• (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.

* • (5.1, 5.3, 5.6.1, 5.6.2.1, 5.6.2.1.1, new 5.6.2.1.3, now 5.6.2.1.4, new 5.8, 13.1.8, now C.1.23, C.2.12) added CP-SVR pipe persistence processing

* • (5.6.3, new 5.6.3.3) added -RSP(REQDACTPU) DLUR processing

* • (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

Chapter 5. CP-SVR Pipe

5.1 Overview

* CP-SVR pipe initiation is usually tied to PU activation. A DLUS or DLUR node will normally only acti-
* vate a CP-SVR pipe when a PU requires activation and no CP-SVR pipe between the DLUR and the
* DLUS (or SSCP) supporting the PU already exists.

* • For those situations when a CP-SVR pipe must be activated without an associated PU requiring
* activation, or when a CP-SVR pipe must remain active after all associated PUs have been deacti-
* vated, a ***persistent*** CP-SVR pipe can be activated (for more information about persistent CP-SVR
* pipes, see 5.8, “Persistent CP-SVR Pipes” on page 5-75).

A PU would require such activation under several circumstances including:

1. Receipt of an XID requesting ACTPU
2. Take over from loss of an active CP-SVR pipe
3. Explicit node operator command

In addition, the CP-SVR pipe may be initiated in one of two directions:

1. From the requester to the server
2. From the server to the requester

It is the intent of this architecture to support dynamic establishment of the CP-SVR pipe with an option to explicitly define the requester/server relationships if so desired. Due to the complexities of such dynamics, however, this version of the document requires that the DLUR and DLUS relationships be predefined. Dynamic determination of a DLUS by a DLUR node as well as dynamic determination of a DLUR by a DLUS node will be addressed by a separate architecture extension.

The following sections will describe how the CP-SVR pipe is established in both the DLUS to DLUR and DLUR to DLUS direction.

5.2 CPSVRMGR Mode

The pair of LU 6.2 sessions which make up a CP-SVR pipe will use a new mode known as CPSVRMGR. CPSVCMG will not be used due to architecture assumptions regarding adjacency of CPSVCMG session partners. SNASVCMG will not be used to avoid contention for the session with other node functions such as network management and change number of sessions (CNOS) negotiation. An additional benefit of using a new mode (CPSVRMGR) is that these sessions can be uniquely identified by the CPSVRMGR mode name, thus allowing the node to provide special session management logic to these sessions as described in 5.5, “CP-SVR Pipe Activation” on page 5-8.

The CPSVRMGR mode will use the SNASVCMG class of service (COS). By using this default COS definition, DLUR nodes will not have to worry about matching COS definitions at the NNS. In this way, a DLUR EN can be supported by an existing base APPN NN.

© Copyright IBM Corp. 1993, 1998

5.3 DLUR-Initiated CP-SVR Pipe Activation

By definition, all dependent LUs require the services of an SSCP. This is done via SSCP-PU and SSCP-LU sessions. The SSCP-PU session flows between the SSCP and PU2.0 or T2.1 node providing local support for the dependent LU. This session is created when the SSCP sends an ACTPU to the PU2.0/T2.1 node. The SSCP-LU session is created when the SSCP sends an ACTLU over the SSCP-PU session to the dependent LU itself. Normally a T2.1 or PU2.0 device supporting dependent LUs is Boundary Function (BF)-attached to a subarea node and is either provided the SSCP services upon activation or (in the case to T2.1 nodes) requests them via XID3 (this is reflected in the ACTPU suppression indicator: bytes 8-9 bit 8 of XID3).

DLUR nodes are APPN T2.1 nodes (EN or NN) which provide both local and downstream support for dependent LUs. Since DLUR nodes provide BF-like support for these LUs, each DLUR node must signal to the SSCP (or DLUS node in this case) when a PU (either local or downstream) requires SSCP services. For locally supported dependent LUs, the DLUR node requests SSCP services upon node activation. For downstream nodes supporting dependent LUs, the DLUR node waits for the downstream node to request them via XID. When a T2.1 or T2.0 PU exchanges XIDs (or equivalent) with a DLUR node, it signals via XID that it requests SSCP services (this is done by setting the ACTPU suppression indicator, bytes 8-9 bit 8 of XID3, to 0; for XID0 or SNRM without XID, ACTPU is always sent).

