defined for LU 6.2) 1 brackets are used and bracket state managers′ reset states are BETB

BIND (BIND SESSION)

Byte	Bit	Content
	3	Bracket termination rule selection; 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) *Note:* This bit is reserved if both of the following conditions are true. 1. Brackets are not used (byte 4, bit 7 = 0, byte 5, bit 7 = 0 , and byte 6, bit 2 = 0). 2. The FM profile is not 19 (byte 2 ≠ X′13′).
	4	Alternate code set allowed indicator: 0 alternate code set will not be used 1 alternate code set may be used
	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.
	6	BIS sent (reserved for TS profiles other than 4): 0 BIS not sent 1 BIS sent
	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	0– 1	Normal-flow send/receive mode selection: 00 full-duplex 01 half-duplex contention 10 half-duplex flip-flop 11 reserved *Note:* Values 00 and 10 are the only values defined for LU 6.2.
	2	Recovery responsibility: 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) *Note:* This bit is reserved if the normal-flow send/receive mode is full-duplex and the FM profile is not 19 (byte 2 ≠ X′13′).
	3	Contention winner/loser: 0 secondary is contention winner and primary is contention loser 1 primary is contention winner and secondary is contention loser

BIND (BIND SESSION)

Byte	Bit	Content
		Note 1: Contention winner is also brackets first speaker. *Note 2:* This bit is reserved if either condition (1) or condition (2) holds. 1. The normal-flow send/receive mode is full-duplex and the FM profile is not 19 (byte 2 ≠ X′13′). 2. All of the following are true. • The normal flow send/receive mode is HDX-FF (byte 7, bits 0–1 = 10). • Brackets are not used (byte 4, bit 7 = 0, byte 5, bit 7 = 0 , and byte 6, bit 2 = 0). • The FM profile is not 19 (byte 2 ≠ X′13′). • Symmetric responsibility for recovery is used (byte 7, bit 2 = 1).
	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.
	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).
	7	Half-duplex flip-flop reset states: 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) *Note:* This bit is reserved unless either condition (1) or conditions (2) *and* (3) hold. 1. The FM profile is 19. 2. The normal-flow send/receive mode is half-duplex flip-flop (byte 7, bits 0–1 = 10). 3. Brackets are not used or the bracket state manager′s reset state is INB (byte 6, bit 2 = 0).
		TS Usage
8	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 (or APPN hop-by-hop) 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 *Note:* The meanings of 0 and 1 are reversed from the corresponding staging indicator for the primary-to-secondary normal flow.
	1	Reserved
	2–7	Secondary send window size, in binary, for session-level pacing: a value of 0 indicates that there will be no pacing of requests flowing from the secondary. *Note:* If pacing on a session stage in a particular direction is not to be performed, the values for the window size on that stage are set to 0. For example, if there is to be no pacing in the secondary to primary direction, the primary receive and secondary send window sizes are both set to 0.

BIND (BIND SESSION)

