# Distributed Routing Software

# Systems Network Architecture Guide

Part Number: AA-QU5SB-TE

**December 1996**

This manual provides information about SNA interfaces and protocols for the Distributed Routing Software system.

**Revision/Update Information:** This is a revised manual.
**Software Version:** Distributed Routing Software V2.0

**Digital Equipment Corporation**
**Maynard, Massachusetts**

This manual was produced by Shared Engineering Services.

# Contents

## 2 Configuring and Monitoring the DLSw Protocol



## 3 Configuring and Monitoring SDLC Interfaces



## 4 Using Boundary Access Node

## 5 Using SDLC Relay



## 6 Configuring and Monitoring SDLC Relay

## A DLSw and SDLC MIB Support



## B nteroperating with the IBM 6611 Router



## Glossary

## Index

## Figures

## Tables

# Preface

## Objectives

This manual explains how to use the DLSw, SDLC Relay, and Boundary Access Node products to bridge and route SNA traffic across wide area networks. Specifically, this guide enables you to:

- Configure, monitor, and use the DLSw protocol.
- Configure, monitor, and use the SDLC interfaces.
- Configure, monitor, and use SDLC Relay.
- Configure, monitor, and use Boundary Access Node.

This preface describes how to use this book and the documentation set to which it belongs.

## Audience

This manual is intended for persons who install and operate computer networks. Although experience with computer networking hardware and software is helpful, you do not need programming experience to configure, monitor, and manage your network.

# Preface

## Using This Guide

The following table helps you locate information in this guide:

| If You Want Information About... | See Chapter or Appendix... |
|---|---|
| • Summary of Document Contents<br>• Related Documentation<br>• Document Set Structure<br>• Documentation Conventions | Preface |
| • Setting Up DLSw<br>• A Sample DLSw Configuration<br>• On Demand and Explicitly Configured TCP Sessions<br>• Using DLSw Groups<br>• Mixing T2.0 and T2.1 Link Stations on Multipoint Lines | **1** Using the DLSw Protocol |
| • DLSw Configuration and Console Commands<br>• Accessing the DLSw Configuration Prompt<br>• Accessing the DLSw Console Prompt<br>• DLSw Commands | **2** Configuring and Monitoring the DLSw Protocol |
| • SDLC Configuration and Console Commands<br>• Accessing the SDLC Configuration Environment<br>• Accessing the SDLC Console Environment<br>• SDLC Commands | **3** Configuring and Monitoring SDLC Interfaces |
| • Using BAN<br>• Using Multiple DLCIs for BAN Traffic<br>• Checking the BAN Configuration<br>• Enabling BAN ELS Messages<br>• BAN Configuration and Console Commands | **4** Using Boundary Access Node |

## Preface

| If You Want Information About... | See Chapter or Appendix... |
| --- | --- |
| • How SDLC Relay Works<br>• Setting Up SDLC Relay<br>• A Sample SDLC Relay Configuration | **5** Using SDLC Relay |
| • SDLC Relay Configuration and Console Commands<br>• Accessing the SDLC Relay Configuration Environment<br>• Accessing the SDLC Relay Console Environment<br>• SDLC Relay Commands | **6** Configuring and Monitoring SDLC Relay |
| • DLSw and SDLC MIB Support | **A** DLSw MIB Support |
| • Interoperating with IBM 6611 Router | **B** Interoperating with the IBM 6611 Router |

# Using Related Documentation

## Digital Documents

| This Document... | Describes... |
| --- | --- |
| *RouteAbout Access EI Installation* EK-DEXBR-TE | Installation and use of the RouteAbout Access EI router. |
| *RouteAbout Access EW Installation* EK-DE28R-IN | Installation and use of the RouteAbout Access EW router. |
| *RouteAbout Access TW Installation* EK-DEWTR-IN | Installation and use of the RouteAbout Access TW router. |
| *RouteAbout Central EW Installation* EK-DEZ8R-IN | Installation and use of the RouteAbout Central EW router. |
| *RouteAbout Central EI Installation* EK-DEZBR-IN | Installation and use of the RouteAbout Central EI router. |
| *Bridging Configuration Guide* AA-QL29D-TE | Configuration and monitoring procedures for bridging methods, and describes bridging features that enhance system performance. |
| *Event Logging System Messages Guide* AA-QL2AD-TE | How events are logged and how to interpret Event Logging System (ELS) messages. Provides a description of each ELS message with a corresponding corrective action. |
| *Network Interface Operations Guide* AA-QL2BD-TE | Configuring and monitoring the network interfaces in the Distributed Routing Software bridging router. |

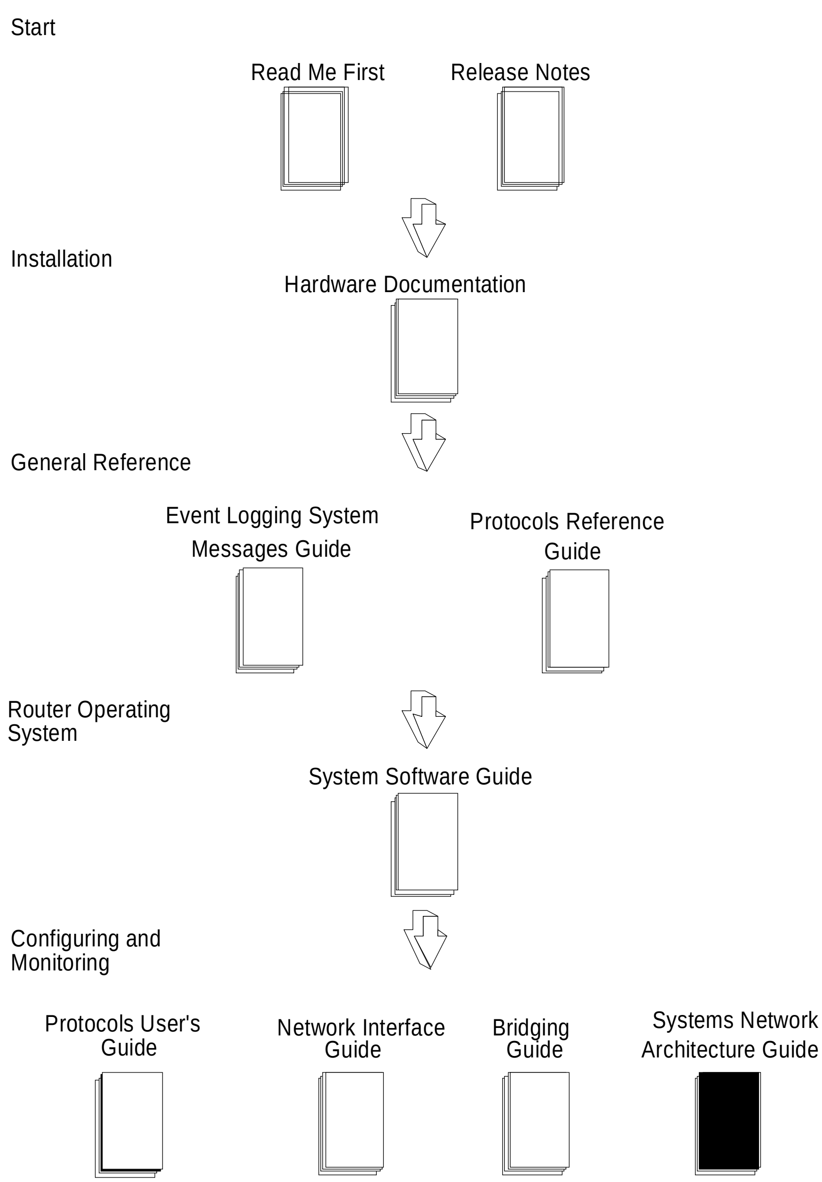
| This Document... | Describes... |
|---|---|
| *Routing Protocol User's Guide* AA-QL2DD-TE | How to configure and monitor the following protocols:<br>• AppleTalk Phase 1<br>• AppleTalk Phase 2<br>• ARP<br>• Bandwidth Reservation<br>• BGP4<br>• DVMRP<br>• IP<br>• IPX<br>• OSPF<br>• OSI/DNA V<br>• PIM<br>• SNMP<br>How to use the Digital Trace Facility. |
| *Routing Protocol Reference Guide* AA-QL2CD-TE | Reference information about the micro-operating system structure, and the protocols and interfaces that bridging routers support. |
| *Protocol Quick Reference Card Set* | How to configure and monitor a protocol, feature, or interface, and lists the associated commands. |
| *System Software Guide* AA-QL2ED-TE | Installing, configuring, and operating the Distributed Routing Software system software. |

## Document Set Structure

Figure 1 shows the structure of the documentation set.

## Preface

**Figure 1 Document Set Structure**



## Conventions

The following conventions are used in this manual:

| | |
|---|---|
| `Monospace type` | Monospace type in examples indicates system output or user input. |
| **`Boldface type`** | Boldface type in examples indicates user input. Boldface type is also used for file names and command names within text. |
| *lowercas-italics* | Lowercase italics in command syntax or examples indicate variables for which either the user or the system supplies a value. |
| [ ] | Brackets enclose operands or symbols that are either optional or conditional. Specify the operand and value if you want the condition to apply. Do not type the brackets in the line of code. |
| *key* | A key name enclosed in a box indicates that you press the specified key. |
| CTRL/x | indicates that you hold the Ctrl key while pressing the key specified by the *x*. The server displays the key combination as ^x. |
| underscore | Characters underlined in a command listing represent the fewest number of characters you must enter to identify that command to the interpreter. Characters are also underlined to indicate emphasis, such as notes and cautions. |

## Symbols

**C** **M**

The configuring and monitoring chapters contain a description of all commands you can use to configure and monitor the protocol, feature, or interface.

**C** means you use the command to configure the router. You access configuration commands after you enter **talk 6** at the * prompt. Configuration commands change the router's nonvolatile database; a router restart is necessary to activate the change.

**M** means you use the command to monitor and dynamically configure the router. You access monitoring commands after you enter **talk 5** at the * prompt. Changes made in this mode take effect immediately, but are not made in the router's nonvolatile database (and are therefore not preserved after a router restart).

**C** **M** means you can use the command either to configure or monitor the router.

**Note:** Talk 5 monitoring commands are also referred to as console commands in this guide. **Talk 6** configuration commands are sometimes referred to as just config commands.

## Commands

Figure 2 shows command components.

**Figure 2 Command Components**

> **Command Name**
>
> Description of commands.
>
> **Syntax:** command name
> parameter 1 . . .
> parameter 2 . . .
>
> parameter *option*
>
> Description of parameter and options.
>
> Example: **command name parameter**
>
> Prompt? **options**

**Syntax:** The command followed by each parameter you can configure using that command. If an ellipsis follows a parameter, you need to enter additional information (*options*). When you enter a command, you can save time by typing only the underlined letters.

parameter Description of each parameter.

*option* (in italics) Information you must enter with the command and parameter.

**Example:** An example of how you enter that command and parameter.

### Entering Commands

Instead of being prompted for options, you can save time by entering the complete command on one line. For example, you can enter the **set framesize** command shown in Figure 3 as follows:

```
set framesize 2048
```

If you abbreviate the command using the underlined letters, you can enter

```
s f 2048
```

**Figure 3  Set framesize command**

## Set

Configure frame size and local address.

**Syntax:** set
framesize . . .

framesize *1024* or *2048* or *4096*

Thes size of the network-layer portion of frames transmitted and received on the interface.

Example: **set framesize**

```
Framesize in bytes (1024/2048/4096) [1024]? 2048
```

## Accepting the Current Setting

When the software prompts you for information, the current setting appears in brackets [ ]. To accept the information in the brackets, press RET. In this example, the current setting is 1024.

```
Framesize in bytes (1024/2048/4096) [1024]?
```

## Reader's Comments

If you have comments or suggestions about this document, contact the Network Product Business Group.

- Send Internet electronic mail to: doc_quality@lkg.mts.dec.com
- Send comments via FAX to: 508-486-5655
- Send hardcopy mail to:

Digital Equipment Corporation
Shared Engineering Services
550 King Street (LKG1-3/L12)
Littleton, MA 01460-1289

## Preface

## How to Order Additional Documentation

To order additional documentation, use the following information:

| To Order: | Contact: |
|---|---|
| By Telephone | USA (except Alaska, New Hampshire, and Hawaii):<br>1-800-DIGITAL (1-800-344-4825)<br>Alaska, New Hampshire, and Hawaii: 1-603-884-6660<br>Canada: 1-800-267-6215 |
| Electronically (USA only) | Dial 1-800-DEC-DEMO<br>(For assistance, call 1-800-DIGITAL) |
| By Mail (USA and Puerto Rico) | DIGITAL EQUIPMENT CORPORATION<br>P.O. Box CS2008<br>Nashua, New Hampshire 03061<br>(Place prepaid orders from Puerto Rico with the local Digital subsidiary: 809-754-7575) |
| By Mail (Canada) | DIGITAL EQUIPMENT of CANADA LTD.<br>940 Belfast Road<br>Ottawa, Ontario, Canada K1G 4C2<br>Attn.: A&SG Business Manager |
| Internationally | DIGITAL EQUIPMENT CORPORATION<br>Attn.: A&SG Business Manager<br>c/o local Digital subsidiary or approved distributor |
| Internally | U.S. Software Supply Business (SSB)<br>DIGITAL EQUIPMENT CORPORATION<br>8 Cotton Road<br>Nashua, New Hampshire 03063 |

# 1

# Using the DLSw Protocol

This chapter describes Digital's implementation of the Data Link Switching (DLSw) protocol. The DLSw protocol offers a wide range of functionality that enables you to integrate LAN or SDLC-based SNA traffic into heterogeneous multiprotocol wide area networks (WANs). This implementation of the DLSw protocol includes support of RFC 1795 (DLSw V1.0), handling both NetBIOS and SNA traffic.

## 1.1 About DLSw

DLSw is essentially a forwarding mechanism for NetBIOS and SNA protocols running over both SDLC and LLC2 data links. It relies on the Switch-to-Switch Protocol (SSP) running over TCP/IP to provide a reliable transport of SNA and NetBIOS traffic over the internet. DLSw does not provide full routing capabilities. Instead, it works by providing switching at the data link layer. Rather than bridging LLC2 frames, DLSw terminates the LLC2 (or SDLC) connection locally and encapsulates only the Information (I) and Unnumbered Information (UI) frames in TCP frames. The router ships the TCP frames over the WAN link to a neighbor DLSw router for delivery to their intended end station addresses.

### 1.1.1 How DLSw Works

LLC2 and SDLC are connection-oriented protocols. DLSw gives these protocols the dynamic characteristics of routable protocols. Equally important, DLSw preserves the end-to-end reliability and control features that make these protocols effective for communication.

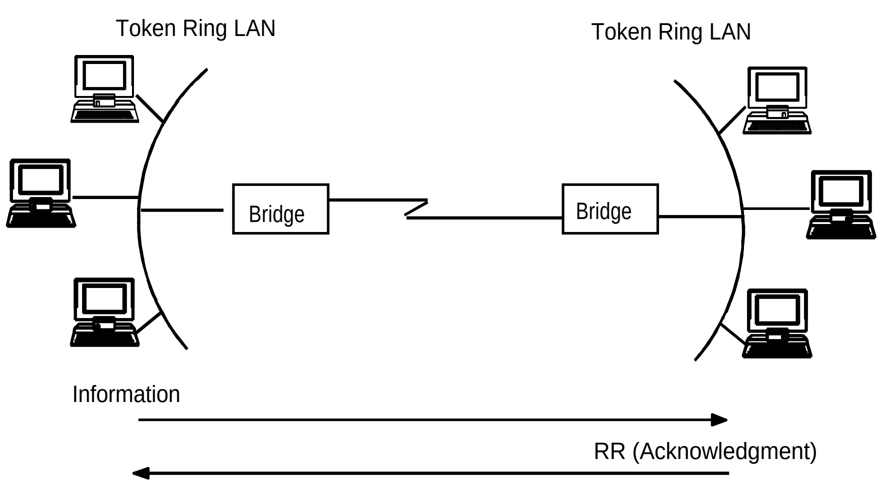
#### 1.1.1.1 Problems Inherent in the Bridging Solution

Figure 1–1 illustrates the traditional approach to bridging LLC2 frames across WAN links. The problem with this approach is that network delays occur much more frequently in the WAN than on a LAN. Such delays can arise from simple network congestion, slower line speeds, or other factors. Each of these factors increases the possibility of a session timing out and of data failing to arrive at the destination.

In addition, LAN protocols such as LLC2 use much shorter retransmit/response times than those designed for use in the WAN. This makes maintaining end-to-end connections across WAN links extremely difficult, causing session timeouts to occur.

The frequency of session timeouts is not the only problem. Another problem arises when data is delayed while crossing the WAN. When a sending station retransmits data that has not been lost, but delayed, LLC2 end stations may end up receiving duplicate data. While this would seem to safeguard the data, it can lead to confusion of the LLC2 procedures on the receiving side. This may, in turn, lead to inefficient use of the WAN link.

**Figure 1–1 Traditional Approach to Bridging over the Internet**

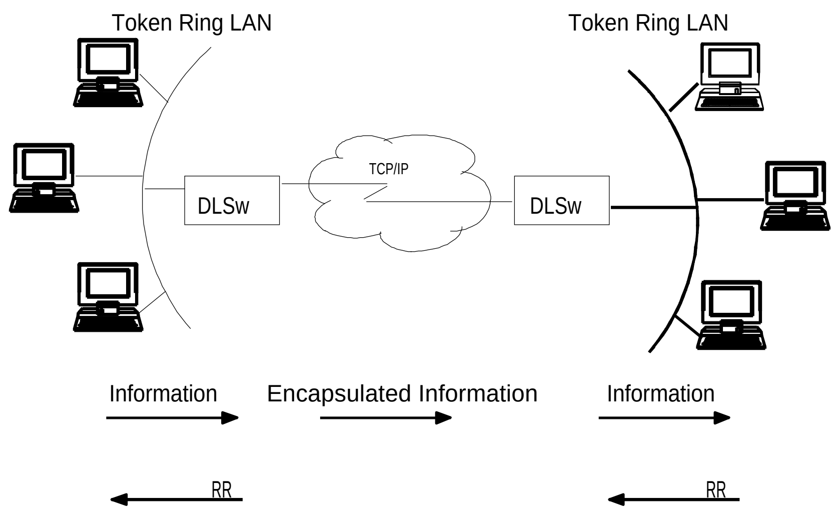



#### 1.1.1.2 Protocol Spoofing

To reduce the chance of session timeouts and to maintain the appearance of end-to-end connectivity for sending stations, DLSw works by terminating or spoofing connections at the local router. When terminating the connection, the local router sends acknowledgments to the sending station. The acknowledgment tells the sender that data previously transmitted have been received, and prevents the station from retransmitting.

From this point forward, assuring that data gets through is the responsibility of the DLSw software. The software accomplishes this by encapsulating the data in routable IP frames, then transporting them (via TCP) to a DLSw peer. The neighbor DLSw router strips away the frame headers, determines the address of the data's intended recipient, and establishes a new LLC2 or SDLC connection with that end station. Figure 1–2 illustrates this relationship between two DLSw neighbor routers. In this example, the routers are connected via a Token Ring LAN (LLC2).

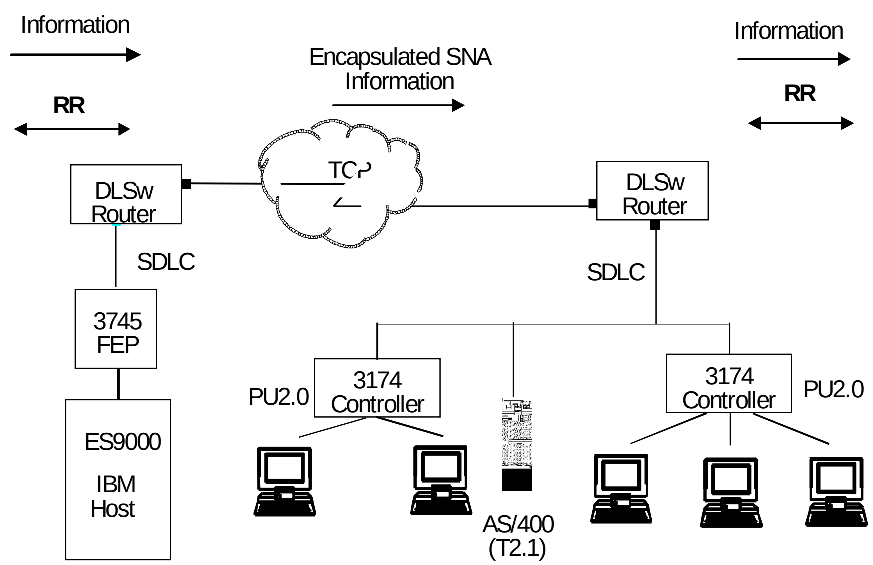
**Figure 1–2 Data Link Switching over the Wide Area Network**




## 1.1.2 SDLC Data Link Support

In addition to LAN data link support for SNA (LLC2) and NetBIOS, DLSw supports SDLC data link termination for SDLC-attached SNA devices. You can configure the router to act in either a primary or a secondary local link role. Support for SNA data link type is independent of the corresponding neighbor DLSw router; that is, the local router can have SDLC devices attached and the remote router's SNA devices can be on a Token Ring (LLC2).

**Figure 1–3 SDLC Support**




#### 1.1.2.1 Primary and Secondary Link Roles

In an SDLC primary local link role, the router polls downstream SNA PU2.0 or T2.1 devices such as IBM 3174 cluster controllers or an IBM AS/400. In Figure 1–3, the DLSw router on the right is configured with an SDLC primary local link role. In an SDLC secondary local link role, the router is itself polled by an adjacent (primary) station, most typically a mainframe Front End Processor (FEP) such as the IBM 3745/3746. The DLSw router shown on the left in the figure is configured with an SDLC secondary local link role.

In both the primary and secondary local link role, the DLSw router is capable of running SDLC multipoint. This capability is shown in Figure 1–3; the local secondary router is receiving polls from the FEP for three distinct stations (two 3174s and an AS/400), while the local primary router is polling those three devices. In the example provided, local secondary multipoint is running on a physical point-to-point link (router on left), while local primary multipoint is running on a physical multidrop link (router on the right).

An optional feature of T2.1 SDLC devices in a point-to-point link configuration is their ability to *negotiate* the link role. Digital's DLSw SDLC implementation supports this feature by allowing you to initially configure the role as negotiable. As the

T2.1 end-to-end role is resolved (via the T2.1 SDLC XID3 exchange protocol), the router senses the resolution and adjusts its own role accordingly. The router does not support dynamic role negotiation on multipoint links, nor does it support dynamic T2.1 link station address resolution.

If you configure respective SNA T2.1 end stations for role negotiation, but configure the router with a non-negotiable link role (that is, the link role is configured as either primary or secondary), the router attempts to “bias” the XID3 role negotiation protocol such that the local link station role is resolved accordingly.

#### 1.1.2.2 SNA Peripheral Node Types

You can configure the type of SNA node (PU2 or T2.1) for each SDLC link station. In addition to the link role consideration, the router uses the node type to determine whether or not to forward XID frames to the adjacent physical device.

For example, a local station configured with a PU2 node type on a local primary link does not forward NXID frames it receives (from the neighbor router) to the actual attached device. Instead, the router generates the appropriate XID0 response using the configured IDNUM and IDBLK values directly. This feature isolates the actual physical device configuration from the IBM host’s configuration parameters and permits, for example, transparent substitution of a remote SDLC device for an existing local Token Ring configuration.

With T2.1 SDLC devices, on the other hand, the router explicitly forwards all XID frames end-to-end, allowing true end-to-end XID3 parameter negotiation support. Mixed node types may be supported on a single multidrop physical link.

#### 1.1.2.3 Group Poll Feature

Digital’s implementation of the DLSw SDLC datalink also includes support for the SDLC Group Poll. This feature can greatly improve response time on SDLC links, since a single group (unnumbered) poll received by the router is used to address any available station with data to send.

Group polling is typically a feature of an IBM mainframe FEP, where it is known as the IBM 3174 Group Poll function. It is enabled on the mainframe Network Control Program (NCP) via the GP3174 keyword on the NCP PU definition; consult your IBM systems programmer for further details. On the router, individually configured link stations are explicitly included in the link group poll list, allowing full flexibility in poll service priorities.

Group Poll support is available only when the router is configured in a local secondary link role (that is, the router is capable or receiving and processing group polls, not sending them).

### 1.1.3 Benefits of DLSw

Because DLSw terminates the LLC connection at the local router, it is especially effective at eliminating SNA session timeouts and reducing WAN overhead on shared circuits. The protocol has these main benefits:

- Reduces the possibility of session timeouts by terminating LLC2, NetBIOS, and SDLC traffic at the local LAN.
- Reduces WAN network overhead by eliminating the need to transmit acknowledgments (RRs) over the WAN. The RR (Receive Ready) acknowledgments are confined to the LANs that are local to each DLSw router.
- Provides flow and congestion control and broadcast control of search packets between DLSw routers and their attached end stations.
- Increases Source Route Bridging (SRB) hop-count limits.
- Allows protocol conversion from LLC2 to SDLC.
- Supports routing (switching) of NetBIOS traffic.

## 1.2 Setting Up DLSw

The following sections explain the procedures to follow to set up DLSw:

- Configuration Requirements
- Configuring Adaptive Source Route Bridging (ASRT) for DLSw, as needed
- Configuring IP for DLSw
- Configuring SDLC Interfaces
- Configuring DLSw

In addition, a sample DLSw configuration with explanatory notes appears later in this chapter.

### 1.2.1 Configuration Requirements

Digital supports DLSw over IEEE 802.5 Token Ring, SDLC, and Ethernet.
To use DLSw, you must perform the following actions:

- Configure ASRT, as needed
- Configure IP
- Configure OSPF and MOSPF, as needed
- Configure SDLC devices, as needed
- Configure DLSw

The sections that follow explain step-by-step how to complete these actions. An annotated example of an actual DLSw configuration follows these procedures.

### 1.2.2 Configuring Adaptive Source Route Bridging (ASRT) for DLSw

Since the DLSw router appears as a bridge to attached end stations, you need to configure source route bridging. Note that in SDLC-only configurations, you do not need to set up ASRT.

Follow these steps to configure:

1. Enter **protocol asrt** at the `Config>` prompt to enter the ASRT configuration module.
2. Enter **enable bridge** to enable bridging on the router. Each bridge must have a unique bridge address.
3. Enter **add port** to add a bridge port for each interface that will be used by DLSw. The display prompts you for an interface number and a port number.
4. Configure LAN interfaces.
   - For Token Ring interfaces:

     Enter **disable transparent** to disable transparent bridging. Then, enter **enable source routing** to turn on source routing for the bridge port. You will be prompted for an SRB segment number.
   - For Ethernet interfaces:

   Enter **enable transparent** to enable transparent bridging on the bridge port.

5. If you are configuring the router for parallel DLSw and bridging paths, create a protocol filter against the SAPs (Service Access Points) you intend DLSw to use. If the router is performing bridging operations, plus forwarding packets via DLSw, it is essential to do this. If you do not, DLSw will both bridge and forward the packets it receives.

   To create a SAP filter,

   – Enter **add protocol-filter dsap 4** at the `ASRT Config>` prompt.

   – Specify the bridge port to which the filter applies. The command tells the router to filter all traffic that has a DSAP of 4 on a designated port. (Note that this assumes you have chosen a SAP of 4 for DLSw traffic. Assigning a SAP is something you do during the DLSw configuration.)