DLUR nodes request SSCP services from a DLUS node via the CP-SVR pipe and the REQACTPU RU. When a DLUR node detects the need for SSCP services (as described above), it activates a conwinner CPSVRMGR session (assuming one does not already exist) to a DLUS node and sends a REQACTPU RU. The DLUS node then responds by sending an ACTPU to the T2.0 PU/T2.1 supporting the dependent LUs via a conwinner CPSVRMGR session to the DLUR node. This creates an SSCP-PU session encapsulated in a CP-SVR pipe, enabling the DLUS node to provide SSCP services to the dependent LUs. A DLUR optionally may establish a conwinner CPSVRMGR session with a DLUS prior
* to receiving any requests for SSCP services from its PUs, e.g., a ***persistent*** CP-SVR pipe can be acti-
* vated (for more information about persistent CP-SVR pipes, see 5.8, “Persistent CP-SVR Pipes” on
* page 5-75).

% 5.3.1 DLUR PU Identification

% When the DLUR sends a REQACTPU RU it must identify the PU for which it is requesting SSCP ser-
% vices. The PU will be identified by including with the REQACTPU one or both of the following:

% 1. ***XID*** - This is either an image of an XID received from a PU externally attached to the DLUR or a
% DLUR-generated XID image (for an internal PU or an external PU not supporting XID transmission).

% 2. ***PU TG Descriptor*** - This item consists of one or both of the following:

% a. ***PU Name*** - The PU name can be sent to the DLUS, either to match a predefined PU statement
% or to be used in the creation of a dynamic PU at the DLUS. The PU name can be sent in a
% network name (CV X′0E′ - type X′F1′ = PU name) control vector either with or without a
% NETID:

% 1) non-network qualified (without NETID) -in this case the DLUS will match only on PU name.

% 2) network qualified (with NETID) - in this case the DLUS will match on NETID.PUname. It is
% an implementation option for the DLUR to choose how to get the PU name and the NETID to
% put together a network-qualified PU name to send to the DLUS. The DLUR could require
% the user to define a network-qualified PU name, or only define a PU name and use its own
% NETID, or in the case of a downstream PU learn about the NETID via XID (remember: a

5.6 CP-SVR Pipe Deactivation

There are several situations where a CP-SVR pipe will be deactivated, including normal deactivation and loss of session partner. In general, the DLUS will normally bring down the pipe, but either the DLUR or DLUS may deactivate the pipe if error or outage conditions have been detected. In every case, whichever end initiates the deactivation will deactivate both its conwinner and conloser sessions. In addition, when a DLUS or DLUR is requested to deactivate its conloser session, it should also deactivate its own conwinner session. Reactivation of the CP-SVR pipe must not begin until both conwinner and conloser sessions are deactivated (see 5.7, “Pipe And Session Failure Logic” on page 5-58 for more information on recovery of the CP-SVR pipe). LU 6.2 BIS protocols will be used to clear the pipe first during deactivation only when the UNBIND is issued without sense data.

5.6.1 Normal CP-SVR Pipe Deactivation

When the DLUS receives an encapsulated positive RSP(DACTPU) on the CP-SVR pipe, in addition to de-encapsulating the RU and passing it on to the SSCP, it will check to see if it can deactivate the CP-SVR pipe. The DLUS will deactivate the CP-SVR pipe if

- there are no active SSCP-PU sessions using the pipe and
- the pipe is not persistent (for more information about persistent CP-SVR pipes, see 5.8, “Persistent CP-SVR Pipes” on page 5-75) and
- one of the following is true
  - there are pending sessions but a DLUR/S capabilities mismatch was detected; or
  - there are no pending SSCP-PU sessions, e.g.,
    - REQACTPU was received from the DLUR, but no ACTPU has been sent by the SSCP yet
    - ACTPU was sent by the SSCP, but no RSP(ACTPU) has been received by the DLUS yet

If these conditions exist, the DLUS will deactivate its conwinner and conloser sessions by issuing an UNBIND on each session. Depending on the order and timing of the arrival of the UNBINDs, the DLUR will act as follows:

- ***second UNBIND arrives before first is processed*** - the DLUR will return a positive RSP(UNBIND) on each session, and the pipe will be deactivated;
- ***UNBIND(DLUS conloser) arrives after UNBIND(DLUS conwinner) is processed*** - the DLUR will return a positive RSP(UNBIND) on its conloser session and send an UNBIND on its own conwinner session. When the DLUS and DLUR receive the UNBINDs on this (DLUR conwinner) session, they will each return a positive RSP(UNBIND), and the pipe will be deactivated. The RSP(UNBIND)s will clean up the session connectors as they flow, so neither of them will actually arrive at their destination.
- ***UNBIND(DLUS conwinner) arrives after UNBIND(DLUS conloser) is processed*** - the DLUR will return a positive RSP(UNBIND) on its conwinner session and send an UNBIND on its own conloser session. When the DLUS and DLUR receive the UNBINDs on this (DLUR conloser) session, they will each return a positive RSP(UNBIND), and the pipe will be deactivated. The RSP(UNBIND)s will clean up the session connectors as they flow, so neither of them will actually arrive at their destination.

Figure 5-4 on page 5-22 through Figure 5-6 on page 5-27 illustrate these normal CP-SVR pipe deactivation scenarios:

5.6.2 Abnormal CP-SVR Pipe Deactivation

There are other situations where the DLUS and DLUR should deactivate the CP-SVR pipe:

- If either the DLUR or DLUS should receive an UNBIND on its conloser session, it should return a positive RSP(UNBIND) on that session and, if its conwinner session is active or pending, send an UNBIND on the conwinner session.
- If an outage is detected on the CP-SVR pipe, and an UNBIND cannot be successfully routed to the session partner, then the DLUS or DLUR should deactivate and clean up both its conwinner and conloser sessions.
- Whenever a DLUR/S capabilities mismatch is detected, the CP-SVR pipe should be deactivated.

Reactivation of the CP-SVR pipe can be initiated from either the DLUS or DLUR, or both, upon completion of the deactivation of the current pipe (see 5.7, “Pipe And Session Failure Logic” on page 5-58 for more information on recovery of the CP-SVR pipe).

5.6.2.1 UNBIND-Initiated CP-SVR Pipe Deactivation

One of the DLUS requirements is to support forced deactivation of the DLUR-related sessions, i.e.,

- the CP-SVR pipe between the DLUS and the DLUR
- the SSCP-PU sessions between the SSCP in the DLUS and the PU serviced by the DLUR
- the SSCP-LU sessions between the SSCP in the DLUS and the LU serviced by the DLUR
- the LU-LU sessions involving the LU which has an SSCP-LU session with the SSCP in the DLUS

Normally, DACTLUs and DACTPUs would be sent followed by an UNBIND for the CPSVRMGR session. To speed things up, new sense data has been created for the CP-SVR pipe UNBIND which will indicate
* to the DLUR to deactivate some or all of the session types listed above. There are three major sense data categories:

- disruptive - all sessions (CPSVRMGR, SSCP-PU, SSCP-LU, LU-LU) will be deactivated by the DLUR for its PUs and LUs serviced by this DLUS
- nondisruptive - all sessions except active LU-LU sessions will be deactivated
* - persistent - all sessions (if any) have been deactivated and now the persistent CP-SVR pipe should
* be deactivated (for more information about persistent CP-SVR
* Pipes” on page 5-75)

* ***5.6.2.1.1 Disruptive UNBIND-Initiated CP-SVR Pipe Deactivation:*** There are two disruptive session
* deactivation sense data, X′08A0 0007′ and X′08A0 000C′. If either sense data appears in the Extended Sense Data control vector (X′35′) in an UNBIND on the conloser CPSVRMGR session, the DLUR will:

- deactivate all SSCP-PU, SSCP-LU, and LU-LU sessions for its PUs and LUs serviced by this DLUS
- return a positive RSP(UNBIND) on the conloser session and, if its conwinner session is active or pending, send an UNBIND on the conwinner session

