16

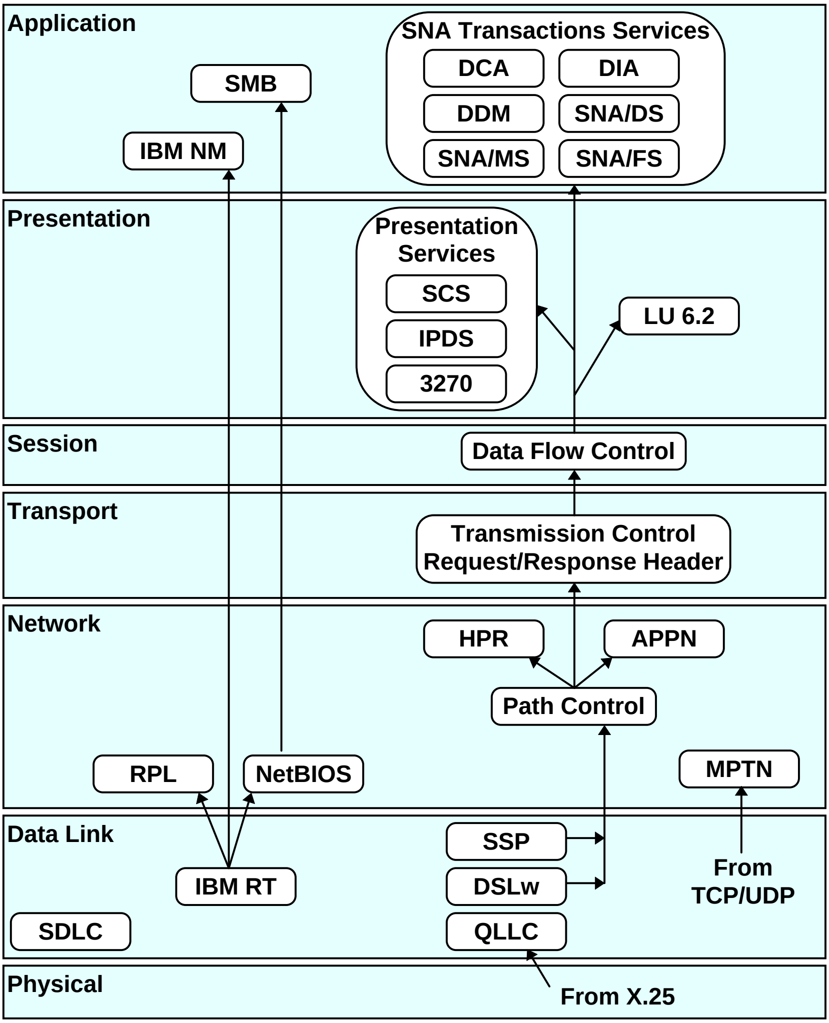
IBM Protocols

The Network Basic I/O System (NetBIOS), TCP/UDP version, was developed for the IBM PC LAN program to support communications between symbolically named stations and transfer of arbitrary data.

The Server Message Block (SMB) is a Microsoft presentation layer protocol providing file and print sharing functions for LAN Manager, VINES and other network operating systems. IBM NetBIOS manages the use of node names and transport layer connections for higher layer protocols such as SMB. The IBM suite includes the following protocols:

- NetBIOS: Network Basic I/O System.
- SMB: Server Message Block.
- SNA: Systems Network Architecture.
- HPR-APPN: High Performance Routing - Advanced Peer to Peer Network.
- NHDR: Network Layer Header.
- THDR: RTP Transport header.
- DLSw: Data Link Switching.

The following diagram illustrates the IBM protocol suite in relation to the OSI model:



IBM protocol suite in relation to the OSI model

NetBIOS

IBM Local Area Network Technical Reference 1990 4th edition

NetBIOS provides a communication interface between the application program and the attached medium. All communication functions from the physical layer through the session layer are handled by NetBIOS, the adapter support software, and the adapter card. A NetBIOS session is a logical connection between any two names on the network. It is usually encapsulated over LLC.

The format of the header is shown in the following illustration:

Len	XxEFFF	Command	Optional Data 1	Optional Data 2	Xmit/resp correlator	Dest name/ num	Source name/ num

NetBIOS header structure

Len

The length of the NETBIOS header.

XxEFFF

A delimiter indicating that subsequent data is destined for the NetBIOS function.

Command

A specific protocol command that indicates the type of function of the frame.

Data 1

One byte of optional data per specific command.

Data 2

Two bytes of optional data per specific command.

Xmit/response correlator

Used to associate received responses with transmitted requests. Transmit correlator is the value returned in a response to a given query. Response correlator is the value expected when the response to that message is received.

Destination name/num

In non-session frames this field contains the 16-character name. In session frames this field contains a 1 byte destination session number.

Source name/num

In non-session frames this field contains the 16-character source name. In session frames this field contains a 1 byte source session number.

SMB

ftp://ftp.microsoft.com/developr/drg/CIFS/
File Sharing Protocol 1987, SMB File Sharing Protocol Extensions, 1998, SMB File Sharing Protocol 1996

Server Message Block (SMB) is a Microsoft presentation layer protocol providing file and print sharing functions for LAN Manager, Banyan VINES and other network operating systems. IBM NetBIOS manages the use of node names and transport layer connections for higher layer protocols such as SMB.

SMB is used for sharing files, printers, serial ports and communications abstractions, such as named pipes and mail slots between computers. It is a client server request-response protocol. Clients connect to servers using TCP/IP. They can then send commands (SMBs) to the server that allows them to access shares, open files, etc., over the network.

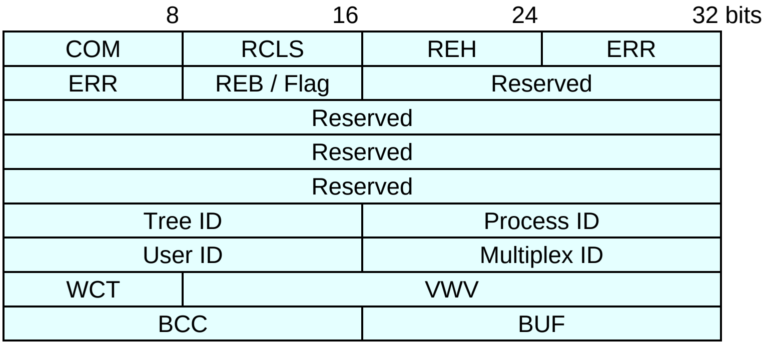
Many protocol variants have been developed. The first protocol variant was the Core Protocol, known also as PC Network Program 1.0. It handled a fairly basic set of operations that included:

- Connecting to and disconnecting from file and print shares.
- Opening and closing files.
- Opening and closing print files.
- Reading and writing files.
- Creating and deleting files and directories.
- Searching directories.
- Getting and setting file attributes.
- Locking and unlocking byte ranges in files.