6. Next, verify the ASRT configuration. You do not have to do this, but it is a good idea to check the bridge configuration before proceeding. Use the **list bridge** command to verify the configuration of the ASRT protocol.

7. Enable the DLSw protocol using the **enable dls** command.

### 1.2.3 Configuring IP for DLSw

You need to configure IP so the local DLSw router can form the TCP connection to its DLSw peer. To do this, proceed as follows:

1. At the `Config>` prompt, enter **protocol ip** to display the IP configuration prompt.

2. Enter **add address** to assign the IP address to the hardware interface you are using to connect to the other DLSw peer.

3. Enable dynamic routing:

   If you do not define static routes between DLSw neighbors, you must choose either OSPF or RIP as your routing protocol. Using OSPF is recommended because it entails less network overhead than RIP.

   – To enable OSPF, enter **protocol ospf** at the `Config>` prompt. This brings you to the `OSPF Config>` prompt. To use DLSw group functionality, enable Multicast OSPF.

     For more information on using OSPF, see Using the OSPF Protocol in the *Routing Protocols User's Guide*.

   – To enable RIP**,** enter **enable RIP** at the `IP Config>` prompt.

4. Enter **set internal-ip-address** to set the address for the router as a whole. The router uses the internal IP address when it connects via TCP with its DLSw peer.

### 1.2.4 Configuring SDLC Interfaces

You must configure SDLC links if you intend to support SDLC over DLSw. This section explains how to access the SDLC configuration process and describes SDLC-related commands.

For more information about these commands, see Chapter 3.

1. At the `Config>` prompt, enter **set data-link sdlc** to set the data link protocol for the serial interface over which SDLC will run. The router prompts you for an interface number. To first see a list of available interface numbers, enter **list devices** at the `Config>` prompt.

   As the **set data-link sdlc** command is entered, a default SDLC link configuration is automatically created.

2. At the `Config>` prompt, enter **network** to display the SDLC configuration prompt. The router prompts you for an interface number.

3. Optionally use the **add station** command to explicitly configure one or more SDLC stations on the link associated with the interface. SDLC station(s) are also defined in the DLSw configuration (see Define Each SDLC Link Station later in this chapter), but doing so here provides additional configuration options not available when defined to DLSw alone.

   Explicit use of the **add station** command should be done in the following siuations:

   - These defaults for SDLC stations are not satisfactory:
     - Maximum BTU is maximum allowable by interface
     - Tx and Rx Windows are 7 for MOD 8, 127 for MOD 128
   - The SNA devices on the link are of mixed node types.
   - You want to use the group poll feature.
   - You want greater flexibility and control by using the SDLC monitoring commands.

   If you do not explicitly add SDLC stations, the router assumes the following:

   - The attached stations are of type PU2 if the router’s link role is primary or secondary.

- The attached stations are of type T2.1 if the router's link role is negotiable.

4. If the default link role of primary is not suitable, change the role to secondary or negotiable with the **set link role** command. Configuring the role values so that they are not conducive to each other and to the actual SNA devices in use can prevent successful link activation. Configure the link role as follows:

   - Specify secondary if the SDLC link is connected to an adjacent SDLC primary device, such as a FEP.
   - Specify negotiable if a T2.1 (APPN) device is attached and the role of the T2.1 device is itself negotiable (via XID3 exchange).
   - Retain the default primary role value if attaching secondary (PU2.0 or non-negotiable, secondary T2.1) devices.

     Connecting multiple T2.1 (or PU2.0) devices on a multidrop link by definition denotes that true link role negotiation is not being performed, and you should use a predefined link role on both the router and the actual device(s).

   It is not required that the respective T2.1 devices perform true end-to-end role negotiation when you configure the router's link as negotiable; the router senses the actual role, whether predetermined or not, and adjusts accordingly. Conversely, if you anticipate end-to-end T2.1 role negotiation and do not configure the router's link role as negotiable, the value you configure influences the role negotiation.

   Once the link is active, the resolved role will be displayed with the SDLC console **list link** command.

5. You should configure the router Group Poll feature if all of the following conditions exist:

   – the router will be configured to run local SDLC secondary

   – the router will support more than one station (secondary multipoint)

   – the router is connected to a mainframe that is configured to use the group poll feature (GP3174)

   In addition to the SDLC station addresses themselves, the use of group poll includes an additional *group poll address*. The address is configured on the router using the **set link group-poll** command. The value supplied must match that defined on the mainframe NCP definition. Note that a group poll address must be non-zero; a zero address on the router indicates that group polling is not configured.

Once the group poll address is configured, each secondary station defined on the router must be explicitly included in the "group poll list." Typically all stations will be included (if you don't include a given station, it won't resond to the group poll if it has data to send). There are two ways to include stations in the group poll list. You can use the **add station** command when initially promted, or you can use the **set station group-inclusion** command after creating the station. You should only add or remove stations from the group poll list in the *"talk 6"* (config) mode. Dynamic insertion or removal of stations on an active link is not supported.

You can determine the group poll list status of a specific station with the **list station** command. The first column listed under the "address" heading displays the configured station address and, when currently configured in the group poll list, the group poll address. For distinction, the group poll address is in parenthesis.

6. As required on limited number hardware platforms, set the cable type using the **set link cable** command.
7. Enter **list link** to verify the SDLC configuration. To change any of the parameters, use the **set** commands described in Chapter 3.

### 1.2.5 Configuring DLSw

Before you begin configuring DLSw, enter **list device** at the `Config>` prompt to list the available interface numbers.

To configure the DLSw protocol, follow these steps.

1. At the `Config>` prompt, enter **protocol dls**. This brings you to the `DLSw Config>` prompt.
2. Enter **enable dls** to enable DLSw in the router.
3. If your configuration is handling LLC2 or NetBIOS traffic, enter **set srb** to designate an SRB (Source Route Bridging) segment number for the DLS router.

   This segment number should be the same for all DLSw routers, and unique in the SRB domain. The bridge uses this number in the Routing Information Field (RIF) when the router sends the frames on the LAN. The segment number is the key to preventing loops.
4. Enter **open-sap** for each SAP that you wish DLSw to switch. The router prompts for interface numbers. To open commonly used SNA SAPs (0, 4, 8, and C), specify SNA. To open the NetBIOS SAP, specify NB or F0.

You do not need to open SAPs in a pure DLSw/SDLC router configuration.

**Note:** If you enable SAP List Filtering, you must open SAPs on the target DLSw router.

5. Use the **add tcp** command to add the IP address of each DLSw neighbor. You must use the internal address configured on the DLSw neighbor router. You can also make this connection using multicast OSPF with the **join-group** command.

**Note:** A router can participate in a group only if its neighbor router is a platform based on V2.0 running DLSw. If you configure one DLSw router for a group, you must enable OSPF and MOSPF on all DLSw routers in the group.

6. For your DLSw configuration to support SDLC, you must add an SDLC link station using the **add sdlc** command.

   Adding SDLC link stations requires knowledge of the device link station address, the optional NodeID field information (IDNUM and IDBLK), and the source and destination MAC addresses and SAPs for mapping to the corresponding remote SNA device.

   The link station address is required, and must match the address of the actual physical device you are connecting. If you have explicitly defined SDLC stations on the SDLC interface, the configured address must also match the SDLC interface address.

   See the DLSw **add sdlc** command in *Chapter 2* for more information.

## 1.3 Sample DLSw Configuration

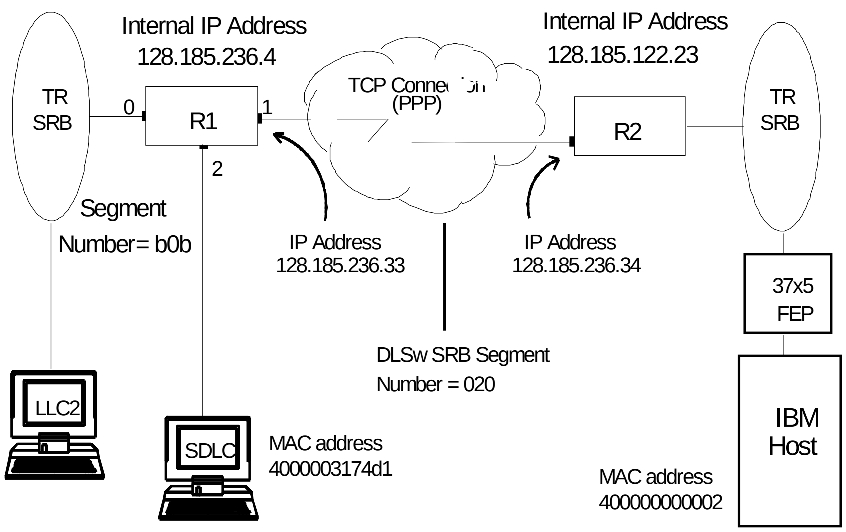
Following is a complete DLSw configuration based on the information shown in Figure 1–4. The DLSw router being configured (R1 in the diagram) supports one LLC and one SDLC connection to its DLSw neighbor (R2). The SDLC interface is configured with the router's link role as primary (local primary), and it is polling an SDLC PU2.0 device. The TCP connection between the two routers is PPP.

Configuring R1 for DLSw requires all of the information shown. This information includes the following:

- The internal IP addresses of R1 and R2.
- The IP address of each port used to maintain the TCP connection between the routers.

- The interface numbers assigned to the Token Ring and SDLC devices and used for the TCP connection.
- The SNA device addresses (station address for SDLC, LAN addresses for LAN-attached configurations).
- The source route bridge segment number of the attached Token Ring.

**Figure 1–4 Context Diagram for DLSw Configuration**




**Note:** A second DLSw configuration shown in Figure 1–5 follows this example. This configuration includes both primary and secondary SDLC stations.

## 1.3.1 Configuring the Token Ring Device

The following example shows how to configure Token Ring. Note that the **list** command shown here is not required at this point or at any other time during configuration of the router.

```
Config>network 0
Token-Ring interface configuration

TKR config>speed 16
```

```
TKR config>list
Token-Ring configuration:

Packet size (INFO field):  2052
Speed:                     16 Mb/sec
Ring select:               Backplane
RIF Aging Timer:           120
Source Routing:            Enabled
MAC Address:               000000000000
Netware IPX encapsulation: TOKEN-RING MSB

TKR config>exit
```

### 1.3.2 Configuring the WAN Interface

Use interface 1 for the WAN (TCP/IP) link (see Figure 1–4). The router defaults to PPP as the data link for the WAN.

```
Config>network 1
Point-to-Point user configuration
PPP Config>list all

Maximum frame size in bytes = 2048
Encoding: NRZ
Idle State: Flag
Clocking: External
Cable Type: RS232 DTE
Internal Clock Speed: 0

Transmit Delay Counter: 0

LCP Parameters . . .
```

The **list all** command displays the PPP interface parameters and options for all point-to-point parameters. The list has not been duplicated for this example.

If necessary, use the PPP **set** commands to change any of these defaults.

```
PPP Config>exit
```

### 1.3.3 Configuring the SDLC Device

The next step is to configure SDLC on interface number 2.

#### 1.3.3.1 Set the Data Link to SDLC

Set the data-link protocol for serial interface number 2 to SDLC.

```
Config>set data-link sdlc
Interface Number [0]? 2
```

#### 1.3.3.2 Display the SDLC Configuration Prompt

To access the SDLC configuration, enter **network** and the number of the SDLC interface (in this case, 2). Note that a default SDLC link configuration is created for you at this point.

```
Config>network 2
SDLC user configuration
Creating a default configuration for this link
SDLC 2 Config>
```

#### 1.3.3.3 Add an SDLC Station (optional)

See Configuring SDLC Interfaces for information on when and how to add SDLC stations. To add an SDLC station, enter **add station**.

```
SDLC 2 Config>add station
Enter station address (in hex) [C1]?
Enter station name [SDLC_C1]?
Include station in group poll list ([Yes] or No):
Enter max packet size [2048]?
Enter receive window [7]?
Enter transmit window [7]?
Enter PU2 or T2.1 node type [PU2]?
```

#### 1.3.3.4 Configure the SDLC Link

Enter **list link** to display the current configuration of the SDLC link.

```
SDLC 2 Config>list link
Link configuration for: LINK_2   (ENABLED)
Role:         PRIMARY          Type:         POINT-TO-POINT
Duplex:       FULL             Modulo:       8
Idle state:  FLAG              Encoding:     NRZ
Clocking:     EXTERNAL         Frame Size:  2048
Speed:        0                Group Poll:  00
Cable:         V.36 DTE

Timers:    XID/TEST response:  2.0 sec
           SNRM response:      2.0 sec
           Poll response:      0.5 sec
           Inter-poll delay:   0.2 sec
           RTS hold delay:     DISABLED
           Inter-frame delay:  DISABLED
           Inactivity timeout: 30.0 sec

Counters:  XID/TEST retry:  4
           SNRM retry:      6
           Poll retry:      10
```

If any SDLC link settings do not apply to the link role, the software ignores them.

You can change any of these settings using the SDLC configuration commands described in Chapter 3.

## 1.3.4 Configuring Protocols

This section describes how to configure IP, OSPF (or RIP), ASRT, and the DLSw protocol.

### 1.3.4.1 Configure IP

This example begins with the creation of a minimal IP configuration. For more information on configuring IP, see the *Routing Protocols User's Guide*.

To configure IP, begin by entering **protocol ip** at the `Config>` prompt:

```
Config (only)>protocol ip
Internet protocol user configuration
```

### 1.3.4.2 Assign an Internet Address to the WAN Link

Add an internet address and assign it to one of the interfaces associated with the WAN link configured earlier:

```
IP config>add address
Which net is this address for [0]? 1
New address [0.0.0.0]? 128.185.236.33
Address mask [255.255.0.0]? 255.255.255.0
```

### 1.3.4.3 Set the Internal IP Address

Set the internal IP address. This is the address that remote DLSw routers use to connect to the router you are configuring. See the *Routing Protocols Reference Guide* for more information about IP internal addresses.

```
IP config>set internal-ip-address 128.185.236.49
```

Enter **list** to display the newly added information.

```
IP config>list all
Interface addresses
IP addresses for each interface:
intf 1  128.185.236.33  255.255.255.0 Network broadcast, fill 0
Internal IP address: 128.185.236.49
```

```
Routing Protocols

BOOTP forwarding: disabled
Directed broadcasts: enabled
ARP Subnet routing: disabled
RFC925 routing: disabled
OSPF: disabled
Per-packet-multipath: disabled
RIP: disabled
EGP: disabled

IP config>exit
```

#### 1.3.4.4 Configure OSPF or RIP

This sample configuration uses OSPF rather than RIP. You can use either of these routing protocols. However, if you choose RIP, you will not be able to use DLSw group functionality.

The **list all** command displays the current OSPF configuration.

```
Config>protocol ospf
Open SPF-Based Routing Protocol configuration console
OSPF Config>list all

                  --Global configuration--
          OSPF Protocol:          Disabled
          External comparison:    Type 2
          AS boundary capability: Disabled
          Multicast forwarding:   Disabled

                 --Area configuration--
Area ID       AuType     Stub?   Default-cost Import-summaries?
0.0.0.0       0=None     No      N/A           N/A
```

#### 1.3.4.5 Enable OSPF

The first step consists of enabling OSPF and estimating the number of external routes and OSPF routers.

```
OSPF Config>enable ospf
Estimated # external routes [0]? 100
Estimated # OSPF routers [0]? 25
```

#### 1.3.4.6 Enable Multicast OSPF as Needed

Since this example implements DLSw Group Functionality, you must enable multicast OSPF, as shown:

```
OSPF Config>enable multicast-routing
Inter-area multicasting enabled? [No]: N
```

#### 1.3.4.7 Define the Interfaces That Will Use OSPF

You must enter **set interface** for every physical IP interface that will use OSPF. This example assumes that the backbone is the OSPF area (0.0.0.0). At this point, only one IP interface has been defined.

```
OSPF Config>set interface 128.185.236.33
Attaches to area [0.0.0.0]?
Retransmission Interval (in seconds) [5]?
Transmission Delay (in seconds) [1]?
Router Priority [1]?
Hello Interval (in seconds) [10]?
Dead Router Interval (in seconds) [40]?
Type Of Service 0 cost [1]?
Authentication Key []?
Retype Auth. Key []?
OSPF Config>
```

#### 1.3.4.8 Check the OSPF Configuration

Following is the OSPF display after it has been configured. To see what has changed in the configuration, compare this display with the display of the default OSPF configuration shown in this chapter.

```
OSPF Config>list all

            --Global configuration--
     OSPF Protocol:          Enabled
     # AS ext. routes:       100
     Estimated # routers:    25
     External comparison:    Type 2
     AS boundary capability: Disabled
     Multicast forwarding:   Enabled
     Inter-area multicast:   Disabled
     Inter-AS multicast:     Disabled


                        --Area configuration--
Area ID      AuType       Stub? Default-cost Import-summaries?
0.0.0.0      0=None       No         N/A           N/A

                        --Interface configuration--
IP address      Area     Cost  Rtrns  TrnsDly  Pri  Hello  Dead
128.185.236.33  0.0.0.0   1      5       1      1     10    40

                                  Multicast parameters
IP address       MCForward    DLUnicast    IGMPPoll    IGMPtimeout
128.185.236.33   On           Off          60          180
```

```
OSPF Config>exit
```

## 1.3.5 Configuring ASRT (Bridging)

DLSw requires SRB (Source Route Bridging) to run over a Token Ring interface. Conversely, transparent bridging is required for Ethernet devices, but does not work if the attached device is Token Ring.

This example is based upon a Token Ring connection to the DLSw router. Begin by enabling the bridge as shown:

```
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>enable bridge
```

### 1.3.5.1 Disable Transparent Bridging

The **list port** command shows that the port defaults to transparent bridging.

```
ASRT config>list port
Port Id (dec)    : 128:01, (hex): 80-01
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only
Assoc Interface  : 0
Path Cost        : 0
+++++++++++++++++++++++++++++++++++++++++++
```

Begin by disabling transparent bridging on the Token Ring port. Port number one is *port 1 on interface 0*. In other words, port 1 is the logical bridge port for the physical interface set up for the Token Ring.

```
ASRT config>dis transparent
Port Number [1]?
ASRT config>
```

### 1.3.5.2 Enable Source Route Bridging

Next, enable source route bridging for the Token Ring port as shown:

```
ASRT config>enable source
Port Number [1]?
```

### 1.3.5.3 Assign a Port Segment Number and Enable DLSw

Now, assign a segment number for the port. You only have to assign segment numbers when configuring a source route bridge device, such as Token Ring. In this example (see Figure 1–3) **b0b** is the hexadecimal number assigned to the Token Ring device.

```
Segment Number for the port in hex(1 - FFF) [1]? b0b
Bridge number in hex (1 - 9, A - F) [1]?
```

After assigning a segment number, enable DLSw for the bridge.

```
ASRT config>enable dls
```

Listing the bridge configuration confirms that you have configured ASRT correctly.

```
ASRT config>list bridge
```

```
                    Source Routing Transparent Bridge Configuration
                    ===============================================


    Bridge:                  Enabled                Bridge Behaviour: Unknown
                       +----------------------
    -------------------| SOURCE ROUTING INFORMATION |-
                       +----------------------
    Bridge Number:          01                      Segments:          1
    Max ARE Hop Cnt:        14                      Max STE Hop cnt:  14
    1:N SRB:                Not Active              Internal Segment:  0x000
    LF-bit interpret:       Extended



                       +-------------------+
    -------------------| SR-TB INFORMATION |--
                       +-------------------+
    SR-TB Conversion:        Disabled
    TB-Virtual Segment:      0x000                  MTU of TB-Domain:  0

                       +----------------------
    -------------------| SPANNING TREE PROTOCOL INFORMA
                       +----------------------
    Bridge Address:          Default                Bridge Priority:   32768/0x8000
    STP Participation:       IEEE802.1d

                       +----------------------
    -------------------| TRANSLATION INFORMATION |-
                       +----------------------
    FA<=>GA Conversion:      Enabled                UB-Encapsulation:  Disabled
    DLS for the bridge:      Enabled


                       +------------------+
    -------------------| PORT INFORMATION |---
                       +------------------+
    Number of ports added: 1
    Port:   1      Interface:       0      Behaviour:    SRB Only   STP:  Enabled
```

### 1.3.6 Implementing Protocol Filtering

This is an important step when configuring DLSw.

**Note:** You only need to implement the filter described here if you configure parallel bridging and DLSw. Such is not the case in this example. The procedure for creating a SAP filter is provided for reference purposes only.

DLSw forwards traffic by Service Access Point (normally using SAPs 04, 08, and 0C), so a bridging protocol filter is added for these SAPs. The filter prevents the bridge from forwarding, on other ports, packets that only DLSw should handle.

The command shown below creates a filter that works on all packets with a destination SAP of 4. The **list** command issued subsequently displays the filter characteristics.

```
ASRT config>add prot-filter dsap 4
Filter packets arriving on all ports?(Yes or [No]): yes


ASRT config>list prot-f dsap
Protocol Class: DSAP
Protocol Type : 04
Protocol State: FILTERED
Port Map      : 1
==========================
No ETHER type Filter Records Associated
No SNAP Filter Records Associated
```

Once the filtering you need is in place, exit the ASRT configuration module.

```
ASRT config>exit
```

### 1.3.7 Configuring DLSw

The final step involves configuring the DLSw protocol. The **list** command below shows the defaults.

```
Config (only)>protocol dls
DLSw protocol user configuration
DLSw config>list dls


DLSw is                            ENABLED
LLC2 send Disconnect is            ENABLED
Automatic TCP connection           ALWAYS CONNECT

SRB Segment number                 000
MAC <-> IP mapping cache size      128
Max DLSw sessions                  1000
DLSw global memory allotment       141056
LLC per-session memory allotment   8192
SDLC per-session memory allotment  4096
NetBIOS UI-frame memory allotment  40960
```

```
Database age timer                  1200  seconds
Max wait timer for ICANREACH        20    seconds
Wait timer for LLC test response    15    seconds
Wait timer for SDLC test response   15    seconds
Join Group Interval                 900   seconds
Neighbor priority wait timer        2.0   seconds
```

Enable DLSw and set the SRB segment number. The segment number is the virtual segment number that identifies DLSw in the RIF (source route routing information field) of all DLSw-destined LLC frames. Use an identical DLSw SRB segment number for all DLSw routers configured for source routing.

```
DLSw config>enable dls
DLSw config>set srb 020
```

#### 1.3.7.1 Configuring DLSw Groups and Static Sessions

You must define either a DLSw group or a static TCP session to connect to a neighbor DLSw router. This example defines both a group and a static (explicitly configured) TCP session. Using DLSw groups requires a valid OSPF configuration.

The **join** command is used to join a DLSw group. You designate each group member as Client, Server or Peer. Client is the default.

#### 1.3.7.2 Using the Join-Group Command

The **join-group** command executed for R1 (see Figure 1–4), designates this DLSw router as a Client in group 1. To join this group, R2 would have to be added as a Server in group 1. All clients in a group locate and establish TCP connections for DLSw with servers within the same group.

```
DLSw config>join
Group ID (1-64 Decimal) [1]?
Client/Server or neighbor Group member (C/S/P)- [C]?
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal)  [5120]?
Maximum Segment Size (Decimal) [1024]?
Enable/Disable Keepalive (E/D)- [D]?
Neighbor Priority (H/M/L) [M]?


DLSw config>list group
Group Role   Xmit Bufsize  Rcv Bufsize  Max Segsize  Keepalive   Priority
  1   CLIENT     5120         5120         1024      DISABLED    MEDIUM
```

### 1.3.7.3 Using the Add TCP Command

The **add tcp** command creates explicitly configured DLSw routes. The neighbor DLSw IP Address added here is the internal IP Address of the neighbor DLSw router (called R2 in Figure 1–3). Note that you must also configure R2 with the neighbor IP Address of R1.

```
DLSw config>add tcp
Enter the DLSw neighbor IP Address [0.0.0.0]? 128.185.122.234
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal)  [5120]?
Maximum Segment Size (Decimal) [1024]?
Enable/Disable Keepalive? (E/D) - [D]?
Neighbor Priority (H/M/L) [M]?
```

```
DLSw>list tcp
Neighbor  Xmit Bufsize  Rcv Bufsize  Max Segsize  Keepalive  Priority
--------  ------------  -----------  -----------  ---------  --------
128.185.122.234   5120          5120         1024      DISABLED    MEDIUM
```

### 1.3.7.4 Define Each SDLC Link Station

You must define each SDLC link station as shown.

```
DLSw config>add sdlc
Interface # [0]? 2
SDLC Address [C1]?
Source MAC Address [000000000000]? 4000003174d1
Idblk in Hex (0-0xfff) [0]? D15
Idnum in Hex (0-0xfffff) [0]? A0001
LLC Source SAP (0 for auto-assign) [0]?
LLC Destination SAP [4]?
Destination MAC Address [000000000000]? 400000000002
```

```
DLSw config>list sdlc all
Net  Addr   Status    Idblk  Idnum   Source SAP/MAC    Dest SAP/MAC
 4    C1    Enabled    D15   A0021   00 4000003174d1  04 400000000002
```

It is very important that the values assigned with **add station** are entered correctly, and that they match those defined externally to the router. The supplied SDLC station address must match that of the actual attached SDLC station (and also that of the router SDLC datalink definition, if also supplied). The LAN address components (source and destination MAC and SAP values) must match those defined on the remote IBM mainframe token ring network. This is particularly critical because the

DLSw logic translates between LAN-addressed SNA traffic arriving from the remote DLSw neighbor router (as it in turn is received from the mainframe Token Ring), and the SDLC station.

The Idblk and Idnum values are used for SDLC PU2.0 directed traffic. When a NXID (null XID) is received from the mainframe, the local router generates and returns an XID0 (XID format 0) which identifies the specific PU2.0 to the predefined mainframe configuration. As with LAN address components, Idblk and Idnum values must match those defined on the mainframe.

#### 1.3.7.5 Open SAPs

Next, open SAPs on each bridging interface that performs DLSw switching.