5.6.2.1.2 Nondisruptive UNBIND-Initiated CP-SVR Pipe Deactivation: There are four nondisruptive
session deactivation sense data: X′08A0 0008′, X′08A0 0009′, X′08A0 000A′, and X′08A0 000B′. If one of these sense data appears in the CV X′35′ in an UNBIND on the conloser CPSVRMGR session, the DLUR will:

- deactivate all SSCP-PU and SSCP-LU sessions for PUs and LUs serviced by this DLUS

- deactivate all active and pending LU-LU sessions for LUs
  - serviced by this DLUS, and
  - defined with ANS=STOP
- return a positive RSP(UNBIND) on the conloser session and, if its conwinner session is active or pending, send an UNBIND on the conwinner session

The distinctions between each of these sense data are:

- X'08A0 0008' - DLUR can immediately reinitiate CP-SVR pipe activation
- X'08A0 0009' - DLUR can immediately reinitiate CP-SVR pipe activation / this sense data is used when a protocol violation is detected
- X'08A0 000A' - DLUR should wait for a DLUS (need not be the same DLUS sending the UNBIND) to reinitiate CP-SVR pipe activation / DLUR-implementing products may choose to wait:
  - indefinitely, or
  - for a designated (either system- or user-defined) period of time for a CP-SVR pipe activation from a DLUS, after which the DLUR will go ahead with CP-SVR pipe activation, with either the same or a different DLUS.
- # X'08A0 000B' - DLUR can immediately reinitiate CP-SVR pipe activation, but not with the same # DLUS; if no other DLUS is defined at the DLUR, then the DLUR should perform the same actions as # it does for sense data X'08A0 000A' when waiting for a DLUS to reinitiate CP-SVR pipe activation
  - # At CP-SVR pipe activation, both DLURs and DLUSs will indicate support for this sense data by # setting on the nondisruptive DLUS-DLUR session deactivation type X'08A0 000B' supported # indicator in the DLUR/S Capabilities (X'51') control vector.

* ***5.6.2.1.3 Persistent UNBIND-Initiated CP-SVR Pipe Deactivation:*** There is one persistent session * deactivation sense data, X'08A0 000D', sent by the DLUR in the CV X'35' in an UNBIND on its * conwinner CPSVRMGR session to deactivate a persistent CP-SVR pipe. The DLUR can deactivate a * persistent pipe in this manner at any time, as long as the number of current and pending SSCP-PU * sessions is zero.

* ***5.6.2.1.4 Reactivation After UNBIND-Initiated CP-SVR Pipe Deactivation:*** With three exceptions, after the CP-SVR pipe deactivation has completed, both the DLUS and the DLUR can immediately activate a # new CP-SVR pipe, with the same or a different DLUR/S partner. The exceptions are:

- # upon receipt of sense data X'08A0 000A', the DLUR must wait for a DLUS to activate the pipe; it is a product option as to whether the DLUR waits indefinitely or for a finite (system- or user-defined) period of time before activating a new CP-SVR pipe
- # upon receipt of sense data X'08A0 000B', the DLUR can immediately reinitiate CP-SVR pipe acti- # vation, but not with the same DLUS; if no other DLUS is defined at the DLUR, then the DLUR should # perform the same actions as it does for sense data X'08A0 000A' when waiting for a DLUS to reini- # tiate CP-SVR pipe activation
- * upon receipt of sense data X'08A0 000C', the DLUR can immediately reinitiate persistent CP-SVR * pipe activation, but not with the same DLUS; if no other DLUS is defined at the DLUR, then the * DLUR can perform the same actions as it does for sense data X'08A0 000A' when waiting for a * DLUS to reinitiate CP-SVR pipe activation or it can attempt to activate a nonpersistent CP-SVR pipe * with the same DLUS if it has a PU needing service

* 5.8 Persistent CP-SVR Pipes

* Normally, CP-SVR pipe activation/deactivation architecture requires:

* • the DLUR and DLUS to activate a CP-SVR pipe only when a PU requires activation, and

* • the DLUS to deactivate a CP-SVR pipe when there are no pending or active SSCP-PU sessions
* associated with it.