There are several different versions and sub-versions of this protocol. A particular version is referred to as a dialect. When two machines first come into network contact they negotiate the dialect to be used. Different dialects can include both new messages as well as changes to the fields and semantics of existing messages in other dialects. Each server makes a set of resources available to clients on the network. A resource being shared may be a directory tree, named pipe, printer, etc. So far as clients are concerned, the server has no storage or service dependencies on any other servers; a client considers the server to be the sole provider of the resource being used.

The SMB protocol requires server authentication of users before file accesses are allowed, and each server authenticates its own users. A client system must send authentication information to the server before the server will allow access to its resources.

The general format of the header is shown in the following illustration:



SMB frame structure

COM

Protocol commands. The following are possible commands within SMB frames:

Command	***Description.***
[bad command]	Invalid SMB command.
[bind (UNIX)]	Obtain file system address for file.
[cancel forward]	Cancel server recognition of name.
[change/check dir]	Change to directory or check path.
[change group]	Change group association of user.
[change password]	Change password of user.
[close file]	Close file handle and flush buffers.
[close spoolfile]	Close print buffer file.
[consumer logon]	Log on with consumer validation.
[copy file]	Copy file to specified path.
[copy new path]	Copy file to new path name.
[create & bind]	Create file and get file system address.
[create directory]	Create new directory.
[create file]	Create new or open existing file.
[delete dir]	Delete the specified directory.
[delete file]	Delete the specified file.
[echo]	Request echo from server.

Command	*Description.*
[find & close]	Search for file and close directory (UNIX).
[find & close /2]	Search for file and close directory (OS/2).
[find first file]	Find first matching file (OS/2).
[find unique]	Search directory for specified file.
[flush file]	Flush all file buffers to disk.
[fork to PID]	Provide same access rights to new process.
[forward name]	Cause server to accept messages for name.
[get access right]	Get access rights for specified file.
[get exp attribs]	Get expanded attributes for file (OS/2).
[get unix attribs]	Get expanded attributes for file (UNIX).
[get file attribs]	Get attributes for specified file.
[get file queue]	Get print queue listing.
[get group info]	Get logical group associations.
[get machine name]	Get machine name for block messages.
[get pathname]	Get path of specified handle.
[get resources]	Get availability of server resources.
[get server info]	Get total and free space for server disk.
[get user info]	Get logical user associations.
[IOCTL]	Initiate I/O control for DOS-OS/2 devices.
[IOCTL next]	Initiates subsequent I/O control for DOS-OS/2 devices.
[IOCTL (UNIX)]	I/O control for UNIX-Xenix devices.
[link file]	Make an additional path to a file.
[lock and read]	Lock and read byte range.
[lock bytes]	Lock specified byte range.
[lock/unlock & X]	Lock/unlock bytes and execute next command.
[logoff & execute]	Log off and execute next command.
[mail announce]	Query availability of server nodes.
[mailslot message]	Mail slot transaction message.
[make/bind dir]	Make dir and get file system address.
[make temp file]	Make temporary data file.
[make new file]	Make new file only if it does not exist.
[make node]	Make file for use as a device.
[move file]	Move file to specified path (OS/2).
[move new path]	Move file to specified path (UNIX/Xenix).
[multi-block data]	Send data for multi-block message.
[multi-block end]	Terminate multi-block message.
[multi-block hdr]	Send header for multi-block message.
[named pipe call]	Open, write, read, or close named pipe.
[named pipe wait]	Wait for named pipe to become ready.

Command	*Description.*
[named pipe peek]	Look at named pipe data.
[named pipe query]	Query named pipe handle modes.
[named pipe set]	Set named pipe handle modes.
[named pipe attr]	Query named pipe attributes.
[named pipe R/W]	Named pipe read/write transaction.
[named pipe read]	Raw mode named pipe read.
[named pipe write]	Raw mode named pipe write.
[negotiate protoc]	Negotiate SMB protocol version.
[newfile & bind]	Make new file and get file system address.
[notify close]	Close handle used to monitor file changes.
[open file]	Open specified file.
[open & execute]	Open specified file and execute next command.
[open spoolfile]	Open specified print buffer file.
[process exit]	Terminate consumer process.
[read & execute]	Read file and execute next command.
[read and hide]	Read directory ignoring hidden files.
[read block mplex]	Read block data on multiplexed connection.
[read block raw]	Read block data on unique connection.
[read block sec/r]	Read block secondary response.
[read check]	Check file accessibility.
[read from file]	Read from specified file.
[read w/options]	Read from file with specified options.
[rename file]	Rename the specified file to a new name.
[reserve resourcs]	Reserve resources on the server.
[search dir]	Search directory with specified attribute.
[seek]	Set file pointer for handle.
[send broadcast]	Send a one block broadcast message.
[session setup]	Log-in with consumer-based authentication.
[set exp attrib]	Set expanded file attributes (OS/2).
[set unix attribs]	Set expanded file attributes (UNIX/Xenix).
[set file attribs]	Set normal file attributes.
[single block msg]	Send a single block message.
[transaction next]	Subsequent name transaction.
[tree & execute]	Make virtual connection and execute next command.
[tree connect]	Make a virtual connection.
[tree disconect]	Detach a virtual connection.
[unbind]	Discard file system address binding.
[unlock bytes]	Release a locked byte range.
[write & close]	Write to and close specified file handle.

Command	*Description.*
[write & execute]	Write to file and execute next command.
[write & unlock]	Write to and unlock a byte range.
[write block raw]	Write block data on unique connection.
[write block mplx]	Write block data on multiplexed connection.
[write block sec]	Write block secondary request.
[write complete]	Terminate a write block sequence.
[write spoolfile]	Write to the specified print buffer.
[write to file]	Write to the specified file handle.
[X2 open file]	Open file.
[X2 find first]	Find first file.
[X2 find next]	Find next file.
[X2 query FS]	Get file system information.
[X2 set FS info]	Set file system information.
[X2 query path]	Get information on path.
[X2 set path]	Set path information.
[X2 query file]	Get file information.
[X2 set info]	Set file information.
[X2 FS control]	File system control information.
[X2 IOCTL]	I/O control for devices.
[X2 notify]	Monitor file for changes.
[X2 notify next]	Subsequent file monitoring.
[X2 make dir]	Make directory.