SAP numbers 0, 4, 8, and C are commonly used SNA SAPs. To open all of these SAPs, use the SNA option with the **open-saps** command as shown. To open SAPs for NetBIOS, choose NB. You can also enter SAPs individually by entering a hexadecimal number.

```
DLSw config>open-sap
Interface # [0]?
Enter SAP in hex (range 0-F0), 'SNA', 'NB' [4]? sna
SAPs 0 4 8 C opened on interface 0
```

Following is the DLSw display after configuring.

```
DLSw config>list dls

DLSw is                             ENABLED
LLC2 send Disconnect is             ENABLED
Automatic TCP connection            ALWAYS CONNECT

SRB Segment number                  020
MAC <-> IP mapping cache size       128
Max DLSw sessions                   1000
DLSw global memory allotment        141056
LLC per-session memory allotment    8192
SDLC per-session memory allotment   4096
NetBIOS UI-frame memory allotment   40960

Database age timer                  1200  seconds
Max wait timer for ICANREACH        20    seconds
Wait timer for LLC test response    15    seconds
Wait timer for SDLC test response   15    seconds
Join Group Interval                 900   seconds
Neighbor priority wait timer        2.0   seconds
```

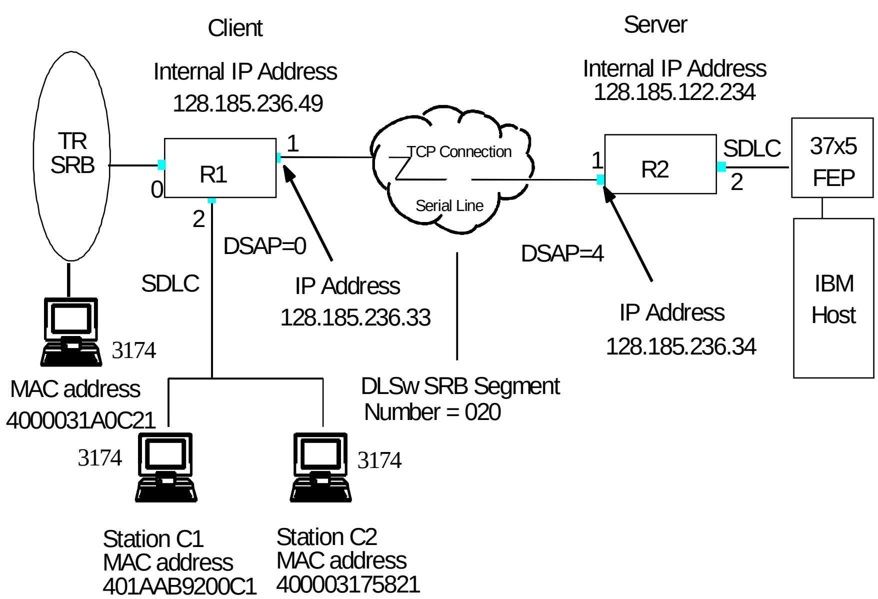
When you have finished configuring DLSw, exit the DLSw configuration environment and restart the router.

```
DLSw config>exit
Config>restart
Are you sure you want to restart the gateway? (Yes or [No]): yes
```

## 1.4 Sample DLSw Configuration Using Primary and Secondary SDLC Stations

Figure 1–5 shows a sample DLSw configuration using both Token Ring-to-SDLC and SDLC-to-SDLC neighbor DLSw circuits. Router R1 is running LLC2 over the Token Ring, communicating with the IBM mainframe on the remote end via router R2. Router R1 is also configured for SDLC itself (local primary role), connecting the 3174 controllers shown. Router R2 is connected to the IBM mainframe via SDLC; the mainframe is primary and router R2 therefore has its link configured as secondary (local secondary link role).

**Figure 1–5 Sample DLSw Configuration Using Primary and Secondary SDLC Stations**




Most of the steps for this configuration are the same as those in the previous section for Figure 1–4. This section describes only the differences in how to set up SDLC and DLSw for this configuration.

Notice that the SDLC Destination SAPs (DSAPs) on the SDLC local primary router (R1) are 0 (zero). This puts the stations in passive mode, and prevents the local primary router from attempting to establish DLSw sessions with the local secondary router (R2). The DSAP on the secondary router is not set to zero, allowing that router to initiate sessions.

Because the FEP must poll the SDLC local secondary router before the router can engage in session establishment, if the local primary router attempts to establish a session and the secondary router is not being polled by the FEP, the attempt fails. Therefore, configuring the primary and secondary stations so that only the secondary router establishes sessions can avoid delays in establishing sessions. See the **add sdlc** command in Chapter 2 for more information on configuring DSAPs.

### 1.4.1 Configuring Router R1

On Router R1, the primary SDLC router, you need to

1. In the SDLC configuration, set the SDLC link to multipoint.

```
SDLC 2 Config>set link type multipoint
```

2. In the DLSw configuration, add an SDLC station for each of the 3174s. See the add sdlc command in Chapter 2 for more information.

```
DLSw config>add sdlc
Interface # [0]? 2
SDLC Address [C1]?
Source MAC Address [000000000000]? 401AAB9200C1
Idblk in Hex (0-0xfff) [0]?
Idnum in Hex (0-0xfffff) [0]?
LLC Source SAP (0 for auto-assign) [0]? 4
LLC Destination SAP [4]? 0
Destination MAC Address [000000000000]? 40002DECDEC0

DLSw config>add sdlc
Interface # [0]? 2
SDLC Address [C1]? C2
Source MAC Address [000000000000]? 400003175821
Idblk in Hex (0-0xfff) [0]?
Idnum in Hex (0-0xfffff) [0]?
LLC Source SAP (0 for auto-assign) [0]? 4
LLC Destination SAP [4]? 0
Destination MAC Address [000000000000]? 40002DECDEC1
```

### 1.4.2 Configuring Router R2

On Router R2, the secondary SDLC router, you need to

1. In the SDLC configuration, set the SDLC link role to secondary.

```
SDLC 1 Config>set link role secondary
```

2. In the DLSw configuration, add SDLC stations to interface 1 for each of the 3274s.

```
DLSw config>add sdlc
Interface # [0]? 2
SDLC Address [C1]?
Source MAC Address [000000000000]? 40002DECDEC0
Idblk in Hex (0-0xfff) [0]?
Idnum in Hex (0-0xfffff) [0]?
LLC Source SAP (0 for auto-assign) [0]? 4
LLC Destination SAP [4]? 4
Destination MAC Address [000000000000]? 401AAB9200C1
```

```
DLSw config>add sdlc
Interface # [0]? 2
SDLC Address [C1]? C2
Source MAC Address [000000000000]? 40002DECDEC1
Idblk in Hex (0-0xfff) [0]?
Idnum in Hex (0-0xfffff) [0]?
LLC Source SAP (0 for auto-assign) [0]? 4
LLC Destination SAP [4]? 4
Destination MAC Address [000000000000]? 400003175821
```

## 1.5 On Demand and Explicitly Configured TCP Sessions

DLSw can automatically reestablish TCP sessions both after a session breaks and at startup. The software accomplishes this through the use of two DLSw configuration or console commands.

- enable auto-tcp-reconnect
- disable auto-tcp-reconnect

The **enable auto-tcp-reconnect** command allows preconfigured TCP sessions to establish themselves automatically upon start-up, and causes broken sessions to re-establish. This is the default behavior for the router.

**Note:** The **enable auto-tcp-reconnect** command only applies if you have explicitly added TCP neighbor addresses. TCP sessions created through group membership always reconnect.

If you disable the default using **disable auto-tcp-reconnect**, DLSw sessions are not established until they are needed, and broken TCP sessions do not reestablish themselves until they are needed again. TCP group connections reestablish themselves after the interval specified using **set timers** expires. This command is available within the DLSw configuration and monitoring modules.

## 1.6 Using DLSw Groups

You can use DLSw Group capability to designate groups of DLSw routers. Setting up groups can be extremely beneficial because it reduces the need for long lists of static IP addresses and the cost associated with maintaining them. A DLSw router can be a member of up to 64 groups.

There are three types of groups: Client, Server, and Peer-to-Peer. Routers designated as Servers only form DLSw connections with Client routers; likewise, Client routers can only form connections with Servers. In Peer-to-Peer groups, all routers form connections with each other.

## 1.6.1 Setting Up DLSw Groups

You need to configure OSPF and MOSPF if you want to use the DLSw group feature. See "Configuring OSPF and RIP" in this chapter for instructions to configure those protocols.

### 1.6.1.1 Issue the DLSw Join-Group Command

At the DLSw config> prompt, enter **join-group**. The router prompts you for a group number, transmit and receive buffer sizes, type of membership in the group, and the neighbor priority for the group.

```
DLSw config>join-group

Group ID (1-64 Decimal) [1]? 4
Client/Server or neighbor Group member (C/S/P)- [C]?
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal)  [5120]?
Maximum Segment Size (Decimal) [1024]?
Enable/Disable Keepalive (E/D)- [D]?
Neighbor Priority (H/M/L) [M]?
DLSw config>
```

### 1.6.1.2 Enable OSPF and Multicast OSPF

Enable the OSPF routing protocol and OSPF multicast routing at the OSPF config> prompt, as shown:

```
OSPF Config>enable ospf
Estimated # external routes [0]? 100
Estimated # OSPF routers [0]? 25
OSPF Config>enable multicast
Inter-area multicasting enabled? [No]:
OSPF Config>
```

# 1.7 Mixing PU2.0 and T2.1 Link Stations on Multipoint Lines

Digital routers support co-existence of SNA PU2.0 and T2.1 link stations on SDLC multipoint lines.

By default, the router software treats all SDLC link stations as if they are of the same type; that is, these stations all function as either PU2.0 (link role primary) or T2.1 (link role negotiable) nodes from the router’s perspective.

## 1.7 Mixing PU2.0 and T2.1 Link Stations on Multipoint Lines

To mix SNA node types or attributes among the link stations on a single SDLC link, you must configure the router to support whichever nodes (stations) do not match the default. Link station defaults are as follows:

| | |
|---|---|
| *MaxBTU* | The maximum allowed by the interface |
| *Receive Window* | 7 for MOD 8, 127 for MOD 128 |
| *Transmit Window* | 7 for MOD 8, 127 for MOD 128 |

To mix PU2.0 and T2.1 node types on a single multidrop link, do the following:

- Define the corresponding link role as primary or negotiable (PU2.0 connections are not supported with a router local role of secondary).
- Explicitly define each SDLC station, and set the SNA node type on each to correspond with the physical device. The station is defined with the **add station** command. If already defined, the SNA node type may be changed with the **set station node-type** command.

# 2

# Configuring and Monitoring the DLSw Protocol

This chapter includes all of the configuration and console commands applicable to DLSw.

## 2.1 About DLSw Configuration and Console Commands

DLSw configuration commands are available at the `DLSw config>` prompt. Changes made to the router's configuration do not take effect immediately; they become part of the router's nonvolatile configuration memory when it restarts.

Conversely, DLSw console commands are available at the `DLSw>` prompt. Console commands take effect immediately, but do not become part of the router's nonvolatile configuration memory. Thus, while console commands allow you to make real-time changes to the router's configuration, the changes are temporary. The router's configuration memory overwrites them when the router restarts.

Monitoring consists of these actions:

- Using console commands to monitor the protocols and network interfaces currently in use by the router
- Displaying ELS (Event Logging System) messages relating to router activities and performance
- Making real-time changes to the DLSw configuration without permanently affecting the router's non-volatile configuration memory

## 2.2 Accessing the DLSw Configuration Prompt

Use the router's configuration process to change the configuration of the router. The new configuration takes effect when you restart the router.

To enter the configuration environment, enter **talk 6**, or just **t 6**, This brings you to the `Config>` prompt as shown here:

```
MOS Operator Control

* talk 6

Config>
```

If the `Config>` prompt does not appear immediately, press RET again.

All DLSw configuration commands are entered at the `DLSw Config>` prompt. To access this prompt, enter **protocol dls** as shown:

```
Config>protocol dls
DLSw protocol user configuration
DLSw config>
```

## 2.3 Accessing the DLSw Console Prompt

To enter the console environment, enter **talk 5**, or just **t 5**. This brings you to the console environment as shown:

```
MOS Operator Control

* talk 5
+
```

You enter DLSw console commands at the `DLSw>` prompt. To access this prompt, enter **protocol dls** at the + prompt as shown:

```
+ protocol dls
Data Link Switching Console
DLSw>
```

## 2.4 DLSw Commands

Table 2–1 lists the DLSw configuration and console commands. Enter DLSw configuration commands at the `DLSw config>` prompt and console commands at the `DLSw>` prompt.

**Table 2–1 DLSw Command Summary**

| Command | Task | Function |
|---|---|---|
| **? (Help)** | Configure/ Monitor | Lists the configuration commands or lists any parameters associated with the command. |
| **Add** | Configure/ Monitor | Adds an SDLC link station or a TCP neighbor IP address. |
| **Ban** | Configure/ Monitor | Displays the Boundary Access Node prompt. |
| **Close-Sap** | Configure/ Monitor | Closes a currently opened Service Access Point (SAP). A SAP is used by SDLC interfaces for communication on the network. |
| **Delete** | Configure/ Monitor | Removes configured SDLC link stations and TCP connections. |
| **Disable** | Configure/ Monitor | Disables the DLSw protocol, Auto-TCP-Reconnect, SDLC link station, and LLC disconnect functionality. |
| **Enable** | Configure/ Monitor | Enables the DLSw protocol, Auto-TCP-Reconnect, SDLC link station, and LLC disconnect functionality. |
| **Join-Group** | Configure/ Monitor | Allows DLSw neighbors to find each other dynamically. |
| **Leave-Group** | Configure/ Monitor | Removes the router from the specified DLSw group. |
| **List** | Configure/ Monitor | Displays information for SDLC link stations, SAPs, circuit priority, DLSw groups, and DLSw sessions. The command also provides detailed information on TCP capabilities, connections, and statistics. |
| **NetBIOS** | Configure/ Monitor | Displays the NetBIOS prompt. |
| **Open-Sap** | Configure/ Monitor | Allows DLSw to transmit data over the specific SAP. |
| **Set** | Configure/ Monitor | Configures LLC2 parameters, number of DLSw sessions, SRB segment number, TCP buffer size, memory allocation, protocol timers, and circuit priority. |
| **Exit** | Configure/ Monitor | Returns you to the `Config>` prompt or + prompt. |

### ? (Help) C M

Lists the commands available from the current prompt level. You can also enter **?** after a specific command name to list its options.

**Syntax:** ?

```
Example:  ?

    ADD
    BAN
    CLOSE-SAP
    DELETE
    DISABLE
    ENABLE
    JOIN-GROUP
    LEAVE-GROUP
    LIST
    NETBIOS
    OPEN-SAP
    SET
    EXIT
```

### Add C M

Configures an SDLC link station or a TCP neighbor IP address.

**Syntax:** add

sdlc
tcp

**sdlc**

Adds information specifically for adding an SDLC link station to the configuration on a SDLC serial interface. You must use **add sdlc** once for each secondary station on the SDLC link.

The source and destination MAC addresses and SAPs are mandatory and must be correct for a DLSw connection to take place. If the local devices are to communicate with remote SNA devices on an SNA LAN, such as a Token Ring, then the SAPs must correspond to those in use on the remote LAN. However, if the local SDLC devices are to communicate with remote SNA devices that are attached by an SDLC data link, then the MAC addresses and SAPs are arbitrary, provided they are legal values. In this case, the MAC addresses and SAPs must logically map to the reverse source and destination addresses at the remote router.

In SDLC-to-SDLC configurations, the destination SAP (DSAP) of the local primary link role router has special significance. If you set it to zero, it designates that a successful SDLC protocol handshake with the adjacent device should not initiate a DLSw connection (CANUREACH). For PU2 (non-negotiable) links with each router connected via an SDLC interface, set the DSAP of the local primary router to zero. This prevents unnecessary DLSw circuit startups from occurring. Otherwise, the local primary router attempts a DLSw CANUREACH connection to the local secondary router, but since the secondary router cannot itself activate the data link to the adjacent SDLC primary station, the connection is guaranteed to fail.

`Example:` **`add sdlc`**

| | |
|---|---|
| *Interface #* | The interface number of the router you are adding to the SDLC link station. |
| *SDLC Address* | The SDLC address of the link station that you are connecting between 01 - FE (hex). |
| *Source MAC Address* | The MAC address for the attached SDLC PU, supplied in non-canonical bit order (Token Ring) format. This is the MAC address to which the remote link station will forward data. The local router will convert it to the corresponding SDLC station address. |
| *Idblk in Hex* | The 3-digit hexadecimal value that identifies the device (PU) to which you are connecting. Normally you will use Idblk for PUs defined as switched major nodes. Therefore, this value should match this same parameter in the VTAM Switched Major Node that corresponds to this PU. |
| *Idnum in Hex* | The 5-digit hexadecimal value that identifies the specific device type (2.0) that you are connecting. Normally you will use Idnum for PUs that are switched major nodes. Therefore, this value should match this same parameter in the VTAM Switched Major Node that corresponds to this PU. |

| | |
|---|---|
| *LLC Source SAP* | Identifies the PU link station to the DLSw domain. This can be explicitly assigned via configuration or automatically assigned by software. SAPs only apply to LLC use. |
| *LLC Destination SAP* | Defines the SAP to be used when automatically attempting a connection when the link station comes up. If this SAP is 0, then the link station is in *passive* mode and does not attempt to establish a circuit. In this case, the router ignores the destination MAC address. |
| *Destination MAC Address* | The MAC address of the remote link station with which a connection is established when the SDLC device becomes active. The MAC address is in non-canonical bit order (Token Ring) format. This is true even if the remote end station is on the Ethernet. |

**tcp**

Adds the IP address of the DLSw neighbor to which the TCP is connected. You can make this connection in two ways: manual configuration of IP neighboring addresses or with DLSw groups.

Example: **add tcp**

```
Enter the DLSw neighbor IP Address [0.0.0.0]? 128.185.14.1
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal)  [5120]?
Maximum Segment Size (Decimal) [1024]?
Enable/Disable Keepalive? (E/D) - [D]?
Neighbor Priority (H/M/L) [M]?
```

| | |
|---|---|
| *Enter the DLSw neighbor IP Address* | The IP address of the remote DLSw neighbor in the IP network to which you want to make a connection. |
| *Transmit Buffer Size* | The size of the packet transmit buffer between 1024 and 32768. The default size is 5120. |

| | |
|---|---|
| *Receive Buffer Size* | The size of the packet receive buffer between 1024 and 32768. The default size is 5120. |
| *Maximum Segment Size* | The maximum size of the TCP segment is between 1024 and 16384. The default size is 1024. |
| *Enable/Disable Keepalive (E/D)* | Indicates whether you want the DLSw neighbor to send link keepalive messages. The default is D (Disable). |
| *Neighbor Priority* | Allows you to specify the neighbor priority as either High, Medium, or Low. DLSw uses this parameter to determine which DLSw neighbor to choose when multiple neighbors can reach a target station. |

### BAN C M

Displays the Boundary Access Node (BAN) configuration or console prompt. See Chapter 4, *Using Boundary Access Node*, for more information on the commands available at these prompts.

**Syntax:** ban

Example: **ban**

```
BAN (Boundary Access Node) configuration
BAN config>
```

### Close-Sap C M

Disables DLSw switching for the specified Service Access Point (SAP) by the DLSw protocol. LLC uses these SAPs for configuration on the network.

**Syntax:** close-sap

Example: **close-sap**

```
Interface # [0]?
Enter SAP in hex (range 0-F0), 'SNA' or 'NB' [4]? nb
SAP F0 closed on interface 0
```

*Interface #* — The interface number used by the open SAP.

*Enter SAP* — You can enter individual SAPs in hex or you can enter SNA or NB (NetBIOS).

If you enter SAPs in hex, the range is 0 to F0, and the SAP must be an even number.

If you enter SNA, SAPs 0, 4, 8, and C are opened.

If you enter NB, SAP F0 is closed for NetBIOS.

### Delete C M

Removes an SDLC link station or a TCP neighbor IP address from the DLSw configuration.

C In the configuration environment, **delete** works as follows:

**Syntax:** delete
sdlc
tcp

**sdlc**

Removes the specified SDLC link station from the list of stations to which DLSw can connect. This also terminates any existing session.

Example: **delete sdlc**

```
Interface #[0]?
SDLC Address [C1]?
Record deleted
```

*Interface #* — The interface number of the router that connects to the SDLC link station.

*SDLC Address* — The SDLC address of the remote link station that you are deleting. Values are in the range 01 to FE.

**tcp**

Removes the IP address of the DLSw neighbor to which you are making the TCP connection.

Example: **delete tcp**

```
IP Addess [0.0.0.0]? 128.185.14.1
```

**M** In the console environment, **delete** works as follows:

**Syntax:** delete
sdlc
tcp

**sdlc**

Removes the specified SDLC link station from the list of stations to which DLSw can connect. This also terminates any existing session.

Example: **delete sdlc**

```
Interface #[0]?
SDLC Address [C1]?
Record deleted
```

| | |
|---|---|
| *Interface #* | The interface number of the router that connects to the SDLC link station. |
| *SDLC Address* | The SDLC address of the remote link station that you are deleting. Values are in the range 01 to FE. |

**tcp**

Removes the IP address of the DLSw neighbor to which you are making the TCP connection. This also terminates the TCP connection if one exists.

Example: **delete tcp**

```
IP Addess [0.0.0.0]? 128.185.14.1
```

### Disable C M

Disables the DLSw protocol, an SDLC link station, the LLC disconnect functionality, and automatic TCP reconnection.

**Syntax:** disable

dlsw
llc disconnect on session loss
sdlc
auto-tcp-reconnect

**Note:** The **disable dlsw** command parameter works only in the configuration environment.

#### dlsw

Prevents the router from transmitting DLSw functions over all DLSw configured interfaces.

```
Example:  disable dls
```

#### llc

Prevents the router from terminating an LLC connection actively by issuing a DISC LLC frame when a DLSw session terminates.

This command does not affect switching functionality for LLC in DLSw. Use the **close-sap** command to stop LLC switching functionality.

```
Example:  disable llc
```

#### sdlc

Prevents DLSw connections to the specified SDLC link station.

If you enter this command in the console environment, it terminates the existing SDLC connection.

```
Example:  disable sdlc

Interface #[0]?
SDLC Address [C1]?
Record updated
```

**auto-tcp-reconnect**

Disables automatic TCP station re-establishment. When this feature is disabled, TCP sessions are not established until DLSw needs them.

Example: **disable auto**

## Enable C M

Enables the DLSw protocol, SDLC link stations, the LLC switching functionality, and auto-tcp-reconnect.

C In the configuration environment, **enable** works as follows:

**Syntax:** enable

dlsw
llc
sdlc
auto-tcp-reconnect

**dls**

Enables DLSw operation on the router.

Example: **enable dls**

**llc**

Allows the router to terminate an LLC connection upon the loss of the TCP connection.

Example: **enable llc**

**sdlc**

Enables DLSw connections to the specified SDLC link station.

Example: **enable sdlc**

```
Interface #[0]? 1
SDLC Address [C1]?
Record updated
```

**auto-tcp-reconnect**

Enables DLSw to automatically establish TCP sessions at startup and to reestablish a session when it breaks. The default is enabled.

Example: **enable auto-tcp-reconnect**

**M** In the console environment, **enable** works as follows:

**Syntax:** enable
llc
sdlc
auto-tcp-reconnect

**llc**

Allows the router to terminate an LLC connection upon the loss of the TCP connection.

Example: **enable llc**

**sdlc**

Enables DLSw connections to the specified SDLC link station.

Example: **enable sdlc**

```
Interface #[0]? 1
SDLC Address [C1]?
Record updated
```

**auto-tcp-reconnect**

Enables DLSw to automatically establish TCP sessions at startup and to re-establish a session when it breaks. The default is enabled.

Example: **enable auto**

### Join-Group C M

Allows DLSw neighbors to find and to create TCP sessions with each other dynamically. This eliminates the need to define TCP neighbors with the **add tcp** command.

There are three types of groups: Client, Server, and Peer-to-Peer. DLSw groups alleviate the need for long lists of static IP addresses and the costs associated with maintaining them. The IP internet being used must support multicast routing.

A DLSw router can be a member of a maximum of 64 groups. DLSw group membership uses the MOSPF protocol. To use the functionality of the **join-group** command, you must configure OSPF and MOSPF from the `OSPF Config>` prompt. Refer to the *Routing Protocols User's Guide*, Chapter 12 for more information.

When you assign a DLSw router to a group, the DLSw protocol automatically adds one of two addresses to the group number to form a multicast address. The router transmits the multicast address to identify itself to other group members and to transmit packets to those members. The two addresses that are added to the group number are **225.0.1.0** for DLSw clients and neighbors, and **225.0.1.64** for DLSw servers.

For example, the multicast address for client in group 2 would be 225.0.1.2.

**Syntax:** join-group

Example: **join-group**

```
Group ID (1-64 Decimal) [1]? 2
Client/Server or neighbor Group member (C/S/P)- [C]?
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal) [5120]?
Maximum Segment Size (Decimal) [1024]?
Enable/Disable Keepalive (E/D)- [D]?
Neighbor Priority (H/M/L) [M]?
```

| | |
|---|---|
| *Group ID* | The number of the group that you want this router to join. |
| *Client/Server or neighbor Group Member* | The type of group that you want to join, **C** for client, **S** for server, and **P** for peer-to-peer. A server forms a TCP connection with a client. |
| *Transmit Buffer Size* | The size of the packet transmit buffer in the range of 1024 to 32768. The default is 5120. |
| *Receive Buffer Size* | The size of the packet receive buffer between 1024 and 32768. The default size is 5120. |
| *Maximum Segment Size* | The maximum size of the TCP segment in the range of 64 to 16384. The default is 1024. |
| *Enable/Disable Keepalive* | Indicates whether you want the DLSw neighbor to send link keepalive messages. Default is **D** (Disable). |

| | |
|---|---|
| *Neighbor Priority (H/M/L) [M]?* | Allows you to specify the neighbor priority as either High, Medium, or Low. DLSw uses this parameter to determine which DLSw neighbor to choose when multiple neighbors can reach a target station. |

### Leave-Group C M

Removes the router from any specified DLSw groups that you configured with the **join-group** command.

C In the configuration environment, **leave-group** does not affect existing TCP connections belonging to the specified group.

**Syntax:** leave-group *group#*

```
Example:  leave-group 2
```

M In the console environment, **leave-group** terminates existing TCP connections belonging to the specified group.

**Syntax:** leave-group *group#*

```
Example:  leave-group 2
```

### List C M

Displays DLSw information on SDLC link stations, circuit priority, SAPs, TCP neighbors, and groups.

C In the configuration environment, **list** works as follows:

**Syntax:** list

dls
groups
llc2 sap parameters
open llc2 saps
priority
sdlc link stations

tcp neighbors

**dls**

Displays the information configured with the **enable** and **set** commands.

Example: **list dls**

```
DLSw is                            ENABLED
LLC2 send Disconnect is            ENABLED
Automatic TCP connection           ALWAYS CONNECT

SRB Segment number                 0030
MAC <-> IP mapping cache size      128
Max DLSw sessions                  3000
DLSw global memory allotment       60000
LLC per-session memory allotment   8192
SDLC per-session memory allotment  4096
NetBIOS UI-frame memory allotment  40960

Database age timer                 1200  seconds
Max wait timer for ICANREACH       20    seconds
Wait timer for LLC test response   15    seconds
Wait timer for SDLC test response  15    seconds
Join Group Interval                900   seconds
Neighbor priority wait timer       2.0   seconds
```

| | |
|---|---|
| *DLSw is* | Status of the DLSw protocol, enabled or disabled. |
| *LLC2 send Disconnect is* | Status of preventing the router from terminating an LLC2 connection upon the loss of the TCP connection. Values are enabled or disabled. |
| *SRB Segment number* | The SRB segment that identifies DLSw in the RIF. |
| *MAC <-> IP mapping cache size* | Specifies the size of the MAC-IP mapping cache. |
| *Max DLSw Sessions* | The maximum number of DLSw sessions that the router will support. |
| *DLSw global memory allotment* | The maximum amount of memory allowed for use by DLSw. |

| | |
|---|---|
| *LLC per-session memory allotment* | The maximum amount of memory allowed for use by each LLC session. |
| *SDLC per-session memory allotment* | The maximum amount of memory allowed for use by each SDLC DLSw session. |
| *NetBIOS UI-frame memory allotment* | The number of bytes the router allocates as a buffer for NetBIOS UI frames. |
| *Database age timer* | The maximum time to hold active database entries. |
| *Max wait timer for ICANREACH* | The time to wait for a response to a CANUREACH before giving up. |
| *Wait timer for LLC response* | The maximum amount of time (in seconds) the router waits for an LLC TEST response before retransmitting an LLC TEST frame. |
| *Wait timer for SDLC test response* | The maximum amount of time (in seconds) the router waits for an SDLC TEST response before retransmitting an SDLC TEST frame. |
| *Join group interval* | The amount of time (in seconds) between DLSw group advertisement broadcasts. |
| *Neighbor priority wait timer* | The amount of time DLSw waits before selecting a neighbor. |

#### groups

Displays group information for a DLSw neighbor previously configured with the **join-group** command.