Byte	Bit	Content
9	0	Adaptive session-level pacing support: 0 adaptive pacing not supported by the sending node: pacing window values in bits 2–7 of bytes 8, 9, 12, and 13 specify the fixed value implied in each pacing response; a 0 value in bits 2–7 of bytes 8 and 13 specifies no pacing in the secondary-to-primary direction; a 0 value in bits 2–7 of bytes 9 and 12 specifies the maximum window size is requested 1 adaptive pacing supported by the sending node: pacing window values in bits 2–7 of bytes 8, 9, 12, and 13 specify the *preferred minimum value* for each ISOLATED PACING MESSAGE; a 0 value specifies that the preferred minimum value is as large as possible; each adaptive pacing partner initializes its own send window size to 1 at session activation *Note:* If the PLU specifies adaptive pacing in BIND, and the SLU is able to support adaptive pacing, the SLU responds with this bit set to 1 in RSP(BIND). If the PLU indicates it does not support adaptive pacing, or if the SLU does not support adaptive pacing, this bit will be set to 0 in RSP(BIND). See C.1.30.1, “Request/Response Headers (RHs)” for further discussion of adaptive pacing.
	1	Reserved
	2–7	Secondary receive window size, in binary, for session-level pacing: a value of 0 causes the boundary function (BF) to substitute the value set by a system definition pacing parameter (if the system definition includes such a parameter) before it sends the BIND RU toward the secondary node; a value of 0 received at the secondary from the BF is interpreted to mean no pacing of requests flowing to the secondary. When fixed session-level pacing is used (byte 9, bit 0 = 0), this value is the fixed window size for the primary-to-secondary direction of the session stage. When adaptive session-level pacing is used (byte 9, bit 0 = 1), this value is the preferred minimum window size the primary end of the session stage recommends the secondary end of the session stage place in the IPMs it sends.
10		Maximum RU size sent on the normal flow by the secondary half-session. Bit 0 is interpreted as follows. 1. If bit 0 is set to 0, no maximum is specified and the remaining bits 1–7 are ignored. 2. If bit 0 is set to 1 (only value defined for LU 6.2), the byte is interpreted as X′ab′ = $a \times 2^b$ (Notice that, by definition, $a \geq 8$ and therefore X′ab′ is a normalized floating point representation.) See Figure C-1 on page C-22 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	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 (or APPN hop-by-hop) pacing (The primary send window size is always equal to the secondary receive window size.) *Note:* The meanings of 0 and 1 are reversed from the corresponding staging indicator for the secondary-to-primary normal flow (byte 8, bit 0).
	1	Reserved
	2–7	Primary send window size, in binary, for session-level pacing: a value of 0 causes the value set by a system definition pacing parameter (if the system definition includes such a parameter) to be assumed for the session; if this is also 0, it means no pacing of requests flowing from the primary (For one-stage pacing in the primary-to-secondary direction, this field is redundant with, and will indicate the same value as, the secondary receive window size—see byte 9, bits 2–7, above.)
13	0–1	Reserved

BIND (BIND SESSION)

Byte	Bit	Content
	2–7	Primary receive window size, in binary, for session-level pacing: a value of 0 means no pacing of requests flowing to the primary (For one-stage pacing in the secondary-to-primary direction, this field is redundant with, and will indicate the same value as, the secondary send window size—see byte 8, bits 2–7, above.)
		PS Profile
14	0	PS Usage field format: 0 basic format (only value defined)
	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 field *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)
16–21		Reserved
22	0	Extended security mechanism support: 0 Extended security mechanisms are not supported. 1 At least one extended security mechanism is supported. (And a structured data subfield X'14' is present.)
	1	Security extended sense data support: 0 This RU sender does not support receipt of sense data values in the range X'080FFF00' to X'080FFFFF'. 1 This RU sender supports receipt of sense data values in the range X'080FFF00' to X'080FFFFF' (and it does not automatically send UNBIND for the session if it does not understand a sense data value that it receives in this range).
	2–7	Reserved
23		Security Support Indicators
	0–2	Retired
	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.
	4	LU-LU verification protocol supported: 0 basic verification protocol 1 enhanced verification protocol
	5	Password substitution support: 0 not supported 1 supported
	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.
	7	Persistent verification capability: 0 Persistent Verification indicator is not supported on incoming FMH-5s. 1 Persistent Verification indicator is supported on incoming FMH-5s.

BIND (BIND SESSION)

Byte	Bit	Content
24	0	Reserved
	1– 2	Synchronization level: 01 confirm is supported 10 confirm, sync point, and backout are supported
	3	Reserved
	4– 5	Responsibility for session reinitiation (reserved when bit 6 of this byte is set to 1): 00 operator controlled 01 primary half-session will reinitiate 10 secondary half-session will reinitiate 11 either may reinitiate
	6	Parallel session support for LU-LU pair: 0 not supported 1 supported
	7	Change Number of Sessions GDS variable flow support (set to 1 if byte 24, bit 6 = 1): 0 not supported 1 supported
25	0	Reserved
	1	Limited-resource session indicator: 0 not a limited-resource session and thus the contention-winner LU will not deactivate it when it is no longer busy 1 a limited-resource session and thus the contention-winner LU will deactivate it when it is no longer busy
	2– 5	Reserved
	6– 7	Length-checked compression options: 00 no compression 01 compression supported by the PLU — compression to be determined by the SLU *Note:* This value indicates that the PLU supports compression but the PLU does not have any information that would indicate that this session would benefit from compression. 10 reserved 11 compression requested by the PLU *Note:* This value indicates that the PLU both supports and desires compression on this session. *Note 1:* On extended BINDs, the Length-Checked Compression (X′66′) control vector serves to define the requested compression options. *Note 2:* Bits 6-7, as defined, apply also to LU types 0, 1, 2, 3, and 6.1.
		End of PS Usage Field
26 – k		Cryptography Options
26	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