RCLS

Error Class. The second field of the decoded SMB protocol contains the error class and error code for each frame as in *E=1/22*, where *1* is the error class and *22* is the error code. The error class identifies the source of the error as shown in the following table:

Error Class	*Name*	*Source of Error*
0		No error, or error was handled by system.
1	ERRDOS	Server operating system.
2	ERRSRV	Server network file manager.
3	ERRHRD	System or device.
4	ERRXOS	Extended operating system.
225-227	ERRRMX	RMX operating system.
255	ERRCMD	Invalid SMB commands.

REH

Reserved.

ERR

Error messages. The following is a list of SMB error messages:

Error Message	*Description*
{Access denied}	Unable to service request.
{Access list full}	Access control list full.
{Bad attrib mode}	Invalid attributes specified.
{Bad disk request}	Disk command invalid.
{Bad drive spec}	Specified drive invalid.
{Bad environment}	Environment invalid.
{Bad EXE file}	Bad executable file format.
{Bad file access}	Invalid access to read only file.
{Bad file ID}	File handle invalid.
{Bad filespec}	Path name invalid.
{Bad format}	Format invalid.
{Bad function}	Function not supported.
{Bad I/O data}	Data invalid on server I/O device.
{Bad math argument}	Math argument invalid.
{Bad media type}	Unknown media type.
{Bad memory block}	Memory block address invalid.
{Bad open mode}	Open mode invalid.
{Bad permissions}	Specified permissions invalid.
{Bad print FID}	Print file ID invalid.
{Bad print request}	Printer device request invalid.
{Bad reqst length}	Bad request structure length.
{Bad semaphore}	Semaphore identifier invalid.
{Bad SMB command}	SMB command invalid.
{Bad Tree ID}	Tree ID invalid.
{Bad User ID}	User ID invalid.
{Bad user/passwrd}	Bad password or user name.
{Bad wait done}	Wait done for unwaited process.
{Continue in MPX}	Continue in block multiplexed mode.
{Can’t delete dir}	Cannot delete current directory.
{Can’t init net}	Network cannot be initialized.
{Can’t mount dev}	Device cannot be mounted.
{Can’t RAW, do MPX}	Cannot use raw blocks, use multiplexed.
{Can’t ren to vol}	Attempt to rename across volumes failed.
{Can’t support RAW}	Cannot support raw block access.
{Can’t write dir}	Attempt to write on a directory failed.
{Command not recvd}	Initial command not received.
{CRC data error}	Data CRC error on device.
{Dev out of space}	Device out of space.

Error Message	*Description*
{Device is remote}	Referenced device remote.
{Dir not found}	Directory not found.
{Disk write error}	Disk write fault.
{Disk read error}	Disk read fault.
{Disk seek error}	Disk seek error.
{Drive not ready}	Drive not ready.
{Dup filename}	File name already exists.
{EOF on printer}	End of file found on print queue dump.
{Err buffered}	Error message buffered.
{Err logged}	Error message logged.
{Err displayed}	Error message displayed.
{File not found}	Filespec not found.
{File too big}	Maximum file size exceeded.
{Gen disk failure}	General disk failure.
{Insuf acc rights}	Insufficient access rights.
{Invalid name}	Invalid name supplied on tree connect.
{Invalid pipe}	Invalid pipe specified.
{Lock conflict}	Lock/Unlock conflicts with other locks.
{Memory blks lost}	Memory control blocks destroyed.
{More data coming}	Cannot terminate; more data coming.
{Need block device}	File used where a block device needed.
{Need data file}	Must specify data file.
{No FCBs available}	Out of file control blocks.
{No more files}	No more matching files found.
{No proc to pipe}	No process available to pipe.
{No read process}	Write to a pipe with no read processes.
{No resources}	Server out of resources.
{No room f/message}	No room to buffer message.
{Not a directory}	Must specify directory.
{Not receiving}	Not receiving messages.
{No semaphores}	Semaphore not available.
{OK}	SMB command completed successfully.
{Out of disk space}	Print queue out of disk space.
{Out of handles}	Too many open files.
{Out of memory}	Insufficient memory on server.
{Out of paper}	Printer out of paper.
{Pipe is busy}	Pipe process busy; wait.
{Pipe is closing}	Terminating pipe process.
{Print Q full}	Print file queue table full.
{Proc table full}	Server process table full.

Error Message	*Description*
{Rem I/O error}	Remote I/O error.
{Sector not found}	Sector not found.
{Seek on pipe}	Seek was issued to a pipe.
{Server error}	General server error.
{Server paused}	Server paused.
{Share buffer out}	Share buffer out of space.
{Share conflict}	Share conflicts with existing files.
{Syntax error}	Syntax error in path name.
{Sys call intruptd}	Interrupted system call.
{Table overflow}	Internal table overflow.
{Terminal needed}	Terminal device required.
{Timed out}	Operation has run out of time.
{Too many links}	Too many links.
{Too many names}	Too many remote user names.
{Too many UIDs}	User ID limit exceeded.
{Unit unknown}	Unknown unit.
{Unknown error}	Non-specific error.
{Unknwn process}	No such process.
{Write protected}	Write on write-protected diskette.
{Wrong diskette}	Wrong diskette inserted in drive.

REB/Flag

Reserved. This field is associated with the Core protocol only. The flag field appears in protocol versions later then the Core Protocol.

Tree ID

Uniquely identifies a file sharing connection between consumer and server where this protocol uses a server-based file protection.

Process ID

Identifies a specific consumer process within a virtual connection.

User ID

Used by the server to verify the file access permissions of users where consumer-based file protection is in effect.

Multiplex ID

Reserved for multiplexing multiple messages on a single virtual circuit (VC). A response message will always contain the same value as the corresponding request message. Only one request at a time may be outstanding on any VC.

WCT

Number of parameter words.

VWV

Variable number of words of parameter.

BCC

Number of bytes of data following.

BUF

Variable number of data bytes.

SDLC

IBM SNA Formats GA27-3136-10 1989-06

The SDLC (Synchronous Data Link Control) protocol was developed by IBM to be used as the layer 2 of the SNA hierarchical network. SNA data is carried within the information field of SDLC frames. The format of a standard SDLC frame is as follows:

Link Header				Link Trailer	
Flag	Address field	Control field	Information	FCS	Flag

SDLC frame format

Flag

The value of the flag is always (0x7E). In order to ensure that the bit pattern of the frame delimiter flag does not appear in the data field of the frame (and therefore cause frame misalignment), a technique known as Bit Stuffing is used by both the transmitter and the receiver.