Example: **list groups**

```
Group  Role    Xmit Bufsize  Rcv Bufsize Max Segsize  Keepalive   Priority
 1     CLIENT    5120          5120        1024      DISABLED    MEDIUM
```

*Group* The group number.

*Role* The type of group: Client, Server, or Peer-to-Peer.

*Xmit Bufsize* The size of the TCP transmit buffer in the range of 1024 to 32768. The default is 5120.

*Rcv Bufsize* The size of the TCP receive buffer in the range of 1024 to 32768. The default is 5120.

*Max Segsize* The maximum size of the TCP segment in the range of 64 to 16384. The default is 1024.

*Keepalive* The status of the keepalive functionality, enabled or disabled.

*Priority* The priority of the neighbor router in the selection process. Neighbor priority is either High, Medium, or Low.

#### llc2 sap parameters

Displays the LLC2 parameters configured with the **set llc2** command (see the **set** command for a complete explanation of these tunable parameters). These parameters are set for each interface. If no changes to the LLC2 parameters were made using the **set llc2** command, no output will be generated.

Example: **list llc2**

```
SAP   t1   t2   ti   n2   n3   tw   rw   nw   acc
 0     1    1   30    8    1    2    2    1    0
```

*SAP* SAP number.

*t1* Reply timer.

*t2* Receive Ack timer.

*ti* Inactivity timer.

| | |
|---|---|
| *n2* | Maximum retry value. |
| *n3* | Number of I-frames received before sending ACK. |
| *tw* | Transmit window. |
| *rw* | Receive window. |
| *nw* | ACKs needed to increment the working window (Ww). |
| *acc* | The current LLC2 implementation does not use access priority. As a result, this parameter always defaults to 0. |

**open**

Displays all open SAPs and their associated interfaces.

Example: **list open**

**priority**

Lists the circuit priorities selected for SNA and NetBIOS circuits, the transmit ratios between the various circuit priorities, and the largest frame size configured for NetBIOS.

Example: **list priority**

```
Priority for SNA DLSw sessions is           MEDIUM
Priority for NetBIOS DLSw sessions is       MEDIUM
Message allocation by C/H/M/L priority is   4/3/2/1
Maximum frame size for NetBIOS is           2052
```

Circuit priorities are Critical, High, Medium, or Low. The router uses the priority value you assign to selectively limit the burstlength of specific types of traffic. For example, if you assign SNA traffic a priority of Critical and NetBIOS traffic a priority of Medium, with a message allocation of 4/3/2/1, the router processes 4 SNA frames before it processes 2 NetBIOS frames. After the router processes 2 NetBIOS frames, it processes 4 SNA frames, and so on. In this scenario, two- thirds of available bandwidth is dedicated to SNA traffic (a ratio of 4 to 2). Note that the router counts frames, rather than bytes, when allocating bandwidth according to the priorities you assign.

### sdlc

Displays the SDLC link station information configured with the **add sdlc link station** command.

Example: **list sdlc**

```
Net  Adr  Status   Idblk   Idnum   Source   SAP/MAC       Dest  SAP/MAC
 5    C1  Enabled  017     A0021     04     4018997E05C1   04   401AA92000C1
```

*Net* The number of the interface that connects to the SDLC link station.

*Adr* The SDLC address, between 01 and FE, of the connecting link station.

*Status* The status, enabled or disabled, of the link station.

*Idblk* The 3-digit hexadecimal value that identifies the device (PU) that you are connecting. Normally, you will use Idblk for PUs on switched lines (as opposed to leased lines). Therefore, this value should match this same parameter in the VTAM Switched Major Node that corresponds to this PU.

*Idnum* The 5-digit hexadecimal value that identifies the specific SDLC PU type (2.0). Normally, you will use Idnum for PUs on switched lines (as opposed to leased lines). Therefore, this value should match this same parameter in the VTAM Switched Major Node that corresponds to this PU.

*Source SAP/MAC* The PU link to the DLSw domain and the MAC address of the local link station. The MAC address is in non-canonical bit order (Token-Ring) format. This is true even if the remote end station is on the Ethernet. Use the ASRT console **flip** command to flip the MAC address, in such cases.

*Dst SAP/ MAC* The remote side of the connection to the DLSw domain. If this SAP is 0, then the link station is in passive mode and does not attempt to establish a circuit. The MAC address is in non-canonical bit order (Token-Ring) format. This is true even if the remote end station is on the Ethernet. Use the ASRT console **flip** command to flip the MAC address, in such cases.

**tcp neighbors**

Displays configured DLSw neighbors that are TCP neighbors. The neighbors were configured with the **add tcp neighbor ip address** command.

Example: **list tcp**

```
Neighbor        Xmit Bufsize  Rcv Bufsize  Max Segsize  Keepalive  Priority
--------        ------------  -----------  -----------  ---------  --------
128.185.122.234     5120         5120         1024       DISABLED    MEDIUM
```

*Neighbor* The IP address of the TCP neighbor.

*Xmit buf size* The size of the packet transmit buffer in the range of 1024 to 32768. The default is 5120.

*Rcv Buf size* The size of the TCP receive buffer in the range of 1024 to 32768. The default is 5120.

*Max segment size* The maximum size of the TCP segment in the range of 64 to 16384. The default is 1024.

*Keepalive* Displays the status of the keepalive functionality, enabled or disabled.

*Priority* Displays the priority of the neighbor router in the selection process. Neighbor priority is either High, Medium, or Low.

**M** In the console environment, **list** works as follows:

**Syntax:** list

dls
dls global
dls sessions all
dls sessions ban
dls sessions dest
dls sessions detail
dls sessions ip
dls sessions nb
dls session range
dls session src
dls session state
dls cache all
dls cache range
dls memory
groups
llc2 open
llc2 sap
llc2 sessions all
llc2 sessions range
sdlc sessions
sdlc link
tcp capabilities
tcp config
tcp sessions
tcp statistics

**dls**

Displays information that pertains to the DLSw protocol. The options (*global, sessions, cache,* and *nb)* for the DLSw parameters appear below and on the following pages.

| | |
|---|---|
| *Global* | Displays status, timer, and MAC address information about the DLSw protocol. |
| *Sessions* | Displays current DLS session information including source, destination, state, flags, destination IP address, and ID. |
| *Cache* | Lists the addresses in the DLSw MAC address cache. |

| | |
|---|---|
| *NB* | Lists information on current active circuits that support NetBIOS. |

#### dls global

Displays DLS global parameter information.

Example: **list dls global**

```
DLSw is                            ENABLED
LLC2 send Disconnect is            ENABLED
Automatic TCP connection           ALWAYS CONNECT

SRB Segment number                 000
MAC <-> IP mapping cache size      128
Max DLSw sessions                  1000
DLSw global memory allotment       141056
LLC per-session memory allotment   8192
SDLC per-session memory allotment  4096
NetBIOS UI-frame memory allotment  40960


Database age timer                 1200  seconds
Max wait timer for ICANREACH       20    seconds
Wait timer for LLC test response   15    seconds
Wait timer for SDLC test response  15    seconds
Join Group Interval                900   seconds
Neighbor priority wait timer       2.0   seconds
```

| | |
|---|---|
| *DLSw is* | Status of the DLSw protocol, enabled or disabled. |
| *LLC2 send Disconnect is* | Status of preventing the router from terminating an LLC2 connection upon the loss of the TCP connection. Values are enabled or disabled. |
| *SRB Segment number* | The SRB segment that identifies DLSw in the RIF. |
| *MAC <-> IP mapping cache size* | The size of the MAC-IP mapping cache. |

| | |
|---|---|
| *Max DLSw Sessions* | The maximum number of DLSw sessions that the router can support. |
| *DLSw global memory allotment* | The maximum amount of memory allowed for use by DLSw. |
| *LLC per-session memory allotment* | The maximum amount of memory allowed for use by each LLC session. |
| *SDLC per-session memory allotment* | The maximum amount of memory allowed for use by each SDLC session. |
| *NetBIOS UI-frame memory allotment* | The number of bytes the router allocates as a buffer for NetBIOS UI frames. |
| *Database age timer* | The maximum time to hold active database entries. |
| *Max wait timer for ICANREACH* | The time to wait for a response to a CANUREACH before giving up. |
| *Wait timer for LLC test response* | The maximum amount of time (in seconds) the router waits for an LLC TEST response before retransmitting an LLC TEST frame. |
| *Wait timer for SDLC test response* | The maximum amount of time (in seconds) the router waits for an SDLC TEST response before retransmitting an SDLC TEST frame. |
| *Join Group Interval* | The amount of time (in seconds) between DLSw group advertisement broadcasts. |
| *Neighbor priority wait timer* | The amount of time to wait for ICANREACH responses before selecting a transport. |

**dls sessions all**

Displays current dls session information.

```
Example:  list dls sessions all

  Source             Destination         State       Flags   Dest. IP Addr   Id
1 400000000003 04    500000000003        Connected           128.185.236.51   0
```

*Source* The source MAC address of the session.

*Destination* The destination MAC address of the session.

*State* The current state of the session:

DISCONNECT The initial state with no circuit or connection established.

RSLV_PEND The target DLSw is waiting for a STARTED indication following a START request.

CIRC_PEND The target DLSw is waiting for a REACHACK response to an ICANREACH message.

CIRC_EST The end-to-end circuit has been established.

CIR_RSTRT The DLSw that originated the reset is awaiting the restart of the data link and an RESTARTED response to an RESTART message.

CONN_PEND The origin DLSw is awaiting an CONTACTED response to an CONTACT message.

CONT_PEND The target DLSw is awaiting an CONTACTED confirmation to an CONTACT message.

CONNECTED The origin DLSw is awaiting an CONTACTED response to an CONTACT message.

DISC_PEND The DLSw that originated the disconnect is awaiting an HALTED response to an HALT message.

| | |
|---|---|
| HALT_PEND | The remote DLSw is awaiting an HALTED indication following an HALT request. |
| REST_PEND | The remote DLSw is awaiting an HALTED indication following an HALT request. |
| WAIT_NOACK | The local DLSw is performing DLC reset functions prior to sending HALT_DL_NOACK to the target DLSw. |
| CIR_STRT | The origin DLSw is awaiting a ICANREACH_cs in response to a CANUREACH_cs message. |
| HALT_NOACK | The remote DLSw is awaiting the DLC_DL_HALTED indication following the DLC_HALT_DL request. |

*Flags* Flags can optionally be:

**A** – CONTACT MSG PENDING
**B** – SAP RESOLVE PENDING
**C** – EXIT BUSY EXPECTED
**D** – TCP BUSY
**E** – DELETE PENDING
**F** – CIRCUIT INACTIVE

*Dest. IP Addr* The IP address of the remote DLSw peer.

*Id* The number used to identify the session. Use this number in any command that requires the session identifier.

#### dls session ban

Displays current information on BAN sessions.

Example: **list dls session ban**

```
BAN port number (user 0 for all ports) [0]?
No active sessions
```

#### dls session dest

Displays DLS session information by destination MAC address.

Example: **list dls session dest**

```
Destination MAC Address [40000000001]? 50000000003

   Source         Destination    State       Flags  Dest. IP Addr   Id
-----------------------------------------------------------------------
1. 400000000003  500000000003    Connected          128.185.236.51  2
2. 400000000002  500000000003    Connected          128.185.236.52  3
```

#### dls session detail

Displays detailed DLS session information.

Example: **list dls session detail**

```
Session Identifier [1]? 2

    Source        Destination       State       Flags    Dest. IP Addr  Id
--------------------------------------------------------------------------
1   400000000003  500000000003 04   Connected            128.185.236.51  2

        Personality:          TARGET
        XIDs sent:            2
        XIDs rcvd:            0
        Datagrams sent:       0
        Datagrams rcvd:       0
        Info frames sent:     15
        Info frames rcvd:     0
        RIF:                  0620 0202 B0B0
        Local CID:
```

| | |
|---|---|
| *Personality* | The ORIGINATOR (initiator) or TARGET (recipient) of the connection. |
| *XIDs sent*<br>*XIDs rcvd* | XIDs that this DLSw peer has sent and received from the remote DLSw peer. |
| *Datagrams sent*<br>*Datagrams rcvd* | Datagrams that this DLSw peer has sent and received from the remote DLSw peer. |

| | |
|---|---|
| *Info frames sent* *Info frames rcvd* | I-frames that this DLSw peer has sent and received from the DLSw peer. |
| *RIF* | The information that is included in the RIF of the LLC test frame. |

**dls session ip**

Displays IP session information.

Example: **list dls session ip**

```
Enter the DLSw neighbor IP address [0.0.0.0]?

  Source        Destination       State      Flags   Dest. IP Addr   Id
 --------------------------------------------------------------------
1 400000000003  500000000003 04   Connected          128.185.236.51  2
```

**dls sessions nb**

Lists information about the current active circuits that support NetBIOS.

Example: **list dls sessions nb**

```
  Source           Destination       State    Flags   Dest. IP Addr    Id
 ------------------------------------------------------------------------
1 0000C91373C1 F0  0003152CCCE6 F0   Connected        128.185.236.245  92
```

**dls session range**

Displays the range of DLS sessions that you want to display. This number is located to the left of the source MAC address.

Example: **list dls session range**

```
Start [1]?
Stop [1]?

  Source         Destination      State      Flags    Dest. IP Addr   Id
-------------------------------------------------------------------------
1 400000000003  500000000003 04   Connected           128.185.236.51  2
```

**dls session src**

Displays all DLSw session information by source MAC Address.

Example: **list dls session src**

```
Source MAC Address [400000000001]?

   Source      Destination      State      Flags   Dest. IP Addr    Id
-----------------------------------------------------------------------
1  SDLC 04-C4  400000000002 04  Connected          10.1.49.401       1
```

**Note:** In this example, source MAC address 400000000001 maps to the "SDLC 04" name. If you do not know the source MAC address, enter **list SDLC config all** to obtain it.

**dls session state**

Displays all DLSw sessions in the specified state. The DLSw session states are defined as follows:

Example: **list dls session state**

```
DISCONNECT = 0    RSLV_PEND = 1
CIRC_PEND = 2     CIRC_EST = 3
CIRC_RSTRT = 4    CONN_PEND = 5
CONT_PEND = 6     CONNECTED = 7
DISC_PEND = 8     HALT_PEND = 9
REST_PEND = 10    WAIT_NOACK = 11
CIRC_STRT = 12    HLT_NOACK = 12
Enter state value [7]?
   Source            Destination    State     Flags  Dest. IP Addr   Id
-----------------------------------------------------------------------
1 400000000003 04   10005AF181A4 04 Connected        128.185.236.84   0
2 400000000002 04   400000000088 04 Connected        128.185.236.84   1
```

**dls cache all**

Lists the entries in the DLSw MAC address cache. This cache contains a database of the most recent MAC address to IP neighbor translations. It provides the MAC address, time to live (in seconds) in the cache, and the neighbor's IP address.

Example: **list dls cache all**

```
   Mac Address     Secs to live      IP Address(es)      Largest Frame
 1 10005AF1809B    810               128.185.236.84      1470
 2 10005AF181A4    1170              128.185.236.84      2052
 3 400000000088    1170              128.185.236.84      2052
```

**dls cache range**

Displays information for a specified range of cache entries.

Example: **list dls cache range**

```
Start[2]?
Stop[2]?
 Mac AddressSecs to live  IP Address(es)      Largest Frame
2  10005AF181A4 1170       128.185.236.84       2052
```

### dls memory

Lists all existing DLSw sessions and the amount of memory in use by each session. It also displays the following flow control states:

*Ready* The session is not congested.

*Session* The session has used most of its session allotment and probably has flow controlled the data link.

*Global* The session is congested due to a shortage of memory in the router.

The *currently in use* field shows the total amount of memory currently allocated by DLS. This includes all session allocations, control messages, and TCP receive buffers.

**Note:** You can change the memory allocation using the **set memory** command.

Example: **list dls memory**

```
Total DLSw bytes requested:             141056
Global receive pool bytes granted:       84633
  Currently in use:                          0
Global transmit pool bytes granted:      56423
  Currently in use:                        232

ID   Source        Destination    Initial     Current   Congest   DLC Xmits
                                   Alloc       Free       State    Queued
--   ------        -----------    ------      ------    ------   --------
11  400000000003   10005AF181A4    4096         4096     READY        0
```

### groups

Displays information for all configured groups to which the router belongs.

```
Example:  list groups

Group  Role    Xmit Bufsize  Rcv Bufsize  Max Segsize  Keepalive  Priority
 1     CLIENT    5120          5120          1024      DISABLED   MEDIUM
```

*Group* — Number of the group.

*Role* — Type of group.

*Xmit Bufsize* — Size of the TCP transmit buffer in the range of 1024 to 32768. The transmit buffer size must be at least twice the maximum segment size. Default is 5120.

*Rcv Bufsize* — Size of the TCP receive buffer in the range of 1024 to 32768. The receive buffer size must be at least twice the maximum segment size. Default is 5120.

*Max Segsize* — Maximum size of the TCP segment, in the range of 64 to 16384. The default is 1024.

*Keepalive* — Status of the keepalive functionality, enabled or disabled.

*Priority* — The priority of the DLSw group displayed as either High, Medium, or Low.

**llc2 open**

Displays information that pertains to LLC2. The options (*open SAPs, SAP parameters,* and *sessions)* for LLC2 are described below and on the following pages.

*Open* — Displays information for all currently open SAPs on interfaces between LLC2 peers.

```
Example:  list llc2 open

Interface    SAP
    0         0
    0         4
```

**llc2 sap**

Displays LLC2 parameter configuration information. Only configurations that were changed will be displayed. If you did not use the **set llc2** command, no output is generated. If no values are set, the system will display ”No SAP parameters have been modified - all are using default.”

Example: **list llc2 sap**

```
SAP           T1    t2    ti    n2    n3    tw    rw    nw    acc
---           --    --    --    --    --    --    --    --    ---
4             1     1     30    8     1     2     2     1      0
```

**llc2 sessions all**

Displays current information for all LLC2 sessions.

Example: **list llc2 sessions all**

```
Start [1]?
Stop  [1]?

   SAP  Int.      Remote Addr    Local Addr       State         RIF
1. 04   6         400000000003   500000000003     CONTACTED     0620 0202 B0B0
```

*State* The state of the llc session. The following states can be displayed:

*Disconnected* The state indicating that the data link control structure exists but no data link is established.

*Connect_pend* The connect pending state is entered when a test command frame to NULL SAP is received or when a DLC_START_DL command is received from DLSw.

*Resolve_pend* The resolve pending state is entered when a DLC_RESOLVE_C command has been sent to DLSw.

*Connected* This is a steady state where LLC Type 1 level services are available through the DLSw cloud. This state is entered when a DLC_RESOLVE_R command is received from DLSw or when a TEST response frame is received from the network.

*Contact_pend* This state is entered whenever a response to a transmitted or received SABME is outstanding.

*Contacted* In an active DLSw session, you can pass data on the session. This is the normal operating state.

*Disc_Pend* This state is entered whenever a DISC command has been transmitted or received, or a DLC_HALT has been received from DLSw.

#### llc2 sessions range

Displays current information for the selected range of LLC2 sessions.

Example: **list llc2 sessions range**

```
   SAP  Int.      Remote Addr    Local Addr     State         RIF
1. 04   6         400000000003   500000000003   Contacted     0620 0202 B0B0
```

#### sdlc sessions

Displays information about all SDLC DLS sessions within the router.

Example: **list sdlc sessions**

```
   Net  Addr  Source SAP/MAC     OutQ   State
1  2    41    04 40002DECD150     0    Contacted
```

### sdlc config

Displays configured parameters for the SDLC attached PU.

Example: **list sdlc config**

```
Interface #, or 'ALL' [0]? 5
Net  Addr     Status   Idblk  Idnum  Source SAP/MAC     Dest SAP/MAC
 5     C1     Enabled    000  0000   04 4018997E05C1    04 401AAB9200C1
```

### tcp capabilities

Displays the information received from a partner DLSw router in the capabilities exchange message.

Example: **list tcp capabilities**

```
Enter the DLSw neighbor IP address [0.0.0.0]? 1.1.1.2
Vendor ID:               1000DC
Vendor version:          Digital Distributed Router v2.0
Initial pacing window:   12
Preferred TCP connections: 1
Supported SAPs:          00 04 08 0C
```

### tcp config

Displays information on all configured TCP sessions.

Example: **list tcp config**

```
Neighbor         Xmit Bufsize Rcv Bufsize  Max Segsize  Keepalive  Priority
--------         ------------ -----------  -----------  ---------  ------
128.185.122.234     5120         5120          1024     DISABLED   MEDIUM
```

### tcp sessions

Displays the DLSw protocol version, the number of currently active DLSw sessions using this TCP session, and the number of DLSw sessions that have ever used this TCP session.

```
Example:  list tcp sessions

Group   IP Address       Conn State   Version   Active Sess  Sess Creates
-----   -----------      -----------  -------   -----------  -----------
1       128.185.122.234  ESTABLISHED  AIW V1R0       2             4
```

**tcp statistics**

Displays statistics on the use of TCP sessions.

```
Example:  list tcp statistics

Enter the DLSw neighbor IP Address [0.0.0.0]? 1.1.1.1

                                     Transmitted          Received
                                     -----------         -----------
Data Messages                            217                 314
Data Bytes                             31648               43796
Control Messages                          64                  74

CanYouReach Explorer Messages             6                   0
ICanReach Explorer Messages               0                   4
NameQuery Explorer Messages               0                   0
NameRecognized Explorer Messages          0                   0
```

### Netbios C M

Displays the NetBIOS configuration or console prompt. See the *Bridging Guide* for complete information on the commands available at this prompt.

**Syntax:** netbios

```
Example:  netbios

NetBIOS Support User Configuration

NetBIOS config>
```

### Open-Sap C M

Enables the transmitting of data for the specified link SAP by the DLSw protocol. Note that **open-sap** is not necessary for an SDLC interface (only SNA, NB, or LLC).

You should execute the **open-sap** command on the router that resides on the session initiator side of the connection. For example, if the client is always the sessions initiator, you need to open the SAPs only on the client side router. If you are unsure which side initiates the connection, open the SAPs on both sides. The commonly used SNA SAP values are 04, 08, and 0C. Digital recommends that you open 04, 08, and 0C on all participating DLSw routers.

**Syntax:** open-sap

Example: **open-sap**

```
Interface # [0]?
Enter SAP in hex (range 0-F0), 'SNA' or 'NB' [4]? sna
SAP F4 opened on interface 0
```

*Interface #* — The number of the interface over which you want to open the SAP.

*Enter SAP in hex* — You can enter individual SAPs in hex, or you can enter SNA or NB (NetBIOS).

- SNA opens SAPs 0, 4, 8, and C
- NB opens SAP F0 for NetBIOS

If you enter SAPs in hex, the range is 0 to F0, and the SAP must be an even number.

### Set C M

Configures the size of the MAC address-to-IP address mapping cache, LLC2 parameters, maximum number of DLSw sessions, SRB segment number, protocol timers, and TCP receive buffer size.

**C** In the configuration environment, **set** works as follows:

**Syntax:** set

cache
llc2
maximum
memory

priority
srb
timers

**cache** ***size***

Lets you specify the size of the MAC address-to-IP address mapping cache.

DLSw uses information stored in this cache to discover routes to remote stations. Thus, the larger the cache, the better the chances of DLSw finding a desired remote station without broadcasting CANUREACH frames to all known TCP/IP neighbors.

Nonetheless, it is wise to avoid setting this cache size too large. Doing so will consume memory in the router. The effect will be a reduction in the number of DLSw sessions the router can handle.

Example: **set cache**

```
MAC <-> IP cache size (4 - 65535) [128]?
```

**llc2**

Allows you to configure specific LLC2 attributes for a specific SAP.

Example: **set llc2**

```
Enter SAP in hex (range 0-FE) [0]?
Reply timer (T1) in sec. [1]?
Receive ack timer (T2) in 100 millisec. [1]?
Inactivity timer (Ti) in sec. [30]?
Transmit window (Tw) 1-128 0=default [2]?
Receive window (Rw) 127 Max [2]?
Acks needed to increment (Ww) (Nw) [1]?
Max retry value (N2) [8]?
Number I-frames recvd before sending ACK (N3) [1]?
```

| | |
|---|---|
| *Enter SAP in hex* | The SAP number that you want to tune. Values in the range of 0 - F0. |
| *Reply timer (T1)* | This timer expires when the LLC2 neighbor fails to receive a required acknowledgment or response from the other LLC2 neighbor. |

| | |
|---|---|
| *Receive Ack timer (T2)* | The delay it takes to send an acknowledgment for a received I-format frame in milliseconds. |
| *Inactivity Timer (Ti)* | This timer expires when the LLC does not receive a frame for a specified time period. When this timer expires, the LLC2 neighbor transmits an RR until the LLC2 neighbor responds or the N2 retry count is exceeded. Default is 30 seconds. |
| *Transmit Window (Tw)* | The maximum number of I-frames that can be sent before receiving an RR. Values in the range 1 - 127. 0 sets Tw to the default. Default is 2. |
| *Receive Window (Rw)* | The maximum number of unacknowledged sequentially numbered I-frames that an LLC2 neighbor can receive from a remote host. |
| *Acks needed to increment Ww (Nw)* | The working window (Ww) is a dynamically changing shadow of the transmit window (Tw). After an LLC error is detected, the working window (Ww) is reset to 1. The ‘Acks needed to increment Ww’ value specifies the number of acks that the station must receive before incrementing Ww by 1. The Ww continues to be incremented in this fashion until Ww = Tw. |
| *Max Retry value (N2)* | The maximum number of times the LLC2 neighbor transmits an RR without receiving an acknowledgment when the inactivity timer (Ti) expires. |
| *Number I-frames received before sending ACK (N3)* | The value used with the T2 timer to reduce acknowledgment traffic for received I-frames. This counter is set to a specified value and decrements each time an I-frame is received. When this counter reaches 0 or the T2 timer expires, an acknowledgment is sent. The default is 1. To ensure good performance, set N3 to a value less than the remote LLC’s Tw. |

**maximum** ***#-of-sessions***

Sets the maximum number of concurrent DLSw sessions that the router can support. The default is 1000; increasing this number will impact router memory resources.

Example: **set maximum**

```
Max number of DLSw sessions (1-60000) [1000]?
```

**memory**

Allows you to specify the total amount of memory allocated to DLSw, and the total amount of memory to be allotted to each DLSw session.

Example: **set memory**

```
Number of bytes to allocate for DLSw (at least 26368) [141056]?
Number of bytes to allocate per LLC session [8192]?
Number of bytes to allocate per SDLC session [4096]?
Number of bytes to allocate for NetBIOS UI-frames [40960]?
```

The default for the number of bytes to allocate to DLSw is probably too low to be useful for more than a small number of DLSw sessions. Raise the memory value depending on the anticipated number of DLSw sessions, the TCP neighbors, and the amount of memory available in the router.

The maximum memory required by a single session is approximately the following:

session_allocation * number_of_sessions * 75%

Adjust this number to 80-85% if the data stream includes many small packets.

Each TCP connection to a DLSw neighbor requires roughly 512 bytes.

