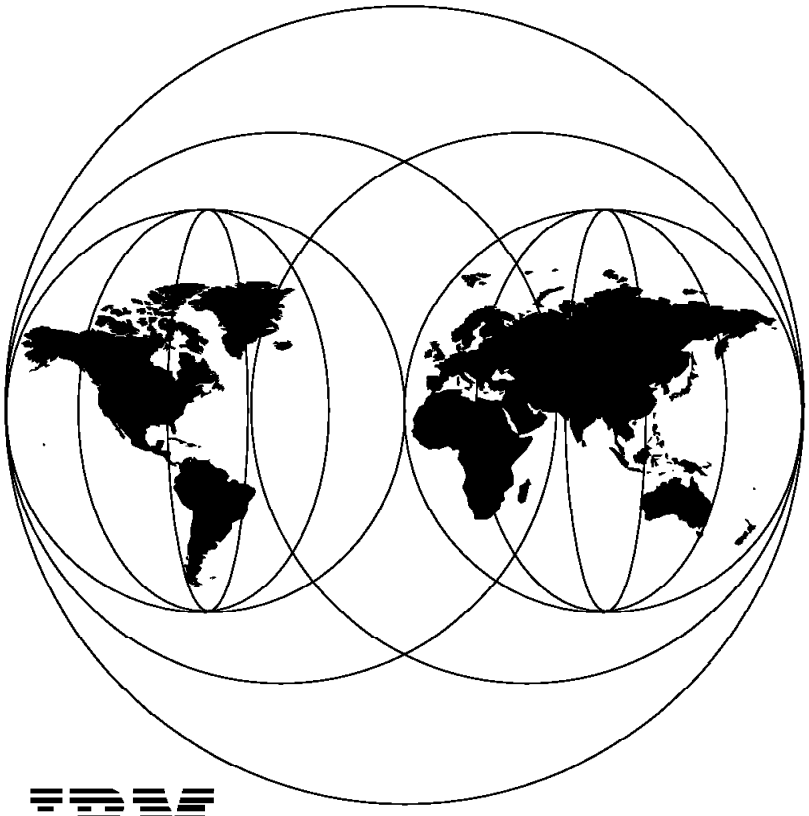
SG24-4916-00

IBM Communications Server for OS/2 Warp Version 4.1 Enhancements

February 1997



IBM

International Technical Support Organization
Raleigh Center

International Technical Support Organization

SG24-4916-00

IBM Communications Server for OS/2 Warp Version 4.1 Enhancements

February 1997

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix H, “Special Notices” on page 333.

First Edition (February 1997)

This edition applies to Version 4, Release Number 1 of the IBM Communications Server for OS/2 Warp, Program Number 5801-AAR (84H1802) for use with the OS/2 Warp Operating System.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Contents

Preface

This redbook will help you to understand the latest release of IBM Communications Server for OS/2 Warp Release 4.1. The redbook describes and positions CM/2′s evolution into Personal Communications and the Communications Server. It focuses on the new functions and enhancements implemented in this product, and will help you plan, install, and configure the new functions quickly in a wide variety of environments.

Some knowledge of previous releases of the IBM Communications Server for OS/2 Warp, as well as a basic knowledge of networking concepts and terminology used in the Systems Network Architecture (SNA) and TCP/IP, is assumed.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the Systems Management and Networking ITSO Center, Raleigh.

Juan R. Rodriguez is a Networking Specialist at the Systems Management and Networking ITSO Center, Raleigh. He holds an M.S. degree in Computer Science from Iowa State University, writes extensively and teaches IBM classes worldwide on areas such as Networking and Data Security. Before joining the ITSO a year ago, he worked at the IBM laboratory in Research Triangle Park (North Carolina, USA) as a designer and developer in networking products.

Max Strableg is a Systems Engineer in IBM Austria. He has ten years of experience in workstation communication networks such as IBM LAN Server, Communications Manager and Novell NetWare in OS/2 and Windows platforms. He wrote several chapters and worked in most of the scenarios for this redbook.

Cesar Portocarrero is a Pre-Sales Support for the Network Software Division in IBM Andean. He has five years of experience in networking products such as VTAM/NCP, NPSI, CM/2, PCOMM, TCP/IP, SNA/Server for AIX and communications controllers. He holds a degree in Industrial Engineering and completed a Masters degree in Systems Engineering at the Universidad de Lima (Lima, Peru). His areas of expertise include SNA/APPN/HPR, TCP/IP architectures, X.25, SDLC, Frame Relay and LAN communication protocols mainframes, PC-based systems, AS/400 and RS/6000 platforms.

Jochem Just is a Customers Engineer in IBM Germany. He has 4 years of experience in OS/2 software support in the IBM German support organization. His areas of expertise include knowledge of early releases of IBM Extended Services, Communications Manager and Communications Server.

Jose J. Lira is a Technical Support Representative for the Availability Services Division in Peru. He has two years of experience in Local Area Networks. He holds a degree in Electronics Engineering from Universidad Catolica in Lima, Peru. His areas of expertise include OS/2, LAN Server and PC Servers.

Thanks to the following people for their invaluable contributions to this project:

John Mitchell
Bart Vashaw

Suvas Shah
Tom Carey
Bob Gibson
IBM Research Triangle Park, NC

Brian Elkins
IBM ITSO, Raleigh Center

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 347 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Home Pages at the following URLs:

  For Internet users `http://www.redbooks.ibm.com`
  For IBM Intranet users `http://w3.itso.ibm.com/redbooks`

- Send us a note at the following address:

  `redbook@vnet.ibm.com`

Part 1. Overview

Chapter 1. Release Overview

The new IBM Communications Server Release 4.1 product provides networking support to allow applications to run in different environments. The IBM Communications Server Release 4.1 allows client/server applications using interfaces such as APPC, CPI-C and Sockets to run over SNA networks, TCP/IP networks or both.

The new IBM Communications Server Release 4.1 includes new functionalities and important enhancements such as:

- IBM Communications Server Release 4.1 multiprotocol enhancements
- TN3270E Server

  This server delivers 3270 terminal and printer emulation to TCP/IP users using open standards and multivendor solutions. (See *TN3270 Server Profile* for information on the TN3270E Server.)
- LAN Gateway

  The LAN Gateway function enables workstations, requesters, or servers located on different local area networks (LANs) to communicate across SNA on TCP/IP wide area networks (WANs). The LAN Gateway supports both Novell NetWare Internet Packet Exchange (IPX) and NetBIOS protocols across WANs. Each LAN attaches to the WAN through one of the following LAN Gateways:

  – IPX over SNA
  – IPX over TCP/IP
  – NetBIOS over SNA
  – NetBIOS over TCP/IP

  (See the *Guide to AnyNet LAN Gateway* for more information about LAN Gateways.)
- AnyNet Multiprotocol Support - Sockets over SNA Enhancements
  – The enhancements to Communications Server Version 4 Sockets over SNA support are described in the *Guide to AnyNet Sockets over SNA*.
  – Backup and Load Balancing
  – Datagram Retry Delay
  – Route Discovery
  – Routing Information Protocol (RIP)
  – Maximum number of connections has been increased to 2000
  – Variable Subnetting Support
- SNA Gateway Enhancements

  The SNA Gateway function has been enhanced with additional pooling capabilities.
- APPN
  – APPN Backup Link

This support provides a backup link capability that can be used to access an alternate node link if activation of the primary link fails.

  - Non-Limited Resource for Connection Networks

    This support enables the user to define a connection network as a non-limited resource so that the sessions and links will not be dropped if there is a period of time with no conversations.

- Dependent LU Support

  The following enhancements have been added to the support of dependent LUs:

  - Multiple PU support on a single SDLC link
  - Dedicated PUs using a single SDLC link
  - Additional pooling capabilities
  - DLUS-Served LU Registration

    This function allows an end node to register its LU so that the network node can locate these LUs without having to pass the locate requests to the DLUR station.

- Frame Relay Support

  This function provides 802.5 (SNA and IP) traffic over frame relay and HPR over frame relay as defined by RFC 1490. (See the *Frame Relay User's Guide* for information about the Communications Server frame relay support.)

- Programming Support

  CPI-C applications are now able to run in a WIN-OS/2 session.

  - CPI-C Support for Win-OS/2 Applications

    This function provides the CPI-C API for WIN-OS/2, enabling Windows (3.1x) CPI-C applications to run in a WIN-OS/2 session.

- User Control of Unlocked Shared Storage Limit

  OS/2 has a block of storage that applications share. This support allows users to control how much of the OS/2 storage Communications Server uses and reduces the default amount to 4 MB.

- IBM Communications Server Release 4.1 performance improvements

1.1 CM/2 Evolution to Communications Server

CM/2 is a very popular product that has proved to be robust by combining emulation, SNA Gateway and APPN functions into one product.

In 1996 CM/2 functions were divided into two products. The *Desktop* function, including emulation and APPC support, has been moved to be part of the *Personal Communication* family of products. The *Server* function has been moved to the Communications Server. The changes represent the natural evolution of the product. Great care was taken to insure that Communications Server and PCOM will smoothly integrate with your existing systems and applications. Upgrading is easy and your investments in existing applications are maintained. The evolution is shown in Figure 1 on page 5.

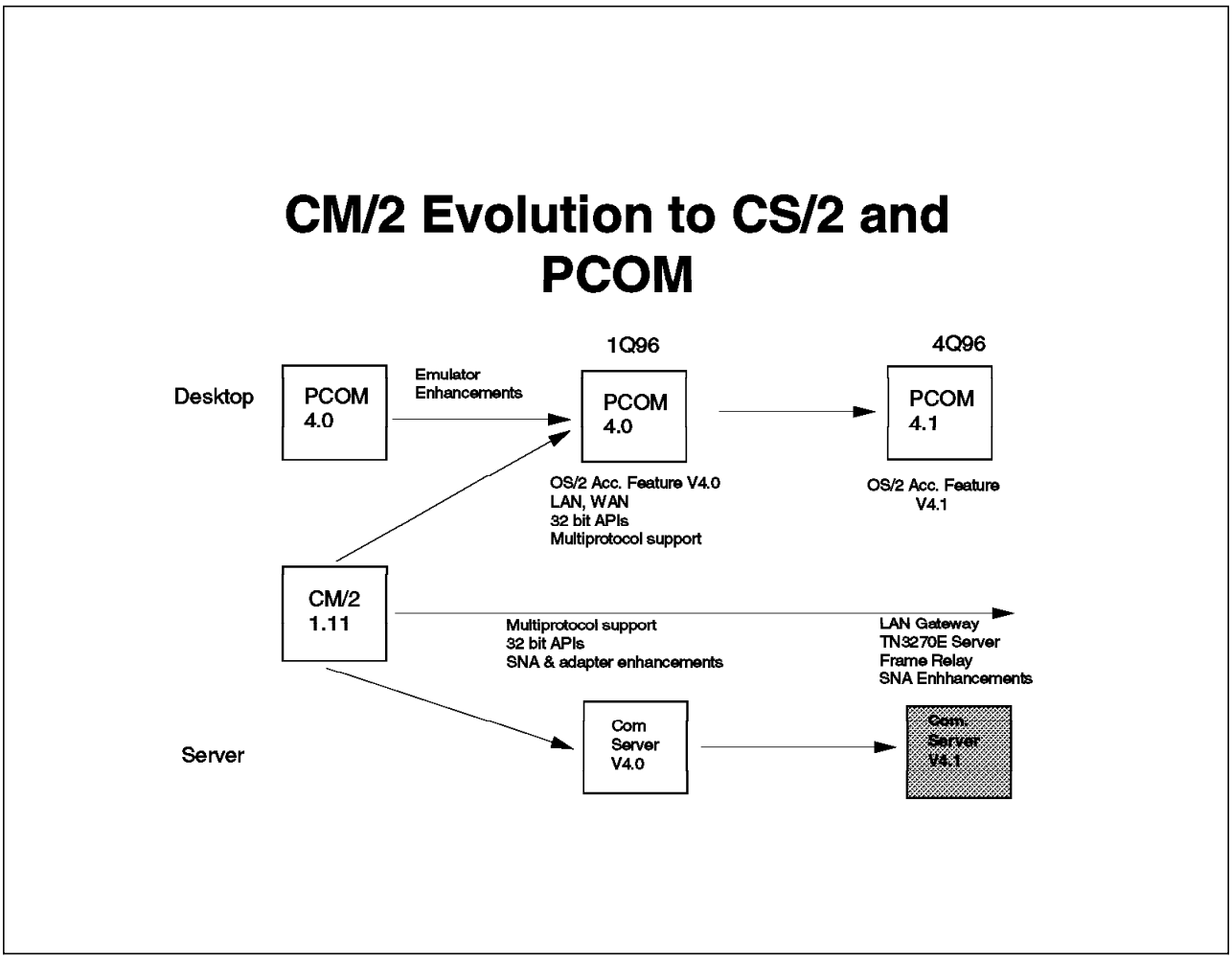


Figure 1. Evolution

The CM/2 desktop emulation function follow-on product is PCOM. The PCOM emulator is the next step in emulator technology. It provides many enhancements to the CM/2 emulator. It has an icon toolbar, provides macro support and many other features. PCOM also includes the OS/2 Access Feature for the desktop. This provides LAN, WAN, APIs (APPC), and AnyNet access nodes (Sockets over SNA and SNA over IP). Version 4.0 AF runs on OS/2 2.11 or later. Version 4.1 AF runs on OS/2 V3.0 (or later) and has a smaller footprint.

The CM/2 Gateway and APPN network node function follow-on product is Communications Server. Communications Server continues the tradition of providing outstanding SNA support. It also provides the technically sophisticated AnyNet family of multiprotocol products:

- IBM Communications Server Release 4.0 included a focus on multiprotocol support with AnyNet Sockets over SNA and SNA over IP Gateway and access node support.
- V4.1 continues the focus on multiprotocol support with LAN Gateway (IPX and NetBIOS over IP and SNA) and TN3270E server support.
- V4.1 also has frame relay support and SNA enhancements.

1.2 Product Packaging

In this section, we indicate how the IBM Communications Server Release 4.0 and IBM Communications Server Release 4.1 are packaged as shown in Figure 2 and Figure 3 on page 7.

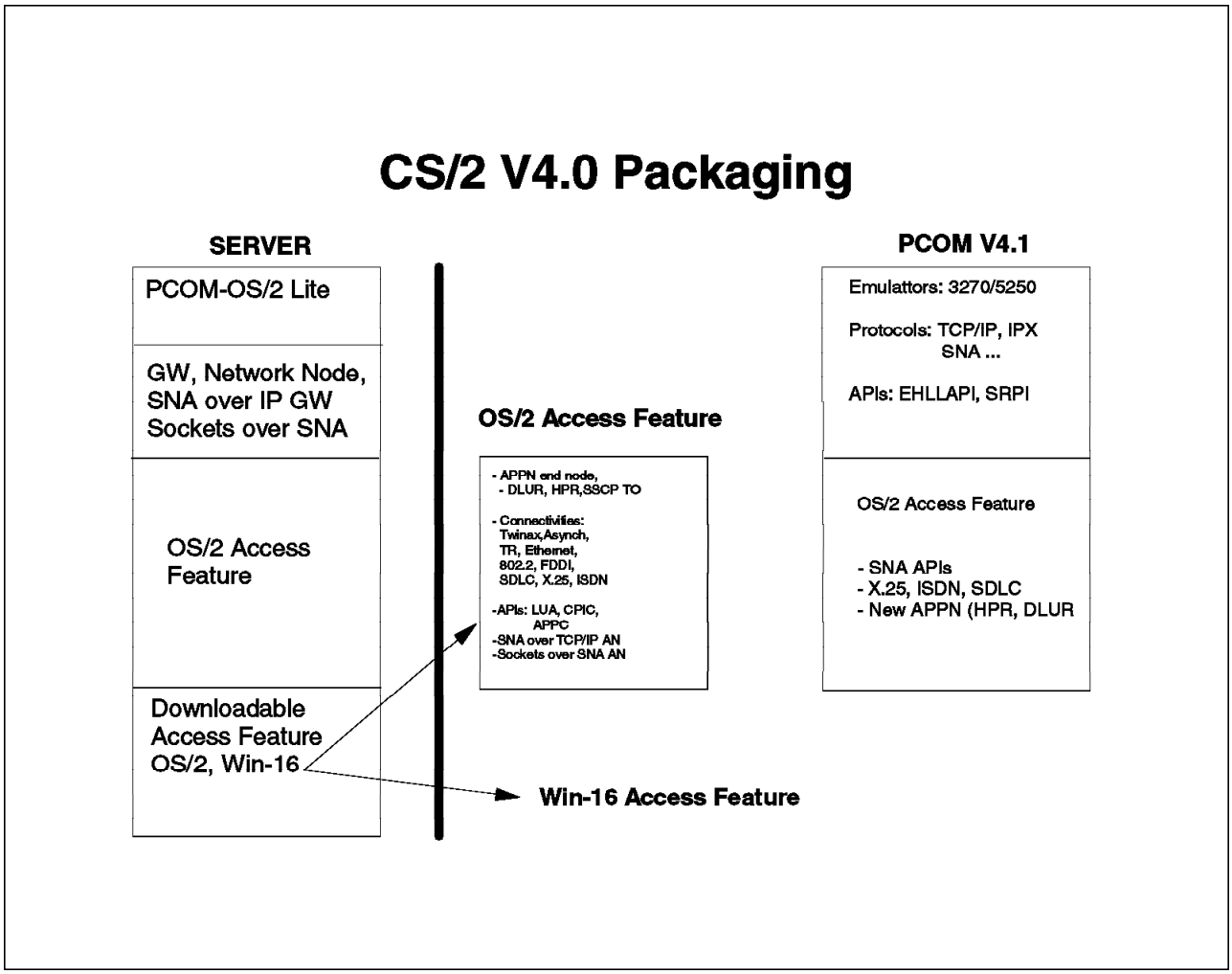


Figure 2. Packaging IBM Communications Server Release 4.0

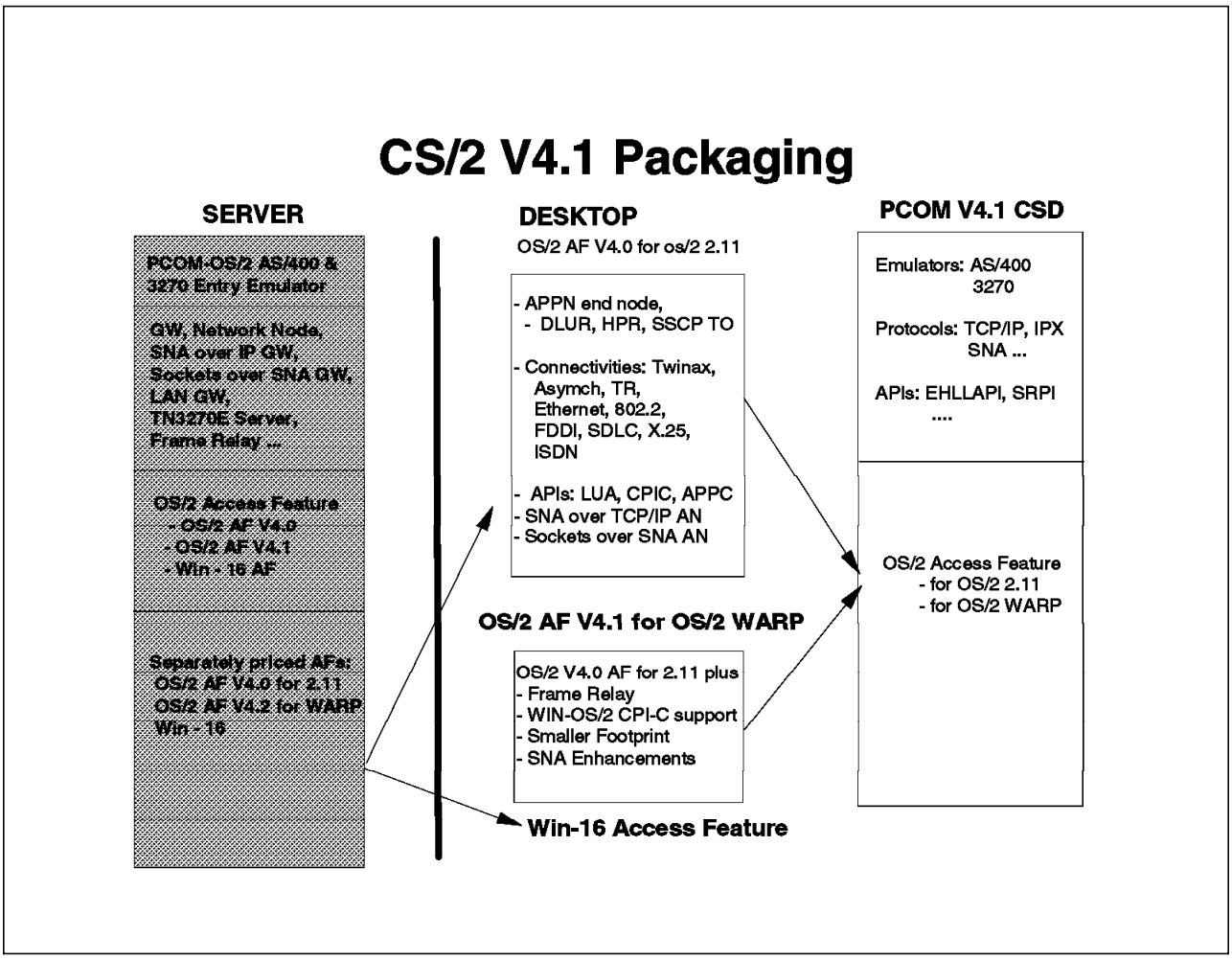


Figure 3. Packaging IBM Communications Server Release 4.1

The IBM Communications Server Release 4.1 package has three enhancements:

1. The Personal Communications AS/400 and 3270-APPC/LUA Entry Level emulator for OS/2. It is a subset of the PCOM V4.1 emulator. It is a small subset of the full function which supports only *two* sessions. This emulator is intended for administrative use on the server only. It is not licensed for use on the desktop.
2. IBM Communications Server Release 4.1.
3. The three access features:
   - The OS/2 Access Feature Version 4.0 for OS/2 2.11
   - The OS/2 Access Feature Version 4.1 for OS/2 3.0 (WARP) and later
   - Windows Access Feature for Windows 3.1

Please refer to Figure 2 on page 6 and Figure 3.

Note

The CS/2 Version 4.1 License is for the server, one of the access features and the entry level emulator for use on one server only.

1.2.1 OS/2 Access Feature

The OS/2 Access Feature distributed with Communications Server provides applications with local and wide area connectivity support. It includes the following functions:

- APPN/HPR end node support
- Low Entry Networking (LEN) support
- 16-bit and 32-bit APIs (LU_A, APPC, CPI-C, X.25 and others)
- Sockets over SNA access node. It allows sockets applications to run over SNA networks.
- SNA over TCP/IP access node. It allows SNA applications to run over TCP/IP networks.
- SNA over TCP/IP downstream dependent LU support (access). It allows SNA applications for dependent LUs to run over TCP/IP networks.
- Data Link Control support (DLCs)
- LU-Application (LU-A)
- 3270 Entry Level emulator using LU-A
- Dependent LU Requester (DLUR)
- MPTS

Figure 4 on page 9 and Figure 5 on page 10 show the difference between CS/2-OS/2 Access Feature 4.0 and CS/2-OS/2 Access Feature more clearly.

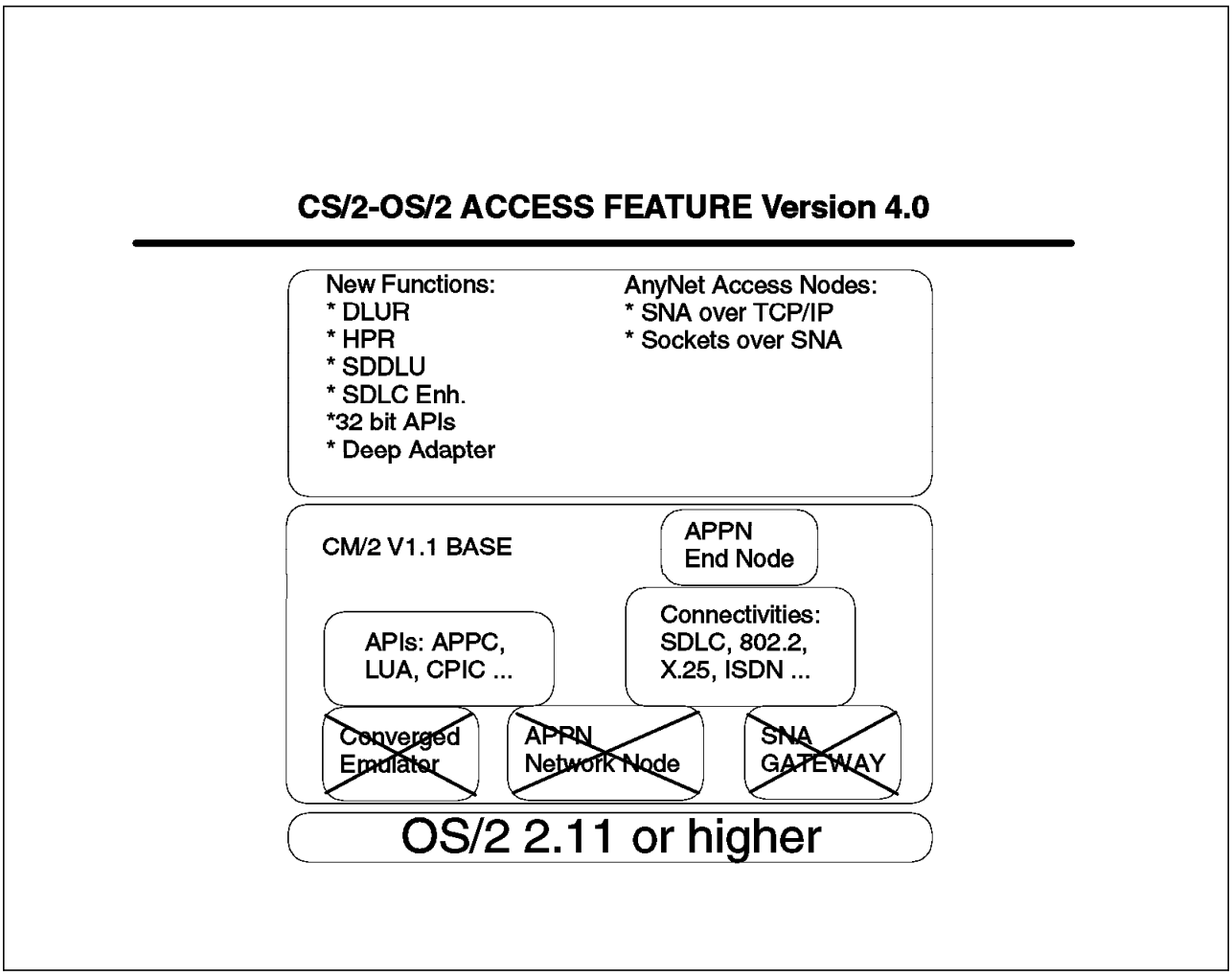


Figure 4. Access Feature V4.0

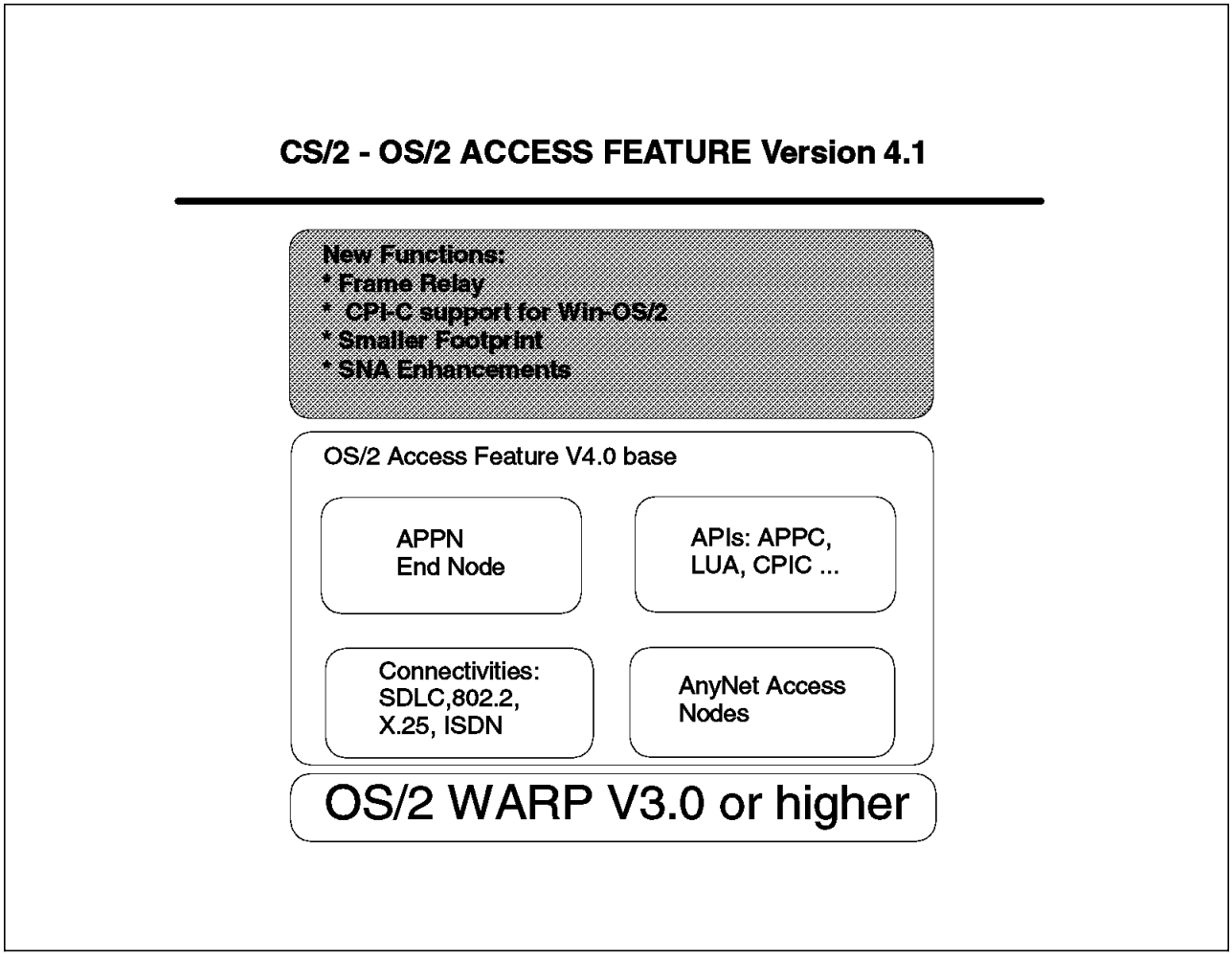


Figure 5. Access Feature V4.1

1.2.2 Windows Access Feature

Windows Access Feature is the compilation of the following products:

- APPC Networking Service for Windows Version 1.0
- AnyNet APPC over TCP/IP Version 1.0 for Windows
- LAN Support Program (distributed in disk images)

The bundling of these products provides APPC programming support and enables APPC applications to run unchanged over either SNA or TCP/IP local and wide area networks. In other words, the Windows Access Feature provides protocol transparent APPC application development and utilization for the Windows platform. The Windows Access Feature includes the IBM LAN Support Program which provides the necessary LAN access.

Part 2. Client/Server Programming

Chapter 2. WIN - OS/2 CPI-C Applications

This chapter provides information about support and configuration of the WIN-OS/2 CPI-C Communications support for IBM Communications Server Release 4.1.

Note

APPC applications are not supported with this function.

2.1 Overview

The main points to be considered are:

1. Support Windows 3.1x CPI-C applications running under WIN-OS/2 using Communications Server, enabling the use of Windows 3.1x applications in a WIN-OS/2 session.
2. Binary-compatible with NS/Windows applications. All of the calls are supported except for the calls relating to non-blocking, duplex and the extract TP name (cmetpn) call.
3. Support for solution providers such as CICS Client for Windows and the Rocket Shuttle applications. This preserves the already developed application providing a more robust platform.
4. Use of the virtual device driver technology to communicate between WIN-OS/2 and OS/2.

Figure 6 on page 14 gives an overview of how this support works.

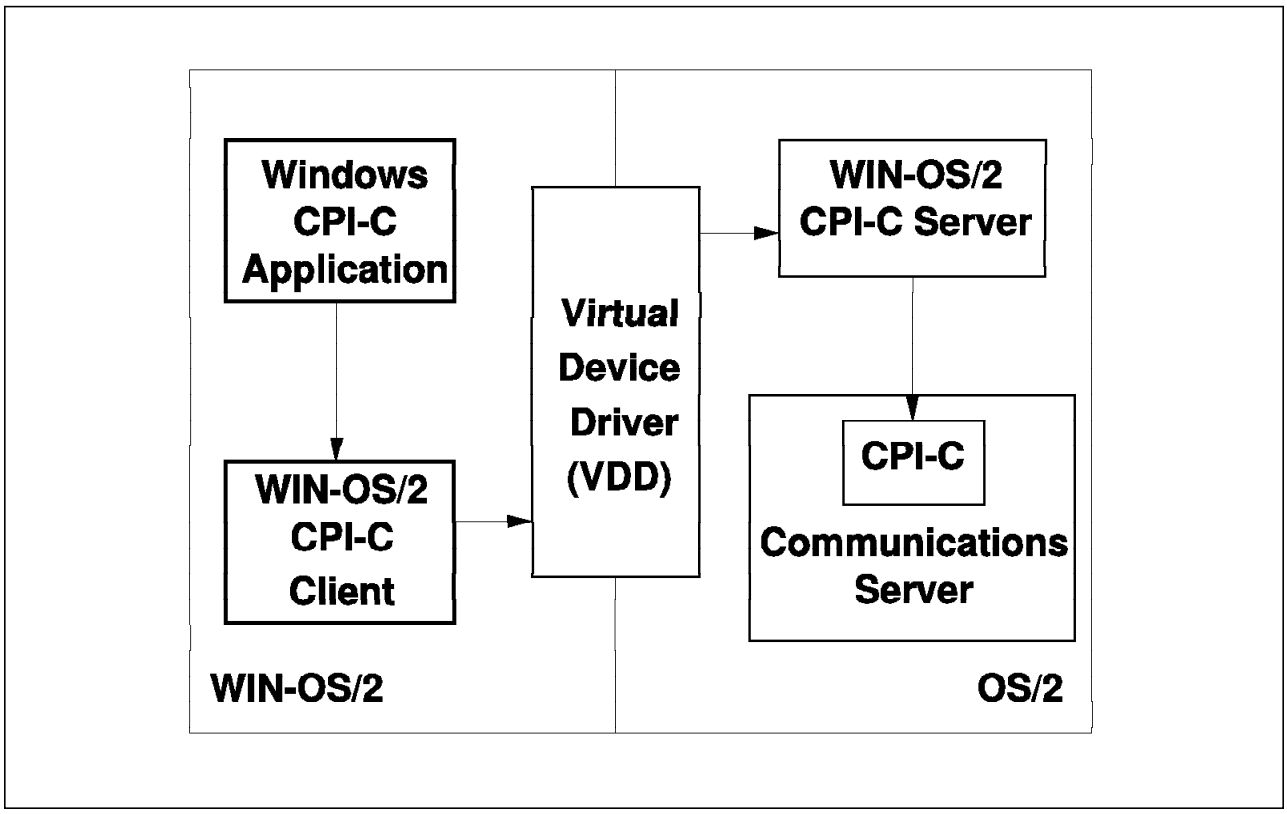


Figure 6. WIN-OS/2 CPI-C Support Overview

2.2 Functionality

Some points to remember that explain the functionality provided are:

- CPI-C API extended to WIN-OS/2 environment.
- Support existing Windows CPI-C applications.
- NS/Windows applications using only CPI-C calls should run unchanged.
- Other applications may need to be re-linked or re-compiled.
- 16-bit C language API only.

2.3 Installation

You can install this feature using the Communications Server Setup window. Select **Options** and then select **Install additional functions**, as shown in Figure 7 on page 15.

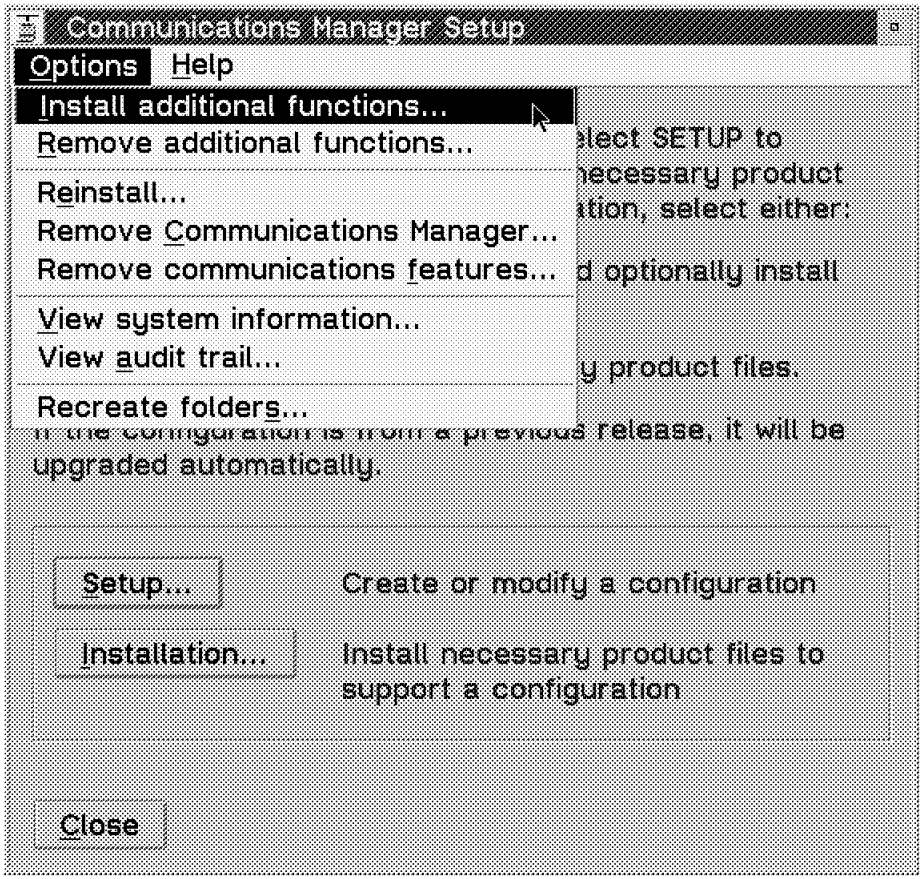


Figure 7. Communications Server Setup

Now you can install WIN-OS/2 CPI-C Support as the base support. Optionally, you can choose WIN-OS/2 Toolkit and Samples in order to install the files needed to compile and link Windows CPI-C programs and a sample application (APING) (refer to Figure 8 on page 16).

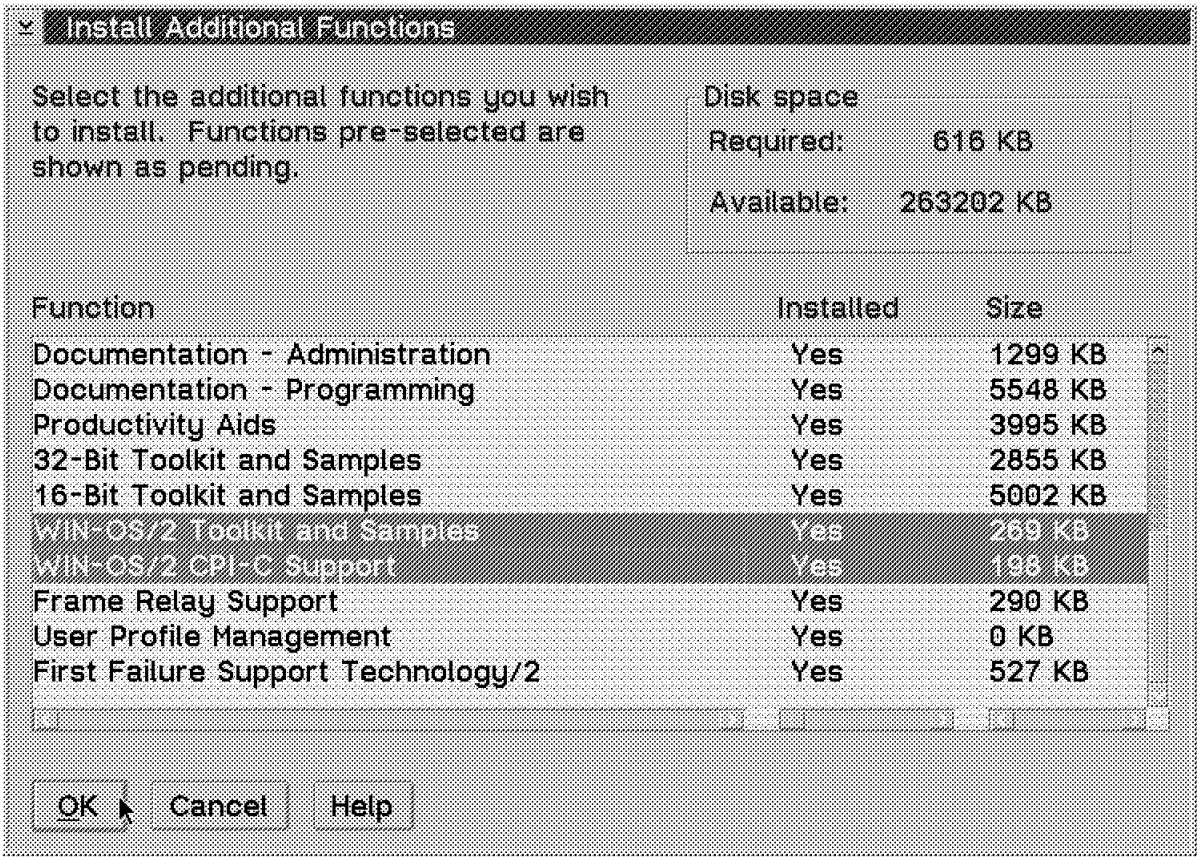


Figure 8. Install Additional Functions

If you want to install this support using the response file, use the following two keywords:

CMINSTALLWINCPIC = 1 (Installs the base support)

CMINSTALLWINCPICAPI =1 (Installs the Toolkit)

After the installation Communications Server updates the following files:

- CONFIG.SYS

  DEVICE=C:\CMLIB\CMQVDD.SYS

  This is the virtual device driver.

- AUTOEXEC.BAT

  C:\CMLIB\DLL is appended to the PATH= statement

  This allows the dynamic link libraries to the WIN-OS/2 CPI-C to work.

2.4 Configuration

From the SNA Features list you could define the new TP. WINDOW is a new program type that replaces the previous VIO_WINDOWABLE.

The other new option DEFAULT makes this decision based on the EXE type of the program.

The DEFINE_TP and DEFINE_DEFAULTS keywords support the new program types in the .NDF file and TP and SNA_DEFAULTS keywords support the new program types in the .RSP files.

Figure 9 and Figure 10 show how the configuration panels look.

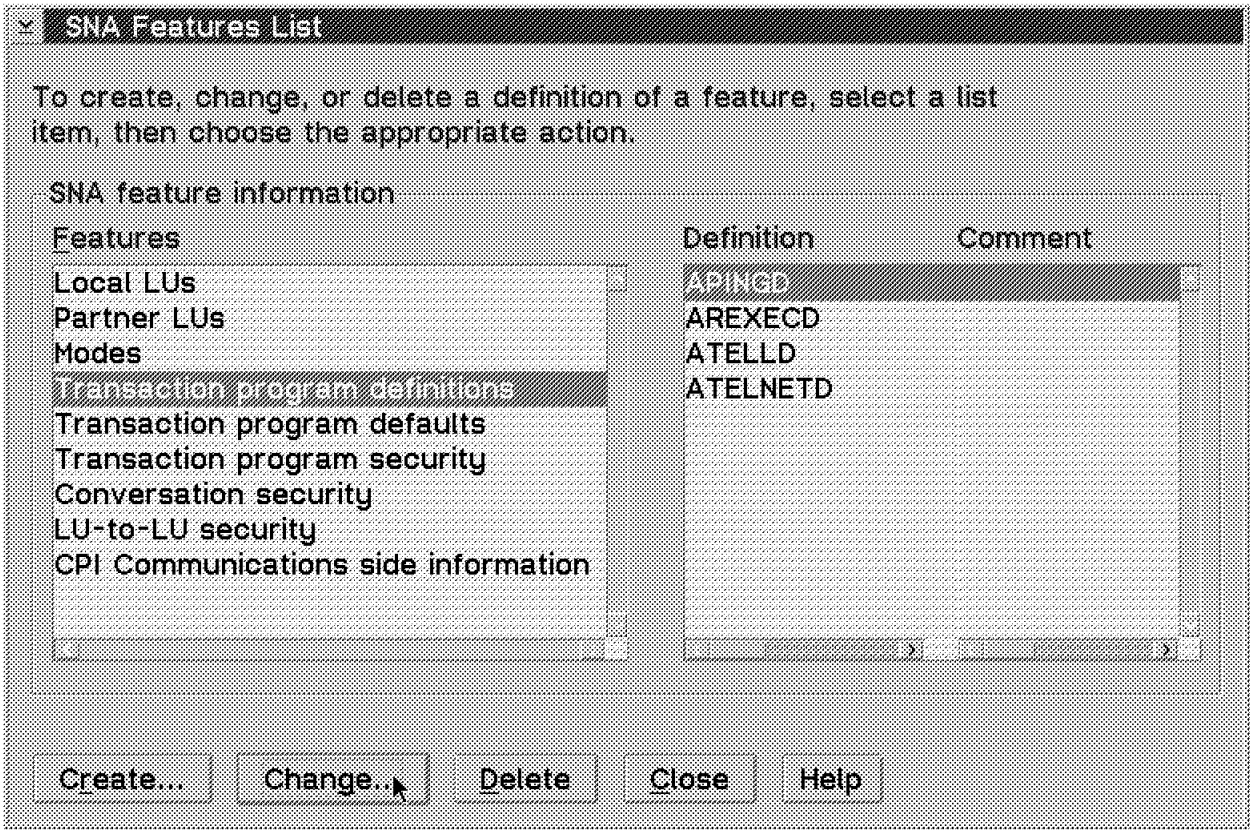


Figure 9. Install Additional Functions

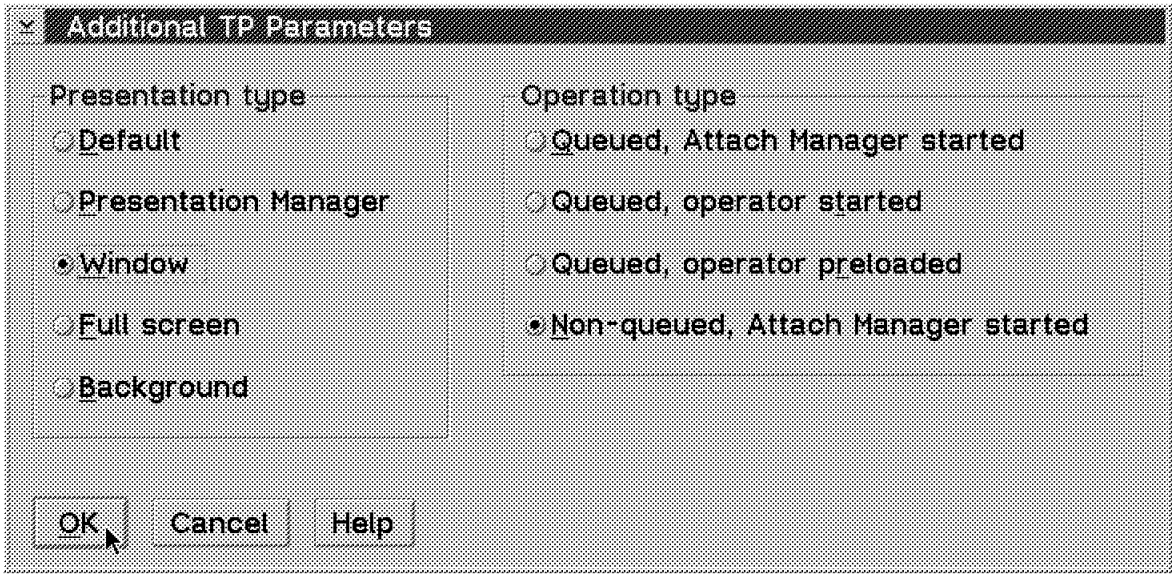


Figure 10. Install Additional Functions

In this case you could use the default window or full-screen presentation type. We prefer a window because it looks better.

2.5 Toolkit

The toolkit support is provided in the \CMLIB\TOOLKWIN directory.

Applications can include either the CMC.H or the CPIC.H (located in \CMLIB\TOOLKWIN\H) header files. Both files are provided with the WIN-OS/2 Toolkit.

Applications should link with NSDW.LIB (located in \CMLIB\TOOLKWIN\LIB).

WIN-OS/2 CPI-C support is for Windows 3.1, 16-bit C language applications.

Since the name of the CPI-C DLL provided with WIN-OS/2 CPI-C matches the name of the CPI-C DLL provided with NS/Windows, you do not need to re-compile or re-link existing NS/Windows applications. However, there are CPI-C calls that are not supported. In this case the application will receive a CM_CALL_NOT_SUPPORTED return code when the call is issued.

For CICS Client for Windows the value of the local LU in the INI file must be the same as the default LU specified in the Communications Server configuration. Review DEFINE_LOCAL_CP in the NDF file.

Note

CICS security requires PTF UN87556 (CICS Client for Windows).

Only the default LU is supported by WIN-OS/2 CPI-C. The APPCLLU environment variable can not be used to change the Local LU from the default LU.

WIN-OS/2 CPI-C does not support the use of the APPCTPN environment variable to specify the TP name to be used for an Accept_Conversation. The TP name for inbound conversations must do either of the following:

- Be set explicitly with a Set_TP_Name (CMSTPN) call between the Initialize_For_Incoming (CMACCI) calls
- Match the executable file name without the extension

Remember this if you want to configure the APINGD for both environments. For example, you could define the APINGD for OS/2 as usual in the CMLIB directory with the TPNAME APINGD and you could define the APINGD program for WIN-OS/2 as the TPNAME APINGWD with the \CMLIB\TOOLKWIN\SAMPLES\SNA\APINGW\APINGWD.EXE path. Then you could rename the original APINGD.EXE for WIN-OS/2 to APINGWD.EXE.

In this way if you want to invoke the server program in OS/2, you do not have to make any changes in your APING client invocation command, for example: `APING NETID.LUNAME`

If you want to invoke the WIN-OS/2 server, you must use the TPNAME option as follows: `APING -t APINGWD NETID.LUNAME`

Note

Please notice that TPNAMEs are case-sensitive.

2.6 APING for WIN-OS/2

In this section we show you the procedure to run the APING program on WIN-OS/2. When you want to run the APING applet for WIN-OS/2 you must be running in a WIN-OS/2 window. However, you only have to invoke the program in an OS/2 window and OS/2 will open a WIN-OS/2 window for you.

You must first define the parameters that you want to use, such as the partner LU, security, and so on. Figure 11 shows this definition panel.

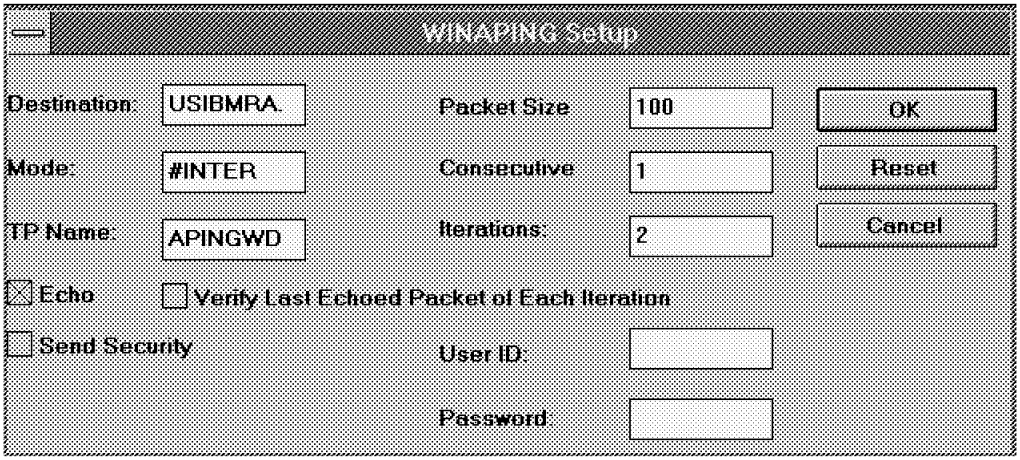


Figure 11. APING for WIN-OS/2 Parameters Panel

Then you have to save the definitions and run the program by clicking on in the green button that appears at the left upper corner.

Figure 12 on page 20 shows the output of the program.

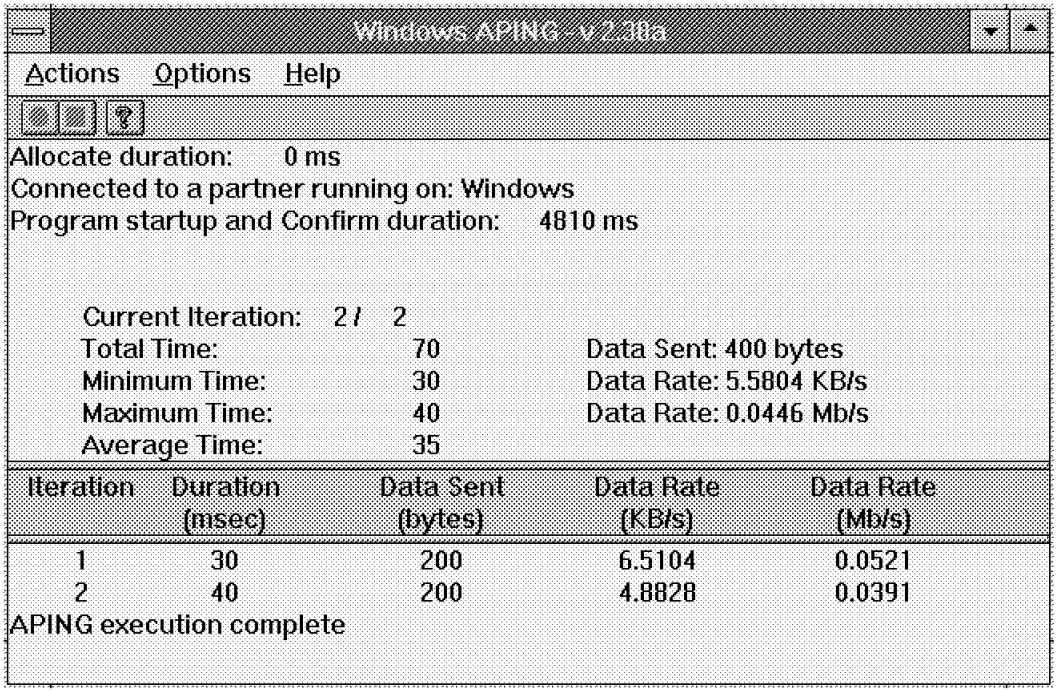


Figure 12. APING for WIN-OS/2 Output and Invocation Panel

As you can see there is a Connected to a partner running on: field. In this case we are defining the APINGD TPNAME, so we connect to a Windows partner running in WIN-OS/2.

The APINGD program for WIN-OS/2 is shown in Figure 13.

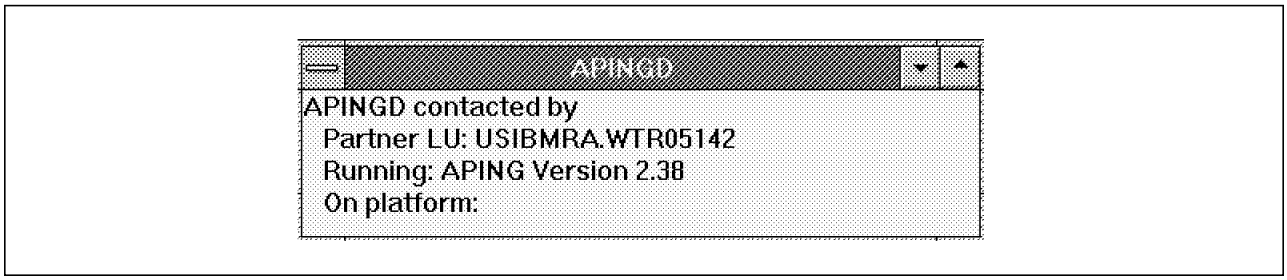


Figure 13. APING for WIN-OS/2 Execution

2.7 Problem Determination

Error messages generated by WIN-OS/2 are not handled by FFST/2. Instead, these messages are logged in a separate message log: CMWIN.LOG.

All of the WIN-OS/2 CPI-C errors are logged here. This file resides in the CMLIB directory by default. In general, it resides in the same directory as NSDW.DLL.

You can use any text editor to read the CMWIN.LOG file. The file contains:

- Time and date the message was logged
- Originator: WINCPI-C
- Probe ID 0E2xxyyy

  - 0E2 is the component ID, always the same.
  - xx corresponds to the module ID.

    You could find the module probe IDs in the CMWPRBID.H file.
  - yyy is the line number (in hexadecimal) of the failing module.
- Text of the message with a message number

Two new messages are added, which are documented in the Message Reference or could be displayed using HELP msg-number OS/2 command:

- CMR0376

  CMR0376: The WIN-OS/2 Common Programming Interface for Communications (CPI-C) support received an unexpected return code %1 from an internal function call.
- CMR0377

  CMR0377: The WIN-OS/2 Common Programming Interface for Communications (CPI-C) communications subsystem could not be loaded.

  This could be because the function has not been installed or you do not have the DEVICE= statement in your CONFIG.SYS for WIN-OS/2 CPI-C or the PATH= statement in the AUTOEXEC.BAT does not point to the CMLIB\DLL.

Each message is appended to the end of the file when it is received. Because this file is not truncated automatically, you might want to delete it occasionally to prevent it from growing too large.

Trace information for WIN-OS/2 can be gathered using the Communications Server trace facility when you select the APPC API and the appropriate DLCs and events.

You can see the WIN-OS/2 entries in either the raw trace file or the detailed trace file. The entries include a WinOS2 CPI-C text in it.

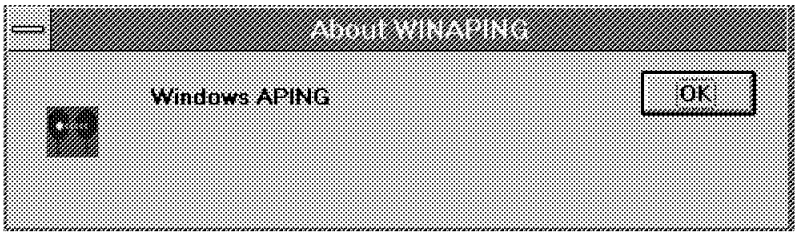


Figure 14. About APING for WIN-OS/2

Part 3. Networking

Chapter 3. APPN Enhancements

In this chapter, we present an overview of the new enhancements included in the IBM Communications Server Release 4.1. They are the backup link capability for independent LUs and the new configuration option to allow a connection network link to be defined as a non-limited resource.

3.1 Backup Link

In addition to the host backup link implemented in Communications Server Version 4.0, the new Communications Server now provides the ability to configure an APPN backup link.

A backup link is a link that Communications Servers will attempt to activate if activation of the *primary link* fails. Once the backup link has been activated, Communications Servers will attempt to restore normal configuration over the primary link and deactivate the backup link without disrupting traffic. A primary link can have only one backup link. Also a backup link never can be defined as the preferred link to a network node server. The backup link is defined as a limited resource and has a connect cost of 255 (maximum).

Communication Server will attempt to activate the backup link automatically only if the primary link fails to activate. If the primary link is active and then fails, Communications Manager will try to recover it first. If the recovery attempt fails, Communication Server will attempt to activate the backup link. If the primary link is deactivated by the user, by an inactivity timeout, or because it is a *limited resource* link, neither the primary link nor the backup link will be activated by Communications Server backup link support.

There are a number of considerations that affect how backup and recovery operate, such as the presence of dependent LU sessions, host focal point support, High-Performance Routing (HPR) traffic, and the different ways to configure reactivation of the primary link.

You can define two types of links with Communications Server:

- Links that carry SSCP traffic (dependent LUs or host focal point support) and may optionally carry APPN traffic.

  These links are referred to as *host links* in this discussion. A primary host link cannot be active at the same time as its backup link. The backup link must be deactivated before the primary link can be restored.

- Links that do not carry SSCP traffic and carry only APPN traffic.

  These links are referred to as *APPN-only links* in this discussion. A primary APPN-only link can be active at the same time as its backup link. The backup link can be deactivated after the primary link has been recovered and traffic has moved to it.

  Note

  Actually the backup link is deactivated when it has only CP-CP sessions, DLUR pipe and HPR connections that are recovered to the primary link since the backup link was quiesced. Also, quiescing the backup link causes all new sessions to be routed over the primary link.

Communications Server can automatically recover the primary link using any of the following techniques. It will defer recovery of primary host links, but not APPN-only primary links, until the backup link is inactive:

- The primary link may be recovered because automatic reactivation is specified. With this approach, the primary link is recovered as soon as possible.
- The primary link may be recovered because a dependent LU application requests sessions (SNA Gateway or LUA) and the backup link is inactive. This approach only works when there are dependent LUs on the link. The primary link is not activated until it is needed for dependent LU sessions. If you are using APPN support as well as dependent LU support, consider using the automatic reactivation technique to ensure the link is up even when there are no dependent LU applications running.
- The primary link may be reactivated because it is a permanent connection. This approach is similar to automatic reactivation; the primary link is recovered as soon as possible.
- The primary link may be reactivated because it is a preferred network node server link. This approach is only useful if your Communications Server node is an APPN end node (EN). With this approach you define only two links to network nodes: the primary and backup links. Communications Server will attempt to activate the primary link only if it loses network node server support; this will occur when the backup link goes down.

If you configure the primary and backup links using the CMSETUP windows, CMSETUP will choose a recovery technique for you. To override this choice, you must update the response file.

Figure 15 on page 28 shows a rough scheme of the APPN backup link function.

3.2 Functionality of APPN Backup Link

Backup link definition is the answer to load sharing between network nodes while removing the network node as the single point of failure. Half of the EN node workstations have NN-1 as the primary link and NN-2 as the backup link while the other half have NN-2 as the primary and NN-1 as the backup link. If the link to the NN-1 node fails, then the EN nodes with NN-1 as the primary first attempt to reactivate it and if that reactivation fails, then the link to NN-2 is activated. The primary link has auto_react(infinite_retry) so reactivation of the primary is also attempted. CP-CP sessions are activated and APPN traffic continues to flow through NN-2. Without backup link definition, even after the NN-1 link is reactivated, the CP-CP sessions continue with NN-2. However, with the backup link, new sessions will use the preferred link and the backup link is deactivated when the remaining sessions can be nondisruptively rerouted to the primary link. New sessions use the primary link since it has lower cost per connect. (Backup links use 255 if the use_adapter default was specified.) When the primary link is activated while the backup is active, the backup link is quiesced to avoid further usage for new sessions (refer to Figure 15 on page 28).

The following sessions can be nondisruptively rerouted:

- CP-CP sessions
- APPC SNASVCMG sessions

- CPSVRMGR DLUR pipe sessions
- HPR sessions

The more obvious reason for a backup link is a *leased* link (or switched line) with a switched WAN backup (dialed SNA phone connection, dialed ISDN, ATM, frame relay, or X.25 link). In this case, there is an additional charge for the second link. The backup and recovery function comes to the rescue again. If the primary link fails, then the node attempts to reactivate it and if that reactivation fails, then the backup link is activated. CP-CP sessions are activated and APPN traffic continues to flow through the backup link. Without backup link definition, even after the primary link is reactivated, the CP-CP sessions continue over the backup link and operator action is required to switch back to the primary link. However, with backup link, new sessions will use the primary link and the backup link is deactivated when the remaining sessions can be nondisruptively rerouted to the primary.

This saves line costs by activating the backup when it is required and deactivating it when it is no longer required.

Auto reactivation of a link may also be used without a backup link. It causes the link to automatically be reactivated. The time between reactivation attempts is increased by 30 seconds for each attempt until the time between reactivation attempts reaches 180 seconds. If there was also an adapter failure, the link retries continue at 30 seconds until the adapter is activated. The adapter reactivation is the same as the above described link activation.

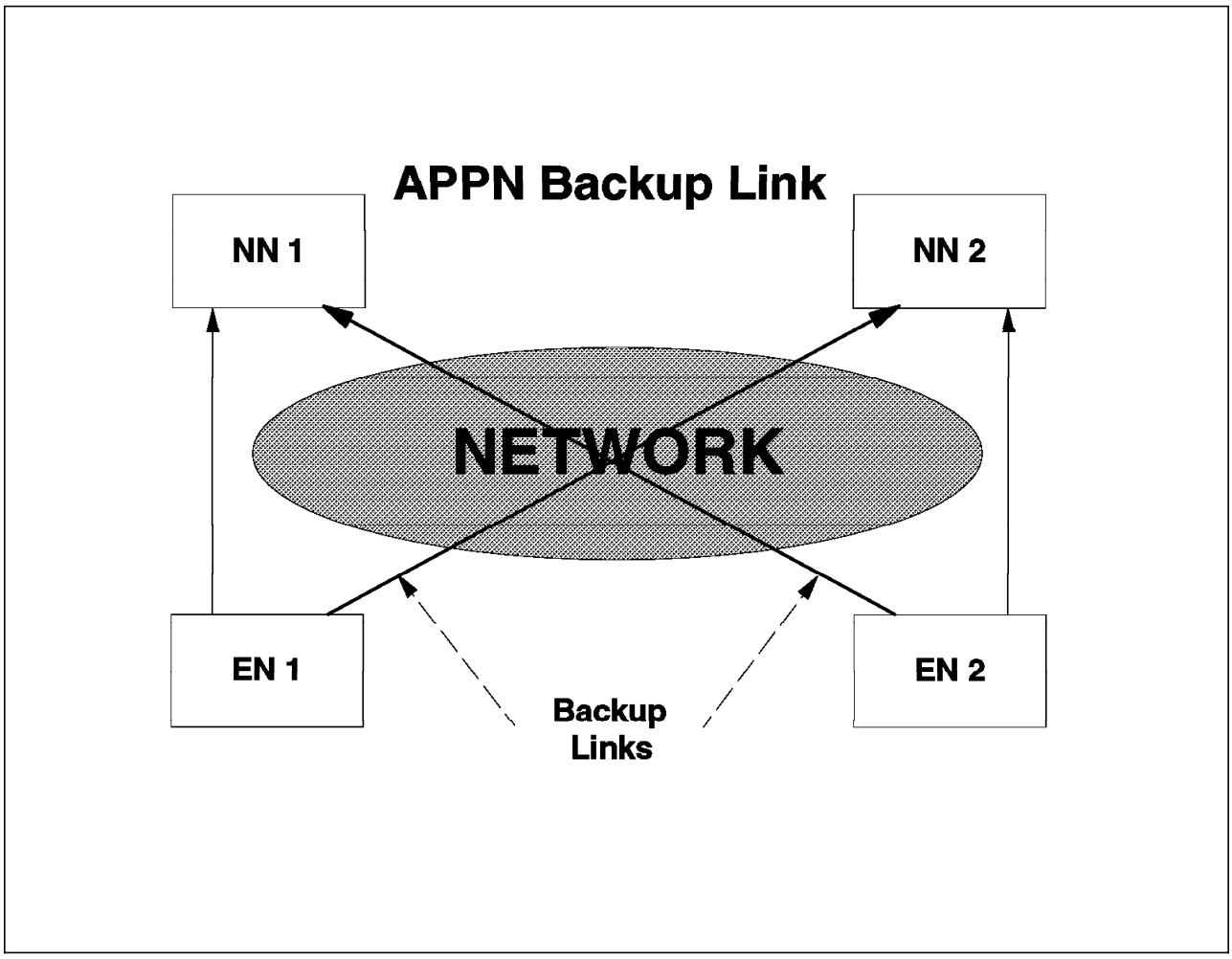


Figure 15. APPN Backup Link Overview

It is desirable to use an APPN backup link in conjunction with HPR support. However, in IBM Communications Server Release 4.1, HPR is not supported over SDLC nor X.25.

The only disadvantage when using an APPN backup link without HPR is that sessions will be terminated when there are no in-flight conversations. That is, when conversations have ended. Therefore, if you are running long conversations (that is, file transfer) the switch to and from the backup link is disruptive in this case.

3.3 Using High-Performance Routing with Backup Link Support

There are some special considerations if you are using HPR and want sessions moved to the backup without path switch failures. You need to make sure that transitions between the primary and backup link occur quickly and that the backup link supports HPR.

Communications Server must detect a primary link failure and also fail a reactivation attempt before it attempts to activate the backup link. In a typical LAN environment this may take longer than the default HPR path switch time. You can shorten this time by reducing the link establishment retransmission count to 2 or even 1 (from a default of 8) and the retransmission threshold to 3

(from a default of 8). You specify these parameters on the CMSETUP windows for LAN or Ethernet DLCs. If you are using response files, the keywords are LINK_ESTABLISHMENT_RETRANSMISSION and RETRANSMISSION_THRESHOLD in the LAN_DLC record. Some additional improvement can be obtained by changing the Group 2 inactivity time (Ti) to 127 (from the default of 255) under MPTS for the IBM IEEE 802.2 protocol. Note that although these changes speed up the response to failures, they may also increase the chance that transient conditions are treated as failures.

If you are still getting path switch failures, you can increase the path switch timeout values. This change must be made at all HPR route endpoints, not just the node with the primary and backup links. For Communications Server, you must make this configuration change using response files; specify PATH_SWITCH_TIMER_LOW, PATH_SWITCH_TIMER_MEDIUM, and PATH_SWITCH_TIMER_HIGH in the SNA_DEFAULTS record.

Unlike other limited resource links, HPR traffic will not keep a backup link up. If the primary link is active and the only user session traffic is being carried by HPR, Communications Server will deactivate the backup link and the HPR traffic will be switched to the primary link.

You should make sure that both links support HPR if possible. If one link supports HPR and the other doesn't, you may get path switch failures after either the primary or the backup link goes down.

3.3.1 Other Considerations

Consider using dependent LU requester (DLUR) support instead of dependent LUs. You may use APPN-only links with DLUR support. This will allow you to recover your primary link while your backup is active. DLUR can also be used in conjunction with HPR. During primary link failure and recovery, DLUR sessions over HPR will be moved from the primary link to the backup link and back again. The return to normal network operation from a failure will be smoother and quicker with APPN-only links than with host links.

You may have to adjust link parameters so that the partner node for the primary link discovers a failure in approximately the same amount of time as Communications Server does. This is especially important if the partner node is the same for both the primary and the backup links. If the partner node does not detect a primary link failure and does not clean up associated sessions before the backup link is activated, there may be additional failures as Communications Server and application programs attempt to recover lost sessions across the backup link.

Backup links are supported for GDLC connections only if the adapter notifies Communications Server when the link goes down. Check with your adapter manufacturer to determine if Communications Server backup is possible.

3.4 Configuration and Installation

No extra modules need to be installed to implement the APPN backup link.

3.4.1 Defining APPN Backup Link Support Using CMSETUP

The following is an example of how to configure an APPN backup link.

Note

If you configure the primary and backup links using the CMSETUP windows, CMSETUP will choose a recovery technique for you. For example, auto reactivate (infinite retry) for the primary link. To override this choice, you must update the response file.

Two network nodes are connected via token-ring. The APPN backup link (secondary link), as shown in Figure 16 on page 30, is provided by a frame relay link.