Address field

The first byte of the frame after the header flag is known as the Address Field. SDLC is used on multipoint lines and it can support as many as 256 terminal control units or secondary stations per line. The address field defines the address of the secondary station which is sending the frame or the destination of the frame sent by the primary station.

Control field

The field following the Address Field is called the Control Field and serves to identify the type of the frame. In addition, it includes sequence numbers, control features and error tracking according to the frame type.

Every frame holds a one bit field called the Poll/Final bit. In SDLC this bit signals which side is 'talking', and provides control over who will speak next and when. When a primary station has finished transmitting a series of frames, it sets the Poll bit, thus giving control to the secondary station. At this time the secondary station may reply to the primary station. When the secondary station finishes transmitting its frames, it sets the Final bit and control returns to the primary station.

Modes of operation

In SDLC there is the notion of primary and secondary stations, defined simply as the initiator of a session and its respondent. The primary station sends commands and the secondary station sends responses.

SDLC operates in Normal Response Mode (NRM). This mode is totally master/slave meaning that only one station may transmit frames at any one time (when permitted to do so). This mode is signified by the SNRM(E) frame. The primary station initiates the session and sends commands. The secondary station sends responses. Full polling is used for all frame transmissions.

FCS

The Frame Check Sequence (FCS) enables a high level of physical error control by allowing the integrity of the transmitted frame data to be checked. The sequence is first calculated by the transmitter using an algorithm based on the values of all the bits in the frame. The receiver then performs the same calculation on the received frame and compares its value to the CRC.

Window size

SDLC supports an extended window size (modulo 128) where the number of possible outstanding frames for acknowledgement is raised from 8 to 128. This extension is generally used for satellite transmissions where the acknowledgement delay is significantly greater than the frame transmission times. The type of the link initialization frame determines the modulo of the session and an "E" is added to the basic frame type name (e.g., SNRM becomes SNRME).

Frame types

The following are the Supervisory Frame Types in SDLC:

RR Information frame acknowledgement and indication to receive more.
REJ Request for retransmission of all frames after a given sequence number.
RNR Indicates a state of temporary occupation of station (e.g., window full).

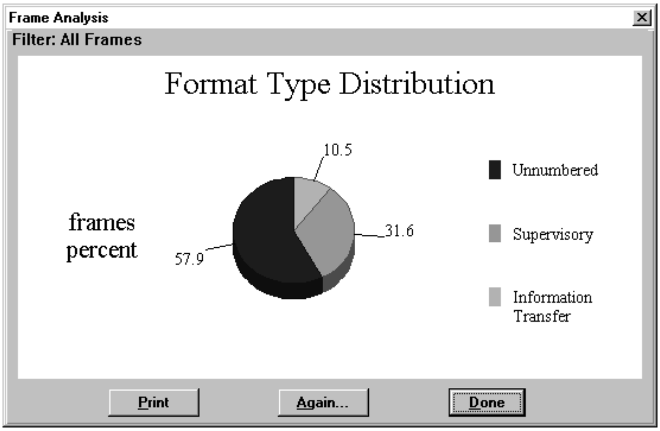
The following are the Unnumbered Frame Types in SDLC:

DISC Request disconnection.
UA Acknowledgement frame.
DM Response to DISC indicating disconnected mode.

FRMR	Frame reject.
CFGR	Configure.
TEST	Sent from primary to secondary and back again.
BCN	Beacon.
SNRM	Initiator for normal response mode. Full master/slave relationship.
SNRME	SNRM in extended mode.
RD	Request disconnect.
RIM	Secondary station request for initialization after disconnection.
SIM	Set initialization mode.
UP	Unnumbered poll.
UI	Unnumbered information. Sends state information/data.
XID	Identification exchange command.

There is one Information Frame Type in SDLC:

Info	Information frame.



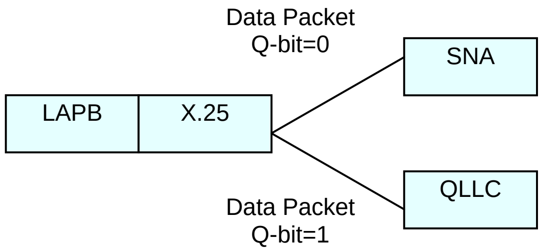
Filter: All Frames

Graph displaying distribution of SDLC frames by format type

QLLC

QLLC is a standard developed for interconnecting SNA LANs over packet switched WANs with X.25. The SDLC header and trailer is stripped off and replaced by similar fields of LAPB before transmission over the network. The standard also defines additional control bytes used to allow the receiving end of the network to reconstruct the original SDLC frame. The SNA information is passed over the network within the X.25 data packet.

The following diagram represents SNA data and QLLC control frames within the X.25 data packet:



SNA data and QLLC control frames are determined by the value of the Q-bit within the X.25 packet header.

QLLC Frame Types

QRR	Receive Ready.
QDISC	Disconnect.
QUA	Unnumbered Acknowledgement.
QDM	Disconnect Mode.
QFRMR	Frame Reject.
QTEST	Test.
QRD	Request Disconnect.
QXID	Exchange Identification.
QSM	Set Mode.

SNA

The Systems Network Architecture (SNA) was introduced by IBM in 1974 in order to provide a framework for joining together the multitude of mutually incompatible IBM products for distributed processing. SNA was one of the first communications architectures to use a layered model, which later became the basis for the OSI model.

The SNA is a hierarchical network that consists of a collection of machines called nodes. There are four types of nodes; Type 1 (terminals), Type 2 (controllers and machines that manage terminals), Type 4 (front-end processors and machines that take some load off the main CPU) and Type 5 (the main host).

Each node has at least one Network Addressable Unit (NAU). The NAU enables a process to use the network by giving it an address. A process can then reach and be reached by other NAUs.

An NAU can be one of three types; an LU (Logical Unit), a PU (Physical Unit) or an SSCP (System Services Control Point). Usually there is one SSCP for each Type 5 node and none in the other nodes.

SNA distinguishes five different kinds of sessions: SSCP-SSCP, SSCP-PU, SSCP-LU, LU-LU and PU-PU.

The SSCP (PU Type 5) is usually implemented in IBM mainframe machines which use channels to connect to control devices such as disks, tapes and communication controllers. These are high speed communications links (up to 17 Mbps).

