% • (C.1.18.1.1, C.1.18.1.6) added to CV X′4696′ a destination MAC address for
% FRPVC

% • (C.1.18.1.3, C.1.18.1.4) updated V.25/V.25bis destination dial address and
% calling identification number information

% • (C.1.18.1.3, C.1.18.1.4, C.1.18.1.5, C.1.18.1.6) changed encodings of Token-Ring
% signalling information from canonical to noncanonical

% • (C.1.18.1.5) retired CV X′4695′ SDLC station role usage for SDLCNS and
% SDLCSW

The following revisions were made based on review comments to Document
Number SV40-1010-01/DLUR-8 (APPN DLUR Architecture Reference Version 1.2).
These changes are indicated by a pound sign (#) throughout the text.

• (5.6.2.1.2, 5.6.2.1.3, 13.1.7, 13.4.6, now C.1.20.1, C.2.12) added a new nondis-
ruptive UNBIND-initiated CP-SVR pipe deactivation sense data to permit the
DLUR to immediately reinitiate CP-SVR pipe activation while requiring the
DLUR to select a different DLUS for the new pipe

• (5.6.3.2) updated the DLUR REQDISCONT processing to return a
RSP(REQDISCONT) upon receipt of RSP(REQDACTPU)

• (5.7.3.2) clarified the sense code used in -RSP(BIND) or UNBIND for
“surprise” CP-SVR pipe activation

• (6.2, 6.2.1, 6.2.2, new 6.2.4, 6.3, 6.3.1, 6.3.3, 13.1.2, 13.1.7, C.1.3.1, C.1.3.1.1,
now C.1.20.1) added the DLUR support for forwarding PU/LU names

• (8.1.2) updated the DLUR processing of CV X′0C′ on +RSP(ACTLU)

• (8.1.4) updated the DLUR processing to unextend an extended BIND

• (8.1.4) updated the DLUR BIND adaptive session pacing processing

• (8.1.5) updated the DLUR +RSP(BIND) processing to include the Route
Selection (X′2B′) control vector

• (8.2) updated the EN DLUR tail vector registration processing

• (8.2.1) updated the DLUS processing for setting the receipt of TDUs support
DLUS capability

• (13.3.1, C.1.8, new C.1.15) added the Route Selection (X′2B′) control vector to
the SESSST format X′03′ RU

• (appendix C) updated formats appendix, removing change bars for format
changes published in SNA Formats editions GA27-3136-15 and GA27-3136-16

• (now C.1.19.1.1, now C.1.19.1.5) added an optional subfield for V.25bis desti-
nation 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

% - reassemble the segments and attempt to send the whole message to the SLU (only if the
% DLUR supports segment reassembly); if the reassembled message is too large, the DLUR
% can segment the message and send the segments to the SLU

% - attempt to send the segments received from the PLU to the SLU; if a segment is too large,
% the DLUR can segment the PLU segment and send the new segments to the SLU

% – if WBRI=1 and DLUR BF can support segment reassembly, set WBRI to 0

% – if WBRI=1 and DLUR BF cannot support segment reassembly, set WBRI to 1;

- ***adaptive session pacing processing -*** the Adaptive Session-Level Pacing support indicator value will be restored to its remembered value, unless the SLU is locally attached and the negotiable or non-negotiable RSP(BIND) does not indicate adaptive session pacing support (ASP indicator set to 0), in which case the DLUR can leave the ASP indicator at 0 regardless of the remembered value;
- ***session awareness processing -*** SESSST will be built and sent to the SSCP on the CP-SVR pipe
  - When a DLUR receives a +RSP(BIND) for a dependent LU and there is no active CP-SVR pipe servicing that dependent LU, the DLUR should try to activate a CP-SVR pipe. When the CP-SVR pipe is activated, the SESSST should not be sent on the new CP-SVR pipe. 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).

and the +RSP(BIND) will be forwarded to the appropriate internal or external component.

8.1.6 LU-LU Session Deactivation

When the DLUR node receives a -RSP(INIT-SELF) or detects a link failure affecting an LU-LU session, it will discard the saved URC and the saved PLU name associated with the dependent LU.

When the DLUR node receives a -RSP(BIND), UNBIND, or detects a link failure affecting an LU-LU session, it will reset the session count for the associated dependent LU to zero.

% The UNBIND will be unextended if necessary, based on the extended BIND support indicators in the CV
% X′0C′ in the +RSP(ACTLU) - this involves:

% • ***all UNBIND types except X′FE′ -*** truncating the UNBIND RU to only the request code plus the
% UNBIND type;

% • ***UNBIND type X′FE′ -*** truncating the UNBIND RU to only the request code, the UNBIND type, and
% the sense data.