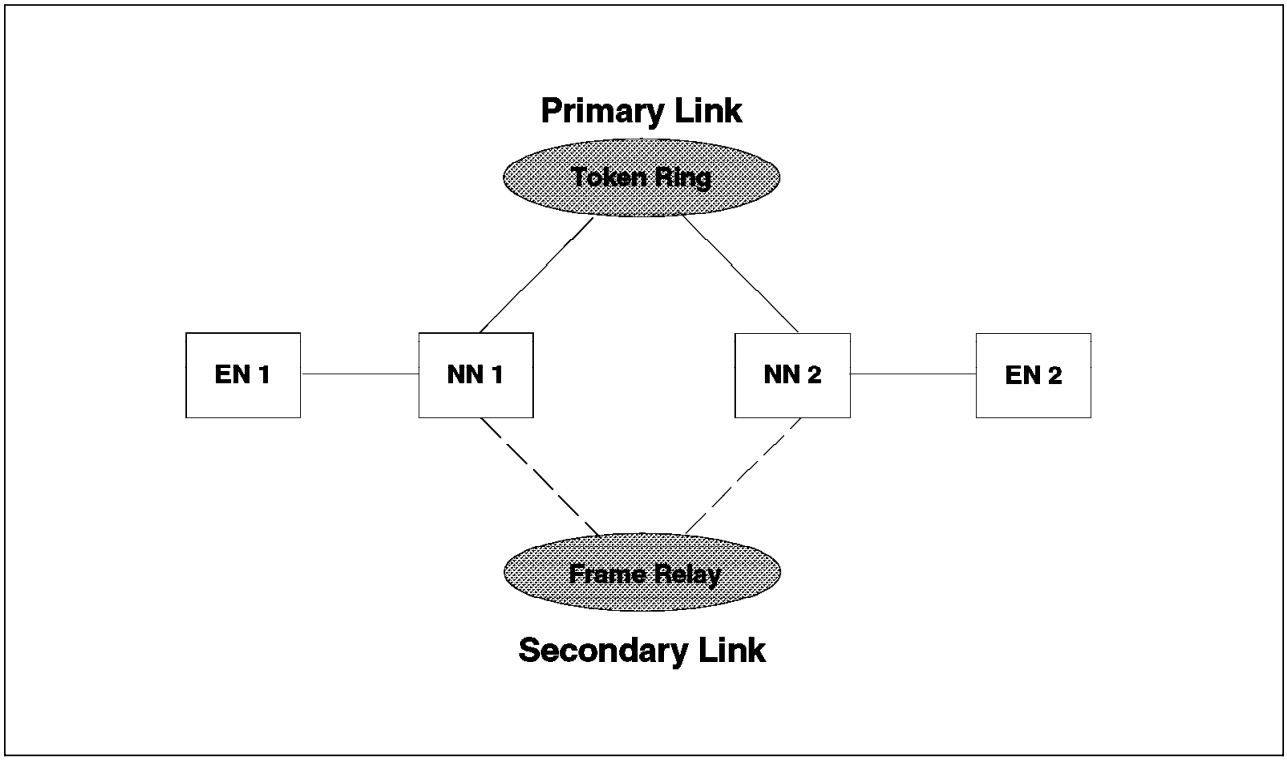


Figure 16. APPN Backup Link Example

1. Use the default parameter whenever possible. Configure the adapter for the type of DLC you want to use for the primary connection.

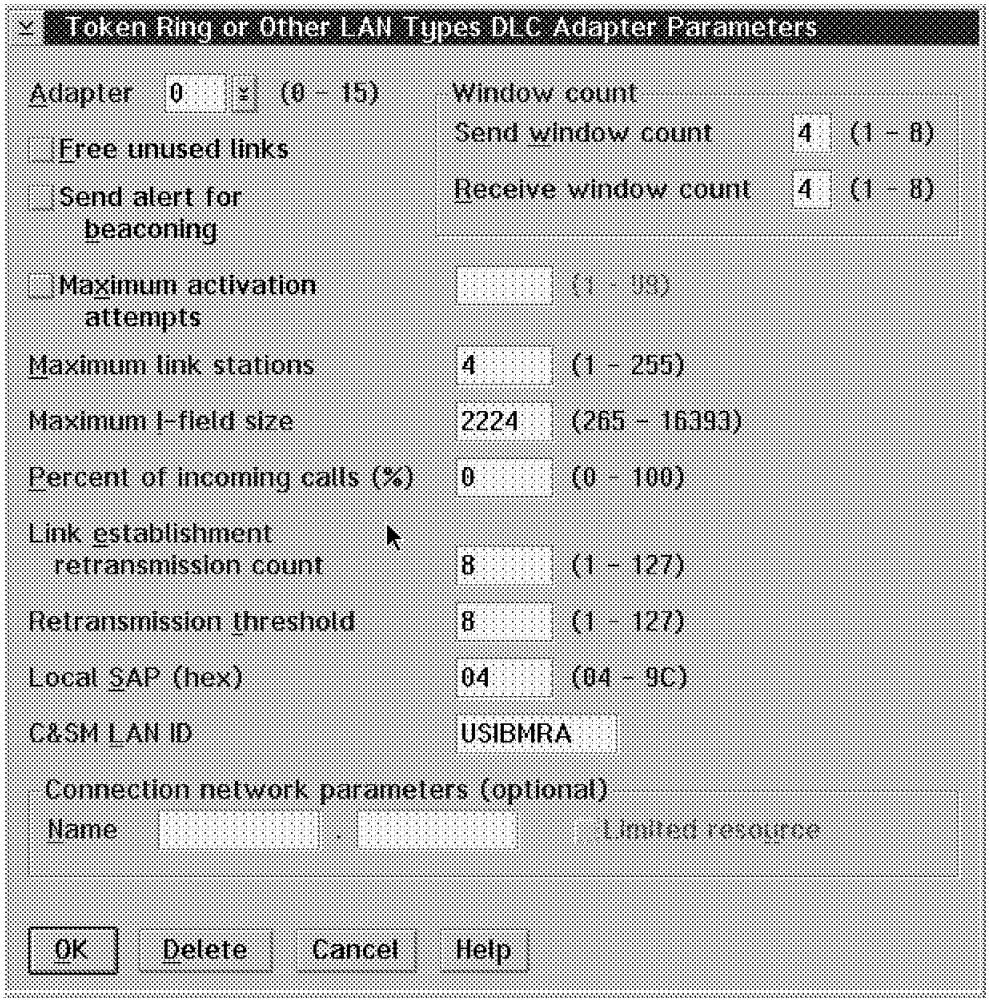


Figure 17. DLC Adapter Parameters - Primary Adapter in Node NN-1

After configuring the adapter, leave this panel and save your definitions by clicking on **OK** before you attempt to configure the second adapter.

Communications Server will not recognize two adapters configured in one attempt in this panel.

2. Use the default parameter if possible. Configure adapter 2 for the type of DLC you want to use for the backup connection.

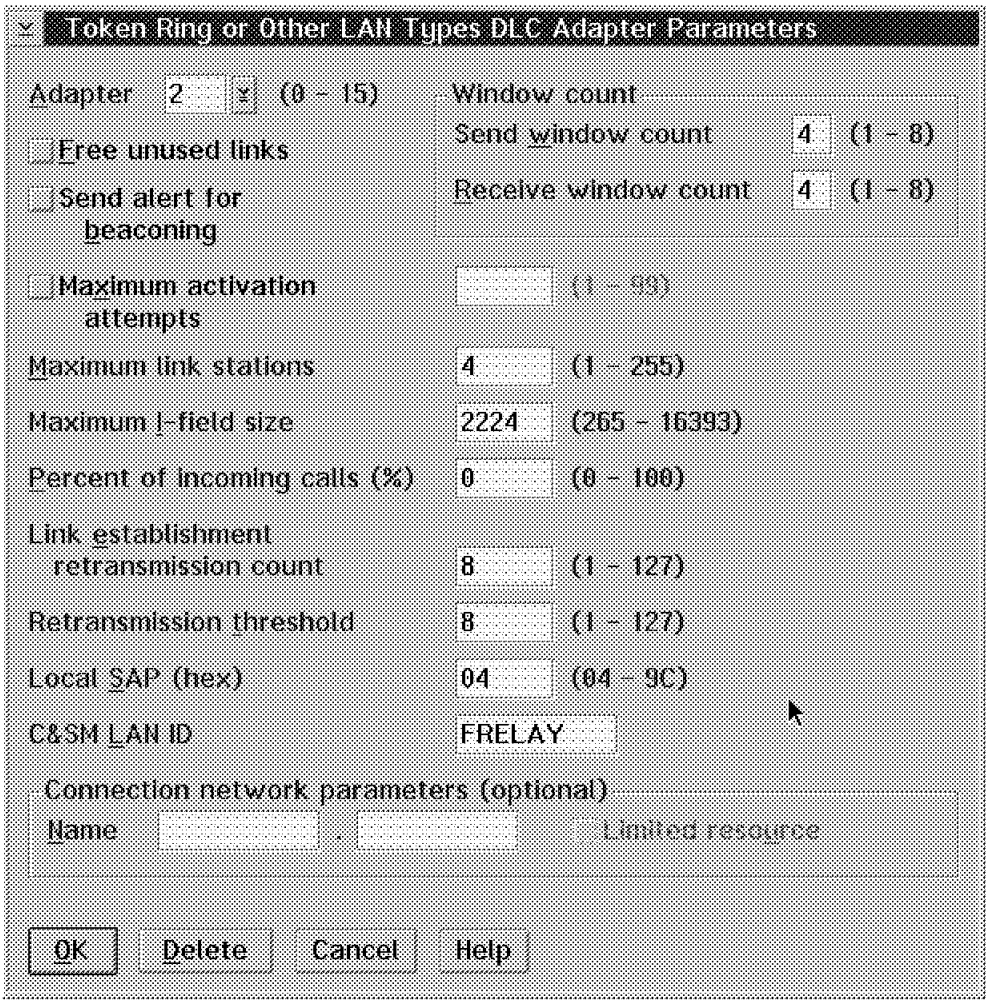


Figure 18. DLC Adapter Parameters - Secondary Adapter for Backup Link

3. Configure your local node characteristics by selecting **Network node**.

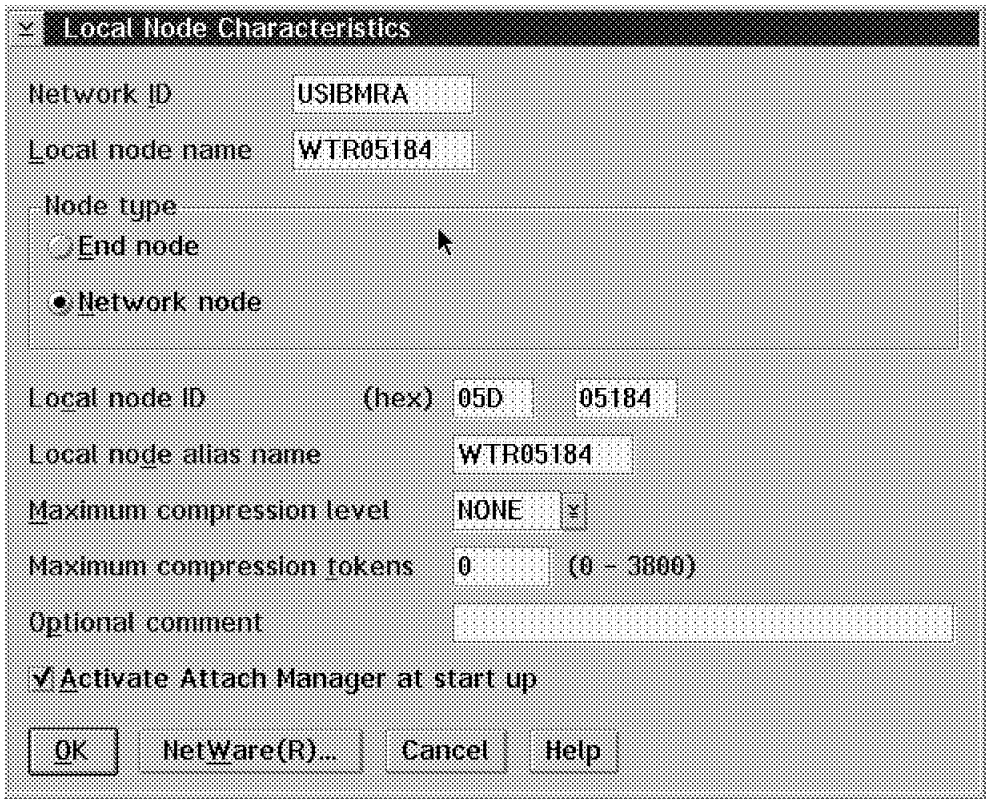


Figure 19. Local Node Characteristics

4. Configure the primary connection by selecting partner network node.

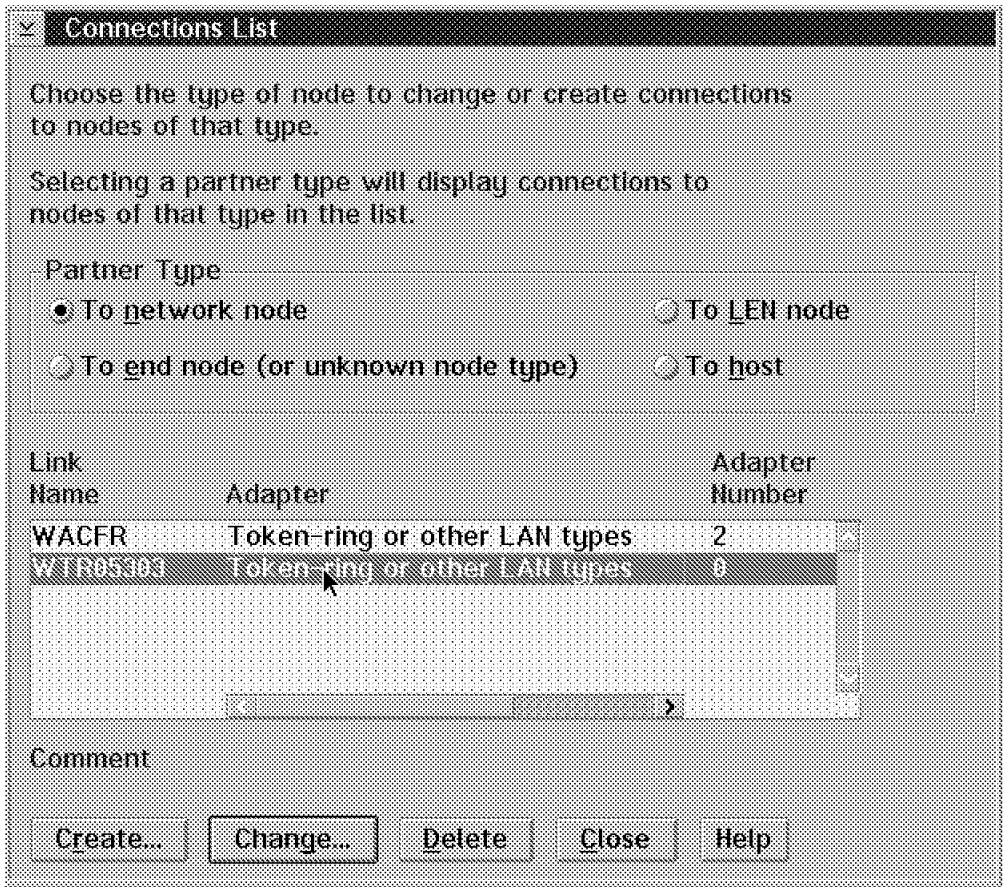


Figure 20. Connection List WTR05303 (Primary) and WACFR (Backup)

5. Select the adapter type for the primary connection link.

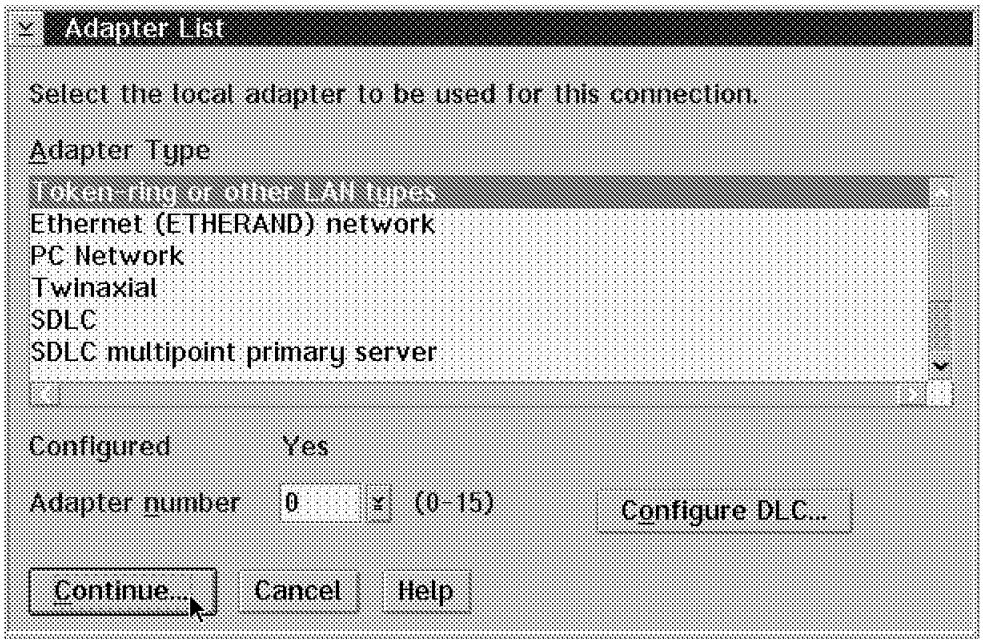


Figure 21. Adapter List

6. Define the link name for the primary link, the partner LU definitions and the LAN destination address. Click on **Additional parameters** to configure the additional connection parameters.

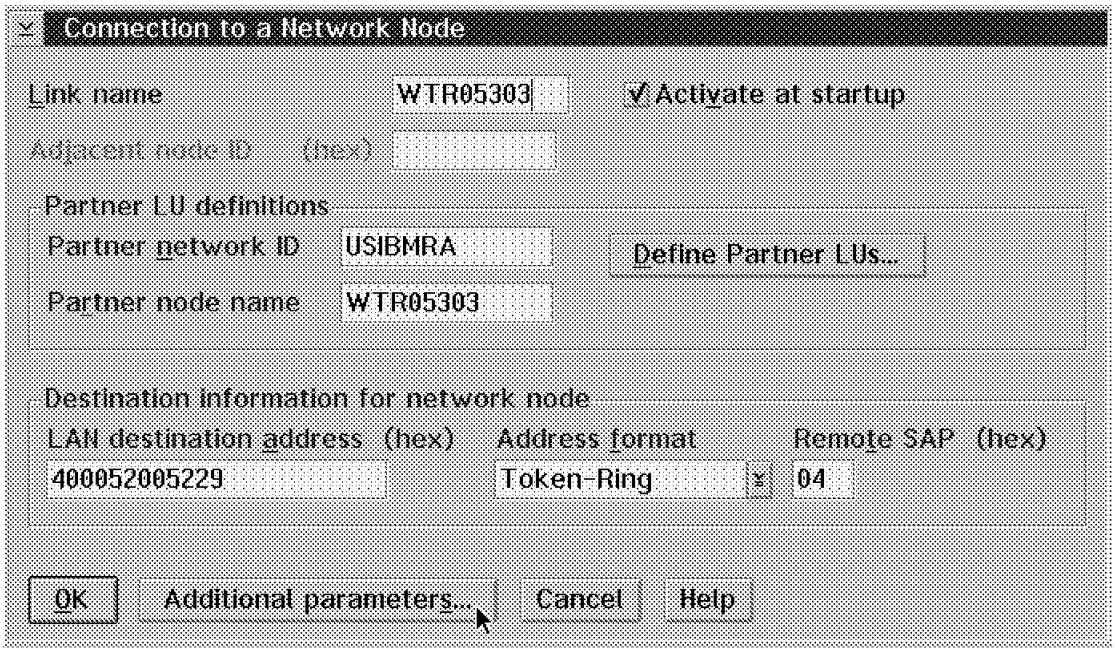


Figure 22. Connection to a Network Node

7. Click on **HPR Support**. Select **OK** and go back to the Connection List to configure for the backup link.

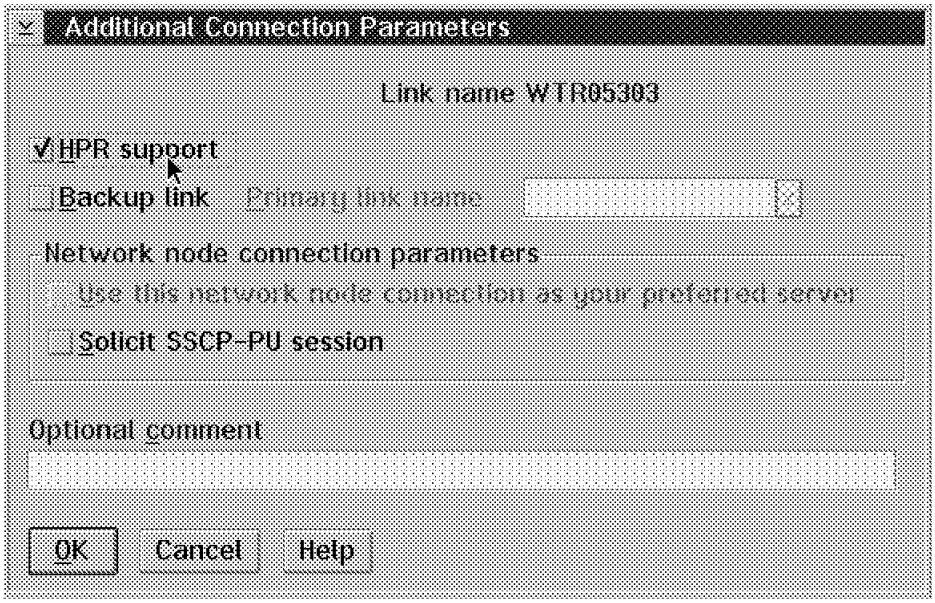


Figure 23. Additional Connection Parameters

8. Create a second link as a backup link to the partner network node by selecting **Create**.

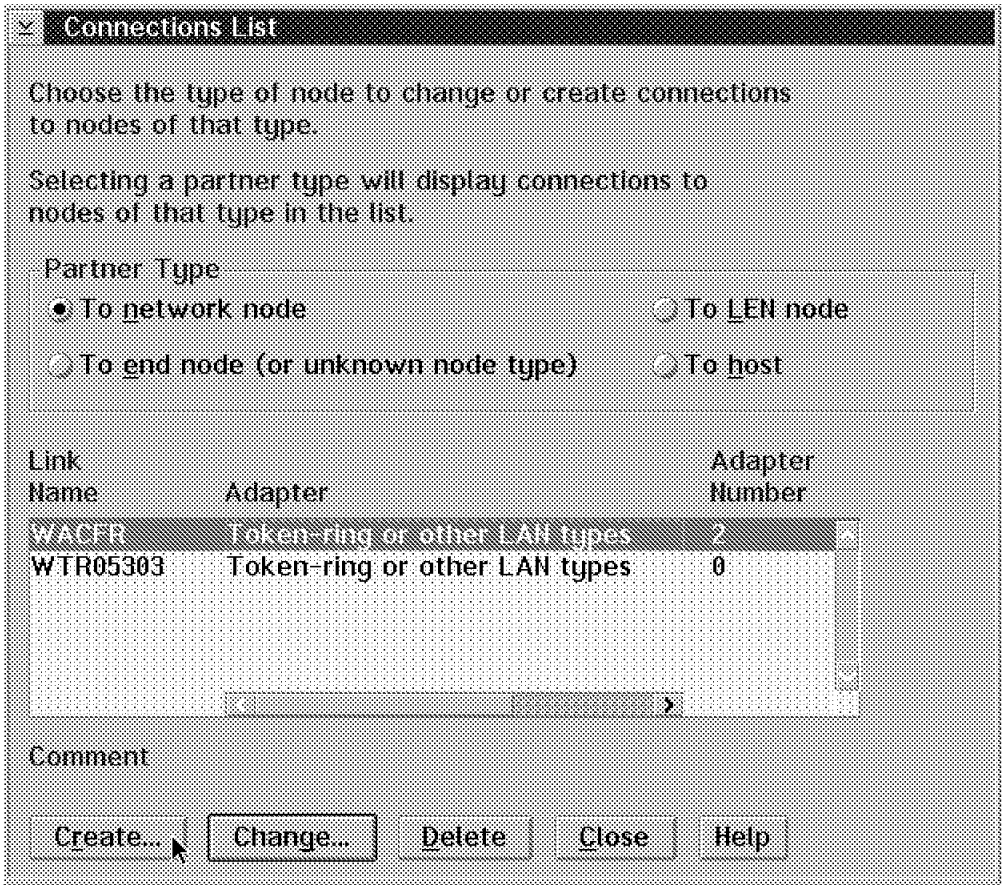


Figure 24. Connection List

9. Configure the Adapter Type used for the backup link. Choose the related Adapter Number from the Adapter number drop down list box.

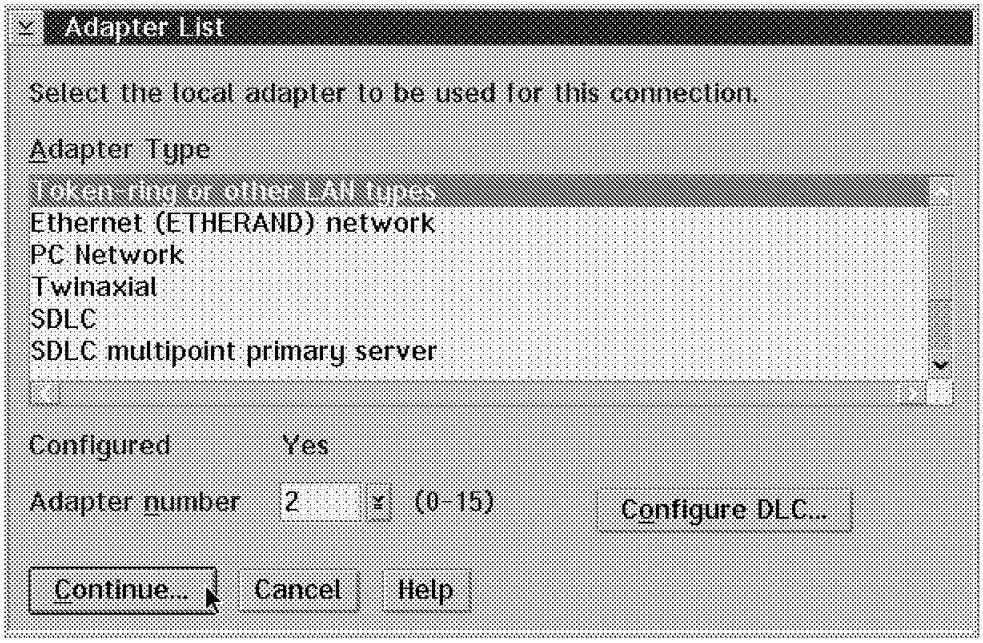


Figure 25. Adapter List

10. Define the link name for the backup link. The Partner LU definitions must be the same as defined for the previous primary link. The LAN destination address is the address of the second adapter in the partner network node. Click on **Additional parameters** to configure additional connection parameters.

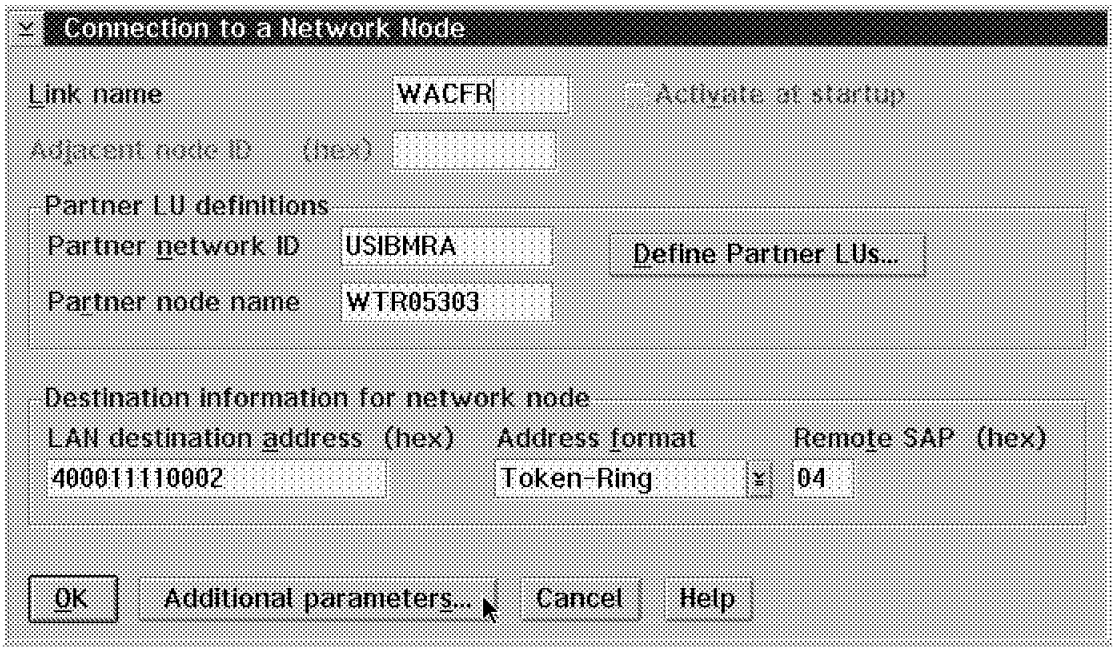


Figure 26. Connection to a Network Node

11. Click on **HPR support** and **Backup link**. Choose the Primary link name from the drop down list box. Click on **OK**.

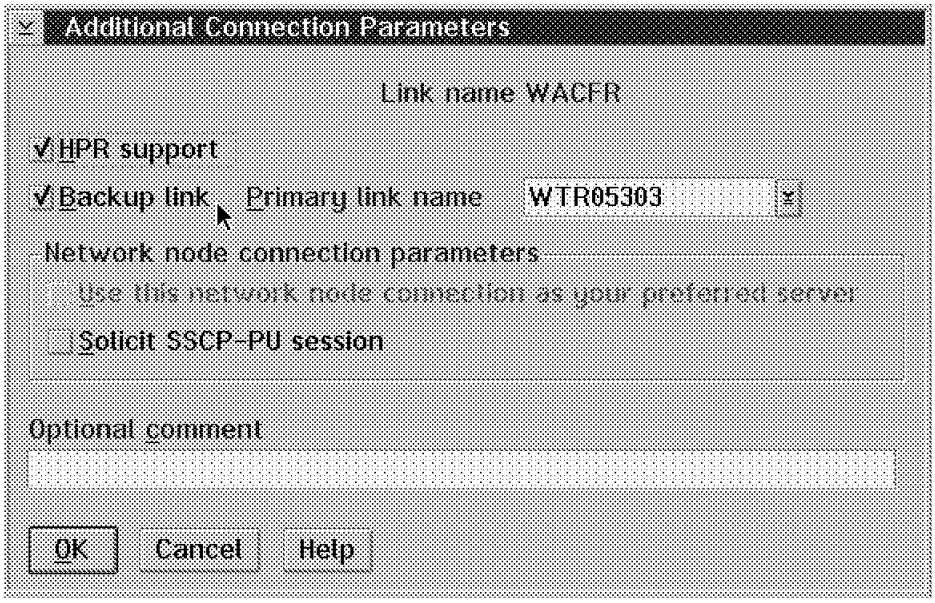


Figure 27. Additional Connection Parameters

12. Leave CM-Setup and verify the configuration.

3.4.2 Configuring APPN Backup Link Support Using Response Files

For the primary link specify the following response file keywords:

- SOLICIT_SSCP_SESSION = 0.
- HOST_BACKUP_LINK = 0.
- AUTO_REACTIVATE = -1 and LIMITED_RESOURCE = 0 if you want the primary link to automatically reactivate as soon as possible. Also specify PERMANENT_CONNECTION_NAME for permanent connections.
- PREFERRED_NN_SERVER = 1, AUTO_REACTIVATE = 0, and LIMITED_RESOURCE = 0 if you want the primary link to be reactivated by preferred network node server support (for end nodes only).
- Be sure *not* to specify the PU_NAME parameter.

For the backup link set the following response file keywords:

- SOLICIT_SSCP_SESSION = 0.
- HOST_BACKUP_LINK = 1.
- Set PRIMARY_LINK_NAME to the name of the primary link.
- AUTO_REACTIVATE = 0.
- LIMITED_RESOURCE = 1.
- ACTIVATE_AT_STARTUP = 0.
- PREFERRED_NN_SERVER = 0.
- PERMANENT_CONNECTION_NAME should not be specified for a backup link.
- Do not specify the PU_NAME parameter.

- The default value of COST_PER_CONNECT_TIME is 255 for backup links. This is higher than any primary link. This will make the backup link a less desirable route than the primary link for most networks.

3.4.3 Example of .NDF File

Figure 28 on page 40 shows parts of the .NDF file of the configuration. Note that the LINK_NAME in both of the two DEFINE_LOGICAL_LINK statements, as well as the DESTINATION_ADDRESS statements are different (HPR_SUPPORT(YES)).

Both links are configured to support HPR (HPR_SUPPORT(YES)), but only WACFR has the statement HOST_BACKUP_LINK(YES) included which marks this link as the backup link for the primary link TRLINK.

There are some define_logical_link parameters that are necessary for the backup and recovery operation:

- HOST_BACKUP_LINK is NO on the primary and YES on the backup link. Note that since a link can have only one INFINITE_RETRY, this means that reactivation of the link occurs at the following intervals (30 seconds, 60 seconds, 90 seconds, 120 seconds, 150 seconds, 180 seconds and continues every 180 seconds until it is activated).
- ACTIVATE_AT_STARTUP must be NO for the backup link and it is typically YES for the primary link, but this is not required.
- LIMITED_RESOURCE is YES for the backup and NO for the primary link. This causes unused sessions to be deactivated if they are unused for the configured limited resource timeout after a conversion ends (typically 10 seconds).
- COST_PER_CONNECT_TIME is USE_ADAPTER_DEFINITION for both links, but the defined logical link processing code sets the cost per connect to 255 (maximum value) on the backup link if use_adapter _definition is set. This means that the primary link will be used for new sessions if it is active. If the primary link is activate_at_startup, then it is not in the topology data base unless it is active.

```
          DEFINE_LOGICAL_LINK  LINK_NAME(TRLINK  )
                FQ_ADJACENT_CP_NAME(USIBMRA.WTR05303 )
                ADJACENT_NODE_TYPE(NN)
                PREFERRED_NN_SERVER(NO)
                DLC_NAME(IBMTRNET)
                ADAPTER_NUMBER(0)
                DESTINATION_ADDRESS(X'40005200522904')
                ETHERNET_FORMAT(NO)
                CP_CP_SESSION_SUPPORT(YES)
                SOLICIT_SSCP_SESSION(NO)
                USE_PUNAME_AS_CPNAME(NO)
                MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                AUTO_REACTIVATE(INFINITE_RETRY)
                ACTIVATE_AT_STARTUP(YES)
                LIMITED_RESOURCE(NO)
                LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                SECURITY(USE_ADAPTER_DEFINITION)
                PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                HPR_SUPPORT(YES)
                LIMITED_RESOURCE_TIMEOUT(USE_ADAPTER_DEFINITION)
                                  .
                                  .
                                  .

     DEFINE_LOGICAL_LINK  LINK_NAME(WACFR   )
                FQ_ADJACENT_CP_NAME(USIBMRA.WTR05303 )
                ADJACENT_NODE_TYPE(NN)
                PREFERRED_NN_SERVER(NO)
                DLC_NAME(IBMTRNET)
                ADAPTER_NUMBER(2)
                DESTINATION_ADDRESS(X'40001111000204')
                ETHERNET_FORMAT(NO)
                CP_CP_SESSION_SUPPORT(YES)
                SOLICIT_SSCP_SESSION(NO)
                USE_PUNAME_AS_CPNAME(NO)
                MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                HOST_BACKUP_LINK(YES)
                PRIMARY_LINK_NAME(TRLINK)
                LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                AUTO_REACTIVATE(NO_RETRY)
                ACTIVATE_AT_STARTUP(NO)
                LIMITED_RESOURCE(YES)
                LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                SECURITY(USE_ADAPTER_DEFINITION)
                PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                HPR_SUPPORT(YES)
                LIMITED_RESOURCE_TIMEOUT(USE_ADAPTER_DEFINITION)
                                  .
                                  .
```

Figure 28. Link Definitions in .NDF File of Backup Link Configuration

3.5 Non-Limited Resource for a Connection Network

Connection networks (CN) allow APPN nodes in a LAN to have direct links with each other without requiring logical definitions at each node. This greatly reduces system definitions by facilitating response files that are very similar for each end node. It also allows new nodes that are added to the LAN to fully participate in APPC conversations without requiring definition changes at the node.

The network nodes (NN) defines all the nodes in a CN to have links directly to each other. The NN calculates the direct route and simply sends the end nodes the address of the partner to use for activating the link. This reduces the performance burden on the NN by avoiding having to route sessions through the NN (see Figure 29 on page 41).

Limited resource CNs disconnect after a conversation ends while non-limited resource CNs do not. This is designed for sessions that will be frequently used and therefore should be kept up. Keeping them up removes the performance overhead of constantly bringing them up and down.

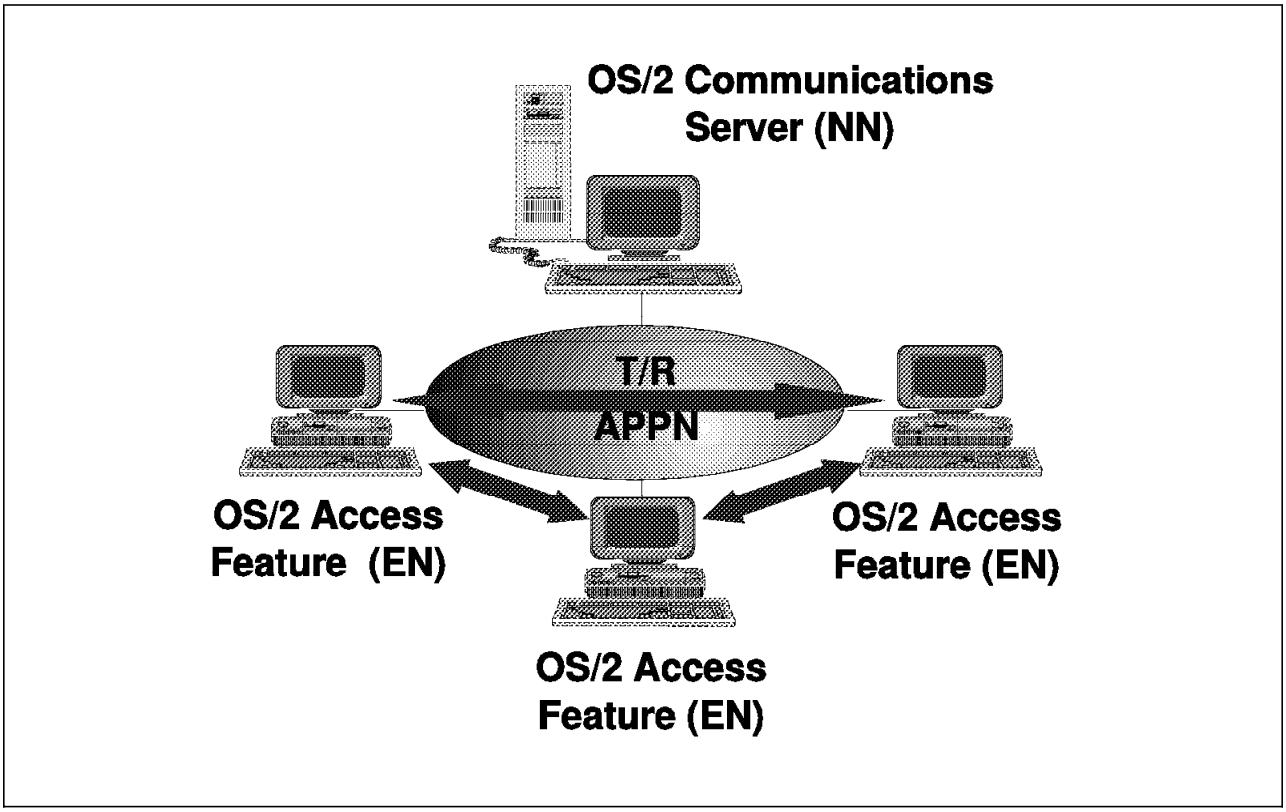


Figure 29. Connection Network, Example

- Connection networks reduce system definition:
  - No logical link definitions are required at each node.
  - New nodes do not require definition changes at each node in order to participate in the network.
  - End nodes have direct links to each other and reduce performance burden on the network nodes and the LAN.
- Connection networks support bridged LANs:

  - APPN views bridge LANs as a single logical network.
  - One connection network supports the entire network.
- Non-limited connection networks enable sessions to remain active.
- This support is designed for sessions that need to remain up for longer periods.

3.5.1 Configuration

To configure for a connection network, simply bring up the Token Ring or Other LAN Types DLC Adapter Parameter window in your configuration.

In the lower portion of the panel you will find a section named Connection network parameter. Enter the name of the connection network and click on the **Limited resource** push button in order to define a limited resource.

Note

Connection networks are identified by a fully qualified name. The connection network name contains two parts:

- Network ID
- Name identifying the LAN to which you are connected

For configuration help, click on the **Help** push button in this panel.

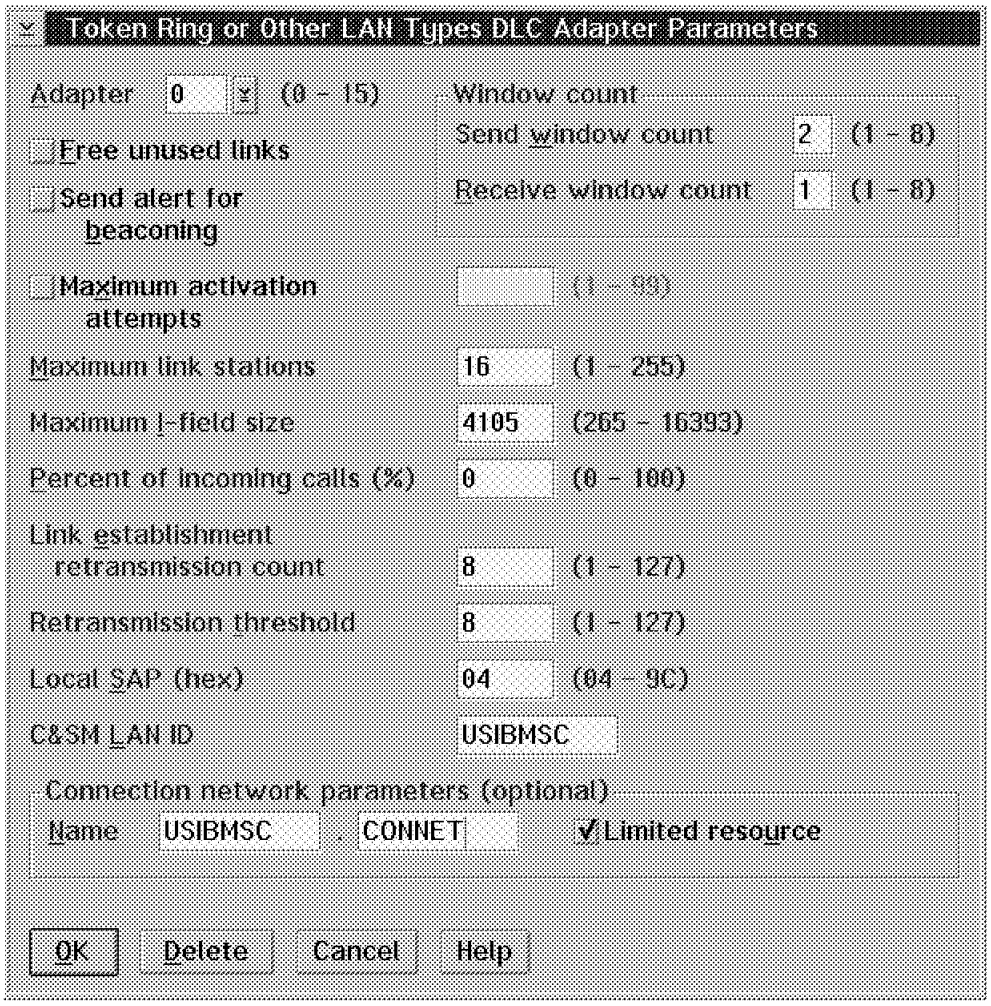


Figure 30. Connection Network Configuration

3.5.2 NDF File Entries for Connection Networks

Configuring the connection network part causes a new section to be entered into the .NDF file after the configuration was verified.

Note the new entry in the example of an .NDF file in Figure 31:

```
DEFINE_LOCAL_CP  FQ_CPNAME(USIBMSC.WTRO5137)
                 CP_ALIAS(WTRO5137)
                     .
                     .
                 NAU_ADDRESS(INDEPENDENT_LU)
                 NODE_TYPE(EN)
                     .
                     .
DEFINE_CONNECTION_NETWORK  FQ_CN_NAME(USIBMSC.CONNET   )
                           LIMITED_RESOURCE(YES)     <============
                           ADAPTER_INFO( DLC_NAME(IBMTRNET)
                                         ADAPTER_NUMBER(0));
```

Figure 31. NDF File Example of Limited Resource CN

Part 4. AnyNet Multiprotocol Support

Chapter 4. Sockets over SNA Access Node and Gateway

This function allows Sockets applications such as PING, FTP, Telnet, Web browsers and many others to run over SNA networks. The transport is an LU 6.2 session and the TCP/IP protocols are translated into APPC full-duplex conversations. Sockets over SNA enables OS/2-based application programs using the IBM TCP/IP socket interface to communicate with each other when connected over SNA networks.

In addition, applications that use Sockets over SNA on operating systems other than OS/2 can also communicate between them. When a Sockets over SNA gateway is attached to the local network, socket application programs can communicate even though the applications may reside on networks using different transport protocols.

The Sockets over SNA access node support is included in both the OS/2 Access Feature and IBM Communications Server Release 4.1. However, the Sockets over SNA gateway function is only available in the Communications Server.

In this chapter, we describe how to configure some scenarios with Sockets over SNA where applications can communicate in a native SNA network or in different networking environments. For practical purposes, we only describe the environments supported by the Sockets over SNA function of the IBM Communications Server Release 4.1: SNA and TCP/IP networks.

The following scenarios are described:

1. Sockets over SNA applications running on an SNA network (no TCP/IP)
2. Running an OS/2 Web browser over SNA
3. Connecting two TCP/IP networks via APPN

Note

Sockets over SNA nodes do not necessarily require TCP/IP but you need to have the IP protocol stack available from either TCP/IP or MPTS. IBM Communications Server Release 4.1 includes MPTS.

4.1 Overview

In this section we review the main function of Sockets over SNA and how it works. We also list the new enhancements included in this release.

4.1.1 What Does Sockets over SNA Do?

Sockets over SNA is one of IBM′s AnyNet software offerings. AnyNet software enables application programs to communicate over different transport networks and across interconnected networks. Using AnyNet can reduce the number of transport networks and reduce operational complexity. These benefits are gained without modification to your existing application programs or hardware.

The Sockets over SNA access node function of Communications Server enables C application programs using the IBM TCP/IP AF_INET socket interface to

communicate over SNA networks. The Sockets over SNA Gateway function enables socket applications in SNA and IP networks to communicate.

Also new functions have been incorporated in this new release of Communications Server:

- Backup and load balancing. Enables multiple or parallel gateways to service the same site, providing balancing and backup support.
- Datagram retry delay. Enables you limit the number of failed SNA allocates in your network.
- Route discovery. Helps you to reduce the number of explicit defined routes in your network.
- Routing Information Protocol (RIP). Updates routing table entries for nodes in IP networks when more direct routes are discovered.
- Maximum number of connections up to 2000.
- Support of variable subnetting.

4.1.2 How Does Sockets over SNA Works?

Figure 32 on page 49 shows the structure of an OS/2 node that is running Sockets over SNA and illustrates how socket application programs and Sockets over SNA operate on an OS/2 node.

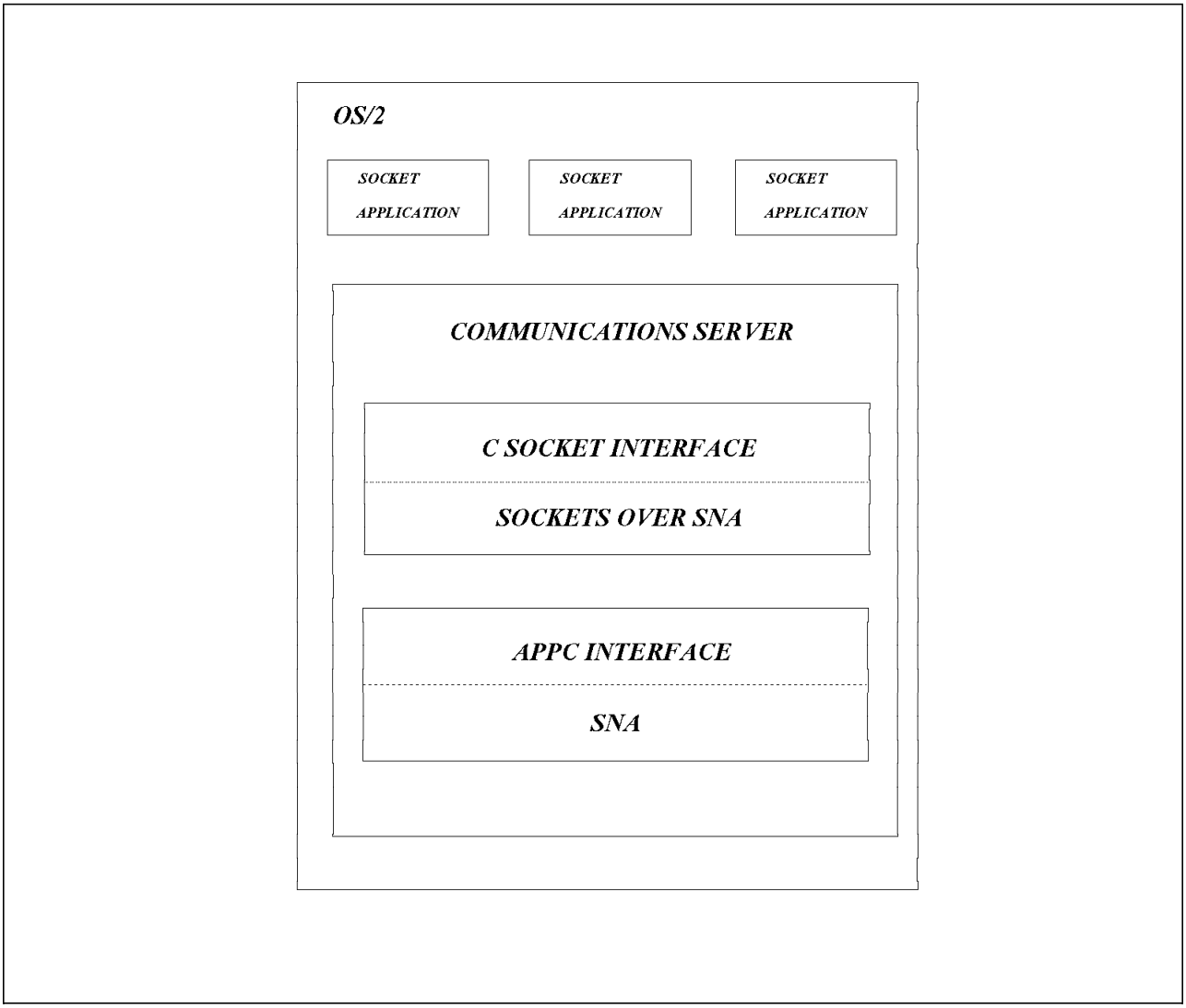


Figure 32. Structure of an OS/2 Node Running Sockets over SNA

Sockets over SNA uses the C socket interface that allows C socket applications to communicate with remote socket applications. When an application attempts to open a stream or datagram connection to another application using Sockets over SNA, it supplies the same information to Sockets over SNA that it would supply to TCP/IP.

Operating as an LU 6.2 application, Sockets over SNA receives socket calls through its application program interface and generates LU 6.2 calls to correspond to the socket calls. Sockets over SNA issues these calls to the LU 6.2 interface of Communications Server.

Note

Sockets over SNA does not support applications that use broadcasting.

4.2 Sockets over SNA Enhancements

In this section we explain the new enhancements included in IBM Communications Server Release 4.1:

- Backup and load balancing
- Datagram retry delay
- Route discovery
- Routing information protocol
- Maximum number of connections
- Variable subnetting

4.2.1 Backup and Load Balancing

The backup and load balancing option provided by Sockets over SNA can improve the reliability of your network by allowing multiple (or parallel) gateways to service the same site. This option provides a solution for large networks with multiple branches or remote sites that are serviced by a single central site.

The default for Sockets over SNA is that the Sockets over SNA Gateway is not a parallel gateway. Use the Sockets backup and load balancing configuration window to select this option and configure parallel gateways. In addition, you need to define routes to all remote subnets.

Using parallel gateways in a central site provides:

- Backup, so that connectivity is assured even if one of the gateways fails.
- Load balancing, so that traffic is routed even if a single gateway is not powerful or fast enough to handle its traffic at peak times.

Some points must be discussed:

- Remote gateways should define default routes through each of the parallel gateways.
- Parallel gateways use the routing information protocol (RIP) to monitor the status of partner parallel gateways. RIP enables parallel gateways to periodically broadcast the contents of their routing tables to partner parallel gateways when changes occur.
- In IBM Communications Server Release 4.1 backup only works when the SNA side of the gateway fails. The feeder gateway reroutes based on SNA allocate failures.

The following example shows a network configuration in which Sockets over SNA gateways connect central and remote sites on IP networks across an SNA backbone.

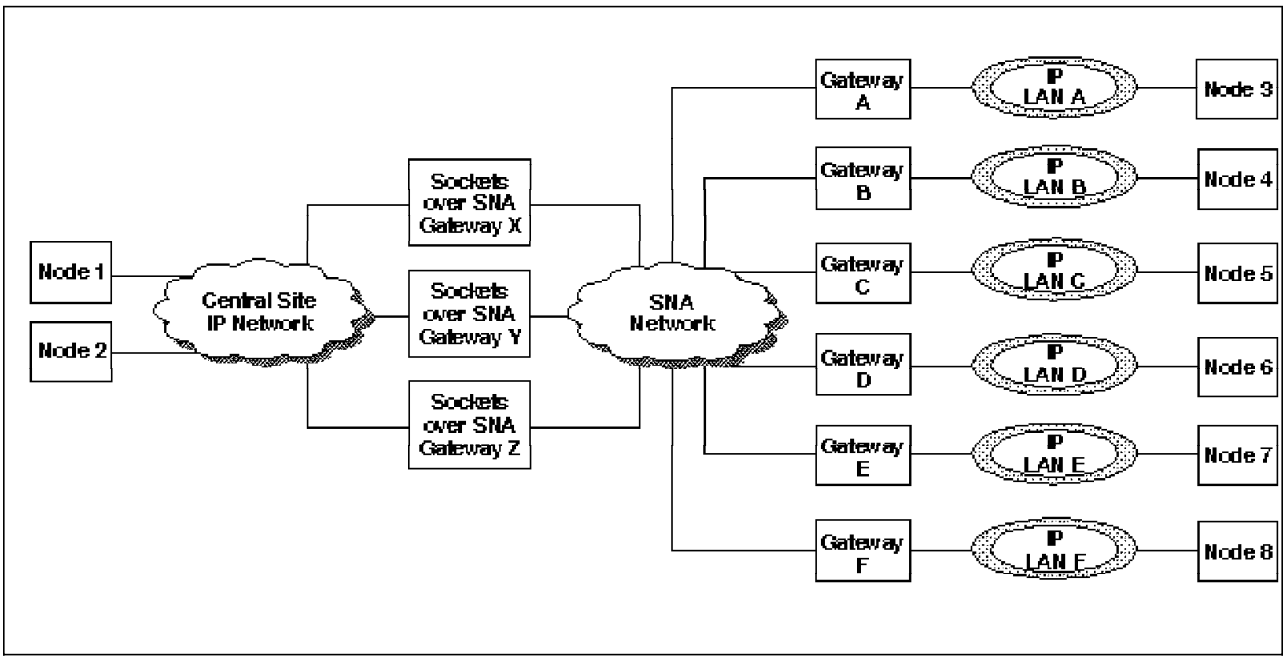


Figure 33. Using Sockets over SNA Parallel Gateways to Provide Backup and Load Balancing

In this example:

1. Gateways X, Y, and Z are Sockets over SNA parallel gateways servicing the central site IP network.

   Nodes 1 and 2 are workstations attached to the IP network. Both nodes must be running the TCP/IP routed or gated programs to receive updates from parallel gateways X, Y, and Z. These programs use the Routing Information Protocol (RIP).

2. Gateways A-F are Sockets over SNA gateways servicing remote workstations attached to IP LANs A-F.

   Gateways A-F must explicitly define three default routes to parallel gateways X, Y, and Z.

3. Nodes 3-8 are workstations attached to the IP LANs.

4. Initially, each parallel gateway services approximately one-third of the remote sites. This partitioning is done by the parallel gateways during startup.

5. If Gateway Y fails, Gateway X or Z takes over Gateway Y's remote sites within a few minutes.

Note

When a gateway fails, connection-oriented applications must be restarted. Connectionless applications continue to run, but data will be discarded until the backup gateway takes control.

When Gateway Y recovers and is operational, it eventually regains its remote sites.

Note

When the original gateway recovers, applications running through a backup gateway continue to run until completion. For applications that run a long time, it is possible that the original gateway cannot regain control until the application is stopped or the remote gateway is restarted.

For a sample scenario showing the definitions required to implement the configuration in this example, see 4.3, “Sockets over SNA Configuration” on page 62.

Important Note

Use the Sockets over SNA routes configuration panel to add or delete routes for parallel gateways. This enables the parallel gateway to broadcast the route addition or change to the native TCP/IP nodes.

This occurs because the gated program only looks in the routing table at the time that it is started and assumes that it is the only program that makes changes (add, deletes, etc.) in the routing table. So if you manually change any gated route, do not incorporate this update to the RIP messages that it sends to the IP network. The only way is to add the new route by using the Communications Server configuration panels or include it in the routing table before the Sockets over SNA Gateway function is started.

But remember that you can not add any route that depends on a non-existing host address. So, you can not add (before Sockets over SNA is started) any route manually that depends on a route that is going to be defined by the Sockets over SNA gateway function.

Actually there is no way to dynamically update these routes in a Sockets over SNA parallel gateway configuration without disabling/enabling the Sockets over SNA function.

Another consideration is that the load balancing split is defined at configuration time. When you define the parallel gateways that participate in the function, the Communications Server internally determines which gateway serves which remote network. But at startup time the parallel gateways that are alive exchange RIP updates, so they can decide how to split the network in case any of the parallel gateways is not operative.

Moreover, the split is so explicit that if you have two local parallel gateways (say LG1 and LG2) and two remote sides (say RG1 and RG2), gateway LG1 routes for RG1 and LG2 routes for RG2. LG1 points to LG2 as its router to RG2 and LG2 points to LG1 as its router to RG1.

You can see those details by issuing the PGWSPLIT, PGWSTAT and NETSTAT -R commands.

A remote network can only be served by one local parallel gateway at a time. But a local parallel gateway can serve one or more remote gateways simultaneously.

In addition, there is no supported way to have a back-to-back parallel gateway configuration. You can not have a local site with a parallel gateway

configuration that communicates with a remote site as well as a parallel gateway configuration.

There is a limitation in the AnyNet support that says you must use the same path to go and to return in the Sockets over SNA network. This means you can not go using one gateway and return using another gateway for the TCP connection. For datagrams this is different because there is no TCP connection; in this you can go and return by any path.

Note

All parallel gateway partners (including the machine being configured) need to be configured on each machine. Order is important. Failed routes will be rerouted to the next active machine in the table.

4.2.2 Datagram Retry Delay

This field specifies the time, in seconds, that Sockets over SNA waits after an SNA allocate fails before trying another allocate to the same destination. The Datagram Retry Delay timer applies only to SNA conversations that are allocated for sending datagrams. Use this timer to limit the number of failed allocates in your network.

If the default value of 0 seconds is used, there is no delay; the next datagram for the destination will again try to allocate the conversation after an allocate fails.

If excessive ping traffic is being routed to a remote gateway that is frequently unavailable, you might want to set the Datagram Retry Delay timer to a value larger than 0 seconds. This will reduce the number of errors logged by your host processor and the amount of traffic on the wide area network (WAN).

If the session is already established between the local and the remote gateway, any request (with an unavailable host destination address) is sent over the data link the same way any IP router will do it.

Note

This parameter is useful to reduce the traffic due to allocate retries to unavailable Sockets over SNA gateways only. It is not used for application data to unavailable hosts connected to an available Sockets over SNA gateway.

For each Sockets over SNA allocate that fails there is an entry that is logged by FFST/2 in the System Message Log. So, if you try to PING to an unavailable gateway or to a host that can only be reached using this unavailable gateway, then you can see a constant activity in your local gateway disk. This activity is due to the entries that are added to the message log.

You can see these error messages by going through the FFST/2 folder to the Message Log Formatter icon. The errors may look like the following:

```
10-04-1996  18:17:03   MPTNSOS    SOS0030E: Unable to allocate a conversation to USIBMRA.IPSNA01Q.
10-04-1996  18:17:03   MPTNSOS    SOS0020E: Sense Data:            08400007
10-04-1996  18:17:02   MPTNSOS    SOS0019E: Secondary return code: 00000004 ALLOCATION_FAILURE_NO_RETRY
10-04-1996  18:17:02   MPTNSOS    SOS0018E: Primary return code:   0003     ALLOCATION_ERROR
10-04-1996  18:17:02   MPTNSOS    SOS0017E: Failing Verb:          0100     ALLOCATE
10-04-1996  18:17:01   MPTNSOS    SOS0063E: ----- APPC verb failure occurred -----
10-04-1996  18:17:01   MPTNSOS    SOS0030E: Unable to allocate a conversation to USIBMRA.IPSNA01Q.
10-04-1996  18:17:01   MPTNSOS    SOS0020E: Sense Data:            08400007
10-04-1996  18:17:01   MPTNSOS    SOS0019E: Secondary return code: 00000004 ALLOCATION_FAILURE_NO_RETRY
10-04-1996  18:17:00   MPTNSOS    SOS0018E: Primary return code:   0003     ALLOCATION_ERROR
10-04-1996  18:17:00   MPTNSOS    SOS0017E: Failing Verb:          0100     ALLOCATE
10-04-1996  18:17:00   MPTNSOS    SOS0016E: ----- APPC verb failure occurred at 18:16:59 on 10/04/96 -----
```

Figure 34. Message Log Entries for Allocate Failed for Sockets over SNA

Setting the timer to a higher value will reduce the error log entries and network traffic but it will take longer to find the partner once the partner is up.

4.2.3 Route Discovery

The route discovery function provided by Sockets over SNA Gateway can help you route TCP/IP traffic more efficiently and reduce the number of explicitly defined route statements in your network. You do not have to select or configure this function.

One of the problems with large networks is how to discover that new networks or subnetworks have been added and what router to use to get to the new network or subnetwork. Sockets over SNA solves this problem by having all nodes initially use a default router that notifies other nodes when a more direct route is discovered. This is more efficient than using the typical TCP/IP solution of broadcasting routing information.

Note

To effectively use this function, algorithmic mapping of IP addresses to LU names and an APPN backbone network should be used. Otherwise, nodes must explicitly define LU names and IP addresses for all remote nodes with which they communicate.

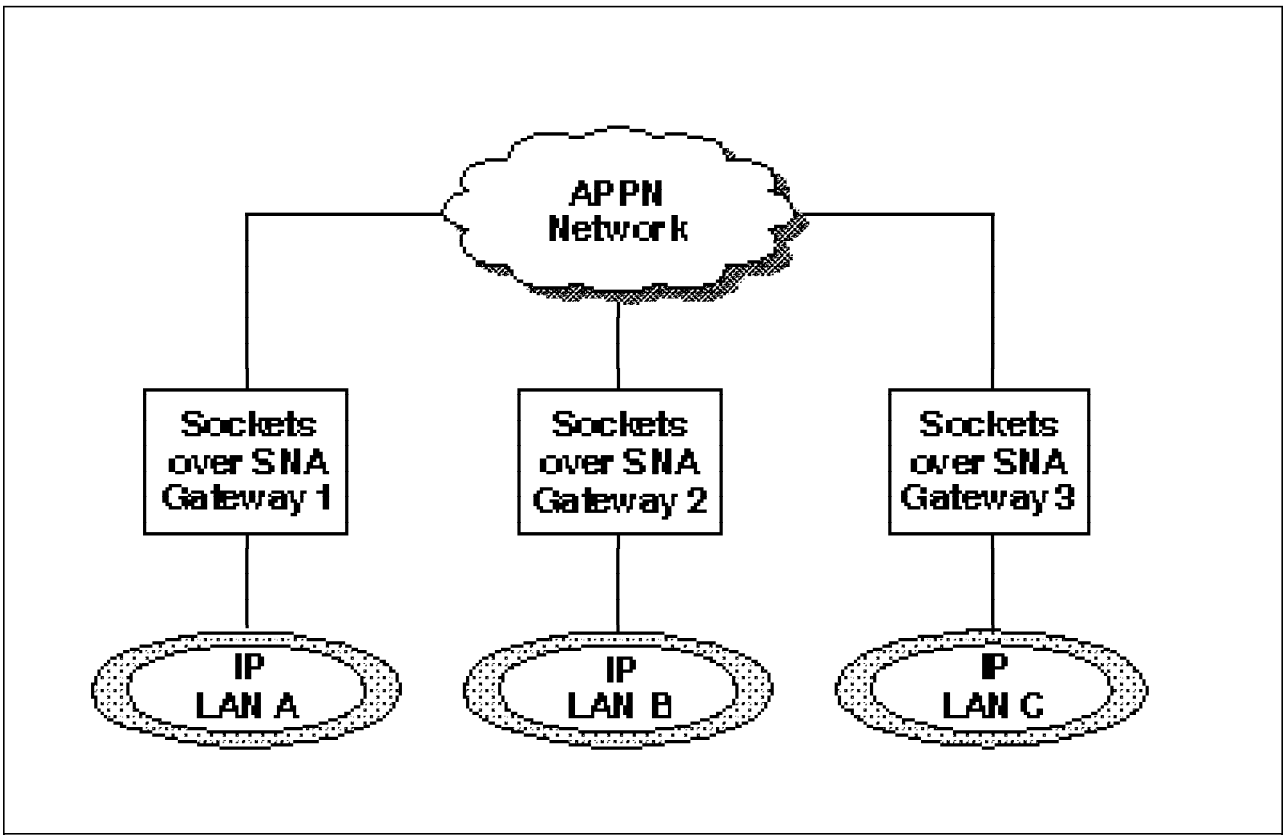


Figure 35. Example of a Network Using the Sockets over SNA Route Discovery Function

In this scenario:

- Gateways 1, 2, and 3 are Sockets over SNA gateways connected to IP LANs. These gateways connect the IP LANs to an APPN backbone network.
- Gateway 1 is the only gateway where you maintain a complete, permanent routing table.
- Gateways 2 and 3 define Gateway 1 as their default router. If a remote network or subnetwork is known to Gateway 1, Gateways 2 and 3 do not have to explicitly define these routes.
- When Gateways 2 and 3 route data to undefined networks or subnetworks, these requests are sent to their default router, Gateway 1.

  If the network or subnetwork is known to Gateway 1 and a more direct path is available, Gateway 1 sends an ICMP redirect message back to the requester indicating the path to take in the future. This ICMP redirect message updates the requester's routing table. Therefore, Gateways 2 and 3 dynamically build their routing tables for remote networks and subnetworks as needed.

  The Route Discovery function updates IP routing table entries for requester nodes located in SNA networks. To update routing entries for nodes in IP networks, you must select the Sockets over SNA option to enable RIP.
- If new gateways are added, update the permanent routing table of Gateway 1. No explicit route statements are needed for Gateways 2 and 3.

  In the routing table, route entries added by ICMP redirect messages contain a D in the flags column. This indicates that the route entry was created dynamically.

As route discovery uses ICMP redirect requests, you can add routes to the routing table using the route add command. In turn, these routes will be propagated in the SNA network as required.

Therefore, when a remote gateway from the SNA network sends a request for a route, the default Sockets over SNA gateway searches in the routing table and sends the response accordingly. This way, the remote gateway and any other IP machine, that points to it as theirs default router can get access to the newly added route dynamically.

It is a different situation that in backup and load balancing which use RIP for the local IP network updates.

Note

Routes are learned through ICMP redirects from the default router back to the requesting node. There is *no* broadcast.

Route discovery is used to update route entry tables for SNA subnets only.

No specific configuration fields are defined for this function. However, proper routing tables must be set up (one complete table with all other nodes pointing to it as their default router).

4.2.4 How Sockets over SNA Gateway Routes Data

The following steps briefly describe how the Sockets over SNA Gateway determines whether to route data over the SNA or TCP/IP interface, and address mapping is handled:

1. Sockets over SNA searches the IP routing table to find a route that enables data to reach the destination IP address. If Sockets over SNA does not find any matching routes, the connection request fails and returns completion error code EHOSTUNREACH.
2. If Sockets over SNA finds a matching route, the route entry indicates:
   - Which configured network interface is used to reach the destination:

Transport Protocol	Network Interface
SNA	sna0
TCP/IP	lan0

   - How the destination can be reached:
     - If the router address is the address of a local network interface, such as sna0, the destination network, subnet or host address can be reached directly.
     - If the router address is the address of a gateway or router, the destination can only be reached through that intermediate gateway or router.

   Figure 36 on page 57 shows an example of an IP routing table.

destination	router	netmask	refcnt	use	flags	snmp metric	intrf
default	9.37.56.1	0.0.0.0	1	43729	UG	0	lan0
9.37.56.0	9.37.57.61	255.255.248.0	0	0	U	0	lan0
9.37.56.1	9.37.57.61	255.255.255.255	1	3	UH	0	lan0
9.37.56.2	9.37.57.61	255.255.255.255	0	0	UH	0	lan0
9.37.56.3	9.37.57.61	255.255.255.255	0	440	UH	0	lan0
9.37.57.61	9.37.57.61	255.255.255.255	3	20571	UH	0	lan0
9.37.57.73	9.37.57.61	255.255.255.255	0	24	UH	0	lan0
9.37.61.110	9.37.57.61	255.255.255.255	0	50	UH	0	lan0
127.0.0.0	127.0.0.2	255.0.0.0	0	0	U	0	gw0
199.245.253.0	199.245.253.61	255.255.255.0	0	0	U	0	sna0

Figure 36. Example of an IP Routing Table

3. If the chosen route indicates that data should be routed over a native TCP/IP interface such as lan0, the destination is either on or can be reached through a native IP network. Refer to your TCP/IP documentation for more information on how TCP/IP routes data.
4. If the chosen route indicates that data should go over the SNA interface (sna0), Sockets over SNA looks up the next-hop address in the IP-LU mapping table:
   - If Sockets over SNA finds a matching entry, an LU 6.2 connection is established.
   - If Sockets over SNA does not find a matching entry, the connection attempt fails and Sockets over SNA returns completion error code EHOSTUNREACH.
5. Sockets over SNA passes the destination address and data to Communications Server or TCP/IP.

4.2.5 Routing Information Protocol (RIP)

The RIP option provided by Sockets over SNA uses the routing functions provided by the TCP/IP gated program. If you select this option, the Sockets over SNA Gateway can update routing table entries for nodes in IP networks when a new route has been created, or a route is temporarily unavailable or a more direct route is discovered.

This dynamically creates and maintains network routing tables. RIP arranges to have IP routers periodically broadcast their routing tables to neighbors.

All nodes in the IP network must be running RIP for this to occur. This can be done using the routed program provided by the TCP/IP in different platforms.

In RIP, there are two types of participants:

- Passive participants update their routing tables when they receive new information.
- Active participants also advertise their routes to other hosts in addition to updating their routing tables.

When the routing server starts, it transmits a request packet on each interface, supports broadcast packets, and then enters a loop, listening for request and response packets from other hosts.

In normal operation, the server monitors the UDP socket for the route service to receive routing information packets. When a request packet is received, the server prepares a reply based on the information maintained in its internal tables. It then sends a response packet, which contains a list of known routes and their metrics. A metric of 16 or greater is considered infinite and therefore unreachable.

Note

The RIP option updates IP routing table entries for requester nodes located in IP networks only. Routing entries for nodes in SNA networks are updated by the Sockets over SNA Route Discovery function.

To select the RIP option, use the Sockets backup and load balancing configuration window.