The communication controller (the FEP, Front End Processor, PU type 4) is used to connect low speed SDLC lines. All together the SSCPs, FEPs, channels and SDLC lines connecting them create the SNA backbone. Using SDLC, the FEPs also connect Token Ring LAN or X.25 links and other types of SNA devices such as cluster controllers and RJE stations. These are PU type 2/2.1 devices and are used to manage LUs which are the endpoint of SNA network - elements such as the display terminal (the 3270 family).

SNA frames have the following format:

Transmission header (TH)	Request / response header (RH)	Request / response unit (RU)

SNA frame structure

Transmission header

The TH field contains the Format Identifier value (FID). This value corresponds to the type of communication session and the environment in which it is used.

FID2 is the format used between a T4 or T5 node and an adjacent T2.0 or T2.1 node, or between adjacent T2.1 nodes. FID3 is used on links to a PU T1 (such as AS/400 controllers). FID4 is used on links between PU T4s.

The TH field also contains a mapping field (MPF) which indicates whether the frame is a complete SNA frame (containing TH, RH and RU) or just a segment. When the SNA frame is too large to be sent as one frame, it is divided into several segments (first, middle, last or whole). The first segment includes a TH (indicating that it is the first), an RH and the beginning of the RU. Other segments (middle and last) contain a TH (identical to the one of the first except for the MPF field) and the remainder of the RU.

Request/response header

The RH field denotes the SNA category of the frame, the format of the RU, whether requests are chained together, bracket indicators, pacing information and various other SNA frame properties.

Request/response unit

The RU contains the ‘user data’ that one LU sends to its session partner or a special SNA frame. A field within the RH distinguishes between cases and several classes of SNA frames. There are three categories of SNA frames: NS (function management data), DFC (data flow control) and SC (session control).

SNA TH0 & TH1

SNA TH0 and TH1 correspond to the FID header 0 and 1 respectively.

The format of the packet is shown in the following illustration:

4	6	7	8 bits
FID	MPF	EFI	
DAF (2 bytes)			
OAF (2 bytes)			
SNF (2 bytes)			
DCF (2 bytes)			

SNA TH0, SNA TH1 packet structure

FID

Format Identification: 0=FID 0, 1=FID 1.

MPF

Mapping field:

0	Middle segment of a BIU
1	Last segment of a BIU
2	First segment of a BIU
3	Whole BIU

EFI

Expedited flow indicator:

0	Normal flow
1	Expedited flow

DAF

Destination address field. Network address denoting the BIU's destination network addressable unit (NAU).

OAF

Origin address field. Network address denoting the originating NAU.

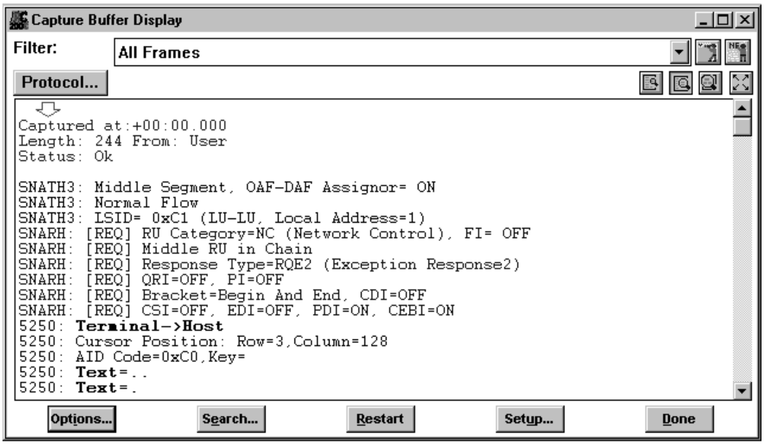
SNF

Sequence number field. Numerical identifier for the associated BIU.

DCF

Data count field. A binary count of the number of bytes in the BIU if the BIU segment is associated with the transmission header.

5250 is located in frames with an RH field as may be viewed in the multi-protocol view of the capture buffer.



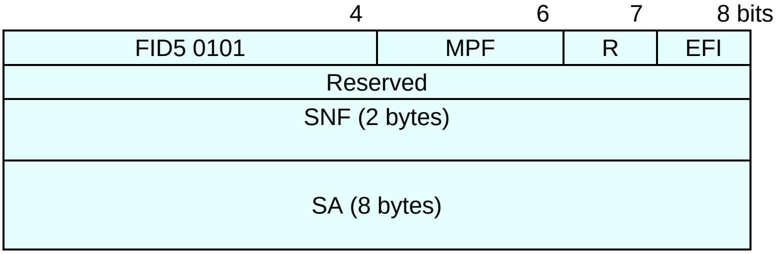
```
Captured at:+00:00.000
Length: 244 From: User
Status: Ok

SNATH3: Middle Segment, OAF-DAF Assignor= ON
SNATH3: Normal Flow
SNATH3: LSID= 0xC1 (LU-LU, Local Address=1)
SNARH: [REQ] RU Category=NC (Network Control), FI= OFF
SNARH: [REQ] Middle RU in Chain
SNARH: [REQ] Response Type=RQE2 (Exception Response2)
SNARH: [REQ] QRI=OFF, PI=OFF
SNARH: [REQ] Bracket=Begin And End, CDI=OFF
SNARH: [REQ] CSI=OFF, EDI=OFF, PDI=ON, CEBI=ON
5250: Terminal->Host
5250: Cursor Position: Row=3,Column=128
5250: AID Code=0xC0,Key=
5250: Text=..
5250: Text=.
```

5250 as viewed in the RH field of the SNA frame

SNA TH5

SNA TH 5 is the FID 5 header.

The format of the packet is shown in the following illustration:



4	6	7	8 bits
FID5 0101	MPF	R	EFI
Reserved			
SNF (2 bytes)			
SA (8 bytes)			

FID 5 header structure

FID 5

The value of this field is 0101.

MPF

Mapping field.

R

Reserved bit.

EFI

Expedited flow indicator (1 bit).

SNF

Sequence number field.

SA

Session address.

HPR-APPN

HPR network is an extension of the SNA network. HPR (High Performance Routing) is an extension of the base-APPN that provides some key advancements. These new functions include:

- Non-disruptive path switching.
- Better utilization of high-speed communication paths.
- An advanced congestion control methodology.
- Additional functionality provided by two new components: Rapid Transport Protocol (RTP) and Automatic Network Routing (ANR). These components provide the added functionality exhibited by HPR nodes.

NHDR

The packet transported along an RTP connection has a specific format. It consists of 3 components. NHDR, THDR and data. The Network Layer Header (NHDR) begins the frame used by RTP (Rapid Transport Protocol) nodes. It provides addressing for the packet as it transverses the HPR network. The components of this header include the transmission priority and the ANR (Automatic Network Routing) labels. NHDR consists of some indicators that identify the packet as a network layer packet.