* There are requirements, e.g., that DLUR's duplicate Token-Ring Interface Coupler (TIC) support must
* match NCP's, which mandate that some CP-SVR pipes be allowed to be activated and to remain active
* even when there are no associated PU or LU sessions.

* For those situations when a CP-SVR pipe must be activated without an associated PU requiring acti-
* vation, or when a CP-SVR pipe must remain active after all associated PUs have been deactivated, a
* ***persistent*** CP-SVR pipe can be used.

* There remains a limit of one active CP-SVR pipe per DLUS/DLUR pair, and that pipe may or may not
* be persistent. If a pipe is activated as a persistent pipe, it cannot be changed while active to a non-
* persistent pipe, or vice versa.

* 5.8.1 CP-SVR Pipe Persistence Indicator

* To indicate support for persistent CP-SVR pipes, the DLUR and DLUS will set a DLUR/S capabilities
* indicator in CV X'51' which

* • when set by the DLUS indicates support for CP-SVR pipe persistence in general

* • when set by the DLUR indicates a request for CP-SVR pipe persistence for this particular pipe

* Therefore, a DLUS which can support persistent pipes will always set the indicator on while a DLUR
* which can support persistent pipes will only set the indicator on for those pipes it wishes to be per-
* sistent pipes.

* 5.8.2 Persistent CP-SVR Pipe Activation

* If the CP-SVR pipe persistence indicator is set by both the DLUS and the DLUR during pipe activation,
* then the CP-SVR pipe is considered persistent.

* DLUR processing will vary depending on whether or not there is a PU needing service and whether the
* DLUS supports (uplevel) or does not support (downlevel) pipe persistence.

* 5.8.2.1 Persistent CP-SVR Pipe Activation - PU Needing Service

* A DLUR which supports CP-SVR pipe persistence can activate a persistent CP-SVR pipe if there is an
* associated PU needing activation, just as long as there is no currently active pipe to that DLUS. In this
* case the DLUR will include a REQACTPU RU in the FID2 Encapsulation (X'1500') GDS variable it sends
* to the DLUS during DLUR-initiated CP-SVR pipe activation. The DLUR will send a GDS X'1500' with
* the RU and a CV X'51' indicating support of CP-SVR pipe persistence.

* ***5.8.2.1.1 PU Needing Service - Uplevel DLUS:*** A DLUS which supports CP-SVR pipe persistence will
* accept a FID2 Encapsulation (X'1500') GDS variable with a REQACTPU RU when the CV X'51' indi-
* cates support of CP-SVR pipe persistence, and the DLUS will respond with a GDS X'1500' with a
* RSP(REQACTPU) RU and a CV X'51' indicating support of CP-SVR pipe persistence.

* ***5.8.2.1.2 PU Needing Service - Downlevel DLUS:*** A DLUS which does not support CP-SVR pipe per-
* sistence will accept a FID2 Encapsulation (X′1500′) GDS variable with a REQACTPU RU when the CV
* X′51′ indicates support of CP-SVR pipe persistence, and the DLUS will respond with a GDS X′1500′
* with a RSP(REQACTPU) RU and a CV X′51′ indicating non-support of CP-SVR pipe persistence. At this
* point the DLUR can either:

* • accept the non-persistent pipe and continue session activation, or

* • it can terminate the pipe, using DLUR/S capabilities mismatch processing described in Figure 5-9
* on page 5-38 in 5.6.2.2.2, “DLUR-Detected Capabilities Mismatch (DLUS-DLUR Pipe)” on page 5-37,
* and attempt to activate a persistent pipe with a different DLUS.

* 5.8.2.2 Persistent CP-SVR Pipe Activation - No PU Needing Service

* A DLUR which supports CP-SVR pipe persistence can activate a persistent CP-SVR pipe even if there
* is no associated PU requiring activation, just as long as there is no currently active pipe to that DLUS.
* In this case the DLUR will not be required to include a REQACTPU RU in the FID2 Encapsulation
* (X′1500′) GDS variable it sends to the DLUS during DLUR-initiated CP-SVR pipe activation. The DLUR
* will send a GDS X′1500′ without an RU and with a CV X′51′ indicating support of CP-SVR pipe persist-
* ence.