For more information about RIP, see RFC 1058.

Note

To display the contents of the routing table, use the netstat -r command.

When you select the RIP option Communications Server uses the gated program to RIP into the IP network. Only the RIP learned routes by the gated program and the Communications Server defined routes are propagated with RIP. If you manually add a route (for example with the route add command), this route is not sent by the gated program in their RIP updates. This is because the gated program reads the routing table only at startup and assumes that it is the only one that makes route updates.

Figure 37 on page 59 shows an example of an IP trace of an RIP update.

```
------------------------- #:6 -------------------------
 Delta Time:  0.000sec   Packet Length: 116 bytes (74 hex)
 802.5:  Dest: FF: FF: FF: FF: FF: FF   Source: C0: 00: 52: 00: 51: 83
 802.5:  Dest: 128.001.001.255    Source:  128.001.001.001
---------------------- IP HEADER -----------------------
 IP:  Version: 4 Correct    Header Length: 20 bytes
 IP:  Type Of Service: 00
 IP:     000. ....  Routine
 IP:     ...0 ....  Normal Delay
 IP:     .... 0...  Normal Throughput
 IP:     .... .0..  Normal Reliability
 IP:  Total Len: 92 (x5C) bytes          Id: 46E9
 IP:  Flags: 0
 IP:     .0..       May Fragment
 IP:     ..0.       Last Fragment
 IP:  Fragment Offset: 000
 IP:  Time To Live: 64 sec    Protocol: 17  UDP
 IP:  Header Checksum: 30A6    (Correct)
 IP:  No Options

---------------------- UDP HEADER ----------------------
 UDP:  Source Port: 520 (RouteD)    Dest Port: 520 (RouteD)
 UDP:  Length: 72 (x48)
 UDP:  Checksum: 37EB   (Correct)
---------------------- RIP Packet ----------------------
 RIP:  Command: 2  Response
 RIP:  Version: 1
 RIP:  Address Family Identifier: 2
 RIP:  IP Address:  130.001.000.000
 RIP:  Metric: 1
 RIP:  Address Family Identifier: 2
 RIP:  IP Address:  192.168.221.000
 RIP:  Metric: 1
 RIP:  Address Family Identifier: 2
 RIP:  IP Address:  192.168.222.000
 RIP:  Metric: 3
```

Figure 37. Sample IP Trace of an RIP Update

The trace in Figure 37 corresponds to a broadcast RIP update that WTR05184 (128.1.1.1) sends to the whole network 128.1.1 (.255 means all the hosts).

The trace shows you the following:

- The IP part shows IP Version 4, which carries UDP protocol, the time to live, etc.
- The UDP part shows the PORTs that are used (it is the only function of UDP).
- The RIP part shows the version of RIP and the routes that can be reached using this router.

  Sockets over SNA supports RIP Version 1 and also RIP Version 2 which can be used for variable subnetting.

Note

You should be aware that the RIP protocol uses broadcast messages. Therefore, you must be careful when you enable the RIP option in WAN links such as a TCP/IP over frame relay link. Moreover, your network may be impacted when the link capacity is not large enough and routing tables with many entries are used.

RIP information is broadcasted every 30 seconds, whether this is a LAN or WAN link.

4.2.6 Maximum Number of Connections Increased to 2000

The maximum number of gateway connections for a Sockets over SNA Gateway is 2000. Within the gateway node, a connection is represented by a gateway entry.

There are two kinds of gateway entries:

- Gateway entries for stream socket connections are created when the sockets are connected. These entries are deleted when both applications close their side of the connection.
- Gateway entries for datagram sockets are created as needed and deleted after being unused for a period of time.

During normal operation the idle timeout for a datagram gateway connection is 120 seconds. However, under a heavy load (more than 90 of the maximum allowed gateway connections are in use), the idle timeout for datagram gateway connections is reduced to 10 to 15 seconds.

Do not confuse the idle timeout for gateway entries for datagram sockets and the timeout for datagram conversations.

You have access to and control of the conversation timer that relates to the matter of bringing up a new conversation or keeping it up. This is the parameter that you can set in the Sockets over SNA Local Parameters that defaults to 90 seconds. You can see that a conversation is allocated using the subsystem management, displaying the LU 6.2 sessions and seeing for a particular mode whether the sessions have a conversation associated with them or using the display -se command to see if the corresponding session has a conversation ID different from X'00000000'.

The gateway entry timer is not able to be set. It manages the timeout for gateway entries and is auto adjustable by Communications Server in case of a heavy load. You can see the entries that are in use using the GWSTAT command.

Other parameters that must be reviewed in a loaded Sockets over SNA Gateway are:

- The maximum number of threads supported by OS/2 is set by the CONFIG.SYS entry THREADS. The default value is 64, but Communications Server changes the entry to 4095. Each gateway connection requires a maximum of two threads. In addition, the maximum number of SNA sessions per gateway connection is one for full-duplex and two for half-duplex.

The connection type may affect the achievable limit. Stream connections require a separate set of threads for each connection. Many datagram connections can use the same set of threads.

- Maximum number of OS/2 global descriptor table (GDT) entries assigned to Sockets over SNA that determines the maximum amount of available memory. The default is 80 and is probably sufficient for up to 250 gateway connections. The message SOS0002E can indicate that this number is too low. To increase this amount see the following example:

  ```
  DEVICE=drive:\MPTN\PROTOCOL\SOCKETS.SYS /GDT:500
  ```

  The maximum number supported by Sockets over SNA (SOCKETS.SYS driver) is 500.

  If there are not enough GDTs, then we receive memory allocation errors.

GWSTAT is a good tool to use to evaluate connection information. The connections are displayed as *gateway entries*.

4.2.7 Variable Subnetting Is Now Supported

A subnet mask is a bit template that identifies to the TCP/IP protocol code the bits of the host address that are to be used for routing to specific subnets.

There is a default subnet masks for each IP address class, that says what octets are used for the network address and the host address.

For example, an A class network has a default subnet mask of 255.0.0.0. This means that only the first octet is meaningful for the network address. The mask 255.0.0.0 means ff.0.0.0 in hexadecimal and 11111111.0.0.0 in binary, so that the first eight bits are used for the network address.

In the same way, Sockets over SNA can use a variable subnet mask. This means that we can use other subnet masks, not only the defaults.

For example, if we use a subnet mask of 255.255.255.0 all of the addresses that begin with 9.10.11 are in the same network but all that are in 9.10.12 correspond to another network. This is because the first three octets are meaningful for network address decision.

In this way we can subdivide a given IP network address into several subnetwork addresses as we need according to our network topology.

The subnet mask is configured on the Sockets over SNA Routes panel.

The variable subnetting support requires TCP/IP 4.0 for the IP part of the network.

Routes are defined to remote subnets by including a subnet mask in the route table entries.

4.3 Sockets over SNA Configuration

The Sockets over SNA configuration notebook has been replaced with a set of Communications Server panels.

Panel access is still done through the Sockets menu item.

In order to explain the Sockets over SNA enhancements in IBM Communications Server Release 4.1, we set up a scenario (see Figure 38 on page 64) that includes the following stations and gateways:

1. WTR05200. A machine with an OS/2 and TCP/IP stack. It is token-ring connected to ring number 1.
2. WTR05184. A machine with an OS/2 and TCP/IP stack. It also has IBM Communications Server Release 4.1 installed and it is token-ring connected to ring 1 for TCP/IP. It has a second token-ring adapter connected to ring 2 for SNA connectivity and has a frame relay link to WTR05303 which acts as a Sockets over SNA parallel gateway with WTR05217.
3. WTR05303. An OS/2 machine with IBM Communications Server Release 4.1 and TCP/IP stack. It is token-ring attached to ring 2 for SNA connectivity and has a second token-ring adapter for TCP/IP connectivity to ring 3. It is also a Sockets over SNA Gateway.
4. WTR05217. An OS/2 machine with IBM Communications Server Release 4.1 and TCP/IP stack. It is token-ring connected to ring 1 for TCP/IP traffic and has a second token-ring adapter connected to ring 2 for SNA traffic. It is a partner Sockets over SNA gateway of WTR05184 in the Sockets over SNA parallel gateway function.
5. WTR05142. An OS/2 machine with a TCP/IP stack that acts as a Web server. It is token-ring attached to ring 3.
6. WTR05600. An OS/2 machine with IBM Communications Server Release 4.1 and a TCP/IP stack. It is token-ring attached to ring 2 for SNA connectivity and it has a second token-ring adapter for TCP/IP connectivity to ring 4. It is a Sockets over SNA Gateway.
7. WTR05700. An OS/2 machine with a TCP/IP stack. It is token-ring attached to ring 4.

Table 1 gives a detailed description of the configurations of the machines, showing their addresses, protocols and functions.

Table 1 (Page 1 of 2). Machine Configurations

Machine Name	**WTR05184**		
Functionality	Sockets over SNA Parallel Gateway		
Adapters	Address		Protocol
0	Token-Ring	400052005184	802.2 * 192.168.221.1
	Ring Number	2	
1	Token-Ring	No Relevant	TCP/IP 128.1.1.1
	Ring Number	1	
2	Frame Relay	400011110002	802.2
	Ring Number	151	

Table 1 (Page 2 of 2). Machine Configurations

Machine Name	**WTR05200**		
Functionality	TCP/IP Machine requester		
Adapters	Address		Protocol
0	Token-Ring	No relevant	TCP/IP 128.1.1.9
	Ring Number	1	
Machine Name	**WTR05142**		
Functionality	TCP/IP Machine Server		
Adapters	Address		Protocol
0	Token-Ring	No Relevant	TCP/IP 192.168.222.8
	Ring Number	3	
Machine Name	**WTR05303**		
Functionality	Sockets over SNA Gateway		
Adapters	Address		Protocol
0	Token-Ring	400052005229	802.2 * 192.168.221.2
	Ring Number	2	
1	Token-Ring	No Relevant	TCP/IP 192.168.222.2
	Ring Number	3	
2	frame relay	400011110002	802.2
	Ring Number	151	
Machine Name	**WTR05217**		
Functionality	Sockets over SNA Parallel Gateway		
Adapters	Address		Protocol
0	Token-Ring	No Relevant	TCP/IP 128.1.1.3
	Ring Number	1	
1	Token-Ring	400052005217	802.2 * 192.168.221.3
	Ring Number	2	
Machine Name	**WTR05600**		
Functionality	Sockets over SNA Gateway		
Adapters	Address		Protocol
0	Token-Ring	400052005600	802.2 * 192.168.221.4
	Ring Number	2	
1	Token-Ring	No Relevant	TCP/IP 192.168.223.4
	Ring Number	4	
Machine Name	**WTR05700**		
Functionality	TCP/IP Machine Server		
Adapters	Address		Protocol
0	Token-Ring	No Relevant	TCP/IP 192.168.223.7
	Ring Number	4	

Note: An asterisk (*) in front of an IP address denotes that this IP address is used for Sockets over SNA traffic.

Figure 38 on page 64 shows you the connections that we used for this Sockets over SNA scenario.

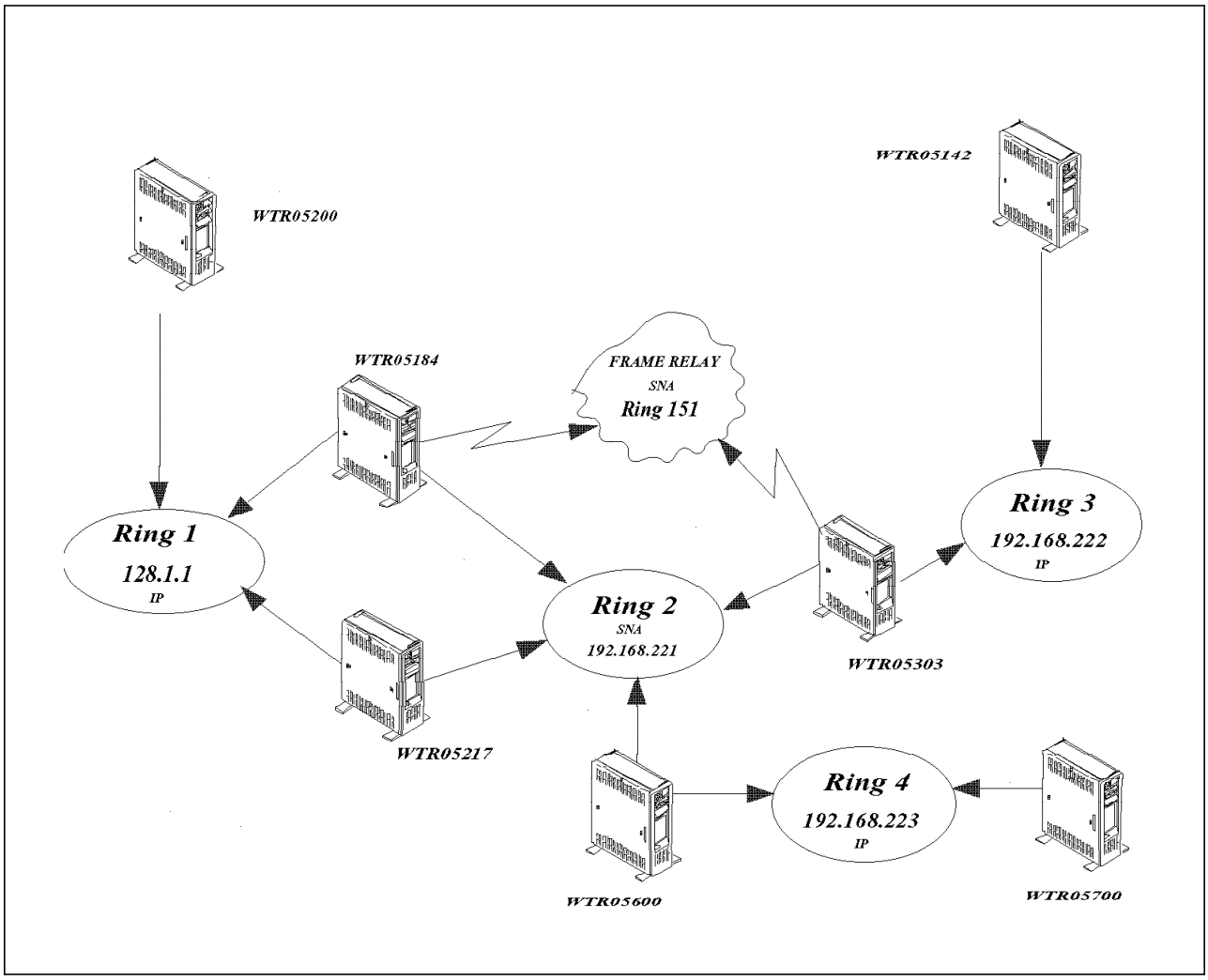


Figure 38. Sockets over SNA Scenario

We are going to follow the configuration steps needed to configure the WTR05184 machine that is a Sockets over SNA Parallel Gateway and also has a frame relay backup link to WTR5303. The other machines are very similar, at least to the Sockets over SNA Gateways, so we are going to remark when a special consideration must be taken.

In order to properly configure Sockets over SNA you must consider certain planning steps:

1. Plan the routing in your Sockets over SNA network
2. Set up Sockets over SNA
3. Configure SNA and MPTS

4.3.1 Planning the Routing in Your Sockets over SNA Network

For planning purposes we assume that you are already familiar with the TCP/IP terminology and routing aspects. Refer to the corresponding TCP/IP documentation for these concepts.

Here we want to explain certain concepts about how the SNA network ID is used by the Sockets over SNA function.

For each route you define to the SNA interface (sna0), there must be a corresponding SNA network ID to which the IP network address is mapped. The number of SNA network IDs you define depends on how you want to map the IP network to the SNA network.

For example, if the socket applications using SNA are configured to use subnetworks 9.67.0.0 and 9.77.0.0, you can define an SNA network ID that corresponds to each IP subnetwork, or you can define one SNA network ID that corresponds to both subnetworks. Sockets over SNA does not require unique one-to-one mapping between an IP network address and an SNA network ID.

If you have already set up your SNA network, you can use the existing network ID.

4.3.2 Setting Up Sockets over SNA

In this section we are going to explain the steps that are necessary to configure the Sockets over SNA Gateway function of Communications Server using a parallel gateway scenario.

We access this configuration using the Sockets pull-down menu of the Communications Server Configuration Definition panel as shown in Figure 39 on page 66.

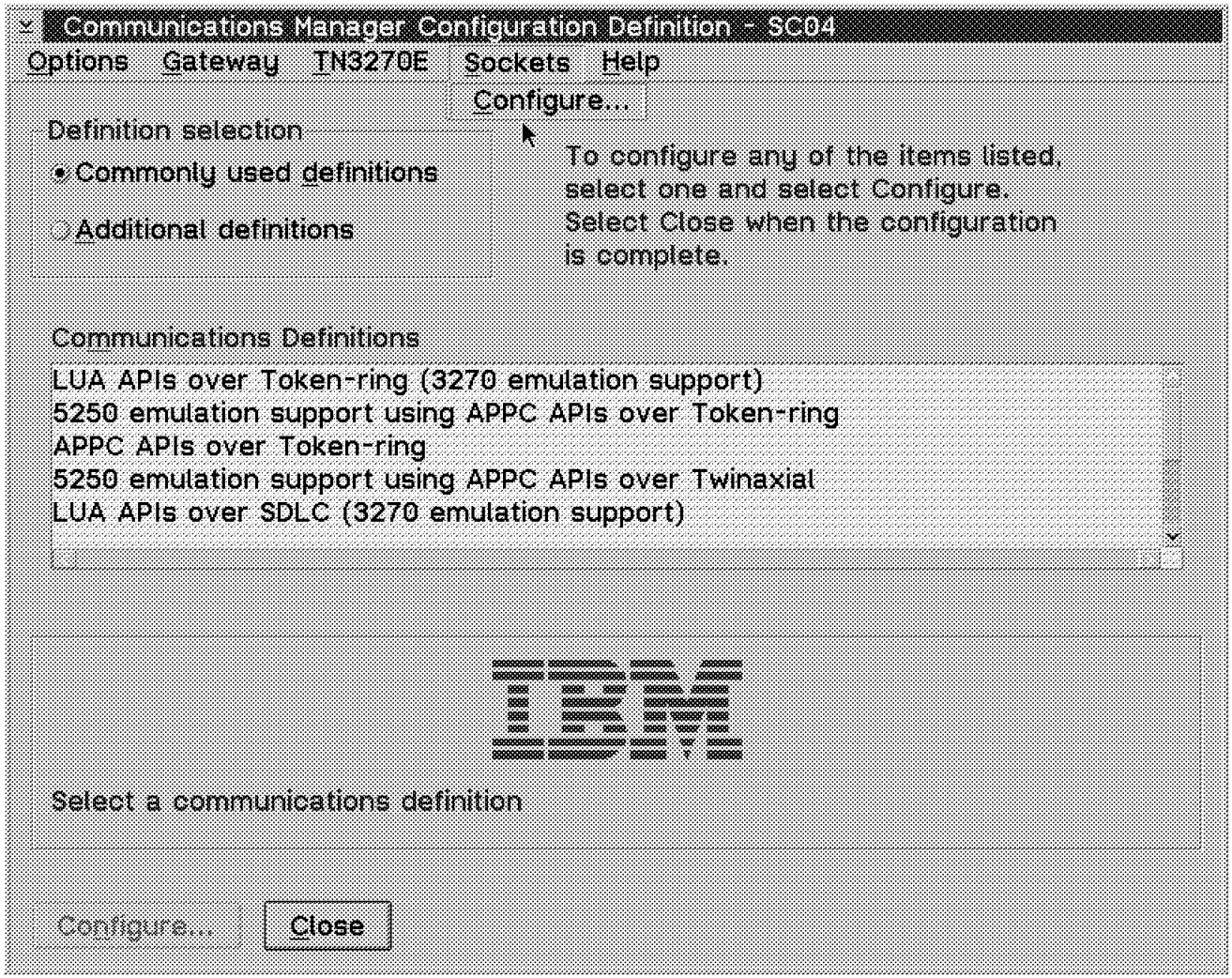


Figure 39. Communications Server Configuration Definition

There are four items that must be configured:

1. The Sockets over SNA local parameters
2. The Sockets over SNA IP address to LU mappings
3. The Sockets over SNA routes
4. The Sockets over SNA backup and load balancing

They appear in the following figure:

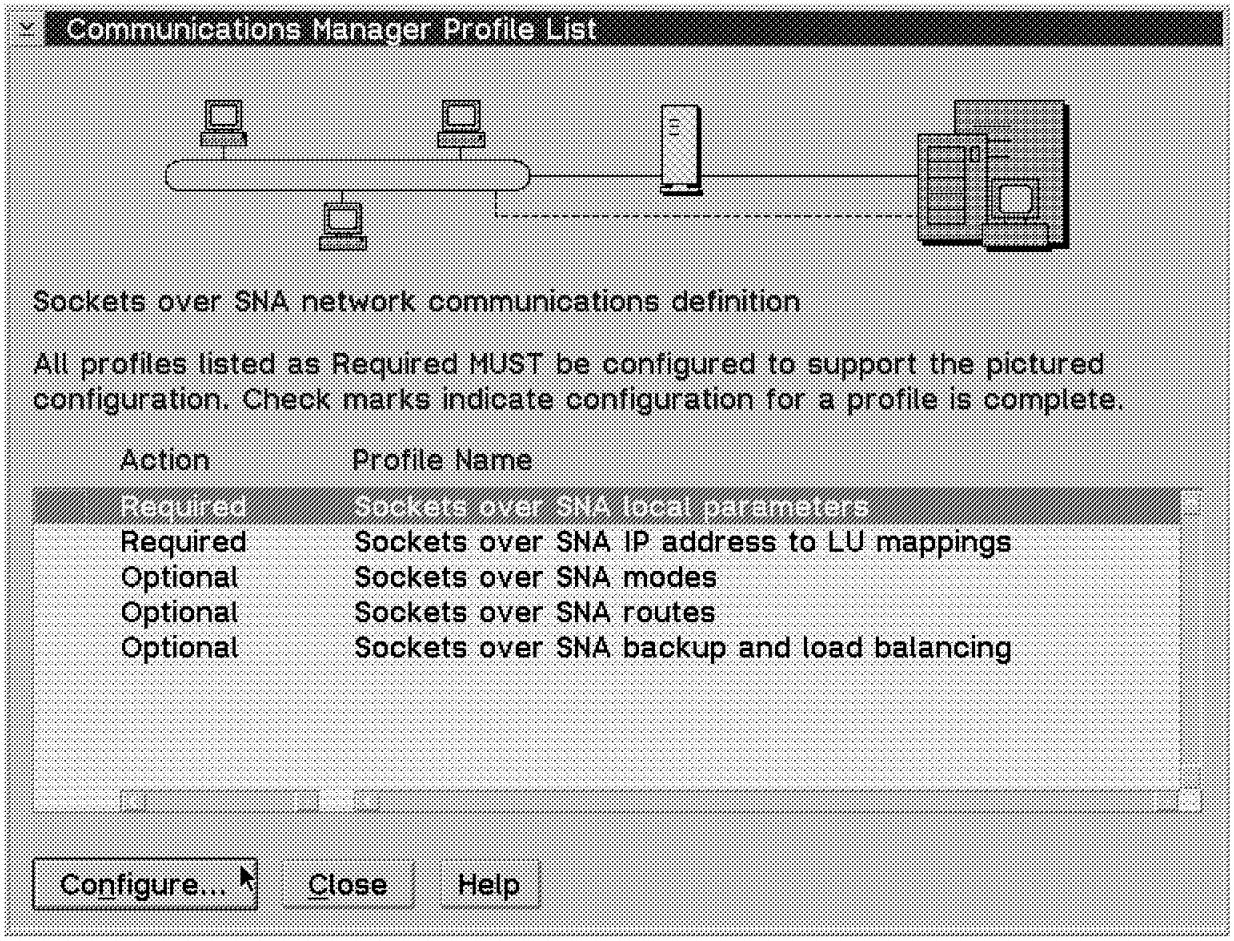


Figure 40. Communications Server Profile List

4.3.2.1 The Sockets over SNA Local Parameters

Here we must define the local characteristic of our Sockets over SNA interface.

The IP address is the address that we are assigning to the SNA interface. This interface is going to be known as sna0 for the TCP/IP protocol stack.

Also, now we can define the subnet mask that corresponds to this IP address and the idle timeout and datagram retry delay.

Figure 41 on page 68 gives an example of this panel configuration. To get there you have to select the **Sockets over SNA local parameters** in the Profile Name list.

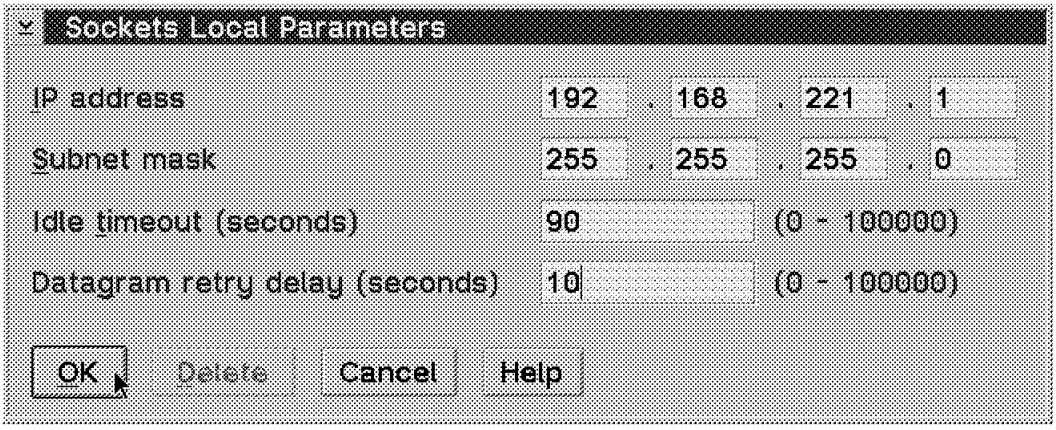


Figure 41. Sockets Local Parameters

This configuration in TCP/IP is the equivalent of the SNA Local Characteristics in Figure 59 on page 83 for SNA. It defines the local host address and the network where we are located. Remember that the netmask in conjunction with the IP address determines the host address and the network address in the TCP/IP world.

When you save the Sockets over SNA local node configuration information, Communications Server builds an ifconfig statement using the specified address and subnet mask. When you start Communications Server:

1. The ifconfig command is executed and initializes the SNA network interface (sna0).
2. Sockets over SNA starts monitoring sna0 for any data sent to its local IP address.
3. A route entry for sna0 is added to the routing table.

4.3.2.2 The Sockets over SNA IP Address to LU Mappings

So far, we have defined both interfaces, SNA and TCP/IP, but we now need some kind of connection between them. In order to complete the address mapping process, there must be a configuration option that allows you to specify what IP address corresponds to what SNA address.

The address mappings in Sockets over SNA are entered in the following configuration panel as shown in Figure 42 on page 69.

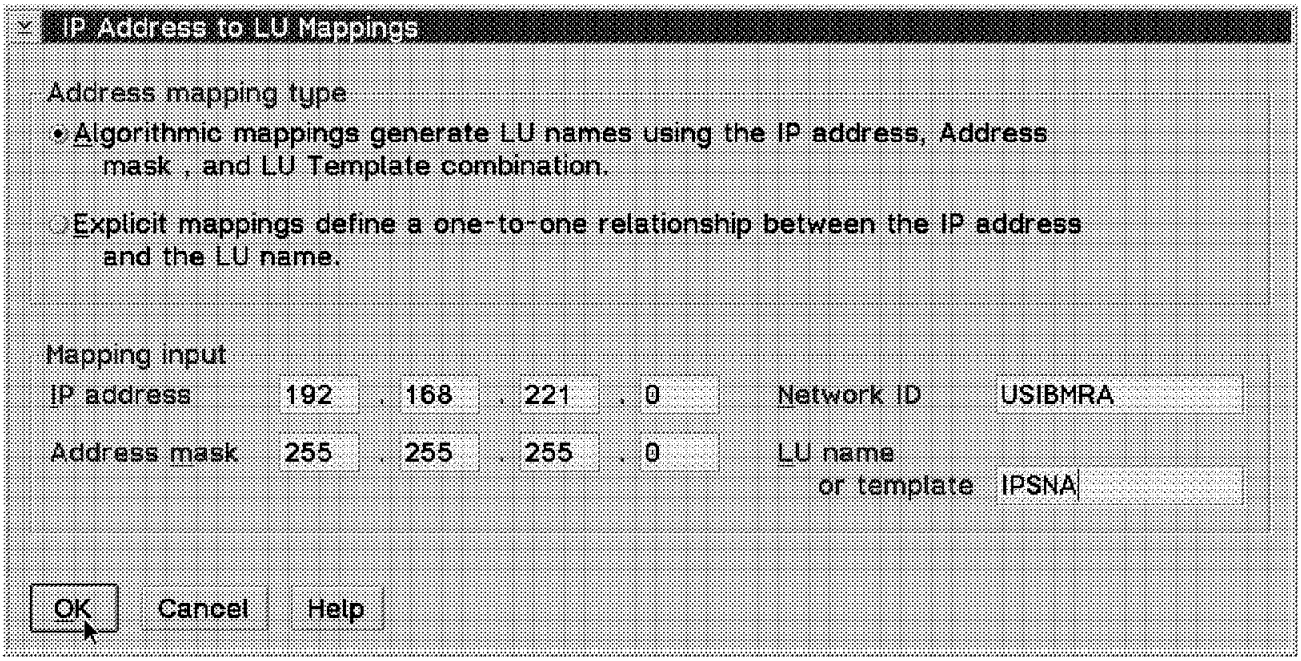


Figure 42. IP Address to LU Mappings

We use algorithmic mapping, because it is the preferred method if you are connecting several nodes. If you want to use explicit mappings you must replicate exactly the same definitions for each node using Sockets over SNA in each machine. So this can be a long tedious task and, of course, subject to errors.

In this case the IP subnet address 192.168.221 represents all the nodes that can be in the Sockets over SNA subnet. The algorithmic address mapping procedure will generate the proper fully qualified LU names by using the configured network ID (USIBMRA) and the template (IPSNA).

We need this same definition in any node that participates in the Sockets over SNA cloud so that all have the same definitions.

So we are going to have LU names as USIBMRA.IPSNA001, USIBMRA.IPSNA002 and so on for each Sockets over SNA node that participates.

The algorithm uses the masked part of the IP address to map it to the SNA network ID. The unmasked part is mapped to the corresponding LU name in conjunction with the LU template.

For example, if address mask 255.255.255.0 is specified, the 32-bit IP address is used as follows:

- The first 24 bits are mapped to the specified network ID.
- The last 8 bits are used to generate the SNA LU name.

It is recommended that you use one LU template for all Sockets over SNA nodes for the following reasons:

- Usually only one entry is needed in the IP-LU mapping table of each node to represent all other Sockets over SNA nodes.

- New nodes can be added to the network without requiring any changes in the mapping tables of existing nodes.

4.3.2.3 The Sockets over SNA Routes

You do not have to define routes to local networks or subnetworks. IP addresses that you assign to local interfaces are used, in conjunction with a subnet mask, to generate entries in the IP routing table.

Explicitly define routes for the following scenarios:

- The Sockets over SNA node is routing data to a node that can only be reached through another gateway or router.
- Add a route with a hop count of 1 to 15. This value indicates the number of hops to the destination network, subnetwork, or host.

  Enter the IP address of the intermediate gateway or router as the router address.
- The Sockets over SNA node is routing data to a node that can be reached directly, but the destination IP address is not on the local network or subnetwork.

  Add a route with a hop count of 0. Enter the IP address of the local interface as the router address.

In order to define routes, use the Sockets routes configuration window. The information you specify is used by Communications Server to generate a routing table entry.

In order to define routes select the Sockets over SNA routes optional feature in the Profile List and then select the **Insert After** option in order to add a route definition as shown in Figure 43.

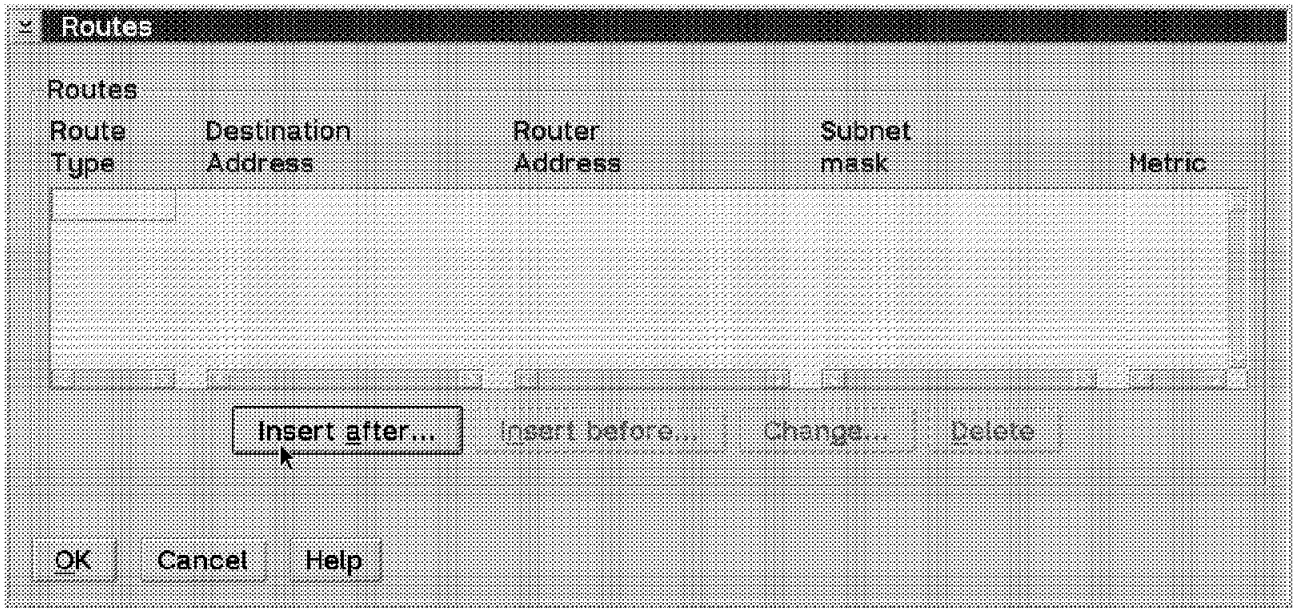


Figure 43. Routes

Note

Use the Sockets over SNA routes configuration panel to add or delete routes for parallel gateways. This enables the parallel gateway to broadcast the route addition or change to the native TCP/IP nodes.

Next, we define the required routes. We are using the two parallel gateways as the default routers for the Sockets over SNA subnet and therefore WTR05303 will point to both machines as default gateways and WTR05184 and WTR05217 are going to define subnet routes to the 192.168.222 IP subnetwork.

This way, if we assume that the 128.1.1 subnetwork is the central site and the main or only place where we want to maintain the routes for all the network, we only need to update the explicit routes in the two parallel gateways WTR05184 and WTR05217. Also, updates will have to be included in both gateways since they act as a backup to each other.

In any remote site that we add, such as the one where WTR05303 acts as a gateway (that is, 128.168.222), you only have to define a default router to the parallel gateways; all the other routes will be learned via Sockets over SNA in order to reach any other sites.

Figure 44 shows the definitions that we have entered in WTR05184 and WTR05217.

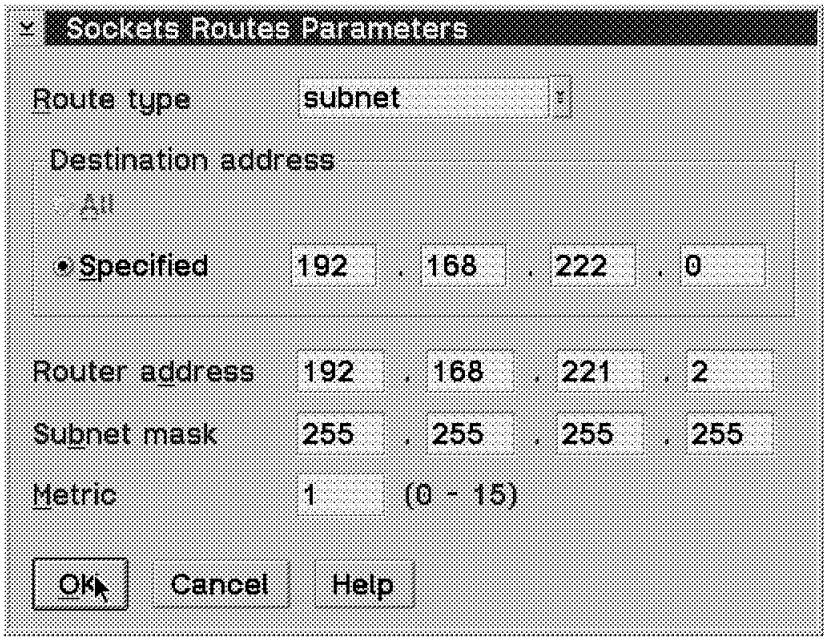


Figure 44. Routes

In WTR05303 we use a default type route as shown in the following figure.

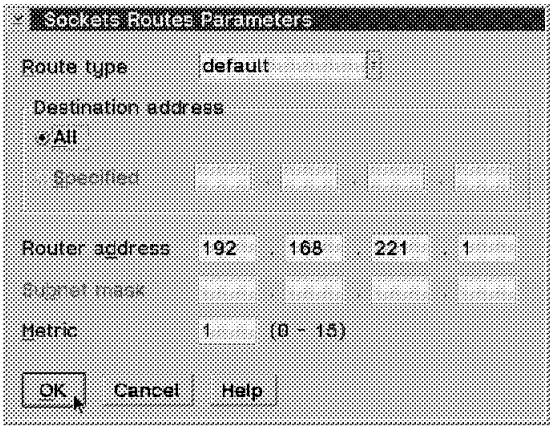


Figure 45. Sockets Routes Parameters

4.3.2.4 The Sockets over SNA Backup and Load Balancing

These definitions only apply to WTR05184 and WTR05217 because they are Sockets over SNA parallel gateways.

Select the Sockets over SNA backup and load balancing optional feature of the Profile List and define the machine as a parallel gateway as in Figure 46.

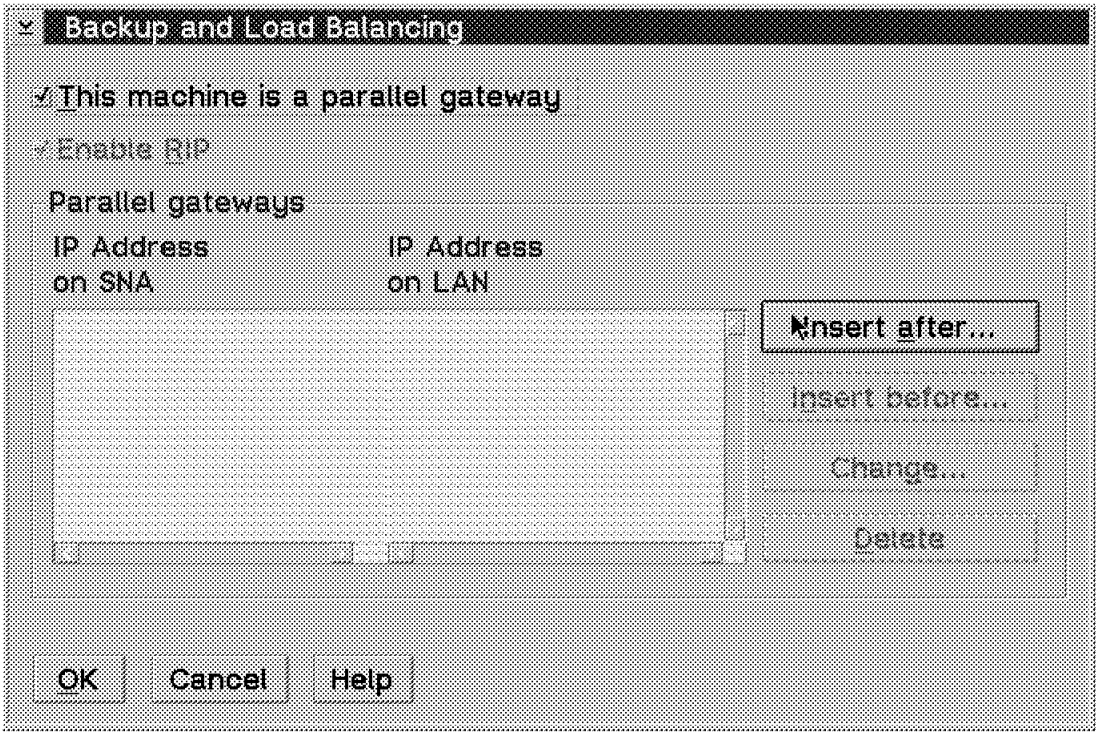


Figure 46. Backup and Load Balancing

Then select the **Insert After** button to define all the machines that participate in the parallel function (see Figure 47 on page 73 and Figure 48 on page 73).

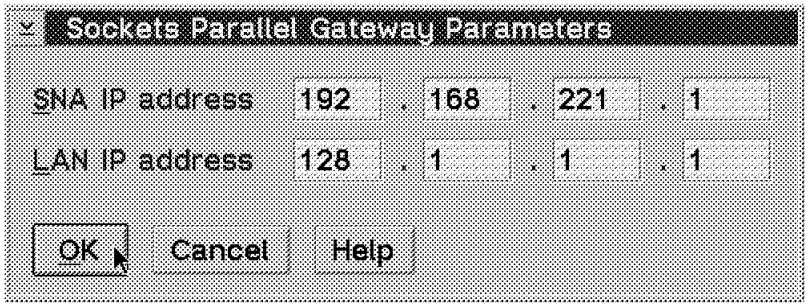


Figure 47. Sockets Parallel Gateway Parameters WTR05184

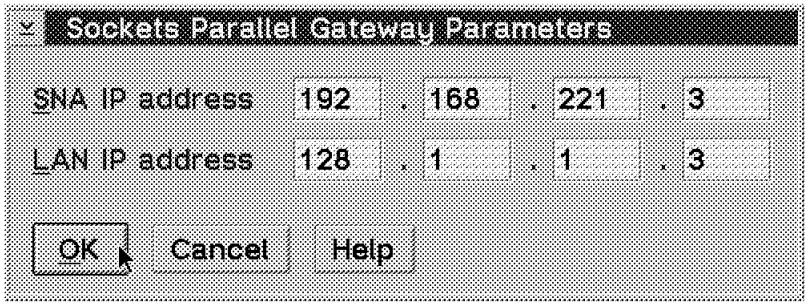


Figure 48. Sockets Parallel Gateway Parameters WTR05217

To define a parallel gateway each machine must know the two IP addresses that the other machine uses.

4.3.2.5 The Sockets over SNA Modes

Sockets over SNA uses LU 6.2 conversations to enable communication between Sockets applications programs. When an LU 6.2 conversation is established, Sockets over SNA defines the mode and its associated session characteristics of the connection. Communications Server uses the mode name to identify the characteristics of the connection between the two Sockets over SNA nodes.

The default Sockets over SNA mode is BLANK. You can use the default or define your own. It is especially useful when you have different kinds of traffic flowing in the same link and this traffic has different priorities.

For example, in a remote site you can have a mission-critical SNA application such as a CICS transaction in a bank office. This transaction is high priority and therefore it must have preference over all other traffic. However, using the same link or connection, you also have other less critical SNA transactions, some Telnet applications going to a local server and some FTPs (TCP/IP file transfers) of large files.

All applications must run concurrently using the same link at the same hours, and with same origin and destination.

In this scenario, the following scheme can be implemented:

- Use a session mode for the mission-critical traffic (high priority) and let this mode have the highest priority in the class of service (COS).
- Use a second mode for both the less critical SNA applications and the Telnet applications and let this mode use a medium priority in the class of service.

- Use a third mode for your file transfers or batch traffic and let this mode use low priority in the class of service, limit the RU sizes and pacing values. Fixed pacing should also be considered.

So, now you can share the link between all of these applications in a user-defined way, with priorities, flow control, congestion control and of other benefits that SNA/APPN/HPR offers you.

By using these features, you can also improve your TCP/IP traffic. To accomplish this, you will have to configure the session mode based on the port number of your sockets application. By following these rules, you can prevent low priority traffic from saturating the communication link or the intermediate nodes.

The following are examples of the Sockets over SNA panels that you can configure in order to change your mode definition.

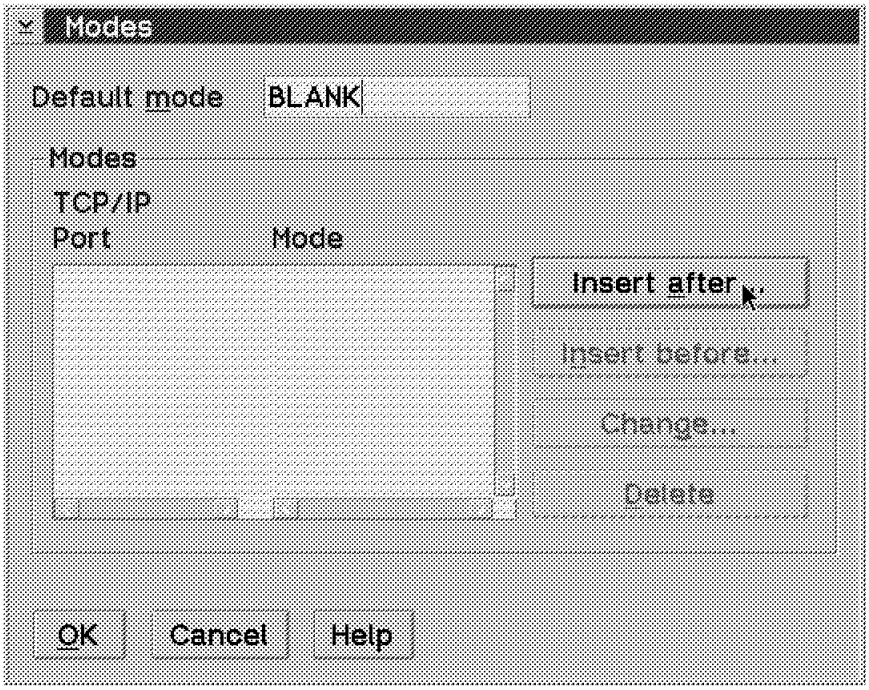


Figure 49. Sockets over SNA Modes

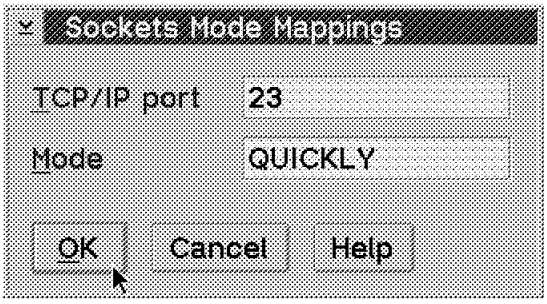


Figure 50. Sockets over SNA Mode Assignment

The mode that you use for Sockets over SNA must be defined in the SNA features. You can use an existing mode or create your own one.

In the Communications Server Configuration Definitions panel use the Options pull-down menu and select **Configure Any Profile and Feature**. A long list appears. Then select **SNA Features and Modes**. Figure 51 on page 75 gives an example of a mode addition.

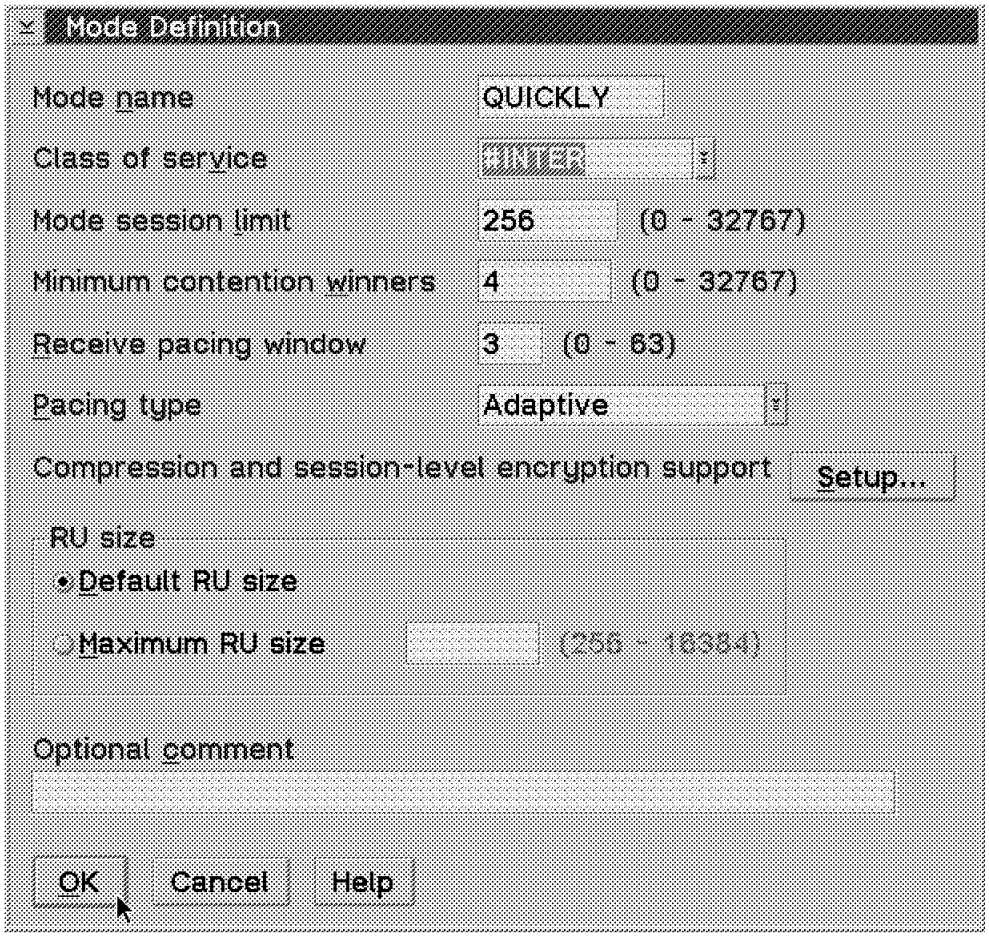


Figure 51. SNA Features Mode Definition

The following figures show you how the response files look for the Sockets over SNA parallel gateways definitions.

```
SOCKETS=(
     SNA_STARTOPT_IDLE=180
     SXRETRYWAIT=0
     SNA_MODE_DEFAULT=BLANK
     SXRIP=YES
     SXPGWFILE=YES
     SNA_LOCAL_IPADDR=192.168.221.1
     SNA_LOCAL_MASK=255.255.255.0
     SNA_ALGORITHMIC=(
          IPADDR=192.168.221.0
          MASK=255.255.255.0
          NETID=USIBMRA
          LUTEMP=IPSNA
)
     PGWPAIRS=(
          SNANET=192.168.221.1
          IPNET=128.1.1.1
)
     PGWPAIRS=(
          SNANET=192.168.221.3
          IPNET=128.1.1.3
)
     ROUTE=(
          DEST=192.168.223.0
          TYPE=subnet
          ROUTER=192.168.221.4
          NETMASK=255.255.255.0
          METRIC=1
)
     ROUTE=(
          DEST=192.168.222.0
          TYPE=subnet
          ROUTER=192.168.221.2
          NETMASK=255.255.255.0
          METRIC=1
)
     SNA_MODE=(
          PORT=23
          MODE=QUICKLY
)
)
```

Figure 52. Response File for Sockets over SNA Parallel Gateway WTR05184

```
SOCKETS=(
     SNA_STARTOPT_IDLE=90
     SXRETRYWAIT=0
     SNA_MODE_DEFAULT=BLANK
     SXRIP=NO
     SXPGWFILE=NO
     SNA_LOCAL_IPADDR=192.168.221.2
     SNA_LOCAL_MASK=255.255.255.0
     SNA_ALGORITHMIC=(
          IPADDR=192.168.221.0
          MASK=255.255.255.0
          NETID=USIBMRA
          LUTEMP=IPSNA
)
     ROUTE=(
          DEST=ALL
          TYPE=default
          ROUTER=192.168.221.1
          METRIC=1
)
     ROUTE=(
          DEST=ALL,1
          TYPE=default
          ROUTER=192.168.221.3
          METRIC=1
)
     SNA_MODE=(
          PORT=23
          MODE=QUICKLY
)
)
```

Figure 53. Response File for Sockets over SNA Gateway WTR05303

4.3.3 Configure SNA and MPTS

The following steps show the other configurations that must be used with the Sockets over SNA configuration in order for it to work.

You have to define the SNA part of the Communications Server, the connections that are going to be used and the adapters and the TCP/IP part of the protocol stack.

4.3.3.1 Configure SNA

After you configure Sockets over SNA, you must configure SNA DLC connectivity. From the Communications Server Definition window, we select **APPC APIs over Token-Ring** (that is, the ones that we are going to use for this example) as shown in Figure 54 on page 78.

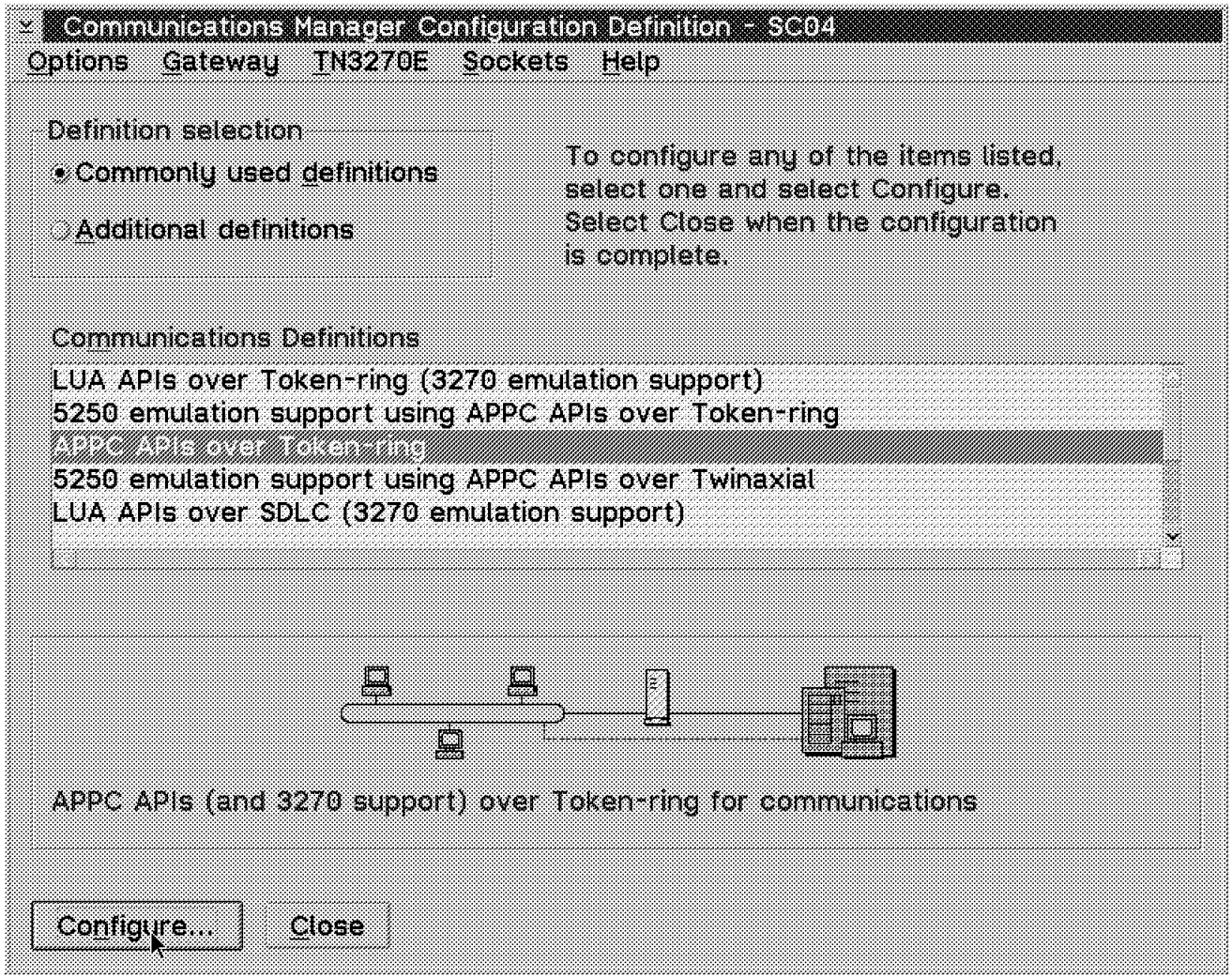


Figure 54. Communications Server Configuration Definition

Now you have to define your SNA Network ID and your Local node name. These two parameters define a single node in any SNA network. This combination must be unique across networks that want to connect.

We define the node WTR05184 as a network node. It will act as an SNA router (for SNA traffic) as well as a Sockets over SNA gateway in order to handle TCP/IP Sockets applications.

Figure 55 on page 79 gives an example.

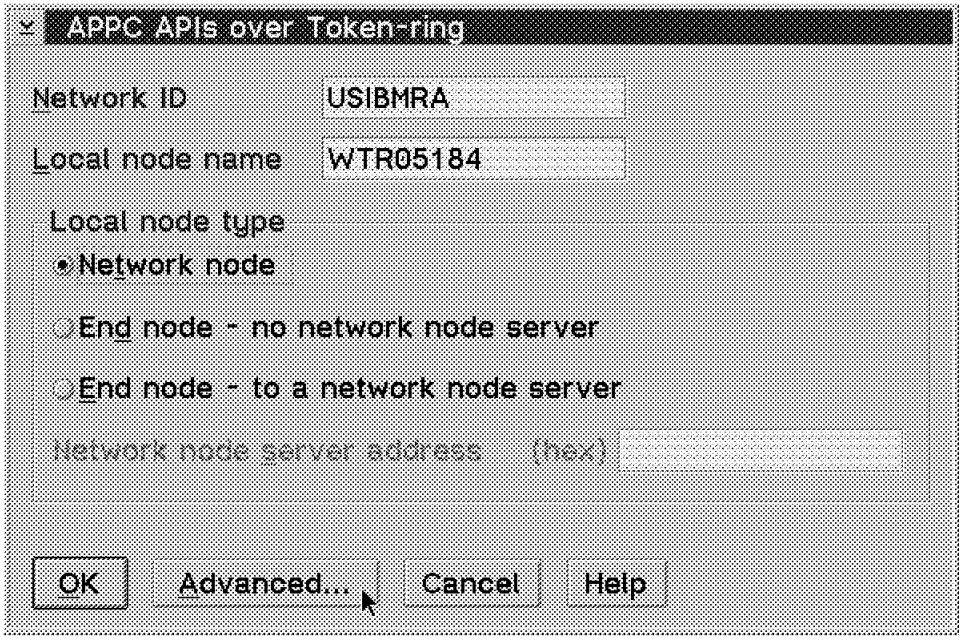


Figure 55. APPC APIs over Token-Ring

We use the advanced configuration in order to review the definitions that Communications Server automatically adds and also to add the connection definitions that we need.

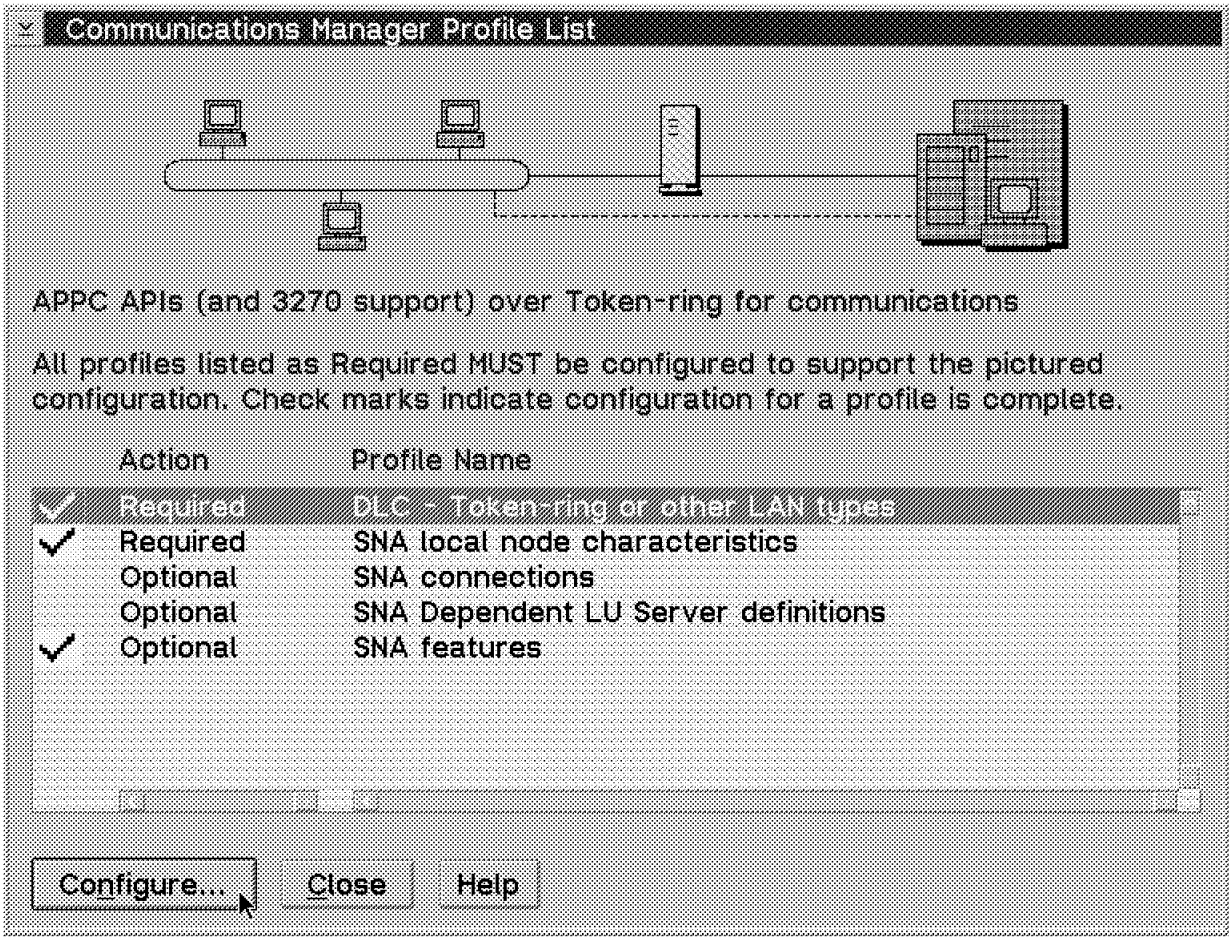


Figure 56. Communications Server Profile List

As shown in Figure 57 on page 81 and Figure 58 on page 82, remember to configure the correct adapter. We need adapter Zero for token-ring 802.2 traffic and adapter Two for frame relay 802.2 traffic.

Both definitions, are the same. For documentation purposes only, put in a different LAN ID to remember which number corresponds to which adapter.

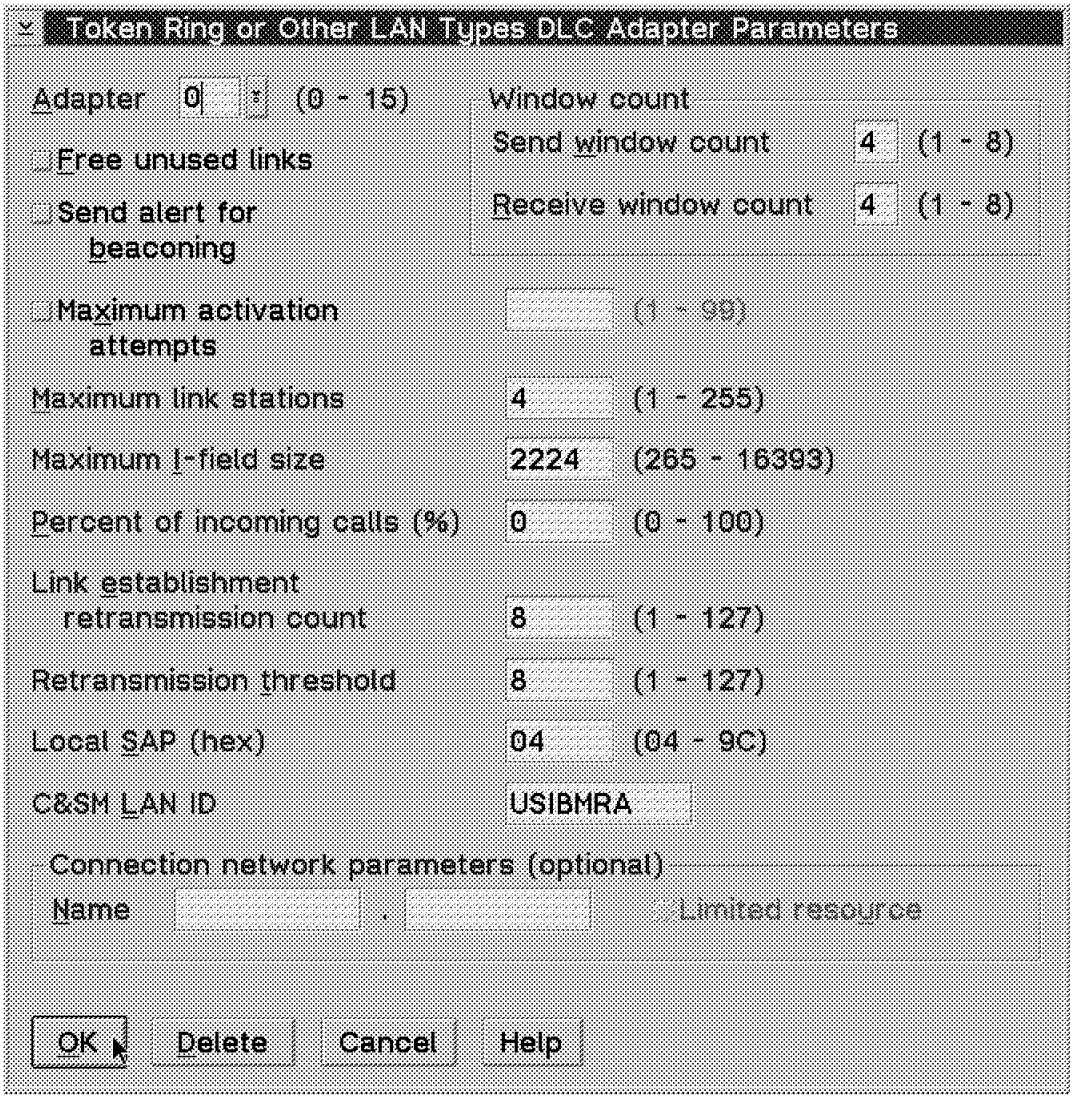


Figure 57. Token-Ring DLC Parameters - Token-Ring Adapter

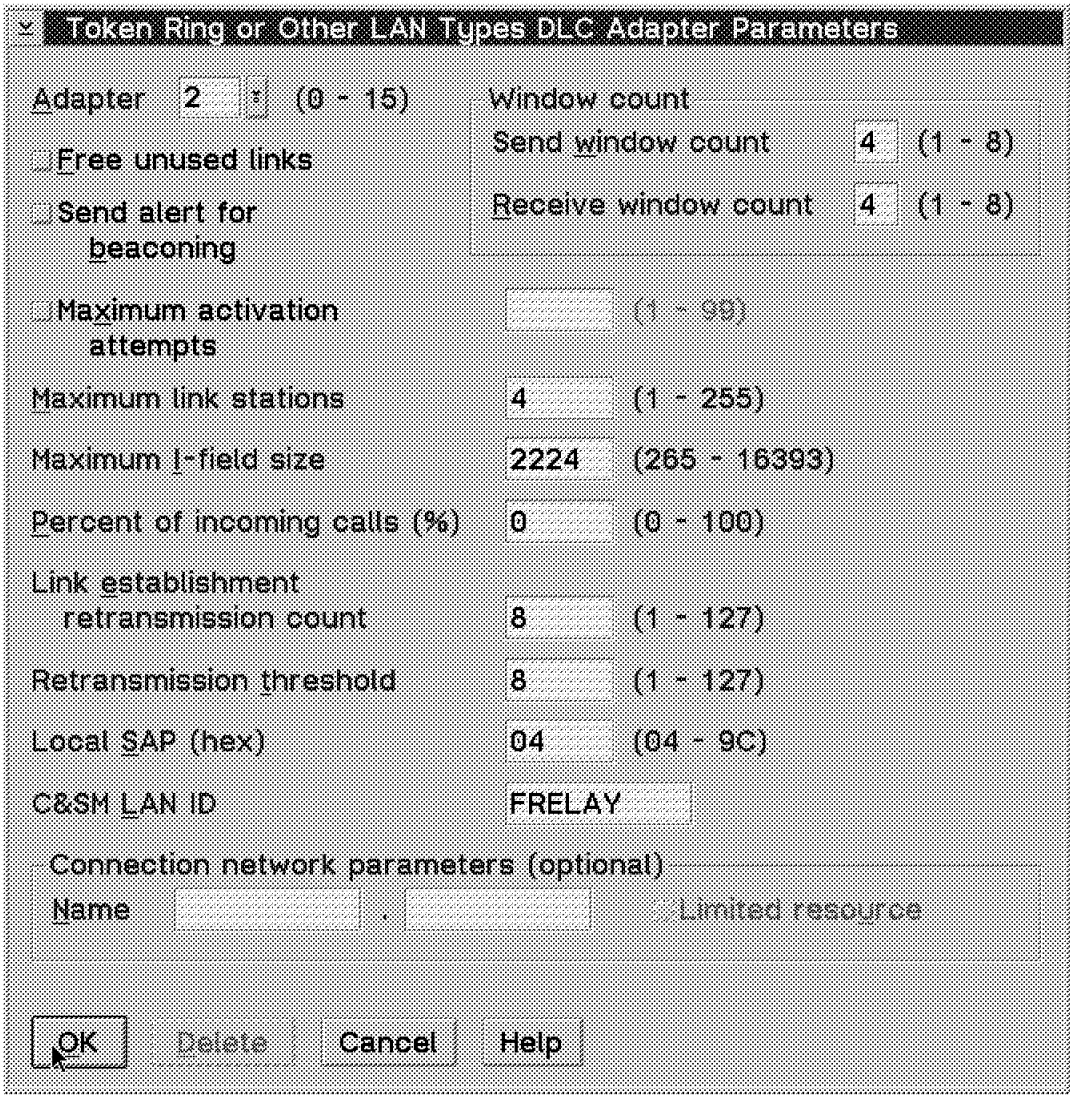


Figure 58. Token-Ring DLC Parameters - Frame Relay Adapter

Also we can review the local node characteristics of the Communications Server machine.

Select SNA local node characteristics in the Profile List and review the configuration as shown in Figure 59 on page 83.

You can also modify the local node ID for documentation purposes and to match the node name. It is not a required step but it helps to make the definitions more clear.