% No reextending of the RSP(UNBIND) is required.

8.2 DLUR EN Tail Vector Registration

In order for the DLUS node to send Locate Find/Found flows with the appropriate associated resource hierarchy as described above, it will have to know the DLUR tail vectors. In the case where the DLUR node is a NN this will not be necessary. The EN DLUR node will be required to register its tail vectors with the DLUS node; this registration will be done only when the DLUS and DLUR are in the same subnet.

• The EN DLUR can determine whether or not it is in the same subnet with the DLUS by examining
the Locates it receives during CP-SVR pipe activation, either:

– the Locate sent by the DLUS to activate its conwinner CPSVRMGR session, or

– the Locate reply sent by the DLUS in response to the Locate sent by the DLUR to activate its conwinner CPSVRMGR session

In the Intersubnetwork Search (X′82′) Locate control vector there is an intersubnetwork TGs crossed count:

– if the count is zero, the DLUS and DLUR are in the same subnet

– if the count is nonzero, the DLUS and DLUR are not in the same subnet

An EN DLUR in the same subnet as the DLUS will register or not register its tail vectors dependent on the DLUS′s setting in the DLUR/S Capabilities control vector of the indicator for support of the RECEIVE_TDU_TP (X′22F0F0F4′) service transaction program (see 8.2.1, “DLUR/S Capabilities Exchange” on page 8-7 for more information about when the DLUS will set the indicator):

• if the indicator is off, the EN DLUR will not register its tail vectors

• if the indicator is on, the EN DLUR will register its tail vectors

DLUR tail vector registration will involve sending TOPOLOGY_DATABASE_UPDATEs (TDUs) to the DLUS node:

% 1. at initial TGV registration,

% a. if the DLUR and the DLUS are adjacent, then the DLUR may optionally send the TDUs over the CP-CP session between the two nodes

% b. otherwise the TDUs will be sent over the CP-SVR pipe to the DLUS node

% 2. if the CP-SVR pipe is selected for initial registration:

% a. continue to use it for subsequent TGV registration even if a CP-CP session between adjacent DLUR and DLUS is activated later

% b. if the CP-SVR pipe fails, once the pipe is reactivated (to either the same or a different DLUS), return to step 1 and send all tail vectors

% 3. if the CP-CP session is selected for initial registration:

% a. continue to use it for subsequent TGV registration

% b. if the CP-CP session fails, return to step 1 and send all tail vectors

% c. if the CP-SVR pipe fails, once the pipe is reactivated (to either the same or a different DLUS), return to step 1 and send all tail vectors

The DLUS node will receive, but not propagate, these TDUs.

This will require the DLUR and DLUS CPs to be modified such that the TDU send and receive TPs can also run over the CP-SVR pipe using the CPSVRMGR mode. No special CP-SVR pipe should be brought up for this purpose. The existing CP-SVR pipe should be used. Since a DLUR may have CP-SVR pipes with multiple DLUS nodes, these TDUs must be sent over each CP-SVR pipe to each partner DLUS node.

8.2.1 DLUR/S Capabilities Exchange

TDUs can be sent on the pipe any time after CP-SVR pipe activation completes. Activation includes the exchange of DLUR/S Capabilities control vectors, so TDUs can be sent by the DLUR after:

- DLUR-initiated pipe activation: receipt of +RSP(REQACTPU) from DLUS
- DLUS-initiated pipe activation: transmission of +RSP(ACTPU) to DLUS

Included within the DLUR/S Capabilities control vector are fields relevant to EN tail vector registration:

- a Flow Reduction Sequence Number (FRSN) -
  - the DLUR will always set this field to zero in the DLUR/S Capabilities control vector; this will tell the DLUS to throw away any tail vectors previously registered by this DLUR;
  - the DLUS will set this field to the last FRSN value received from the DLUR;
- an indicator for support of the RECEIVE_TDU_TP (X′22F0F0F4′) service transaction program - when set, the sending node supports receipt of TDUs on this session
  - the DLUS may set this indicator on unless the TDU information will not be used for route calculation;
  - the DLUR should set this indicator off.

8.2.2 Initial EN Tail Vector Registration

After CP-SVR pipe activation, the DLUR will send all of its tail vectors to the DLUS (for more information, see “Processing CP_STATUS Signals” in the *SNA APPN Architecture Reference*).

The DLUS will process the received TDUs as described in “TDU Processing” in the *SNA APPN Architecture Reference*, with one exception: these TDUs should not be propagated to other NNs.

8.2.3 EN Tail Vector Registration Updates

When there is a change to a previously registered tail vector, the DLUR will send an update TDU to the DLUS (for more information, see “Processing Resource Updates” in the *SNA APPN Architecture Reference*).