* ***5.8.2.2.1 No PU Needing Service - Uplevel DLUS:*** A DLUS which supports CP-SVR pipe persistence
* will accept a FID2 Encapsulation (X′1500′) GDS variable without an RU when the CV X′51′ indicates
* support of CP-SVR pipe persistence, and the DLUS will respond with a GDS X′1500′ without an RU and
* with a CV X′51′ indicating support of CP-SVR pipe persistence.

* ***5.8.2.2.2 No PU Needing Service - Downlevel DLUS:*** A DLUS which does not support CP-SVR pipe
* persistence will, upon receipt of a FID2 Encapsulation (X′1500′) GDS variable without an RU, deacti-
* vate the CP-SVR pipe using UNBIND-initiated forced deactivation with the following sense data:

* • X′08A0 000C′ DLUS-DLUR Session Deactivation (Disruptive): The DLUS received a CP-SVR pipe
* activation request without a REQACTPU; DLUR should activate a DLUS-DLUR session with a dif-
* ferent DLUS if possible.

* (For more information about UNBIND-initiated CP-SVR pipe deactivation, see 5.6.2.1, “UNBIND-Initiated
* CP-SVR Pipe Deactivation” on page 5-30)

* A DLUR receiving an UNBIND with this sense data will understand that it attempted to activate a per-
* sistent pipe with a downlevel DLUS and will then try to activate a persistent pipe with another DLUS (if
* any).

* Downlevel DLUSs will be PTF′ed to send the X′08A0 000C′ sense data.

* 5.8.3 Persistent CP-SVR Pipe Deactivation

* When the number of current and pending SSCP-PU sessions associated with a persistent CP-SVR pipe
* reaches zero, the DLUS will leave the pipe active.

* The DLUS can deactivate a persistent CP-SVR pipe using UNBIND-initiated forced deactivation with dis-
* ruptive or nondisruptive sense data as appropriate.

* If the number of current and pending SSCP-PU sessions is zero, the DLUR can deactivate a persistent
* CP-SVR pipe using UNBIND-initiated forced deactivation with the following persistent sense data:

* • X′08A0 000D′ DLUS-DLUR Session Deactivation (Persistent): The DLUR is initiating deactivation of
* a persistent DLUS-DLUR session.

* (For more information about UNBIND-initiated CP-SVR pipe deactivation, see 5.6.2.1, “UNBIND-Initiated
* CP-SVR Pipe Deactivation” on page 5-30)

5.9 CP-SVR Pipe Performance Considerations

5.9.1 Pacing Considerations

It is recommended that DLUR products close their network priority pacing windows last (instead of the
more typical simultaneous closing of all pacing windows regardless of priority). Since SSCP-PU/LU
session activation and deactivation commands flow at network priority, they should be allowed to flow
as long as possible so that they can be used to help reduce network congestion.

5.10 Additional Error Cases

5.10.1 CP-SVR Pipe Limit Exceeded

DLUR products will have the option to limit the number of concurrently active CP-SVR pipes (what number will be selected as the limit will also be a product option). If a limit is implemented and reached, any attempts to activate additional CP-SVR pipes will be rejected by sending an UNBIND request with sense data X′0812 0000′.

5.10.2 GDS Variable Other Than X′1500′ Received After Receive_Encap Attach

If the Receive_Encap_Msg_TP is attached and the GDS variable received is not X′1500′, the DLUR will send an UNBIND request with sense data X′1001 0000′.

5.10.3 CPSVRMGR Session Over Non-TCP/IP LEN Connection

If a CPSVRMGR session is established over a LEN connection which is not a connection into a TCP/IP network, the DLUS will send an UNBIND request with sense data X′0877 0056′.

% – X.25 PVC

% – X.25 SVC

Additional DLC and physical interface types will be included at a later time.

New subfields in the range X'92'-X'F0' are being defined to carry the signaling information, e.g., addresses and dial digits, that is appropriate for each DLC and physical interface type. At present, subfields X'92' to X'96' have been defined.

13.1.8 DLUR/S Capabilities (X'51') Control Vector

This is a new control vector which describes the capabilities of the DLUS or DLUR to its partner.