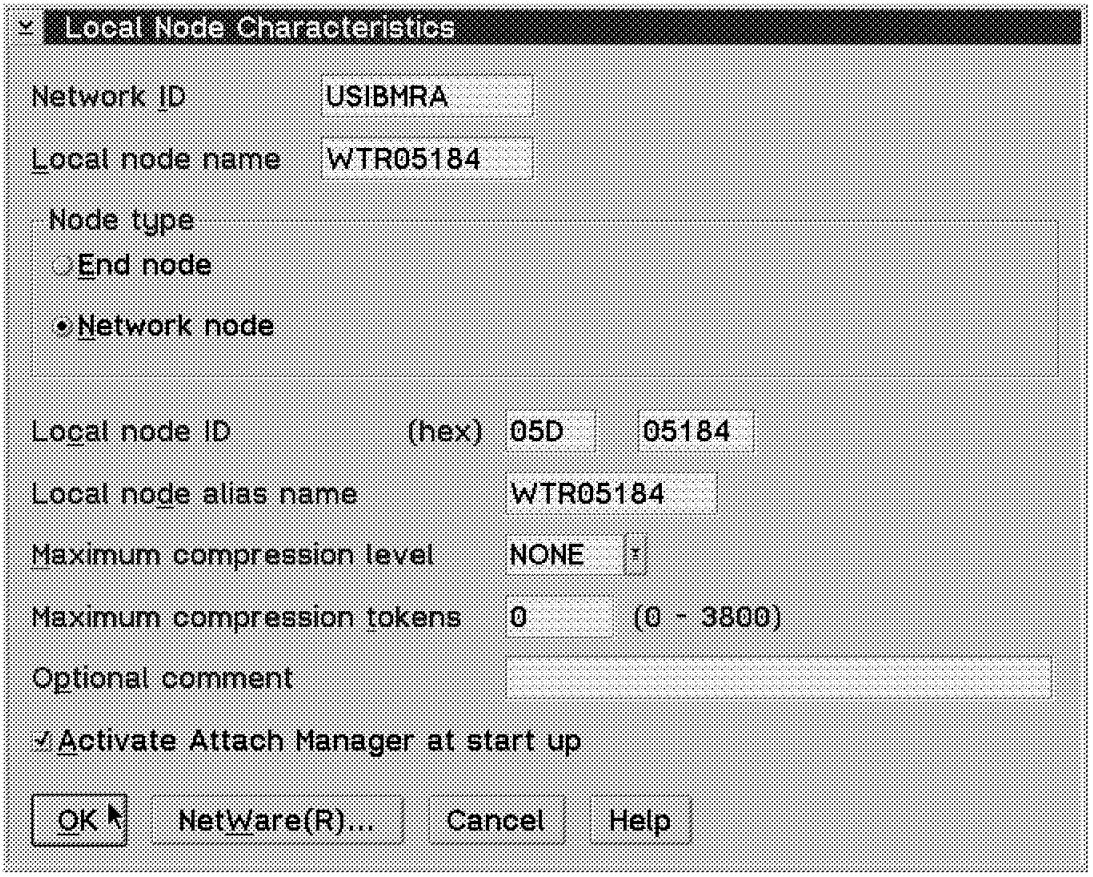


Figure 59. Local Node Characteristics

Next we want to add the connections that we are going to use for the SNA connectivity.

Select the **To network node** radio button and click on **Create** to define a connection to a network node as shown in Figure 60 on page 84.

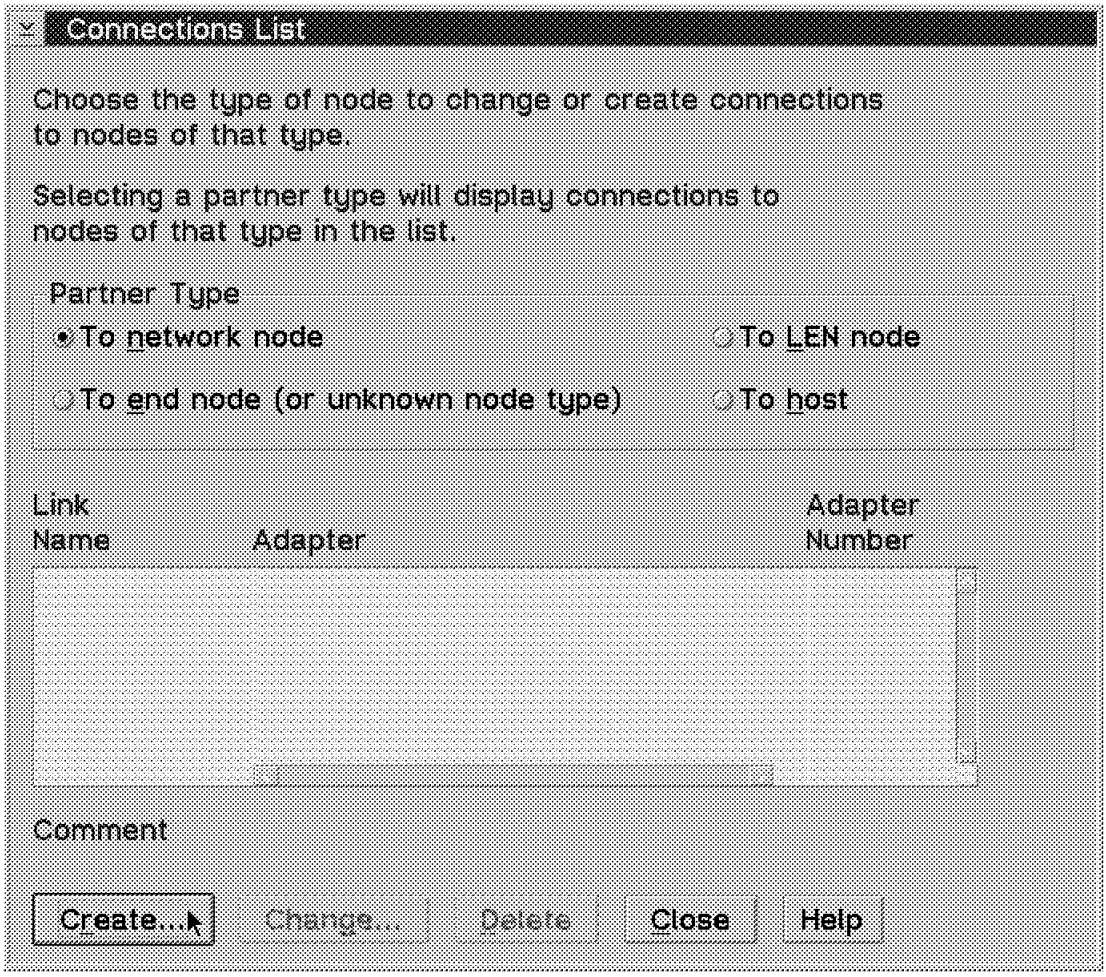


Figure 60. Connections List

Now we need two connections, both pointing to WTR5303.

The first is the main connection that uses the token-ring adapter (adapter number Zero). The following three figures show the panels that we use.

The following are a few recommendations:

- Remember to use the correct adapter number (Zero).

  Each machine that we configure can have a different assignment between the adapter numbers and the protocols that we use.

  You must review the MPTS configuration in order to make sure that you are using the correct number.

  Also review the help in MPTS for the Adapter Assignment in the parameters for the token-ring adapter. There is a value of UAA1 or UAA2 that can help you to determine the correct number of an adapter if you have more than one token-ring adapter in a machine (for example, WTR05184, WTR05303 and WTR05217).

  If you are not sure about the assignments, use the LANTRAN.LOG output to visually verify the address used by the adapter.

- Make the Link Name the name of the CPNAME or local node name of the machines that you want to connect (from WTR05184 we use WTR5303 as the

Link Name) in order to easily remember the correct link that we are using. This can help you in the problem determination steps.

- Use a figure or a table that shows the MAC addresses of each machine for each adapter, so you are not confused about the correct MAC address.
- Use the same address format to the MAC addresses of all the machines in the same LAN segment. Remember that the token-ring format and the Ethernet format are byte reverse to each other (for example, 4000 0000 0001 in one corresponds to 0200 0000 0080).
- Normally you do not have to change the default remote SAP address of X'04'.
- Active at Startup is useful in a server machine. If it is not active, someone has to activate the link.
- We are adding HPR for nondisrupting backup. This is because we are also going to define a backup link.

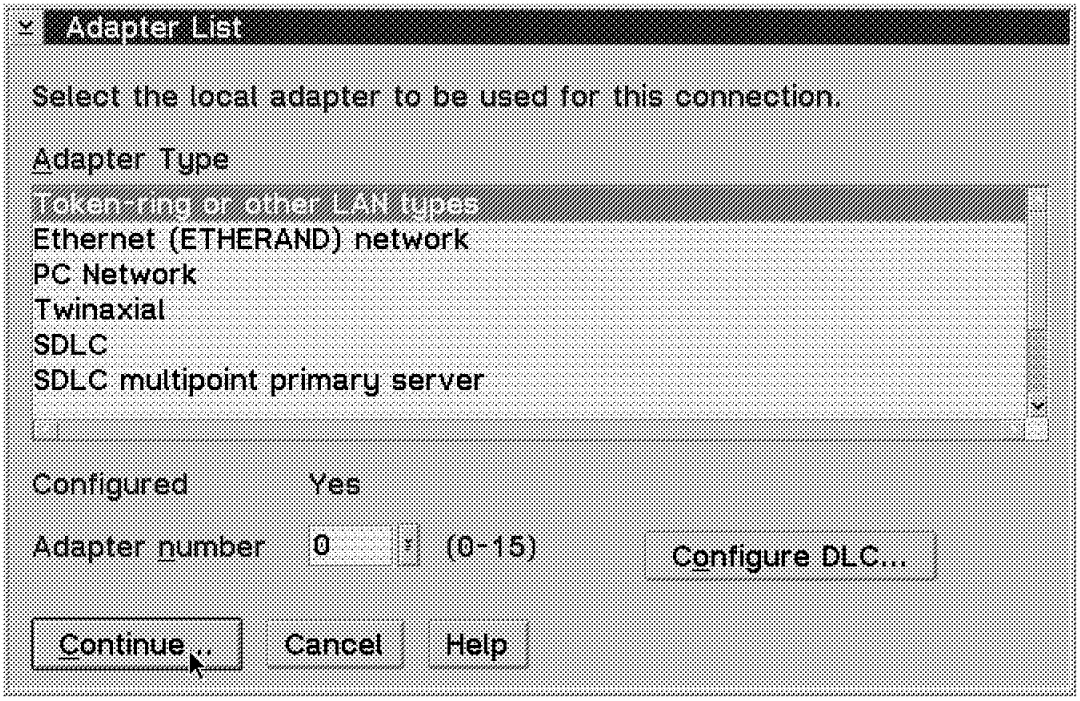


Figure 61. Adapter List

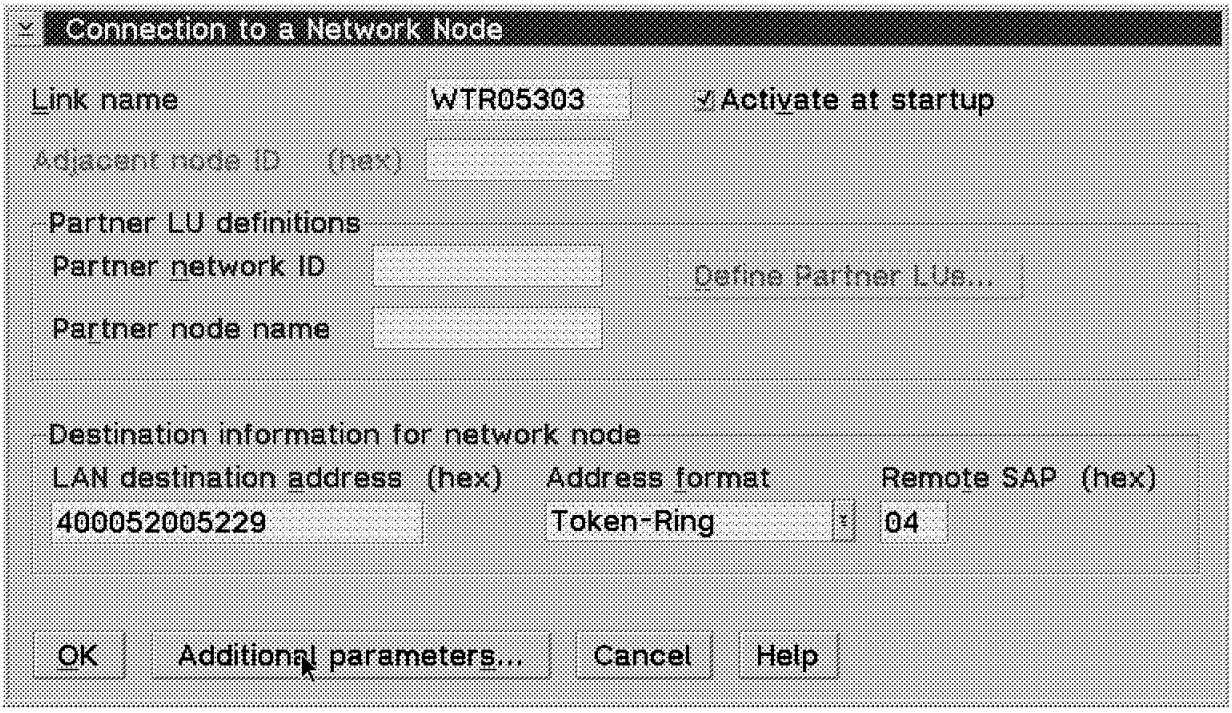


Figure 62. Connection to a Network Node

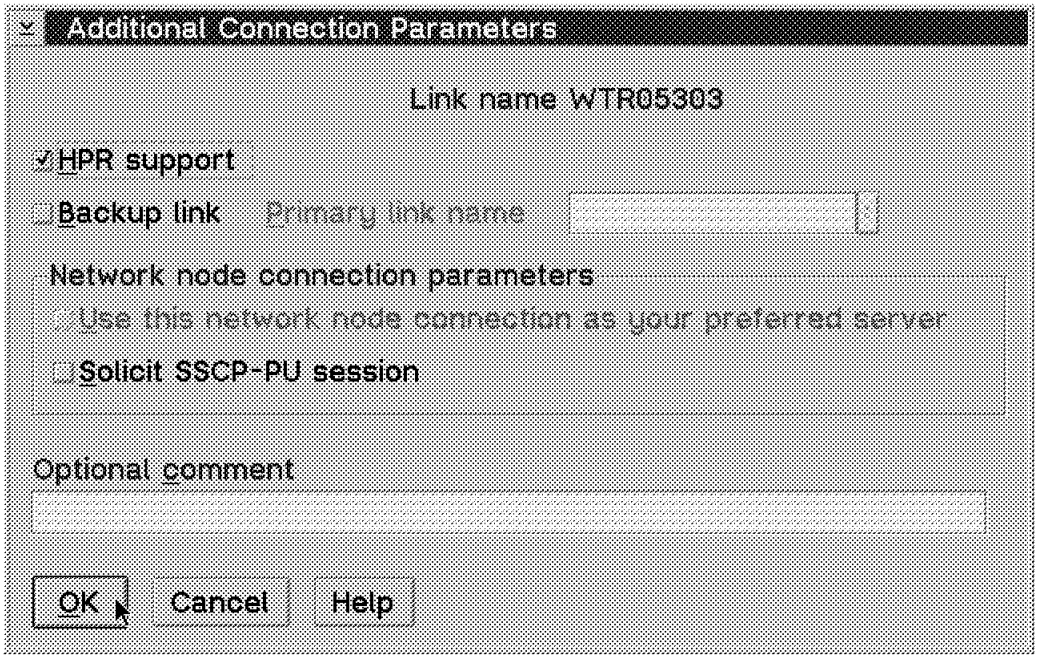


Figure 63. Additional Connection Parameters

The second is the backup connection that uses the frame relay adapter (adapter number two).

In this case the name for the link is WACFR, which also uses HPR capabilities and is defined as a backup link of the link with the name WTR05303.

In this configuration, we are using the routed frame format as explained in the frame relay section in Chapter 11, “Frame Relay Support” on page 245.

The following three figures show the panels that we used.

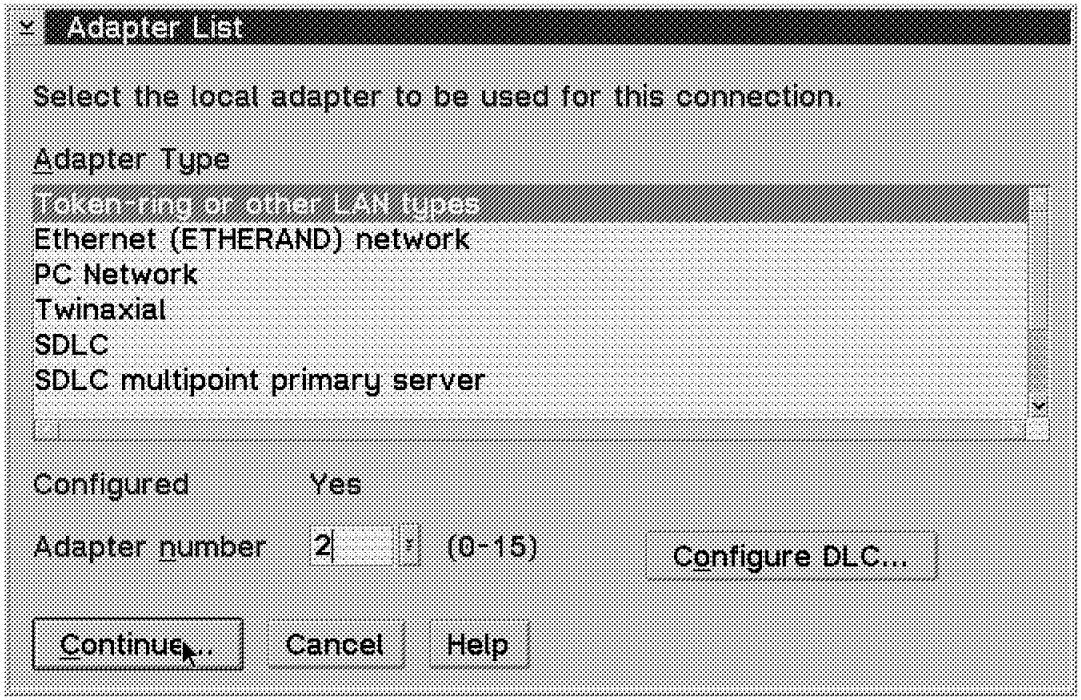


Figure 64. Adapter List

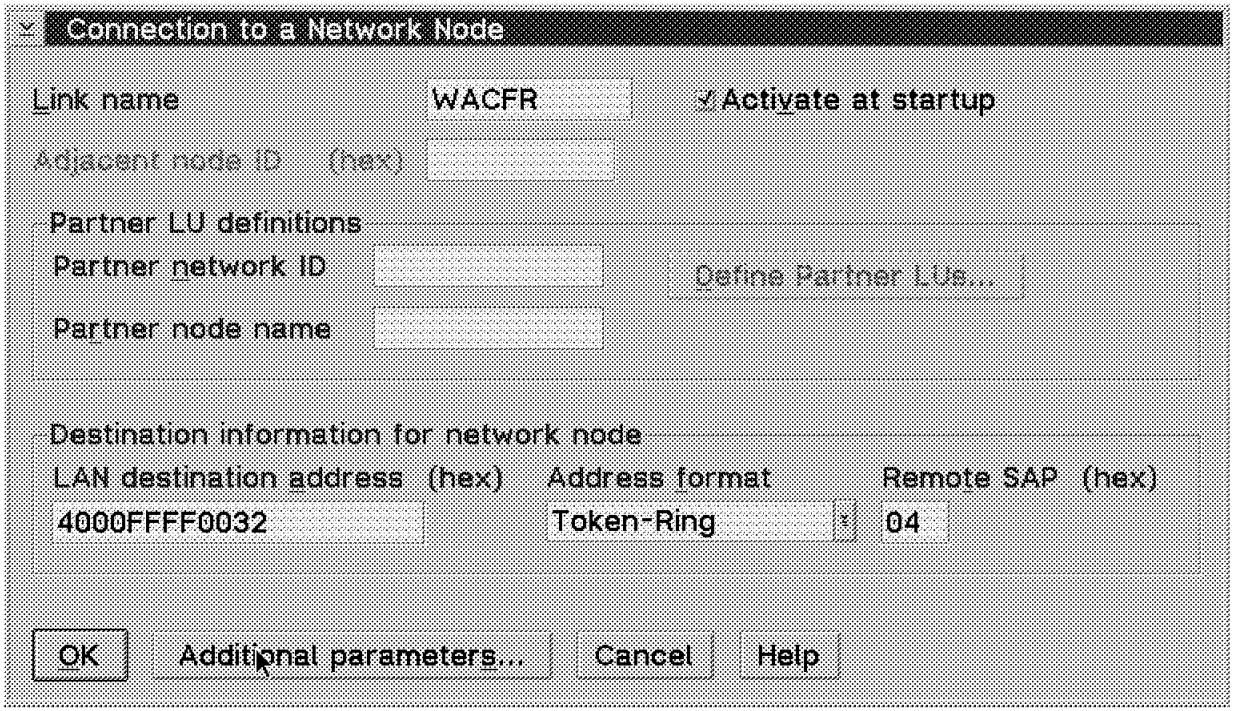


Figure 65. Connection to a Network Node

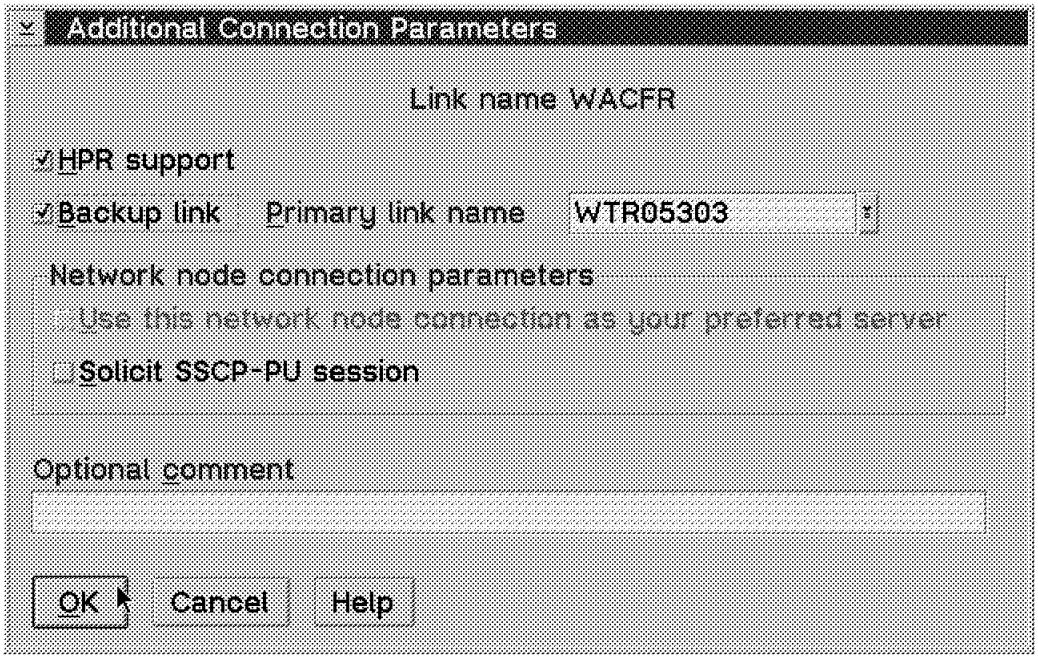


Figure 66. Additional Connection Parameters

Machines WTR05184 and WTR05217 almost have the same SNA configuration. WTR05217 does not use a frame relay backup link and obviously the SNA local node names are different.

Machines WTR05600 and WTR05700 have very similar configurations to that of WTR05303 and WTR05142 respectively. This is because both are remote IP networks that default to the parallel gateways as routers.

4.3.3.2 Configure MPTS

You must also configure Multiprotocol Transport Services (MPTS). To configure MPTS, select **LAN adapters and protocols** and **TCP/IP configuration**.

Important Note

For Sockets over SNA support, you must use the level of Multiprotocol Transport Services (MPTS) shipped with Communications Server or a later version.

Here are two different options:

1. You have the TCP/IP product.
2. You have only the TCP/IP stack provided by MPTS.

In both cases you need the TCP/IP stack that is provided by the MPTS product for OS/2, so you need to put the TCP/IP protocol in the corresponding adapter.

As shown in Figure 67 on page 89, you only need to add the protocol support for TCP/IP in the adapter that you are going to use for the TCP/IP traffic.

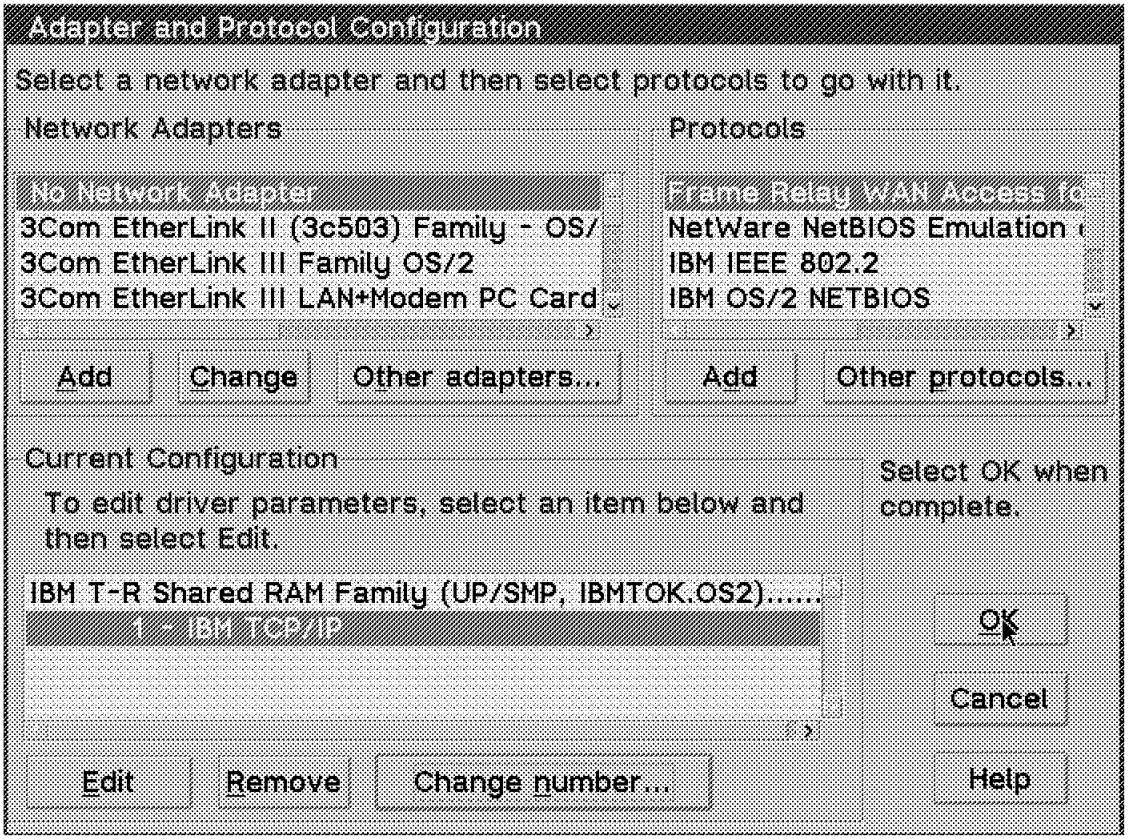


Figure 67. Example of TCP/IP Protocol Support for a LAN Adapter

If you do not have the TCP/IP product, then the TCP/IP configuration is done in the MPTS configuration panels.

Figure 68 shows you how to access this configuration panel.

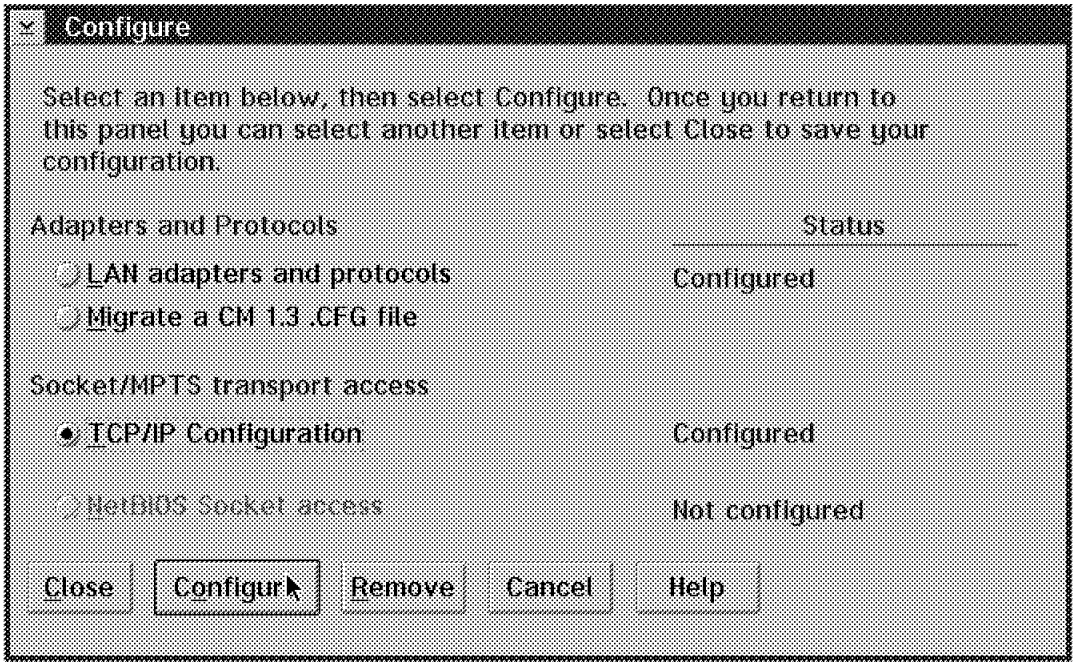


Figure 68. TCP/IP Protocol Configuration Within MPTS Access

Then you must select the option to configure the adapters, as shown in Figure 69 on page 90.

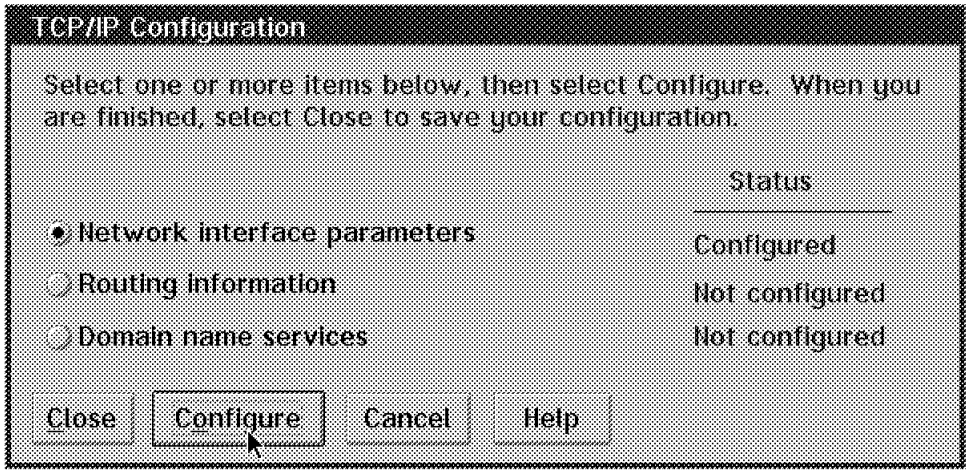


Figure 69. TCP/IP Protocol Configuration Within MPTS Configure Adapter

Figure 70 shows you the parameters that you need to configure. Mainly, you only need to define the LAN adapter that is going to use the TCP/IP protocol stack by making it available and defining the IP address that will be used.

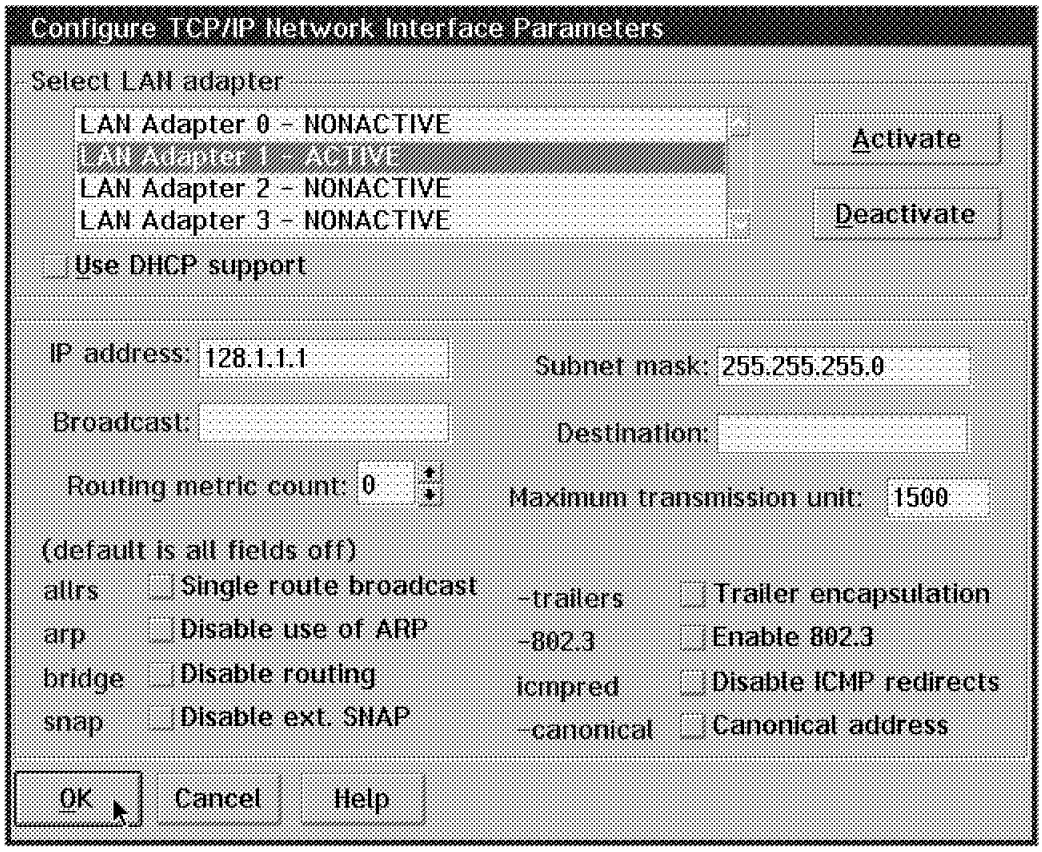


Figure 70. TCP/IP Protocol Configuration Within MPTS Details

You do not need to define any route, because it is going to be made in Communications Server.

Here you can define the TCP/IP services that you want to be auto-started, your host name for the TCP/IP applications and other things that we needed for your TCP/IP applications to run.

If you have the TCP/IP product, the previous TCP/IP configuration must be done with the TCP/IP configuration interface. You can also make these definitions by directly editing the corresponding TCP/IP configuration files, but it is strongly recommended that you use the configuration interface. It is located in the TCP/IP folder or can be invoked at the OS/2 command prompt with the TCPCFG command.

Figure 71 shows you the main panel that must be configured in the TCPCFG facility. The previous consideration about routes and applications also applies.

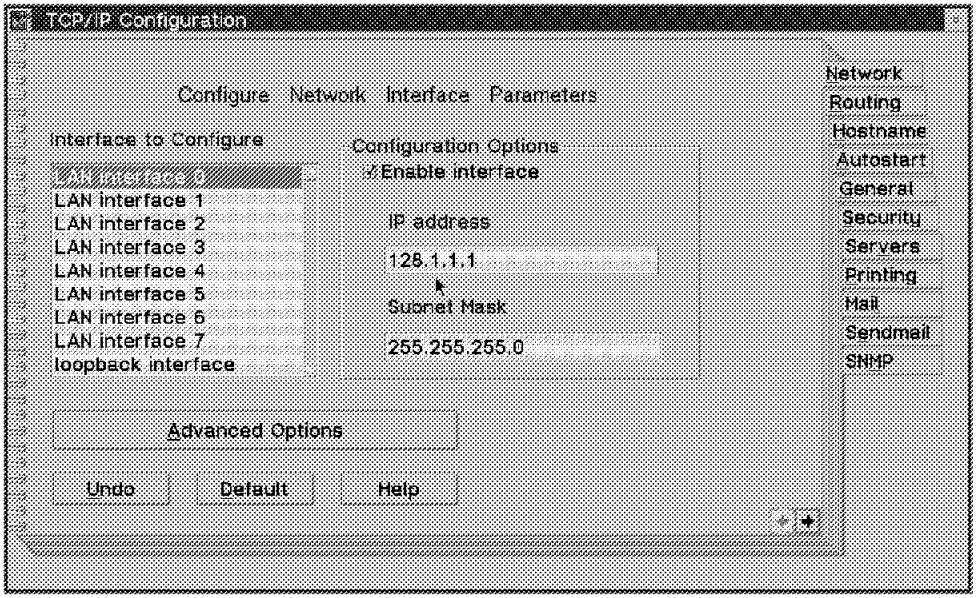


Figure 71. TCP/IP Protocol Configuration with TCPCFG Interface

4.3.4 Configuration Extract

For those of you that already know the SNA and TCP/IP world and how to maneuver in the Communications Server configuration panels, we have an extract of the configuration definitions that we made for this scenario. Only the TCP/IP and Sockets over SNA definitions are shown, so remember to add your SNA and MPTS definitions.

Refer to Table 1 on page 62 for the address assignments.

4.3.4.1 Definitions for Native TCP/IP Node WTR05200

Use the TCPCFG utility or the IFCONFIG command:

```
ifconfig lan0 128.1.1.9 netmask 255.255.255.0
```

Note

This node must be running the TCP/IP routed or gated program to receive updates from the Sockets over SNA parallel gateway via RIP.

4.3.4.2 Definitions for Sockets over SNA Parallel Gateways WTR05184 and WTR05217

In this section, we show you the configuration values for the Sockets over SNA parallel gateways.

- For IP addresses:
  - SNA uses the Sockets over SNA local parameters configuration.

Table 2. SNA Local Parameters Configuration

Machine Name	IP address	IP subnet mask
WTR05184	192.168.221.1	255.255.255.0
WTR05217	192.168.221.3	255.255.255.0

Table 3. SNA Generates

Machine Name	ifconfig command
WTR05184	ifconfig sna0 **192.168.221.1** netmask **255.255.255.0**
WTR05217	ifconfig sna0 **192.168.221.3** netmask **255.255.255.0**

  - TCP/IP uses the TCPCFG utility or the IFCONFIG command.

Table 4. TCP/IP IFCONFIG Command

Machine Name	ifconfig command
WTR05184	ifconfig lan0 **128.1.1.1** netmask **255.255.255.0**
WTR05217	ifconfig lan0 **128.1.1.3** netmask **255.255.255.0**

- Enable algorithm mapping.

  Use the Sockets over SNA IP address to LU mapping configuration. Use the same configuration for both machines.

Table 5. Sockets over SNA IP Address to LU Mapping

Parameter	Value
IP Address	192.168.221.0
Address Mask	255.255.255.0
Network ID	USIBMRA
LU template	IPSNA

  Then SNA generates the following command:

```
sxmap add sna 192.168.221.0 255.255.255.0 USIBMRA IPSNA
```

- Route to network 192.168.222.

  Both machines define the same route using the Sockets over SNA Route parameters configuration window, as follows:

Table 6 (Page 1 of 2). Sockets over SNA Route Parameter

Parameter	Value
Route Type	Subnet

Table 6 (Page 2 of 2). Sockets over SNA Route Parameter	
Parameter	**Value**
Destination Address	192.168.222.0
Router Address	192.168.221.2
Metric	1
Note: If you add or delete routes for a parallel gateway using the TCP/IP route command, these routes cannot be broadcasted to native TCP/IP nodes. See the Sockets over SNA enhancements.	

- Define each gateway as a parallel gateway.

  Both machines must be defined using the Sockets over SNA backup and load balancing configuration window.

Table 7. Sockets over SNA Backup and Load Balancing	
SNA IP Address	**LAN IP Address**
192.168.221.1	128.1.1.1
192.168.221.3	128.1.1.3

Note

The RIP option is automatically selected when you select the parallel gateway option. This is because there must be a way to inform the IP machines that there is a change in the routes when a change occurs or a gateway goes down and the backup can be used.

4.3.4.3 Definition for Sockets over SNA Gateway WTR05303 and WTR05600

This has a very similar configuration to the parallel gateways:

- Define the IP addresses and interfaces for SNA and TCP/IP.

  In SNA use the Sockets over SNA local parameters that generates the following:

  For WTR05303:

  ```
  ifconfig sna0 192.168.221.2 netmask 255.255.255.0
  ```

  For WTR05600:

  ```
  ifconfig sna0 192.168.221.4 netmask 255.255.255.0
  ```

  And use the TCP/IP command or TCPCFG tool to define:

  For WTR05303:

  ```
  ifconfig lan0 192.168.222.2 netmask 255.255.255.0
  ```

  For WTR05600:

  ```
  ifconfig lan0 192.168.223.4 netmask 255.255.255.0
  ```

- Enable the algorithm mapping exactly the same way as for the parallel gateways.
- Define the default routes to the parallel gateways in both machines.

Table 8. Sockets over SNA Route Parameter		
Parameter	**Value 1st route**	**Value 2nd route**
Route Type	Default	Default
Destination Address	All	All
Router Address	192.168.221.1	192.168.221.3
Metric	1	1

4.3.4.4 Definitions for Native TCP/IP Nodes WTR05142 and WTR05700

Use the TCPCFG utility or the IFCONFIG command to define these nodes:

For WTR05141, use:

```
ifconfig lan0 192.168.222.8 netmask 255.255.255.0

route add default 192.168.222.2 1 netmask 255.255.255.0
```

This machine has a Web server running; it is configured to be a proxy server.

For WTR05700, use:

```
ifconfig lan0 192.168.223.7 netmask 255.255.255.0

route add default 192.168.223.4 1 netmask 255.255.255.0
```

4.3.5 Capabilities of the Sample Configuration

We connect to an IP network using the Sockets over SNA gateway function.

We can run WebExplorer from 128.1.1 to our Web server at WTR05142.

Since WTR05142 is a Web proxy server no matter what IP address you use inside your network, if this machine has access to Internet, any machine that points its WebExplorer to this machine can also have access to the WWW. Now you can use your SNA network to give access to the Internet for any remote site. This machine should be connected to a firewall in order to guarantee the security of our intranet.

We also can run WebExplorer from any of the Sockets over SNA gateway machines. Internally they use the Sockets over SNA access node to arrive up to WTR05142 via the WTR05303 gateway.

Moreover we can use any machine in the SNA network (only need the OS/2 Access Feature, a way to access the Internet and a Web server SW) that we define as a Sockets over SNA access node to provide this access to Internet.

4.4 Troubleshooting

There are some basic troubleshooting steps that can help you in the process of problem determination of a Sockets over SNA configuration. Before any attempt to fix the problem, you should:

- Look for any errors messages that describe the failure.
- Check your cables and connections.

- A diagram of the network indicating SNA names, IP address and MAC address is always useful.

4.4.1 Troubleshooting Steps

Here are the steps that you can follow in order the isolate a Sockets over SNA problem.

4.4.1.1 Initialization Problems

Review the Error Message Log using the Message Log Formatter to see if there are any error messages in the Sockets over SNA initialization.

4.4.1.2 Connectivity Problems

For connectivity problems, you may want to proceed as follows:

1. Use the APING command to verify the SNA connectivity. The APING command verifies that you can set up an LU 6.2 session and send data between the local and the remote nodes. The APING applet is part of the productivity aids and must be installed to have it available in your workstation. Here are two examples:

```
[C:\]aping usibmra.wtr05303
IBM APING version 2.43.3c  APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Allocate duration:                          218 ms
Program startup and Confirm duration:      1093 ms

Connected to a partner running on: OS/2
        Duration        Data Sent        Data Rate        Data Rate
        (msec)          (bytes)          (KB/s)           (Mb/s)
        --------        ---------        ---------        ---------
               0              200

               0              200

Totals:        0              400
Duration statistics:   Min = 0   Ave = 0   Max = 0
```

Figure 72. Example of a Successful APING

Next, we show you a display for an unsuccessful APING.

```
[C:\]aping usibmra.wtr05700
IBM APING version 2.43.3c  APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Problem detected by the client.
APP0201: An allocation failure occurred.

         OS/2 Sense Data: 08400007
```

Figure 73. Example of a Failed APING

This example shows an SNA sense code of 08400007. Refer to the *Problem Determination Guide* for more information about sense codes.

There is a file named PDGUIDE.INF, which can be viewed with the VIEW OS/2 command. It is located in \CMLIB\EN_US\BOOK and gives good information about sense codes and other codes.

For example, 08400007 says that the resource was not found. This LU is not reachable by the SNA network.

For detailed information and a complete list of SNA sense codes, refer to the *SNA Formats Manual*.

If APING fails, check to see if the link is ACTIVE to this node or to the network node that must give you access to it. Issue the CMLINKS command and check the link status.

2. Use the IFCONFIG command to check the network interface status. Make sure the connection is active (UP) and visually verify that the IP address and netmask are correct.

```
[C:\]ifconfig lan1
lan1: flags=3863<UP,BROADCAST,NOTRAILERS,RUNNING,BRIDGE,SNAP>
         inet 128.1.1.1 netmask xffffff00 broadcast 128.1.1.255

[C:\]ifconfig sna0
sna0: flags=1<UP>
         inet 192.168.221.1 netmask xffffff00
```

Figure 74. Example of a IFCONFIG Command for LAN0 and SNA0

3. Use the SXMAP command to verify the address mapping.

   Use it to verify that the entries in the IP-LU mapping table are what you expect.

   There are two useful options:

   - Use the SXMAP GET SNA to obtain the entire IP-LU mapping table.

```
[D:\]sxmap get sna
Address         Mask            SNA Network     LU Template
 ------------------------------------------------------------
192.168.221.0   FFFFFF00        USIBMRA         IPSNA
```

Figure 75. Example of a SXMAP GET SNA Command

   - Use the SXMAP QMAP to obtain a particular IP-LU mapping.

```
[D:\]sxmap qmap 192.168.221.2
IP address maps to SNA address: USIBMRA.IPSNA002.
```

Figure 76. Example of a SXMAP QMAP Command

   These commands must give the same output in any gateway machine that you want to connect in a Sockets over SNA environment.

4. Use the NETSTAT command to verify that the routes exists.

NETSTAT -R gives us the routing table information so we can view if the route exists, if it is using the correct router and the correct netmask and what adapter it is using.

```
[D:\]netstat -r

destination       router          netmask refcnt   use  flags  snmp intrf
                                                             metric
   9.24.104.0 192.168.221.8   255.255.255.0   0       3  UG       0  sna0
    127.0.0.0     127.0.0.2       255.0.0.0   0     231  U        0  gw0
    128.1.1.0     128.1.1.1   255.255.255.0   0       0  U        0  lan1
    128.1.1.1     128.1.1.1 255.255.255.255   1       4  UH       0  lan1
    128.1.1.3     128.1.1.1 255.255.255.255   2    3256  UH       0  lan1
  128.1.1.255     128.1.1.1 255.255.255.255   0       0  UH       0  lan1
    130.1.1.0     130.1.1.1   255.255.255.0   0       0  U        0  lan2
  130.1.1.255     130.1.1.1 255.255.255.255   0       0  UH       0  lan2
192.168.221.0 192.168.221.1   255.255.255.0   6   40152  U        0  sna0
192.168.222.0 192.168.221.2   255.255.255.0   0       0  UG       0  sna0
192.168.223.0     128.1.1.3   255.255.255.0   0       0  UG       0  lan1
```

Figure 77. Example of a NETSTAT -R Command from WTR05184

```
m310[C:\]netstat -r

destination       router          netmask refcnt   use  flags  snmp intrf
                                                             metric
    127.0.0.0     127.0.0.2       255.0.0.0   0       0  U        0  gw0
    128.1.1.0     128.1.1.3   255.255.255.0   0       0  U        0  lan0
    128.1.1.1     128.1.1.3 255.255.255.255   2      42  UH       0  lan0
    128.1.1.3     128.1.1.3 255.255.255.255   1       2  UH       0  lan0
  128.1.1.255     128.1.1.3 255.255.255.255   0       0  UH       0  lan0
    130.1.0.0     128.1.1.1     255.255.0.0   0       0  UG       0  lan0
192.168.221.0 192.168.221.3   255.255.255.0   3      16  U        0  sna0
192.168.222.0     128.1.1.1   255.255.255.0   0       0  UG       0  lan0
192.168.223.0 192.168.221.4   255.255.255.0   0      26  UG       0  sna0
```

Figure 78. Example of a NETSTAT -R Command from WTR05217

5. Use the PING command to verify IP connectivity.

   With the previous checks verified, you are now successfully running the PING command, with an output like the following:

```
[D:\]ping 192.168.221.2
PING 192.168.221.2: 56 data bytes
64 bytes from 192.168.221.2: icmp_seq=0. time=30. ms
64 bytes from 192.168.221.2: icmp_seq=1. time=0. ms

----192.168.221.2 PING Statistics----
2 packets transmitted, 2 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/15/30
```

Figure 79. Example of a PING Command

The PING command is like the APING command, but is used for the IP network. Once you get successful packets sent, you have the IP connectivity

working. Review the percentage of lost packets and the time in milliseconds (ms) they were last sent and received from the packet.

At this point you can say that the connectivity is working and any other error should be the responsibility of the application layer. In any case, always review the error messages and logs and also the GWSTAT output for any other information about the network.

4.4.2 Sockets over SNA Gateway and Parallel Gateway Tools

There are three utilities apart from those previously mentioned that can be useful for problem determination issues. They are:

- GWSTAT
- PGWSPLIT
- PGWSTAT

4.4.2.1 GWSTAT

The GWSTAT utility displays the status of the connections established through a Sockets over SNA gateway.

It can display the following details:

1. Status of all the connections

```
[D:\cmlib]gwstat
Current # of gateway entries:  9
Maximum # of gateway entries:  9
Total # of gateway entries:    21

Current # of gateway threads:  2
Maximum # of gateway threads:  5
Total # of gateway threads:    21

Maximum gateway chain length:  1

UDP bytes sent native to MPTN: 117362
UDP bytes sent MPTN to native: 920
TCP bytes sent native to MPTN: 8584
TCP bytes sent MPTN to native: 82793

Gateway entry limit:           2000
Connections refused:           0
Datagrams dropped:             0

Raw input transfer queue counts (idle/busy/free/total): 2/0/0/2
```

Figure 80. GWSTAT Output

Current numbers are for active connections, maximum numbers are obviously the maximum reached concurrently and total numbers are for the total number since Sockets over SNA was loaded (this always increases).

There is an option (-R) that allows you to reset the counters of sent bytes. This is useful for statistical purposes.

Also see that there is a counter for connections refused and datagrams dropped that works only for the Sockets over SNA part of the network and

goes up only when the maximum operating capacity (gateway entry limit) is reached.

2. Status of all the active connections

```
[D:\cmlib]gwstat -c

  ID    Proto       Native endpoint          MPTN endpoint  Flags
-------------------------------------------------------------------
130044   TCP        128.1.1.9/ 1040    192.168.221.8/   80   00e0
14004b   UDP        128.1.1.9/ 1026    192.168.221.8/   69   0010
0d005e   TCP        128.1.1.9/ 1034    192.168.221.8/   80   00e0
0e005f   TCP        128.1.1.9/ 1035    192.168.221.8/   80   00e0
0f0060   TCP        128.1.1.9/ 1036    192.168.221.8/   80   00e0
100061   TCP        128.1.1.9/ 1037    192.168.221.8/   80   00e0
110062   TCP        128.1.1.9/ 1038    192.168.221.8/   80   00e0
120063   TCP        128.1.1.9/ 1039    192.168.221.8/   80   00e0
15009b   UDP        128.1.1.9/ 1026    192.168.221.8/ 1177   0110
```

Figure 81. GWSTAT -C Output

All connections actually active are displayed.

See that there is an identification for the PORT that is used, as well as the protocol (for example, TCP or UDP). There is a list of well-known PORTs in \MPTN\ETC\SERVICES.

In this example we used a TFTP (UDP traffic) and a Web browser. TCP traffic sees the port number 80 at the destination. You can also see the Web server is running at 192.168.221.8 and in a Sockets over SNA machine that has SNA connectivity into the SNA network and TCP/IP to Internet using the same adapter.

3. Status of a single active connection

```
[D:\cmlib]gwstat -d 130044

ID:                              130044
Protocol:                           TCP
Native endpoint:       128.1.1.9/ 1040
MPTN endpoint:     192.168.221.8/   80
Idle time:                 145 seconds
Flags:                            00e0
    connection in CLOSEWAIT state
    have received MPTN SO_TERM
    have sent MPTN SO_TERM
Bytes sent native to MPTN:         615
Bytes sent MPTN to native:        2894

[D:\cmlib]gwstat -d 15009b

ID:                              15009b
Protocol:                           UDP
Native endpoint:       128.1.1.9/ 1026
MPTN endpoint:     192.168.221.8/ 1177
Idle time:                 110 seconds
Flags:                            0110
    inbound native connection exists
    outbound native connection exists
Bytes sent native to MPTN:       58655
Bytes sent MPTN to native:         460
```

Figure 82. GWSTAT -D Output

Remember that depending on the kind of connection, the applications have special ways to maintain or not maintain the connection actively. In the previous case the TCP connections for the WebExplorer are in CLOSEWAIT, so after a while they disappear from the GWSTAT outputs.

MPTN SO_TERM means that the sender or the gateway will not send any more data on this connection ID.

The connection for TFTP, which uses UDP, goes away after it reaches the timeout value for datagram connections that are inactive.

Note that the byte counters are only for TCP and UDP connections. There is no counter for PING or route discovery bytes because both use ICMP.

As a comparison the following figure shows what happens with a TCP Telnet connection.

```
ATELNET [C:\]gwstat -c

  ID    Proto       Native endpoint          MPTN endpoint  Flags
-------------------------------------------------------------------
1e0012   TCP        128.1.1.9/ 1049    192.168.221.8/   23   0114

ATELNET [C:\]gwstat -d 1e0012

ID:                              1e0012
Protocol:                           TCP
Native endpoint:       128.1.1.9/ 1049
MPTN endpoint:     192.168.221.8/   23
Idle time:                 800 seconds
Flags:                            0114
    inbound MPTN connection exists
    inbound native connection exists
    outbound native connection exists
Bytes sent native to MPTN:          47
Bytes sent MPTN to native:         154
```

Figure 83. GWSTAT -D Output

The connection remains active because a connection exists until the client or the server closes the application or if there is some problem in the network. Some Telnets also have a keepalive mechanism in order to make sure that the partner is already alive.

4.4.2.2 PGWSPLIT

This utility is used to display the subnetworks that are serviced by each parallel gateway.

As you can see this command runs instead of the response file, so the output is based only on the information provided in it. No connection is necessary to produce the output. It only applies to the same mechanisms that Communications Server uses to determine what gateway serves what subnetwork.

The following is an example of the output of this command in any of the two parallel gateway machines.

```
[D:\cmlib]pgwsplit sc04.rsp
Gateway 128.1.1.1/192.168.221.1 routes for subnets:
        192.168.222.0

Gateway 128.1.1.3/192.168.221.3 routes for subnets:
        192.168.223.0
```

Figure 84. PGWSPLIT Output for Sockets over SNA Parallel Gateway Machines

4.4.2.3 PGWSTAT

This utility is used to display the status information of the parallel gateways that participate in a backup and load balancing configuration.

This information is based on the actual connectivity of the machine that runs the command.

The following is an example of the output of this command.

```
[D:\cmlib]pgwstat

There are 2 parallel gateways

Gateway 192.168.221.1/128.1.1.1 is up and routes for:
        subnet 192.168.222.0 through gateway 192.168.221.2

Gateway 192.168.221.3/128.1.1.3 is up and routes for:
        subnet 192.168.223.0 through gateway 192.168.221.4
```

Figure 85. PGWSTAT Output for Sockets over SNA Parallel Gateway Machines

4.4.3 Traces

Sockets over SNA is an APPC application, so that the normal way for debugging APPC session problems must be used.

Use the CMTRACE facility when looking for problems in the SNA network. Also add the DLC ANYNET (and if needed the EVENT ANYNET Internal) to the commonly used APPCs EVENTs, APPC APIs and the DLC you are using (such as token-ring and Ethernet).

Part 5. LAN Gateway

Chapter 5. Introducing the LAN Gateway

This chapter describes the LAN Gateway and how it can be used. It contains the following sections:

- 5.1, "What Does the LAN Gateway Do?" summarizes the functions provided by the LAN Gateway and contains network configuration examples.
- 5.2, "Some Basics about LAN Protocols" on page 108 provides some basics concerning PC LAN protocols that are handled by the LAN Gateway.
- 5.3, "How Does the LAN Gateway Work?" on page 112 shows the structure of a LAN Gateway node and explains how the LAN Gateway routes data.

5.1 What Does the LAN Gateway Do?

The LAN Gateway feature of Communications Server is one of IBM's AnyNet software offerings. AnyNet software enables application programs to communicate over different transport networks and across interconnected networks. Using AnyNet, you can reduce the number of transport networks and reduce operational complexity. These benefits are gained without modification to your existing application programs or hardware.

The LAN Gateway enables workstations, requesters, or servers located on different local area networks (LANs) to communicate across SNA or IP wide area networks (WANs). The LAN Gateway supports both Novell NetWare Internetwork Packet Exchange (IPX) and NetBIOS protocols across WANs. Each LAN attaches to the WAN through a LAN Gateway.

WAN sessions between two LAN Gateways use Advanced Program-to-Program Communication (APPC) or Transmission Control Protocol/Internet Protocol (TCP/IP). The LAN Gateways appear to the WAN as independent LU 6.2 application programs or TCP/IP application programs. The WAN connection between the LAN Gateways is either an LU 6.2 session or a TCP/IP stream.

The following network configurations illustrate how the LAN Gateway can be used:

- 5.1.1, "Connecting NetBIOS and IPX Applications over an SNA WAN"
- 5.1.2, "Connecting Socket Applications over an SNA WAN" on page 106
- 5.1.3, "Connecting NetBIOS and IPX Applications over an IP WAN" on page 107
- 5.1.4, "Running NetBIOS and IPX Applications on the LAN Gateway Using the Loopback Mode" on page 107

5.1.1 Connecting NetBIOS and IPX Applications over an SNA WAN

Figure 86 on page 106 shows how the LAN Gateway supports communication between LANs across an SNA WAN. In this example:

- LAN A and LAN B support both IPX and NetBIOS applications. Each LAN can be either a token-ring or an Ethernet LAN.
- Workstation A connects LAN A to the WAN and runs as an independent LU 6.2 application program. In this example, Workstation A is a LAN Gateway.

- Workstation B connects LAN B to the WAN and is shown as a LAN Gateway. Workstation B could be running the following products:
  - IBM AnyNet IPX over SNA Gateway (IPX traffic on SNA WANs only)
  - IBM LAN-to-LAN wide area network (LTLW) program (on SNA WANs only)
  - IBM Nways 2217 Multiprotocol Concentrator

Using an SNA WAN, NetBIOS and IPX applications running on Workstation 1 in LAN A and on Workstation 2 in LAN B can communicate. The LAN Gateway supports communications across both APPN and subarea WANs.

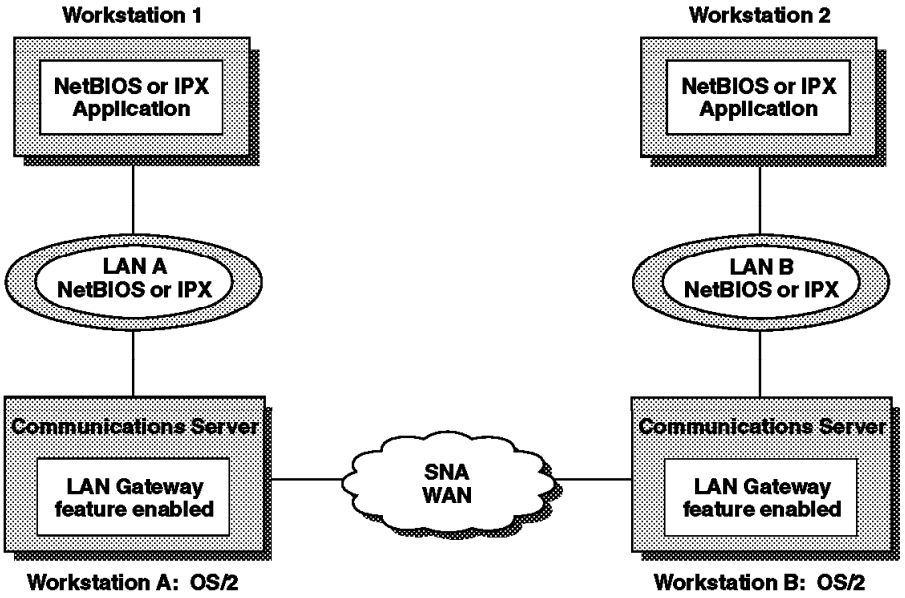


Figure 86. Configuration Using LAN Gateways to Support NetBIOS and IPX Communication across an SNA WAN

5.1.2 Connecting Socket Applications over an SNA WAN

As opposed to the IBM LAN-to-LAN wide area network program, the LAN Gateway no longer has the IP over SNA function. This functionality has been incorporated in the IBM Communications Server for OS/2 Warp Version 4.1 – AnyNet Sockets over SNA Gateway and is covered in detail in Chapter 4, “Sockets over SNA Access Node and Gateway” on page 47. It provides a higher level of availability than the LAN Gateway through the load balancing and backup mechanisms also treated in Chapter 4, “Sockets over SNA Access Node and Gateway” on page 47.

Both the IBM Communications Server for OS/2 Warp Version 4.1 – Sockets over SNA Gateway and the LAN Gateway can coexist in one machine and also be active at the same time.

5.1.3 Connecting NetBIOS and IPX Applications over an IP WAN

Figure 87 shows how the LAN Gateway supports communication between LAN workstations across an IP WAN. In this example:

- LAN A and LAN B support both IPX and NetBIOS applications.
- Each LAN can be either a token-ring or an Ethernet LAN.
- Workstation A, a LAN Gateway, connects LAN A to the WAN.
- Workstation B, a LAN Gateway, connects LAN B to the WAN.

Using an IP WAN, NetBIOS and IPX applications running on Workstation 1 in LAN A and on Workstation 2 in LAN B can communicate.

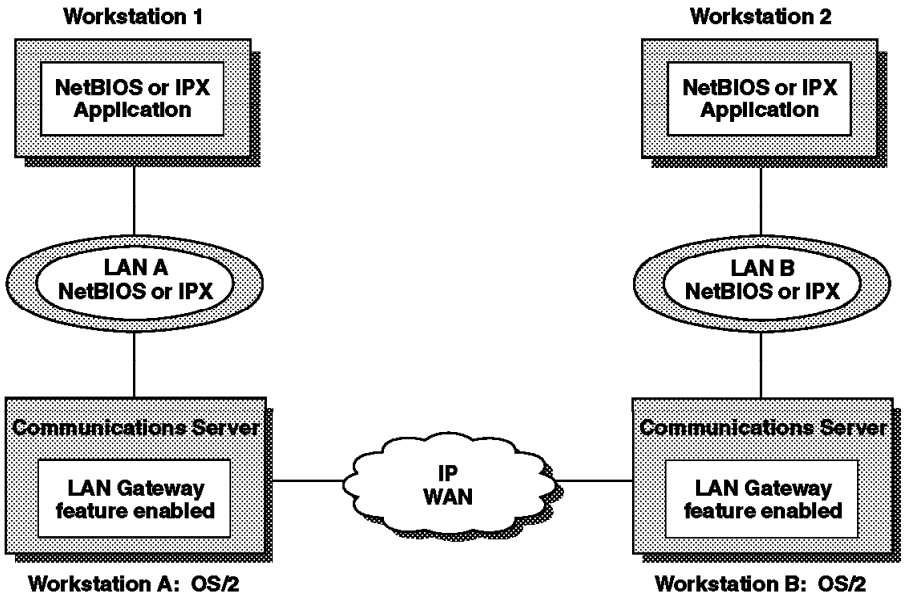


Figure 87. Configuration Using LAN Gateways to Support NetBIOS and IPX Communication across an IP WAN

5.1.4 Running NetBIOS and IPX Applications on the LAN Gateway Using the Loopback Mode

Figure 88 on page 108 shows how the LAN Gateway loopback driver can enable stand-alone workstations, such as laptop computers or non-LAN desktop workstations, to access an IP or SNA WAN using a modem instead of LAN hardware. The loopback driver simulates an IBM Token-Ring LAN adapter (including protocols supported by such an adapter) for OS/2.

In this example:

- LAN A and LAN B are connected over an SNA or IP WAN.
- Workstation A and Workstation B are LAN Gateways that have not enabled the loopback option.
- Workstation C is a LAN Gateway that has enabled the loopback option.

  Workstation C operates as an access node, allowing local application programs to communicate with application programs running on Workstation 1 in LAN A and on Workstation 2 in LAN B.

For information about installing the LAN Gateway loopback driver, see Appendix C, “Loopback Mode” on page 307.

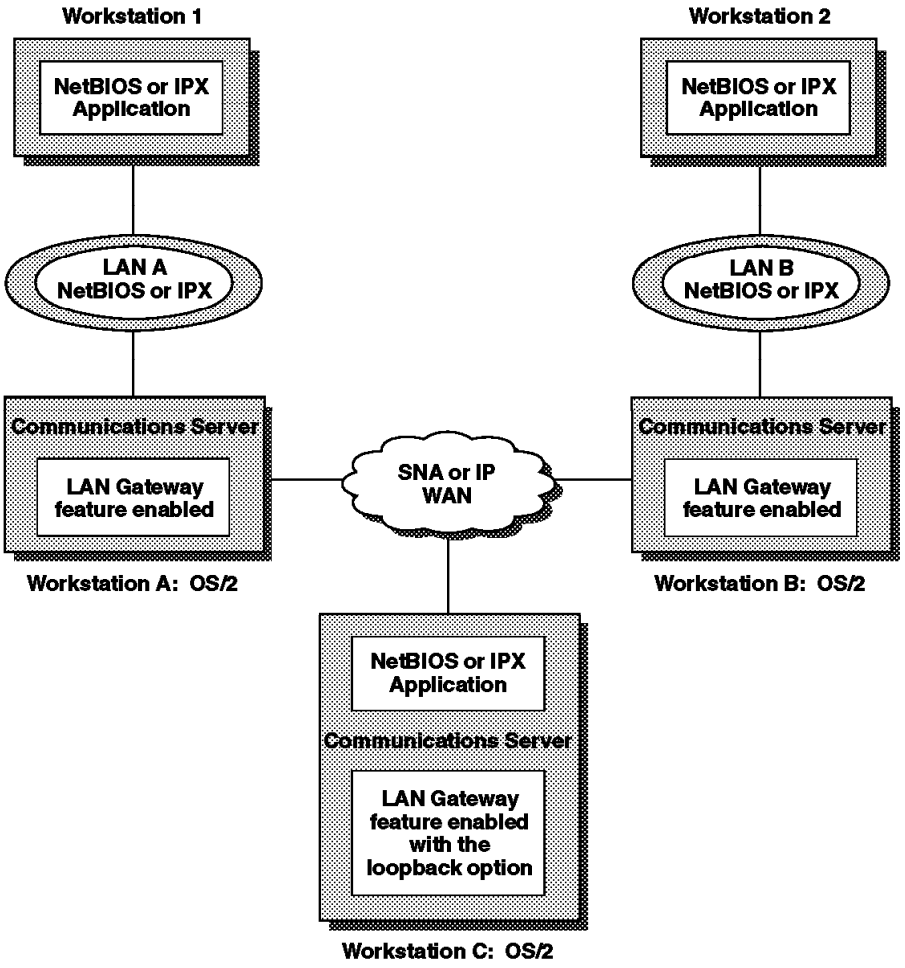


Figure 88. Configuration Using the Loopback Mode on the LAN Gateway Workstation

5.2 Some Basics about LAN Protocols

We begin with some background information about the LAN protocols that AnyNet LAN Gateway software can route. This is not intended to be a rigorous introduction to LAN protocols, so some simplifications have been made in the interest of not boring the reader.

Even this is a bit more detail than necessary, but many of these terms will come up again later in questions of how to configure the LAN Gateway. Thus, it seems like a good idea to present them first in context so they are more familiar later on.

The two kinds of physical LANs the AnyNet LAN Gateway supports are *Ethernet* and *token-ring*. These have many differences, but also have a great deal in common, and the LAN Gateway can treat them the same in most cases.

The hardware used to attach a computer to a LAN is a *LAN adapter*. Each of the above mentioned LAN adapters has a 6-byte address, called a *MAC address*,

which is used to send data to the adapter on the LAN. Adapters send data to one another in packages called *frames*.

A *LAN segment* consists of a group of adapters connected by a LAN in such a way that frames can be sent directly from one adapter to another. *Bridges* are devices that connect two or more LAN segments, passing frames from one LAN segment to another. LANs are divided into segments for several reasons, including physical limitations on segment sizes and the difficulty of managing very large segments.

Henceforth, when we use the term LAN, we mean any number of LAN segments that are connected by bridges. Note that sometimes *routers* are configured to do bridging for certain protocols. If a router is configured to bridge NetBIOS or IPX, then we would consider it to be a bridge for our purposes, even if it is routing other types of frames. LANs have the ability to send a frame to multiple destinations; this is called *broadcast*. Excessive broadcasts can use up the LAN′s bandwidth and cause heavy processing loads on all the machines on the LAN that have to look at each broadcast frame.

Typically, broadcasts are used either to disseminate information to many machines in a single message, or to locate a particular partner. Once a partner is located, traffic is sent directly to that partner′s MAC address.

The simplest frames used by Ethernet or token-ring to send data from one adapter to another consist of:

- Source MAC address
- Destination MAC address
- Frame length
- Data

5.2.1 The IEEE 802.2 Protocol

There is a standard format for the beginning of the data part of a LAN frame. This standard is called *IEEE 802.2*. The 802.2 standard allows multiple networking protocols (such as NetBIOS, IPX, TCP/IP, and SNA) to run simultaneously on the same LAN.

The most important part of the IEEE 802.2 header is a byte called the destination *service access point* (SAP). The SAP identifies to which networking protocol the rest of the data in the frame belongs. For example, NetBIOS uses X′F0, IP uses X′AA, SNA usually uses X′04, and IPX can use X′E0 or X′AA.

The SAP value X′AA is a catch-all SAP, and frames using SAP X′AA have another 5-byte header added, called the *SNAP header*, to identify the specific protocol being used. This is used, for example, to distinguish IP frames from IPX frames.

802.2 defines two types of LAN frames. The type 1 frame, better known as a *datagram*, is an individual frame sent between two adapters. 802.2 also provides the ability to set up a connection between two adapters. Frames sent on such a connection are called *type 2 frames*, and the connection itself may be called an *LLC 2 connection*. This connection provides error-detection, timeouts, retries, and detects the loss of connections. It supplies full-duplex communication. It is

used by SNA and by NetBIOS sessions (but not NetBIOS datagrams). Frames sent using a connection are more reliable than datagrams.

5.2.2 The IPX Protocol

Here are some of the frame formats used by IPX.

Table 9. Novell Frame Formats

SAP	Ethernet	Token-Ring
X′E0	ETHERNET_802.2	TOKEN-RING
X′AA	ETHERNET_SNAP	TOKEN-RING_SNAP
(non 802.2)	ETHERNET_II ETHERNET_802.3	

There are other frame formats on Ethernet, called ETHERNET_II and ETHERNET_802.3 that were introduced by Novell, which do not use the 802.2 format. Older versions of Novell software defaulted to this, but newer versions (3.12 and later) default to the more standard 802.2 frames.

Note: Non-802.2 frames are completely ignored by the LAN Gateway. Any frame types not mentioned here are also ignored.

In the AnyNet LAN Gateway documentation and user interface, frames using the X′AA SAP are typically referred to as SNAP frames; X′E0 frames are called 802.2 frames.

All IPX frames are datagrams; IPX does not use 802.2′s LLC 2 connection ability. To have a more reliable (session-oriented) protocol, Novell uses the Sequenced Packet Exchange (SPX) protocol.

IPX frames include:

- Source IPX address
- Destination IPX address
- Length
- IPX frame type
- IPX Data

IPX addresses have two parts: a four-byte *network number* and a six-byte *node address*. Most of the time, the node address is the MAC address of the machine′s adapter.

The network number is assigned by a network administrator to a LAN; each LAN must have a different network number. Furthermore, each IPX server has another network number assigned to it; all of the services inside a server act as if they are attached to an internal network, which therefore needs its own number.

The network number allows IPX routers (such as the LAN Gateway) to route IPX frames to the correct LAN based on the network number, without having to keep track of every adapter in the network.

Every 60 seconds each router on a LAN broadcasts *routing information protocol* (RIP) frames that list all of the network numbers that the router can reach. This

is used by IPX machines to determine where to send any frame that needs to go to a different network; the frame is sent to the router advertising the shortest route to that network.

IPX servers broadcast *Service Advertising Protocol* (SAP) frames every 60 seconds identifying what services they provide. IPX routers propagate the SAP information to other LANs so that every IPX machine in the network will ″see″ every server.

A Novell client will be unable to log into a server unless that server′s SAP information is available on the client′s LAN, and there is RIP information for routing to that server′s network.