For example, assuming 8K per LLC session and 4K per SDLC session, a total of 100 DLSw sessions (20 SDLC and 80 LLC) through a combination of 4 DLSw neighbors requires approximately

(20 * 4K * 75%) + (80 * 8K * 75%) + (4 * 512) = 555,008 bytes

If you anticipate many small packets, then

(20 * 4K * 85%) + (80 * 8K * 85%) + (4 * 512) = 628,736 bytes

At no point does bad judgment in determining the DLSw allocation result in lost data. In general, the more memory allocated to DLSw, the better the overall DLSw performance. When DLSw runs out of memory, an ELS message, DLS.161 (Entering GLOBAL congestion on global DLS pool), is generated. It is acceptable for these messages to appear occasionally. If they appear very often, consider increasing the DLSw allocation value.

**priority**

Lets you specify the circuit priorities to use for SNA circuits and NetBIOS circuits. You can use this command to specify circuit priority as Critical, High, Medium, or Low. Note that you must assign circuit priorities in descending order from Critical to Low.

The router uses the priority value you assign to selectively limit the burstlength of specific types of traffic. For example, if you assign SNA traffic a priority of Critical and NetBIOS traffic a priority of Medium, with a message allocation of 4/3/2/1, the router processes 4 SNA frames before it processes 2 NetBIOS frames. After the router processes 2 NetBIOS frames, it processes 4 SNA frames, and so on. In this scenario, two-thirds of available bandwidth is dedicated to SNA traffic (a ratio of 4 to 2). Note that the router counts frames, rather than bytes, when allocating bandwidth according to the priorities you assign.

You can also use this command to set the maximum frame size to use for NetBIOS. Set this parameter to the largest frame size you expect to need, and no larger. Setting the frame size larger than needed reduces the number of available buffers.

Example: **set priority**

```
Priority for SNA DLSw sessions (C/H/M/L) [M]?
Priority for NetBIOS DLSw sessions (C/H/M/L) [M]?
Message allocation by C/H/M/L priority (4 digits) [4/3/2/1]?
Maximum NetBIOS frame size (516, 1470, 2052, or 4399) [2052]?
```

**srb** ***segment-number***

Sets the Source Routing Bridge (SRB) segment number that identifies DLSw on Token Ring networks. Specify the segment number as a three-digit hexadecimal value.

Example: **set srb**

```
Enter segment number hex (1-FFF) [0]?
```

**timers**

Sets the DLSw protocol timers.

```
Example:  set timers

Database age timer (1-1000 secs. Decimal) [1200]?
Max wait timer ICANREACH (1-1000 secs Decimal) [20]?
Wait timer LLC test response (1-1000 secs. Decimal) [15]?
Wait timer SDLC test response (1-1000 secs. Decimal) [15]?
Group join timer interval (1-60000 secs. Decimal) [900]?
Neighbor priority wait timer (1.0-5.0 secs. Decimal) [2.0]?
```

| | |
|---|---|
| *Database age timer* | Indicates how long to hold unused DLSw database entries. Database entries map destination MAC addresses into the set of DLSw neighbors that can reach them. |
| *Max wait timer ICANREACH* | Indicates how long to wait for an ICANREACH response for a previously transmitted CANUREACH. |
| *Wait timer LLC test response* | Indicates how long to wait for an LLC test response before giving up. |
| *Wait timer SDLC test response* | Indicates how long to wait for an SDLC test response before giving up. |

*Group join timer interval*

The group interval timer is significant when you configure a pair of DLSw routers to use a TCP group with the **join** command, rather than statically configuring each router with the adjacent IP address of its DLS neighbor using the **add tcp** command.

When you use **set timer** from the `DLSw>` prompt, you are prompted for a group update interval value. When the router is first powered up, it sends group packets every 15 seconds or the configured group update interval, whichever is smaller, for the first 6 transmissions, and then the configured time thereafter.

If an IP router between two partner DLSw routers goes down, the attempt to reestablish the TCP connection takes place once the configured group update interval value has elapsed after the IP router has recovered. If the configured value is 15 seconds, then the attempt to reestablish the TCP connection takes place 15 seconds after the recovery of the IP router is detected.

The range is 1 to 60000 seconds in decimal. The default is 900 seconds.

*Neighbor priority wait timer*

Amount of time (in seconds) to wait during exploration before selecting a neighbor.

**M** In the console environment, **set** works as follows:

**Syntax:** set

llc2
memory
priority
timers

**llc2**

Allows you to configure specific LLC2 attributes for a specific SAP.

Example: **set llc2**

```
Enter SAP in hex (range 0-FE) [0]?
Reply timer (T1) in sec. [1]?
Receive ack timer (T2) in 100 millisec. [1]?
Inactivity timer (Ti) in sec. [30]?
Transmit window (Tw) 1-128 0=default [2]?
Receive window (Rw) 127 Max [2]?
Acks needed to increment (Ww) (Nw) [1]?
Max retry value (N2) [8]?
Number I-frames recvd before sending ACK (N3) [1]?
```

*Enter SAP in hex* — The SAP number that you want to tune. Values in the range of 0 - F0.

*Reply timer (T1)* — This timer expires when the LLC2 neighbor fails to receive a required acknowledgment or response from the other LLC2 neighbor.

*Receive Ack timer (T2)* — The delay it takes to send an acknowledgment for a received I-format frame in milliseconds.

*Inactivity Timer (Ti)* — This timer expires when the LLC does not receive a frame for a specified time period. When this timer expires, the LLC2 neighbor transmits an RR until the LLC2 neighbor responds or the N2 retry count is exceeded. Default is 30 seconds.

*Transmit Window (Tw)* — The maximum number of I-frames that can be sent before receiving an RR. Values in the range 1 - 127. 0 sets Tw to the default. Default is 2.

*Receive Window (Rw)* — The maximum number of unacknowledged sequentially numbered I-frames that an LLC2 neighbor can receive from a remote host.

| | |
|---|---|
| *Acks needed to increment Ww (Nw)* | The working window (Ww) is a dynamically changing shadow of the transmit window (Tw). After an LLC error is detected, the working window (Ww) is reset to 1. The ‘Acks needed to increment Ww’ value specifies the number of acks that the station must receive before incrementing Ww by 1. The Ww continues to be incremented in this fashion until Ww = Tw. |
| *Max Retry value (N2)* | The maximum number of times the LLC2 neighbor transmits an RR without receiving an acknowledgment when the inactivity timer (Ti) expires. |
| *Number I-frames received before sending ACK (N3)* | The value used with the T2 timer to reduce acknowledgment traffic for received I-frames. This counter is set to a specified value and decrements each time an I-frame is received. When this counter reaches 0 or the T2 timer expires, an acknowledgment is sent. The default is 1. To ensure good performance, set N3 to a value less than the remote LLC’s Tw. |

#### memory

Allows you to specify the total amount of memory allocated to DLSw, and the total amount of memory to be allotted to each DLSw session.

Example: **set memory**

```
Number of bytes to allocate for DLSw (at least 26368) [141056]?
Number of bytes to allocate per LLC session [8192]?
Number of bytes to allocate per SDLC session [4096]?
Number of bytes to allocate for NetBIOS UI-frames [40960]?
```

The default for the number of bytes to allocate to DLSw is probably too low to be useful for more than a small number of DLSw sessions. Raise the memory value depending on the anticipated number of DLSw sessions, the TCP neighbors, and the amount of memory available in the router.

The maximum memory required by a single session is approximately the following:

session_allocation * number_of_sessions * 75%

Adjust this number to 80-85% if the data stream includes many small packets.

Each TCP connection to a DLSw neighbor requires roughly 512 bytes.

For example, assuming 8K per LLC session and 4K per SDLC session, a total of 100 DLSw sessions (20 SDLC and 80 LLC) through a combination of 4 DLSw neighbors requires approximately

(20 * 4K * 75%) + (80 * 8K * 75%) + (4 * 512) = 555,008 bytes

If you anticipate many small packets, then

(20 * 4K * 85%) + (80 * 8K * 85%) + (4 * 512) = 628,736 bytes

At no point does bad judgment in determining the DLSw allocation result in lost data. In general, the more memory allocated to DLSw, the better the overall DLSw performance. When DLSw runs out of memory, an ELS message, DLS.161 (Entering GLOBAL congestion on global DLS pool), is generated. It is acceptable for these messages to appear occasionally. If they appear very often, consider increasing the DLSw allocation value.

**priority**

Lets you specify the circuit priorities to use for SNA circuits and NetBIOS circuits. You can use this command to specify circuit priority as Critical, High, Medium, or Low. Note that you must assign circuit priorities in descending order from Critical to Low.

The router uses the priority value you assign to selectively limit the burstlength of specific types of traffic. For example, if you assign SNA traffic a priority of Critical and NetBIOS traffic a priority of Medium, with a message allocation of 4/3/2/1, the router processes 4 SNA frames before it processes 2 NetBIOS frames. After the router processes 2 NetBIOS frames, it processes 4 SNA frames, and so on. In this scenario, two-thirds of available bandwidth is dedicated to SNA traffic (a ratio of 4 to 2). Note that the router counts frames, rather than bytes, when allocating bandwidth according to the priorities you assign.

You can also use this command to set the maximum frame size to use for NetBIOS. Set this parameter to the largest frame size you expect to need, and no larger. Setting the frame size larger than needed reduces the number of available buffers.

Example: **set priority**

```
Priority for SNA DLSw sessions (C/H/M/L) [M]?
Priority for NetBIOS DLSw sessions (C/H/M/L) [M]?
Message allocation by C/H/M/L priority (4 digits) [4/3/2/1]?
Maximum NetBIOS frame size (516, 1470, 2052, or 4399) [2052]?
```

**timers**

Sets the DLSw protocol timers.

Example: **set timers**

```
Database age timer (1-1000 secs. Decimal) [1200]?
Max wait timer ICANREACH (1-1000 secs Decimal) [20]?
Wait timer LLC test response (1-1000 secs. Decimal) [15]?
Wait timer SDLC test response (1-1000 secs. Decimal) [15]?
Group join timer interval (1-60000 secs. Decimal) [900]?
Neighbor priority wait timer (1.0-5.0 secs. Decimal) [2.0]?
```

| | |
|---|---|
| *Database age timer* | Indicates how long to hold unused DLSw database entries. Database entries map destination MAC addresses into the set of DLSw neighbors that can reach them. |
| *Max wait timer ICANREACH* | Indicates how long to wait for an ICANREACH response for a previously transmitted CANUREACH. |
| *Wait timer LLC test response* | Indicates how long to wait for an LLC test response before giving up. |
| *Wait timer SDLC test response* | Indicates how long to wait for an SDLC test response before giving up. |

| | |
|---|---|
| *Group join timer interval* | The group interval timer is significant when you configure a pair of DLSw routers to use a TCP group with the **join** command, rather than statically configuring each router with the adjacent IP address of its DLS neighbor using the **add tcp** command.<br><br>When you use **set timer** from the `DLSw>` prompt, you are prompted for a group update interval value. When the router is first powered up, it sends group packets every 15 seconds or the configured group update interval, whichever is smaller, for the first 6 transmissions, and then the configured time thereafter.<br><br>If an IP router between two partner DLSw routers goes down, the attempt to reestablish the TCP connection takes place once the configured group update interval value has elapsed after the IP router has recovered. If the configured value is 15 seconds, then the attempt to reestablish the TCP connection takes place 15 seconds after the recovery of the IP router is detected.<br><br>The range is 1 to 60000 seconds in decimal. The default is 900 seconds. |
| *Neighbor priority wait timer* | Amount of time (in seconds) to wait during exploration before selecting a neighbor. |

### Exit C M

Use the **exit** command to return to the DLSw `Config>` or DLSw> console prompt.

Example: **exit**

# 3

# Configuring and Monitoring SDLC Interfaces

This chapter describes the configuration and console commands applicable to SDLC. Refer to Chapter 1 for more information about SDLC.

## 3.1 About SDLC Configuration and Console Commands

SDLC must first be associated with an interface using the **set data-link** command before entering configuration commands. The SDLC configuration commands are available at the `SDLC # Config>` prompt, where # identifies the interface you specify with the **network** command. Changes made to the router's configuration do not take effect immediately; they become part of the router's nonvolatile configuration memory for use when the router restarts.

Conversely, SDLC console commands entered within the SDLC console module take effect immediately. However, changes made with console commands do *not* become part of the router's nonvolatile configuration.

When the router restarts, the configuration stored in configuration memory supersedes the effects of console commands.

Monitoring consists of these actions:

- Monitoring the protocols and network interfaces the router is using.
- Making real-time changes to the SDLC configuration without permanently affecting the router's nonvolatile configuration memory.
- Displaying ELS (Event Logging System) messages relating to router activities and performance.

## 3.2 Accessing the SDLC Configuration Environment

To enter the configuration process, follow these steps:

1. At the * prompt, enter **talk 6** or just **t 6**. This brings you to the `Config>` prompt.

```
* talk 6

Config>
```

If the `Config>` prompt does not appear immediately, press RET again.

2. Enter **network** and the number of an SDLC interface that you configured earlier using the **set data-link sdlc** command.

```
Config>network 3
SDLC 3 Config>
```

## 3.3 Accessing the SDLC Console Environment

To enter the SDLC console process, follow these steps:

1. At the * prompt, enter **talk 5** or just **t 5**. This brings you to the + prompt.

```
t 5

+
```

2. At the + prompt, enter **network** and the number of an SDLC interface that you configured earlier using the **set data-link sdlc** command.

```
+ network 3
SDLC Console
SDLC-3>
```

Note that the configuration and console prompts differ, allowing you to easily determine the environment you are in.

### Displaying Statistics on SDLC Interfaces

You can use the **interface** command to display statistics for SDLC devices. To do this, enter **interface** and an interface number at the + prompt:

Example: **+ interface 3**

```
                              Self-Test  Self-Test            Maintenance
Nt  Nti  Interface      CSR   Vec    Passed     Failed           Failed
3   FR    SDLC/1    80000000   5C       1          0                0


SDLC MAC/data-link on SCC Serial Line interface.

Level converter:  RS-232/V.35   Adapter cable:  RS-232 DCE

V.24 circuit:  105    106     107   108   109
Nicknames:     RTS    CTS     DSR   DTR   DCD
```

```
RS-232 DCE:    CA     CB      CC     CD     CF
State:         OFF    OFF     OFF    OFF    OFF

Line speed (configured):  9.615 Kbps
Last port reset:   1 minute, 24 seconds ago
Input frame errors:
    CRC error               0       alignment (byte length)     0
    Too short (< 2 bytes)   0       Too long (> 2051 bytes)     0
    aborted frame           0       DMA/FIFO overrun            0

Output frame errors:
   DMA/FIFO Underrun errors 0       Outputs aborts sent         0
```

| | |
|---|---|
| *Nt* | Interface number assigned by software during initial configuration. |
| *Nt*i | Interface number assigned by software during initial configuration. |
| *Interface* | Interface type. |
| *CSR* | Memory location of the control status register for the SDLC interface. |
| *Self-Test Passed* | Number of times the SDLC interface passed its self-test. |
| *Self-Test Failed* | Number of times the SDLC interface was unable to pass its self-test. |
| *Maintenance Failed* | Number of maintenance failures. |

The following six parameters only appear if a cable is connected, and varies according to cable type:

| | |
|---|---|
| *Level converter* | Type of level converter connected to the SDLC interface. |
| *Adapter cable* | Type of adapter cable that the level converter is using. |

| | |
|---|---|
| *V.24 circuit* | Circuits in use on the V.24 circuit. |
| *Nicknames* | Signals in use on the V.24 circuit. |
| *RS-232 DCE* | Current level converter is RS-232 DCE. |
| *State* | State of V24 circuits, signals, and pin assignments (ON or OFF). |

*Line speed (configured)* — Currently configured line speed for the SDLC interface.

*Last port reset* — How long ago the port was last reset.

*Input frame errors* — Input frame error type (CRC error, too short, aborted, alignment, too long, DMA/FIFO overrun) and the total number of errors that have occurred.

*Output frame counters* — Total number of DMA/FIFO overruns and output aborts sent for output frames.

## 3.4 SDLC Commands

This section summarizes and describes SDLC configuration and console commands.

Table 3–1 lists SDLC configuration and console commands and their functions.

**Table 3–1 SDLC Command Summary**

| Command | Task | Function |
|---|---|---|
| **? (Help)** | Configure/ Monitor | Lists the configuration and console commands or lists any parameters associated with the command. |
| **Add** | Configure/ Monitor | Adds an SDLC link station. |
| **Clear** | Monitor | Clears link or link station counters. |
| **Delete** | Configure | Removes an SDLC link station. |
| **Disable** | Configure/ Monitor | Prevents connections to an SDLC link station. |
| **Enable** | Configure/ Monitor | Allows connections to an SDLC link station. |
| **List** | Configure/ Monitor | Displays SDLC link station configurations and link counters. |
| **Set** | Configure/ Monitor | Configures specific interface and link station information. |
| **Test** | Monitor | Performs an echo test on a link station. |
| **Exit** | Configure/ Monitor | Exits the SDLC configuration or console environment. |

### ? (Help) C M

Lists the commands that are available from the current prompt. You can also enter **?** after a specific command to list its options.

**Syntax:** ?

Example: **?**

```
SET
ADD
DISABLE
DELETE
ENABLE
LIST
EXIT
```

### Add C M

Use the SDLC stations that you configure in DLSw or use the **add station** command to explicitly define SDLC stations for the following situations:

- The following defaults for SDLC stations are not satisfactory:
  - Maximum BTU is maximum allowable by interface
  - Tx and Rx Windows are 7 for MOD 8, 127 for MOD 128
- The SNA devices on the link are of mixed node types.
- You want to use the group poll feature.
- You want greater flexibility and control by using the SDLC monitoring commands.

If you do not explicitly add SDLC stations, the router assumes the following:

- The attached stations are of type PU2 if the router's link role is primary or secondary.
- The attached stations are of type T2.1 if the router's link role is negotiable.

**Syntax:** add station

Example: **add station**

```
Enter station address (in hex) [C2]?
Enter station name [SDLC_C2]?
Include station in group poll list([Yes] or No):
Enter max packet size [2048]?
Enter receive window [7]?
Enter transmit window [7]?
Enter PU2 or T2.1 node type [PU2]?
```

| | |
|---|---|
| *Enter station address* | The station's SDLC address in the range 01 - FE. This station address must correspond to the station address added using the **add sdlc** command in the DLSw configuration environment. |
| *Enter station name* | The name of the SDLC station (maximum characters is 8). |
| *Include station in group poll list* | Determines whether or not to include the station in the group poll list for this link. The SDLC software supports the IBM 3174 group poll (GP3174) feature when the router is operating in a local secondary SDLC link role and the adjacent SDLC primary device, typically and IBM mainframe, is suitably configured. You must supply a valid, nonzero group poll address on the link using the **set link group-poll** command in order for station inclusion to have an effect. |
| *Enter max packet size* | The maximum packet size that the router can send to or receive from the link station. This value cannot be greater than that specified for the link with the **set link frame-size** command. |
| *Enter receive window* | The maximum number of packets that the router can receive without sending a response. |
| *Enter transmit window* | The maximum number of packets that the router can transmit without receiving a response. |
| *Enter PU2 or T2.1 node type* | The node type, either PU2 (PU2.0) or T2.1 (PU2.1). |

## Clear M

Clears counters for the end station.

**Syntax:** clear

link

station

**link *name* or *address***

Clears the counters for this SDLC interface. You can display these counters using the **list link counters** command.

Example: **clear link**

**station *name* or *address* or *all***

Clears counters for either a specific end station or all end stations. You can display these counters using the **list station counters** command.

Example: **clear station c1**

### Delete C M

Removes the specified end station from the SDLC configuration.

When used in the console environment, this command terminates any SDLC session in progress.

**Syntax:** delete station *name* or *address*

Example: **delete station C1**

### Disable C M

Prevents connections from being created with an SDLC link or station.

**Syntax:** disable

link

station

**link**

Prevents the establishment of SDLC sessions on any SDLC link stations on the interface.

When used in the console environment, **disable link** also terminates all existing connections on the link.

Example: **disable link**

**station *name* or *address***

Prevents establishment of an SDLC session to the specified station.

When used in the console environment, **disable station** also terminates any existing SDLC session.

Example: **disable station c1**

## Enable C M

Enables the SDLC link entity used by the SDLC station(s). Both links and stations are enabled upon creation. You need to enable them if you previously disabled them.

**Syntax:** enable
link
station

**link**

Allows subsystems in the router (e.g. DLSw) to use SDLCs facilities.

Example: **enable link**

**station *name* or *address***

Allows connections to the specified end station.

Example: **enable station C1**

## List C M

In the SDLC configuration process, **list** displays configuration information on one or all SDLC link stations. In the console process, **list** displays statistics specific to the datalink layer and the interface.

C In the configuration environment, the **list** command does the following:

**Syntax:** list
link
station

**link**

Displays information for the SDLC interface.

Example: **list link**

```
Link configuration for: LINK_2   (ENABLED)

Default role: SECONDARY     Type:          POINT-TO-POINT
Duplex:           FULL          Modulo:          8
Idle State:       FLAG          Encoding:        NRZ
Clocking:         EXTERNAL      Frame Size:      2048
Speed:            0             Group Poll:      F3
Cable:             RS232 DTE:

Timers:      XID/TEST response:       0.5 sec
             SNRM response:           2.0 sec
             Poll response:           0.5 sec
             Inter-poll delay:        0.2 sec
             RTS hold delay:          0.0 sec
             Inter-frame delay:       DISABLED
             Inactivity timeout:      30.0 sec

Counters:    XID/TEST retry:          4
             SNRM retry:              6
             Poll retry:              10
```

| | |
|---|---|
| *Link configuration* | The name and status of SDLC link stations in the router's configuration. |
| *Role* | The role for link stations that you set using **set link role**, primary, secondary, or negotiable. |
| *Type* | The type of link, MULTIPOINT or POINT-TO-POINT. |
| *Duplex* | Duplex configuration, half or full. |
| *Modulo* | The sequence number range to use on the link: MOD 8 (0-7) or MOD 128 (0 - 127). |

| | |
|---|---|
| *Idle State* | The bit pattern (flag or mark) transmitted on the line when the interface is not transmitting data. |
| *Encoding* | The SDLC transmission encoding scheme, NRZ or NRZ1). |
| *Clocking* | Interface clocking, either external, internal, or mixed. |
| *Frame Size* | The maximum frame size that can be sent over the interface. |
| *Speed* | For internal clocking, specifies the speed of the transmit and receive clock lines. For mixed clocking, the speed applies to the transmit clock line only. |
| *Group Poll* | Shows the group poll address configured for this link. |
| *Cable* | Shows the cable type connected to this interface (not used on all platforms). |
| *Timers* | All the timers listed below have a 100 ms resolution. |
| *XID/TEST response* | The time the router waits for an XID or TEST response message before retransmitting the XID or TEST frame. A 0 indicates that the router continues to retry indefinitely. |
| *SNRM response* | The maximum time the router waits for an UA response message before the station retransmits SNRM(E). |
| *Poll response* | The maximum time to wait for a response from any polled station before retrying. |
| *Inter-poll delay* | The amount of time the router (configured with a primary role) waits after receiving a response, before polling the next station. |
| *RTS hold delay* | The amount of time that the primary router waits before dropping RTS low after the transmission of a frame. This parameter is specific to half-duplex operation. |

| | |
|---|---|
| *Inter-frame delay* | The minimum amount of time (in 5.12 microsecond time units) that the primary router waits between transmitting frames. |
| *Inactivity timeout* | For idle NRM/E local secondary stations, sets the time after which the interface transitions the station to its recovery state. A value of 0 allows the station to remain idle indefinetly. |

*Counters:*

| | |
|---|---|
| *XID/TEST retry* | The maximum number of times the router sends an XID or TEST frame without receiving a response before timing out. A 0 (zero) indicates that the router retries indefinitely. |
| *SNRM retry* | The maximum number of times the router will send an SNRM(E) frame without receiving a response before timing out. A value of 0 indicates that the router will continue to retry. |
| *Poll retry* | The maximum number of times the router polls the station without receiving a response before timing out. A value of 0 indicates that the router continues to retry indefinitely. |

**station *all* or *address* or *link station name***

Displays information for the specified SDLC link station on the interface, or for all link stations.

Example: **list station C1**

```
Address   Name    Status   Max BTU  Rx Window  Tx Window   Role
-------   ----    ------   -------  ---------  ---------   ----
  C1    SDLC_C1  Enabled    2048       7          7        T2.1
```

```
Example:  list station all

Address   Name    Status   Max BTU  Rx Window  Tx Window   Role
-------   ----    ------   -------  ---------  ---------   ----
C1(00)   SDLC_C1 Enabled   2048       7           7       T2.1
C2(00)   SDLC_C2 Disabled  2048       7           7       PU2
```

*Address* The address of the SDLC link station. The address in parentheses is the group address of the link. A (00) means there is no group address defined. The group address is shown only if a valid group poll is enabled on the station.

*Name* The name of the SDLC link station.

*Status* The status of the SDLC link station, enabled or disabled.

*Max BTU* The frame size limit of the station. It must not be larger than the maximum Basic Transmission Unit (BTU) packet size configured with the **set link frame-size** command.

*Rx Window* The size of the receive window.

*Tx Window* The size of the transmit window.

*Node Type* The SNA peripheral node type, either PU2 or T2.1.

M In the console environment, the **list** command does the following:

**Syntax:** list

link configuration

link counters

station

#### link configuration

Displays information for the SDLC interface. Once the link is active, entering **list link** at the console prompt displays only parameters that are relevant to the specific link role. It also shows the resolved value of a negotiable link role. See **list link** in the configuration environment for a description of the information displayed.

#### link counters

Displays information for the SDLC counters since the last router restart or the last clear counters. Note that this is the default **list link** action.

```
Example:  list link counters

         I-Frames   I-Bytes    Re-Xmit    UI-Frames      UI-Bytes
         _________  _________  _________  __________  ____________
Send             0          0          0           0             0
Recv             0          0                      0             0

            RR         RNR         REJ
         __________  ________   _________
Send             0          0           0
Recv             0          0           0
```

| | |
|---|---|
| *I-Frames* | Information frames received and sent. |
| *I-Bytes* | Information bytes received and sent. |
| *Re-Xmit* | Retransmitted frames. |
| *UI-Frames* | Unnumbered Information frames sent and received. |
| *UI-Bytes* | Unnumbered Information bytes sent and received. |
| *RR* | RRs (Receive Ready) sent and received. |
| *RNR* | RNRs (Receive Not Ready) sent and received. |
| *REJ* | Rejects sent and received. |
| *UP* | Unnumbered Polls (group polls received. |

**station *all* or *address* or *link station name***

Displays status for the specified SDLC link station or all stations. The software displays an * (asterisk) next to stations that you added in the DLSw configuration process using the **add sdlc** command.

Example: **list station all**

```
Address  Name    Status     Max BTU  Rx Window Tx Window Node Type
-------------------------------------------------------------------------
C1(00)  SDLC_C1   Idle        2048      7          7        PU2
C2(00) SDLC_C2   Disabled   2048        7          7        PU2


list station C1

Address  Name    Status     Max BTU  Rx Window Tx Window Node Type
  *C1   SDLC_C1  Disabled   2048        7         7         PU2
```

*Address* The address of the SDLC link station. The address in parentheses is the group address of the station. A (00) means there is no group address defined.

*Name* The name of the SDLC link station.

*Status* The status of the SDLC link station:

**Enabled** – Enabled, but not allocated.

**Idle –** Allocated, but not in use.

**Connected** – Connected.

**Disconnected** – Disconnected.

**Connecting** – Connection establishment in progress.

**Discnectng** – Disconnection in progress.

**Recovering** – Attempting to recover from a temporary data link error.

*Max BTU* The frame size limit of the remote station. This frame size must not be larger than the maximum basic transmission unit (BTU) packet size configured with the **set link frame-size** command.