The format of the header is shown in the following illustration:

<table>
<tr><th>1</th><th>2</th><th>3</th><th>4 bits</th></tr>
<tr><td colspan="3">SM</td><td>TPF</td></tr>
<tr><td>TPF</td><td colspan="3">Function type</td></tr>
<tr><td>Function type</td><td>TSP</td><td>Slowdown1</td><td>Slowdown2</td></tr>
<tr><td colspan="4">ANR / function routing field (1 or 2 bytes)</td></tr>
</table>

NHDR header structure

SM

Switching mode may have the following values:

5	Function routing.
6	Automatic network routing.

TPF

Transmission priority field may have the following values:

0	Low (L).
1	Medium (M).
2	High (H).
3	Network (N).

Function type (for switching mode 5)

Function type of 1 indicates logical data link control.

TSP

Time-sensitive packet indicator.

Slowdown 1 and 2

This field indicates when ever a minor (slowdown 1) or significant (slowdown 2) congestion condition exists. Possible values are:

0	Does not exist.
1	Exists.

ANR routing field (for SM = 6)

A string of ANR labels 1 or 2 bytes long. The string ends with 0xFF.

Function routing field (for SM = 5)

A 2 byte function routing address (FRA) followed by the value 0xFF.

THDR

THDR is the RTP Transport header. It is used by the RTP endpoints to provide correct processing of the packet. It is used for communication between the endpoints and to identify the RTP connection.

The format of the header is shown in the following illustration:

TCID assignor (7 bytes)
Connection setup (1 bit)
Start-of-message indicator (1 bit)
End-of-message indicator (1 bit)
Status requested indicator (1 bit)
Respond ASAP indicator (1 bit)
Retry indicator (1 bit)
Last message indicator (1 bit)
Connection qualifier field indicator (2 bits)
Optional segments present indicator (1 bit)
Data offset (2 bytes)
Data length (2 bytes)
Byte sequence number (4 bytes)
Control vector 05
Optional segments

THDR header structure

TCID assignor

Transport connection identifier. There are 2 possible values:

0	TCID was assigned by the receiving RTP partner.
1	TCID was assigned by the sending RTP partner.

Connection setup

0	Presented.
1	Not presented.

Start of message indicator

0	Not start of message.
1	Start of message.

End of message indicator

0	Not end of message.
1	End of message.

Status requested indicator

0	Receiver need not reply with a status segment.
1	Receiver must reply with a status segment.

Respond ASAP indicator

1	Sender will retransmit reply ASAP.

Retry indicator

0	Sender will retransmit this packet.

Connection qualifier field indicator

0	None presented.
1	Originator.

Optional segments present indicator

0	Not presented.
1	Presented.

Byte sequence number

Sequence number of the first byte of the data field.

Optional segments

If present the optional segment can contain one or more of the following segments:

0x0E	Status segment.
0x0D	Connection Setup segment.
0x10	Connection Identifier Exchange segment.
0x14	Switching Information segment.
0x22	Adaptive Rate-Based segment.
0x12	Connection Fault segment.
0x0F	Client Out-of-band Bits segment.

The structure of each segment is as follows:

Byte	***Content***
0	Segment length/4.
1	Segment type.
2	Segment data.

Each segment may include control vectors. Supported control vectors are:

0x00	Node identifier Control Vector.
0x03	Network ID Control Vector.
0x05	Network Address Control Vector.
0x06	Cross-Domain Resource Manager Control Vector.
0x09	Activation Request/Response Sequence Identifier Control Vector.
0x0E	Network Name Control Vector.
0x10	Product Set ID Control Vector.
0x13	Gateway Support Capability Control Vector.
0x15	Network-Qualified Address Pair Control Vector.
0x18	SSCP Name Control Vector.
0x22	XID Negotiation Error Control Vector.
0x26	NCE Identifier Control Vector.
0x28	Topic Identifier Control Vector.
0x32	Short-Hold Mode Control Vector.
0x39	NCE Instant Identifier.
0x46	TG Descriptor Control Vector.
0x60	Fully qualified PCID Control Vector.
0x61	HPR Capabilities Control Vector.
0x67	ANR Path Control Vector.
0xFE	Control Vector Keys Not Recognized Control Vector.

DLSw

IETF RFC 1434 http://www.cis.ohio-state.edu/htbin/rfc/rfc1434.html
RFC 1795 http://www.cis.ohio-state.edu/htbin/rfc/rfc1795.html
RFC 2166 http://www.cis.ohio-state.edu/htbin/rfc/rfc2166.html

Data Link Switching (DLSw) is a forwarding mechanism for the IBM SNA (Systems Network Architecture) and IBM NetBIOS (Network Basic Input Output Services) protocols. Over IP networks, DLSw does not provide full routing, but instead provides switching at the SNA Data Link layer (i.e., layer 2 in the SNA architecture) and encapsulation in TCP/IP for transport over the Internet.

A Data Link Switch (abbreviated also as DLSw) can support SNA (Physical Unit (PU) 2, PU 2.1 and PU 4) systems and optionally NetBIOS systems attached to IEEE 802.2 compliant Local Area Networks, as well as SNA (PU 2 (primary or secondary) and PU2.1) systems attached to IBM Synchronous Data Link Control (SDLC) links. For the latter case, the SDLC attached systems are provided with a LAN appearance within the Data Link Switch (each SDLC PU is presented to the SSP protocol as a unique MAC/SAP address pair). For Token Ring LAN attached systems, the Data Link Switch appears as a source-routing bridge. Token Ring Remote systems that are accessed through the Data Link Switch appear as systems attached to an adjacent ring. This ring is a virtual ring that is manifested within each Data Link Switch.

There are two message header formats exchanged between data link switches: Control and Information. These two message formats are as follows:

8	16	Octets
Version number	Header length (=16)	0-1
Message length		2-3
Remote data link correlator		4-7
Remote DLC port ID		8-11
Reserved field		12-13
Message type	Flow control byte	14-15

DLSw information message structure

8	16	Octets
Version number	Header length (=72)	0-1
Message length		2-3
Remote data link correlator		4-7
Remote DLC port ID		8-11
Reserved field		12-13
Message type	Flow control byte	14-15
Protocol ID	Header number	16-17
Reserved		18-19
Largest frame size	SSP flags	20-21
Circuit priority	Message type	22-23
Target MAC address		24-29
Origin MAC address		30-35
Origin link SAP	Target link SAP	36-37
Frame direction	Reserved	38-39
Reserved		40-41
DLC header length		42-43
Origin DLC port ID		44-47
Origin data link correlator		48-51
Origin data link correlator		52-55
Origin transport ID		56-59
Target DLC port ID		60-63
Target data link correlator		64-67
Target Transport ID		68-69
Reserved		70-71

DLSw control message structure

Version number

Set to 0x31 (ASCII 1) to indicate DLSw version 1.