The broadcast of RIP and SAP frames in large IPX networks is quite costly. The LAN Gateway is careful not to send RIP and SAP information across wide area networks except when the information has changed; it does *not* send copies of the RIP and SAP frames themselves. Nevertheless, in a large IPX network, each LAN Gateway may be putting a large number of RIP and SAP frames on its attached LAN every 60 seconds to advertise routes and services in other parts of the network.

5.2.3 The NetBIOS Protocol

A NetBIOS program running on a workstation is identified by one or more NetBIOS *names* (rather than addresses). A NetBIOS name is an arbitrary 16-byte value, although usually they are readable ASCII strings.

The NetBIOS protocol was not designed to be routed. There is no network identifier in the NetBIOS name to help with routing.

NetBIOS frames, such as 802.2 frames, come in two basic types, allowing NetBIOS applications to communicate either by sending datagrams or by establishing connections. The type 1, or datagram, frames have the format:

- Destination NetBIOS name
- Source NetBIOS name
- Data

Type 2 frames are sent on a connection that is already established between two adapters, so no names are required, and the data is sent using minimal additional header data.

To send a datagram, the machine broadcasts the NetBIOS frame. Every NetBIOS machine on the LAN reads the NetBIOS frame header to see if the destination NetBIOS name matches any of that machine′s NetBIOS names. This allows you to send information to many machines at once if desired.

To establish a connection (called a *session* in NetBIOS, and a *circuit* in AnyNet LAN Gateway), NetBIOS first broadcasts to find the partner machine. Then it sets up an IEEE 802.2 LLC 2 connection.

NetBIOS uses broadcast frames for both datagram traffic and for locating other machines. In large LANs, the amount of broadcast traffic can become overwhelming. This often becomes a problem when remote LANs are bridged together for the first time; suddenly, each LAN has all of the other LAN′s broadcast traffic added to its own.

The AnyNet LAN Gateway can alleviate this problem by a form of routing; it only forwards frames to other LANs when the destination NetBIOS names might be on the other LAN.

5.3 How Does the LAN Gateway Work?

Now we want to use the basics to better understand why the LAN Gateway can do certain things and not others.

Figure 89 shows the structure of a LAN Gateway node, and illustrates how the LAN Gateway interfaces with the LAN and the WAN.

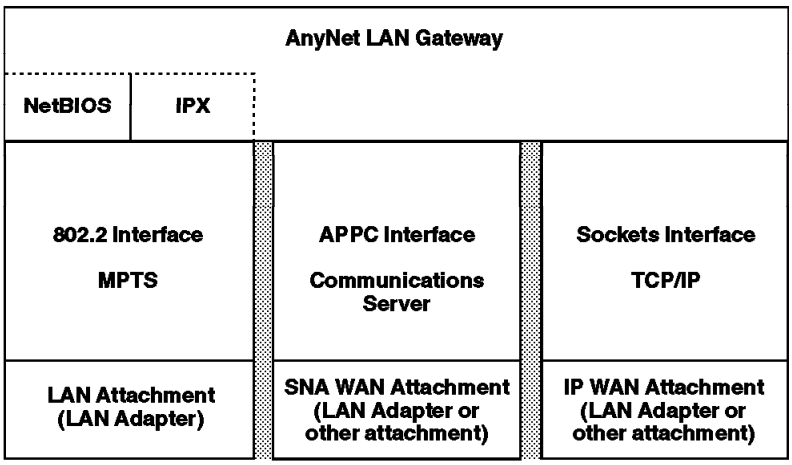


Figure 89. Structure of a LAN Gateway Node

The LAN Gateway interfaces with the LAN using IEEE 802.2 protocol support provided by Multiprotocol Transport Services (MPTS). The LAN adapter for the LAN Gateway must be configured to use the IEEE 802.2 protocol. Here we see that every protocol that shall be routed must be identifiable by a SAP.

The LAN Gateway interfaces with the WAN using either the APPC application interface in Communications Server or the TCP/IP sockets interface in TCP/IP for OS/2.

Notes:

1. The LAN Gateway cannot share the LAN adapter with other IPX or NetBIOS application programs as both protocol drivers are not able to pass a frame on to another protocol driver but use it solely for their own purpose (they do *not* share a SAP).

   For example, you cannot install both a NetWare IPX product and a LAN Gateway which is routing IPX traffic on a workstation that has one LAN adapter. In such a configuration, the LAN Gateway cannot route the IPX frames because the NetWare product receives the LAN IPX frames. If a local application program uses the IPX or NetBIOS transport, configure a different LAN adapter for that application program to use.

2. The WAN and LAN connections can use the same adapter. However, LAN traffic for the LAN Gateway cannot share the same adapter with applications that use the same LAN protocol.

The following steps describe how data is sent from a source LAN workstation to a destination LAN workstation using LAN Gateways:

1. The LAN Gateway located on the source LAN receives the frames from the source LAN workstation and adds APPC or TCP/IP headers to the frames.
2. The source LAN Gateway, using an LU 6.2 session or a TCP/IP stream, sends the encapsulated frames across the WAN to the LAN Gateway that resides on the destination LAN.
3. The destination LAN Gateway receives the data, removes the APPC or TCP/IP headers, and sends the LAN protocol frames to the destination LAN workstation.

Chapter 6. Planning for the LAN Gateway

This chapter describes what the network planner should consider before installing and configuring the LAN Gateway. Network planning considerations are also included.

This chapter is mainly taken from the manual *IBM Communications Server for OS/2 Warp Version 4.1 – Guide to AnyNet LAN Gateway*. This book is not shipped with the LAN Gateway in hardcopy but can be:

- Ordered separately (see Appendix I, "Related Publications" on page 335 for document number).
- Printed on a PostScript printer. The source is located in the directory BOOKSLISTPS on the IBM Communications Server for OS/2 Warp Version 4.1 distribution CD-ROM; the filename is PXSNGO10.LPS.
- Viewed using the BookManager READ/2, BookManager READ/DOS or BookManager READ/Windows. The book file is located in the directory BOOKSBOOK on the IBM Communications Server for OS/2 Warp Version 4.1 distribution CD-ROM; the filename is PXSNGO10.BOO.
- Viewed using the OS/2 program VIEW.EXE. The AXSGUIDE.INF file is installed when you install the LAN Gateway. For how and where to install it. See Figure 94 on page 136 in 7.2, "Installing the LAN Gateway" on page 134.

Table 10 summarizes the main planning tasks for the LAN Gateway and provides references to more detailed information.

Table 10 (Page 1 of 2). Summary of Planning Tasks for the LAN Gateway

Task	Related Information
Ensure that hardware and software requirements are met.	6.1, "Hardware and Software Requirements and Recommendations" on page 116
Check for compatibility or coexistence problems.	6.2, "Compatibility Considerations" on page 117 6.3, "Coexistence Restrictions" on page 117
Identify: • Which LANs need to be connected. • The type of WAN attachment to be used. • NetBIOS servers to be accessed. • Which workstations will be LAN Gateways.	6.4, "Network Planning Considerations" on page 118
Review requirements and recommendations for setting up the local LAN Gateway, the links between partner LAN Gateways, and the resources that can be accessed over the WAN.	6.5, "Setting Up the LAN Gateway" on page 118
Review considerations for IPX and NetBIOS LANs.	6.6, "Setting Up IPX and NetBIOS LANs" on page 127
Review considerations for SNA and IP WANs.	6.7, "Setting Up WAN Connections" on page 129

Table 10 (Page 2 of 2). Summary of Planning Tasks for the LAN Gateway

Task	Related Information
Review the methods for installing and configuring the LAN Gateway.	6.8, "Installation and Configuration Methods" on page 130
If either the LAN to LAN Wide Area Network (LTLW) program or IPX over SNA Gateway is installed on your system, review migration considerations and procedures for those products.	Appendix D, "Migration Considerations and Procedures for LTLW and IPX over SNA Gateway" on page 311

6.1 Hardware and Software Requirements and Recommendations

This section describes the hardware and software needed to run the LAN Gateway.

6.1.1 System Hardware

The LAN Gateway uses any personal computer supported by Communications Server or TCP/IP with the necessary adapter expansion slots. The LAN Gateway can be connected to the LAN using either a LAN adapter and cable, or a wireless connection.

The LAN Gateway requires the following system hardware at a minimum:

- 386SX, or compatible, processor
- 16 MB RAM (24 to 32 MB recommended for connections to more than 10 other LAN Gateways)
- 2.7 MB of hard drive space for the LAN Gateway program

6.1.2 Network Hardware

This section describes requirements for connecting the LAN Gateway to a LAN and a WAN.

6.1.2.1 LAN Connection

The LAN Gateway can attach to either a token-ring or Ethernet LAN. You can use any LAN adapter that supports Multiprotocol Transport Services (MPTS) to connect the LAN Gateway to the LAN segment.

6.1.2.2 WAN Connection

You can connect the LAN Gateway to the WAN using any adapter that is supported by Communications Server or TCP/IP. One adapter can be used to connect the LAN Gateway to both the LAN and the WAN. If the WAN adapter also connects the LAN Gateway to the LAN, a LAN adapter is not required.

6.1.3 System Software

The following system software is recommended:

- IBM OS/2 Version 3.0, or later
- For SNA WAN connections, either of the following products:
  - IBM Communications Server for OS/2 Warp Version 4.1 (recommended)
  - IBM Communications Manager/2 Version 1.11 (CSD WR01650), or later

For WAN connections, all SNA sessions between LAN Gateways require LU 6.2 support in the SNA WAN.

- For IP WAN connections, IBM TCP/IP for OS/2 Version 2.0 (CSD UN64092), or higher.
- For connection to a LAN, the LAN Gateway requires that the IEEE 802.2 interface is configured in MPTS.
- For SNMP agent support, either of the following IBM SNMP agents must be running on the LAN Gateway workstation:
  - NetView for OS/2 SNMP Agent Version 2, or higher
  - TCP/IP for OS/2 SNMP Agent Version 3.0, or higher

6.2 Compatibility Considerations

The LAN Gateway can communicate with any of the following products:

- IBM AnyNet IPX over SNA Gateway (IPX traffic on SNA WANs only)
- IBM Communications Server for OS/2 Warp
- IBM LAN-to-LAN wide area network (LTLW) program (on SNA WANs only)
- IBM Nways 2217 Multiprotocol Concentrator

The LAN Gateway cannot communicate with AnyNet/2 NetBEUI over SNA access nodes or with software implementations of RFCs 1001 and 1002, such as the NetBIOS over TCP/IP function of MPTS. For more information on how to obtain copies of RFCs, see I.4, “Requests for Comments (RFCs)” on page 336.

The LAN Gateway does not support communication with LAN stations that run the version of NetBIOS shipped with the IBM PC Network Protocol Driver.

6.3 Coexistence Restrictions

An additional LAN adapter is required if you are running application programs on the LAN Gateway workstation that use the same protocol as the traffic being routed by the LAN Gateway. If you enable the LAN Gateway loopback option, the LAN Gateway can support local applications without requiring another LAN adapter. For a description of the loopback option, see Appendix C, “Loopback Mode” on page 307.

The LAN Gateway feature cannot run on the same workstation with the LAN-to-LAN Wide Area Network Program (LTLW) program or the IPX over SNA Gateway:

- If the LTLW program is installed on your system, the LAN Gateway installation program disables the LTLW product. LTLW product files are not changed.
- If the IPX over SNA Gateway is installed on your system, you must delete the IPX over SNA product files before installing the LAN Gateway.

For more information, see Appendix D, “Migration Considerations and Procedures for LTLW and IPX over SNA Gateway” on page 311.

6.4 Network Planning Considerations

Before adding LAN Gateways to your network, determine which LANs need to be connected. Each LAN with clients or servers will need a LAN Gateway installed. Identify the LAN Gateways that need to be connected and whether the connection will be an SNA or IP WAN attachment.

Note: LAN Gateways must be directly attached to communicate. Communication using intermediate LAN Gateways is not supported.

You also need to identify the NetBIOS servers in your network that will be accessed remotely. When you set up the LAN Gateway, identifiers called name qualifiers will be required to access the NetBIOS servers in your network. Name qualifiers are discussed in more detail later in this chapter.

Finally, determine which workstations will be LAN Gateways. For each LAN to be connected, it is recommended that you dedicate a workstation for running the LAN Gateway, unless the traffic through that LAN Gateway will be minimal.

6.5 Setting Up the LAN Gateway

The following sections describe planning for the local LAN Gateway, the links between partner LAN Gateways, and the resources that can be accessed over the WAN. Major tasks include:

- 6.5.1, “Defining the Local LAN Gateway Workstation”
- 6.5.2, “Setting Up LAN Resources” on page 119
- 6.5.3, “Defining the WAN Links between Partner LAN Gateways” on page 126

6.5.1 Defining the Local LAN Gateway Workstation

Every LAN Gateway attached to the WAN must have a unique gateway name. The gateway name is used to identify the local gateway to its partner gateways.

Optionally, you can specify a region name. The region name is used to identify the LAN or LAN segments that attach to the WAN through the local LAN Gateway. If you do not specify a region name, the gateway name is used.

Use the following guidelines to plan regions and determine the physical placement of LAN Gateways between regions:

- A region can contain multiple LAN segments that are interconnected using medium access control (MAC) level bridges.
- The MAC address, ring numbers, and bridge numbers must be unique within a region. They do not have to be unique across regions.
- Broadcast traffic can flow within a region; it cannot flow outside the region onto the WAN.
- Because the IEEE 802.2 logical link control (LLC) protocol is used within a region but not across regions, an 802.2 type 2 link cannot exist between two workstations in different regions.
- Regions are interconnected by LAN Gateways rather than MAC-level bridges.

6.5.2 Setting Up LAN Resources

This section describes performance and resource considerations for the LAN Gateway. Major tasks include:

- 6.5.2.1, “Maximizing LAN Gateway Performance”
- 6.5.2.2, “Adjusting NetBIOS Timers to Compensate for the WAN and LAN Delay” on page 123
- 6.5.2.3, “Adjusting NetBIOS Application Timers to Compensate for Slower Line Speeds” on page 124
- 6.5.2.4, “Controlling Remote Access to Local Resources” on page 124

6.5.2.1 Maximizing LAN Gateway Performance

The LAN Gateway and most OS/2 applications can coexist on the same workstation if the LAN Gateway simultaneously supports 100 or fewer workstations. It is recommended that the LAN Gateway workstation be equipped with at least a 33 MHz, the 486 processor and that any local application programs require minimal workstation resources.

When the LAN Gateway shares a workstation with other applications, the gateway processes each NetBIOS and IPX broadcast frame on the LAN, in addition to frames destined for remote LANs. The gateway processes local LAN broadcasts even during times of low-volume WAN traffic. The higher the volume of LAN traffic, the more processing required of the gateway.

If the LAN Gateway supports more than 100 active workstations or simultaneously connects to more than 20 remote sites, consider configuring a dedicated workstation for the LAN Gateway. In this configuration, the workstation only processes traffic for remote LANs.

The LAN Gateway configuration tool also provides performance parameters to limit each gateway resource. Each resource is associated with a resource threshold that is continually monitored by the gateway. If the usage reaches the threshold value, the gateway is placed into critical status for that particular resource. When this occurs, a message is displayed on the gateway workstation, a message is logged, and SNA alerts and SNMP trap messages are sent.

Table 11 summarizes the performance parameters for NetBIOS LANs.

Table 11. Summary of Gateway Performance Parameters for NetBIOS LANs

Gateway Performance Parameter	Range	Resource
Maximum Number of Buffers	50–4000 buffers 8 MB maximum	Buffers
Maximum Number of Circuits	0–512 circuits	Circuit connections
Maximum Number of Link Stations	0–255 stations	802.2 link stations

Table 12 on page 120 summarizes the performance parameters for IPX LANs.

Table 12. Summary of Gateway Performance Parameters for IPX LANs

Gateway Performance Parameter	Range	Resource
Maximum Number of Buffers	50–4000 buffers 8 MB maximum	Buffers
Maximum Number of RIP Entries	10–1000 entries	Routers and servers
Maximum Number of SAP Entries	10–600 entries	Advertised servers

Buffers: This parameter controls the number of buffers the LAN Gateway allocates for temporary storage of the incoming and outgoing frames. This value, based on network size and gateway utilization, impacts the amount of gateway memory reserved for buffer use. The amount of memory required for buffer use cannot exceed 8 MB. You can determine the amount by using the following formula:

buffer size multiplied by the *maximum number of buffers*

where *buffer size* depends on the type of local area network. For example:

- Ethernet: approximately 1650 bytes
- Token-Ring: approximately 4100 bytes

The total amount of memory available for buffer use cannot exceed 8 MB. The buffers allocated by the gateway are locked in memory and are not swapped out when the gateway requires additional memory. To ensure that other applications running on the gateway have sufficient memory, the workstation should have an additional 2 to 3 MB of memory.

Note: Increasing the maximum number of buffers value may result in fewer local busy conditions, because the gateway can store more frames before signalling the workstation to stop sending frames. Decreasing this value may result in more local busy conditions.

The following table shows the recommended range for the maximum number of buffers based on the number of partner links and the number of workstation LANs. By setting the parameter to a value within the recommended range, you can obtain better performance by optimizing the memory that is allocated for routing IPX and NetBIOS traffic over SNA connections.

Table 13. Recommended Value Range for Maximum Number of Buffers

Recommended Range for Maximum Number of Buffers	Number of Partner Links	Number of LAN Workstations
50–200	1–20	1–200
150–400	20–40	200–800
300–600	40–80	500–1000
600–1000	120–250	1000–5000

For example, if a LAN Gateway running on a token-ring LAN connects to five partner LAN Gateways, it probably needs a maximum of 200 buffers. This setting allocates 820 KB of memory (4100*200) for routing IPX and NetBIOS traffic and ensures that the remaining workstation memory is available for other applications, such as SNMP agents, Sockets over SNA Gateway, and SNA Gateway.

Circuits: This parameter limits the number of NetBIOS connections maintained through the local gateway. To determine the amount of memory reserved by the gateway to track circuit activity, use the following formula:

100 bytes multiplied by the value set for *maximum number of circuits*

Because a LAN workstation can require multiple circuits, this value may exceed the actual number of LAN workstations.

Once this limit is reached, the gateway cannot establish additional NetBIOS connections to workstations attached to the LAN.

Link Stations: This parameter limits the number of NetBIOS 802.2 link connections supported by the gateway LAN adapter. The gateway reserves at least one link station for each adapter that uses an 802.2 link connection through the gateway.

This value must be less than or equal to the maximum link station value set in the MPTS 802.2 profile. Each adapter can support a maximum of 255 link stations, including any applications on the LAN workstation that do not connect through the gateway.

Once the maximum number of link stations value is reached, users on the local LAN cannot establish new NetBIOS connections, and users on remote LANs cannot reach destination targets located on the local LAN.

RIP Entries: This parameter limits the number of IPX routers, IPX LANs, and IPX servers that can communicate over a WAN.

Routing Information Protocol (RIP) is used in IPX to distribute routing information in a network. Each IPX router, including the LAN Gateway, maintains a RIP table that contains routing information such as network numbers, hop counts, and a metric for reaching remote networks.

Each server and IPX LAN create one RIP table entry. Because a gateway can support a maximum of 1000 RIP entries, a WAN is limited to 1000 servers and IPX LANs. If your network supports more than 1000 servers and IPX LANs, you can use a filter program to manage the number of RIP entries. Refer to Appendix B, “Filter Program Application Program Interface” on page 303 for more information.

SAP Entries: This parameter limits the number of IPX servers that can use the service advertising protocol (SAP) to identify themselves to remote LANs that are attached to a WAN.

Each IPX router attached to the WAN maintains a SAP table containing information such as server name, address, hop count, and server type. The router advertises the SAP table information to local LAN workstations.

Considerations for Setting RIP and SAP Entry Maximum Values: If the gateway constantly approaches the maximum value for RIP and SAP entries, you may have a problem with LAN congestion.

You might:

- Consider using a filter program to restrict the resources that broadcast their services across the LAN or WAN.

- Apply name qualifiers to the local gateway to limit local servers advertised across the WAN and significantly reduce the load on each IPX workstation. Refer to “Using Name Qualifiers” on page 124 for more information.

If these values are set too low, access to some remote services can be inhibited. Higher values can create excessive broadcast overhead in the network. For example, if a network consistently reaches the maximum value for RIP and SAP entries (1000 for RIP entries and 600 for SAP entries), the following overhead is generated:

- IPX frames of SAP and RIP information are broadcast throughout the LAN every 60 seconds.
- Each IPX workstation must process these frames.
- Servers and routers update internal tables to reflect the workstation processing.

Resource Thresholds: Threshold values can be set to alert you that gateway resources are approaching a critical stage. Table 14 summarizes the LAN Gateway thresholds for NetBIOS LANs.

Table 14. Summary of Gateway Resource Thresholds for NetBIOS LANs

Threshold	Value	Resource	Gateway Performance Parameter
Buffers	Percentage	Buffers	Maximum Number of Buffers
Circuits	Percentage	Circuit connections	Maximum Number of Circuits
Link stations	Percentage	802.2 link stations	Maximum Number of Link Stations
Local busy	Integer	Link stations	Maximum Number of Link Stations

Table 15 summarizes the LAN Gateway thresholds for IPX LANs.

Table 15. Summary of Gateway Resource Thresholds for IPX LANs

Threshold	Value	Resource	Gateway Performance Parameter
Buffers	Percentage	Buffers	Maximum Number of Buffers
RIP	Percentage	RIP entries	Maximum Number of RIP Entries
SAP	Percentage	SAP entries	Maximum Number of SAP Entries

For example, the threshold percentage value of 65 indicates that the gateway is placed in a critical state when 65 percent or more of that resource is in use. When the resource usage falls below the threshold percentage, the gateway status returns to active.

Higher values for percentage thresholds result in the gateway moving to critical status closer to the point that the resource is depleted. A lower value results in earlier and possibly more frequent warnings.

The local busy threshold is a whole number; a value of 5 indicates that the gateway is moved to critical status when 5 or more link stations are in a wait state.

Note: Local busy indicates that a workstation is trying to access the WAN, but cannot because the WAN is not accepting additional traffic. The gateway places these requests on hold until it can process them. Each workstation counts as one local busy resource. When a link station is placed into local busy status, it must wait until the gateway notifies it to transmit more frames.

If the gateway surpasses the local busy threshold frequently, consider increasing the number of buffers available. You should increase the local busy threshold if the gateway is handling:

- A significant number of link stations
- A significant number of active sessions with multiple partner gateways

6.5.2.2 Adjusting NetBIOS Timers to Compensate for the WAN and LAN Delay

When setting up NetBIOS applications and LAN workstations to communicate over the WAN, you might have to adjust application or LAN Gateway timers to include the WAN delay. You can use the poll partner function of the gateway to obtain the round-trip WAN delay times.

For NetBIOS LAN workstations, the following MPTS parameters can be adjusted to control NetBIOS LAN performance:

NETBIOSTIMEOUT (or TRANSMIT.TIMEOUT)
Specifies the amount of time, in half-seconds, that a LAN workstation waits for replies to NetBIOS queries and should include a compensation for LAN and WAN delay.

NETBIOSRETRIES (or TRANSMIT.COUNT)
Specifies the number of times a workstation resends NetBIOS queries if it continues to timeout without receiving a reply.

Note: The names of these parameters might vary, depending on the version and release of MPTS installed on the LAN workstation.

In the following example, the network delay and retry values are:

WAN delay	2 seconds in one direction
Local LAN delay	3 seconds
Remote LAN delay	3 seconds
NETBIOSRETRIES	6 retries
NETBIOSTIMEOUT	10 half-seconds

Use the following steps to adjust the NETBIOSTIMEOUT value:

1. Obtain the WAN delay (for one direction) and multiply the delay by 2:

   2 seconds multiplied by 2 = 4 seconds

2. Add the local and remote LAN delays:

   4 seconds + 6 seconds (local and remote LAN delays) = 10 seconds

3. Divide the sum by the current NETBIOSRETRIES value:

   10 seconds divided by 6 (NETBIOSRETRIES) = 1.66 seconds

4. Multiply the number of seconds by 2 to obtain the number of half-seconds:

   1.66 seconds multiplied by 2 = 3.3 half seconds

5. Round up the result to the next whole number, then add this number to the current NETBIOSTIMEOUT value.

```
3.3 half-seconds rounded up to 4 half-seconds plus
10 half-seconds (NETBIOSTIMEOUT) =
14 half-seconds (adjusted timer value)
```

Note: You might need to further increase the timer value to allow for gateway congestion or slow response time on the LAN.

Refer to the *IBM Local Area Network Technical Reference* and the DXMINFO.xxx file on the *IBM Local Area Network Support Program* diskette for more information.

6.5.2.3 Adjusting NetBIOS Application Timers to Compensate for Slower Line Speeds

Networks that use slower line speeds (such as 9600 bytes per second) for WAN connections might require that NetBIOS application timers be tuned. The time delay imposed by the SNA WAN link might require timer changes in the NetBIOS workstation's IBMLAN.INI file.

In the IBMLAN.INI file, you can set the server heuristic (SRVHEURISTIC) bit 15 to a value of 9. This setting disables the file transfer timeout to prevent file transfers from failing on slow links.

You can also increase the *sesstimeout* value for requesters to avoid timeout conditions. The default value is 45 seconds. It is recommended that you increase this value to 120 seconds.

6.5.2.4 Controlling Remote Access to Local Resources

The LAN Gateway can apply name qualifiers, filter programs, or both to restrict or permit remote access of local resources.

Using Name Qualifiers: Name qualifiers are required for NetBIOS LANs and are optional for IPX LANs. When two LAN Gateways establish connections, they exchange name qualifier lists. The LAN Gateway uses the lists it receives from partner LAN Gateways to determine which LAN will receive session startup or datagram frames.

If a LAN Gateway does not specify name qualifiers, NetBIOS workstations in remote LANs cannot initiate communication with NetBIOS workstations in the local LAN. If a local workstation initiates the communication, the partner LAN Gateway can route frames to the local workstation as long as the session is active.

Each LAN Gateway can have one name qualifier list with up to 100 name qualifiers. A name qualifier is a 1- to 8-character value that is case-sensitive and contains the characters A – Z and 0– 9 . It consists of all or part of a LAN server name.

In Figure 90 on page 125 the LAN Gateways establish connections across the WAN and exchange name qualifier lists. Using the server names on LAN A and LAN B, suggested name qualifiers for the LAN Gateways are:

- For LAN Gateway A: J28 and H76
- For LAN Gateway B: Home and QUAL

Name qualifiers SERV, SRV, or PRT are not good choices because they match server names on both LANs.

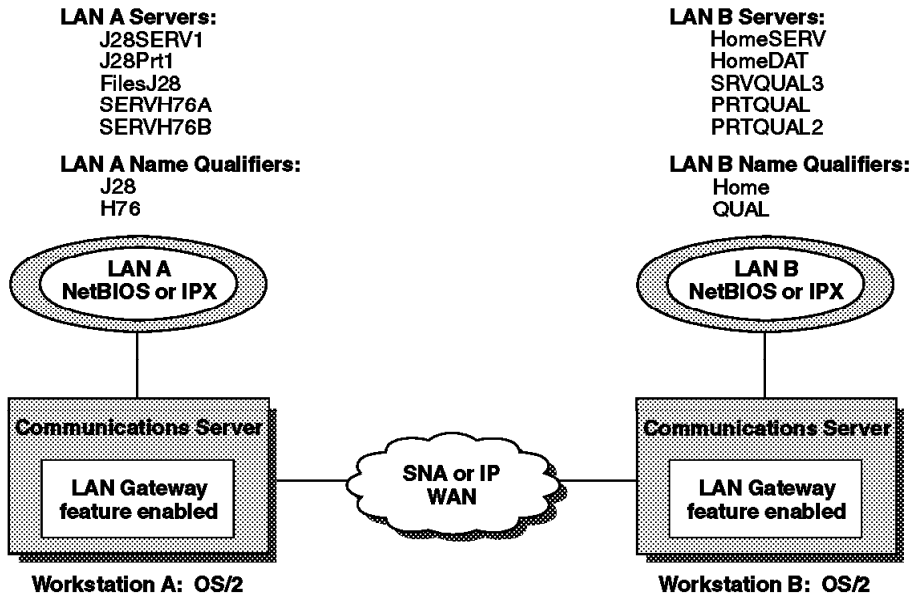


Figure 90. Configuration Showing NetBIOS Name Qualifiers Used in the Network

A name qualifier should be unique within the network of partner LAN Gateways and should be explicit enough to distinguish a specific name or group of names from similar names in other LANs. Any portion of the server name can be used for the name qualifier.

It is more efficient for LAN Gateway processing to choose name qualifiers that match server names on only one LAN. If name qualifiers match servers on more than one LAN, this results in excessive and unnecessary WAN traffic.

You can apply name qualifiers to IPX service advertising protocol (SAP) tables to limit the servers that advertise to partner LAN Gateways. If the local LAN Gateway applies name qualifiers, partner LAN Gateways can only access servers with names matching one of the name qualifiers.

It can take up to three minutes to dynamically apply name qualifiers to the SAP entries.

Using Filter Programs: The name qualifier lists received from partner gateways provide the primary filtering of destination names and addresses. A filter program can then be called to further restrict LAN or individual workstation traffic leaving the LAN.

You can use a filter program to:

- Restrict certain destination addresses (for example, if the name qualifier list defined in this gateway is too general)
- Permit or restrict access to LAN resources
- Permit or restrict access between specific source and destination pairs
- Forward or filter frames with certain values in specific fields

The sample filter program (AXSFILTR.DLL) restricts communication on a station-to-station basis. You can modify the filter input file to specify the workstations that are permitted (or restricted) to communicate across the link between the LANs. To restrict communication in both directions, specify the appropriate pairs of workstations in the partner gateway's filter program as well.

If you write your own filter program:

- Specify the name of the filter program and the input data file, if any, to the configuration program when setting up the gateway configuration file for operation.
- Ensure that the supplied or user-written DLL program is in a DLL subdirectory accessed by the LIBPATH command in your CONFIG.SYS file.

See Appendix B, "Filter Program Application Program Interface" on page 303 for details.

6.5.3 Defining the WAN Links between Partner LAN Gateways

Each LAN Gateway defines links to those partner LAN Gateways to which it initiates connections. If a LAN Gateway is defined to accept connections from non-configured partners, it does not have to define links to those partner gateways that it accepts connection attempts from.

While the type of link depends on the WAN transport protocol, SNA or IP, other configuration issues include:

- Link startup type
- Transmission mode (SNA links only)

6.5.3.1 Defining the Type of Link Startup

The Link Startup option of the LAN Gateway configuration tool lets you specify when and how a WAN link starts:

- Autolinks start automatically.
- Manual links must be started by the operator.
- Dynamic links are started when a query is received on the LAN for a NetBIOS name that matches the name qualifier associated with a partner gateway.

Note: You can start a manual or dynamic link at any time after starting the LAN Gateway communications function.

When determining how a particular link should start, consider the factors in Table 16.

Table 16 (Page 1 of 2). Factors That Affect Link Definitions

Factor	Related Information
Cost of the link	Expensive links are candidates for dynamic or manual links.
Amount of traffic on the link	Heavily used links are candidates for autolinks.
Dispersion of traffic over time on the link	Links supporting cyclical traffic are candidates for dynamic links.

Table 16 (Page 2 of 2). Factors That Affect Link Definitions

Factor	Related Information
Availability of operators to manage manual links	Manual links must be stopped and started manually.

6.5.3.2 Specifying the Transmission Mode for SNA Connections

LAN Gateway uses LU 6.2 conversations to enable communication over an SNA WAN. When an LU 6.2 conversation is established, both the local and remote LAN Gateways define a mode name for the SNA link.

Each LAN Gateway supplies the transmission mode name to Communications Server. Communications Server uses the mode name to identify the characteristics of the SNA connection, such as pacing parameters, receive and request unit sizes, and session limits.

A LAN Gateway can use multiple transmission modes for its SNA connections as long as each transmission mode is defined by Communications Server or TCP/IP and is defined throughout the WAN.

For a summary of SNA profile options and an overview of transmission modes, see *IBM Communications Server for OS/2 Warp Version 4.1 - Up and Running!*. For more detailed information, see the *Network Administration and Subsystem Management Guide*.

6.6 Setting Up IPX and NetBIOS LANs

The LAN connection for the LAN Gateway workstation must be defined to MPTS on the LAN Gateway workstation. In addition, the LAN Gateway cannot share the LAN adapter with other application programs that use the same protocol as the data being routed through the LAN Gateway.

For example, if the LAN Gateway is routing NetBIOS data through adapter 1, NetBIOS applications on the same workstation must use a different adapter. IPX applications on this same workstation can use adapter 1 only if the LAN Gateway is not routing IPX traffic.

6.6.1 IPX LAN Considerations

For LANs that support IPX traffic, each LAN must have an IPX network number. For IEEE 802.2 LAN connections, the IPX network number is defined by the LAN Gateway configuration tool.

6.6.1.1 Specifying IPX LAN Protocols

You can configure LAN Gateway to support the following IPX LAN protocols for token-ring or Ethernet:

- 802.2, using the X′E0 service access point
- SNAP, using the X′AA service access point

The gateway keeps information about both server types in its SAP table, and provides concurrent access to 802.2 IPX and SNAP IPX workstations.

Table 17 on page 128 lists the IPX frame types (and associated gateway settings) supported for IPX networks.

Table 17. Supported IPX Frames for IPX Networks

LAN Type	Frame Type	Gateway Setting
Ethernet	ETHERNET_802.2	802.2
	ETHERNET_SNAP	SNAP
Token-Ring	TOKEN-RING	802.2
	TOKEN-RING_SNAP	SNAP

6.6.1.2 Routing Data to IPX Workstations

At the link layer, IPX is a connectionless protocol. IPX relies on applications to advertise their services in order to find addresses. Routers pass the servers' advertisements around the network. These functions use Service Advertising Protocol (SAP) and Routing Information Protocol (RIP).

When a requester wants to access a server, it sends out frames asking where the nearest server is and then how to access the destination server. Each router on the network helps local workstations address the correct server by maintaining and supplying SAP and RIP information from other regions of the network.

The LAN Gateway supports the SAP and RIP protocols. It obtains SAP and RIP information from the local LAN during startup and passes this information to each partner gateway it connects to over the WAN. As SAP and RIP information is updated, the gateway resends the information to each partner gateway.

Each LAN Gateway is limited to storing a maximum of 600 SAP entries and 1000 RIP entries. If your network has more than 600 servers, you can apply name qualifiers to the SAP entries. Name qualifiers restrict which servers are advertised across the WAN and, therefore, can be remotely accessed. Servers that are only used locally do not need to be advertised remotely. By applying name qualifiers, you can manage servers that are addressable over the WAN and restrict the number of advertised IPX resources.

Unlike the LAN Gateway support for NetBIOS connections, IPX support does not require you to manually update network topology information (SAP and RIP entries) when the network configuration changes, except when the local IPX network number changes. To limit the number of WAN broadcasts required for SAP and RIP entry updates, the gateway regularly updates the tables. Approximately once every minute, the gateway scans for network configuration changes. If the configuration has changed during the time interval, such as a server shutting down, the gateway makes the appropriate changes to its tables and sends the changes to its partner gateways.

6.6.1.3 Using the NetWare Packet Burst Feature to Enhance IPX Performance across the WAN

By default, IPX requires an acknowledgment on every frame before sending the next packet. This is satisfactory when the server and requester are located on the same LAN. When devices are on separate LANs, connected by a WAN, performance can be affected.

To improve IPX performance across the WAN, NetWare provides a feature called Packet Burst. IPX workstations running this feature send 4 to 8 frames before requiring an acknowledgement from the destination workstation. If your WAN

links are 64 Kbps or less, the 4 to 8 frames multiplied by the size of your frames can create a significant delay.

If you are running the Packet Burst feature and have slow links, configure the workstation to use smaller frame sizes (for example, 1500 bytes per frame). For NetWare Version 3.12 or Version 4.0 (or later), frame size is specified in the NET.CFG file. Optionally, you can configure large frame sizes (for example, 4000 bytes per frame) and not use Packet Burst.

Depending on the version of IPX software installed on the workstation, this feature is available as either an additional virtual loadable module (VLM) or an optional configuration parameter. For more information, refer to your Novell NetWare documentation.

6.6.2 NetBIOS LAN Considerations

For Ethernet LANs, NetBIOS can run either IEEE 802.3 or Ethernet DIX frame types. Configure the frame type in MPTS. On token-ring LANs, NetBIOS uses only the 802.2 frame type; configuration is not required.

The NetBIOS protocol uses Logical Link Control (LLC) Type 1 or datagram frames (connectionless) and LLC Type 2 or session (connection-oriented) frames.

Note: Type 1 frames also include queries, responses, and acknowledgements.

The protocol uses names for addressing. A requester broadcasts to a domain using a name to find a resource. When the domain controller or server responds, a few other LLC Type 1 frames flow. These frames include names for addressing. The workstations then initialize a session and begin using LLC Type 2 frames that have session numbers, but no names, for addressing.

The gateway routes the broadcasts using the qualifier lists. Most applications restrict NetBIOS resource names to 8 characters. Because frames on the LAN have 16 bytes, applications can add a prefix or a suffix to the name that contains data for the applications' purposes. Because the resource name can be found anywhere in the 16-byte frame name, the gateway uses an implied wild card to determine the qualification of a destination name.

When a NetBIOS workstation locates a resource, it connects to the partner using LLC Type 2 (connection-oriented) frames. The gateway acknowledges all frames sent to remote partners as those frames are received from the LAN; this allows the gateway to satisfy the 802.2 timing parameters and hop-count restrictions for LANs. The gateway logs the NetBIOS session, or circuit, and tracks the amount of data transmitted over the circuit.

6.7 Setting Up WAN Connections

This section describes planning considerations for SNA and IP WANs.

6.7.1 SNA APPN and Subarea WANs

The LAN Gateway appears to the SNA network as an independent LU 6.2 application program.

In APPN networks, all LAN Gateway configurations can use network-qualified LU names. No other configuration of the SNA network is required. In subarea networks, a LAN Gateway can be configured as follows:

- The LU name for each LAN Gateway attached to the WAN and the transmission modes that each gateway uses to communicate with partner gateways must be defined to the SNA access program (such as VTAM) that communicates with the LAN Gateway.
- Each LAN Gateway must define its LU names and the LU names of all its partner LAN Gateways to Communications Server, along with the other parameters that support an LU.

For an overview of SNA configuration options, see *IBM Communications Server for OS/2 Warp Version 4.1 - Up and Running!*. For more detailed information, see the *IBM Communications Server for OS/2 Warp Version 4.1 - Network Administration and Subsystem Management Guide*.

6.7.2 IP WANs

A LAN Gateway can be configured using either IP addresses or host names, but host names are preferred. Each LAN Gateway appears as a host on the IP network.

All stream socket calls transferred between LAN Gateways over IP WANs are bound to TCP port number 1491. It is highly recommended that you do not change the port number used by the LAN Gateways. If an application program needs to use port number 1491, you can reconfigure all LAN Gateways to use a different port number. Refer to Appendix E, “WAN TCP Port Number Used by LAN Gateways” on page 313 for more information.

6.8 Installation and Configuration Methods

There are three methods for installing and configuring the LAN Gateway:

Locally from a CD
This is the recommended method for small networks or for testing connectivity between a few nodes.

Interactively from a code server
Consider using this method for a large network. Each remote node can log on to a server and interactively install and configure LAN Gateways from the server, just as installation and configuration is done locally from a CD. This method eliminates the need to distribute the installation CD to each target node.

Noninteractively from a code server
Consider using this method to install and configure a large network. Installing and configuring LAN Gateway noninteractively allows you to establish installation and configuration parameters in one location, then distribute that information at a specified time. End users are not required to use the installation and configuration utilities to set up their workstations.

You can use either NetView Distribution Manager/2 (NVDM/2) or the LAN Gateway CID Setup Tool to install and configure the LAN Gateway noninteractively.

The LAN Gateway configuration tool supports two methods for configuring LAN Gateway workstations:

- A single configuration defines and sets up a local workstation as a LAN Gateway.
- A master configuration allows you to create configuration files for multiple gateways at a single workstation. These files can be saved to diskettes or hard disk for distribution to remote LAN Gateway workstations.

If your network utilizes a small number of LAN Gateways, the single configuration method might be adequate. If you plan to configure a large number of LAN Gateways or anticipate having to handle a large number of future updates or gateway modifications, you should consider using the master configuration method.

For more information on installation and configuration using the Master Configuration Method, see Chapter 7, “LAN Gateway Scenarios” on page 133 as well as Chapter 4 and Appendix C of the *IBM Communications Server for OS/2 Warp Version 4.1 – Guide to AnyNet LAN Gateway* manual.

Chapter 7. LAN Gateway Scenarios

Let us start with a very generic (and also very simple) scenario shown in Figure 91 and proceed to the possible combinations and connectivities.

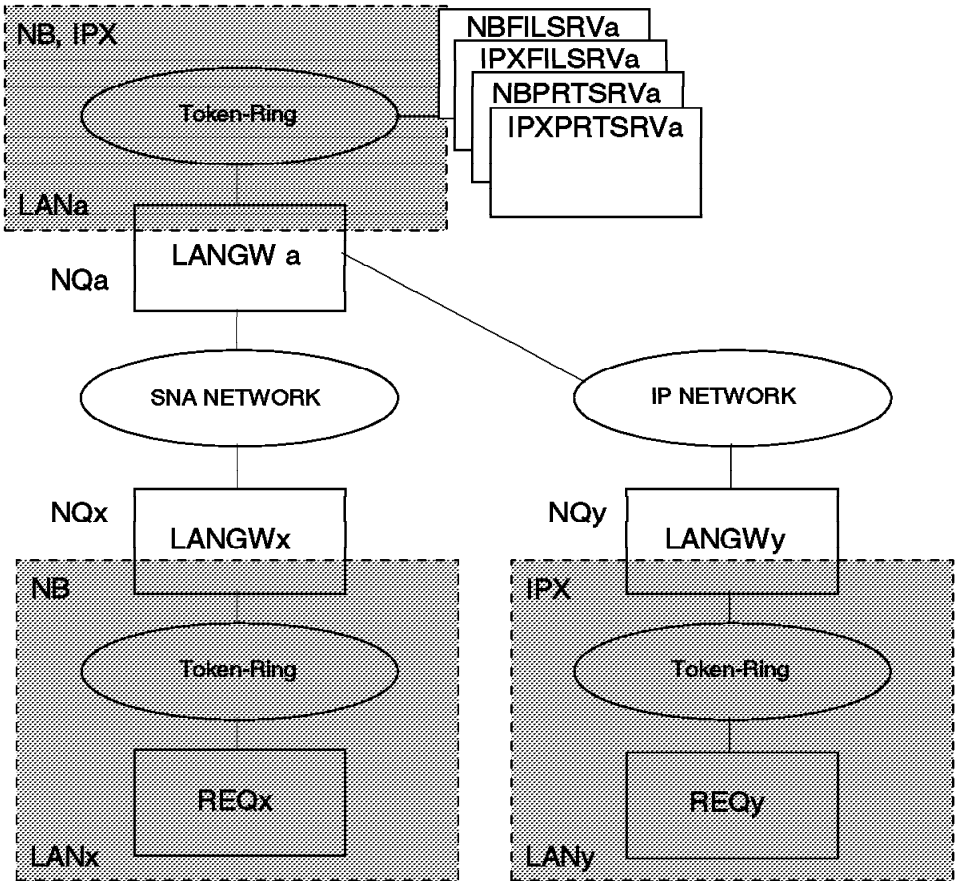


Figure 91. A Generic LAN Gateway Scenario

The following four chapters are intended to provide a step-by-step walkthrough showing the setup process for the LAN Gateways shown in Figure 91.

You can have pairs of LAN Gateways:

- Using SNA as the WAN Link
  - Routing IPX over the SNA WAN
  - Routing NetBIOS over the SNA WAN
- Using IP as the WAN Link
  - Routing IPX over the IP WAN
  - Routing NetBIOS over the IP WAN

You can also combine all of those scenarios depending on how complex your network is. You may have IP and SNA WANs connecting any pair of LAN Gateways.

Note: The LAN Gateway cannot route traffic from an IP WAN to an SNA WAN.

So in the scenario shown in Figure 91 on page 133, Workstation REQx on LANx will not be able to communicate with Workstation REQy on LANy using the NetBIOS or IPX protocol unless either LAN Gateway LANGW x or LANGW y is connected to WAN IP or SNA respectively. The *or* in the above sentence may be interpreted as an exclusive or.

7.1 Definitions Used in the Scenarios

We used the following definitions throughout the scenarios:

Table 18. LAN Gateway Configuration Values

	LANGWa	**LANGWx**	**LANGWy**
Region name	LANA	LANX	LANY
Name Qualifiers	NBFILSRV (for example)	REQX (optional)	REQY (optional)
TCP/IP host address	129.1.1.1	n/a	129.1.1.10
SNA FQ Node Name	USIBMRA.LANGWA	USIBMRA.LANGWX	n/a

7.2 Installing the LAN Gateway

There is more than one method of installing the LAN Gateway but we concentrate on the fully attended method. To start the installation, change to the directory SERVERLANGW on the IBM Communications Server for OS/2 Warp Version 4.1 distribution CD-ROM or an equivalent subdirectory on a code server. At the command prompt type INSTALL. You will see the following picture overlayed by the README.ANY file. After selecting **Continue**, this overlay will go away:

Figure 92. Installing the LAN Gateway: The Installation Screen

Select **LAN Gateway** and **Install** (see Figure 93).

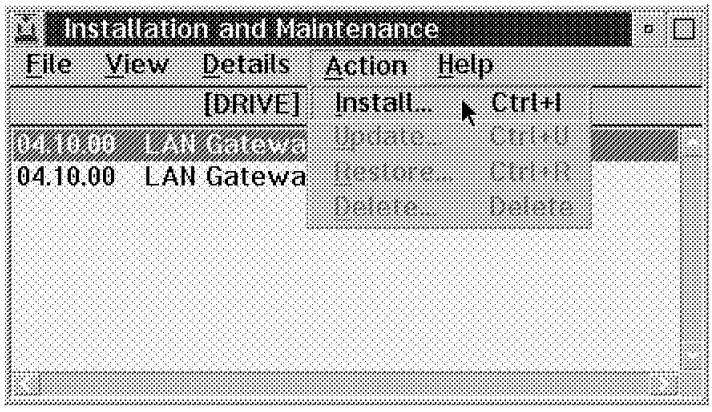


Figure 93. Installing the LAN Gateway: Available Actions

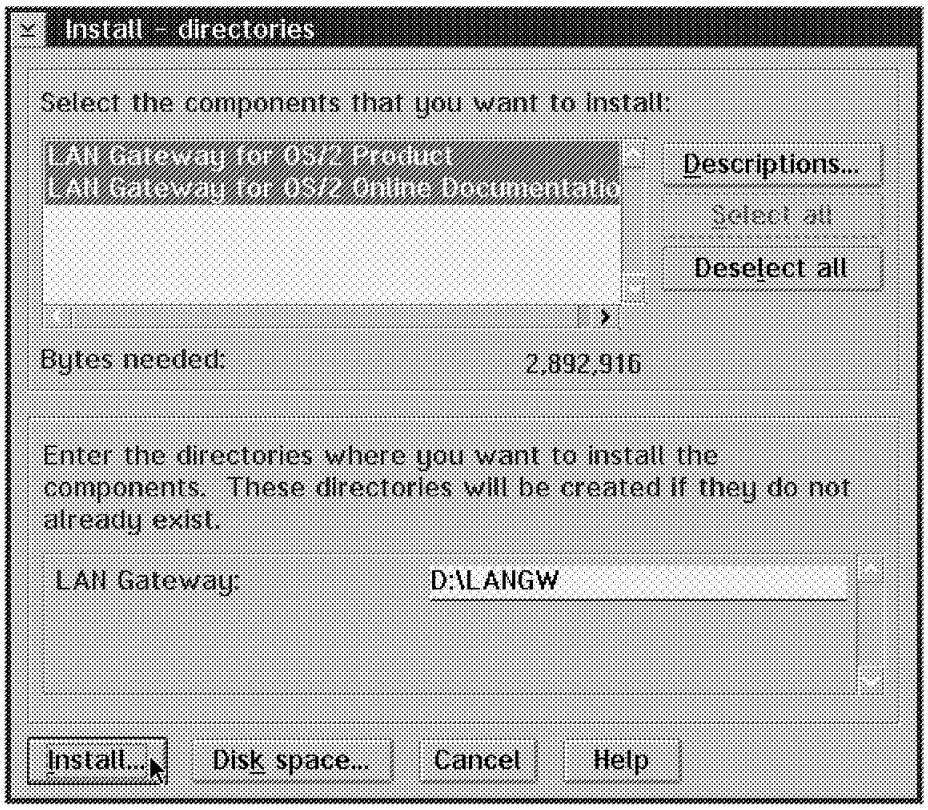


Figure 94. Installing the LAN Gateway: Select Components to Install

You may now choose the products to install. The Online Documentation is an .INF file which has already been mentioned in Chapter 6, “Planning for the LAN Gateway” on page 115.

If you choose the **Disk space** push button, you can easily check the free disk space on all your drives and change to that drive without keying anything in.

Selecting the **Install** push button takes you through the following three screens.

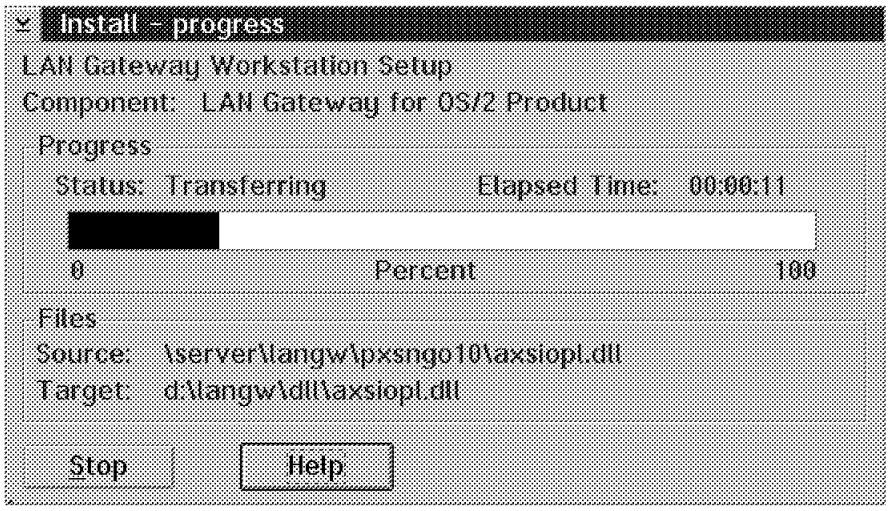


Figure 95. Installing the LAN Gateway: Progress Indicator

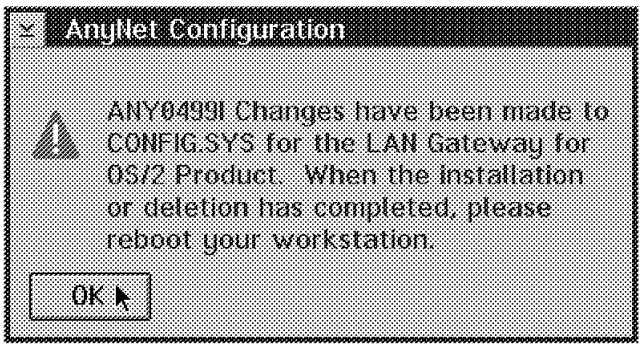


Figure 96. Installing the LAN Gateway: The LAN Gateway Is Installed

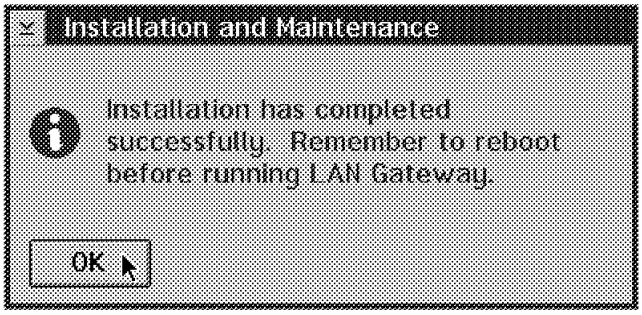


Figure 97. Installing the LAN Gateway: Success

After rebooting the computer you will notice a new folder on the OS/2 Desktop. When you open the LAN Gateway Folder (shown in Figure 98 on page 137) you will find the following icons:

- LAN Gateway Configuration Tool
- Start LAN Gateway
- LAN Gateway Users Guide
- README.ANY

Figure 98. The LAN Gateway Folder

Now the configuration of the LAN Gateway(s) can begin.

The product manual *IBM Communications Server for OS/2 Warp Version 4.1 – Guide to AnyNet LAN Gateway* also describes a configuration scenario but without any screens. We also use the master configuration method.

Let us start with the LAN Gateway LANGWx which has an SNA link to LAN Gateway LANGWa. The configuration panels for this LAN Gateway are shown in 7.5.2, “Setting Up System: LANGWx” on page 158 and in 7.3, “Configuring the LAN Gateways” on page 138.

Then we take a look at the configuration of LAN Gateway LANGWy which has a TCP/IP link to LAN Gateway LANGWa. The configuration panels for this LAN Gateway are shown in 7.5.3, “Setting Up System: LANGWy” on page 161 and in 7.3, “Configuring the LAN Gateways” on page 138.

Finally we are going to configure LAN Gateway LANGWa with two simultaneous (one SNA and one TCP/IP) links. The configuration panels for this LAN Gateway are shown in 7.5.1, “Setting Up System: LANGWa” on page 154 and in 7.3, “Configuring the LAN Gateways.”

As we use the master configuration method, the resulting configuration files need only be copied to the actual machines:

- <configname>.RSP
- AXSCONF.INI

Note: We configured the scenario on LAN Gateway LANGWa. Keep this in mind throughout the configuration and setup session.

In Appendix A, “Example Configuration Files” on page 289 we have included the configuration files for our three LAN Gateways.

7.3 Configuring the LAN Gateways

Locate the LAN Gateway folder on your desktop, open it and start the **LAN Gateway Configuration Tool**. You will see the following screen:

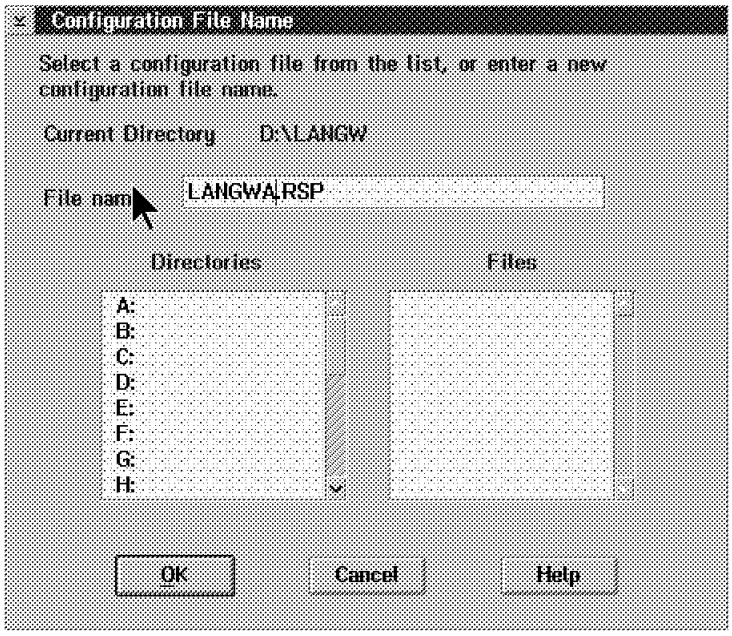


Figure 99. Enter or Select Your Configuration Here

Enter the desired master configuration file name and proceed to the next one or two screens. (The number of screens you will see depends on the existence of the configuration you enter.)

Note: Do not forget the .RSP extension as the AXSCONF.INI file references the configuration file with the extension.

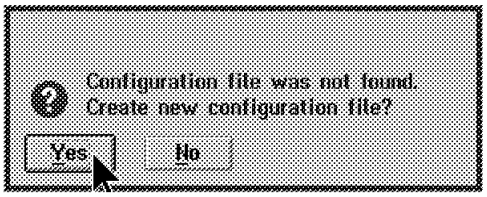


Figure 100. Creating a Configuration File

Select **Yes** as shown in Figure 100.

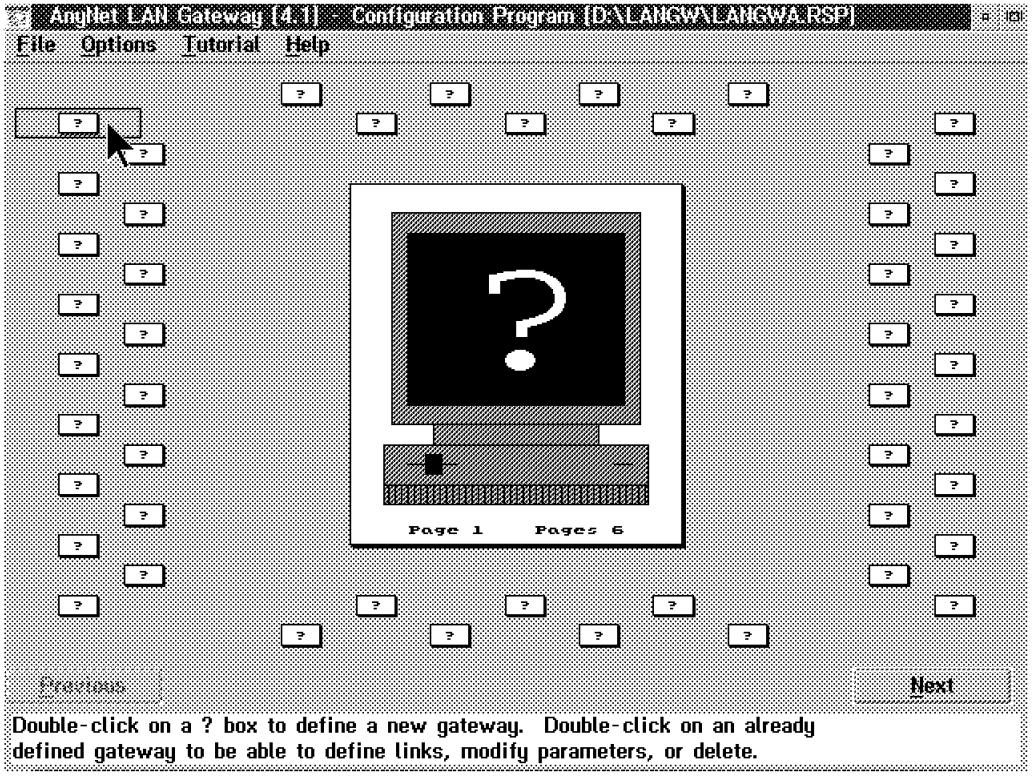


Figure 101. The Main Configuration Panel

7.3.1 Configuration of LAN Gateway: LANGWx

You may start configuring the LAN Gateway LANGWx by double-clicking on any of the question marks. This brings up Figure 102 on page 140.

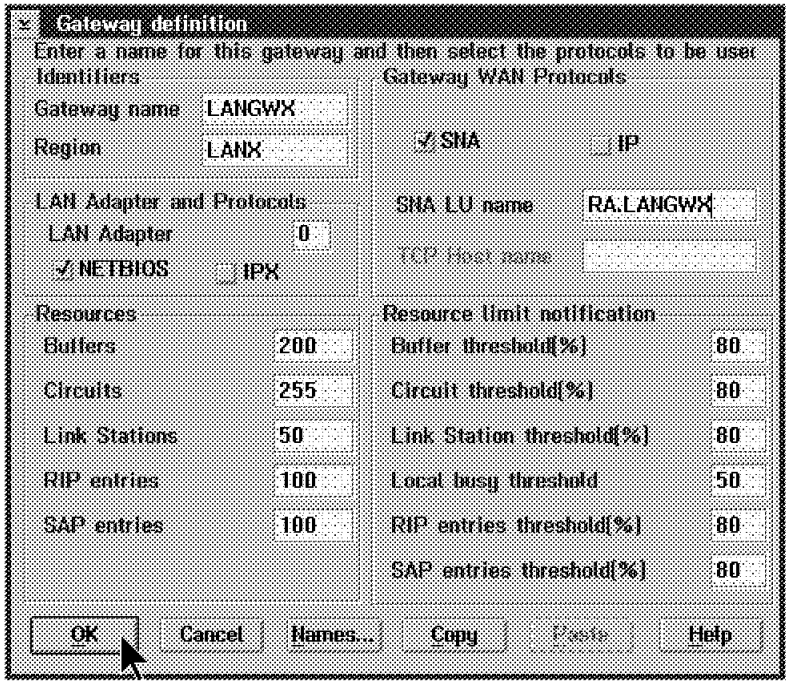


Figure 102. Parameters for LAN Gateway LANGWx

The only parameters that need to be entered are the Name, Region, LAN Adapter and Protocol and the WAN Connection method and Identification unless a closer study of Chapter 6, “Planning for the LAN Gateway” on page 115 has revealed a need to change some of the Resources parameters.

After selecting **OK**, you will see a modified Main Configuration Screen.

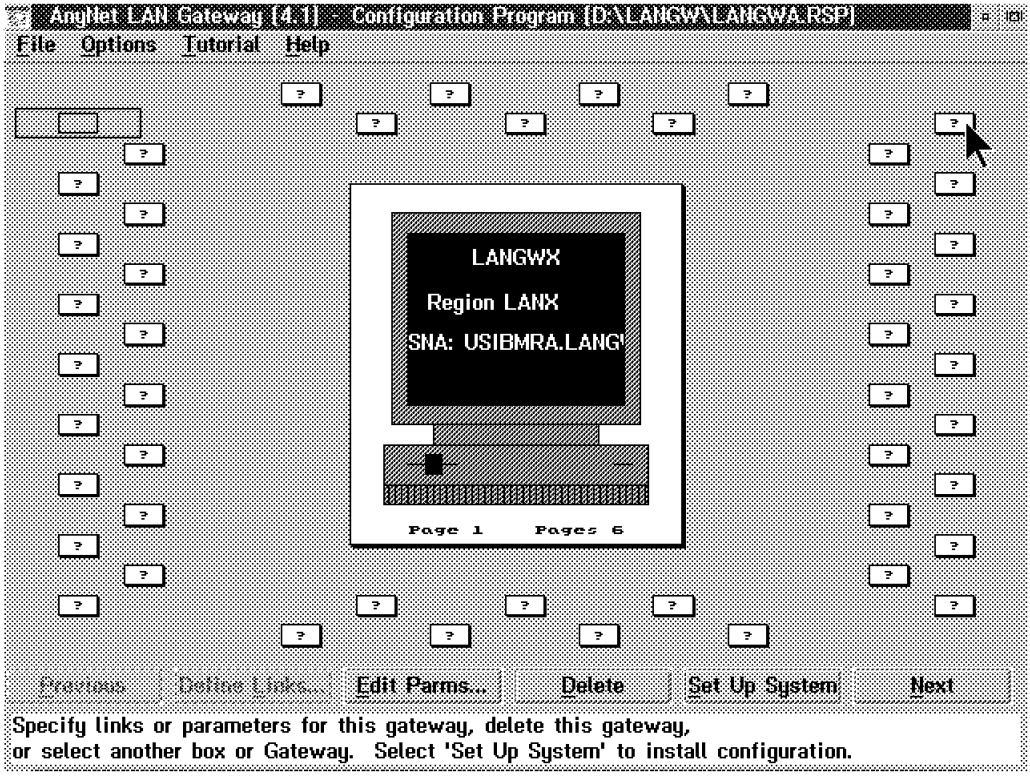


Figure 103. Modified Main Configuration Screen for LANGWx

7.3.2 Configuration of LAN Gateway: LANGWy

The Configuration of the LAN Gateway LANGWy is also started by double-clicking on any of the boxes with a question mark. This brings up Figure 104 on page 142.

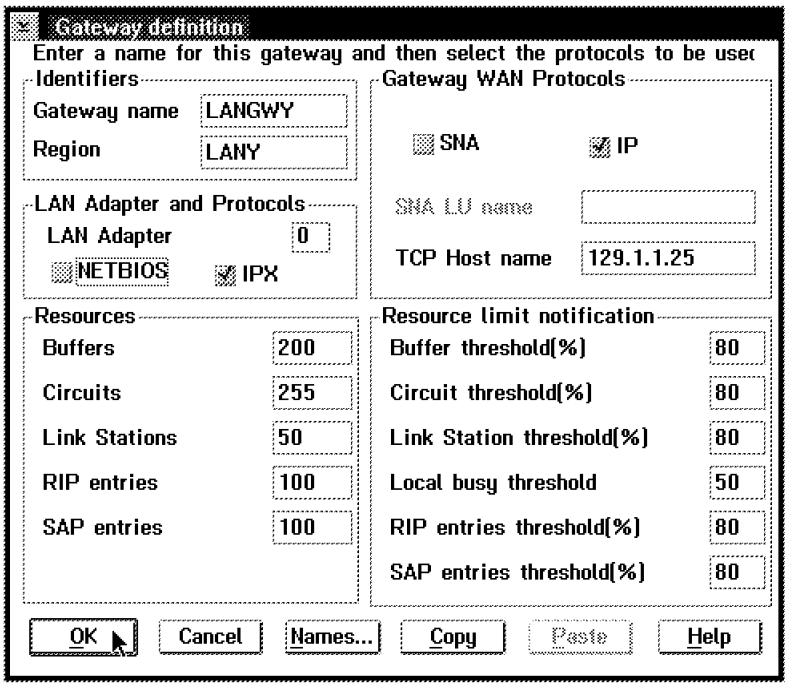


Figure 104. Parameters for LAN Gateway LANGWy

The only parameters that need to be entered are the Name, Region, LAN Adapter and Protocol and the WAN Connection method and Identification unless a closer study of Chapter 6, “Planning for the LAN Gateway” on page 115 has revealed a need to change some of the Resources parameters.

After selecting **OK** you will again see a modified main configuration screen.

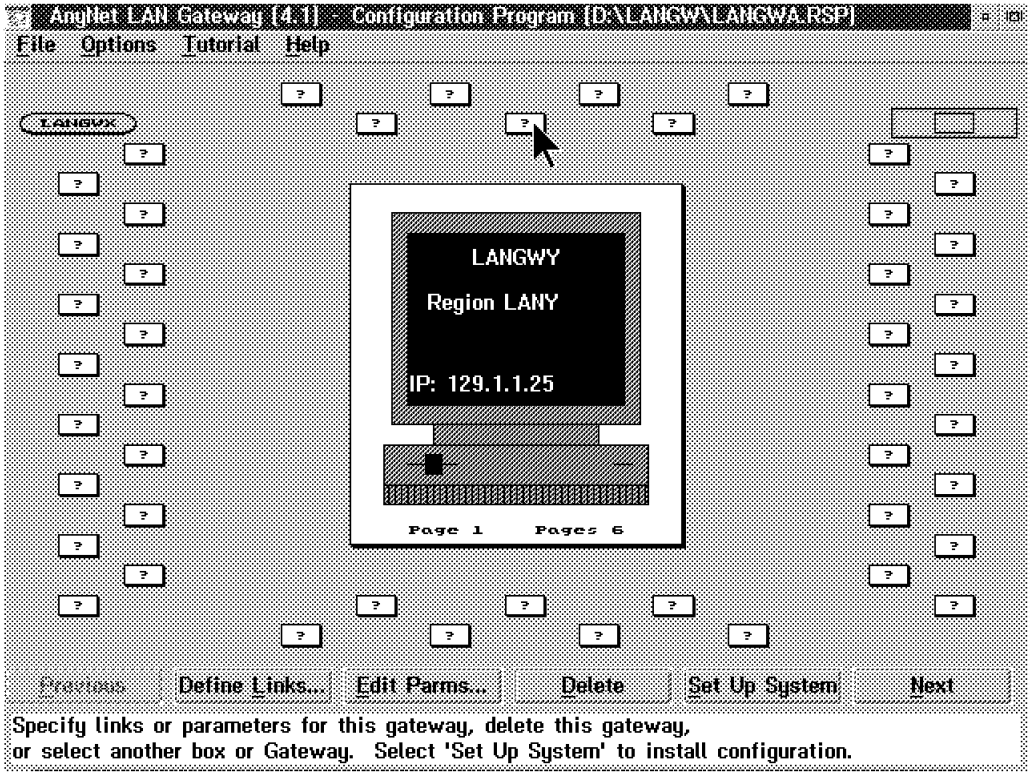


Figure 105. Modified Main Configuration Screen for LANGWy

7.3.3 Configuration of LAN Gateway: LANGWa

The configuration of the LAN Gateway LANGWa is started by double-clicking on any of the boxes with a question mark. This brings up Figure 106.

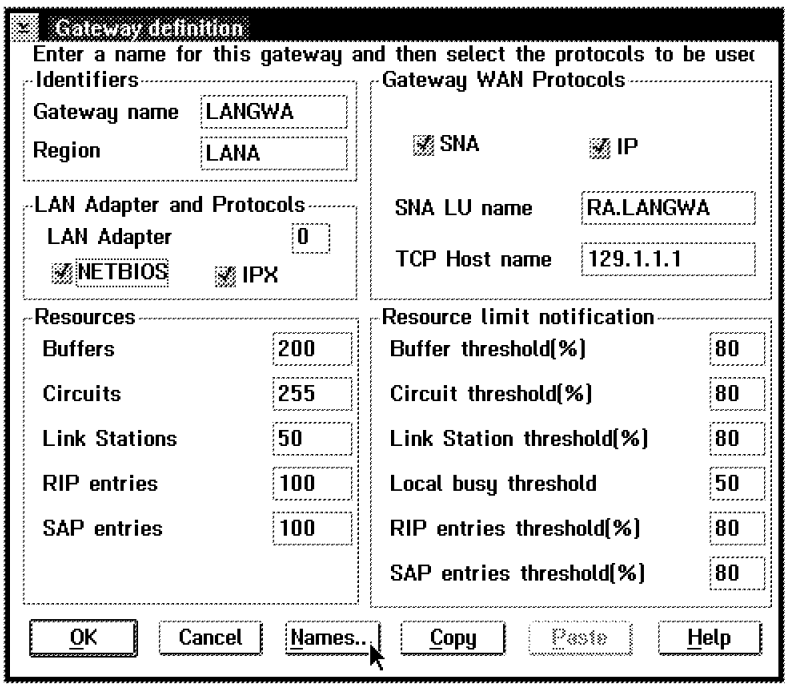


Figure 106. Parameters for LAN Gateway LANGWa

The only parameters that need to be entered are the Name, Region, LAN Adapter and Protocol and the WAN Connection method and Identification unless a closer study of Chapter 6, “Planning for the LAN Gateway” on page 115 has revealed a need to change some of the Resources parameters.

For completeness we also show how to enter name qualifiers that are necessary for NetBIOS operation. Only servers whose names are in the name qualifier list will be *seen* by requesters in other LANs that are connected by partner LAN Gateways. A more detailed discussion of the name qualifiers can be found in “Using Name Qualifiers” on page 124.

When you click on the **Names** push button you will see the panel shown in Figure 107 on page 144.

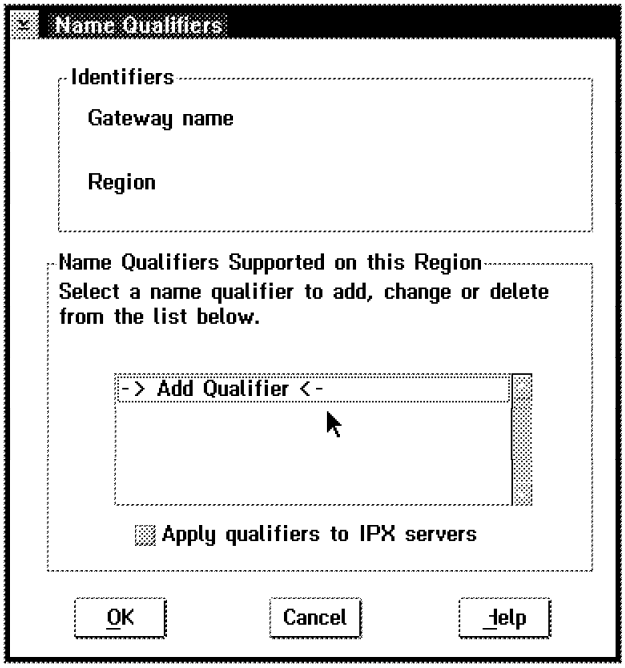


Figure 107. Name Qualifiers

Double-click on **Add Qualifier** and you will see the panel shown in Figure 108 on page 144.