*Rx Window* — The size of the receive window.

*Tx Window* — The size of the transmit window.

*Node Type* — The SNA peripheral node type, either PU2 (PU 2.0) or T2.1 (PU 2.1).

**station *name* or *address* counters**

Displays frame transmit and receive counts for the specified link station.

Example: **list station c1 counters**

```
Counters for: SDLC_C1 , address C1 (ENABLED)

      I-Frames     I-Bytes   Re-Xmit        UI-Frames UI-Bytes  XID-Frames
      --------     -------   --------       --------  --------  ----------
Send       569     88870            0              0         0           0
Recv       345      4804            0              0         0           0

            RR         RNR       REJ            TEST     SNRM       DISC
      --------     -------  --------         --------  -------   --------
Send      4779           0         0               1         1          0
Recv      4443           0         0               1         0          0

            UA          DM        FRMR
      --------      -------   --------
Send         0           0           0
Recv         0           0           0
```

*I-Frames* — Information frames received and sent.

*I-Bytes* — Information bytes received and sent.

*Re-Xmit* — Frames retransmitted.

*UI-Frames* — Unnumbered Information frames received and transmitted.

*UI-Bytes* — Unnumbered Information bytes received and transmitted.

| | |
|---|---|
| *XID-Frames* | Exchange Identification frames received and transmitted. |
| *RR* | Receive Ready frames received and transmitted. |
| *RNR* | Receive Not Ready frames received and transmitted. |
| *REJ* | Rejects received and transmitted. |
| *TEST* | Test frames received and transmitted. |
| *SNRM* | Set Normal Response Mode frames received and transmitted. |
| *DISC* | Disconnect frames received and transmitted. |
| *UA* | Unnumbered Acknowledgment frames received and transmitted. |
| *DM* | Disconnected Mode frames received and transmitted. |
| *FRMR* | Frame Reject frames received and transmitted. |

### Set C M

In the SDLC configuration process, **set** configures specific information for one or all SDLC link stations.

In the SDLC monitoring process, **set** dynamically configures information for one or all SDLC link stations without affecting the router's configuration memory. You can issue **set** only on disabled links or stations. Not all parameters are available in the monitoring process. Enter **?** after you type a command to display the available parameters.

All time values are in seconds, with a 100ms resolution.

**Syntax:** set

link cable
link clocking
link duplex
link encoding
link frame-size
link group-poll
link idle
link inactivity
link inter-frame delay
link modulo
link name
link poll
link role
link rts-hold
link snrm
link speed
link type
link xid/test
station

**link cable *type***

Sets the type of cable connected to this interface (not used on all platforms). The options are:

- RS-232
- V35
- V36
- X21

Example: **set link cable rs-232**

**link clocking *internal* or *external* or *mixed***

Configures the SDLC link's clocking. To connect to a modem or DSU, configure clocking as external. If the modem provides the receive clock lines and expects the transmit clock line, use mixed. For internal and mixed clocking, you must enter **set link speed** to configure a clock speed in the range 0 to 6250000 bits per second.

**Note:** Internal and mixed clocking is not supported for all platforms.

Example: **set link clocking internal**

**link duplex *full* or *half***

Configures the SDLC link for full-duplex or half-duplex modem control.

Example: **set link duplex full**

**link encoding *nrz* or *nrzi***

Configures the SDLC transmission encoding scheme as NRZ (Non-Return to Zero) or NRZI (Non-Return to Zero Inverted). NRZ is the default.

Example: **set link encoding nrz**

**link frame-size**

Configures the maximum size of the frames that can be transmitted and received on the data link. The valid entries are 576 to 18000. The default is 2048.

You must set the link frame size greater than the maximum packet size configured with **set station max packet**. Otherwise, the router automatically resets the max packet size to the link frame size, and issues the following ELS message:

```
SDLC.054: nt 3 SDLC/0 Stn C4-MaxBTU too large for Link adjusted (4096->2048)
```

Example: **set link frame-size**

```
Frame size in bytes (576 - 18000) [2048]?
```

**link group-poll *address***

Sets a group poll (unnumbered poll) address for the link. The address must be non-zero.

The SDLC software supports the IBM 3174 group poll (GP3174) feature when the router is configured for a local secondary SDLC link role and the adjacent SDLC primary device, typically and IBM mainframe, is suitably configured. The router ignores this value when it is in local primary mode.

The **add station** or **set station group-inclusion** commands include a station in the group poll list. The **list station** monitoring command displays a parenthesized group address, which confirms the inclusion.

Example: **set link group-poll**

```
Enter group poll address (in hex) [00]? F3
Group poll support enabled
```

**link idle flag *or mark***

Configures the transmit idle state for SDLC framing. The default is the *flag* option, which provides continuous flags (7E hex) between frames. Mark puts the line in a marking state (OFF, 1) between frames.

Example: **set link idle flag**

**link inactivity *#-of-seconds***

Applies to local secondary stations that have been activated (NRM/E) by the SDLC protocol. If the station does not receive any frames for the number of seconds specified, the station enters into a recovery state. The range is 0 to 7200 seconds. The default is 30 seconds. A 0 (zero) allows the station to remain idle indefinitely.

Example: **set link inactivity**

```
Enter secondary link station inactivity timeout [30.0]?
```

**link inter-frame delay**

Inserts a delay between transmitted packets. This command ensures a minimum delay between frames so that the SDLC interface is compatible with slower serial devices at the other end. This value is passed in 5.12 microsecond units.

Example: **set link inter-frame delay**

```
Transmit Delay Counter [0]?
```

**link modulo *8* or *128***

Specifies the sequence number range to use on the link: MOD 8 (0-7) or MOD 128 (0 - 127). The default is 8.

**Note:** When you change this value, the transmit and receive window sizes become invalid.

Use **set station** to change the receive-window and transmit-window sizes. Valid window sizes for mod 8 are 0 to 7; valid window sizes for mod 128 are 8 to 127.

At connection startup, an SNRME (rather than an SNRM) and extended SDLC frame headers are used.

Example: **set link modulo 8**

**link name *name***

Establishes a name for the link that you are configuring. This parameter is for informational purposes only.

Example: **set link name**

```
Enter link name: [LINK_0]?
```

**link poll delay**

Configures the time delay between each poll that is sent over the interface.

Example: **set link poll delay**

```
Enter delay between polls [0.2]?
```

**link poll retry**

Configures the number of times the interface retries to poll a secondary SDLC link station before it decides the link station is down and closes the connection.

Example: **set link poll retry**

```
Enter poll retry count (0=forever) [10]?
```

**link poll timeout**

Sets the amount of time the router waits for a poll response before timing out.

Example: **set link poll timeout**

```
Enter poll timeout [0.5]?
```

**link role *primary* or *secondary* or *negotiable***

Configures the interface as an SDLC primary, secondary, or negotiable link station. The default is primary.

Configuring the role values so that they are not conducive to each other and to the actual SNA devices in use can prevent successful link activation. Configure the link role as follows:

- Specify primary if connecting adjacent secondary SDLC devices such as an IBM 3174 cluster controller.
- Specify secondary if the SDLC link is connected to an adjacent SDLC primary device, such as a front end processor.
- Specify negotiable if a T2.1 (APPN) device is attached and you anticipate XID3 negotiable role exchange.

Connecting multiple T2.1 devices on a multidrop link by definition denotes that true link role negotiation is not being performed, and you should use a predefined link role on both the router and the T2.1 device(s).

It is not required that the respective T2.1 devices perform true end-to-end role negotiation when you configure the router's link as negotiable; the router senses the actual role, whether predetermined or not, and adjusts accordingly. Conversely, if you anticipate end-to-end T2.1 role negotiation and do not configure the router's link role as negotiable, the value you configure influences the role negotiation.

Example: **set link role primary**

#### link rts-hold

The time to hold RTS high after transmitting a frame. This setting is for half-duplex mode. This setting has no effect in full-duplex mode.

Example: **set link rts-hold**

```
Enter RTS hold duration after transmit complete [0.0]?
```

#### link snrm timeout

Sets the time to wait for an Unnumbered Acknowledgements (UA) response before retransmitting an SNRM(E). Applies only to primary stations.

Example: **set link snrm timeout**

```
Enter SNRM response timeout [2.0]?
```

#### link snrm retry

Configures the number of times to retransmit an SNRM(E) without receiving a response before giving up. Applies only to primary stations.

Example: **set link snrm retry**

```
Enter SNRM retry count (0=forever)[6]?
```

**link speed**

For internal clocking, this command specifies the speed of the transmit and receive clock lines. For mixed clocking, the speed applies to the transmit clock line only. The range is 0 to 6250000 for CNX, 0 to 8000000 for DNX, and 0 to 10000000 for RBX.

Example: **set link speed**

```
Internal Clock Speed [0]?
```

**link transmit-delay**

Allows the insertion of a delay between transmitted packets. This command ensures a minimum delay between frames so that it is compatible with older, slower serial devices at the other end. This value is passed in 5.12 microsecond units.

Example: **set link transmit-delay 6**

**link type** ***multipoint*** **or** ***point-to-point***

Configures the SDLC link to either a multipoint link or a point-to-point link.

Example: **set link type multipoint**

**link xid/test timeout**

Sets the maximum amount of time to wait for an XID or TEST frame response before retransmitting the XID or TEST frame. Apllies only to primary stations.

Example: **set link xid timeout**

```
Enter XID and TEST frame response timeout [2.0]?
```

**link xid/test retry**

Configures the maximum number of times the SDLC interface resends an XID or TEST frame before giving up. A 0 (zero) causes the router to retry indefinitely. Applies only to primary stations.

Example: **set link xid retry 5**

```
Enter XID and TEST retry count (0=forever) [4]?
```

**station *address* or *name* address**

Changes the station’s SDLC address in the range 01 to FE.

Example: **set station C1 address**

```
Enter station address (in hex) [C1]? ce
```

**station *address* or *link station name* group-inclusion *no* or *yes***

For SDLC secondary stations, sets whether or not to include this station in the group poll list for this link. The SDLC software supports IBM 3174 group poll function. You must add a non-zero group poll address using **set link group-poll** for this to have an affect.

Example: **set station c1 group-inclusion yes**

**station *address* or *name* max-packet**

The maximum size of the packet that the station can receive. The default is 2048 bytes.

Do not set the maximum packet size larger than the link frame size configured with **set link frame-size**. If you do, the router automatically resets the max packet size to the link frame size, and issues the following ELS message:

```
SDLC.054: nt 3 SDLC/0 Stn C4-MaxBTU too large for Link adjusted (4096->2048)
```

Example: **set station c1 max-packet**

```
Enter max packet size [2048]?
```

**station *address* or *name* name**

The name of the SDLC station.

Example: **set station c1 name**

```
Enter station name [SDLC_C1]?
```

**station *address* or *name* receive-window**

The maximum number of frames the router can receive before sending a response. The range is 1 to 7. The default is 7.

Example: **set station c1 receive-window**

```
Enter receive window [7]?
```

**station *address* or *name* node-type *pu2* or *t2.1***

The node type of the station, either PU2 (PU 2.0) or T2.1 (PU 2.1).

Example: **set station c1 node-type pu2**

**station *address* or *name* transmit-window**

The maximum number of frames the router can transmit before receiving a response frame. The range is 1 to 7. The default is 7.

Example: **set station c1 transmit-window**

```
Enter transmit window [7]?
```

## Test M

Transmits a specified number of TEST frames to the specified link station and waits for a response. Use this command to test the integrity of the connection. Press any key to cancel the test.

**Note:** This command does not apply to adjacent primary link stations (that is, when the local router link is defined as secondary).

**Syntax:** test station *name or address #frames frame-size*

Example: **test station c1**

```
Number of frames to send [1]? 5
Frame length [265]?
Starting echo test -- press any key to abort
5 frames sent, 5 frames received, 0 compare errors, 0 timeouts
```

| | |
|---|---|
| *Number of frames* | Total number of frames to send. |
| *Frame length* | Length of the frame sent. This frame cannot be any larger than the maximum frame length of the specified station. |

## Exit C M

Use the **exit** command to return to the previous prompt level.

**Syntax:** exit

Example: **exit**

# 4

# Using Boundary Access Node

This chapter describes Digital's implementation of Boundary Access Node (BAN). Developed in close collaboration with IBM, BAN provides a reliable, low-cost way for attached PU Type 2.0 and 2.1 end stations to communicate with the SNA environment across wide area links.

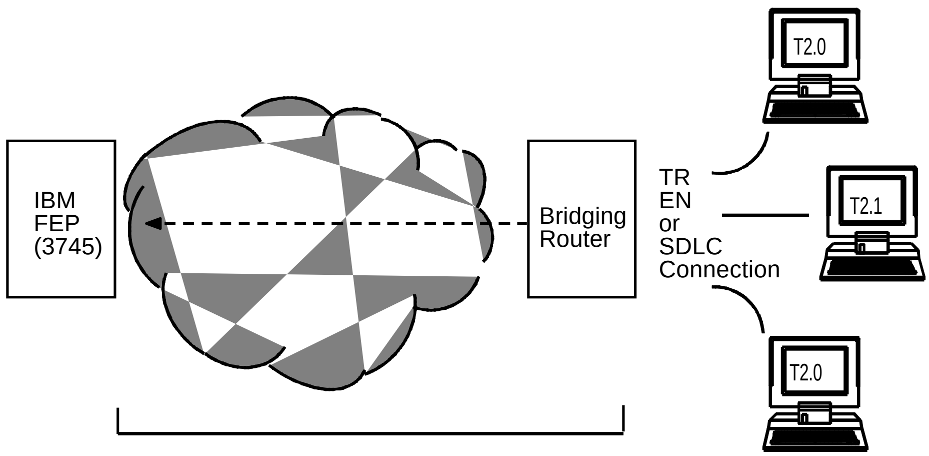
## 4.1 About Boundary Access Node

Boundary Access Node (BAN) is an enhancement of the Frame Relay (FR), DLSw, and Adaptive Source Route Bridging (ASRT) capabilities of the router software.

BAN is designed to meet the business goals of customers who do not yet need a full DLSw implementation. It provides a low-cost method for connecting to IBM environments, enabling SNA end stations to bridge Ethernet and Token Ring traffic directly to the FEP without frame conversion by another DLSw router. This saves significantly on capital equipment costs since it removes the need for another router, a Token Ring, and a TIC-3745 interface card attached to the remote SNA device.

BAN accomplishes this by enabling IBM type 2.0 and 2.1 end nodes connected to a Digital router to make a direct connection via Frame Relay with the front end processor (FEP) attached to an IBM mainframe as shown in Figure 4–1.

**Figure 4–1 Direct Connection of End Nodes to IBM FEP Using BAN**




Though traffic passes through them, the bridging router and FR network are transparent to end nodes when using BAN.

### 4.1.1 How BAN Works

BAN works by filtering the frames sent by Type 2.0 or 2.1 end stations. The router modifies each BAN frame to comply with Bridged 802.5 (Token Ring) Frame format. The router subsequently examines each frame and allows only those *with the BAN DLCI MAC address* to pass over a DLCI (Data Link Connection Identifier) to the FEP.

**Note:** To support BAN, an IBM FEP must be running NCP software 7.3 or greater, or NCP software 7.1 or 7.2 with a software patch. If you have questions about whether your FEP can support BAN, contact your IBM representative.

With BAN, one DLCI is ordinarily all that is needed. However, BAN may use many DLCI connections between the router and the IBM environment. In some cases, you may want to set up more than one DLCI to handle BAN traffic. See the section *Setting Up Multiple DLCIs* in this chapter for more information.

There are two ways to use BAN: straight bridging (using the router's bridging capability) and DLSw terminated. In the majority of cases, you should choose the bridging option. However, you may consider choosing the terminated option if you want to reduce session timeouts on the DLCI. The sections that follow explain how to set up each option.

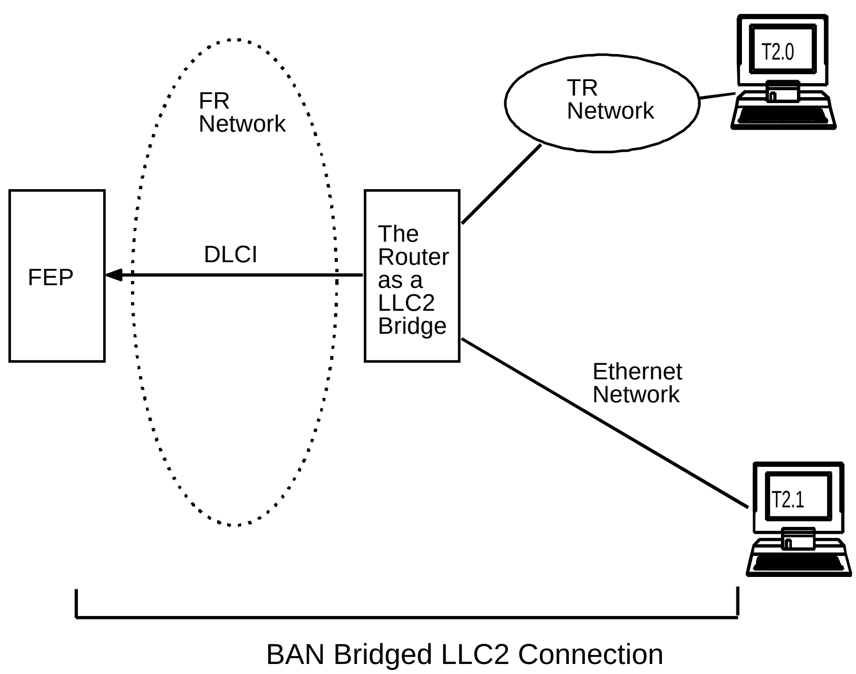
### 4.1.2 Bridged and DLSw-Terminated BAN

Digital enables you to implement BAN in two ways. With the straight bridging method, you configure BAN to bridge LLC2 frames from Type 2.0 or Type 2.1 end stations straight into the NCP. With the DLSw Terminated method, BAN terminates the LLC2 connection at the DLSw router.

Within this discussion, we refer to these two methods as BAN Type 1 and BAN Type 2, respectively.

Figure 4–2 shows a BAN Type 1 (bridged) connection. In this illustration, notice that the router does not terminate the LLC2 traffic received from attached end nodes. Instead, the router converts the BAN DLCI-addressed frames it receives to bridged Token Ring format (RFC 1490) frames, and bridges them directly to the NCP.

**Figure 4–2 BAN Type 1: The Router as an LLC-2 Bridge**

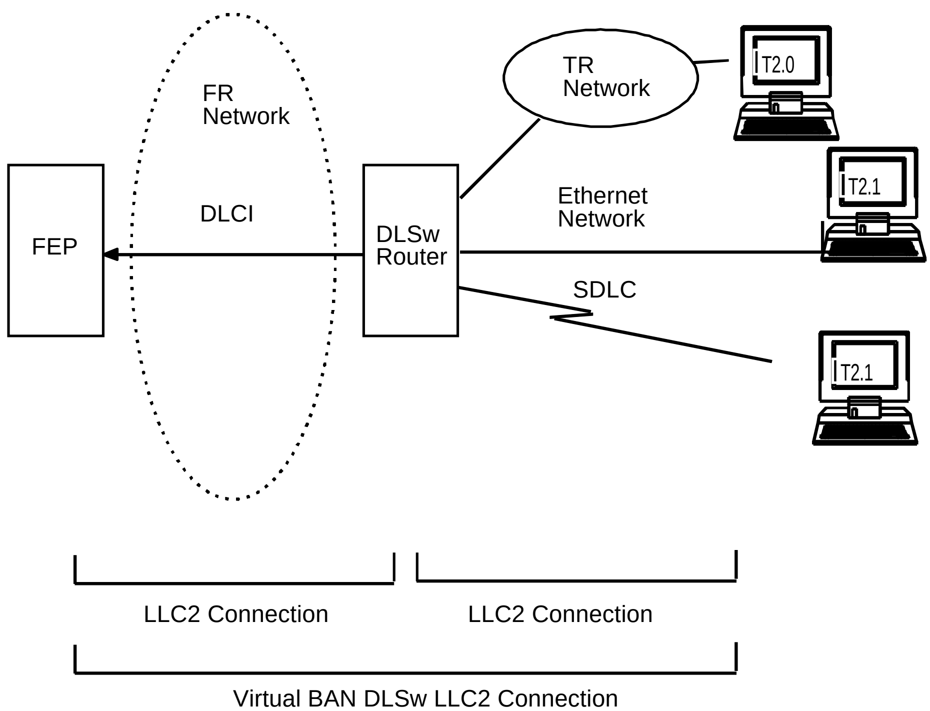



In this case, the router acts as a bridge between the FEP and end stations. DLSw does not terminate the LLC2 session at the router, as in BAN Type 2. End station frames can be Token Ring, SDLC, or Ethernet, provided the bridge is configured to support that type of frame.

Figure 4–3 shows a BAN Type 2 (Virtual BAN DLSw) connection. In this illustration, notice that the DLSw router does not function as a bridge. The router terminates the LLC2 traffic received from attached end nodes. At the same time, the router establishes a new LLC2 connection to the NCP over the Frame Relay network. Thus, though two LLC2 connections exist within the transaction, the break between them is transparent both to the NCP and the end nodes. The result is a virtual LLC2 connection between NCP and end nodes.

**Note:** Network datalink support is dependent on the specific router platform. Read the hardware documentation and/or Software Product Description for details.

**Figure 4–3 BAN Type 2: Local DLSw Conversion**




### 4.1.3 Which Method Should You Use?

Straight bridging of frames (BAN Type 1) is generally preferable. This method provides fast delivery of data with minimal network overhead. However, there are exceptions to this rule. If usage on a DLCI is too high, session timeouts may occur in a bridged configuration.

Conversely, session timeouts rarely occur in a DLSw-terminated configuration (BAN Type 2) because this type of configuration terminates and then recreates LLC2 sessions at the local (DLSw) router. For this reason, you may want to use DLSw-terminated BAN in situations where reducing the possibility of session timeouts is a concern. When running in DLSw-terminated mode, the router terminates *all* traffic on the DLCI. This mode also limits the number of remote end stations the BAN configuration can support.

## 4.2 Using BAN

**Note:** BAN router support requires both a Frame Relay WAN port and internal support for source route bridging. Read the Software Product Description to determine BAN capability for your specific router.

To configure BAN, follow these steps:

1. Configure the router for Frame Relay (FR).
2. Configure the router for Adaptive Source Route Bridging (ASRT).
3. Configure the router for BAN.
4. Open Service Access Points (SAPs) on the FR and LAN interfaces.

These steps are documented in the example that follows.

This example assumes that you are setting up a single DLCI to carry BAN traffic. Depending on your circumstances and needs, you may want to set up multiple DLCIs for the sake of redundancy, or to increase total bandwidth to the IBM environment. See the section *Setting up Multiple DLCIs* for more information.

#### 4.2.0.1 Configuring Frame Relay

To access the Frame Relay configuration area, use the **network** command at the `Config>` prompt as shown:

**Note:** The `set data-link` command is used to establish Frame Relay on a given network interface. See the *System Software Guide* for more information about the `set data-link` command.

```
Config>network 2
Frame Relay user configuration
FR Config>
```

At the `FR Config>` prompt, add a permanent circuit as shown below. The router prompts you for a circuit number. This is the DLCI number. The router then prompts you for a committed information rate, and for a circuit name.

The circuit name is *extremely important*. It tells the bridge which DLCI to use for BAN frames. In doing so, it provides the linkage between the router (which is acting as a bridge in this case) and the FR protocol.

```
FR Config>add permanent
Circuit number  [16]? 20
Committed Information Rate (CIR) in bps [64000]?
Comitted Burst Size (Bc)in Bits [64000]?
Excess Burst Size (Be) in Bits [0]?
Assign circuit name []? 20-ncp10
```

| | |
|---|---|
| *Circuit Number* | Indicates the circuit number in the range of 16 to 1007. |
| *Committed Information Rate* | Indicates the committed information rate (CIR) in a range of 300 bps to 2048000 bps. The default is 64 Kbps. |
| *Committed Burst (Bc)* | Indicates the maximum amount of committed data that the PVC can transmit, in the range of 300 bps to 2048000 bps. The default is 64 Kbps. |
| *Excess Burst (Be)* | Indicates the maximum allowed amount of uncommitted data for the PVC in the range of 0 bps to 2048000 bps. The default is 0 bps. |
| *Assign Circuit Name* | Indicates the ASCII string that is assigned to describe the circuit. This parameter is optional. It is recommended that you use a name that describes the characteristics of the circuit. The default is *unassigned*. |

You should assign a circuit name that identifies the IBM NCP in some obvious way (as in this example, where the assigned circuit name is `20-ncp10`). You should also use a name that has 8 or fewer characters. Choosing a short name may prevent it from being truncated on some bridge configuration screens.

The DLCI you create by assigning a circuit number and name becomes the PVC that connect Digital's router with the IBM FEP when using BAN. The next step consists of configuring this PVC as a bridge port.

**Note:** If you want to set up multiple BAN DLCIs connected to the same or different FEPs, you have to configure Frame Relay separately for each DLCI. For more information on this, see the section *Setting up Multiple DLCIs*.

#### 4.2.0.2 Configuring the Router for Adaptive Source Route Bridging

Next, configure the PVC as a bridge port. To do this, enter **protocol asrt** at the `Config>` prompt.

```
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT Config>
```

**Note:** You must enable bridging before adding a port as described below.

At the `ASRT Config>` prompt, add a port as shown below. The router prompts you for an interface number. The number you assign is the FR interface number on the bridge. The router then prompts you for a port number and for a circuit name. The circuit name you assign must be the same as that used when configuring the router for bridging over FR.

```
ASRT config>add port
Interface Number [0]? 2
Port Number [5]?
Assign circuit name []? 20-ncp10
```

The next step consists of enabling source routing and defining source routing segment numbers for the FR port. The bridge virtual segment number prompt is optional, and may or may not appear.

```
ASRT config>enable source-routing
Port Number [3]? 5
Segment Number for the port in hex(1 - FFF) [1]? 456
Bridge Virtual Segment Number in hex(1-9, A-F)) [1]? 789
```

Then, disable transparent bridging on the bridge port as shown:

```
ASRT config>disable transparent bridging
Port Number [3]? 5
```

#### 4.2.0.3 Configuring the Router for BAN

You configure BAN from the `ASRT Config>` prompt. The addition of a BAN port is not verified until you restart the router.

```
Config> protocol asrt
ASRT config>ban
BAN (Boundary Access Node) configuration
```

At the `BAN Config>` prompt, add the port number (5) on which you want to enable BAN. The router prompts you to enter a BAN DLCI MAC address, and the Boundary Node Identifier address as shown:

```
BAN config>add 5
Enter the BAN DLCI MAC Address []? 400000000001
Enter the Boundary Node Identifier MAC Address [4FFF00000000]?
```

In this example, 400000000001 is the MAC address of the DLCI: this is the address to which attached end stations will send data. The other address, 4FFF00000000, is the default Boundary Node Identifier Address. To accept it, press RET.

**Note:** You should always choose the default Boundary Node Identifier address unless the Boundary Node Identifier address of the receiving FEP has been changed. This is because the Boundary Node Identifier address *must match* the corresponding value in the NCP definition. This value is specified by the LOCADD keyword of the LINE statement that defines the physical Frame Relay connection.

#### 4.2.0.4 Specifying the Type of BAN Connection You Need

The next prompt asks you to specify which type of BAN connection you want to add, bridged (described earlier as BAN Type 1) or DLSw-terminated (Type 2). Type 1, straight bridging, is the default. You should accept the default unless you want inbound traffic to be terminated at the router.