Header length

Set to 0x48 for control messages and 0x10 for information and Independent Flow Control messages.

Message length

Specifies the number of bytes within the data field following the header.

Remote data link correlator / remote DLC port ID

The contents of the DLC and DLC Port ID have local significance only. The values received from a partner DLSw must not be interpreted by the DLSw that receives them and should be echoed as is to a partner DLSw in subsequent messages.

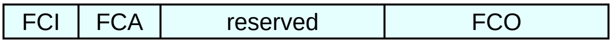
Message type

The following message types are available:

CANUREACH_ex	Can U Reach Station-explorer
CANUREACH_cs	Can U Reach Station-circuit start
ICANREACH_ex	I Can Reach Station-explorer
ICANREACH_cs	I Can Reach Station-circuit start
REACH_ACK	Reach Acknowledgment
DGRMFRAME	Datagram Frame
XIDFRAME	XID Frame
CONTACT	Contact Remote Station
CONTACTED	Remote Station Contacted
RESTART_DL	Restart Data Link
DL_RESTARTED	Data Link Restarted
ENTER_BUSY	Enter Busy
EXIT_BUSY	Exit Busy
INFOFRAME	Information (I) Frame
HALT_DL	Halt Data Link
DL_HALTED	Data Link Halted
NETBIOS_NQ_ex	NETBIOS Name Query-explorer
NETBIOS_NQ_cs	NETBIOS Name Query-circuit setup
NETBIOS_NR_ex	NETBIOS Name Recognized-explorer
NETBIOS_NR_cs	NETBIOS Name Recog-circuit setup
DATAFRAME	Data Frame
HALT_DL_NOACK	Halt Data Link with no Ack
NETBIOS_ANQ	NETBIOS Add Name Query
NETBIOS_ANR	NETBIOS Add Name Response
KEEPALIVE	Transport Keepalive Message
CAP_EXCHANGE	Capabilities Exchange
IFCM	Independent Flow Control Message
TEST_CIRCUIT_REQ	Test Circuit Request
TEST_CIRCUIT_RSP	Test Circuit Response

Flow control byte

Format of the flow control is as follows:



Flow control format

FCI	Flow control indicatory.
FCA	Flow control ack.
FCO	Flow control operator bits.

Protocol ID

Set to 0x42, indicating a decimal value of 66.

Header number

Set to 0x01, indicating a value of one.

Largest frame size

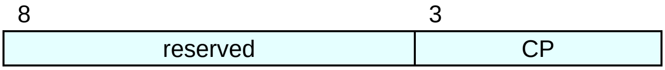
Carries the largest frame size bits across the DLSw connection to ensure that the two end-stations always negotiate a frame size to be used on a circuit that does not require the origin and target DLSw partners to re-segment frames.

SSP flags

Contain additional information related to the SSP message.

Circuit priority

Circuit priority is only valid for CANUREACH_cs, ICANREACH_cs, and REACH_ACK frames.



Circuit priority format

The value of the Circuit Priority bits (CP) can be as follows:

000	Unsupported, meaning that the Data Link Switch that originates the circuit does not implement priority.
001	Low Priority.
010	Medium Priority.
011	High Priority.
100	Highest Priority.

101 to 111 are reserved for future use.

Target/Origin MAC address
Origin/Target link SAP

Each attachment address is represented by the concatenation of the MAC address (6 bytes) and the LLC address (1 byte). Each attachment address is classified as either Target, in the context of the destination MAC/SAP addresses of an explorer frame sent in the first frame used to establish a circuit, or Origin, in the context of the source MAC/SAP addresses. All MAC addresses are expressed in non-canonical (Token-Ring) format.

Frame direction

Set to 0x01 for frames sent from the origin DLSw to the target DLSw, and is set to 0x02 for frames sent from the target DLSw to the origin DLSw.

DLC header length

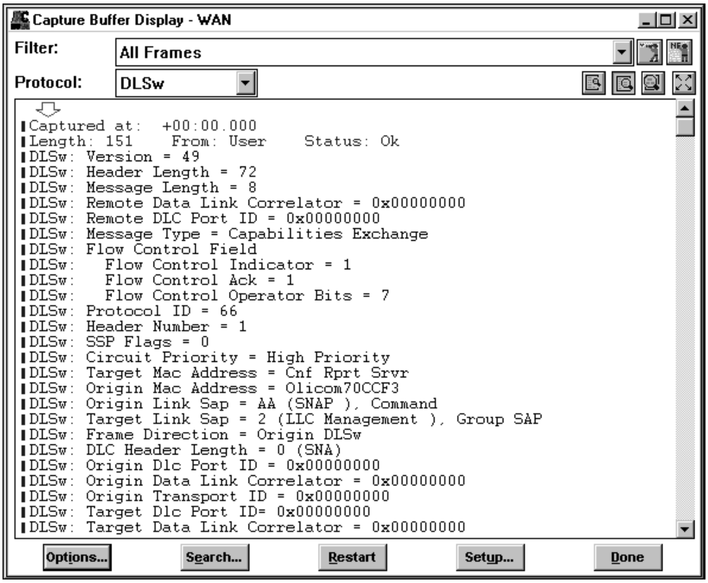
Set to zero for SNA and is set to 0x23 for NetBIOS datagrams, indicating a length of 35 bytes. This includes the Access Control (AC) field, the Frame Control (FC) field, Destination MAC Address (DA), the Source MAC Address (SA), the Routing Information (RI) field (padded to 18 bytes), the Destination link SAP (DSAP), the Source link SAP (SSAP), and the LLC control field (UI).

Origin/Target DLC port ID
Origin/Target data link correlator

An end-to-end circuit is identified by a pair of Circuit IDs. A Circuit ID is a 64 bit number that identifies the DLC circuit within a single DLSw. It consists of a DLC Port ID (4 bytes), and a Data Link Correlator (4 bytes). The Circuit ID must be unique in a single DLSw and is assigned locally. The pair of Circuit IDs along with the Data Link IDs, uniquely identify a single end-to-end circuit. Each DLSw must keep a table of these Circuit ID pairs, one for the local end of the circuit and the other for the remote end of the circuit. In order to identify which Data Link Switch originated the establishment of a circuit, the terms, Origin DLSw and Target DLSw, are used.