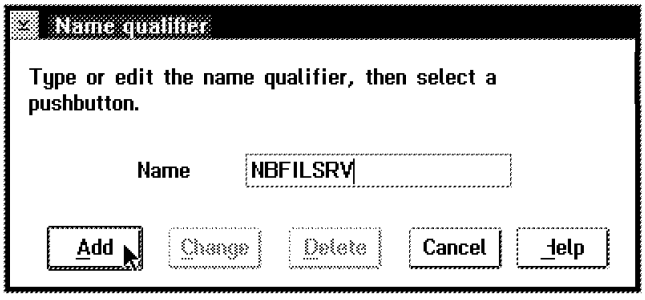


Figure 108. Name Qualifier

You will see a slightly modified Figure 107 which will look similar to Figure 109 on page 145 depending on the number of name qualifiers you specified.

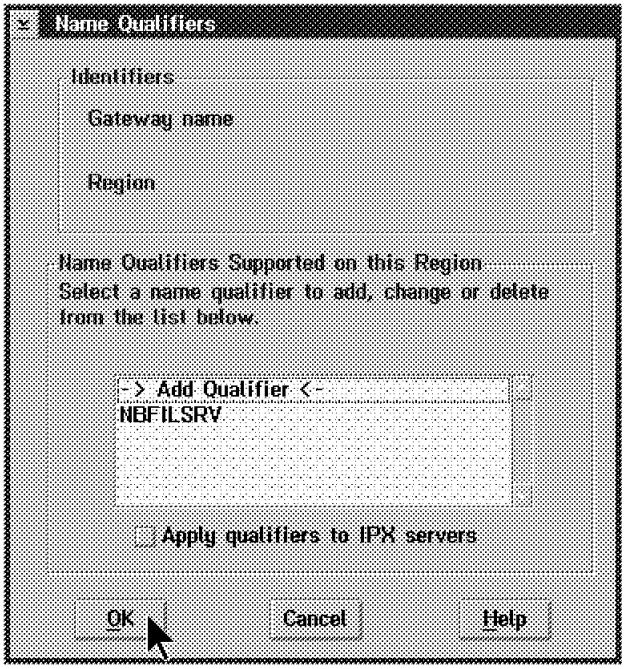


Figure 109. Defined Name Qualifiers

You may repeat the previous two steps for every name qualifier that you want to enter.

Note: If they get too numerous you might consider editing the .RSP file directly.

If you are done with defining name qualifiers, select **OK** on the panels shown in Figure 109 on page 145 and Figure 106 on page 143.

You will again see a modified main configuration screen as shown in Figure 110 on page 146.

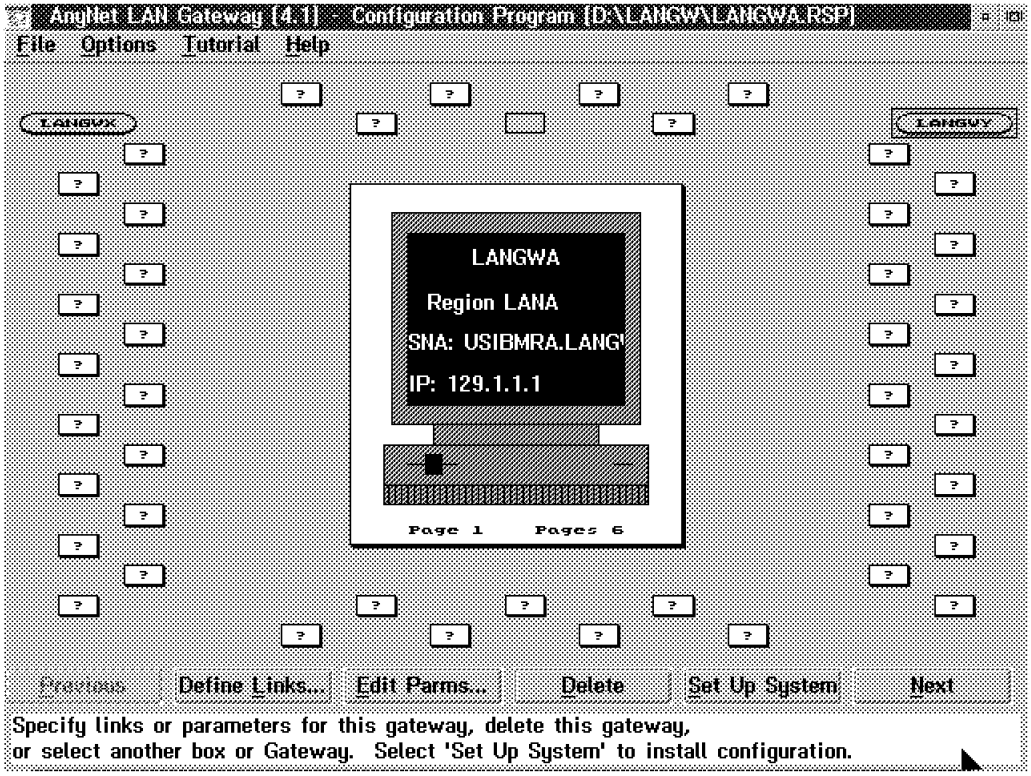


Figure 110. Modified Main Configuration Screen for LAN Gateway LANGWa

7.4 Defining the Links between the LAN Gateways

In this section, we define the links between LAN Gateways in order to transport NetBIOS and/or IPX traffic over wide area networks (WAN).

7.4.1 Defining the Links between LAN Gateway LANGWx and Its Partner(s)

You may start configuring the links between this LAN Gateway and any number of other LAN Gateways following the screens shown below.

First you double-click on **LANGWx**, which brings up the panel shown in Figure 111 on page 147.

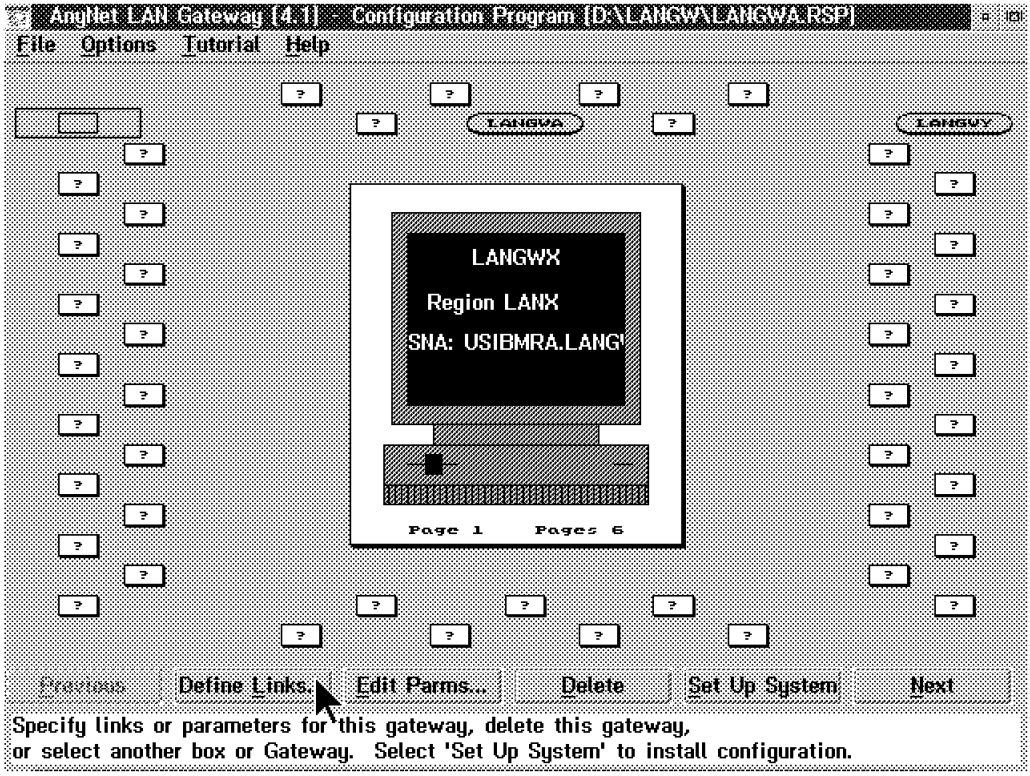


Figure 111. Defining Links from LANGWx to Other LAN Gateway(s)

Clicking on **Define Links** displays the panel shown in Figure 112 on page 147.

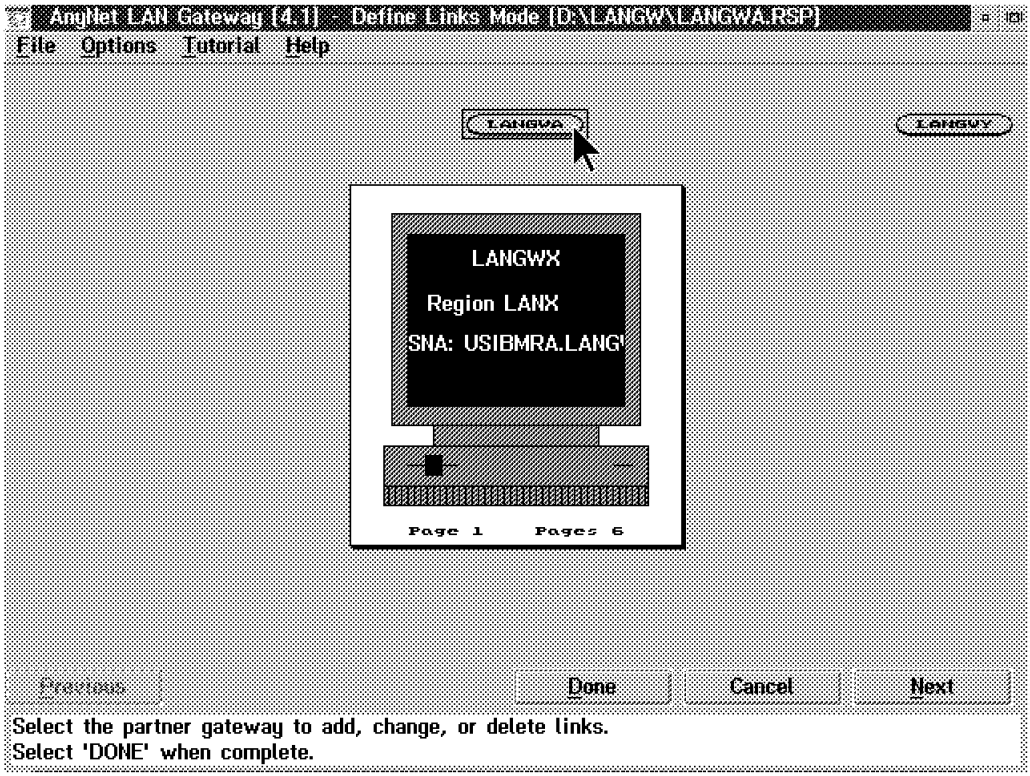


Figure 112. Selecting Partner LAN Gateway(s)

Double-click on the LAN Gateway to which you want to create a link and fill out the panel shown in Figure 113 on page 148.

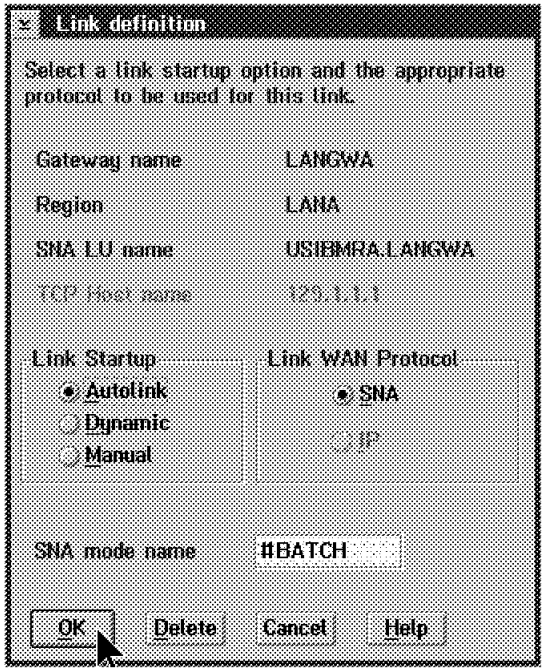


Figure 113. Link Parameters

Use Autolink for very frequently used links, Dynamic for links that should be activated whenever access to a machine in the destination LAN (determined by the name qualifiers) is requested and Manual for backup or very costly links.

When there are no more links to be defined, select **Done** as shown in Figure 114 on page 149 and you will return to the modified main configuration screen. This screen now shows lines between the selected LAN Gateway (LANGWx) and all partners. So you can easily check if you forgot a partner LAN Gateway.

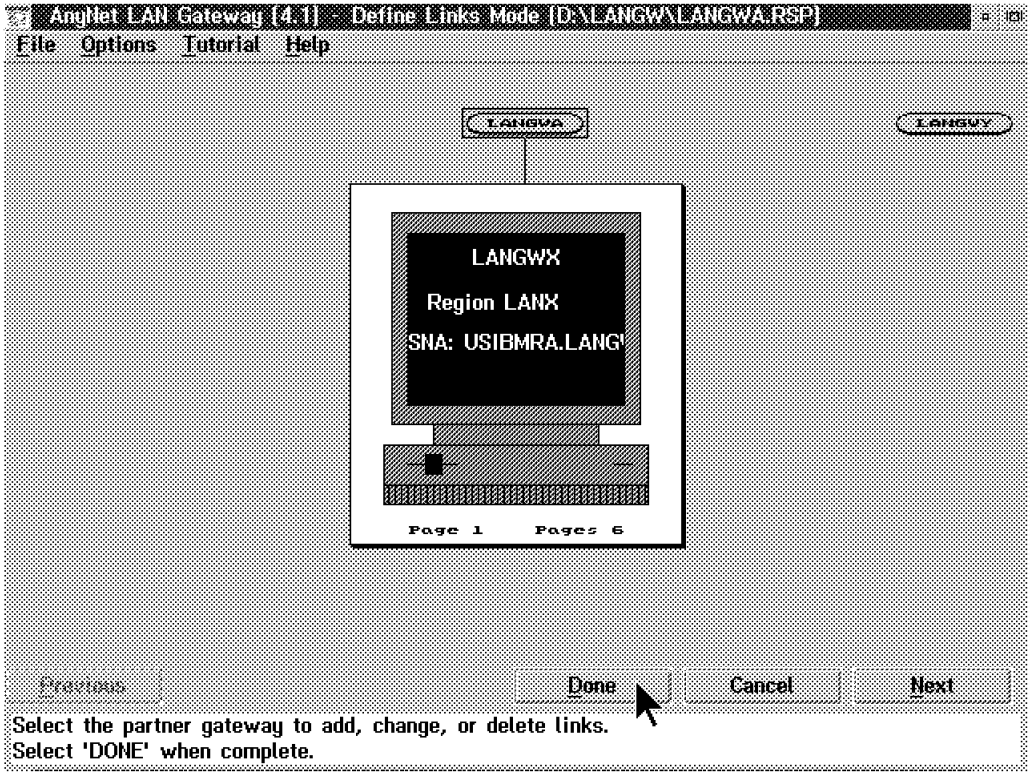


Figure 114. Defined Links for LANGWx

7.4.2 Defining the Links between LAN Gateway LANGWy and Its Partner(s)

You may start configuring the links between this LAN Gateway and any number of other LAN Gateways by entering the proper options in the configuration panels as the following screens indicate.

First you double-click on **LANGWy**, which brings up the screen shown in Figure 115 on page 150.

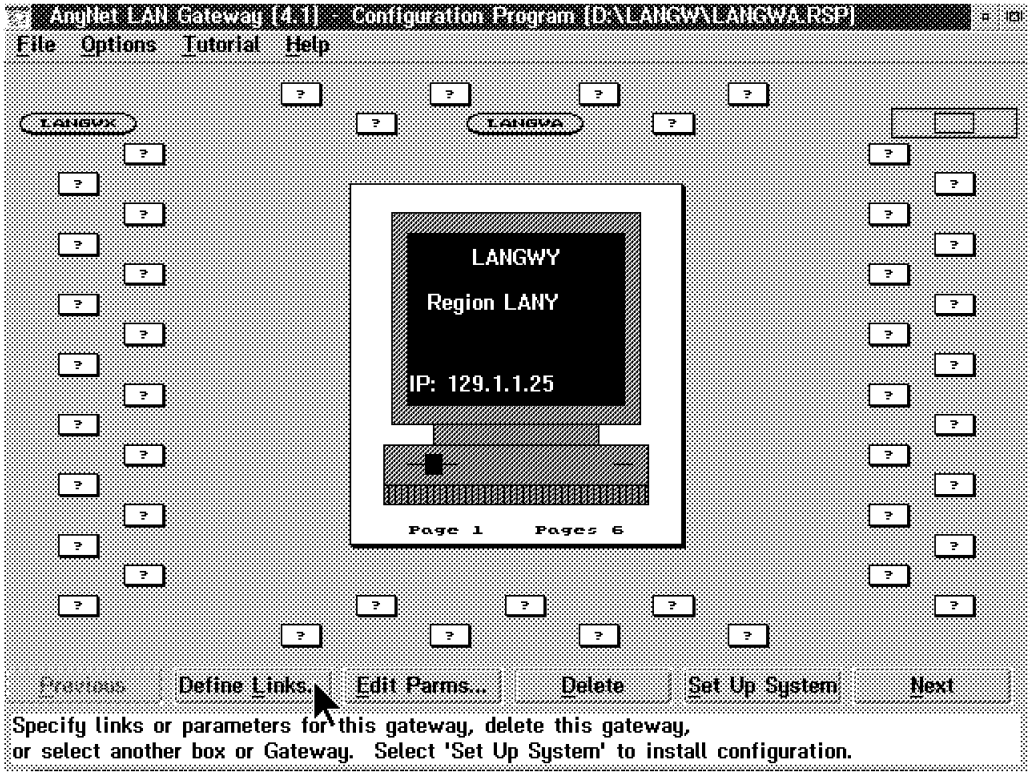


Figure 115. Defining Links from LANGWy to Other LAN Gateway(s)

Clicking on **Define Links** displays the panel shown in Figure 116 on page 150.

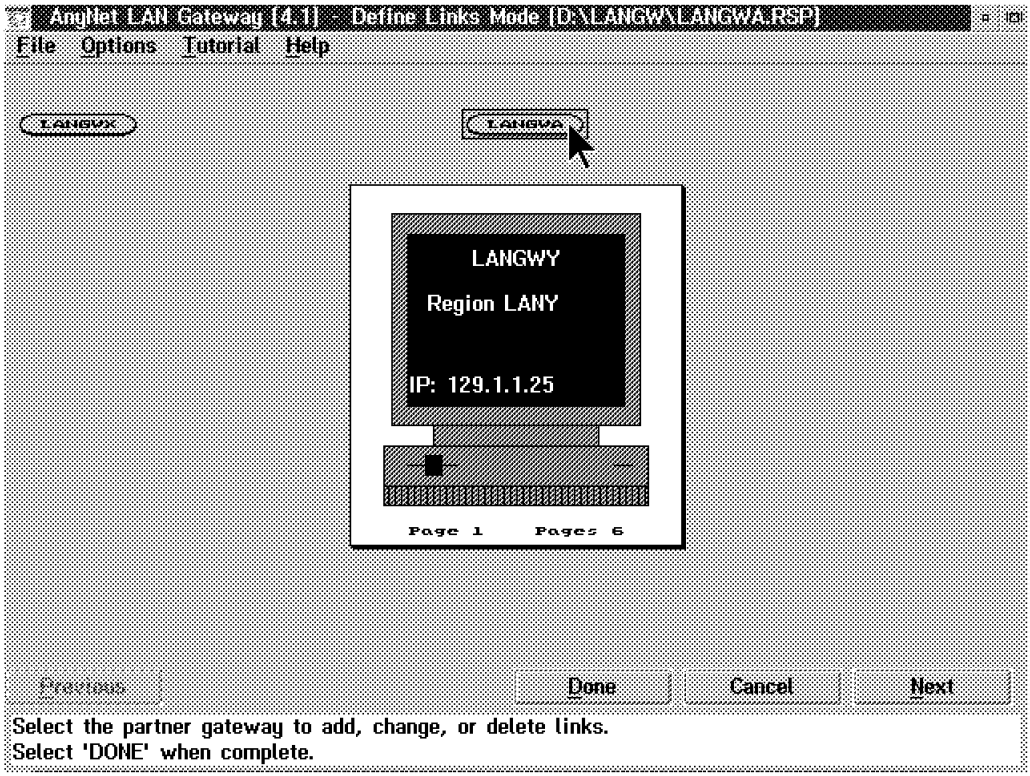


Figure 116. Selecting Partner LAN Gateway(s)

Double-click on the LAN Gateway to which you want to create a link and fill out the panel shown in Figure 117 on page 151.

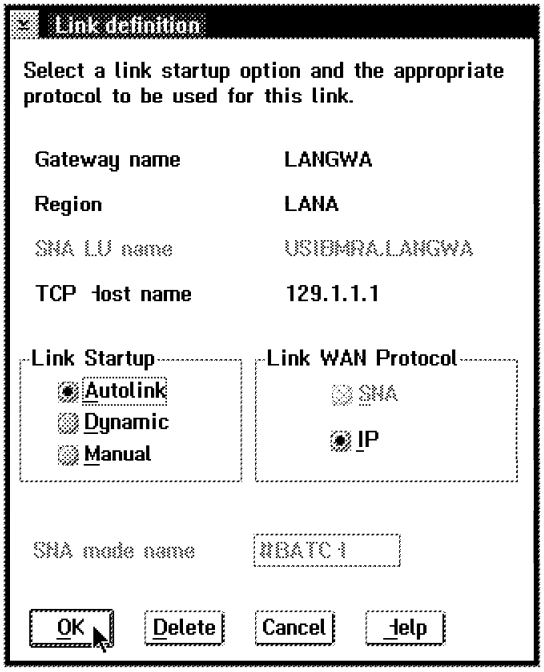


Figure 117. Link Parameters

Use Autolink for very frequently used links, Dynamic for links that should be activated whenever access to a machine in the destination LAN (determined by the name qualifiers) is requested and Manual for backup or very costly links.

When there are no more links to be defined (as in our scenario) select **Done** as shown in Figure 118 on page 152 and you will return to the modified main configuration screen. This screen now shows lines between the selected LAN Gateway (LANGWy) and all partners. So you can easily check if you forgot a partner LAN Gateway.

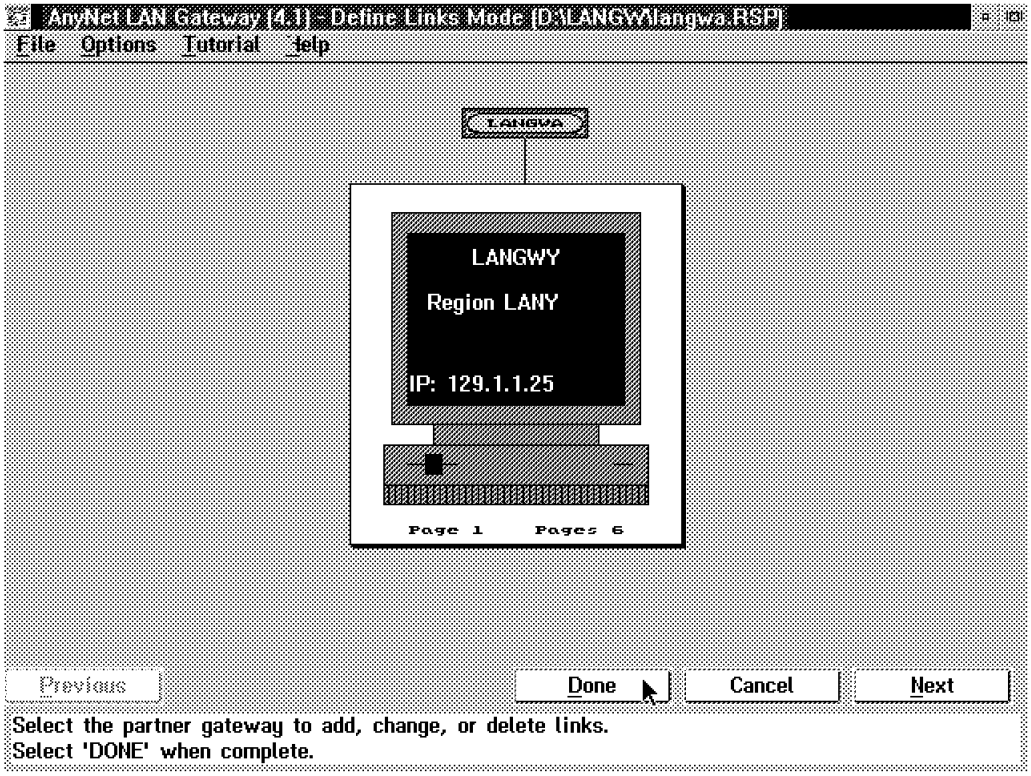


Figure 118. Defined Links for LANGWy

7.4.3 Defining the Links between LAN Gateway LANGWa and Its Partner(s)

In our scenario, LAN Gateway LANGWa is the *root* of the network. Both LAN Gateways LANGWx and LANGWy have their links already defined to LANGWa. If you now double-click on the **LANGWa** icon shown in Figure 119 on page 153, a panel showing all of the defined links will be displayed (see Figure 120 on page 153).

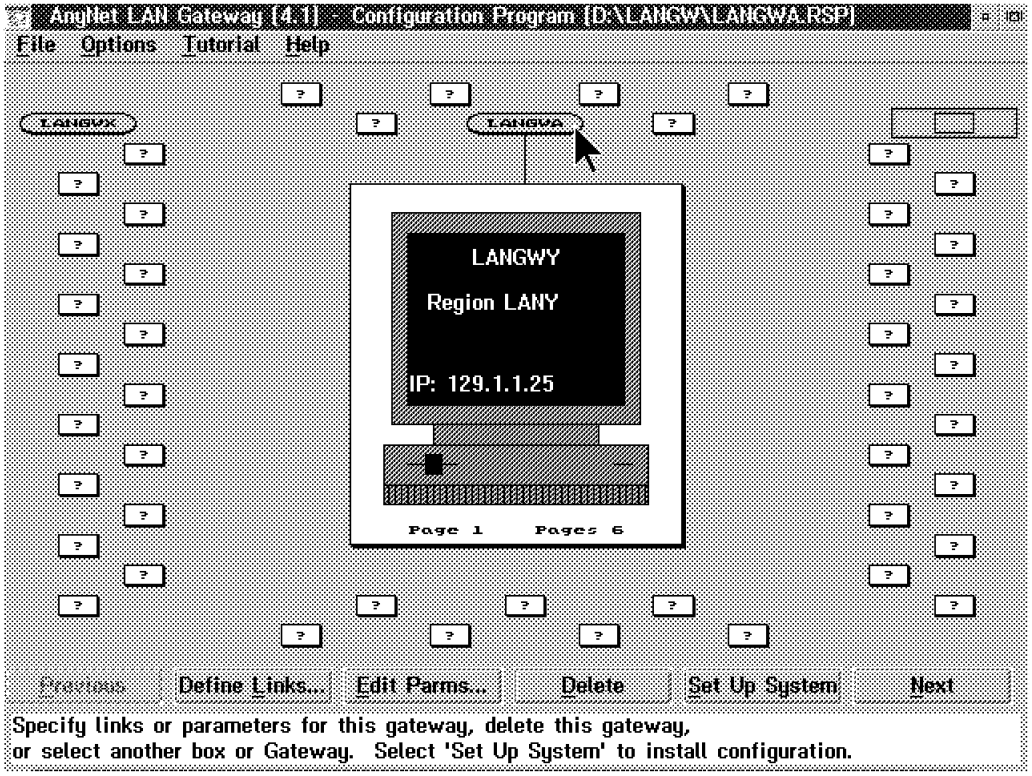


Figure 119. Main Configuration Screen with LANGWy Selected (Last Selection)

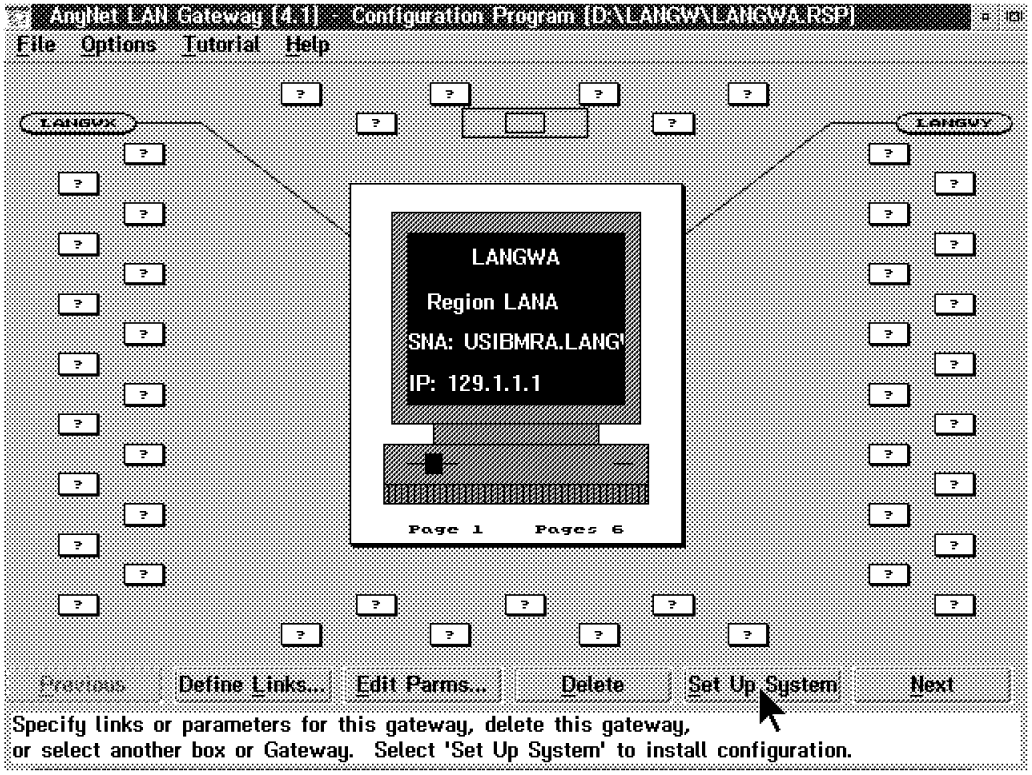


Figure 120. LANGWa and Its Links

You are now ready to set up all of your LAN Gateways. This is described in detail in the following section.

7.5 Setting Up the LAN Gateways

In this section, we show you the configuration panels required to define a LAN Gateway.

7.5.1 Setting Up System: LANGWa

Start the setup of the LAN Gateways by clicking on the **Set Up System** push button as shown in Figure 120 on page 153. This gives you an informational message that you must acknowledge (see Figure 121 on page 154).

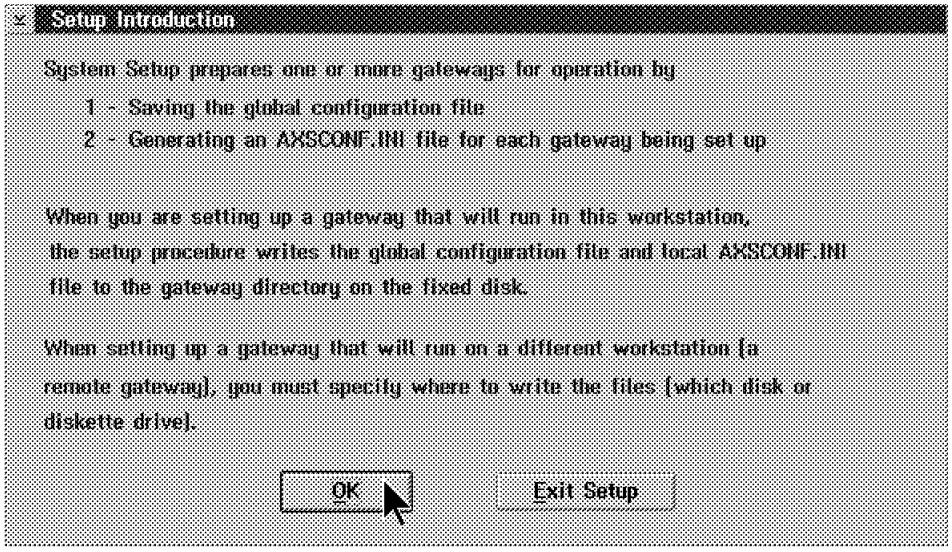


Figure 121. Setup Introduction

Select **OK** and the Type of Setup panel is displayed (Figure 122 on page 154).

Figure 122. Type of Setup

For LAN Gateway LANGWa you need to select **Set up for gateway in this workstation**. The second choice is basically needed when you have only changed a parameter on one of the remote LAN Gateways that does not affect the links.

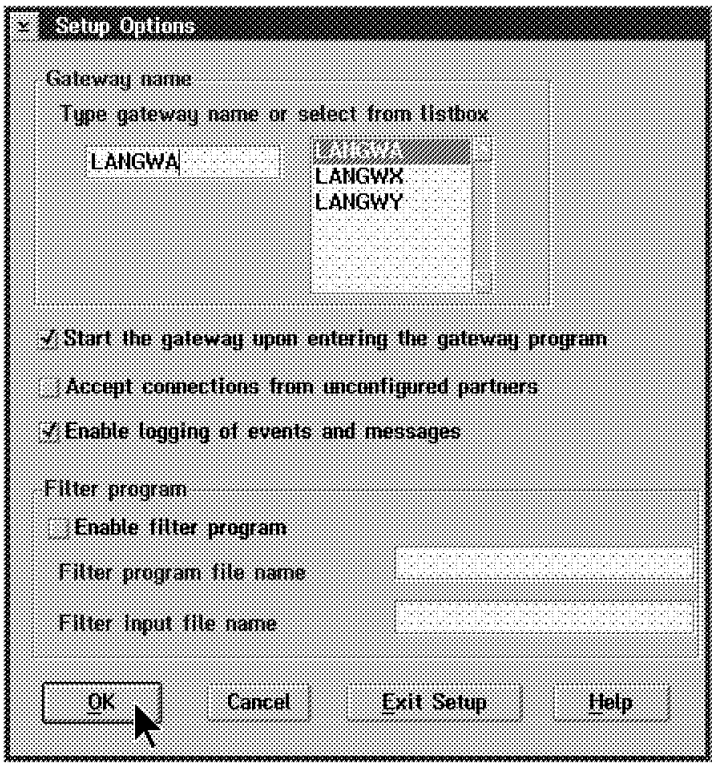


Figure 123. Setup Options for LAN Gateway LANGWa

In Figure 123 on page 155 you may specify additional parameters for the LAN Gateway you select in the list box of the available (defined) LAN Gateways. The selection should already be the local LAN Gateway (LANGWa).

In our scenario, we selected **OK** to proceed to Figure 124 on page 155.

Figure 124. Logging Options

You can accept the defaults or change the file names to tailor them to your own naming conventions.

By clicking on the **OK** button you are confronted with the panel shown in Figure 125 on page 156 and the selection: which IPX networks you are going to advertise to the partner LAN Gateways. This means that you need to get these values from the NetWare server administrator for this LAN (unless you are responsible for that also).

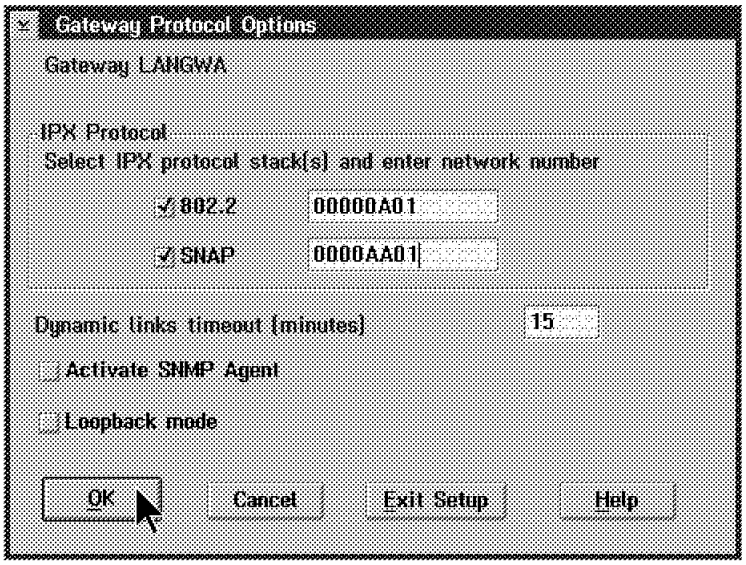


Figure 125. Gateway Protocol Options (IPX)

Specify if you are going to use the loopback mode (discussed in 5.1.4, “Running NetBIOS and IPX Applications on the LAN Gateway Using the Loopback Mode” on page 107 and Appendix C, “Loopback Mode” on page 307).

The next two panels (Figure 126 on page 157 and Figure 127 on page 157) show the configured parameters and allow for a final modification.

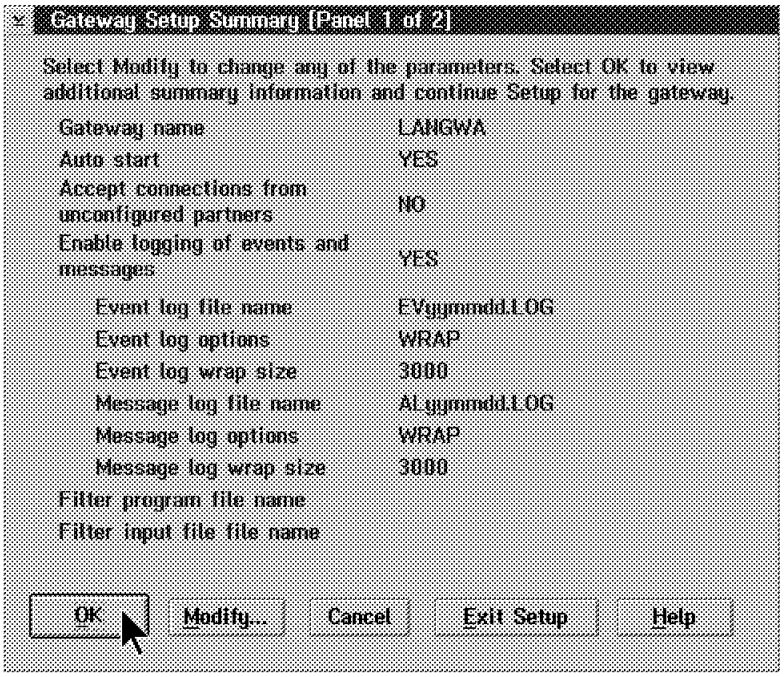


Figure 126. Gateway Setup Summary for Review of the Parameters (Panel 1 of 2)

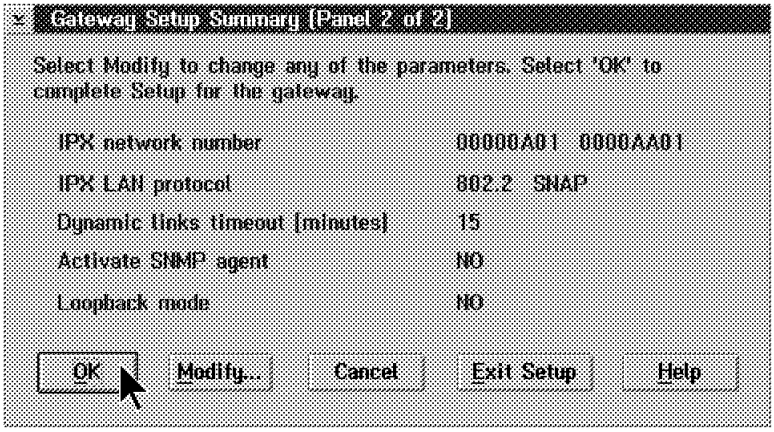


Figure 127. Gateway Setup Summary for Review of the Parameters (Panel 2 of 2)

After clicking on **OK** on the second panel (Figure 127 on page 157) an information pop-up window is displayed (Figure 128 on page 157). Click on **OK** and Figure 129 on page 158 appears.

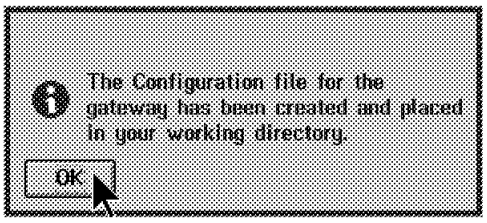


Figure 128. Your LANGWa Setup Was Successful

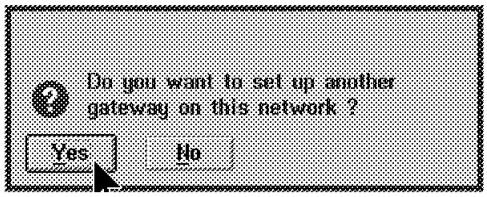


Figure 129. Setting Up Another Gateway

As we have two LAN Gateways left to configure, you need to select **Yes** on this panel. This takes you back to the Setup Options panel shown in Figure 130 on page 158.

7.5.2 Setting Up System: LANGWx

In Figure 130 on page 158 you may specify additional parameters for the LAN Gateway you select in the list box of the available (defined) LAN Gateways. The selection should already be the first remote LAN Gateway (LANGWx).

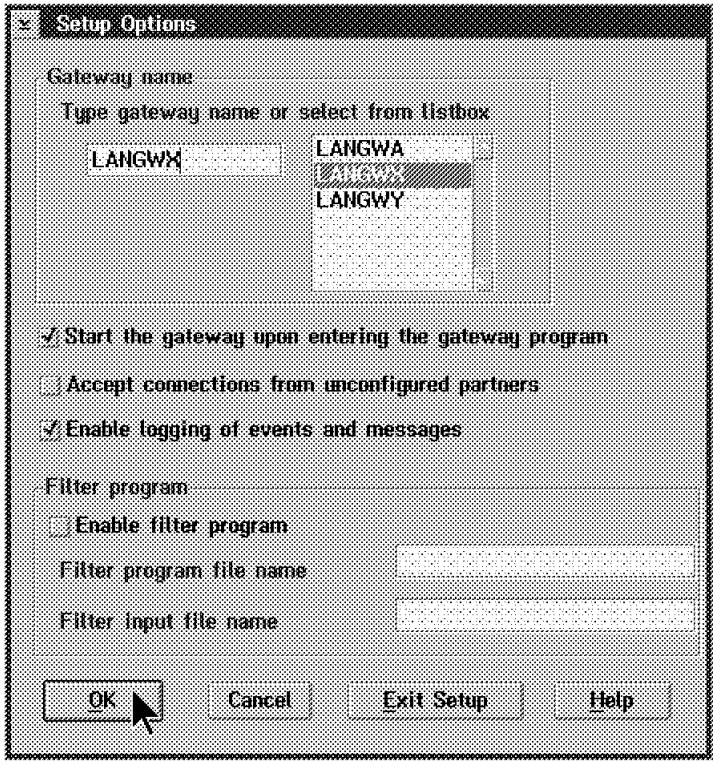


Figure 130. Setup Options for LAN Gateway LANGWx

Clicking on **OK** will take you to the Gateway Protocol Options panel (Figure 131 on page 159), which differs from Figure 125 on page 156 as LANGWx is only attached to a NetBIOS LAN and, therefore, has no IPX parameters to be set.

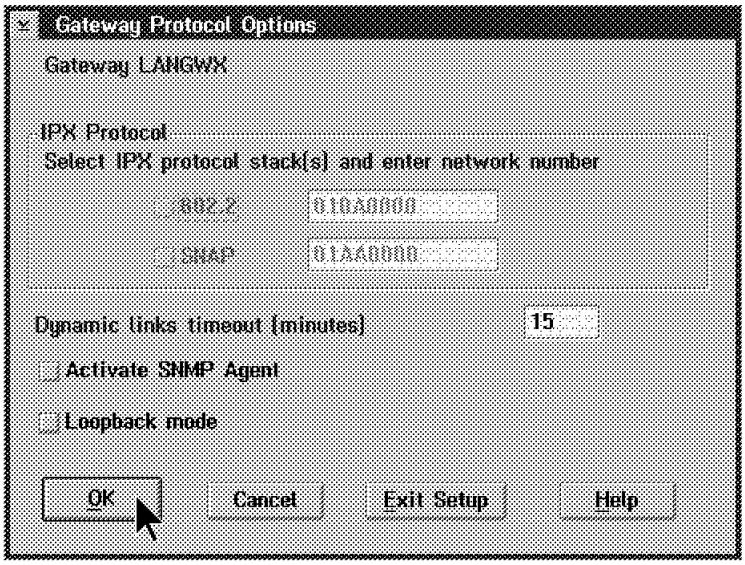


Figure 131. Gateway Protocol Options (NetBIOS)

The following two panels are for reviewing the configuration parameters and usually no change is required.

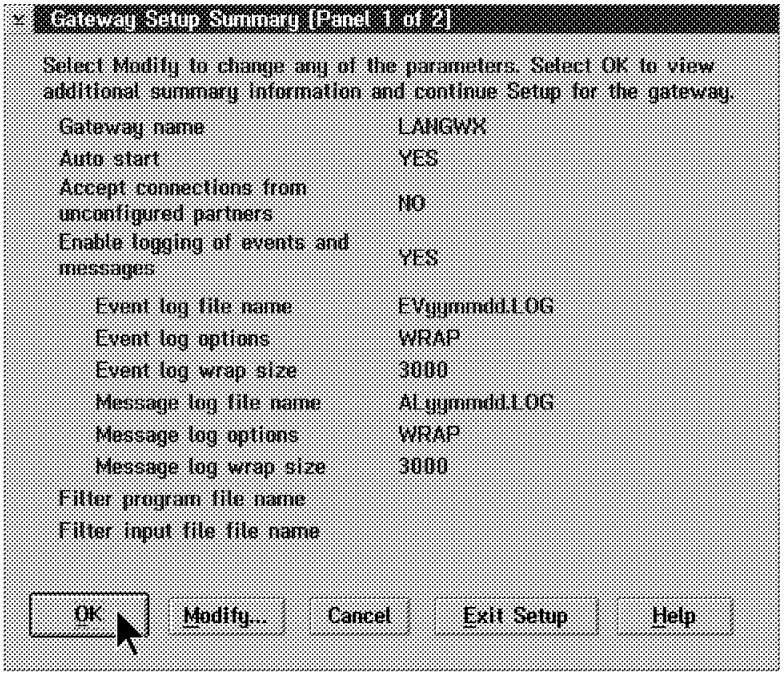


Figure 132. Gateway Setup Summary for Review of the Parameters (Panel 1 of 2)

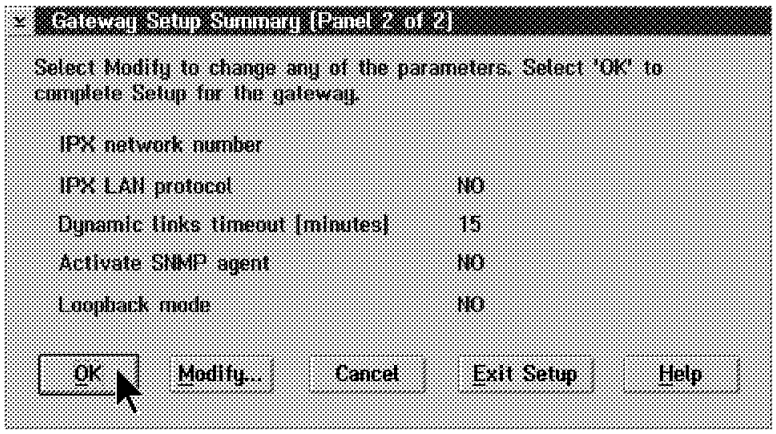


Figure 133. Gateway Setup Summary for Review of the Parameters (Panel 2 of 2)

As this setup is created for LAN Gateway LANGWx using the master configuration method on LAN Gateway LANGWa, you need to specify a path in which you want to save the configuration for LAN Gateway LANGWx.

You are asked for this path on the panel shown in Figure 134 on page 160. This target drive/directory may be a diskette, a hard disk or a redirected LAN drive.

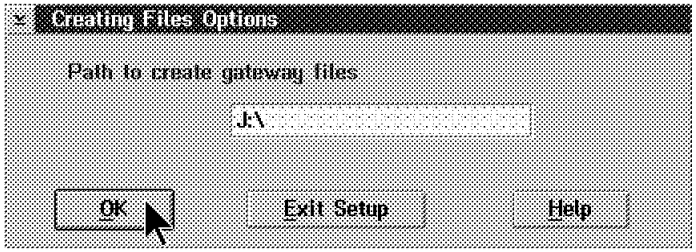


Figure 134. Destination Path for LANGWx Configuration Files

After some disk activity you will find two files on the destination path:

- AXSCONF.INI
- LANGWA.RSP

You may rename the .RSP file to the name of the target LAN Gateway to reduce confusion, as this file is named identical to the .RSP file of the configuring LAN Gateway.

Next you will see the panel shown in Figure 135 on page 160. You should select **Yes** as there is LAN Gateway LANGWy still left to configure.

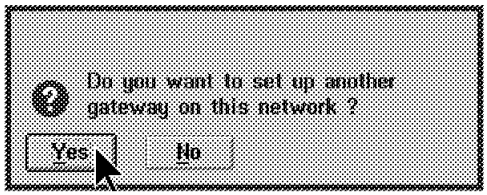


Figure 135. Setting Up Another Gateway

This takes you back to the Setup Options panel shown in Figure 136 on page 161.

7.5.3 Setting Up System: LANGWy

In Figure 136 on page 161 you may specify additional parameters for the LAN Gateway you select in the list box of the available (defined) LAN Gateways. The selection should be changed to LAN Gateway LANGWy.

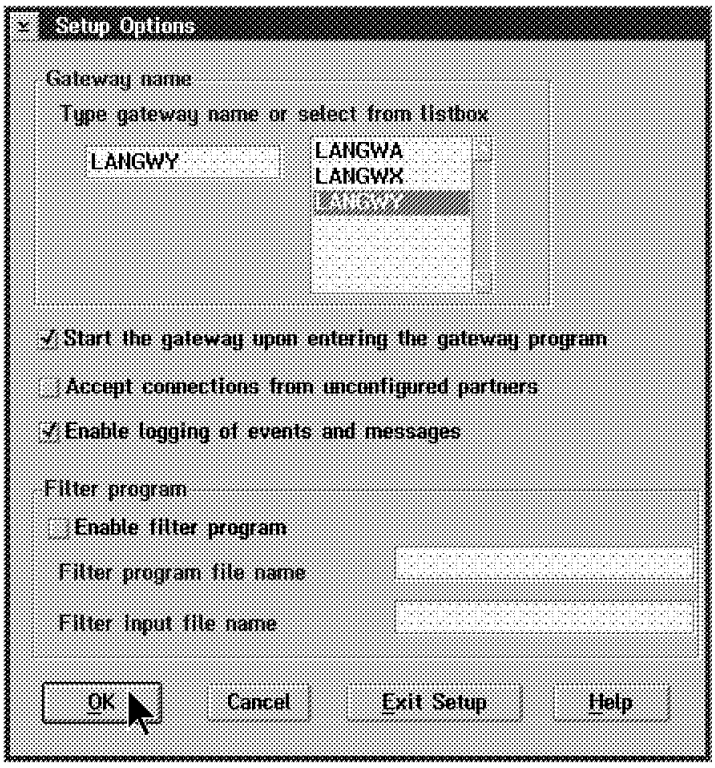


Figure 136. Setup Options for LAN Gateway LANGWy

Clicking on **OK** will take you to the Gateway Protocol Options panel (Figure 137 on page 161) where you specify which IPX networks you are going to advertise to the partner LAN Gateways.

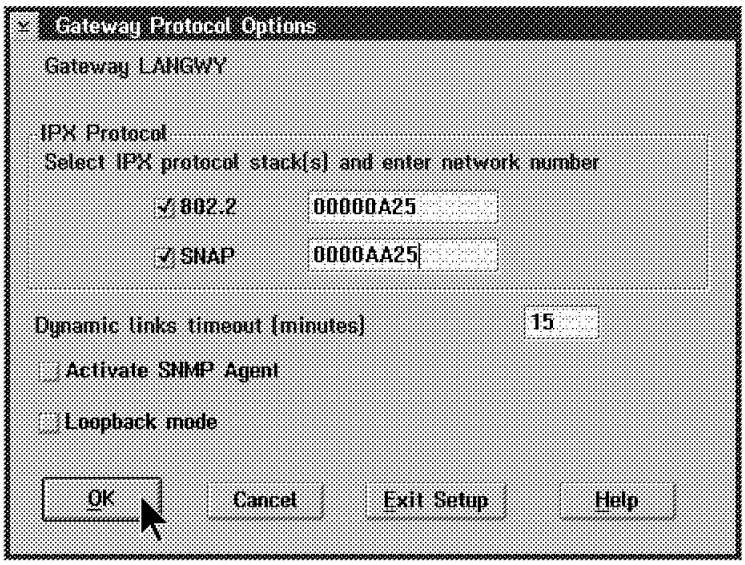


Figure 137. Gateway Protocol Options (IPX)

This means that you need to get these values from the NetWare server administrator for this LAN (unless you are responsible for that too).

Note: This IPX network number *must* be different from the one used in LANGWa.

The following two panels are for reviewing the configuration parameters and usually no change is required.

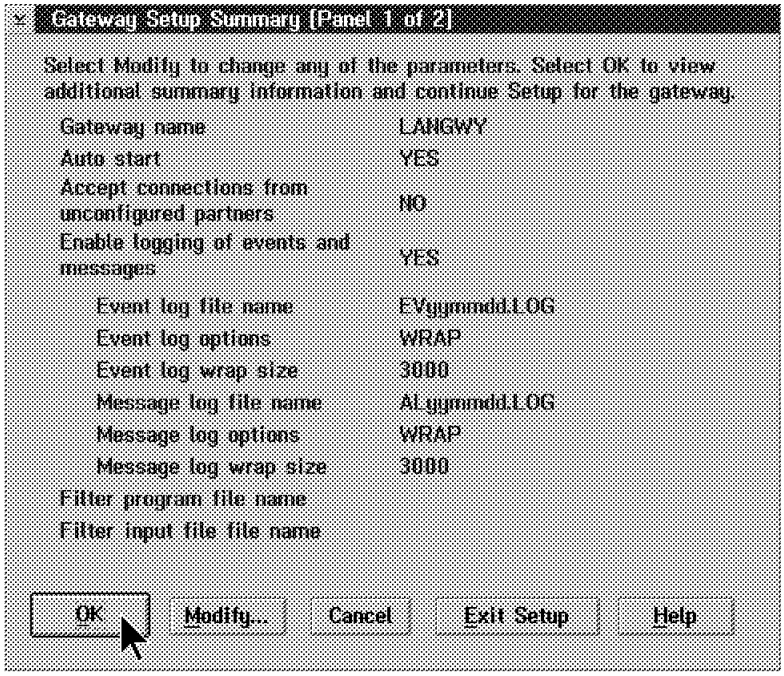


Figure 138. Gateway Setup Summary for Review of the Parameters (Panel 1 of 2)

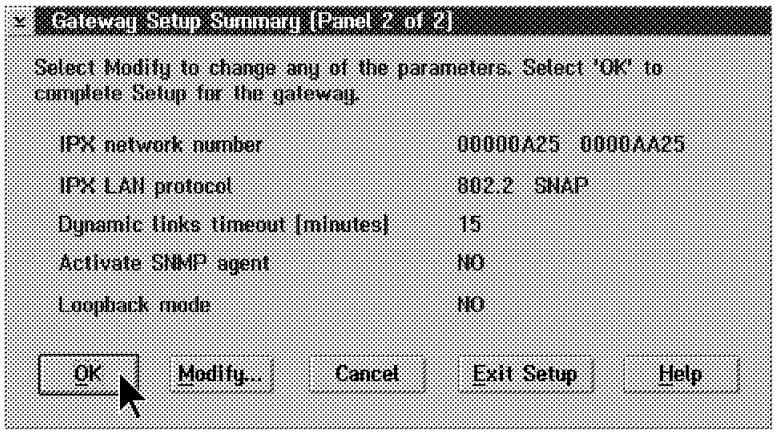


Figure 139. Gateway Setup Summary for Review of the Parameters (Panel 2 of 2)

As this setup is created for LAN Gateway LANGWy using the master configuration method on LAN Gateway LANGWa, you need to specify a path in which you want to save the configuration for LAN Gateway LANGWy.

You are asked for this path on the panel shown in Figure 140 on page 163. This target drive/directory may be a diskette, a hard disk or a redirected LAN drive.

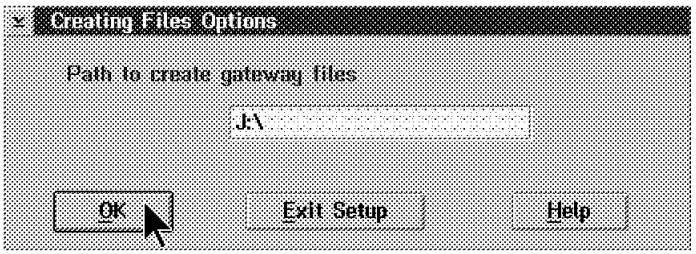


Figure 140. Destination Path for the LANGWx Configuration Files

After some disk activity you will find two files on the destination path:

- AXSCONF.INI
- LANGWA.RSP

You may rename the .RSP file to the name of the target LAN Gateway to reduce confusion, since this file is named identical to the .RSP file of the configuring LAN Gateway.

Next you will see the panel shown in Figure 135 on page 160. Now you should select **No** because there is not another LAN Gateway left to configure (considering this particular scenario).

This takes you back to the End of Setup panel shown in Figure 141 on page 163. If you want to try this setup, click on **No** and you will be returned to the LAN Gateway folder.

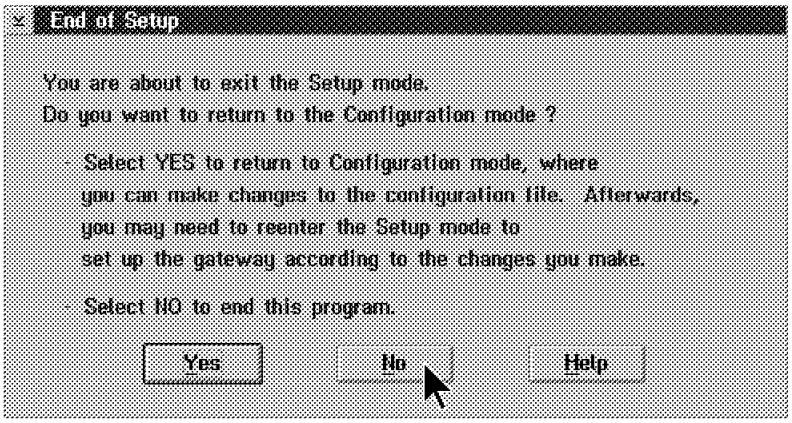


Figure 141. End of Setup

7.6 Starting the LAN Gateway

You may start LAN Gateway LANGWa immediately by double-clicking on the **Start** icon in the LAN Gateway folder. The partner LAN Gateways need to be installed, and the configuration files created above need to be copied to them. If you have all the WAN functionalities up and running (IBM Communications Server Release 4.1 for SNA WANs and the IBM Operating System/2 (OS/2) TCP/IP for TCP/IP WANs), the complete scenario should run correctly.

Part 6. Dependent LU Support

Chapter 8. Multiple PU Support on a Single SDLC Link

In this chapter, we describe the new function in IBM Communications Server Release 4.1 that allows you to define multiple physical units (PUs) in the same machine. The PUs are defined as secondary multidrop to a primary node. The main benefit of this function is the capability of defining dedicated PUs (downstream PU visibility) with no DLUR requirement. However, it is available for SDLC links only.

8.1 Introduction

Multiple PU over a single SDLC link was added in IBM Communications Server Release 4.1.

With DLUR/DLUS the number of SSCP-PU sessions over a link is unlimited. However, many users are slow in moving to the APPN DLUS capability in their host systems. In order to save line costs and share the line for *multiple downstream PUs*, the multiple PU over an SDLC line was added.

This function is especially important if the downstream workstations must be visible from the host. 470x banking workstations downstream are a good example.

using the gateway The configuration has a new SDLC Type as *Multiple PU over SDLC*, and it acts like multiple secondary link stations attached to a multidrop line.

If your host does not have dependent LU server (DLUS) defined, you can define multiple PUs on a single SDLC link through the link station role parameter on the SDLC profile for a workstation. This method gives you multiple PUs over the same physical link, but dependent LU requester (DLUR) is preferred, because it provides these advantages:

- Switched backup
- More than 16 PUs per SDLC
- APPN benefits

However, multiple PU support on a single SDLC link gives you the advantage of being able to have more than 254 LUs per physical link, because there are multiple PUs and each PU has up to 254 LUs.

Defining multiple PUs in this manner is recommended as a temporary solution until you can migrate your network to APPN.

Figure 142 on page 168 shows multiple Communications Server workstations connected to the host through a single SDLC link.

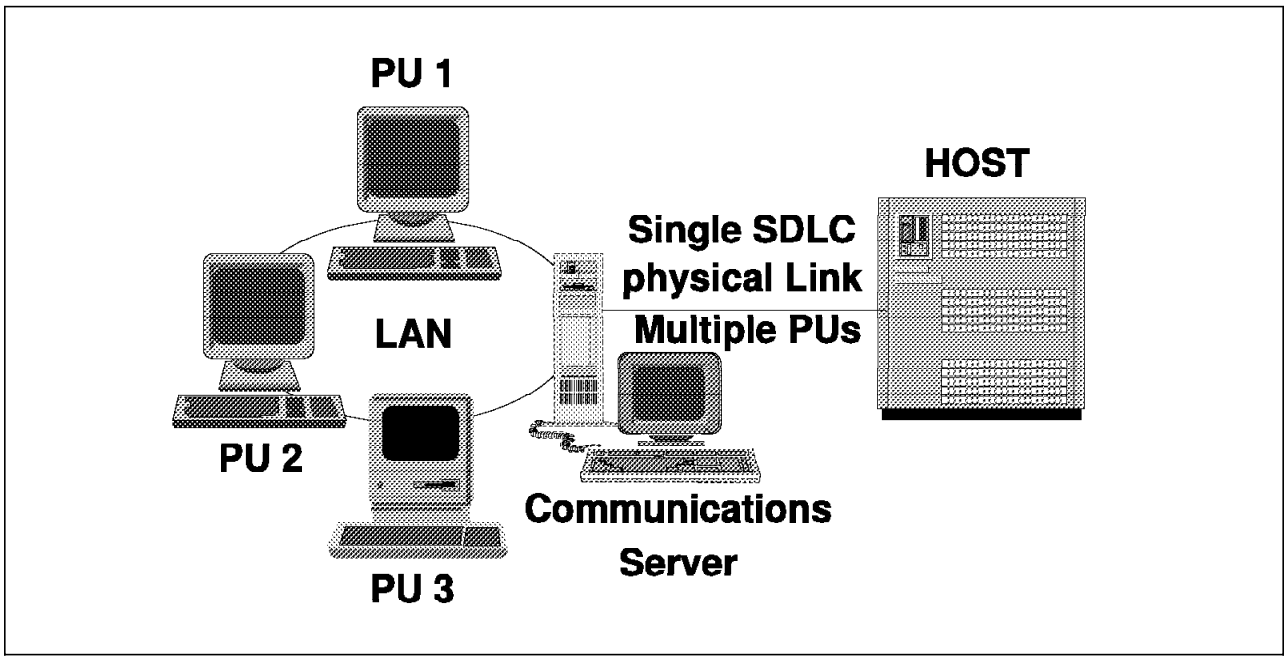


Figure 142. Multiple PU Support on a Single SDLC Link

8.2 Enhancements Related to SDLC Support in Communications Server 4.0

The following are the enhancements related to the SDLC support in Communications Server 4.0:

- **High-Speed SDLC**

  CM supports at least one SDLC line at T1/E1 (12 Mbps) speed over the WAC adapter.

- **Full-duplex data transmission mode**

  The previous implementation had been limited to support SDLC in two-way alternating mode, because the MPA adapter only had a *half-duplex* interface. Since other supported adapters can now take advantage of *full-duplex*, the support has been added on all SDLC connections to better utilize the media.

- **Additional SDLC lines**

  With previous releases of CM/2, SDLC connections were limited to two. This limitation was mainly due to the fact that initially, only the MPA adapter was supported and the maximum number of adapters allowed in a system was two.

  With the advent of SNA Phone Connect and the possibility of using adapters such as the WAC and ARTIC adapter, the need for more than two links, especially for server/gateway machines with a need to support multiple downstream connections, became more urgent.

  The number of SDLC lines supported is therefore increased from two to at least 16 lines, in any mix of upstream and downstream lines. The number 16 is a test statement, not an implementation limit.

- **Multipoint Primary Support**

Up to now, CM/2 could only be configured as a secondary station in a multipoint configuration. This new feature now allows CM/2 to act as a primary station and support up to 16 downstream multipoint SDLC lines.

Multipoint lines are leased connections with multiple workstations on a single line. This support is important for gateways with downstream workstations and network nodes with attached end nodes or LEN nodes.

These configurations are common in European countries and Canada where communication lines are expensive. They generally consist of banking, retail, and point of sale systems downstream. This is for workstations connected by SDLC, which is supported by permanent connections over synchronous adapters such as IBM WAC, MPA and ARTIC adapters. This connection type supports gateway or server workstations connected to SDLC on a multidrop line.

Note that this is not meant to replace the NCP multipoint function; it is a low end alternative allowing the support of some of the older type of equipment, such as 4702s in a banking environment, for example, at a relatively low cost.

- **ARTIC as a multiple port adapter**

  An ARTIC NDIS MAC is supported to allow CM to support the ARTIC Portmaster type adapter (Portmaster/A for MicroChannel and Multiport Model 2 for ISA bus).

- Other miscellaneous enhancements are:
  - Allows use of switched and leased lines at the same time
  - Provides coexistence with LAN Distance
  - Supports programming of IRQ levels on ISA bus workstations (MPA1 for MPA for ISA and Microgate for DSA, and for Communications Port provides support for IRQ and Base Address)

8.3 Configuration of Dedicated PUs Using a Single SDLC Link

To configure for multiple PU support on a single SDLC link, you first have to define the parameters for the SNA Phone Connect Port Connection Manager.

1. Select the appropriate kind of modem connection from the Modem connection drop-down list box.

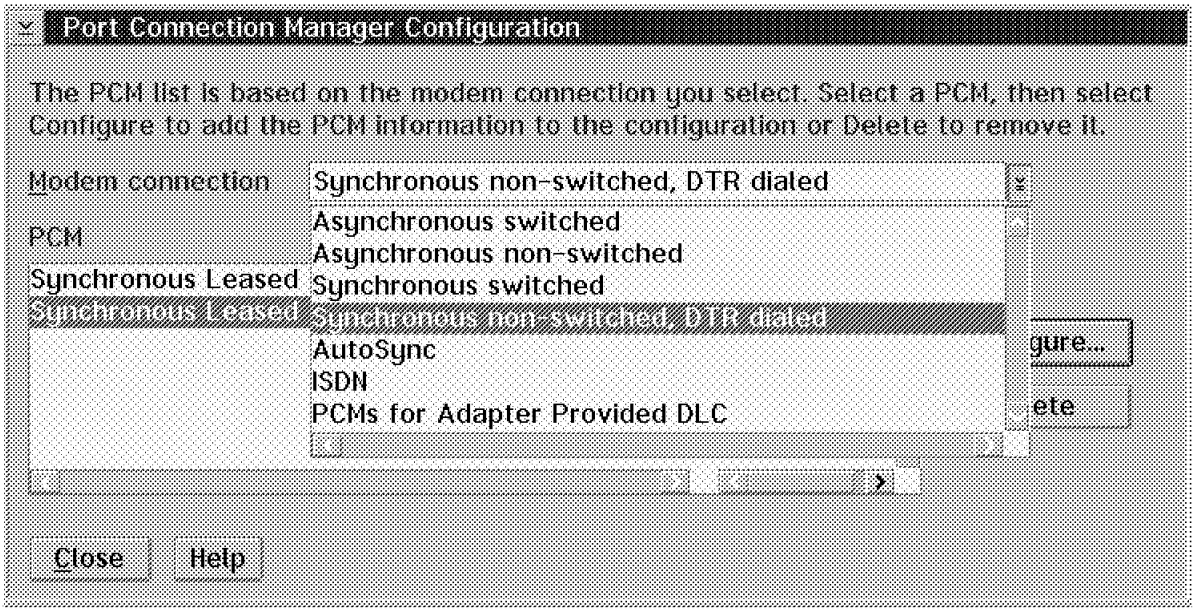


Figure 143. Port Connection Manager Definitions

2. Click on the chosen PCM and select **Configure**. The parameter screen for the selected modem connection appears.

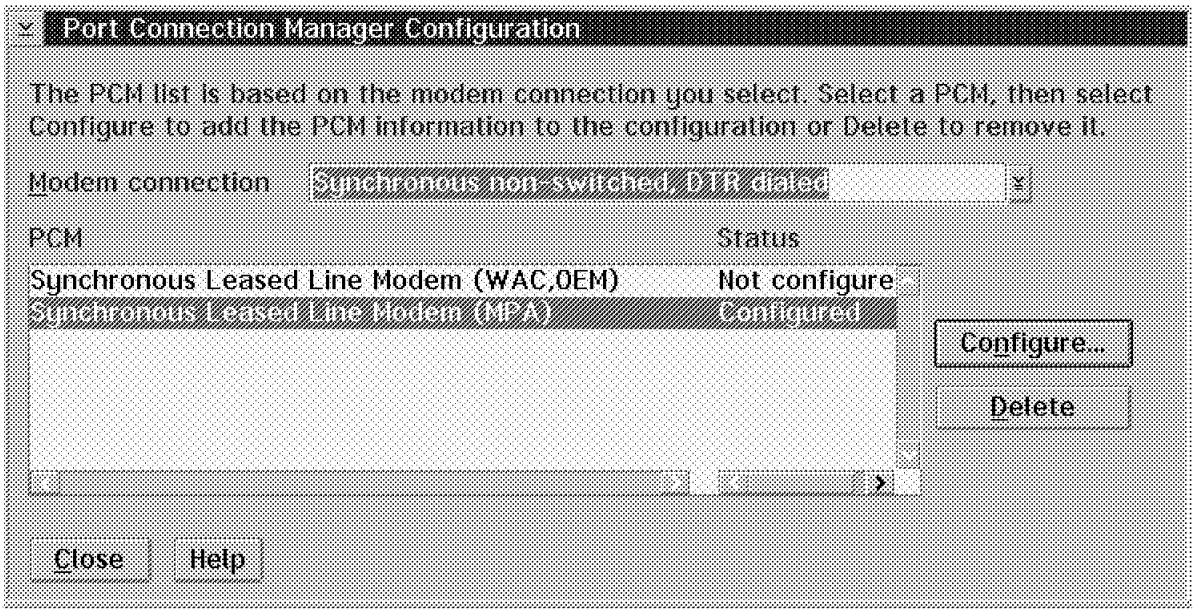


Figure 144. Port Connection Manager Configuration

3. Enter the appropriate parameters for Modem connection type, Port name, Permanent connection name and Encoding scheme. Click on **OK** when you are done.