For this, the initial level of DLUR/S support, there is:

- a one byte release field indicating Release 1 DLUR/S support (X'01'). Future releases will have their own values to aid in uplevel/downlevel support determination.
- a one byte indicator specifying whether the node is a NN (X'00') or an EN (X'01').
- a one byte indicator specifying whether the node is a DLUR (X'00') or a DLUS (X'01').
- a four byte Flow Reduction Sequence Number (FRSN)
- a one byte support indicator field, including:
  - 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
  - an indicator for limited DLUR support of ANS - when set, the DLUR only supports ANS=STOP; when not set, the DLUR supports both ANS=CONT and ANS=STOP
  - an indicator for limited DLUR multi-subnet support - when set, the DLUR responds to dependent LU searches with the OCR bit set; when not set, the DLUR responds to dependent LU searches with the OCR bit set
  - # an indicator for network name forwarding support - when set, the DLUR will, if requested by the # DLUS, forward Network Name control vectors on ACTPUs and ACTLUs to DLUR-served PUs # and their associated LUs; when not set, the DLUR will always remove Network Name control # vectors from ACTPUs and ACTLUs for DLUR-served PUs and LUs (for more details about # PU/LU network name forwarding, see 6.2.4, "PU/LU Network Name Forwarding" on page 6-25)
  - # an indicator for nondisruptive DLUS-DLUR session deactivation type X'08A0 000B' support - # when set, the DLUR will process an UNBIND with X'08A0 000B' as described in 5.6.2.1, # "UNBIND-Initiated CP-SVR Pipe Deactivation" on page 5-30; when not set, the DLUR cannot # process this type of UNBIND
  - # an indicator for FID2 Encapsulation (X'1500') GDS variable blocking support - when set, the # DLUR can, if also supported by the DLUS, send and receive BIUs on the CP-SVR pipe which # include more than one GDS X'1500' (for more details about FID2 Encapsulation blocking, see # 6.1.5, "Blocking Multiple FID2 Encapsulations" on page 6-7)
  - * an indicator for support of CP-SVR pipe persistence - when set, the sender supports CP-SVR * pipes which are active even without any associated pending or active SSCP-PU sessions; in * addition, if the sender is a DLUR, the sender requests persistence for this particular CP-SVR * pipe (for more information about persistent CP-SVR pipes, see 5.8, "Persistent CP-SVR Pipes" * on page 5-75)
- four reserved bytes (for future use)

C.1.23 DLUR/S Capabilities (X'51') Control Vector

C.1.23.1 DLUR/S Capabilities (X'51') Control Vector

The DLUR/S Capabilities control vector is exchanged by the dependent LU requester and the dependent LU server to describe their DLUR/S capabilities once a DLUS-DLUR session is activated.

DLUR/S Capabilities (X'51') Control Vector

Byte	Bit	Content
0– 1		Vector header; Key=X'51' (see C.1.30.3, "Substructure Encoding/Parsing Rules" on page C-66 and Table C-1 on page C-66)
2 – n		Vector Data
2		Dependent LU support level: X'01' Release 1
3		Node type: X'00' APPN network node X'01' APPN end node
4		DLUR/S node type: X'00' DLUR X'01' DLUS
5– 8		Flow-reduction sequence number: a value that identifies the latest Topology Database Update GDS variable received by the sender of the DLUR/S Capabilities control vector from the receiver of the DLUR/S Capabilities control vector
9		Support indicators (bit is set to 1 if the sender supports the function):
	0	RECEIVE_TDU_TP (X'22F0F0F4') service transaction program supported: The sending CP supports receipt of TDUs on this session.
	1	Limited DLUR auto network shutdown (ANS) support: The DLUR terminates all LU-LU sessions immediately when connectivity to the DLUS is lost.
	2	Limited DLUR multisubnet support: The DLUR will not respond to dependent-LU searches when the Owning CP Respond (OCR) bit in the Find is set to 1.
	3	Network name forwarding supported: The sender supports DLUR forwarding of Network Name control vectors on ACTPUs and ACTLUs to DLUR-served PUs and their associated LUs.
	4	Nondisruptive DLUS-DLUR session deactivation type X'08A0 000B' supported: The sender supports the use of UNBIND with sense data X'08A0 000B' for CP-SVR pipe deactivation.
	5	FID2 Encapsulation (X'1500') GDS variable blocking supported: The sender supports sending and receiving more than one FID2 Encapsulation (X'1500') GDS variable in a DLUS-DLUR session BIU.
	6	CP-SVR pipe persistence supported: The sender supports CP-SVR pipes which are active even without any asociated pending or active SSCP-PU sessions; in addition, if the sender is a DLUR, the sender requests persistence for this particular CP-SVR pipe.
	7	Reserved
10– 1 3		Reserved