Origin/Target transport ID

Used to identify the individual TCP/IP port on a Data Link Switch. The values have only local significance. However, each Data Link Switch is required to reflect the values contained in these two fields, along with the associated values for DLC Port ID and the Data Link Correlator, when returning a message to the other Data Link Switch.



```
Captured at:  +00:00.000
Length: 151     From: User     Status: Ok
DLSw: Version = 49
DLSw: Header Length = 72
DLSw: Message Length = 8
DLSw: Remote Data Link Correlator = 0x00000000
DLSw: Remote DLC Port ID = 0x00000000
DLSw: Message Type = Capabilities Exchange
DLSw: Flow Control Field
DLSw:   Flow Control Indicator = 1
DLSw:   Flow Control Ack = 1
DLSw:   Flow Control Operator Bits = 7
DLSw: Protocol ID = 66
DLSw: Header Number = 1
DLSw: SSP Flags = 0
DLSw: Circuit Priority = High Priority
DLSw: Target Mac Address = Cnf Rprt Srvr
DLSw: Origin Mac Address = Olicom70CCF3
DLSw: Origin Link Sap = AA (SNAP ), Command
DLSw: Target Link Sap = 2 (LLC Management ), Group SAP
DLSw: Frame Direction = Origin DLSw
DLSw: DLC Header Length = 0 (SNA)
DLSw: Origin Dlc Port ID = 0x00000000
DLSw: Origin Data Link Correlator = 0x00000000
DLSw: Origin Transport ID = 0x00000000
DLSw: Target Dlc Port ID= 0x00000000
DLSw: Target Data Link Correlator = 0x00000000
```

DLSw decode

SNA Terminology

Systems Network Architecture (SNA)

The description of the logical structure, formats, protocols and operational sequences for transmitting information units and controlling the configuration and operation of networks.

Network Addressable Unit (NAU)

A logical unit, physical unit or system services control point which is the origin or the destination of information transmitted by the path control network. Each NAU has a network address that represents it to the path control network.

Logical Unit (LU)

A port through which end users access the SNA network in order to communicate with other end users and the functions provided by system services control points (SSCPs). An LU can support at least two sessions (one with an SSCP and one with another LU) and may be capable of supporting many sessions with other logical units.

Physical Unit (PU)

One of the three types of network addressable units (NAUs). Each node of an SNA network contains a physical unit (PU) that manages and monitors the resources (such as attached links) of a node, as requested by a system services control point (SSCP) via an SSCP-PU session. An SSCP activates a session with the PU in order to indirectly manage resources of the node such as attached links through the PU.

System Services Control Point (SSCP)

A focal point within an SNA network for managing the configuration, coordinating network operator/problem determination requests and providing directory support and other session services for network end users. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical and logical units within its domain.

Bracket

One or more chains of request units (RUs) and their responses that are exchanged between the two LU-LU half-sessions and that represent a transaction between them. A bracket must be completed before another bracket can be started.

Data Link Control (DLC) Layer

The layer that consists of the link stations that schedule data transfer over a link between two nodes and perform error control for the link.

Normal Flow

A data flow designated in the transmission header (TH) that is used primarily to carry end-user data. The rate at which requests flow on the normal flow can be regulated by session-level pacing. Normal and expedited flows move in both the primary-to-secondary and secondary-to-primary directions.

Expedited Flow

A data flow designated in the transmission header (TH) that is used to carry network control, session control and various data flow control request/response units (RUs). The expedited flow is separate from the normal flow (which carries primary end-user data) and can be used for commands that affect the normal flow.

Explicit Route (ER)

The path control network elements, including a specific set of one or more transmission groups, that connect two subarea nodes. An explicit route is identified by an origin subarea address, a destination subarea address, an explicit route number and a reverse explicit route number.

LU Type 6.2

LU 6.2 is a particular type of SNA logical unit. It uses SNA-defined interprogram communication protocols and is also referred to as Advanced Program-to-Program Communication (APPC).

Network Services (NS) Header

A 3 byte field in an FMD request/response unit (RU) flowing in an SSCP-LU, SSCP-PU or SSCP-SSCP session. The network services header is used primarily to identify the network services category of the RU and the particular request code within a category.

Node Type

A designation of node according to the protocols it supports and the network addressable units (NAUs) that it can contain. Five types are defined: 1, 2.0, 2.1, 4 and 5. Node types 1, 2.0 and 2.1 are peripheral nodes and types 4 and 5 are subarea nodes.

PU Type 2 (T2)

A network node that can attach to an SNA network as a peripheral node.

PU Type 2.1 (T2.1)

A network node that can attach to an SNA network as a peripheral node using the same protocols as type 2.0 nodes; type 2.1 nodes can be directly attached to one another using SNA low-entry networking.

PU Type 4 (T4)

A network node containing an NCP and that is a subarea node within an SNA network.

PU Type 5 (T5)

A network node containing VTAM and that is a subarea node within an SNA network.

RU Chain

A set of related request/response units (RUs) that are consecutively transmitted on a particular normal or expedited data flow. The request RU chain is the unit of recovery: if one of the RUs in the chain cannot be processed, the entire chain is discarded. Each RU belongs to only one chain, which has a beginning and an end indicated via control bits for request/response headers within the RU chain. Each RU chain can be designated as first-in-chain (FIC), last-in-chain (LIC), middle-in-chain (MIC) or only-in-chain (OIC). Response units and expedited flow request units are always sent as OIC.

Session Control (SC)

One of the components of transmission control. Session control is used to purge data flowing in a session after an unrecoverable error occurs, in order to resynchronize the data flow after such an error and to perform cryptographic verification.

A request unit (RU) category used for requests and responses exchanged between the session control components of a session and for session activation and deactivation requests and responses.

SSCP-LU Session

A session between a system services control point (SSCP) and a logical unit (LU); the session enables the LU to request the SSCP to help initiate LU-LU sessions.

SSCP-PU Session

A session between a system services control point (SSCP) and a physical unit (PU); SSCP-PU sessions allow SSCPs to send requests to, and receive status information from individual nodes in order to control the network configuration.

SSCP-SSCP Session

A session between a system services control point (SSCP) in one domain and the SSCP in another domain. An SSCP-SSCP session is used to initiate and terminate cross-domain LU-LU sessions.

Token Ring

A network with a ring topology that passes tokens from one attaching device to another.