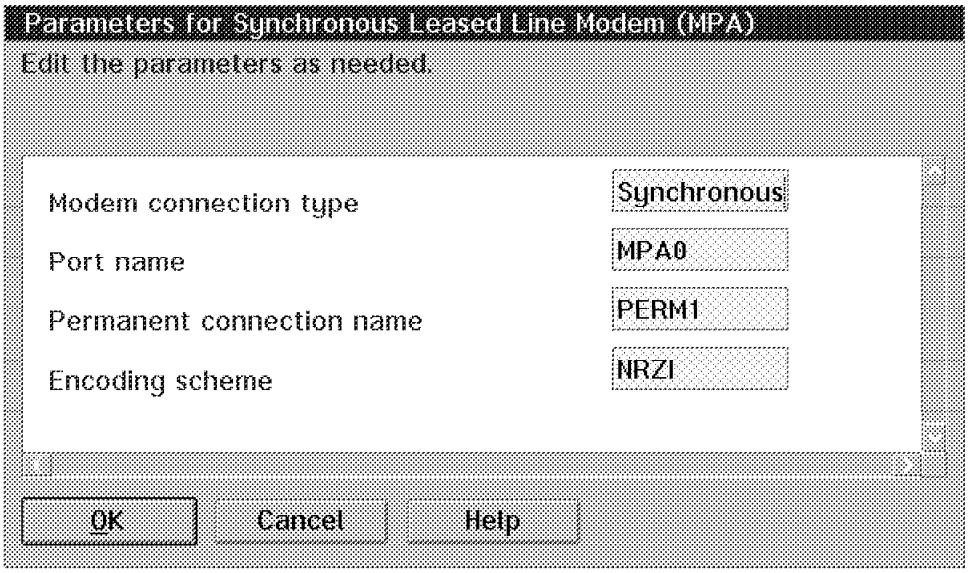


Figure 145. Port Connection Manager Configuration

Next, configure the SDLC DLC Adapter parameters.

4. In the Link station role drop-down list box select **Multiple PU Secondary**.

   Note that if this Link station role was selected, none of the additional parameters can be selected. All additional parameters remain as default.

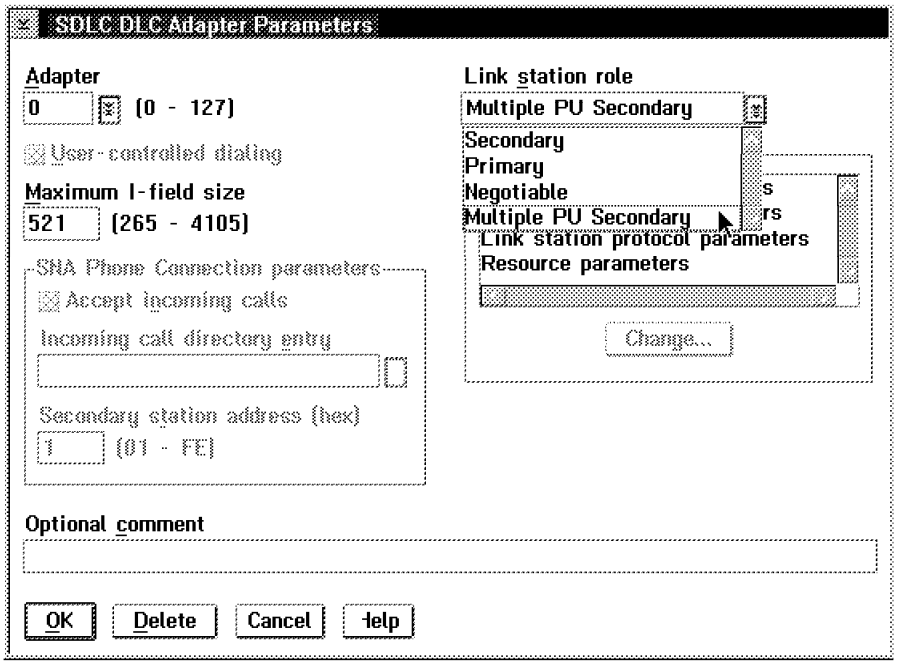


Figure 146. SDLC DLC Adapter Parameter

5. Configure your local node type as a network node and enter the requested local node parameters to meet your needs.

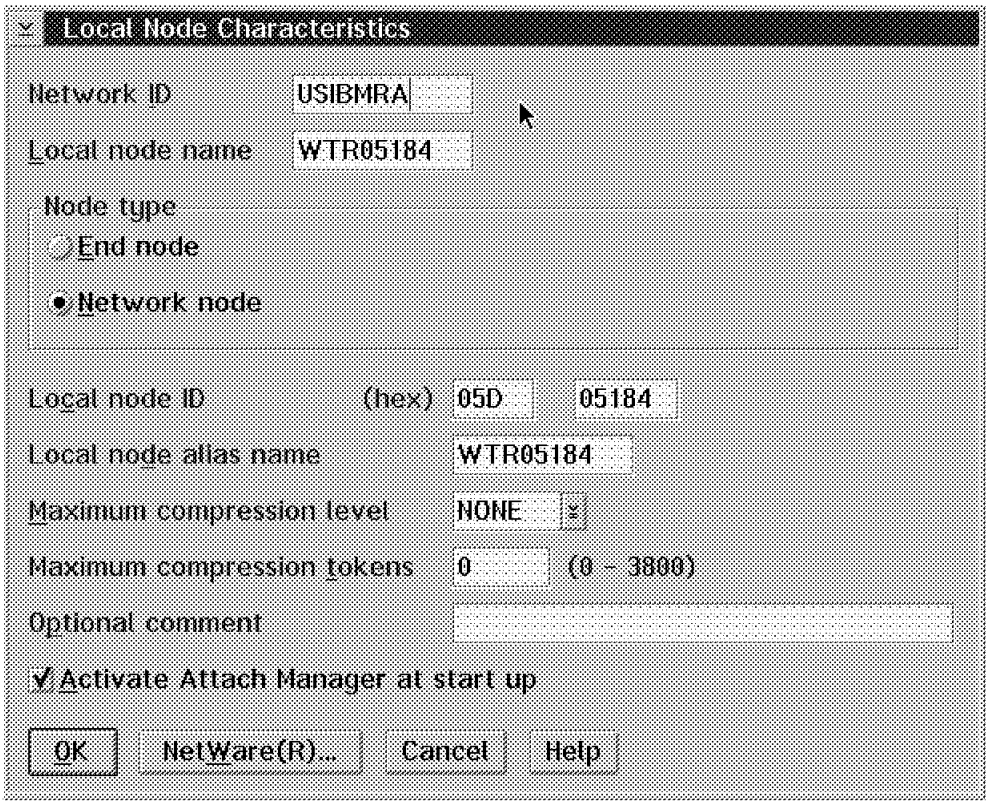


Figure 147. Local Node Characteristics

6. To configure a link to an end node type downstream workstation, select **To end node** in the Partner Type field as shown in Figure 148 on page 173. Click on **Create**.

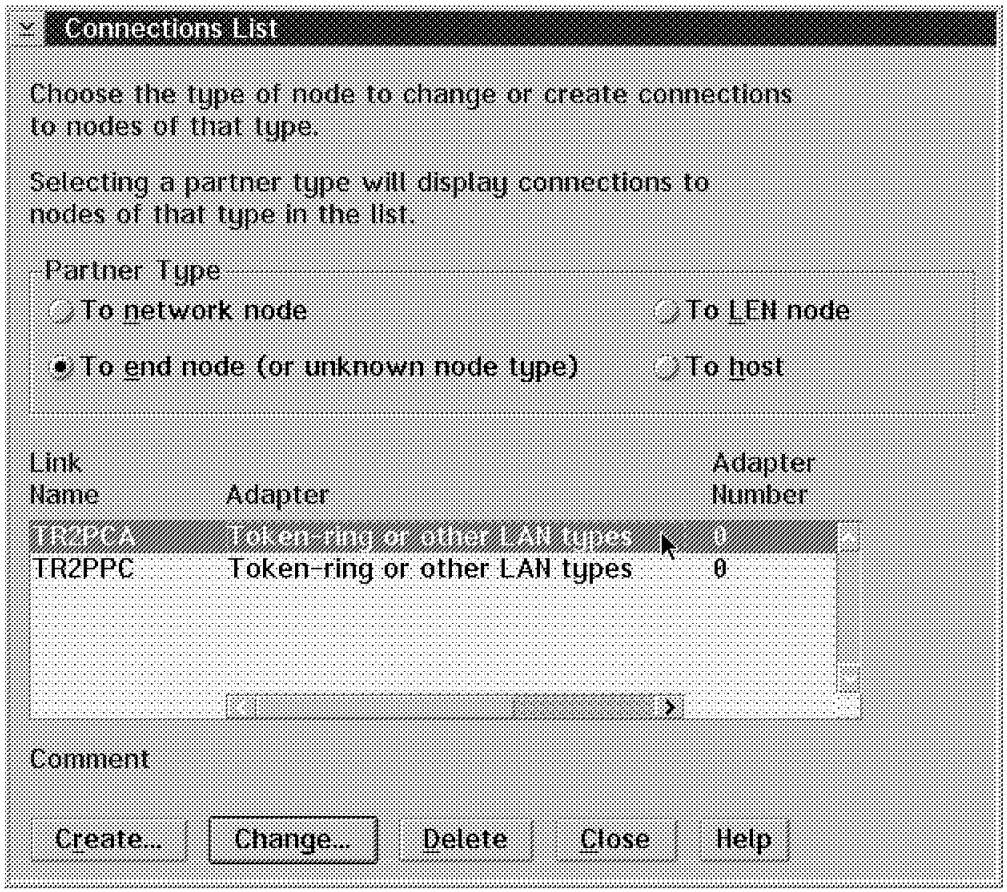


Figure 148. Connection List

7. From the Adapter List window select the type of the local adapter which is used for this connection.

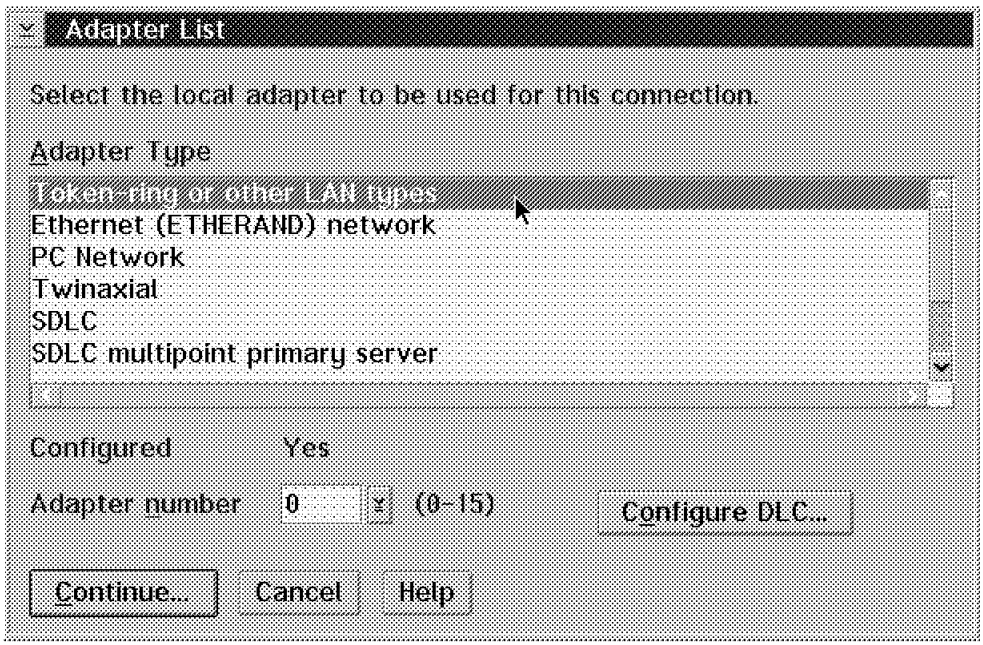


Figure 149. Adapter List

8. Enter the link name of the link to the downstream workstation and the adjacent node ID of the adjacent downstream workstation. No additional parameters are needed. Click on **OK** to finish the configuration of the downstream link.

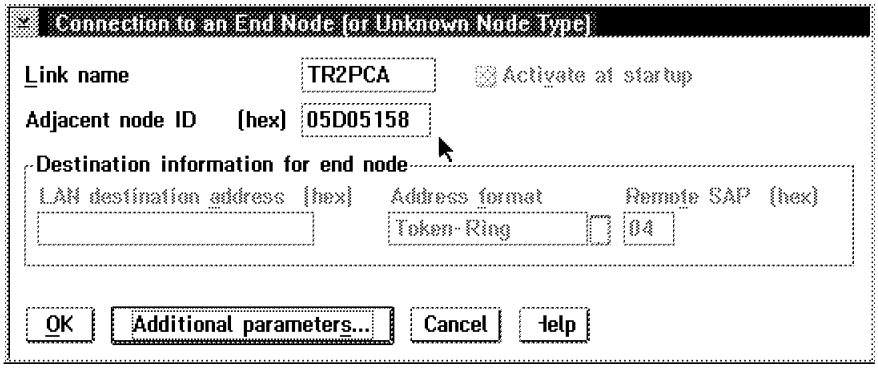


Figure 150. Connection to an End Node

9. To configure a link from the network node to your host system, select **To host** from the Partner Type field of Figure 151 and click on **Create**.

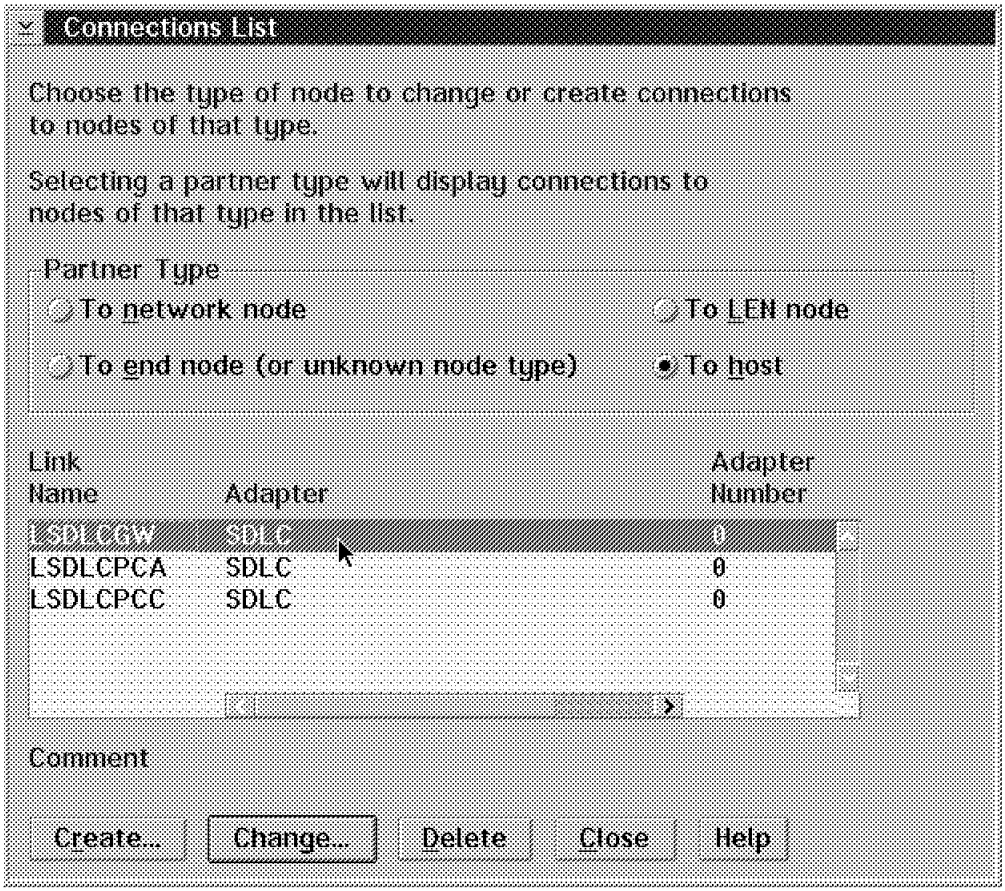


Figure 151. Connection List

10. From the Adapter List window select **SDLC** as the type of adapter used for the connection to the host.

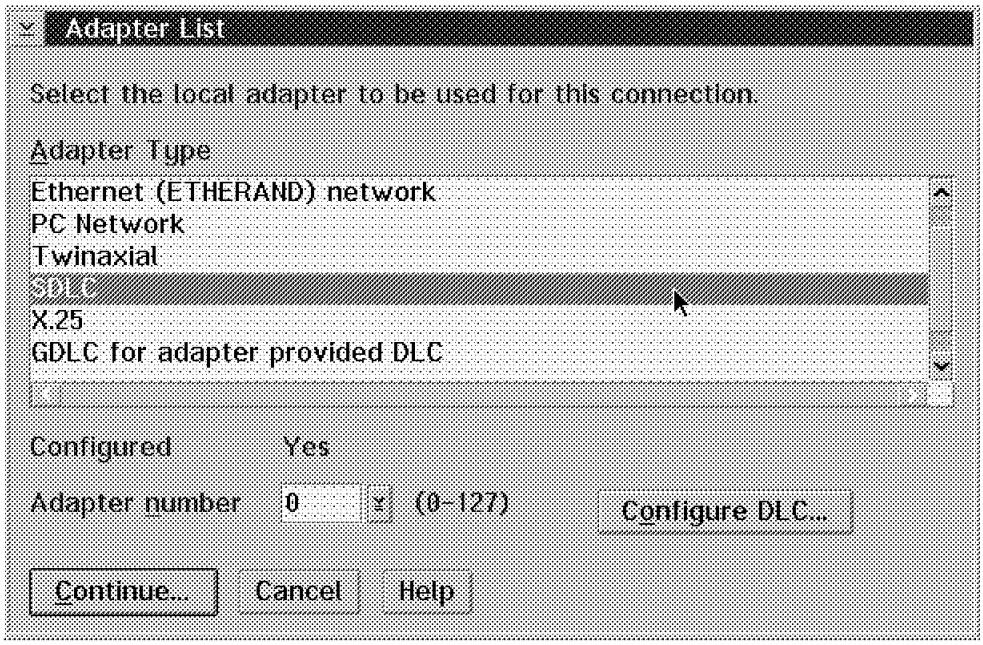


Figure 152. Adapter List

11. In the Connection to a Host window enter the name of the host link. The Secondary station address should be predefined in the VTAM definition. Please ask your VTAM coordinator for the exact definitions. For an example Line and PU definition, see Appendix F, "VTAM Line Description for Multiple PU over a Single SDLC Line" on page 315.

    The Permanent connection name is the name of the connection previously configured.

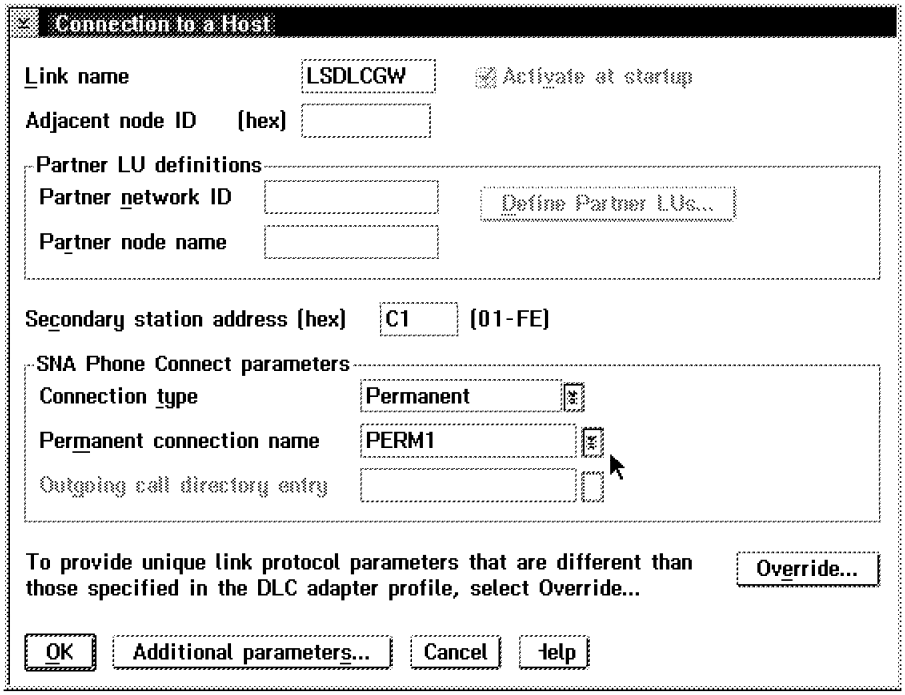


Figure 153. Connection to a Host

12. To configure the downstream workstation host link, click on **To host** in the Partner Type field of Figure 151 on page 174. Click on **Create** for further definitions.

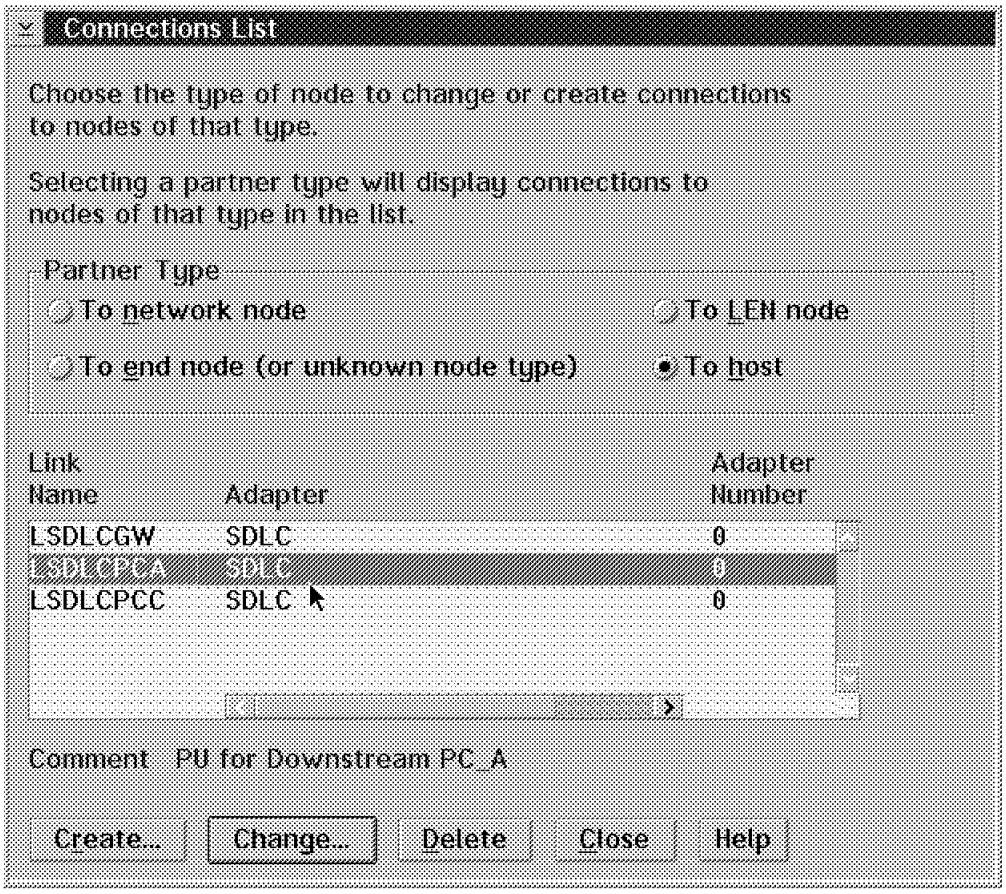


Figure 154. Connection to a Host

13. From the Adapter List window select SDLC as the type of adapter used for the connection from the downstream workstation to your host system. Click on **Continue**.

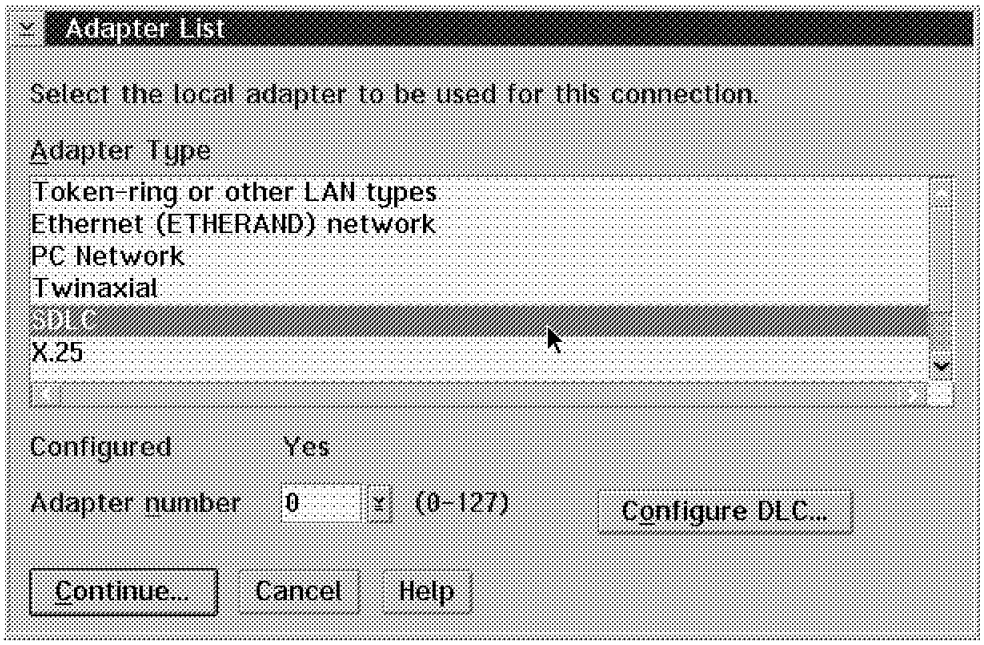


Figure 155. Adapter List

14. In the Connection to a Host window, enter the name of the downstream workstation link. The secondary station address should be predefined in the VTAM definitions. Please contact your VTAM coordinator. For an example Line and PU definition, see Appendix F, "VTAM Line Description for Multiple PU over a Single SDLC Line" on page 315. When the connection is made via one single SDLC link, the Permanent connection name remains the same as the one used in the definition of the network node to host link. Click on **Additional parameters** to define the multiple PU parameters for your downstream workstation.

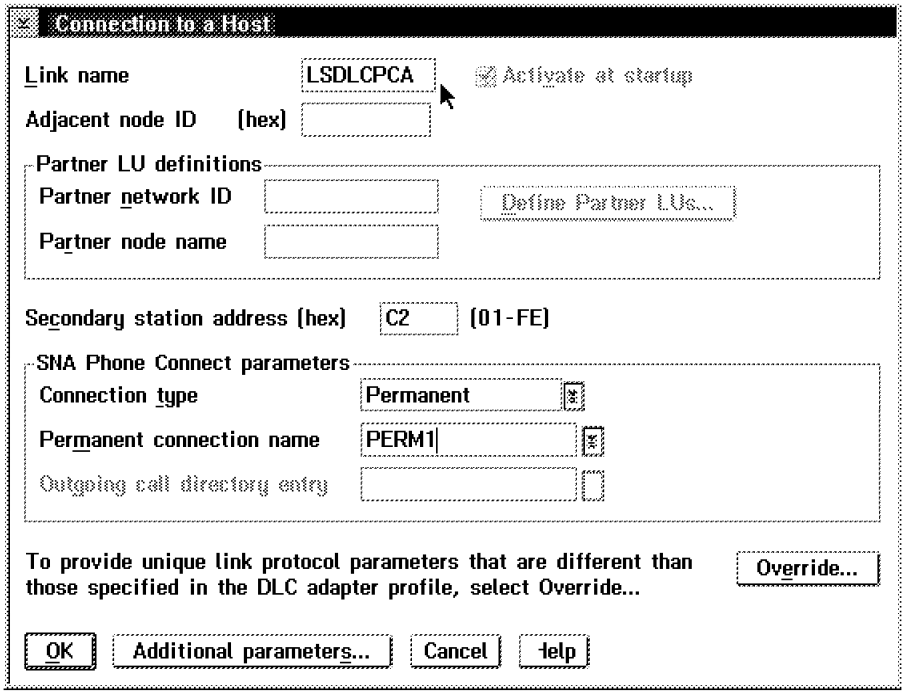


Figure 156. Connection to a Host

15. Enter the Local PU name for the PU you want to use for this particular downstream workstation and the Local node ID of the network node. Click on **OK** to finish these definitions.

    In this way you can define a maximum of 16 PUs with a maximum of 254 LUs each.

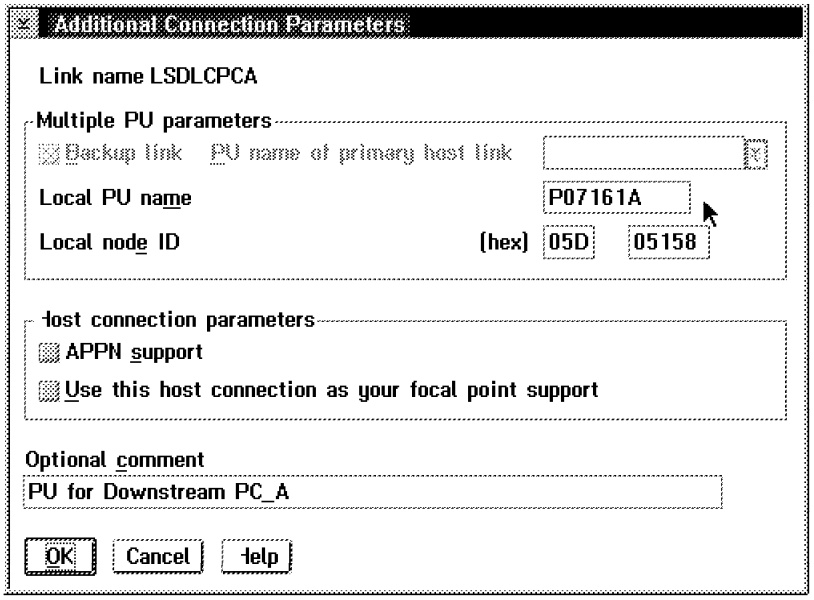


Figure 157. Additional Connection Parameters

8.4 Additional Pooling Capabilities

The LUs defined in the gateway can be dedicated to a particular workstation or pooled among multiple workstations. *Pooling* allows workstations to share common LUs, which increases the efficiency of the LUs and reduces the configuration and startup requirements at the host. You can also define multiple LU pools, each pool associated with a specific application. When a link is defined through the gateway between a workstation and the host, the LU is activated when the session is established and returned to the pool for access by other workstations when the session is ended.

8.4.1 Configuration of Pooled LUs

The pooling capabilities of IBM Communications Server Release 4.1 allows you to use LU pools in order to reduce the configuration requirements.

To configure pooled LUs, follow these steps:

1. In the Gateway Hosts and Host LU Pools window, select the gateway link to the host and click on **Pools**. The Gateway Host LU Pool window appears.

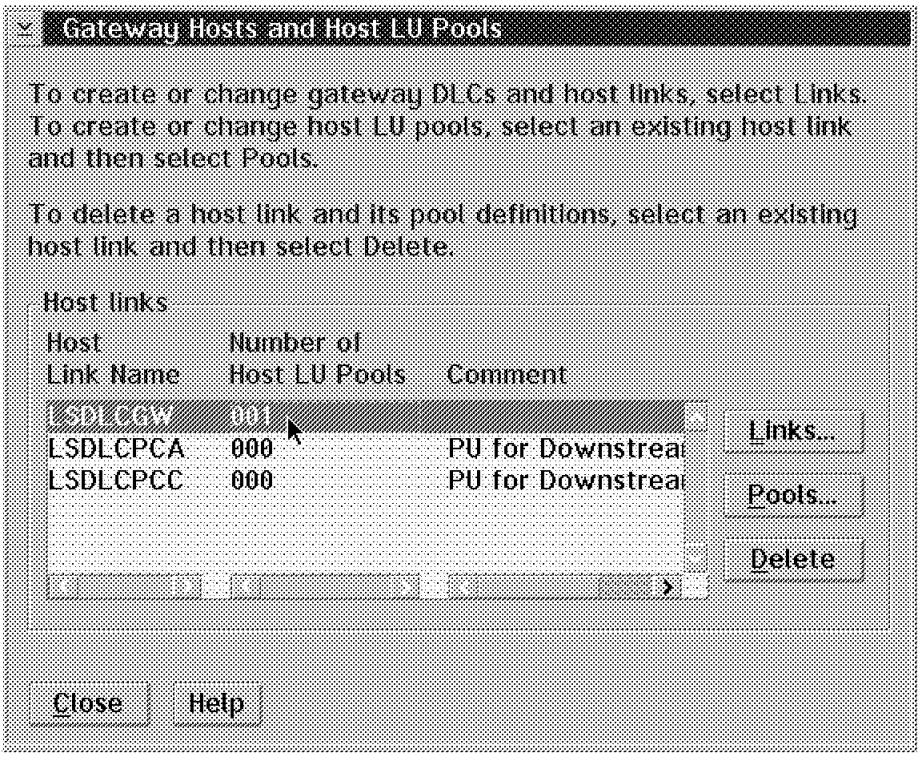


Figure 158. Links to the Gateway

2. In the Gateway Host LU Pools window, click on **Create** to change to the Host LU Pool Definition window.

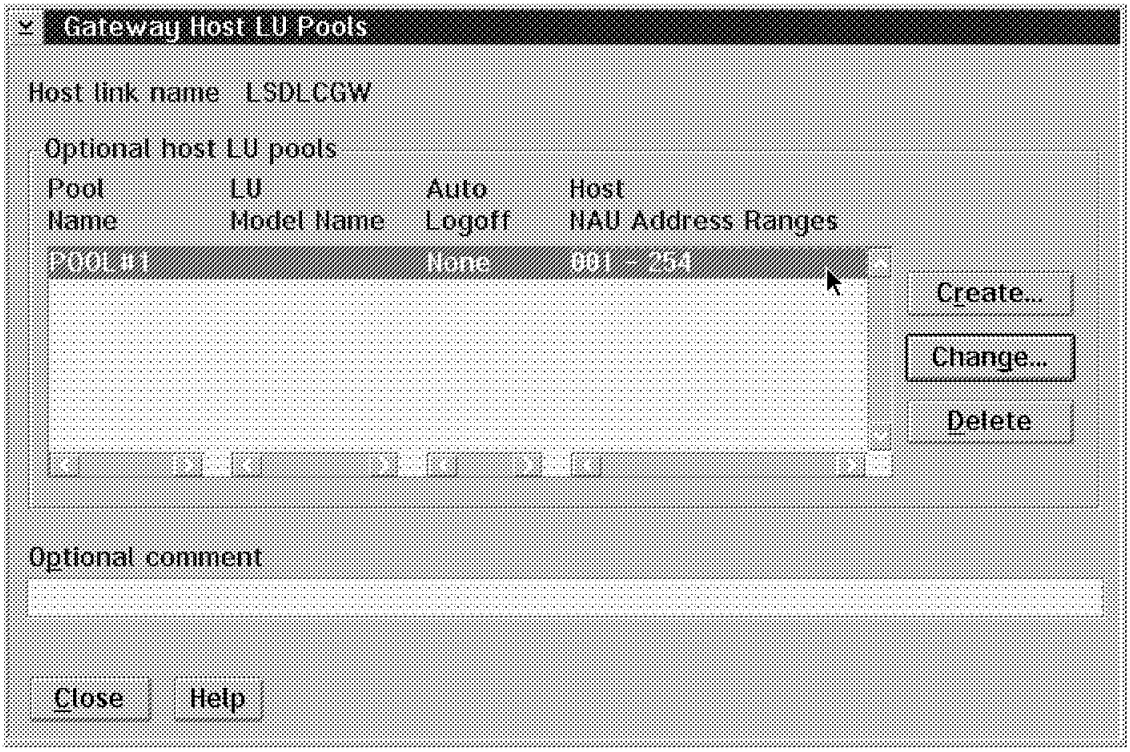


Figure 159. Gateway Host LU Pools

3. In this window enter the Pool name and click on **Create** for the Host NAU address ranges.

   Note the Automatic logoff section of this window. This enables you to choose a session to be logged off automatically if it was not active in a certain amount of time.

   Automatic logoff is helpful in an LU pool. For example, if eight LUs must service 15 LU sessions on 15 workstations, and if all eight LUs are active, the other seven workstation LUs cannot log on to the host.

   By having a workstation automatically log off after a user has not used an LU for a certain period of time, other workstation users have a chance to log on to the host. A workstation is logged off only when all sessions are in use and the gateway receives a session request from another workstation.

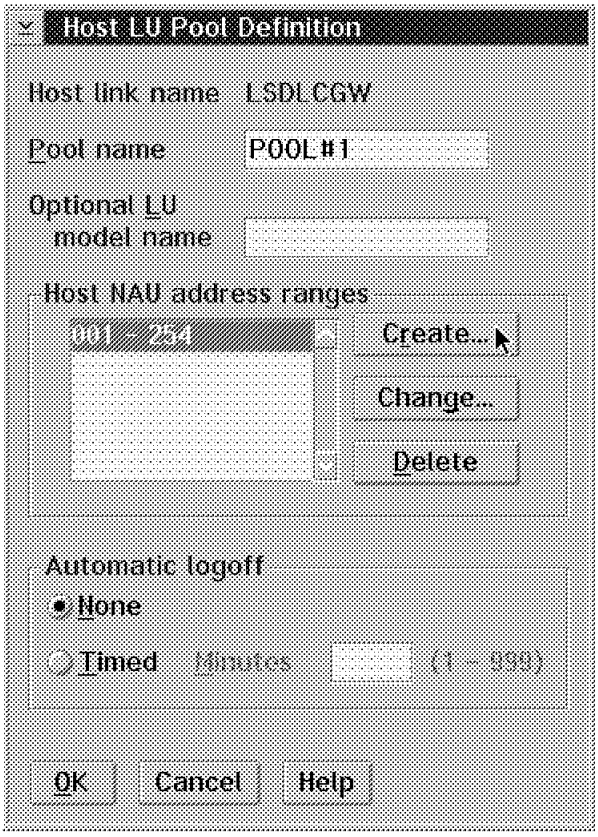


Figure 160. Gateway Host LU Pools

4. In the window for Host NAU Address Ranges, enter the Start NAU address and the End NAU address.

   Note that the address range is from 001 to 254, which means that you are enabled to have 254 LUs for each pool.

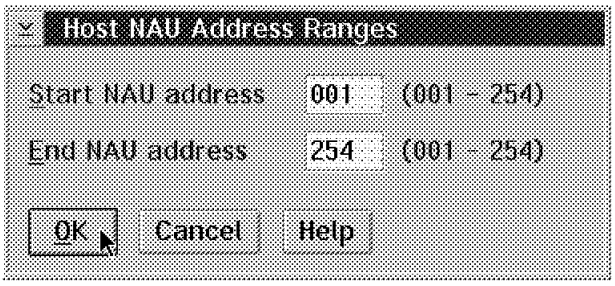


Figure 161. Host NAU Address Ranges

Click on **OK** to save your definitions.

8.5 Implicit Workstations Using the Gateway

Implicit workstations are much easier to configure, but they can use only pooled LUs. Instead of defining a link to each workstation using the gateway, you define a host pool (or pools) and configure the DLCs for the connections that the workstations are using. You then configure a model LU definition for each workstation network addressable unit (NAU) that connects to the gateway.

For example, if each workstation to an Ethernet LAN has two sessions that support 3270 emulation configured with NAU addresses 2 and 3, then you would configure two implicit LU definitions in the gateway: one for NAU 2, and another for NAU 3. In this example, each time a workstation connecting to the gateway over Ethernet is logged on, a link is dynamically created, and the two LUs for NAU 2 and 3 are allocated from the host pool.

For implicit workstations, users connecting to the gateway need to know only the adapter address of the gateway data link control (DLC) that is configured for the implicit workstations and what NAU values have been defined on the gateway.

8.5.1 Configuration for Implicit Workstations Using the Gateway

The Gateway - Implicit workstations using the gateway profile defines a generic workstation LU definition on a gateway for a workstation to have access to a host LU pool. Instead of explicitly creating a definition for every workstation using the gateway, you can use the implicit workstation definition.

You must define at least one gateway host LU pool before defining implicit gateway workstations that use the gateway definition.

To create an implicit workstation definition from a gateway, obtain the workstation NAU address for the implicit definition from your network administrator.

To configure for implicit workstations, perform the following steps:

1. Based on the configuration for Multiple PUs, click on **Implicit workstations using the Gateway**.

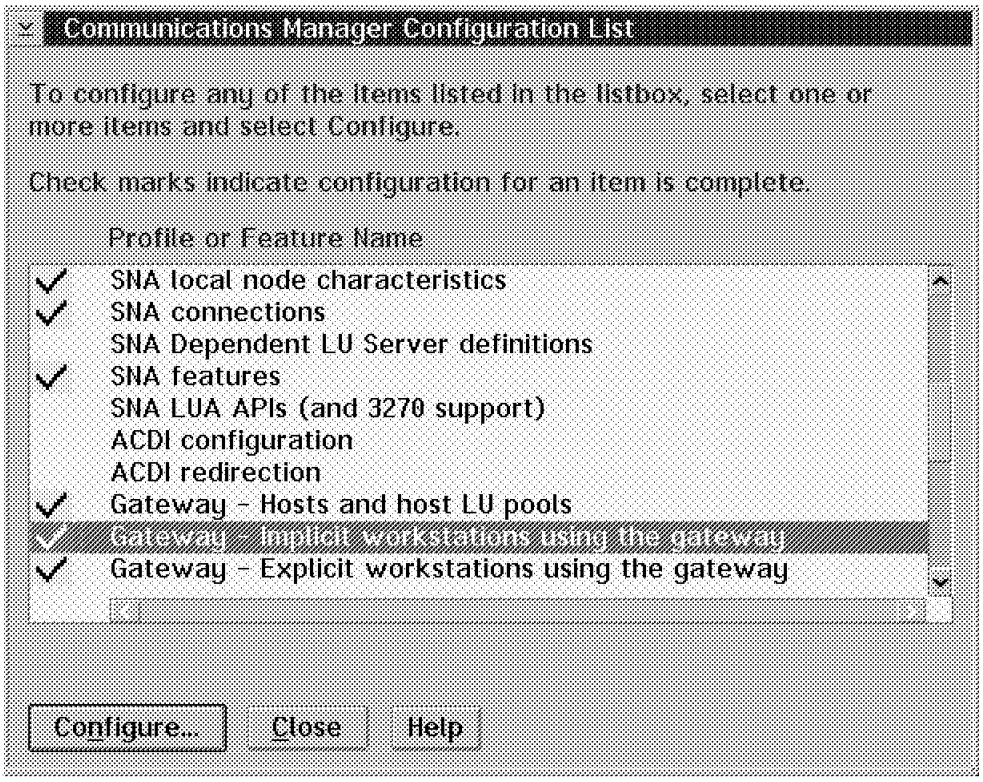


Figure 162. Implicit Workstations Using the Gateway

2. This window shows you the DLC types, Numbers of Adapters and LUs you already have configured and defined. To configure for implicit workstations, select **All configured DLCs** and click on **Implicit LUs**.

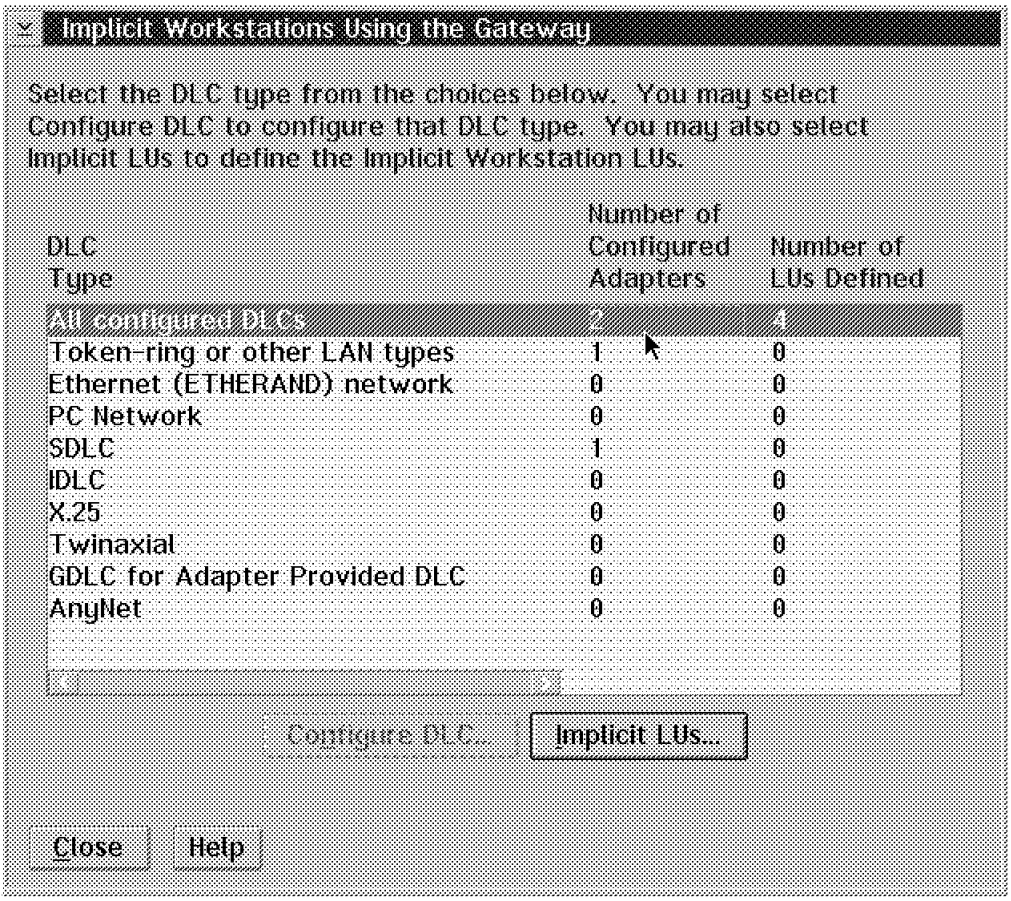


Figure 163. Implicit Workstations Using the Gateway

3. To create an LU definition for an implicit workstation, click on **Create**.

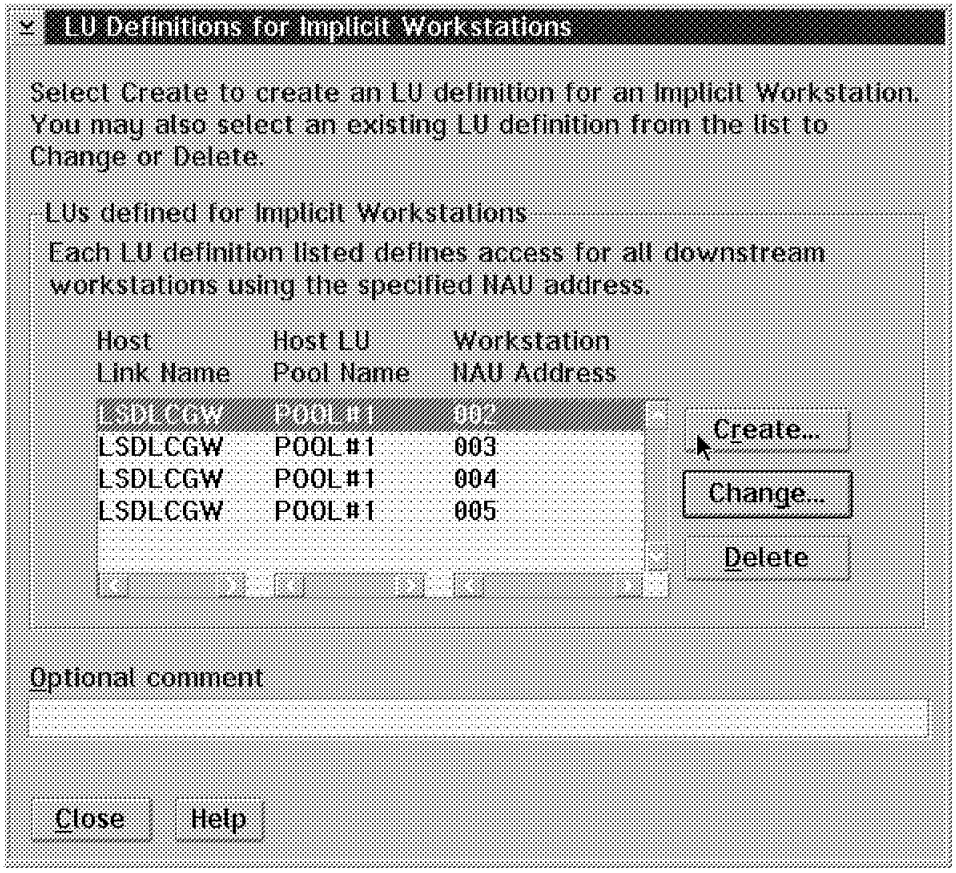


Figure 164. LU Definitions for Implicit Workstations

4. In the LU definition for implicit workstations, choose a host link name and a host pool name from the list boxes. Click on **Apply** to add another implicit workstation or click on **OK** to apply the new definition and to end the implicit workstation configuration.

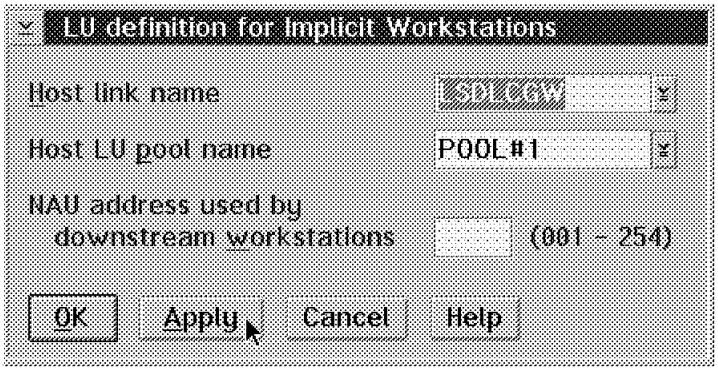


Figure 165. LU Definitions for Implicit Workstations

8.6 Explicit Workstations Using the Gateway

Explicit workstations have defined destination addresses over a particular DLC type (for example, token-ring network and SDLC). To configure explicit workstations, you must know the destination address of each workstation and also define a logical link to the gateway for each workstation.

LUs that are defined for explicit workstations can be pooled or dedicated. Pooled LUs are configured at the SNA gateway and grouped in pool classes. Pooled LUs are not dedicated to a specific workstation. Dedicated LUs are dedicated to a specific workstation, configured at that workstation, and known to the host by an LU local address at the host.

8.6.1 Configuration for Explicit Workstations Using the Gateway

The Gateway - Explicit workstation LU using the gateway profile defines a workstation LU definition at the gateway for a workstation to have access to one or more hosts. This LU can be pooled or dedicated.

To create or change an explicit workstation definition from a gateway, obtain the following information from your network administrator:

- The connection information required for the DLC used to connect to the workstation
- The workstation NAU addresses
- The host NAU address, if the LU is dedicated

1. In order to create explicit workstations, select **Gateway-Explicit workstations using the gateway** in the Communications Manager Configuration List panel and click on **Configure**.

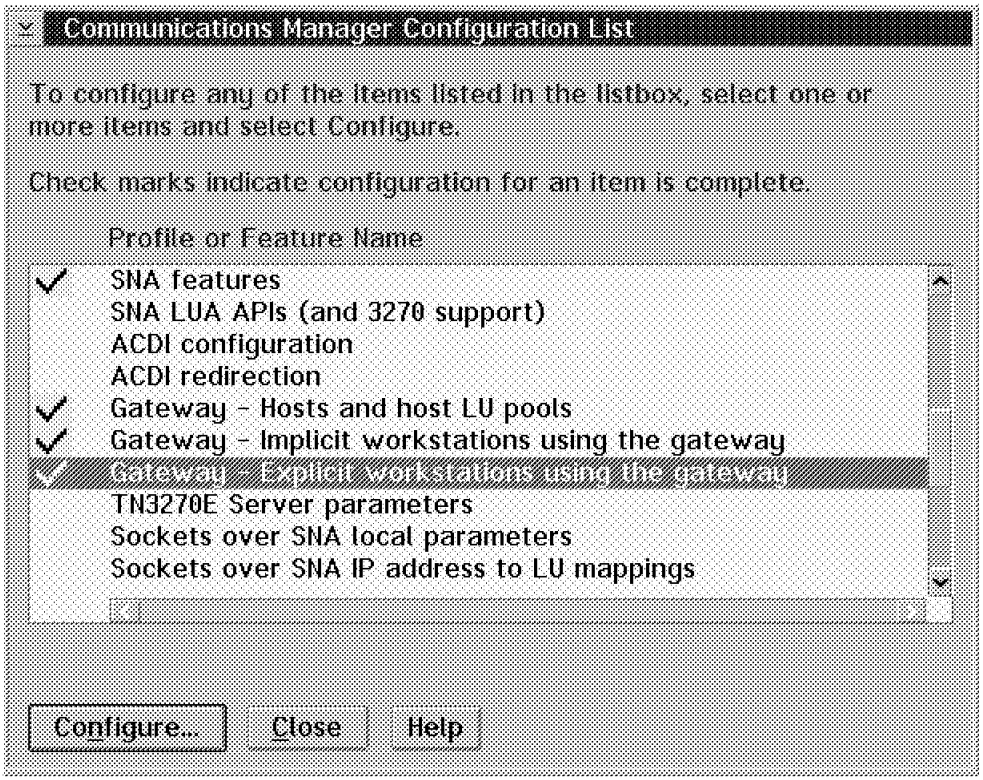


Figure 166. Communications Manager Configuration List

2. The Explicit Workstations Using the Gateway window appears. It shows the workstation links you already configured. Select one of the links and click on **LUs...** to configure for explicit workstation LU.

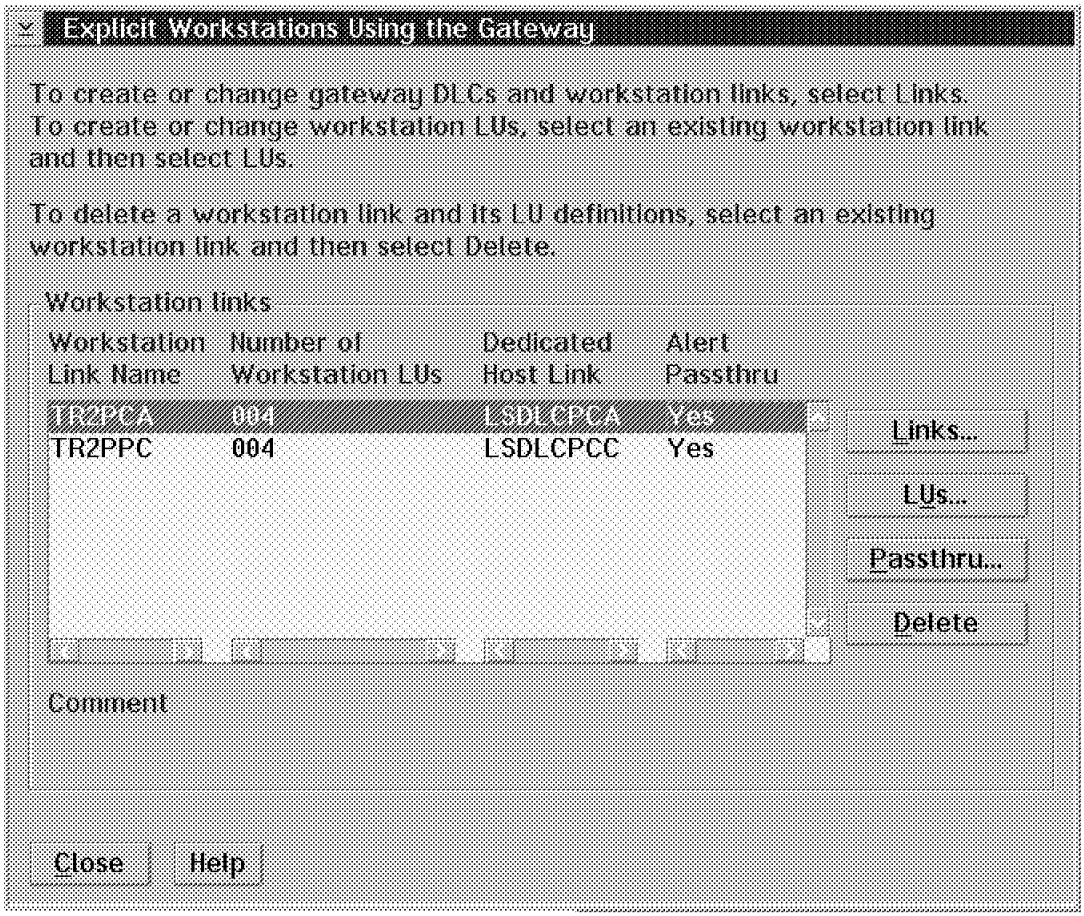


Figure 167. Explicit Workstations Using the Gateway

3. To create an explicit workstation LU click on the **Create** push button of the Workstation Link Definition panel.

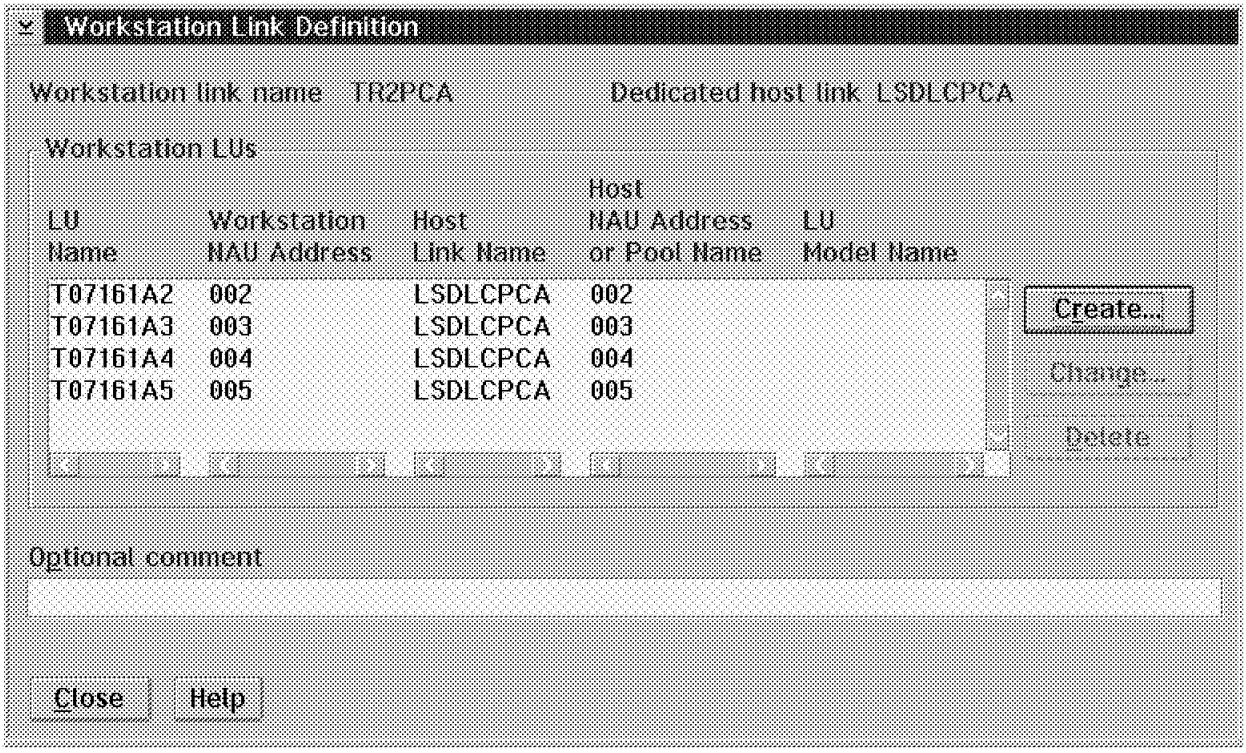


Figure 168. Workstation Link Definition

4. The Workstation LU Definition window appears. Enter the LU name and choose a Host link name from the list box. Enter the Workstation NAU address. In the LU type field of the window, select **Dedicated** and enter the Host NAU address. Click on **Apply** to save the new definition and to start to configure another one, or click on **OK** to apply this definition and to finish the configuration.

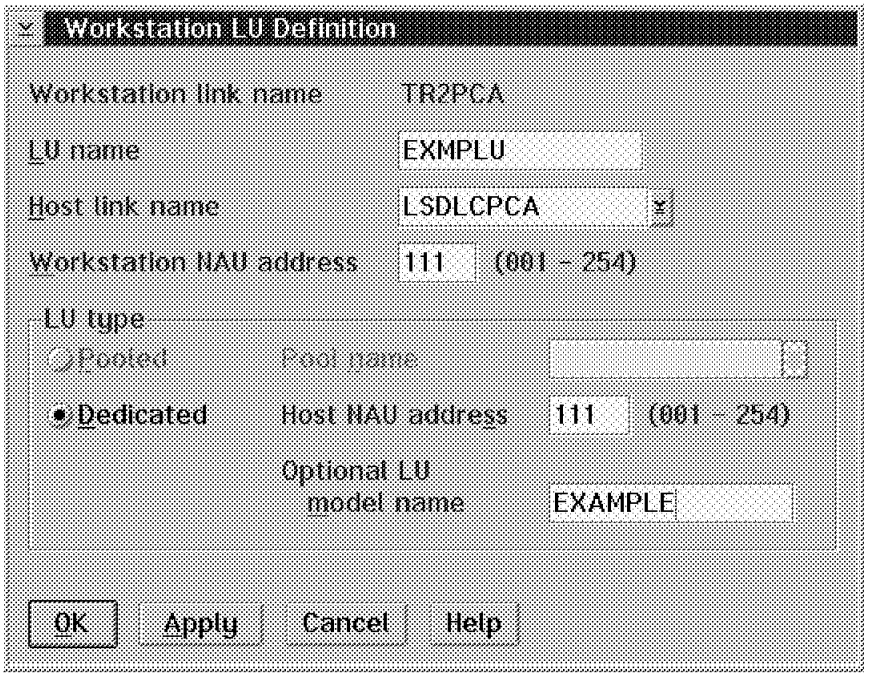


Figure 169. Workstation LU Definition

5. After clicking on **Close**, the Workstation Link Definition window appears again, showing the newly configured workstation LU definition. Click on **Close** until you reach the Communications Manager Configuration List and end your configuration.

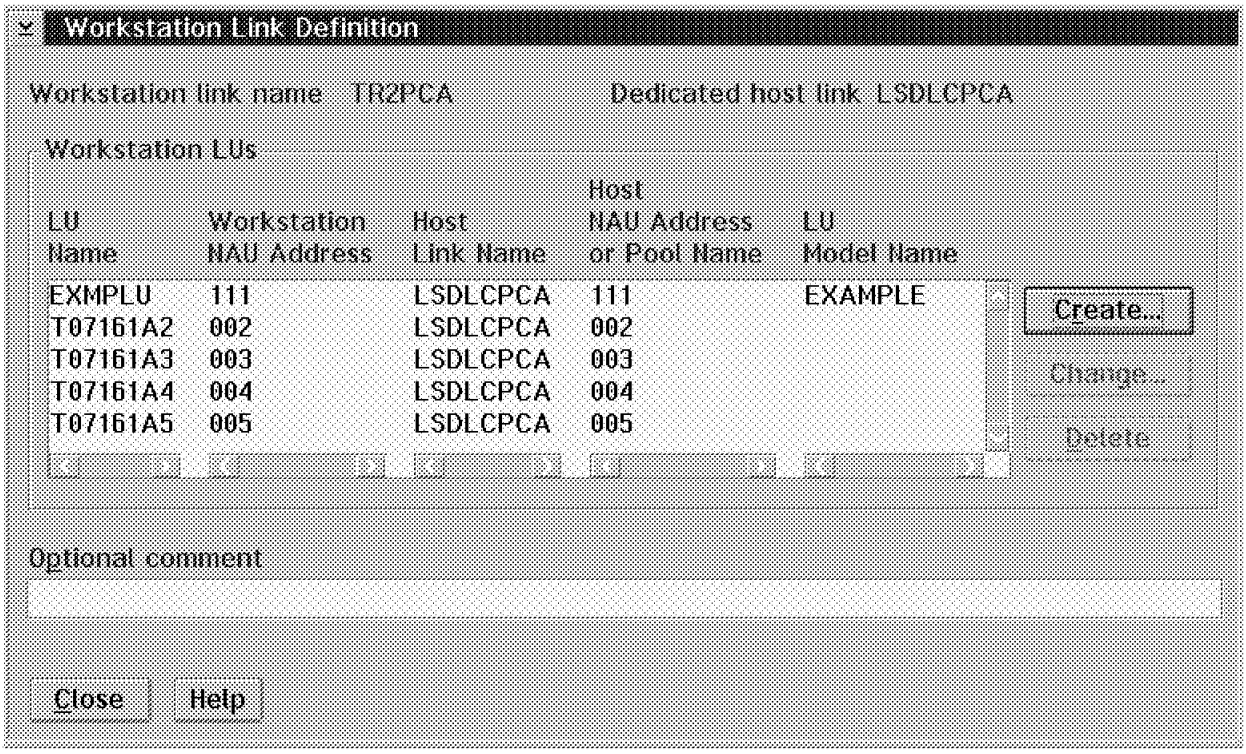


Figure 170. Workstation Link Definition

Chapter 9. The TN3270E Server

TN3270E is now a fully integrated function of IBM Communications Server Release 4.1.

This chapter introduces planning for TN3270E Server and discusses the following topics:

- Overview of TN3270E Server
- Supported Client Workstations under TN3270E Server
- Highlights
- Configuring SNA Connections
- Server LUs
- Pooling

9.1 Overview of TN3270E Server

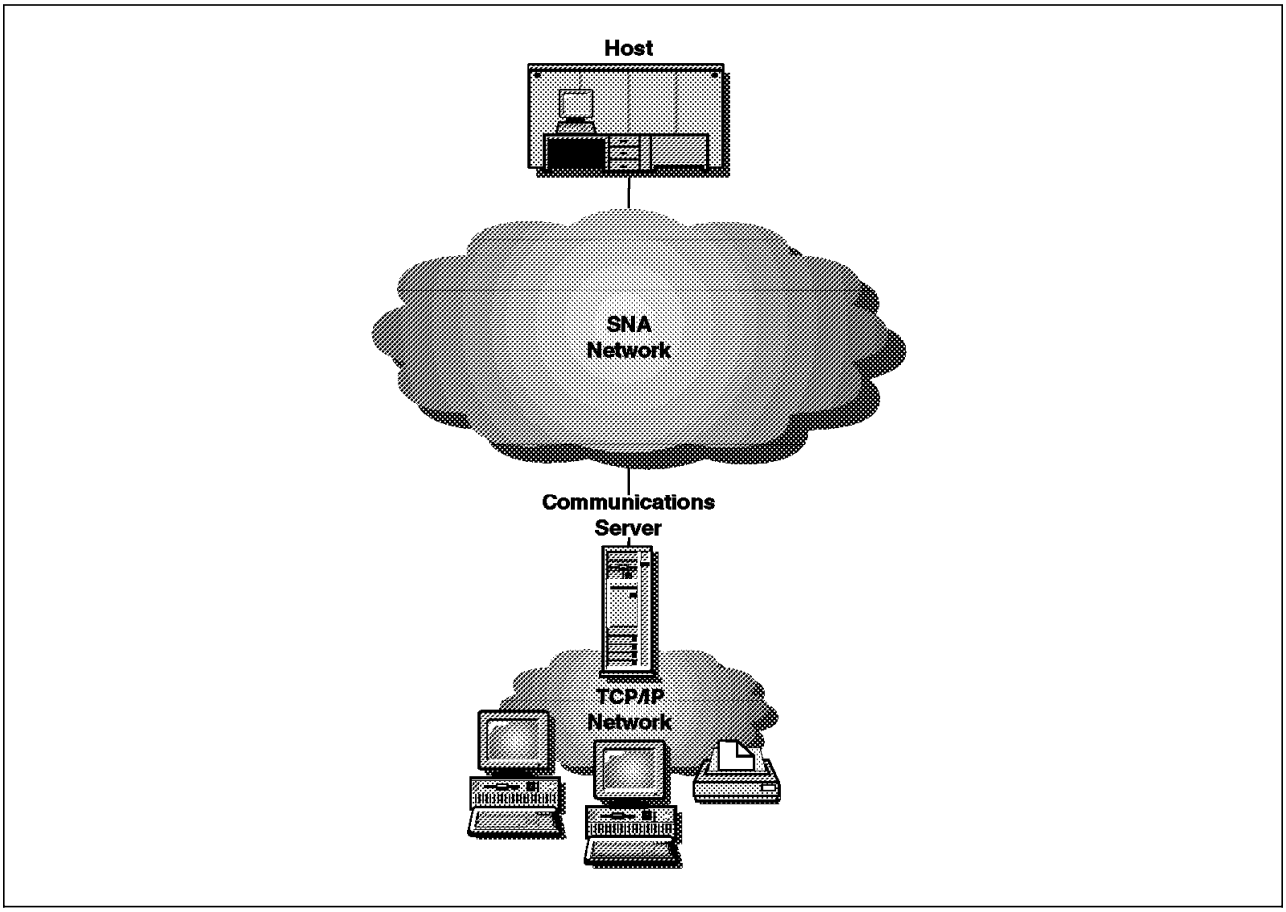


Figure 171. TN3270E Server with Wide Area SNA Network

The TN3270E Server function provides gateway capabilities that allow downstream workstations with TN320 emulation clients to connect into the

Communications Server over an IP network. The gateway converts Telnet 3270 into SNA 3270 data stream. See Figure 173 on page 198.

The TN3270E Server supports *Request For Comment* (RFC) 1576, 1646 and 1647. (To order the appropriate RFCs, please refer to I.4, "Requests for Comments (RFCs)" on page 336.)

- RFC 1576
  - TN3270 current practices
    - *Base* TN3270
    - Allows terminal to access only 3270 applications
    - No SSCP-LU support
    - No printers
    - ATTN and SYSREQ implementations vary
    - No way to request a specific LU
    - No notification of errors
    - Negotiation only consists of Terminal Type, Binary, and EOR (End Of Record)
- RFC 1646
  - TN3270 Extensions for LU names and printers
    - Allows client to request a specific LU
    - Allows printers (LU1 and LU3)
    - No notification of errors for terminals
    - No SSCP-LU support
    - Error notification only for printer sessions
    - Negotiation only consists of Terminal Type, Binary, and EOR
- RFC1647
  - TN3270E
    - Formalizes TN3270E as a standard for 3270 communications in a TCP/IP environment
    - SSCP-LU support
    - ATTN and SYSREQ support
    - Response support for both terminals and printers
    - LU classes
      - Implicit workstation
      - Explicit workstation
      - Implicit printer
      - Explicit printer
    - Associated printer
      - A printer LU that can only be accessed by referencing its associated terminal.

- Negotiation consists of Bind Image, Responses, and SYSREQ; for printer sessions, SCS-CTL-CODES and DATA-STREAM-CTL

TN3270E provides standard Telnet 3270 functions and also support for:

- LU1 and LU3 Printers
- The ATTN (Attention Key) which interrupts the application program
- The SYSREQ (system request) key which queries the host

TN3270E is also available as an APAR to CS/2 V4.0 (not recommended). TN3270 is part of CS/2 product package. CS/2 charges a one-time fee and does not have a usage charge. In contrast, the AIX Client Access license structure charges according to the number of concurrent users. Functionally the products offer the same level of functions.

The TN3270E Server removes the requirement for TCP/IP on the host or to have *Sockets over SNA* for each client.