After you enter **b** (bridged) or **t** (terminated), the router informs you that the BAN port has been added. The default choice is **b** .

```
Do you want the traffic bridged (b) or DLSw terminated (t) (b/t) [b]?

BAN port record added.

Reminder: enable source-routing on the port if you haven't already done so.
```

#### 4.2.0.5 Opening Service Access Points (SAPs)

To use BAN, you must open the Service Access Points (SAPs) associated with the FR interface and the LAN interface. If you fail to open these SAPs, you will not be able to use BAN. Failure to open all SAPs is often the cause of configuration problems.

You open the SAPs from the `DLSw config>` prompt as follows:

**Note:** The interface number that you enter depends upon your router type.

```
DLSw config>open
Interface # [0]?
Enter SAP in hex (range 0-ff) [0]? 4
```

Issuing the **open** command for interface 0 opens the SAP on the LAN interface. You issue the same command to open the SAP on the FR interface. Note that in each case, you enter **4** as the SAP value.

```
DLSw config>open
Interface # [2]?
Enter SAP in hex (range 0-ff) [0]? 4
```

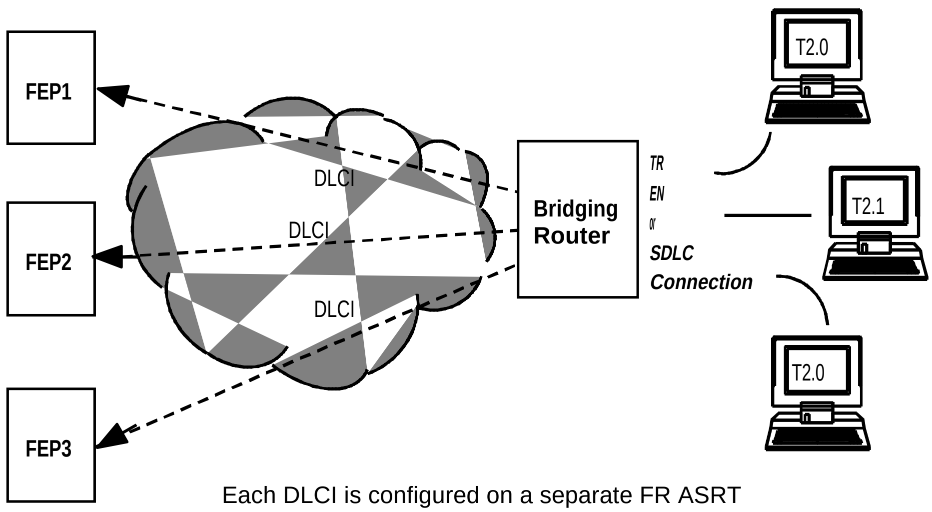
## 4.3 Using Multiple DLCIs for BAN Traffic

While one DLCI is usually sufficient to handle BAN traffic to and from the IBM environment, setting up two or more DLCIs may prove useful in some circumstances.

#### 4.3.0.1 Benefits of Setting Up a Fault-Tolerant BAN Connection

Redundant connections to multiple NCPs protect against a single NCP failure. In addition, sharing BAN traffic among several DLCIs reduces the chance of one NCP becoming overloaded. In a redundant DLCI configuration, PU Type 2.0 and 2.1 end stations can pass BAN traffic to different NCPs, as shown in Figure 4–4.

**Figure 4–4 BAN Configuration with Multiple DLCIs to Different FEPs**




Each DLCI is configured on a separate FR ASRT bridge port with the same DLCI MAC Address

## 4.3.1 Setting Up Multiple DLCIs

Setting up multiple DLCIs is a simple matter, particularly if you elect to do this during the initial BAN configuration.

In setting up multiple connections, keep in mind that each Frame Relay DLCI corresponds with a specific FEP in the IBM environment. To pass BAN frames to that FEP, you must specify the correct circuit number when establishing the Frame Relay connection. Your Frame Relay provider will be able to tell you the circuit number for each of your connections.

To set up DLCI connections to different FEPs (Scenario 1, above) you must:

1. Within the Frame Relay configuration:

   Define another Frame Relay DLCI on a second bridge port.

2. Within the ASRT configuration:

   Add a bridge port for that DLCI.

3. Configure the bridge port for BAN, as shown earlier in this chapter.

## 4.4 Checking the BAN Configuration

When you restart the router, the BAN bridge appears as an FR bridge port with source-routing behavior. You should check the BAN configuration with the **list** command as shown here:

```
BAN config>list

bridge  BAN                Boundary              bridged or
port    DLCI MAC Address   Node Identifier       DLSw terminated
5       40:00:00:00:00:01  4F:FF:00:00:00:00     bridged
```

As this example shows, the **list** command displays each aspect of the BAN configuration, giving the bridge port (5, in this case) the MAC addresses of the router and the NCP, and whether the port is bridged or DLSw terminated.

To check to see that BAN has initialized properly on startup, you can use the router's console environment (at **t 5**) as follows:

```
+p asrt
ASRT>ban
BAN (Boundary Access Node) console
BAN>list

bridge  BAN                Boundary           bridged or
port    DLCI MAC Address   Node Identifier    DLSw terminated  Status
5       40:00:00:00:00:01  4F:FF:00:00:00:00  bridged          Init Fail
```

BAN has three associated status messages:

- `Init Fail` indicates that a configuration problem exists.
- `Up` indicates that the FR DLCI is up and running.
- `Down` indicates that the DLCI is not running.

If you receive a status other than `Up,` you should check the router's ELS messages to diagnose the problem. The following section explains how to enable ELS messages.

## 4.5 Enabling BAN Event Logging System Messages

After initial BAN configuration and restart, it is a good idea to enable ELS messages to see whether the configuration is working as planned. You can enable BAN-specific messages from the `Config>` prompt as shown:

```
Config>event
Event Logging System user configuration
ELS config>display subsystem ban all
```

Entering this command displays all BAN subsystem messages. This will cause ELS to notify you of all BAN-related behavior. After running BAN for a while, you may want to turn off some messages.

You can switch off specific ELS BAN messages using the **nodisplay** command and the specific message number. This example illustrates how to turn off the `ban.9` message.

```
ELS config>nodisplay event ban.9
```

For a list and explanation of all BAN-related messages, see the *Event Logging System Messages Guide*.

## 4.6 BAN Configuration and Console Commands

This section includes all of the Boundary Access Node configuration commands.

### 4.6.1 Accessing the BAN Configuration Environment

Use the router's configuration process to change the configuration of the router. The new configuration takes effect when the router is restarted.

To enter the configuration environment, enter **talk 6**, or just **t 6**, at the + prompt. This brings you to the `Config>` prompt as shown:

```
MOS Operator Control

* talk 6
Gateway user configuration
Config>
```

If the `Config>` prompt does not appear immediately, press RET again.

Enter all BAN configuration commands at the `BAN Config>` prompt. You can access this prompt by entering **ban** at either the `DLSw Config>` or `ASRT config>` prompt as shown:

```
Config>protocol dls
DLSw protocol user configuration
DLSw Config>ban
BAN Config>
```

### 4.6.2 Accessing the BAN Console Environment

To enter the console process, enter **talk 5**, or just **t 5**, at the * prompt. This brings you to the + prompt as shown:

```
MOS Operator Control

* talk 5
+
```

Enter BAN (Boundary Access Node) console commands at the `BAN>` prompt. To access this prompt, enter **ban** at the `DLSW>` or `ASRT>` prompt as shown:

```
+ protocol dls
DLSW>ban
BAN (Boundary Access Node) Console
BAN>
```

### 4.6.3 BAN Commands

Enter BAN configuration commands at the `BAN config>` prompt and console commands at the `BAN>` prompt. Table 4–1 lists the BAN configuration and console commands.

**Table 4–1 BAN Command Summary**

| Command | Task | Function |
|---|---|---|
| **? (Help)** | Configure/ Monitor | Lists available BAN commands or associated parameters. |
| **Add** | Configure | Add a BAN port |
| **Delete** | Configure | Deletes a BAN port. |
| **List** | Configure/ Monitor | Displays the existing BAN configuration, and informs you whether the port has initialized properly. |
| **Exit** | Configure/ Monitor | Exits the BAN configuration process and returns you to the `DLSw config> or ASRT config>` process. |

### ? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a command to list its options.

**Syntax:** ?

```
Example: ?

    ADD
    DELETE
    LIST
    EXIT
```

### Add C

Adds a BAN port.

**Syntax:** add *port #*

```
Example: add 2

Enter the BAN DLCI MAC Address []? 400000000001
Enter the Boundary Node Identifier MAC Address [4FFF00000000] ?

Do you want the traffic bridged (b) or DLSw terminated (t) (b/t) [b]?

BAN port record added.
```

### Delete C

Deletes a previously added BAN port from the configuration.

**Syntax:** delete *port#*

```
Example: delete 2
```

### List C M

Use the **list** command to display information on the existing BAN configuration, or to assess whether the DLCI is functioning properly.

**Syntax:** list

When issued in the BAN configuration module, the **list** command provides general information on the BAN configuration.

Example: **list**

```
bridge  BAN                Boundary              bridged or
port    DLCI MAC Address   Node Identifier       DLSw terminated
5       40:00:00:00:00:01  4F:FF:00:00:00:00     bridged
```

To check to see that BAN has initialized properly on startup, you can use the router's console environment (at **t 5**) as follows:

When issued in the BAN console module, the **list** command provides general information on the BAN configuration. The command also informs you whether each BAN port has initialized properly.

Example: **list**

```
bridge  BAN                Boundary           bridged or
port    DLCI MAC Address   Node Identifier    DLSw terminated  Status
5       40:00:00:00:00:01  4F:FF:00:00:00:00  bridged          Init Fail
```

### Exit C M

Exits the BAN configuration or console process. If you exit from the configuration process, you return to the `DLSw Config>` or `ASRT Config>` prompt. If you exit from the console process, you return to the `DLSW>` or `ASRT>` prompt.

**Syntax:** exit

Example: **exit**

# 5

# Using SDLC Relay

This chapter describes Digital's implementation of Synchronous Data Link Control Relay (SRLY).

## 5.1 About SDLC Relay

Like DLSw (see Chapter 2), SRLY is a method for consolidation of SDLC traffic onto the corporate multiprotocol backbone.

Unlike DLSw, SDLC Relay does not terminate the SDLC data link to reduce the likelihood of session timeouts, and does nothing to help reduce congestion on the WAN link. What SRLY provides is a serviceable method for shipping bit-oriented protocol (SDLC, HDLC, LAPB) frames across WAN links in situations when it is not possible to use data link switching (Digital's DLSw product).

For more information Digital's DLSw product, see Chapter 1, *Using the DLSw Protocol.*

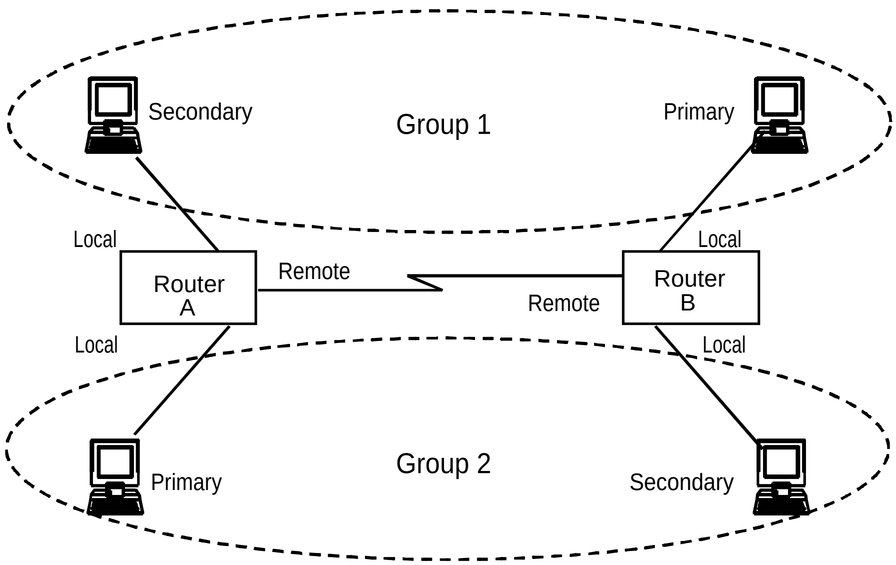
## 5.2 How SDLC Relay Works

Despite its name, the SDLC Relay protocol (SRLY) is designed to handle other protocols besides SDLC. The protocol works by encapsulating SDLC or any bit-oriented protocol in UDP packets, and transmitting them through the IP cloud on a point-to-point connection to another SRLY device.

These connections are established by matching SRLY traffic to specific *ports* and *groups*. During configuration, each group has a unique group number assigned, and exactly two ports: one SDLC *primary* port, and one SDLC *secondary*. Matching SRLY traffic to group numbers and ports ensures that attached end stations can only send packets to the end stations for which they are intended.

Once packets are received, they are stripped of their UDP/IP header and transmitted to their destination address in their original protocol format.

**Figure 5–1 SDLC Primary/Secondary Stations and Local/Remote Ports**




Encapsulation in UDP/IP packets allows for SDLC frames to be handled via IP routing techniques. And since each SDLC frame is encapsulated unchanged, SRLY is transparent to sending and receiving stations. This transparency allows SRLY to support all SNA PU Types.

### 5.2.1 SDLC Primary and Secondary Stations

When configuring SRLY, a router's primary port must be connected to its primary end station. Its secondary port must be connected to its secondary end station. Within the primary-secondary communication process, the primary end station is responsible for initiation, scheduling, and termination of the session. The secondary station does not initiate communication, but responds to commands from its primary partner.

When running balanced protocols such as LAPB or HDLC (or when running SDLC T2.1 negotiable link station traffic), you can assign roles arbitrarily as long as one device is primary, and its connected counterpart is secondary.

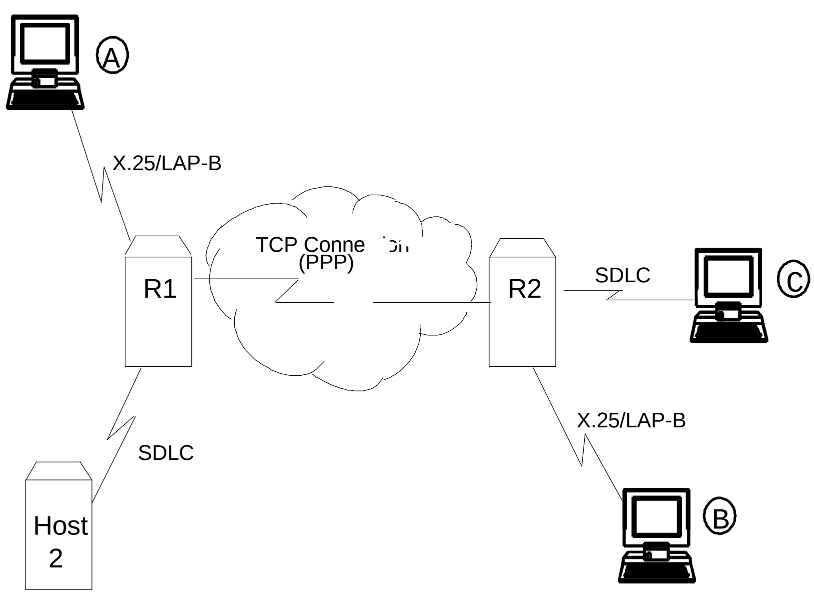
### 5.2.2 When to Use SDLC Relay

Generally, you would use SRLY instead of DLSw when you need to exchange any bit-oriented protocol, such as LAPB, HDLC, or SDLC over a wide area, between SNA or non-SNA devices.

Protocol end-to-end acknowledgements (due to the lack of datalink termination) should be tolerated, and the station traffic must be point-to-point, full duplex modem control.

Since the UDP/IP messages generated by SRLY are recognized by the network as standard IP traffic, any medium or interface that will accomodate IP will also accomodate SRLY. For example, Figure 5–2 shows a PPP link between two routers, but the IP connection could also be Frame Relay (or even LAN-based) as requirements dictate.

**Figure 5–2 SDLC Relay Configurations**



## 5.3 Setting Up SDLC Relay

Configuring SDLC Relay (SRLY) involves performing these steps on each of two routers.

1. Set the data link on the serial line using the **set data-link** command and the appropriate interface number.
2. Assign a group number using the **add group** command. The group number must be the same on each SRLY router. Group number 1 is the default.
3. At the `SDLC Config>` prompt, add a local port with the **add local** command. Be sure you add this port to the group defined in Step 2.
4. This port's data link type must be SDLC Relay (SRLY). Use the **set data- link** command at the `Config>` prompt to set the data link type for the port.
5. At the `SDLC Config>` prompt, add a remote port with the **add remote** command. The IP address of the remote port is that of the cooperating SRLY router.
6. Repeat these steps for the second SRLY router. When prompted for the IP address of the remote port, provide the address of the first router.

## 5.4 Sample SDLC Relay Configuration

Following is a complete SDLC Relay configuration. The example assumes that the router has not been configured for any other protocols or data links.

### 5.4.1 Context Diagram

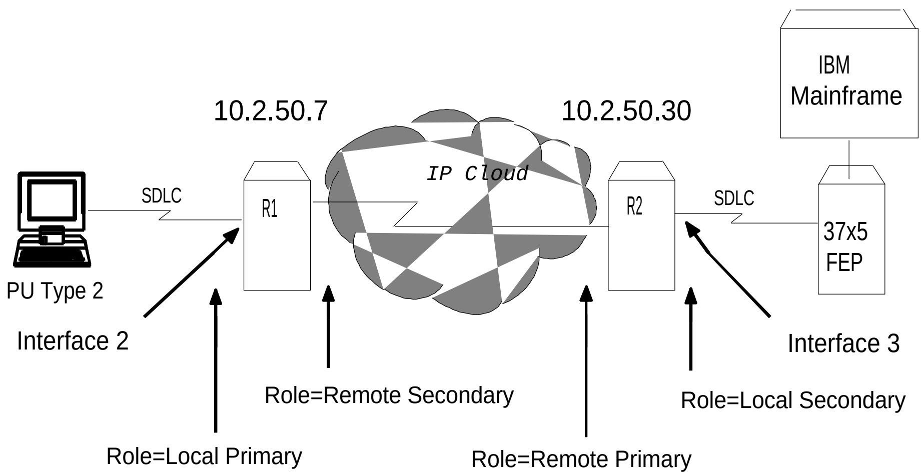
The example is based on the information shown in Figure 5–3. The IP connection between the two routers is over the serial line. The serial line supports NRZ or NRZ1, set in SRLY via the `set encoding` command.

Configuring $R_1$ for SDLC Relay requires all of the information shown. This information includes the following:

- Group numbers for each group of SRLY ports
- Interface numbers for each SRLY port
- The internet addresses for each SRLY router

The example indicates where this information is provided in the course of the configuration procedure.

**Figure 5–3 Context Diagram for SRLY Configuration**




This example explains how to configure two routers for SRLY traffic. Router 1 ($R_1$) is connected to a PU Type 2.0 node. Router 2 ($R_2$) is connected to a front end processor (FEP).

## 5.4.2 Configuring SDLC Relay

On $R_1$, set the data link of interface 2 to an SDLC Relay device. Use the **set datalink** (abbreviated below) command shown here.

```
Config>set data srly 2
```

You can list the devices to confirm that an SDLC Relay device has been added.

```
Config>list dev
Ifc 0 (Token Ring): CSR 6000000, vector 28
Ifc 1 (WAN PPP): CSR  81620, CSR2  80D00, vector 93
Ifc 2 (WAN SDLC Relay): CSR  81640, CSR2  80E00, vector 92
```

### 5.4.2.1 Set Serial Line Parameters

Next, optionally set the line speed and clocking type parameters for the SRLY line. You must also set encoding (NRZ or NRZI), frame size, and idle character. Note that the prompt for the SRLY configuration module is `SRLY # Config>`, where `#` is the number of the SRLY interface.

```
Config>network 2
SDLC relay interface user configuration

SRLY 2 Config>set clock internal
Must also SET SPEED for internal or mixed clocking
SLC Config>set speed
Internal Clock Speed [0]? 56000
```

**Note:** Internal clocking is supported on limited platforms.

The set cable command below is optional, and may not appear with your router.

```
SRLY 2 Config>set cable rs-232 dce
```

After setting the line speed, clocking, and cable type, you can check the configuration with the **list** command as shown:

```
SRLY 2 Config>list
Synchronous serial line interface configuration


Maximum frame size in bytes = 2048
Encoding: NRZ
Idle State: Flag
Clocking: Internal
Cable type: RS-232 DCE
Internal Clock Speed: 56000
Transmit Delay Counter: 0
SRLY 2 Config>exit
```

#### 5.4.2.2 Configuring the SDLC Relay Protocol

Configure the SDLC Relay protocol as shown:

```
Config>protocol sdlc
SDLC Relay protocol user configuration
SDLC Config>
```

As this example shows, the prompt for the SDLC Relay (SRLY) area is `SDLC Config>`. Commands entered at this prompt only affect the SDLC Relay protocol. They have nothing to do with, and do not affect, SDLC data links or devices.

You can exit the SDLC Relay configuration procedure at any time by entering **exit**.

#### 5.4.2.3 Assign a Group Number

The group number provides the association/binding between the router's local and remote ports, as well as the correlation with the corresponding ports on the partner router. The group number is communicated by SRLY between the two routers.

First, assign a group number with the **add group** command. This number is assigned to the *primary* and *secondary* ports on the router you are configuring for SRLY. The group number you designate must be the same for each router.

```
SDLC Config>add group
Group number: [1]?
```

Notice that the **list group** command shows that no ports have yet been configured for group 1.

```
SDLC Config>list group
Group number: [1]? 1

                         SDLC Relay Configuration

 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  Address (hex)
-------------  -----------------  -------  -------------    -----------
No ports configured for group  1
```

**Note:** The SDLC Station Address heading is currently not used. It is reserved for future use.

#### 5.4.2.4 Add a Local Port

Next, add a local port to group 1. The port you add will be the SRLY line defined earlier. The local port is the serial interface over which the native SDLC (or HDLC or LAPB) traffic flows.

```
SDLC config>add local
Group number: [1]?
Interface number: [0]? 2
(P)rimary or (S)econdary: [S]?
```

Notice that the **list all** command shows that a local primary port has been configured for group 1.

```
SDLC config>list all

                         SDLC Relay Configuration

 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  Address (hex)
-------------  -----------------  -------  -------------    -----------
     1 (E)       Local  PRIMRY (E)    2
```

The (E) shown within the Port Status column stands for "Enabled." By default, SRLY ports are enabled; SRLY ports must remain enabled in order to use the feature.

#### 5.4.2.5 Add a Remote Port

Next, add a remote port for group 1. This is the port that leads to the IP cloud. Each group must consist of a pair of ports, one primary, and the other secondary. The remote port added here must be secondary since the local port attached to it is primary.

The IP address provided is that of the router on the other side of the IP cloud, $R_2$.

```
SDLC Config>add remote
Group number: [1]?
IP address of remote router: [0.0.0.0]? 10.2.50.30
(P)rimary or (S)econdary: [S]? S
SDLC Config>list all

                         SDLC Relay Configuration

 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  Address (hex)
-------------  -----------------  -------  -------------    -----------
    1 (E)        Local   PRMRY  (E   2


    1 (E)        Remote SCNDRY (E)                           10.2.50.30
```

### 5.4.3 Configure the Neighbor Router

Up to this point, this example has shown how to configure $R_1$ in Figure 5–3. SRLY requires two routers, one on either side of the IP cloud. You must configure SRLY on each of them.

#### 5.4.3.1 Set Data Link, Add Group, and Add Port

First, set up an SRLY data link for $R_2$. Do this in the same manner as shown earlier for $R_1$.

Next, add a group for $R_2$, assigning the same group number (1, in this case) as that assigned on $R_1$. Add a local port for the assigned group. This is the SRLY line you have already defined. In this case, the port type is *secondary* since a front end processor (FEP) (which, for peripheral "boundary" PU2 traffic, is always primary) is on the line.

```
SDLC config>add local
Group number: [1]?
Interface number: [0]?
(P)rimary or (S)econdary: [S]?
SDLC config>list all

                SDLC Relay Configuration

 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  address (hex)
-------------  -----------------  -------  -------------    -----------

    1 (E)      Local   SCNDRY (E)    0
```

#### 5.4.3.2 Add a Remote Port

Finally, add a remote port for group 1. This is the port that leads to the IP cloud. Since the FEP is primary, this port is secondary. As mentioned earlier, each group must consist of a primary and secondary station.

Since $R_2$ is being configured, the IP address of the remote router belongs to $R_1$. See Figure 5–3 for the addresses of $R_1$ and $R_2$, and their roles in the overall SRLY configuration.

```
SDLC Config>add remote
Group number: [1]?
IP address of remote router: [0.0.0.0]? 10.2.50.7
(P)rimary or (S)econdary: [S]? p

SDLC Config>list all

                SDLC Relay Configuration

 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  address (hex)
-------------  -----------------  -------  -------------    -----------
    1 (E)      Remote PRMRY   (E)                           10.2.50.7
    1 (E)      Local  SCNDRY  (E)    0
```

# 6

# Configuring and Monitoring SDLC Relay

This chapter describes the Synchronous Data Link Control (SDLC) Relay configuration and console commands.

For more information about the SDLC Relay protocol, refer to Chapter 5, Using SDLC Relay.

## 6.1 About SDLC Relay Configuration and Console Commands

Enter SDLC Relay configuration commands at the `SDLC Config>` prompt. Changes made to the router's configuration do not take effect immediately. They affect the operating router only after it is restarted.

Conversely, you enter SDLC Relay console commands at the `SDLC>` prompt. These commands take effect immediately, but do not become part of router's configuration memory. Thus, while console commands allow you to make real-time changes to the router's configuration, these changes are temporary.
Any permanent changes you wish to make (by storing them in FLASH) should be made with SDLC Relay configuration commands.

Monitoring consists of these actions:

- Monitoring the protocols and network interfaces currently in use by the router
- Displaying Event Logging System (ELS) messages relating to router activities and performance
- Making real-time changes to the SDLC Relay configuration without permanently affecting the router's nonvolatile configuration memory

## 6.2 Accessing the SDLC Relay Configuration Environment

Use the SDLC Relay configuration process to change the configuration of the router. The new configuration takes effect when you restart the router.

To enter the configuration process, enter **talk 6**, or just **t 6**, at the * prompt. This brings you to the `Config>` prompt as shown:

```
MOS Operator Control

* talk 6

Config>
```

If the `Config>` prompt does not appear immediately, press RET again.

Enter SDLC Relay configuration commands at the `SDLC Config>` prompt. To access this prompt, enter **protocol sdlc** as shown:

```
Config>protocol sdlc
SDLC Relay user configuration
SDLC Config>
```

## 6.3 Accessing the SDLC Relay Console Environment

To enter the console environment, enter **talk 5**, or just **t 5**, at the * prompt. This brings you to the console environment as shown:

```
MOS Operator Control

* talk 5
+
```

Enter SDLC Relay console commands at the `SDLC>` prompt. To access this prompt, enter **protocol sdlc** at the + prompt as shown:

```
+ protocol sdlc
SDLC>
```

## 6.4 SDLC Relay Commands

Enter the SDLC Relay configuration commands at the `SDLC config>` prompt and console commands at the `SDLC>` prompt. Table 6–1 lists the SDLC Relay configuration and console commands.