The DLUS will process the received TDUs as described in “TDU Processing” in the *SNA APPN Architecture Reference*, with one exception: these TDUs should not be propagated to other NNs.

C.1.22.2 Locate Control Vectors

C.1.22.2.1 Search Scope (X′80′) Locate Control Vector

Search Scope (X′80′) Locate Control Vector

Byte	Bit	Content
0– 1		Vector header; Key=X′80′ (see “Substructure Encoding/Parsing Rules” in -- Heading ′CF′ unknown --)
2 – n		Vector Data
2		Hop count: a binary value specifying the number of hops that may be traversed for a broadcast search (set by the broadcast origin CP and decremented, on the search request, by intermediate CPs participating in the broadcast search)

C.1.22.2.2 PCID Modifier (X′81′) Locate Control Vector

PCID Modifier (X′81′) Locate Control Vector

Byte	Bit	Content
0– 1		Vector header; Key=X′81′ (see “Substructure Encoding/Parsing Rules” in -- Heading ′CF′ unknown --)
2 – n		Vector Data
2– 3		Procedure resubmit number: A 2-byte (always increasing) binary number that indicates the number of times a Locate search request has been resubmitted by any and all nodes
4		Last significant half-byte in the PCID Modifier List field (zero-origin index to the last half-byte in the list that has been claimed by a node on the procedure path)
5 – n		PCID modifier list: up to 20 half-bytes of list entries, with each list entry containing a binary count of the number of new search subprocedures created by a particular node. The Last Significant Half-byte field is used to determine the next available half-byte in the PCID Modifier List. The PCID modifier list has a minimum length of 1 byte, and must always contain an integral number of bytes. Thus, if byte 4 specifies an odd number of half-bytes (i.e., byte 4 is even), then the last byte in the list is padded with X′0′ to form a full byte. A node must be able to receive at least twenty entries in the list.

C.1.22.2.3 Intersubnetwork Search (X′82′) Locate Control Vector

The Intersubnetwork Search Locate control vector is present when the Locate has crossed an intersubnetwork TG. It contains search control information used by extended border nodes.

Intersubnetwork Search (X′82′) Locate Control Vector

Byte	Bit	Content
0– 1		Vector header; Key=X′82′ (see “Substructure Encoding/Parsing Rules” in -- Heading ′CF′ unknown --)
2 – n		Vector Data
2		Subnetwork Controls
2	0	Limit search to local subnetwork indicator: 0 Search may span subnetwork boundaries. 1 Search may not exit the local subnetwork. Set by an EBN to limit a broadcast search that is part of a cross-subnetwork search to the local subnetwork.
	1	Limit search to local and adjacent subnetworks indicator: 0 Search is not limited to local and adjacent subnetworks. 1 Search is limited to local and adjacent subnetworks. Set by the EN image of a PBN after receiving or before sending a search across an ISTG; set by an EBN after receiving, or before sending a search across a peripheral ISTG, if the intersubnetwork-TGs-crossed count (ICC) in byte 4 is nonzero.
	2	Reserved
	3	Reserved
	4	Prevent subarea path indicator: 0 EBNs on the path of the search may set the subarea network search suppression indicator in the Locate GDS variable to 0. 1 Nodes on the search path may not set the subarea network search suppression indicator to 0. (Set by the DLUS or DLUR to ensure an APPN-only CP-SVR pipe between them.)
	5	DLUR search required indicator: 0 A search to obtain the DLUR TG vectors is not required. 1 A search to obtain the DLUR TG vectors is required.
	6– 7	Reserved
3		Maximum subnetwork-visit count (SNVC): The number, in binary, of subnetworks a search may traverse before being terminated with a complete negative reply. SNVC is 1 greater than the number of ISTG crossings permitted in the remainder of the search.
4		Intersubnetwork-TGs-crossed count (ICC): When nonzero, ICC is the number, in binary, of ISTGs a search has crossed. Set to 0 at the origin EBN and incremented by 1 at each EBN after receiving a search across an ISTG, and before sending a search across a peripheral ISTG. When 0, the search has either crossed no ISTGs, or has crossed one peripheral ISTG.
5– 6		Reserved

C.1.22.3 Cross-Subnetwork Loop Prevention (X′84′) Locate Control Vector

The Cross-Subnetwork Loop Prevention control vector identifies the nodes to be used to enter or exit an APPN network (extended border nodes or interchange nodes) in the cross-subnetwork session path through an APPN network. The control vector is carried on the Locate GDS variable during the Locate search request and may be originated, modified, and/or used by border nodes and interchange nodes in the session path.