BIND (BIND SESSION)

Byte	Bit	Content
	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
	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	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)
	2– 4	Reserved
	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 8-byte value that, when deciphered, yields the session cryptography key used for enciphering and deciphering FMD requests
k + 1 – m		Network Services (NS) Primary LU Name Field (always present)
		This parameter is always network-qualified for implementations at the current level of SNA, unless the name is uninterpreted; uninterpreted names and back-level implementations may omit the network ID.
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 an INIT-SELF (or INIT-OTHER), the uninterpreted name as carried in that RU (and also in CDINIT for a cross-domain session)
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:		

BIND (BIND SESSION)

Byte	Bit	Content
m + 3 – n		Structured subfields (For detailed definitions, see C.1.30.2, "User Data Structured Subfields.")
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		Network Services (NS) Secondary LU Name Field (always present for negotiable BINDs; optionally present for nonnegotiable BINDs that include control vectors; otherwise, omitted) This parameter is always network-qualified for implementations at the current level of SNA; back-level implementations may omit the network ID.
p + 1		Length of secondary LU name (values 1 to 17 are valid) *Note:* Value 0 is retired.
p + 2 – r		Secondary LU name
Bytes r+1 – s are included only if byte 7, bit 6 specified that control vectors are included after the SLU name.		
r + 1 – s		Control vectors, as described in "Control Vectors" in C.1.30, "Information About Common Fields." *Note:* The following control vectors may be included; they are parsed according to subfield parsing rule KL (see "Substructure Encoding/Parsing Rules" in C.1.30, "Information About Common Fields"). X′0E′ Network Name control vector: PLU network name, X′F3′ (present in extended BINDs when bytes k+2 – m contain a name that is not network-qualified, such as an uninterpreted name) X′0E′ Network Name control vector: network-qualified CP name, X′F4′ (present in extended BINDs sent from nodes at the current level of SNA for PLU-inititated sessions when the Fully Qualified PCID (X′60′) control vector does not contain the real CP(PLU) name and the real CP(PLU) name is known) X′27′ XRF Session Activation control vector *Note:* Control vector X′27′ specifies that an XRF-active or XRF-backup session is to be activated. X′2B′ Route Selection control vector (present in extended BINDs when the BIND sender has the information available as the result of a Locate search and the session-route calculation) X′2C′ COS/TPF control vector (present in extended BINDs when the BIND sender supports mode-to-COS mapping or when it received the control vector on a Locate search reply) X′2D′ Mode control vector (present in non-LU6.2 extended BINDs unless the default mode name — eight X′40′ characters — is intended) X′60′ Fully Qualified PCID control vector (in which case the BIND is called an *extended BIND*) X′66′ Length-Checked Compression control vector (present in extended BINDs when compression is supported) *Note:* The receiving LU simply ignores unrecognized control vectors.
Note:		The length of the BIND RU cannot exceed 256 or 512 bytes. The length of the basic BIND RU is restricted to 256 bytes including the X′0E′, X′2C′, X′2D′, and X′60′ control vectors; any additional control vectors may cause the length to increase up to 512 bytes.
Note:		If the last byte of a Format 0 BIND 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 $a \times 2^b$. For example, X′C5′ represents (in decimal) $12 \times 2^5 = 384$.

Figure C-1. RU Sizes Corresponding to Values X′ab′ in BIND

C.1.5 INIT-SELF Format 0

C.1.5.1 INIT-SELF Format 0 (INITIATE-SELF)

ILU→SSCP, Norm; FMD NS(s)

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.