C.2.12 08A0

08A0 Session Reset: An LU or PU is resetting an LU-LU session.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 The identified LU-LU session had to be deactivated because of a forced deactivation of the associated SSCP-PU or SSCP-LU session, for example, because of a received DACTPU, DACTLU, or DISCONTACT. The LU will send an UNBIND with a reason code of X'0A' (SSCP Gone).

0002 The LU or session connection manager (SCM) will send UNBIND with a reason code of X'0F' (Cleanup).

0003 A gateway node is cleaning up the session because a gateway SSCP has directed the gateway node (via NOTIFY) to deactivate the session, for example, a session setup error or session takedown failure has occurred. The gateway node will send UNBIND with a reason code of X'11' (Gateway Node Cleanup).

0004 Reversed FRSN Values: The value in the Last FRSN Sent field is greater than the value in the Current FRSN field in a received TDU GDS variable (no retry). The CP will send an UNBIND with a reason code of X'0F' (Cleanup)

0005 TDU Sent Out of Order: The value in the Last FRSN Sent field of the current TDU GDS variable is less than the value of the Current FRSN field in the TDU GDS variable that immediately preceded it, or is greater than it and the receiver cannot store the out-of-sequence value (no retry). The CP will send an UNBIND with a reason code of X'0F' (Cleanup)

0006 Invalid FRSN Value: In a CP Capabilities GDS variable, the adjacent node indicated receipt of a TDU with a FRSN value greater than the last one sent.

0007 DLUS-DLUR Session Deactivation (Disruptive): LU-LU sessions for DLUR-supported dependent LUs should be reset.

0008 DLUS-DLUR Session Deactivation (Nondisruptive): LU-LU sessions for DLUR-supported dependent LUs should not be reset.

0009 DLUS-DLUR Session Deactivation (Nondisruptive): Protocol violation detected; LU-LU sessions for DLUR-supported dependent LUs should not be reset.

000A DLUS-DLUR Session Deactivation (Nondisruptive): DLUR should wait for DLUS reactivation of the DLUS-DLUR session; LU-LU sessions for DLUR-supported dependent LUs should not be reset.

000B DLUS-DLUR Session Deactivation (Nondisruptive): DLUR should activate a DLUS-DLUR session with a different DLUS if possible; otherwise, it should wait for DLUS reactivation of the

DLUS-DLUR session; LU-LU sessions for DLUR-supported dependent LUs should not be reset.

000C DLUS-DLUR Session Deactivation (Disruptive): The DLUS received a CP-SVR pipe activation request without a REQACTPU; DLUR should activate a DLUS-DLUR session with a different DLUS if possible.

000D DLUS-DLUR Session Deactivation (Persistent): The DLUR is initiating deactivation of a persistent DLUS-DLUR session.

C.2.13 8019

8019 Routing Exception: A node is unable to perform the required routing function for a request.

Bytes 2 and 3 following the sense code contain sense code specific information. Settings allowed are:

0000 No specific code applies.

0001 Unknown Routing Group Name.

0002 Unknown Routing Group Name, Routing Element Name combination.

0003 Reserved

0004 No connection is available for level of service required. When this SNA report code is used in an SNA condition report, it is accompanied by a supplemental report containing the service parameter triplet (or combination of triplets) for which a connection could not be found.

0005 The Routing and Targeting Instructions GDS variable is required but is not present.

000C Upon receiving a Locate request for a DLUS-served dependent LU, the DLUS node has determined that at least one intersubnetwork TG on the path between the DLUS and the PLU was not between two extended border nodes.