**Table 6–1 SDLC Relay Commands**

| Command | Task | Function |
|---|---|---|
| **? (Help)** | Configure/ Monitor | Lists the configuration and console commands or parameters associated with a command. |
| **Add** | Configure | Adds groups, local ports, and remote ports. |
| **Clear-Port-Statistics** | Monitor | Clears SDLC statistics for the specified port. |
| **Delete** | Configure | Disables or temporarily suppresses groups, local ports, or remote ports. |
| **Disable** | Configure/ Monitor | Disables or temporarily suppresses groups and ports. |
| **Enable** | Configure/ Monitor | Enables groups and ports. |
| **List** | Configure/ Monitor | Displays SDLC Relay and group-specific configurations. |
| **Exit** | Configure/ Monitor | Exits the SDLC Relay configuration or console environment. |

## ? (Help) C M

Lists the commands available from the current prompt level. You can also enter **?** after a specific command to list its options.

**Syntax:** ?

Example: **?**

```
ADD
CLEAR-PORT-STATISTICS
DELETE
DISABLE
ENABLE
LIST
EXIT
```

### Add C

Adds group numbers, local ports, and remote ports.

**Syntax:** add

group
local-port
remote-port

#### group

Assigns a number to a group of primary or secondary ports added to the router.

Example: **add group**

```
Group number: [1]? 1
```

| | |
|---|---|
| *Group number* | The group number that you are designating for the port. |

#### local-port

Identifies the interface that you are using for the local port.

Example: **add local-port**

```
Group number: [1]?1
Interface number: [0]? 0
(P)rimary or (S)econdary:[S}? p
```

| | |
|---|---|
| *Group number* | The group number for the port. This number must match one of the **add group** parameters configured previously. |
| *Interface number* | The interface number of the router that designates the local port. |
| *Primary or Secondary* | The port type, primary (P) or secondary (S). |

#### remote-port

Identifies the IP address of the port directly connected to the serial line on the remote router.

Example: **add remote-port**

```
Group number: [1]? 1
IP address of remote router:[0.0.0.0]? 128.185.121.97
(P)rimary or (S)econdary:[S]? s
```

| | |
|---|---|
| *Group number* | The group number for the port. This number must match one of the **add group** parameters configured previously. |
| *IP address of remote router* | IP address of the interface on the remote router. |
| *Primary or Secondary* | The port type, primary (P) or secondary (S). |

### Clear-Port-Statistics M

Resets the SDLC Relay statistics for all ports. The statistics being cleared include the number of packets forwarded and the number of packets discarded for each group. You can display statistics with the **list group** and **list all** commands.

**Syntax:** clear-port-statistics

Example: **clear-port-statistics**

```
Clear all port statistics? (Yes or No): Y
```

### Delete C

Removes group numbers, local ports, and remote ports.

**Syntax:** delete

group ...
local-port ...
remote-port ...

#### group *group#*

Removes a group (group#) of SDLC Relay configured ports.

Example: **delete group 1**

**local-port** ***interface#***

Removes the local port for the specified interface (interface#).

`Example:` **`delete local-port 0`**

**remote-port**

Removes the remote port for the specified group.

`Example:` **`delete remote-port`**

```
Group number: [1]? 1
(P)rimary or (S)econdary:[S}? S
```

| | |
|---|---|
| *Group number* | The group number for the remote port. |
| *Primary or Secondary* | The port type, primary (P) or secondary (S). |

### Disable C M

Suppresses forwarding for an entire relay group or a specific relay port.

When you use this command within the console process, its effects are not stored in the router's nonvolatile configuration memory.

**Syntax:** disable
group ...
port ...

**group** ***group#***

Suppresses transfer of SDLC Relay frames to or from a specific group (group#).

`Example:` **`disable group 1`**

**port**

Suppresses transfer of SDLC Relay frames to or from a specific local port.

Example: **disable port**

```
Interface number: [0]? 0
(P)rimary or (S)econdary:[S]? s
```

| | |
|---|---|
| *Interface number* | The interface number of the port that you want to disable. |
| *Primary or Secondary* | The port type, primary (P) or secondary (S). |

## Enable C M

Enables data transfer for an entire group or a specific local interface port.

When you use this command within the console process, its effects are not stored in the router's nonvolatile configuration memory.

**Syntax:** enable
group ...
port ...

**group** ***group#***

Allows transfer of SDLC Relay frames to or from the specified group.

Example: **enable group 1**

**port**

Allows transfer of SDLC Relay frames to or from the specified local port.

Example: **enable port**

```
Interface number: [0]? 0
(P)rimary or (S)econdary:[S]? s
```

| | |
|---|---|
| *Interface number* | The interface number of the port that you want to disable. |
| *Primary or Secondary* | The port type, primary (P) or secondary (S). |

### List C M

Displays the configuration or status of a specific group or of all groups.

**Syntax:** list
all
group ...

#### all

Displays the configurations of all local ports.

Example: **list all**

```
 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  address (hex)
-------------   ---------------    ------  --------------- ------------
 1 (E)          Local PRMRY (D)      2
 1 (E)          Remote SCNDRY (E)                          10.2.50.7
 2 (D)          Local PRMRY (D)      0
 2 (D)          Remote SCNDRY (D)                          10.2.50.7
```

**Note:** While the SDLC station address (hex) appears in the listing, it is currently not implemented.

| | |
|---|---|
| *Group Number* | Group number and the status of the group, enabled (E) or disabled (D). |
| *Port Status* | Type of port (local/remote primary/secondary) and its status, enabled (E) or disabled (D). |
| *Net Number* | Interface number of the local port. |
| *IP Address* | IP address of the remote port. |

#### group *group#*

Displays the configuration of a specified group.

Example: **list group 1**

```
 Group Number     Port Status       Net    SDLC Station     IP Address
                                   Number  address (hex)
-------------   ---------------    ------  --------------- ------------


1 (E)           Local PRMRY (D)      2                       10.2.50.7
1 (E)           Remote SCNDRY (E)
```

| | |
|---|---|
| *Group Number* | Group number and the status of the group, enabled (E) or disabled (D). |
| *Port Status* | Type of port (local/remote primary/secondary) and its status, enabled (E) or disabled (D). |
| *Net Number* | Interface number of the local port. |
| *IP Address* | IP address of the remote port. |

### Exit C M

Exits the SDLC Relay configuration or console process.

**Syntax:** exit

Example: **exit**

# A

# DLSw and SDLC MIB Support

## A.1 DLSw MIB

Table A–1 lists the groups, tables, and objects that Digital supports for the DLSw MIB . These are defined in Version 06 of an Internet Draft standard dated 10 October 1995.

Table A–2 lists exceptions within supported tables or groups.

**Table A–1 DLSw Supported MIB Tables, Groups, and Objects**

| Objects | Supported | Not Supported |
|---|---|---|
| **dlswNode Group** | Y | |
| dlswNodeVersion | Y | |
| dlswNodeVendorID | Y | |
| dlswNodeVersionString | Y | |
| dlswNodeStdPacingSupport | Y | |
| dlswNodeStatus | Y | |
| dlswNodeUpTime | Y | |
| dlswNodeVirtualSegmentLFSize | Y | |
| dlswNodeResourceNBExclusivity | Y | |
| dlswNodeResourceMacExclusivity | Y | |
| dlswSdlcLsEntries | Y | |
| **Other Groups and Tables** | | |
| dlswTConnStat | Y | |
| dlswTConnConfigTable | Y | |

**Table A–1 DLSw Supported MIB Tables, Groups, and Objects (Continued)**

| Objects | Supported | Not Supported |
|---|---|---|
| dlswTConnOperTable | Y | |
| dlswTConnSpecific | Y | |
| **Other Tables and Groups** | | |
| dlswTConnTcpConfigTable | Y | |
| dlswTConnTcpOperTable | Y | |
| dlswIfTable | Y | |
| dlswDirStat | Y | |
| dlswDirMacTable | Y | |
| dlswDirNBTable | Y | |
| dlswDirLocateMacTable | Y | |
| dlswDirLocateNBTable | Y | |
| dlswCircuitStat | Y | |
| dlswCircuitTable | Y | |
| dlswSdlcLsEntries | Y | |
| dlswSdlcLsTable | Y | |
| dlswTrapControl | Y | |
| dlswTraps | | Y |

**Table A–2 DLSw Unsupported Objects in Supported Tables**

| Supported MIB Table | Unsupported Objects in Supported Tables |
|---|---|
| dlswCircuitTable | dlswCircutiDiscReasonLocal<br>dlswCircuitDiscReasonRemote<br>dlswCircuitDiscReasonRemoteData |

## A.2 SDLC MIB

Table A–3 lists the Digital supported tables and traps for the SDLC MIB. These are defined in RFC 1747.

Table A–4 lists unsupported objects in the supported tables and groups.

**Table A–3 SDLC Supported MIB Tables**

| SDLC Table or Group Name | Supported | Not Supported |
|---|---|---|
| sdlcPortAdminTable | Y | |
| sdlcPortOperTable | Y | |
| sdlcPortStatsTable | Y | |
| sdlcLSAdminTable | Y | |
| sdlcLSOperTable | Y | |
| sdlcLSStatsTable | Y | |
| sdlcTraps | | Y |

**Table A–4 SDLC Unsupported Objects in Supported Tables**

| Supported MIB Table | Unsupported Objects in Supported Tables |
|---|---|
| sdlcPortOperTable | sdlcPortOperLastFailTime<br>sdlcPortOperLastModifyTime<br>sdlcPortOperLastFailCause |
| sdlcPortStatsTable | sdlcPortStatsPhysicalFailures<br>sdlcPortStatsInvalidAddresses<br>sdlcPortStatsDwarfFrames<br>sdlcPortStatsProtocolErrs<br>sdlcPortStatsActivityTOs<br>sdlcPortStatsRNRLIMITs<br>sdlcPortStatsRetriesExps<br>sdlcPortStatsRetransmitsIn<br>sdlcPortStatsRetransmitsOut |

**Table A–4 SDLC Unsupported Objects in Supported Tables (Continued)**

| Supported MIB Table | Unsupported Objects in Supported Tables |
|---|---|
| sdlcLSOperTable | sdlcLSOperLastModifyTime<br>sdlcLSOperLastFailTime<br>sdlcLSOperLastFailCause<br>sdlcLSOperLastFailCtrlIn<br>sdlcLSOperLastFailCtrlOut<br>sdlcLSOperLastFailFRMRInfo<br>sdlcLSOperLastFailREPLYTOs |
| sdlcLSStatsTable | sdlcLSStatsProtocolErrs<br>sdlcLSStatsActivityTOs<br>sdlcLSStatsRNRLIMITs<br>sdlcLSStatsRetriesExps<br>sdlcLSStatsRetransmitsIn<br>sdlcLSStatsRetransmitsOut |

# B

# Interoperating with the IBM 6611 Router

A number of configuration issues must be addressed for Digital's DLSw implementation to interoperate with that of the IBM 6611 router.

The following sections provide an overview of these issues, and indicate which features of Digital's DLSw implementation are not interoperable with that of the IBM 6611.

**Note:** The issues cited here derive from testing performed with the IBM 6611's MPNP V1.2 software. The issues may not apply to other MPNP software versions.

## B.1 Bridge Configuration Issues

The following are bridge configuration issues:

- The LAN identification (Segment number) of the DLSw must match on both the Digital and IBM 6611 routers. If a mismatch persistently exists, enter the DLSw configuration environment (**T 6**) and select the DLSw protocol. The **set srb** command can then be used to set a Segment Number value that matches the IBM 6611 equivalent.
- The maximum MTU value that can be used for the Bridge Frame is 2100 bytes. This is the largest value currently supported by the IBM 6611. If MTU values less than 2100 are specified, it is important that the configured values match on both the Digital and IBM 6611 routers.
- Currently, Digital interoperates with the IBM 6611 only for SNA traffic over DLSw. The Digital router does not support NetBIOS traffic over DLSw. There is, however, a proprietary Digital solution that permits NetBIOS traffic to be bridged through an IP tunnel.

## B.2 IP-Related Configuration Issues

- The client/server and peer/peer DLSw group feature that enables Digital DLSw neighbors to dynamically find each other is not interoperable with the IBM 6611 DLSw implementation. As a result, the DLSw's **add tcp neighbor** configuration command must be used to define the static IP addresses of adjacent IBM 6611 DLSw peers. However, DLSw group functionality can still be used to locate other Digital routers even though IBM 6611 routers exist in the network.
- The preceding interoperability restriction on the Digital DLSw group feature has implications for the selection of RIP/OSPF:
  - To utilize DLSw groups on a Digital router, the configuration of OSPF/MOSPF is also required. But since these DLSw groups are not interoperable with the 6611, it is possible to configure the Digital DLSw router with only RIP enabled and no OSPF configuration.
  - Although OSPF and RIP can both be enabled on the Digital side, MOSPF (if selected through the OSPF configuration) is not currently supported by the IBM 6611.
  - For the IBM 6611 MPNP V1R2.0 software, the APPN network node implementation on the 6611 only appears to work with RIP.
- Within the Digital IP configuration, make sure that the fill patterns configured for broadcast addresses on a given interface match their equivalent definitions on the IBM 6611.
- Digital's Bandwidth Reservation System (BRS), which can be utilized to guarantee bandwidth for the transport of SNA traffic over DLSw, is not interoperable with the IBM 6611 DLSw implementation.

Although the prioritization assigned by the Digital hardware for BRS can be implemented in an outbound direction, the prioritization order will not be guaranteed if intermediate IP routers do not support BRS. Also, since the 6611 does not support BRS in its end of the line, BRS could only be applicable in a single direction.

## B.3 TCP-Related Issues

- **TCP Connection Break Detection Differences.** If Keepalive is disabled, the Digital DLSw implementation will not detect a broken TCP connection until it attempts to send data on the connection.

- **TCP Connection Reestablishment Differences.** Once a TCP connection is broken, the Digital DLSw implementation reestablishes the TCP connection when a new DLSw SSP_CANUREACH is generated upon receipt of a DLC TEST message from an end station. The IBM 6611 may not exhibit the same behavior.
- **Keepalive Disable/Enable Related Differences.** The Digital DLSw implementation permits the enabling/disabling of a Keepalive option when a TCP neighbor IP address is added (configured). Although TCP in the IBM 6611 DLSw implementation will respond to Keepalive messages received on a TCP session, there is no mechanism to configure the resident 6611 TCP so as to enable the generation of TCP Keepalive messages.
- **Maximum Number of TCP Connections Supported.** In the Digital DLSw implementation, there is no hardcoded restriction on the maximum number of TCP connections supported. As a result, the maximum number of TCP connections supported is directly related to a Digital DLSw Router's available memory. In the IBM 6611 case, there is a hard coded internal restriction of 100 TCP connections that can be supported in the DLSw implementation.

## B.4 DLSw-Related Issues

- The Digital DLSw implementation does not support generation of SSP_IAMOKAY message (SSP Message Type 'x1D') while IBM 6611 DLSw implementation is supported. This SSP message is undocumented in RFC 1434, and is silently discarded by the Digital DLSw implementation upon receipt.
- The IBM 6611 DLSw implementation processes SSP_ENTER_BUSY/EXIT_BUSY messages received from the Digital DLSw implementation but will not generate similar flow control related SSP messages.
- The Digital DLSw implementation does support the user-defined SSP_TEST_CIRCUIT_REQ message (SSP message type 'x7A') that is generated by an IBM 6611 DLSw router functioning as an APPN network node. Upon receipt of this message, the Digital DLSw implementation will return the user-defined SSP_TEST_CIRCUIT_RSP message (SSP message type 'x7B'). This response is expected by the IBM 6611 DLSw router's APPN network node implementation.

## B.5 Miscellaneous Interoperability Issues

- The IBM 6611 chooses to fill bytes in reserved fields with ’xFF’ values, whereas the Digital DLSw implementation zeros these fields whenever SSP Control or Information messages are transmitted. These differences should be noted whenever a Wide Area Sniffer is being used to monitor DLSw SSP messages flowing across a DLSw WAN connection.
- If a problem is encountered when trying to establish a DLSw connection initiated by the IBM 6611, check the IBM 6611 configuration to ensure that MAC address filtering has not been inadvertently enabled for an associated source or destination MAC address.
- Although RFC 1434 does not specifically address the issue of orphan DLSw sessions (for example, DLSw sessions that remain in a DLSw circuit established state with no subsequent activity), both the Digital and IBM 6611 DLSw implementations resolve this issue by providing orphan DLSw session timeouts. DLSw sessions that remain inactive while in a DLSw circuit established state for longer than 30 seconds are eliminated by both implementations.

# Glossary

## A

**Advanced Peer-to-Peer Networking**

See APPN.

**Advanced Program-to-Program Communication**

See APPC.

**APPC**

Advanced Program-to-Program Communication. The general facility characterizing the LU 6.2 architecture and its various implementations in products.

**APPN**

Advanced Peer-to-Peer Networking. An extension of traditional SNA. APPN features greater distributed network control, avoiding critical hierarchical dependencies, and thereby isolating the effects of single points of failure. It also features dynamic exchange of network topology information among network nodes, fostering ease of connection and reconfiguration, adaptive route selection, simplified network definition, and distributed directory lookup.

## B

**BAN**

Boundary Access Node. An enhancement of Frame Relay, bridging, and DLSw functionality enabling remote T2.0 and T2.1 end stations to establish wide-area communication with an IBM front-end processor over Frame Relay links.

**basic transmission unit**

The unit of data and control information passed between path control components.

**bit-oriented protocol**

A protocol that sends data between devices as a steady stream of bits. Clocks at source and destination are synchronized to use a predetermined time interval to determine where characters begin and end. Examples include SDLC and LAPB.

# Glossary

## C

**cache**
An optional part of a directory database in network nodes where frequently used directory information can be stored to speed directory searches.

**cluster controller**
A device that controls the input/output operations of multiple devices attached to it.

## D

**datagram delivery protocol**
A protocol, such as IP or UDP, designed to deliver data in a series of discrete packets. The packets may take different routes to the same destination, and their delivery may not be guaranteed.

**data link layer**
The second layer in the OSI protocol stack, and the one in which bridging occurs.

**Data Link Connection Identifier**
See DLCI.

**Data Link Switching**
See DLSw.

**DCE**
Data Circuit-terminating Equipment. The X.25 term for a device (a modem, for instance), to which an end node attaches.

**DLCI**
Data Link Connection Identifier. A 10-bit field in the Frame Relay header identifying the permanent virtual circuit between the user and Frame Relay device.

**DLSw**
Data Link Switching. Based on RFC 1795, a technique for reliable delivery of SDLC and LLC2 traffic across WANs. DLSw was originally implemented as RFC 1434.

**DSAP**
Destination SAP. The Service Access Point associated with a destination port.

**DTE**
Data Terminal Equipment. The X.25 term for an end node, such as a terminal.

**dynamic routing**
Routing that adjusts automatically to network topology or traffic changes, based on information from routing protocol transmissions.

## E

**encapsulation**
The insertion of protocol information into the data area of another protocol, such as IP or UDP, for transport across a wide area network.

**end system**
See ES.

**End System Hello**
See ESH.

**ES**
End system. In the OSI protocol, a host system that performs the functions of all of the layers of the OSI reference model.

**ESH**
End System Hello. A packet originating in an end system and passing information to an intermediate system.

## F

**FR**
Frame Relay.

**frame**
Informal name for a data link packet data unit. Control information in the frame provides addressing, sequencing, flow control, and error control to the respective protocol levels.

## H

**HDLC**
High-level Data Link Control. An ISO standard bit-oriented data link protocol that specifies the encapsulation method of data on synchronous data links.

**Hello/I-H-U**
Hello and I-Heard-You. An EGP protocol that requests and confirms neighbor reachability.

**High-level Data Link Control**
See HDLC.

# I

**I-Frame**
Information Frame.

**IGP**
Interior Gateway Protocol. A protocol that distributes routing information to the routers within an autonomous system.

**IP**
Internet Protocol. The Department of Defense (DoD) Internet standard protocol that defines the Internet datagram as the unit of information passed across the Internet. IP corresponds to the OSI reference model layer 3 and provides connectionless datagram service.

**IP datagram**
A packet containing IP control information exchanged between network entities.

# L

**link station**
An SDLC station with which a link has been established. Each SDLC link station has either a primary or secondary role in the communication process.

**logical unit**
See LU.

**Low-Entry Networking**
A capability in Type 2.1 nodes, allowing them to attach directly to one another using peer-to-peer protocols and allowing them to support multiple parallel sessions between logical units. LEN nodes have no APPN routing capability.

**LU**
Logical Unit. A type of network accessible unit that enables end users to gain access to network resources and communicate with one another.

**LU type**
The classification of an LU in terms of the specific subset of SNA protocols and options it supports for a given session.

## M

**MAC**

Media Access Control. The sublayer of the data link control layer that supports media-dependent functions. It includes the medium-access port. MAC protocols put packets from upperlevel protocols into the frame format of the destination network.

**Media Access Control**

See MAC.

**MIB**

Management information base. A database of managed objects accessed from a network management protocol.

**modem eliminator**

A device permitting the connection of two DTE devices without a modem.

**MOSPF**

Multicast OSPF. A protocol required for use of DLSw group functionality.

## N

**NAU**

Network accessible unit. A logical unit (LU), physical unit (PU), system services control point (SSCP), or control point (CP).

**Network accessible unit**

See NAU.

**network layer**

Layer 3 of the OSI reference model at which all routers operate.

**network name**

The symbolic identifier by which end users refer to a network accessible unit, a link, or a link station within a given network.

**node type**

A designation of a node according to the protocols it supports and the network accessible units that it can contain.

**NRZ**

Non-return to zero.

**NRZI**

Non-return to zero inverted.

**NSAP**

Network Service Access Point. The point at the layer boundary where the communications capability of the network layer is made available to its users. An OSI network address.

## O

**Open Shortest Path First**

See OSPF.

**OSI**

Open Systems Interconnection. The ISO architecture for internetworking.

**OSI reference model**

The seven-layer model of computer network architecture and its data functions, specified by ISO.

**OSPF**

Open Shortest Path First. A link-state protocol that IGPs use to exchange routing information between routers.

## P

**packet**

A self-contained block of data containing control and user information transmitted across a network.

**packet switching**

A data transfer scheme in which information is broken into individual packets, transferred across a communications link, and reassembled at the receiving end. In a packet-switching system, each node through which the packet travels determines the route to the next receiver with no previously-established communication path.

**peer-to-peer communication**

Communication between two nodes in an SNA network not requiring explicit mediation by a system services control point.

**physical unit**

See PU.

**PLU**

Primary logical unit. The logical unit that sends a BIND to activate a session with its partner LU.

**port**

The representation of a physical connection to the link hardware.

**Primary logical unit**

See PLU.

**PU**

Physical unit. The component that manages and monitors the resources associated with a node, as requested by an SSCP via an SSCP-PU session. This term applies to Type 2.0, Type 4, and Type 5 nodes.

## R

**RIF**

Routing Information Field. A field in the Token Ring 802.5 header generated by a source node and used by a source route bridge to determine the path a packet must use when passing through a Token Ring network segment.

**RIP**

Routing Information Protocol. A distance-vector IGP used to exchange routing information between routers.

**route**

An ordered sequence of nodes that represent a path from an origin node to a destination node traversed by the traffic exchanged between them.

**routing**

The assignment of a path by which a message can reach its destination.

**Routing Information Field**

See RIF.

**Routing Information Protocol**

See RIP.

**RS-232**

A type of serial interface.

## S

**SAP**

Service Access Point. The interface between a layer in the OSI protocol stack and the layer above. Generally, SAP is preceded by a letter denoting the layer providing the service (for example, network layer services are NSAPs). Well-known services are associated with well-known SAP numbers.

# Glossary

**SDLC**

Synchronous Data Link Control. A link level protocol designed for transfer of information in LAN environments. Transmission exchanges may be full-duplex or half-duplex over switched or nonswitched links. The configuration of the link connection can be point-to-point, multipoint, or looped.

**SDLC Relay**

A Digital product that supports exchange of bit-oriented protocols across the wide area.

**segment number**

A number that identifies an individual LAN, such as a single Token Ring or a serial line.

**serial interface**

An interface that supports connections via serial line.

**Service Access Point**

See SAP.

**session**

A logical connection between two network accessible units that can be activated, tailored to provide various protocols, and deactivated as requested. Each session is uniquely identified in a transmission header accompanying messages exchanged during transmission.

**session limit**

The maximum number of concurrently active LU-LU sessions that a particular LU can support.

**SNA**

Systems Network Architecture. A proprietary networking architecture used by IBM and IBM-compatible mainframes.

**SNA network**

The part of a user-application network that conforms to SNA formats and protocols. It enables reliable transfer of data among end users and provides protocols for controlling the resources of various network configurations. It consists of network accessible units: boundary function, gateway function, intermediate session routing function components, and the transport network.

**SRLY**

See SDLC Relay.

**SSAP**

Source SAP.

**SSCP**

System Services Control Point. A component within a subarea network for managing the configuration, coordinating network operator and problem determination requests, and providing directory services and other session services for end users of an SNA network. Multiple SSCPs can cooperate as peers, dividing the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its own domain.

**SSCP-PU session**

A session between a System Services Control Point and a physical unit.

**subarea**

A portion of the SNA network consisting of a subarea node, any attached peripheral nodes, and their associated resources. Within a subarea node, all network accessible units, links, and adjacent link stations that are addressable within the subarea share a common subarea address and have distinct element addresses.

**subarea network**

Interconnected subareas, their directly attached peripheral nodes, and the transmission groups that connect them.

**subnet**

In IP, a distinct network within a network. In OSI, the connection from the IS to the subnetwork.

**subnet address**

An extension of the IP addressing scheme that allows a site to use a single IP address for multiple physical networks.

**Synchronous Data Link Control**

See SDLC.

**System Services Control Point**

See SSCP.

## T

**TCP**

Transmission Control Protocol. A protocol in the TCP/IP suite of protocols that implements transport functions on the internet

**TCP/IP**

Transmission Control Protocol/Internet Protocol.

**token**

In a local area network, the symbol of authority passed among data stations to indicate the station temporarily in control of the transmission medium. The token becomes a frame when a station appends data to it.

**Token Ring**

A network with a ring topology that passes tokens from one attaching device to another. Examples include FDDI networks and the IBM Token Ring network.

**transparent bridging**

A bridging mechanism implemented by software on bridges and invisible (transparent) to end stations.

**Type 2.0 node**

An SNA peripheral node that requires the services of a PU5 (T5) subarea host in order to communicate. Type 2.0 nodes are known as PU2.0 or T2.0 nodes; the terms are used interchangeably. A 3270 terminal cluster controller (for example, an IBM 3174) is an example of a T2.0 node. T2.0 nodes do not perform dynamic link configuration, and when SDLC-attached, function only as SDLC secondary devices.

**Type 2.1 node**

An SNA peripheral node (T2.1) that has the capability to support communication with another T2.1 node without the mediation of a PU5 (T5) subarea host node. T2.1 nodes come in three basic types with increasing network capabilities: LEN nodes, APPN End Nodes (ENs), and APPN Network Nodes (NN). All three perform dynamic link configuration using XID3s during link activation negotiation. When SDLC-attached, T2.1 nodes can function as SDLC secondary or primary devices (including initial dynamic link role negotiation). DLSw is capable of carrying SNA traffic between all three T2.1 types. An IBM AS/400 is an example of a T2.1 node.

# Index

## C



## D

## E



## F



## G



## I



## L



## M

## N



## O



## P



## R



## S

## T



## U



## W