INIT-SELF Format 0 (INITIATE-SELF)

Byte	Bit	Content
0– 2		X′010681′ NS header
3	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
	4– 5	Reserved
	6	PLU/SLU specification: 0 DLU is PLU 1 DLU is SLU
	7	0 initiate only (I): do not enqueue. 1 initiate/enqueue (I/Q): enqueue the request if it cannot be satisfied immediately
4– 1 1		Mode name: an 8-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		

INIT-SELF Format 0 (INITIATE-SELF)

Byte	Bit	Content
m + 5 – n		Structured subfields (For detailed definitions, see C.1.30.2, "User Data Structured Subfields" on page C-65.)
n + 1 – p		Control vectors, as described in the section "Control Vectors" in C.1.30, "Information About Common Fields." *Note:* The following control vectors may be included; they are parsed according to subfield parsing rule LT (see "Substructure Encoding/Parsing Rules" in C.1.30, "Information About Common Fields"). X'0A' User Request Correlation (URC) control vector (included by the DLUR) X'34' LU Definition Override control vector (present if terminal operator entered one or more of the model terminal support override parameters with an implementation logon request)
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, i.e., the request will be dequeued after the other requests already in the queue.

C.1.6 REQACTPU

C.1.6.1 REQACTPU (REQUEST ACTIVATE PHYSICAL UNIT)

PU→SSCP, Norm; FMD NS(c)

REQACTPU is sent from the PU to an SSCP to request that ACTPU be sent to the PU named in the corresponding FID2 Encapsulation (X'1500') GDS variable.

REQACTPU (REQUEST ACTIVATE PHYSICAL UNIT)

Byte	Bit	Content
0– 2		X'41023E' NS header
3– 4		Reserved
5	0– 3	Format: X'0' Format 0: external PU X'1' Format 1: internal PU
	4– 7	Reserved

RSP(ACTLU)

Byte	Bit	Content
3 – m		Control vectors, as described in "Control Vectors" in C.1.30, "Information About Common Fields." *Note:* The following control vectors may be included; they are parsed according to sub-field parsing rule KL (see "Substructure Encoding/Parsing Rules" in C.1.30, "Information About Common Fields"). X'00' SSCP-LU Session Capabilities control vector (present to override the defaults of a 2-byte response, in which case always first) X'0C' LU-LU Session Services Capabilities control vector (present to override the defaults of a 2-byte response, in which case always second) *Note:* The following control vectors are appended by the T4 boundary node. X'28' Related Session Identifier control vector (present when the LU has an XRF backup session) X'2A' Session Information control vector (present when the LU already has an active session; may appear more than once for LUs that support XRF) X'68' XRF/Session Cryptography control vector (present when the LU has an active XRF session and is using session cryptography) *Note:* The following control vector is appended by the dependent LU requester. X'2A' Session Information control vector (present when the LU already has an active session)

A two-byte response may be sent; 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.

C.1.12 User Request Correlation (URC) (X'0A') Control Vector

C.1.12.1 User Request Correlation (URC) (X'0A') Control Vector

The User Request Correlation control vector carries an implementation-defined correlator that allows an SLU to correlate an incoming BIND with a previous SLU-initiated session request.

User Request Correlation (URC) (X'0A') Control Vector

Byte	Bit	Content
0– 1		Vector header; Key=X'0A' (see C.1.30.3, "Substructure Encoding/Parsing Rules" on page C-65 and Table C-1 on page C-65)
2 – n		Vector Data
2 – n		URC: implementation-defined correlator

C.1.30 Information About Common Fields

C.1.30.1 Request/Response Headers (RHs)

Cross Reference

This information is available in *SNA Formats* Chapter 5, Request/Response Headers (RHs).

C.1.30.2 User Data Structured Subfields

Cross Reference

This information is available in *SNA Formats* Chapter 8, User Data Structured Subfields.

C.1.30.3 Substructure Encoding/Parsing Rules

Cross Reference

This information is available in *SNA Formats* Chapter 9, Common Fields.

C.1.30.4 Control Vector Usage

Cross Reference

This information is available in *SNA Formats* Chapter 9, Common Fields.

Table C-1. Control Vector Usage

This table is shown in *SNA Formats* Chapter 9, Common Fields.

C.1.30.5 SNA Character Sets and Symbol-String Types

Cross Reference

This information is available in *SNA Formats* Appendix A, SNA Character Sets and Symbol-String Types.

C.2 Sense Data

C.2.1 0801

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.
- 0001 Independent LU Does Not Receive ACTLU: An ACTLU has been sent by the SSCP to an independent LU (sent by BF).