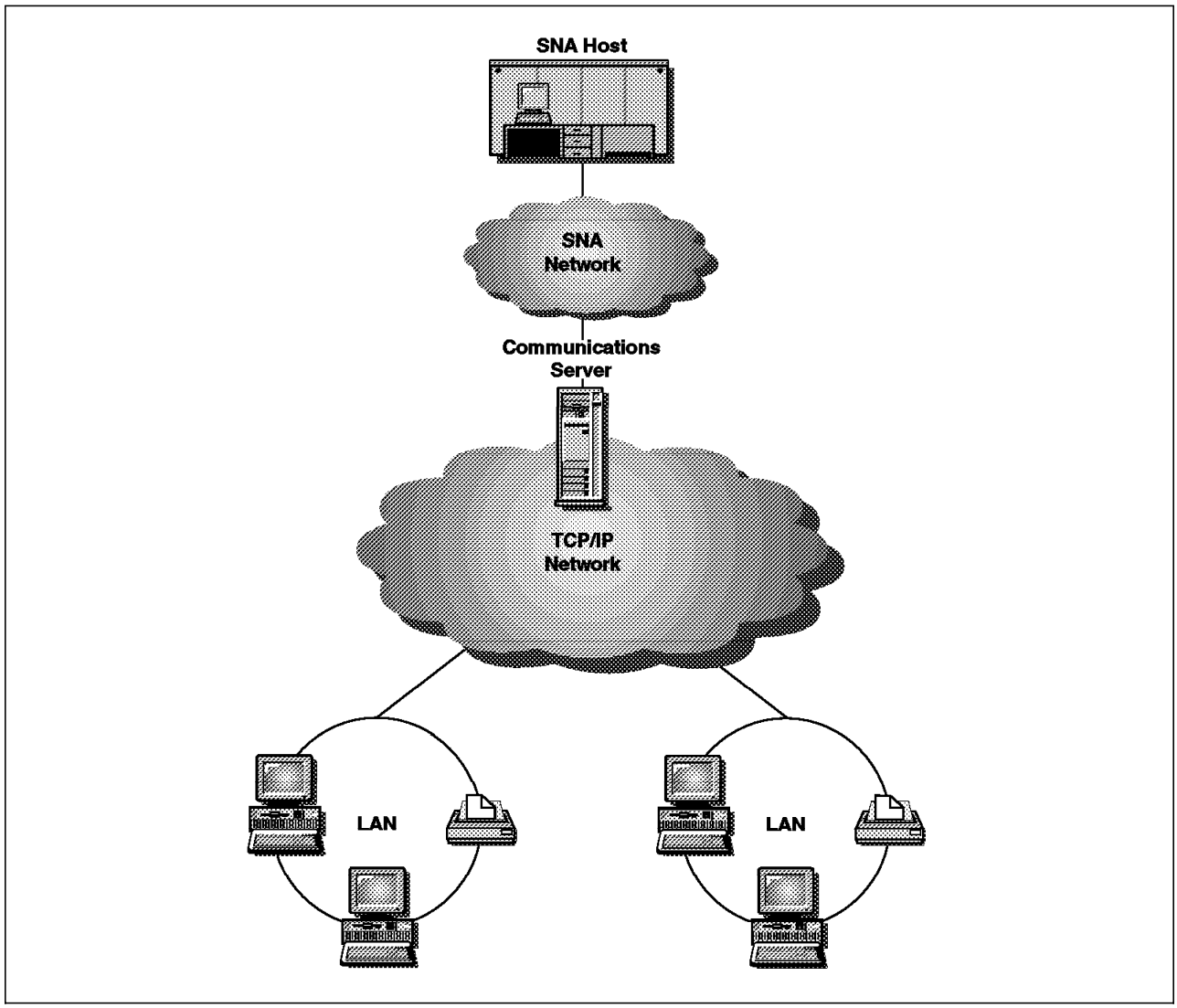


Figure 172. TN3270E Server with Wide Area TCP/IP Network

9.1.1 Functions Supported by TN3270E Server

The TN3270E Server function supports:

- **SSCP-LU**

  Although base TN3270 does not support SSCP-LU sessions, TN3270E Server has implemented it. If you are connected to a host session, you can see *SSCP* displayed in the lower right corner of the screen. This indicates that an SSCP-LU session is active.

  TN3270E handles all requests from the client and converts them to SSCP-LU data.

  TN3270E generates a 3270 data stream necessary to display SSCP data. This is only necessary for base and RFC 1646 clients.

- **LU emulation**

  TN3270E Server supports LU 2, which enables users to run interactive 3270 display application programs.

- **Host print**

  The Telnet 3270 standard extensions (TN3270E) enable users to print from host applications to printers attached to their workstation or in their TCP/IP network using LU 1 and LU 3 print sessions.

  TN3270E Server implements the protocols outlined in RFC 1646 and RFC 1647, enabling the server to pass LU 1 and LU 3 session data to TN3270E-enabled clients, wait for client confirmation of the print request, and respond to the host.

- **Response handling**

  TN3270E-enabled clients can send both positive and negative responses, which TN3270E Server sends on to the host. TN3270E Server generates request responses for standard TN3270 clients.

- **ATTN and SYSREQ key handling**

  TN3270E Server can convert and forward information to the host when the client sends an ATTN or SYSREQ key. Although clients that are not TN3270E-enabled have no explicit definition for ATTN and SYSREQ, the TN3270E Server uses the following Telnet commands to implement these functions:

Table 19. Command Equivalents

Telnet	TN3270E	Standard TN3270
IP	ATTN	SYSREQ
AO	SYSREQ	SYSREQ
BREAK	ATTN	ATTN

- **LU classes**

  Communications Server categorizes user connection with LU classes. Classes consist of LUs configured with common characteristics (for example, those that require a specific host connection). This simplifies user access, groups users by application needs, and maximizes host resources.

TN3270E Server supports both standard and extended Telnet 3270. Typical client programs emulate a 3270 display. Clients that support the TN3270E protocol can emulate LU 1 and LU 3 printers.

Refer to *Up and Running* for instructions on how to configure a TN3270E Server using CMSETUP.

9.2 Supported Client Workstations under TN3270E Server

TN3270E Server supports any TN3270E or TN3270 down stream client that is fully compliant with RFC 1576, 1646 or 1647. The following clients are examples:

- Personal Communications/3270 for OS/2
- Personal Communications/3270 for OS/2 Version 3.x and 4.x for DOS and Windows
- IBM TN3270 (OS/2, DOS, and AIX)
- XANT
- Brixton 3270 Open Client (AIX, OS/2)
- PMANT
- Wall Data RUMBA Office (Windows)
- Attachmate Extra! Personal Client (Windows)

9.3 Highlights

This section provides more information about some of the features you can configure for TN3270E Server.

9.3.1 Changing the Default Port Number

You can configure the port number your server will use for new connections from the TN3270E Server Parameters window or from a response file. The default port number is 23, but the Telnet application also uses this port. If TELNETD is running, and its port number has not been changed, then you must use another port.

If you change the port number, avoid numbers that you know are used by other applications. If two applications use the same port number, one of the applications will fail.

You should notify TN3270E client users when you change the port number, because they will have to configure their emulator applications to match.

9.3.2 Managing System Traffic

There are two ways to control how often unused connections are disconnected: *keepalive processing* and *automatic logoff*.

If you use keepalive processing, you can choose either *NOP* or *timing mark*.

- **NOP processing**

  This sends a Telnet NOP command after a specified keepalive frequency. This causes data to be transmitted on the connection, which causes TCP/IP

to detect that the connection has broken. The server does not expect a response from the client. It can take an unpredictable amount of time for TCP/IP to detect the connection's status.

- **Timing mark processing**

  This sends a Telnet timing mark command to the client. If the client does not respond within the specified period, the connection is closed.

Timing mark processing causes more traffic on the system than NOP processing, but frees unused connections more quickly.

If you choose automatic logoff, the server disconnects any session that has no traffic for the specified period. Traffic from keepalive processing does not keep the connection open; data must be sent to or from the host. Printer sessions are not automatically logged off.

If your client emulators are configured to do keepalive processing, you might want to turn it off at the server, and if keepalive processing is done at the server, you might want to turn it off at the client.

9.3.3 Configuring SNA Connections

You need to configure your host connection and host LUs before you use the TN3270E Server function. For a table that describes and explains host parameters, refer to *Up and Running*.

9.3.4 Server LUs

MPTS can support 2048 sockets for TCP/IP connections. With no connections, TN3270E Server uses five sockets. Each client LU connection uses an additional socket. The total number of client connections supported is limited by the number of sockets available. Other applications using TCP/IP can use up available sockets, thus reducing the number of client connections possible.

9.3.5 Pooling

There are four classes of LU definitions specific to TN3270E Server:

- Implicit workstation
- Explicit workstation
- Implicit printer
- Explicit printer

These classes correspond to the terminal-generic, terminal-specific, printer-generic, and printer-specific classes specified in RFC 1647.

Implicit workstation definitions do not require a specific LU name. You can define a set of these definitions that the TN3270E Server uses to satisfy requests for connections.

You can also define a set of LU definitions used to satisfy requests for a specific LU name. These *explicit workstation* definitions ensure that a terminal device needed by a host application is not assigned to a client that does not specially request it.

Similarly, you can define a set of printer definitions that will be used to satisfy requests for connections that do not require specific LU names *(implicit printer)* and a set used to satisfy requests for a specific LU name *(explicit printer)*.

Implicit and explicit workstation definitions can have printers that are associated with them; that is, each terminal definition can have a printer that is assigned to it, and each printer can have an associated terminal definition. These printers are not included in the explicit or implicit printer definitions.

Associated printer definitions can only be accessed by referencing the terminal LU name. They reduce the amount of information the client user needs, because he or she only needs to know the LU name of the terminal to connect to both the terminal and the printer sessions.

9.4 System Resources

The following is a list of system resources needed by the TN3270E Server:

- Add eight threads to REMMAIN.EXE.
- Limit of approximately 2000 connections. This is because of TCP/IP limitation.
- Use five sockets with no client.
- One additional socket per client connection.
- Use 12 KB of memory per connection initially (may allocate an additional 36 KB if necessary).

9.5 Interfaces

The following interfaces are used by TN3270E Server:

- Created a new DLL (tn3270d.dll) during installation into CMLIB.
- Messages are logged to FFST/2.
- Require MPTS for TCP/IP stack.
- Require LUA for SNA stack.

9.6 TN3270E Server Parameter Profiles

This section describes the TN3270E Server Parameter profile you can configure. The following sections contain more information:

- Introduction
- Configuring the TN3270E Server Function
- Profiles
- Configuration Hints and Tips

The TN3270E Server function enables TCP/IP users to communicate with a host machine.

TN3270E Server does not require any changes to the host application programs. The server accepts SNA traffic from the host and converts it into Telnet format for the client workstation. It also accepts Telnet traffic from the client and

converts it into SNA format for the host. This reduces processor cycles on the host and enables you to place the processing (the TN3270E Server) anywhere you want in the network. See Figure 173 on page 198.

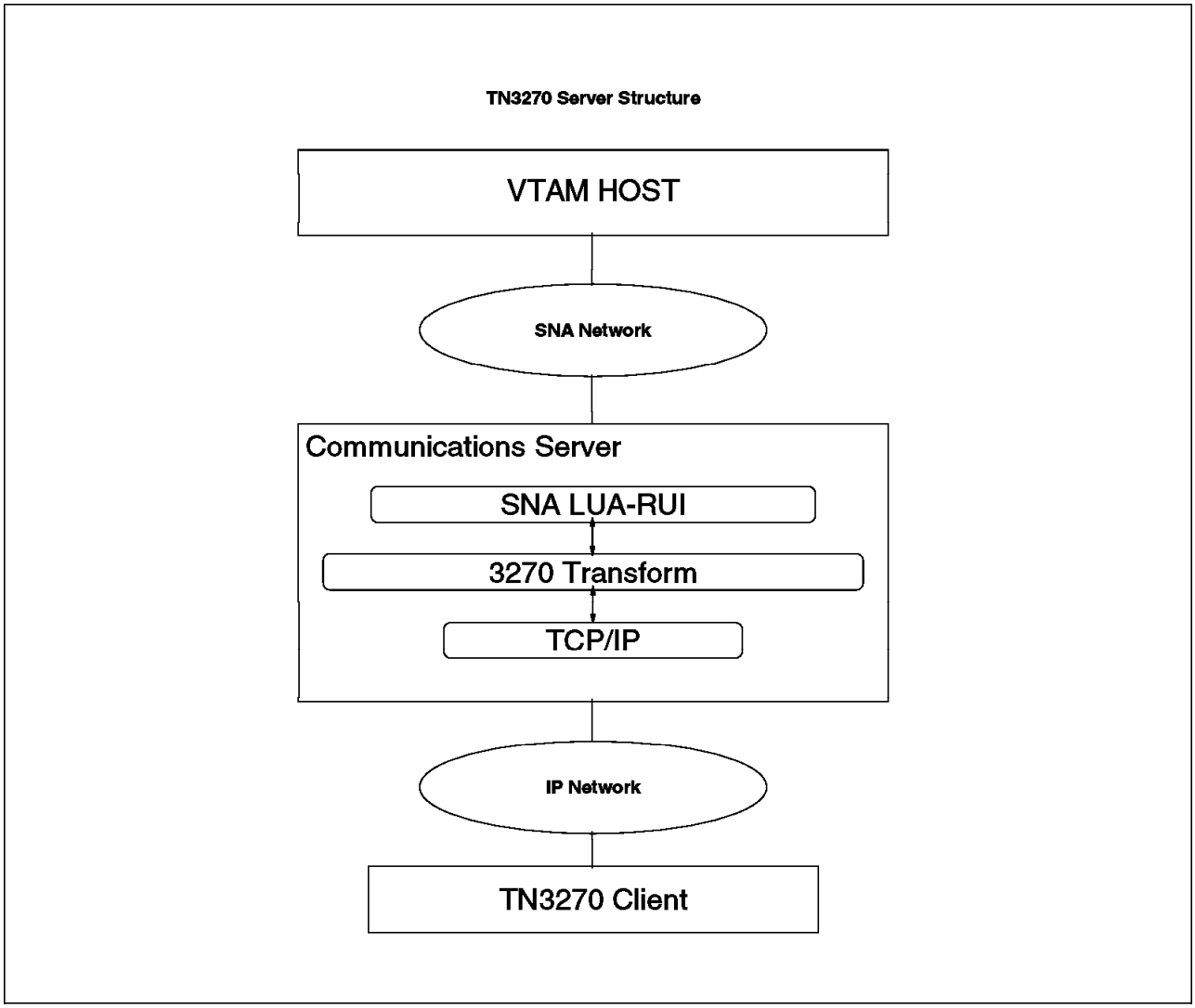


Figure 173. TN3270E Server Schematics

The TN3270E Server can connect to any industry-standard TN3270 or TN3270E client workstation and perform as the server for access to SNA networks. It supports host printing to workstation-attached or IP network printers. It also supports ATTN and SYSREQ key handling.

Note

Only workstations running Communications Server can be configured as TN3270E Servers.

For additional information about TN3270E Server, refer to the online helps and to *Network Administration and Subsystem Management Guide*.

9.7 Configuration Overview

You access the TN3270E Server Parameter profile by selecting **Configure** from the TN3270E pull-down menu in the Communications Manager Configuration Definition window.

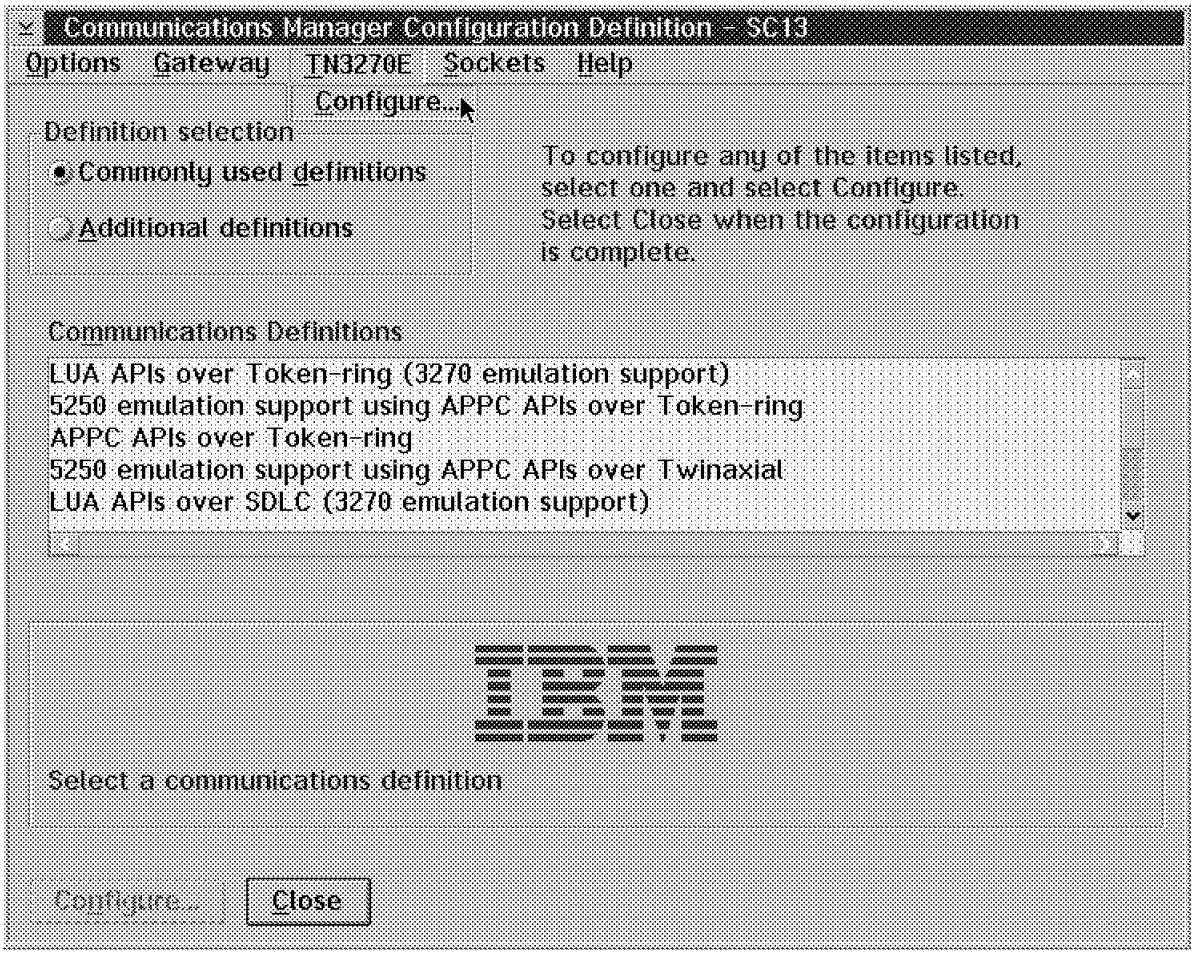


Figure 174. Access TN3270E Server Parameter Profile

9.7.1 Profiles

The profiles for the TN3270E Server are:

- TN3270E Server Parameters
- TN3270E Server Additional Class Definitions
- TN3270 server Optional Parameters

The profile for the server parameter is identified as Required under the Action column in the Communications Manager Profile List window. You must complete the Required profile. The other profiles are available from the Server Parameter panel.

While configuring a profile, use the *online help* for a description of the parameters.

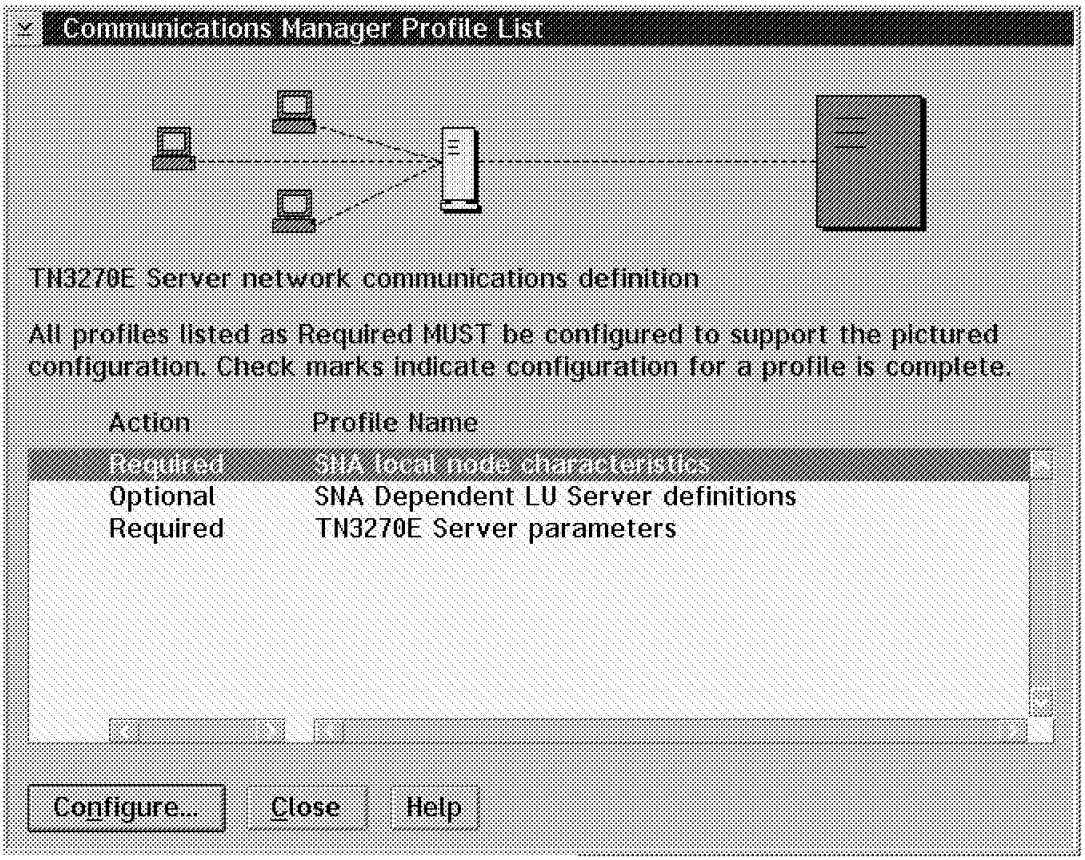


Figure 175. TN3270E Communications Manger Profile List

9.7.2 TN3270E Server Parameters

In the TN3270E Server Parameter profile, you define the following kinds of information:

- Host link name
- Number of implicit workstation definitions

This is a new panel for TN3270E. It is a server configuration fastpath. That is, if you specify a host link name and the number of implicit workstation definitions to create, Communications Server creates the definitions for you. The *implicit* workstation definitions are generated automatically.

If default option values are acceptable and no additional LU classes are required, the TN3270E Server configuration is completed upon exiting this panel.

You may dynamically add LUs, but LUs will not be deleted by the TN3270E Server until it is restarted.

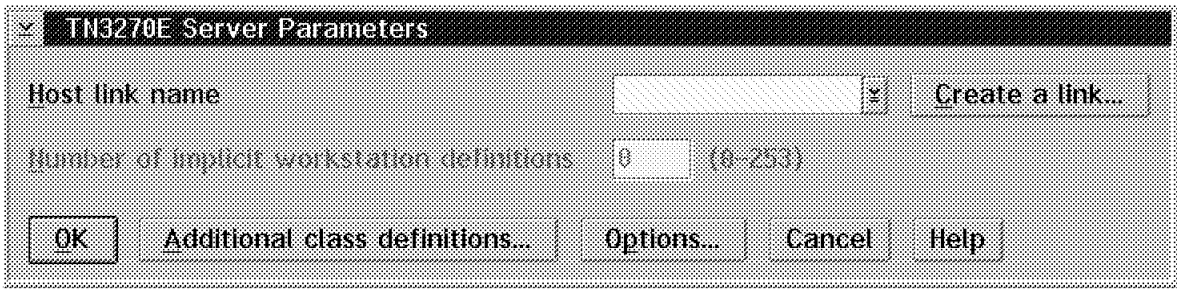


Figure 176. TN3270E Server Parameters

9.7.3 TN3270E Server Additional Class Definitions

In the TN3270E Server Additional Class Definitions profile, you define the LUs of four class types and associate the printer LU. You can also select an existing TN3270E definition and edit it or indicate whether a printer should be associated with it.

Even though this panel allows you to delete LUs, the TN3270E Server will not remove it until the server is restarted.

Note

The function to select an implicit LU requested by name is not fully implemented yet. It will be available in the final version of CS/2 4.1, TN3270E Server.

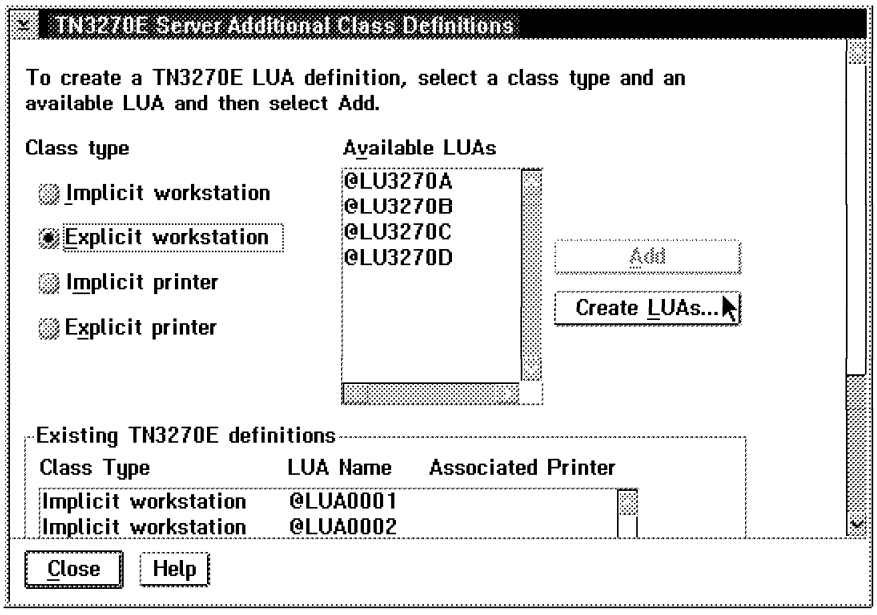


Figure 177. TN3270E Additional Class Definitions

9.7.4 Printer Association

This new panel is displayed by choosing the **Associate printer** push button on the TN3270E Server Additional Class Definition panel.

Associating a printer LU to a workstation LU allows you to access both the printer and the workstation by only knowing the workstation LU name.

A limited number of clients support printer association.

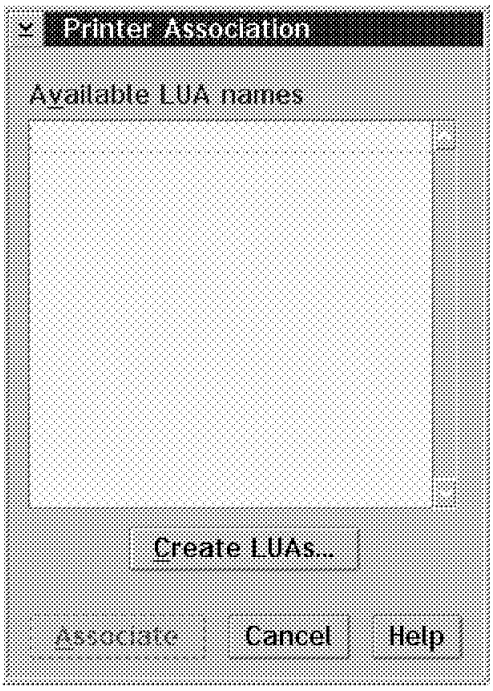


Figure 178. Printer Association

9.7.5 TN3270E Server Optional Parameters

In this new panel, TN3270E Server Optional Parameter, you define the following types of information:

- Port number
- Keepalive processing (use, method, frequency, timer)
- Automatic logoff

The defaults for the parameters are:

- Port = 23
- No automatic logoff
- No keepalive processing

All options on this panel can be updated dynamically.

Note

Port 23 is also used by the TELNETD application. If both TN3270E Server and TELNETD run on the same machine, the port number must be changed here. All clients must connect to the new port number.

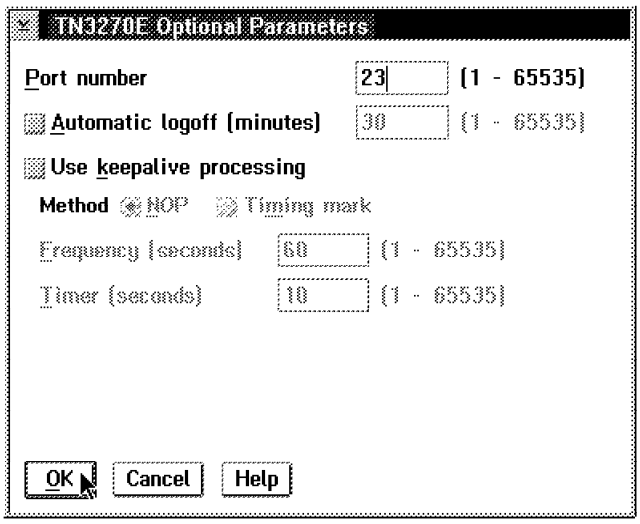


Figure 179. TN3270E Optional Parameters

9.8 Configuration Hints and Tips

The following are some points to remember when configuring the TN3270E Server function.

9.8.1 MPTS

If you are going to use TN3270E Server support, you should:

1. Install the version of MPTS shipped with the product or a later version.
2. Configure MPTS to provide TCP/IP protocols. Make sure that you check the sockets push button.

9.8.2 Creating Definitions

The number of TN3270E Server definitions you can create is limited to the number of sessions the host provides.

9.8.3 Explicit Workstation Definitions

You should consider using explicit workstation definitions when you want to restrict your user's access to specific LUs defined at the host.

9.9 Configuring the TN3270E Server Function

This chapter shows a TN3270E configuration in detail, capturing all of the parameters you need to define for the following scenario. (See Figure 180 on page 204.)

9.10 Scenario

In this section, we show you a sample configuration for the following scenario:

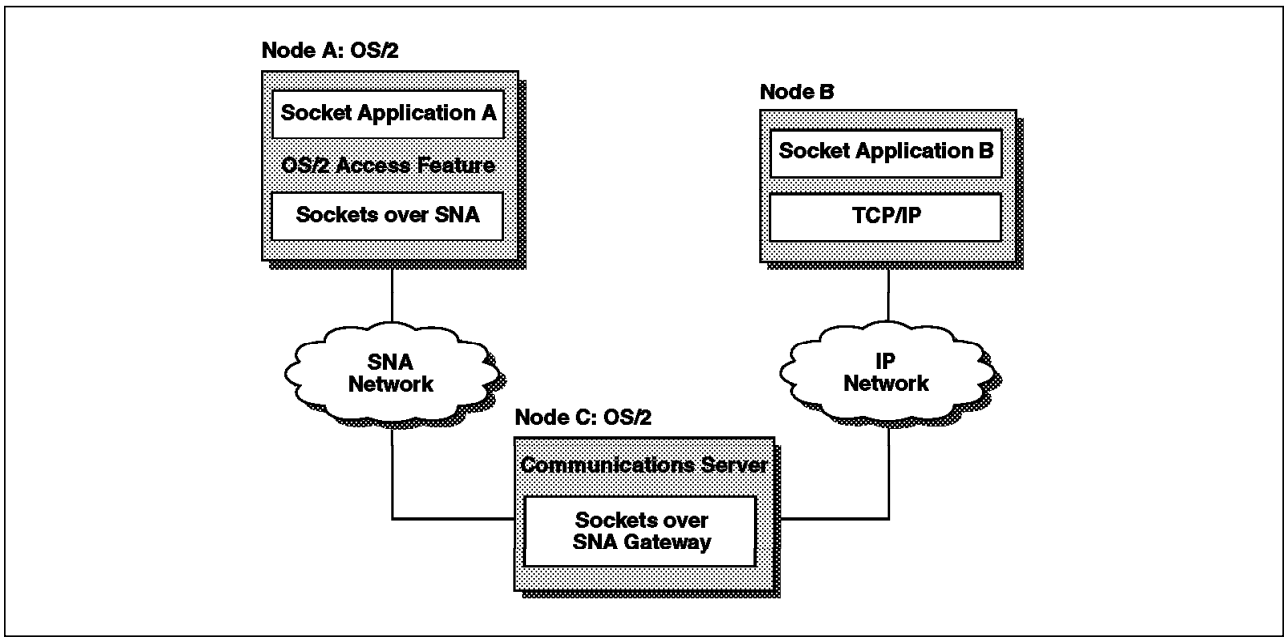


Figure 180. Scenario

9.10.1 Configuration

To start the TN3270E configuration, call CMSETUP from an OS/2 command line or click on the Communications Manager Setup icon in the Communications Server folder.

1. After clicking on **Setup** in the Communications Manager Setup window and entering the name of your configuration in the Open Configuration window, the Communications Manager Configuration Definition window appears.

   Access the TN3270E Server Parameter profile by selecting **Configure** from the TN3270E pull-down menu.

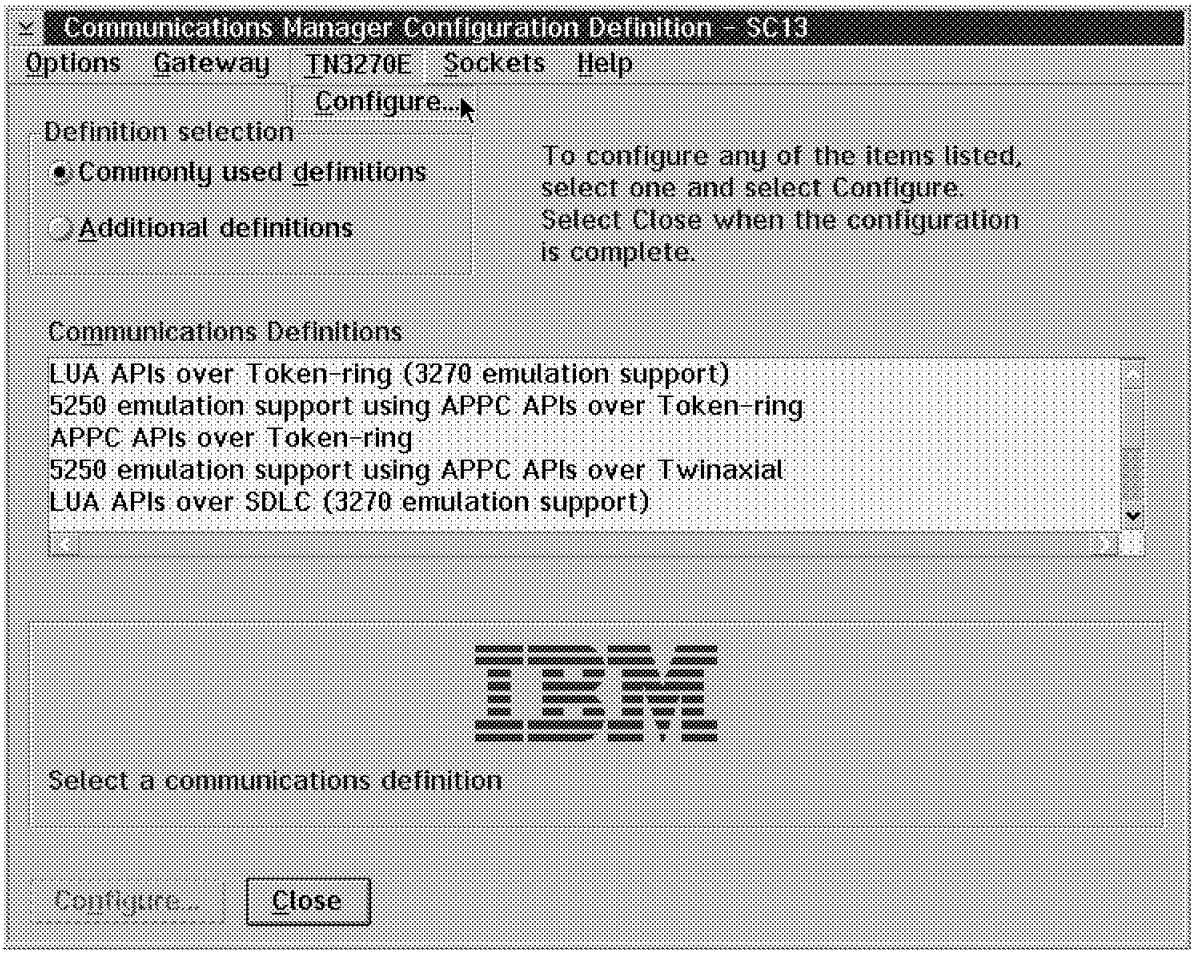


Figure 181. Access TN3270E Server Parameter Profile

2. The Communications Manger Profile List appears.

   Complete the Required profile.

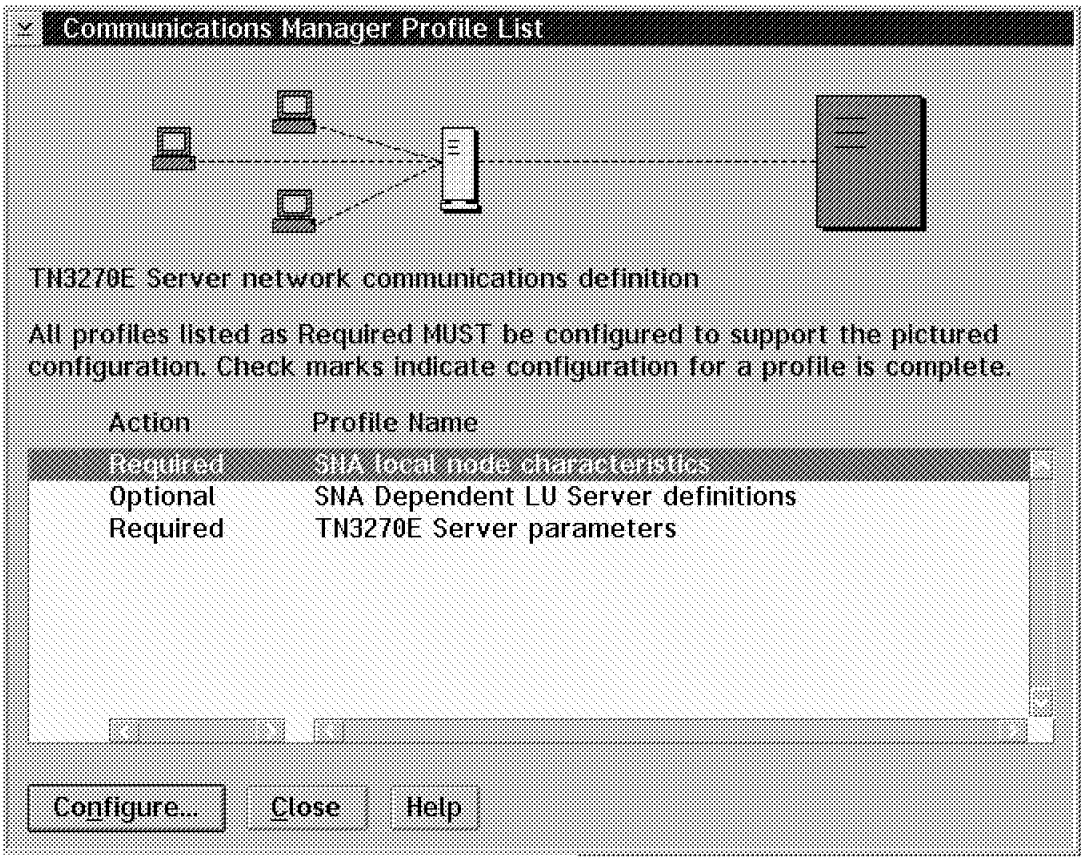


Figure 182. TN3270E Communications Manger Profile List

3. Start with the SNA local node characteristics.

   Enter the Network ID and the Local node name. In the Node type section of Figure 183 on page 207, select whether your node is an End node or a Network node. Enter the Local node ID and the local node alias name (optional).

   If you need data compression for your session, enter the necessary definitions for the data compression fields.

   Click on **OK** to finish these definitions.

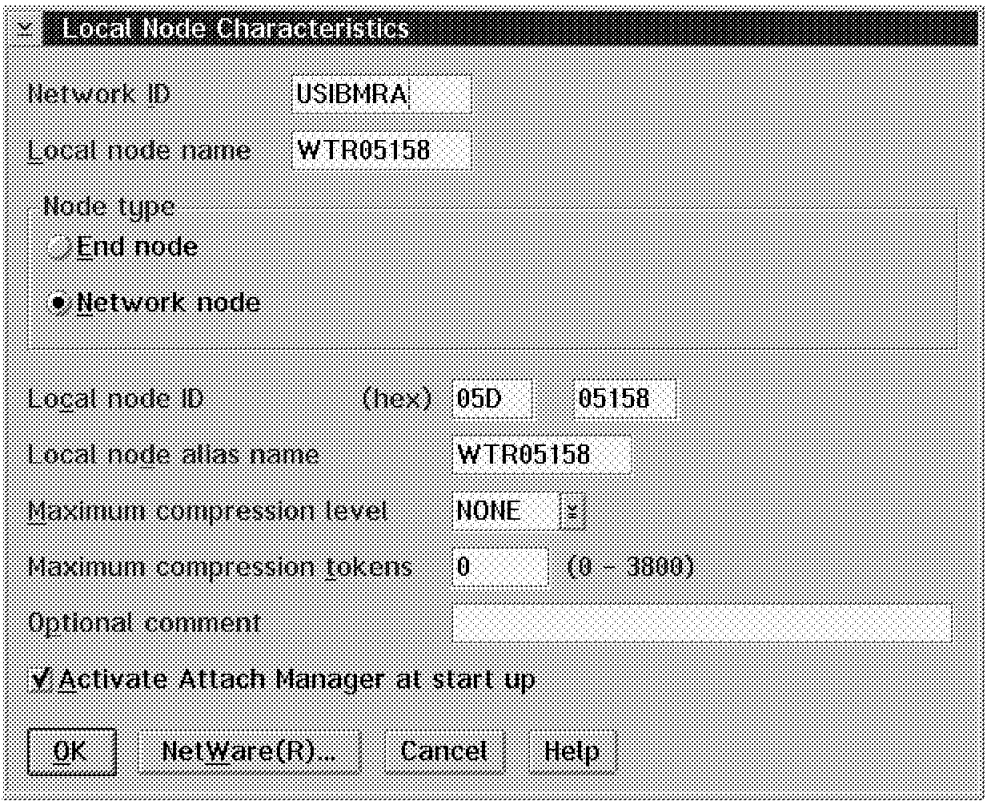


Figure 183. Local Node Characteristics

4. From the Communications Manager Profile List, select **Required -- TN3270E Server parameters**. The TN3270E Server Parameters window appears.

   Click on **Create a link...** to specify a host link name and the number of implicit workstations to create.

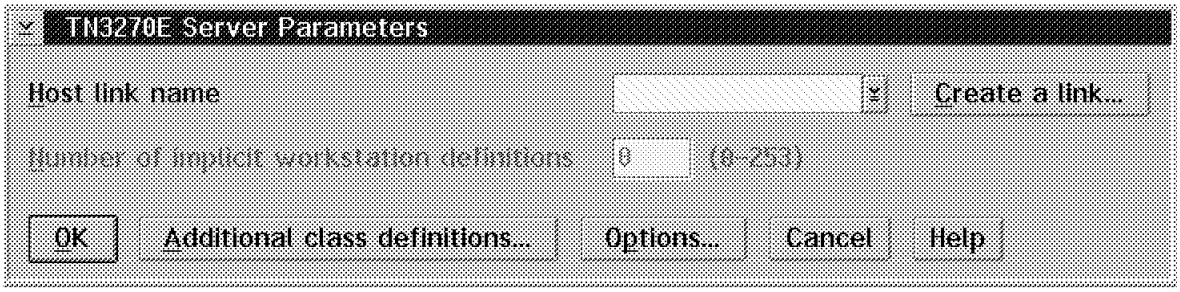


Figure 184. TN3270E Server Parameters

5. In the DLC configuration section of the Link to Hosts window, choose the DLC type and the adapter number for the logical link definition and click on **Configure...** to configure the DLC adapter parameters.

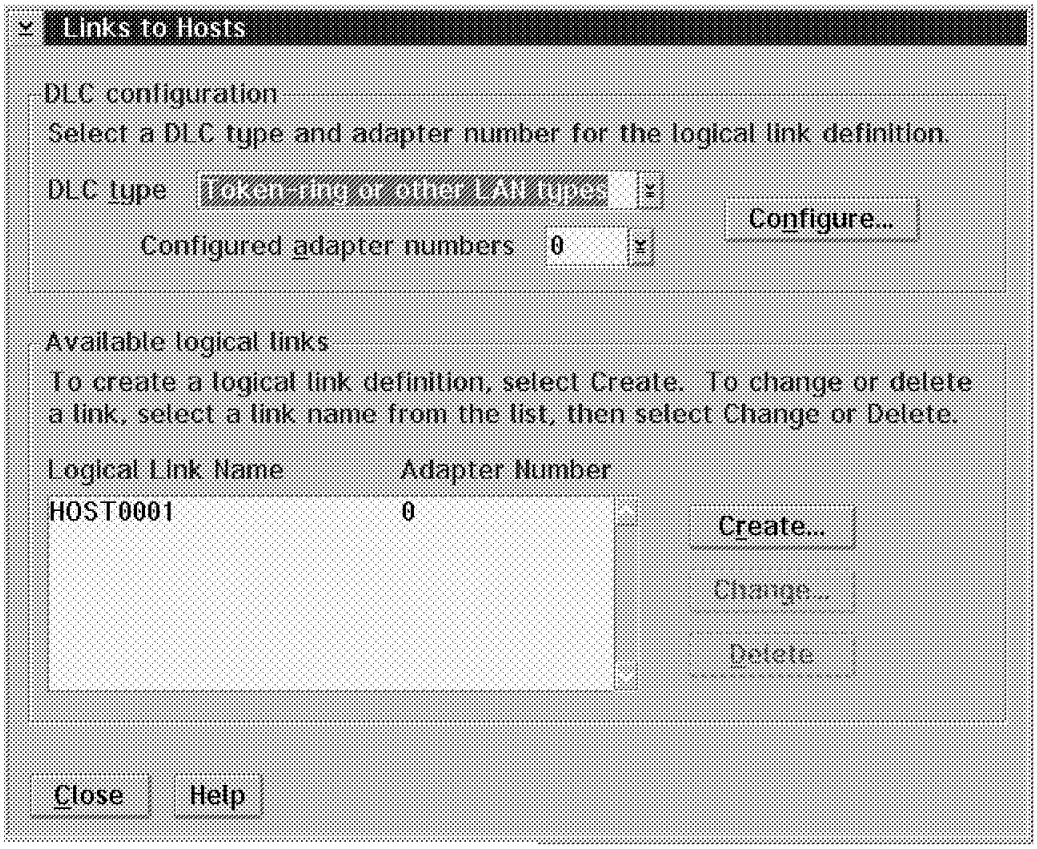


Figure 185. Links to Hosts

6. The parameters used in the Token Ring or Other LAN Types DLC Adapter Parameters window are already set to their default values. Change the parameters due to your needs and finish the definition by clicking on the **OK** push button.

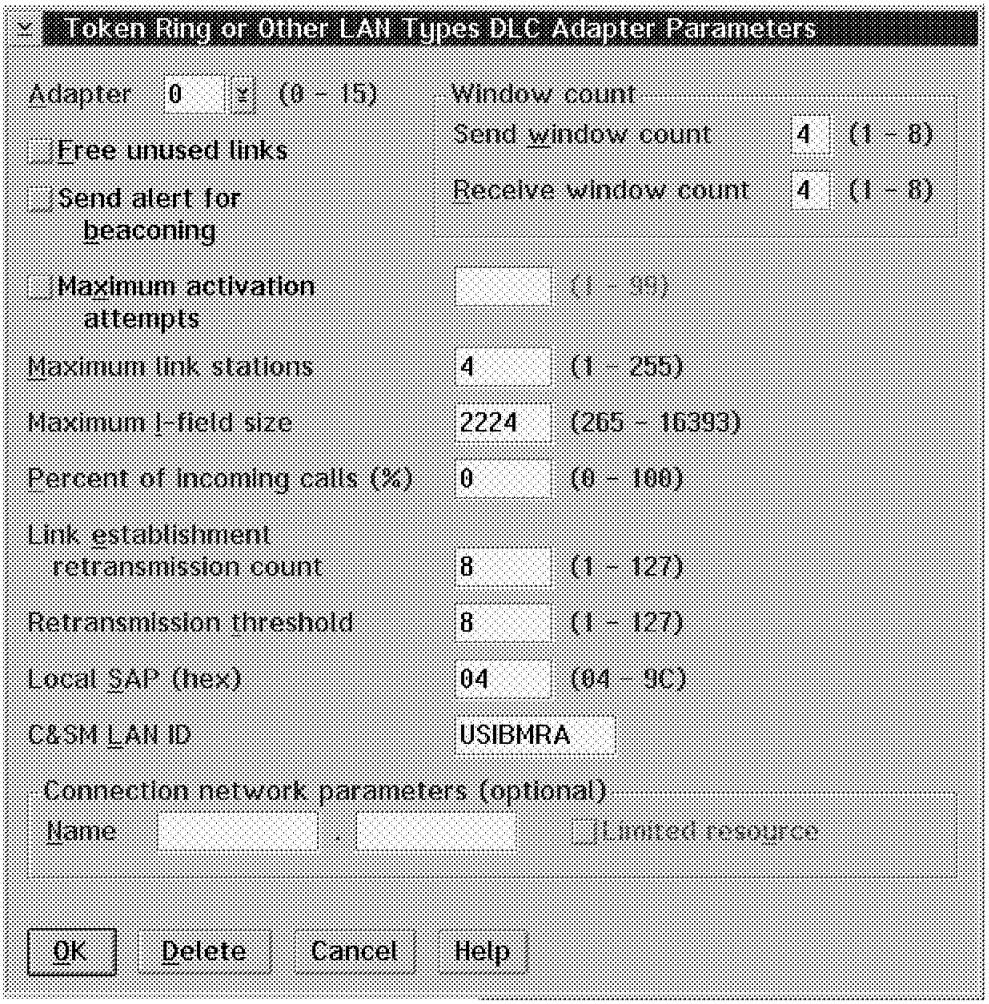


Figure 186. Token Ring or Other LAN Types DLC Adapter Parameters

7. Now that you have defined the adapter parameters, you are able to create a logical link to the host.

   In the Available logical links section of the Links to Hosts window (see Figure 185 on page 208) select **Create...** to create the host link. The Connection to a Host window appears.

   Enter the Link name and the Partner LU definitions such as Partner network ID and Partner node name. Also enter the LAN destination address in the Destination information for host field of Figure 187 on page 210. Click on **OK** to complete these definitions.

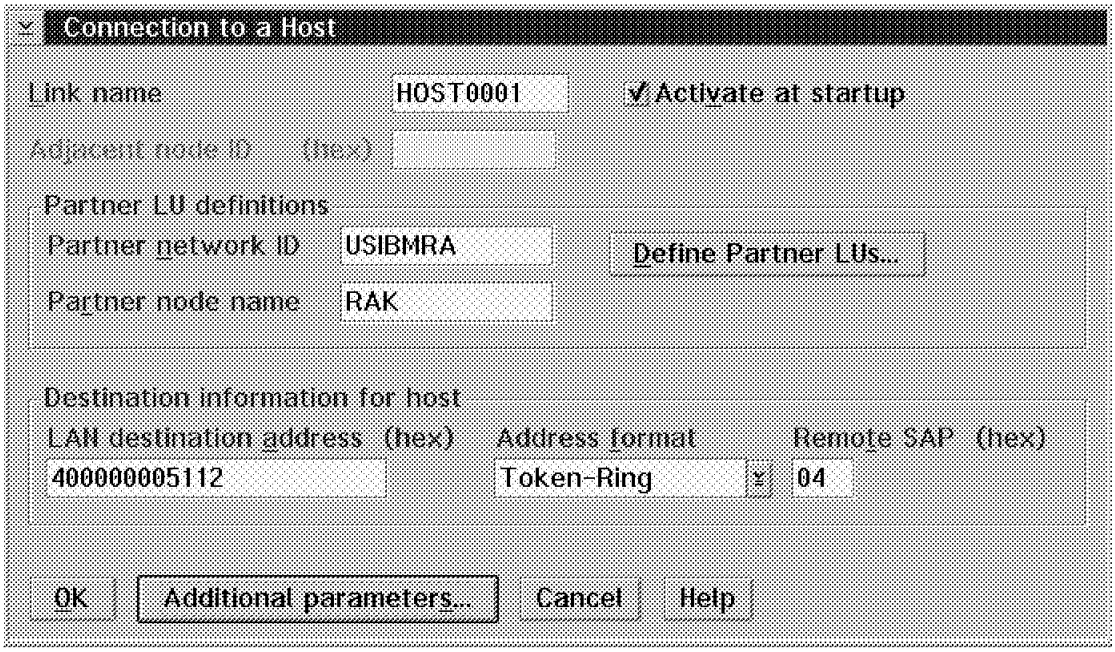


Figure 187. Connection to a Host

8. The Links to Hosts window reappears, now displaying the Logical Link Name and the Adapter Number of the link you created in the Available logical links section.

   Click on **Close** to save and to finish these definitions.

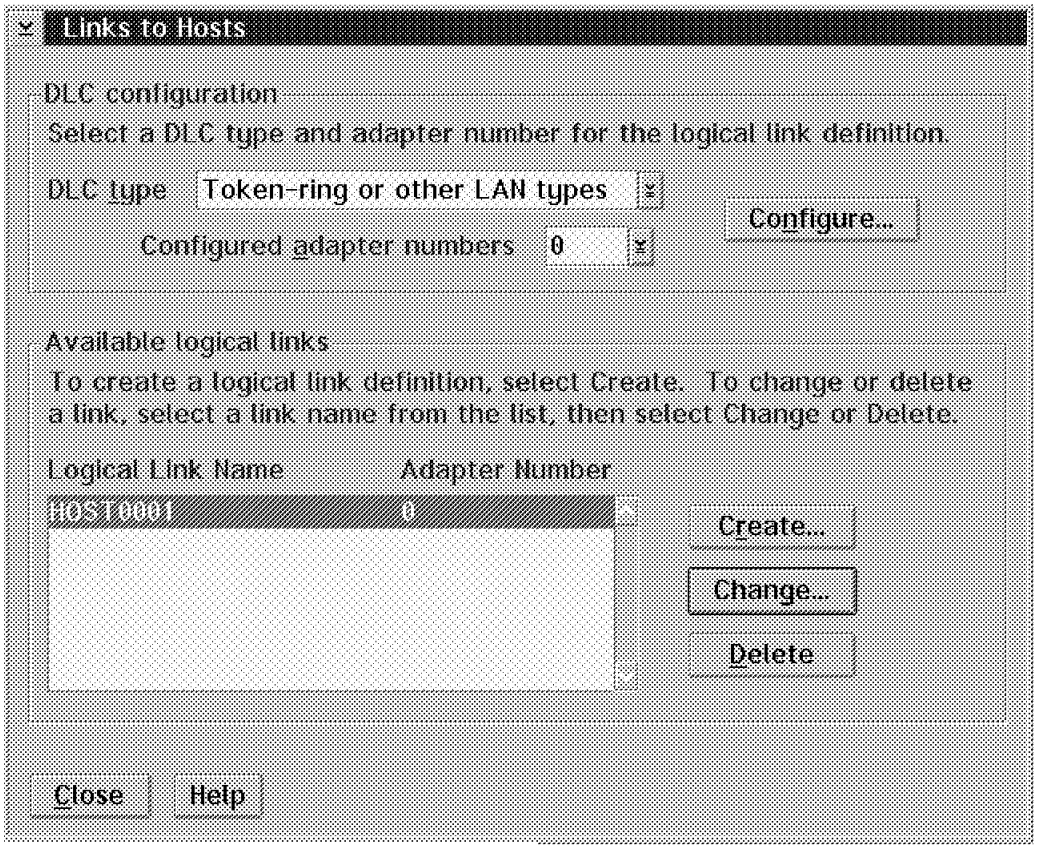


Figure 188. Links to Hosts

9. Back in the TN3270E Server Parameters window, you define the Additional Class Definitions by selecting the appropriate push button.

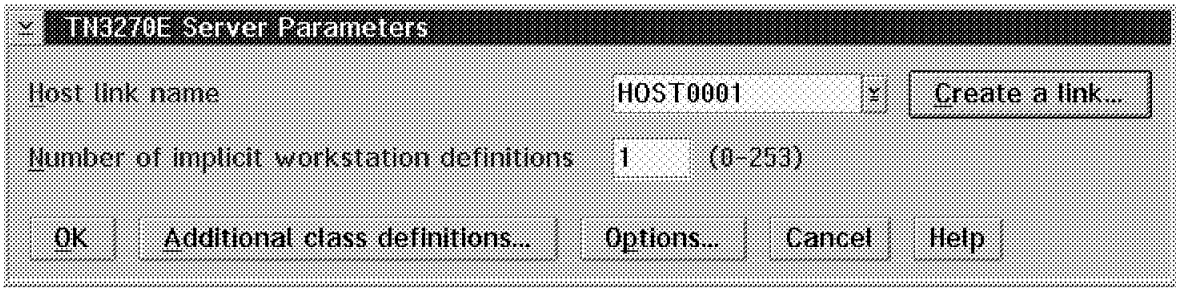


Figure 189. TN3270E Server Parameters

10. In the LUA API Parameters window select if you want to create either multiple LUA definitions for the selected host link or if you want to create individual LUA definitions for the selected host link.

    In this example, only multiple LUA definitions are discussed.

    To create multiple LUA definitions, select this by clicking on the appropriate push button and select **Apply** to apply an LUA definition. To finish the definition, click on **OK**.

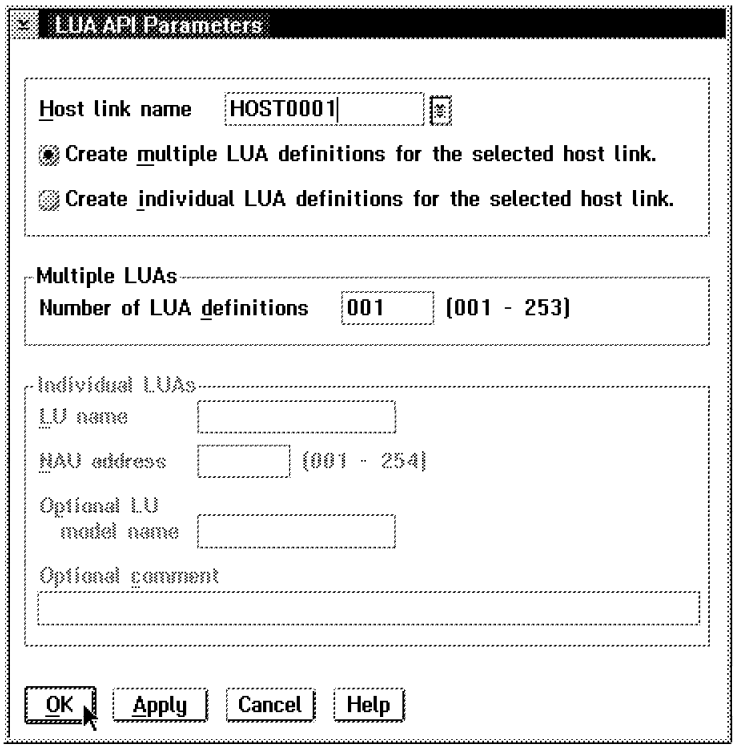


Figure 190. LUA API Parameters

11. The LUA API Definitions window now shows you the LUA definitions. The @ sign in front of LUA0001 means that this LU is an *implicit* LU that is assigned to the host link Host0001. Click on **Close** to end the definition.

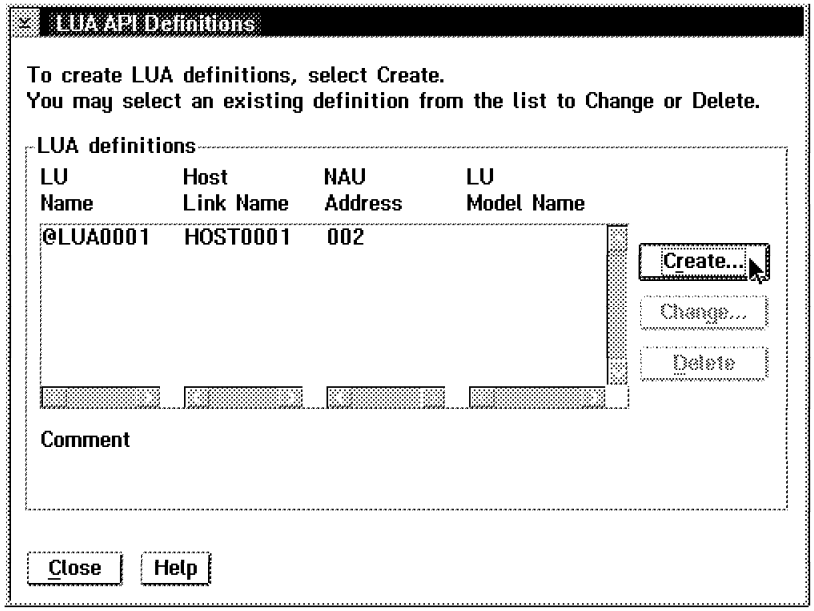


Figure 191. LUA API Definitions

12. The reappearing TN3270E Server Additional Class Definitions window now shows the implicit workstation LU @LAU0001 in the Available LUAs list box.

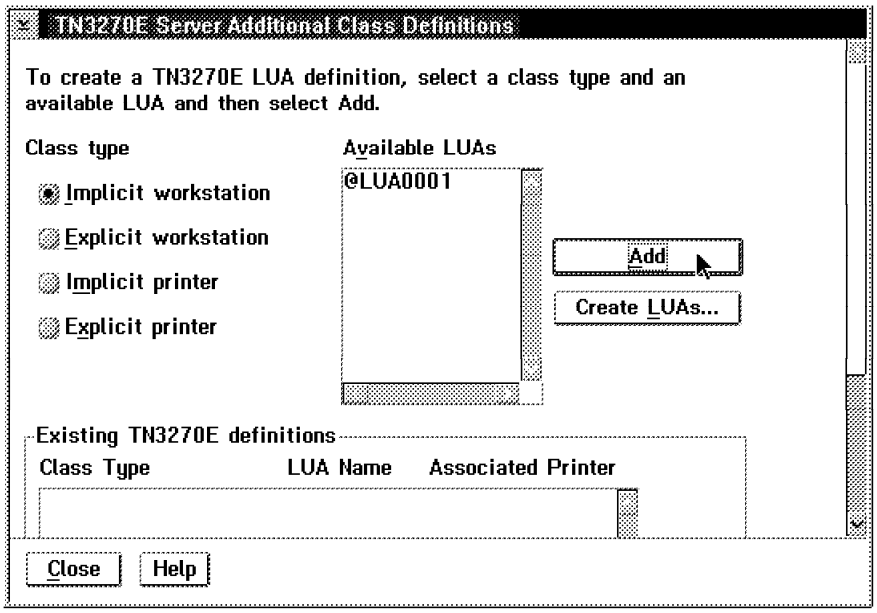


Figure 192. TN3270E Server Additional Class Definitions

13. When selecting the **Add** push button, the LUA definitions are saved and displayed in the Existing TN3270E definitions table. From here, you are able to associate or disassociate a printer for a specific LU or to delete an LU. To create other LUAs, click on the **Create LUAs...** push button. To finish the definitions, click on **Close**.

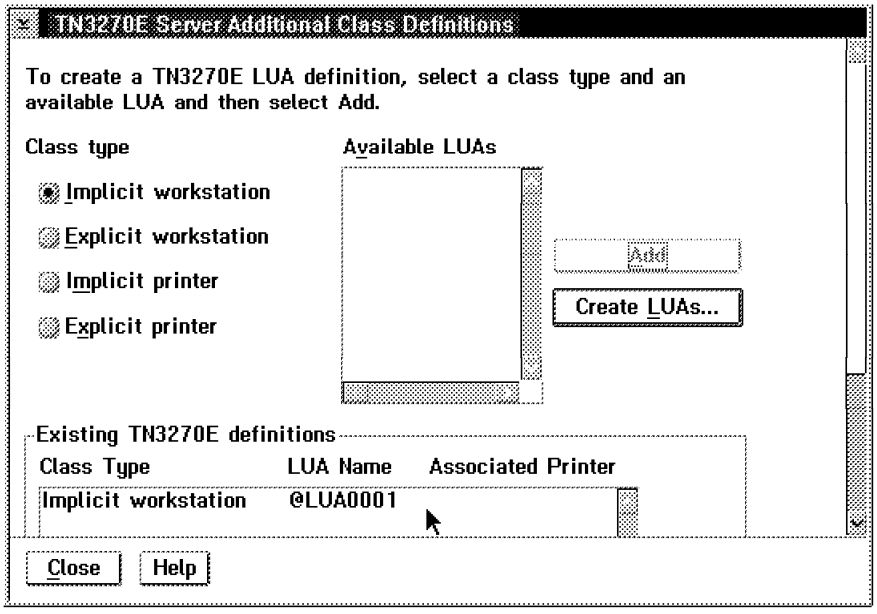


Figure 193. TN3270E Server Additional Class Definitions #2

14. In order to configure optional parameters such as automatic logoff and keepalive processing or to change the port number (default = 23), go to the TN3270E Server Parameters window and click on the **Additional class definition** pushbutton.

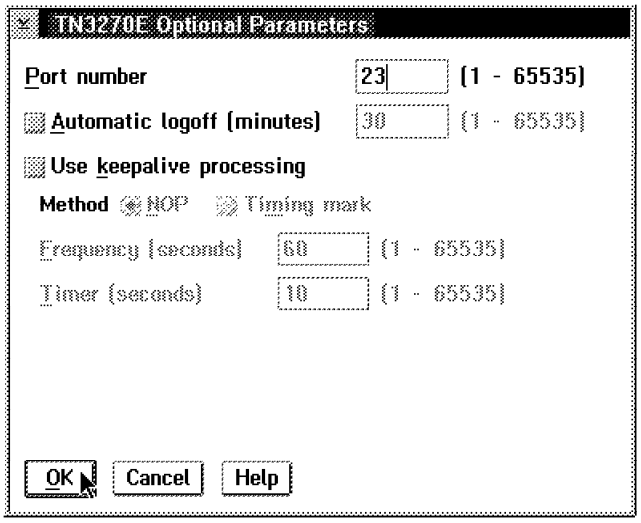


Figure 194. TN3270E Server Optional Parameters

15. When you complete all of your definitions, click on **OK** until you get back to the Communications Manager Profile List.

    Check if the Required profiles are marked. Then close and verify your configuration.

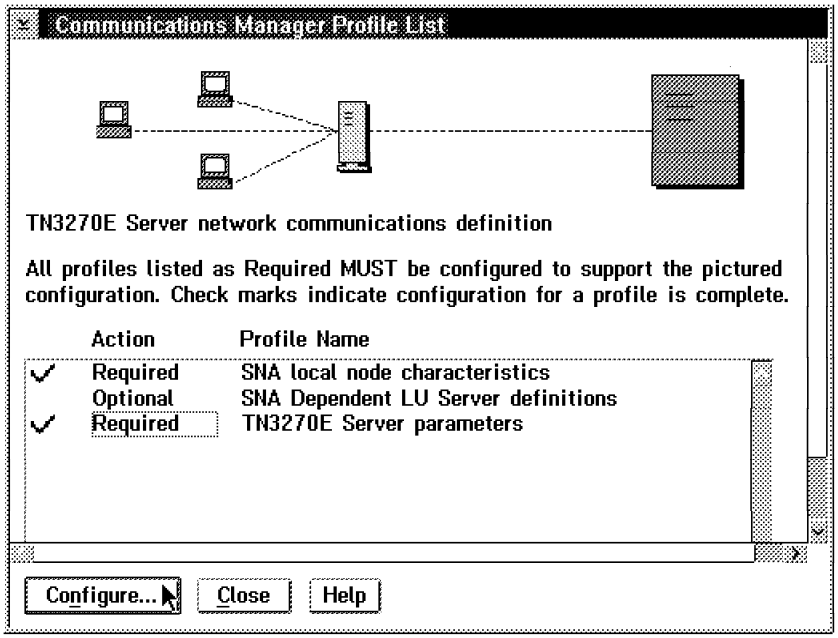


Figure 195. Communication Manager Profile List

9.11 .NDF File Definitions

An example of the node definition file (NDF) of the configuration for TN3270E is shown in Figure 196 on page 216.

Note the DEFINE_LUA section of the .NDF. You can see that the *implicit* LU @LUA0001 is related to the host link HOST0001.

The DEFINE_DEFAULTS section shows the TN3270E optional parameters:

- TN3270E_PORT
- TN3270E_KEEPALIVETYPE
- TN3270E_AUTOMATIC-LOGOFF

You also can see in the DEFINE_TN3270E_LUA section that @LUA0001 is defined as a TN3270E LUA.

```
DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR05158 )
                 CP_ALIAS(WTR05158)
                 NAU_ADDRESS(INDEPENDENT_LU)
                 NODE_TYPE(NN)
                 NODE_ID(X'05D05158')
                         :
                         :
                         :
  DEFINE_LOGICAL_LINK  LINK_NAME(HOST0001)
                       FQ_ADJACENT_CP_NAME(USIBMRA.RAK      )
                       ADJACENT_NODE_TYPE(LEN)
                       DLC_NAME(IBMTRNET)
                       ADAPTER_NUMBER(0)
                       DESTINATION_ADDRESS(X'40000000511204')
                       ETHERNET_FORMAT(NO)
                       CP_CP_SESSION_SUPPORT(NO)
                       SOLICIT_SSCP_SESSION(YES)
                             :
                             :
                             :
       DEFINE_LUA  LU_NAME(@LUA0001)
              HOST_LINK_NAME(HOST0001)
              NAU_ADDRESS(2);

  DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                   DEFAULT_MODE_NAME(BLANK)
                   MAX_MC_LL_SEND_SIZE(32767)
                   DIRECTORY_FOR_INBOUND_ATTACHES(*)
                   DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                   DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                   DEFAULT_TP_CONV_SECURITY_RQD(NO)
                   MAX_HELD_ALERTS(10)
                   DEFAULT_ROUTING_PREFERENCE(NATIVE_FIRST)
                   IMPLICIT_LINK_HPR_SUPPORT(YES)
                   RETRY_COUNT(6)
                   ALIVE_TIMER(60)
                   PATH_SWITCH_TIMER_LOW(480)
                   PATH_SWITCH_TIMER_MEDIUM(240)
                   PATH_SWITCH_TIMER_HIGH(120)
                   TN3270E_PORT(23)
                   TN3270E_KEEPALIVE_TYPE(NONE)
                   TN3270E_AUTOMATIC_LOGOFF(0)
                   DISABLE_DLUR_REGISTRATION(NO);

  START_ATTACH_MANAGER;

  DEFINE_TN3270E_LUA  LUA_NAME(@LUA0001)
                      CLASS(IMPLICIT_WORKSTATION);
```

Figure 196. NDF File of TN3270E Configuration

The node definition file (NDF) may be manually edited to update or add several parameters.

After editing the NDF file, the configuration must be verified. To do so, enter the command:

```
CMVERIFY <configuration_name>
```

9.12 Response File Sample

Figure 197 on page 217 shows an example of a response file, created when verifying a TN3270E Server configuration:

```
LUA=(
  NAME=@LUA0001
* COMMENT is either not used or not configured.
  HOST_LINKNAME=HOST0001
  NAU_ADDRESS=2
* LU_MODEL_NAME is either not used or not configured.
)
SNA_DEFAULTS=(
* COMMENT is either not used or not configured
  DEFAULT_TP_CONV_SECURITY_RQD=0
  DEFAULT_TP_OPERATION=2
  DEFAULT_TP_PROGRAM_TYPE=0
  DIRECTORY_FOR_INBOUND_ATTACHES=*
  IMPLICIT_INBOUND_PLU_SUPPORT=1
  DEFAULT_MODE_NAME=BLANK
* DEFAULT_LOCAL_LU_ALIAS is either not used or not configured.
  MAX_HELD_ALERTS=10
  MAX_MC_LL_SEND_SIZE=32767
  IMPLICIT_LINK_HPR_SUPPORT=1
  RETRY_COUNT=6
  ALIVE_TIMER=60
  PATH_SWITCH_TIMER_LOW=480
  PATH_SWITCH_TIMER_MEDIUM=240
  PATH_SWITCH_TIMER_HIGH=120
  DEFAULT_ROUTING_PREFERENCE=0
  TN3270E_PORT=23
  TN3270E_KEEPALIVE_TYPE=1
  TN3270E_KEEPALIVE_FREQ=60
  TN3270E_KEEPALIVE_TIMER=10
  TN3270E_AUTOMATIC_LOGOFF=0
)
TN3270E_LUA=(.
  NAME=@LUA0001
* COMMENT is either not used or not configured.
  CLASS=2
* ASSOCIATED_PRINTER is either not used or not configured.
)
```

Figure 197. RSP-File of TN3270E Configuration

9.13 Command Line Interface

The following command line interfaces were added for TN3270E Server:

- CMQUERY <opt> TN3270E
  - Where <opt> is:
    - A activate
    - H Deactivate immediately

- S Deactivate after all active connections have completed (same as H for TN3270E)

- CMTN3270 <opt>
  - Where <opt> is:
    - The default is to display all TN3270E Server connections.
    - A display only active connections.
    - C [class type] - display TN3270E Server connections for LUs of the specified class type.
    - D [luname] - display details for the specified connection (same as session details in SSM).
    - H [luname] - deactivate the specified connection.
    - I [ip-address] - display the connection(s) associated with a specific IP address.
    - L [LU name] - display the status of a specific LU.
    - P display TN3270E option settings.
- CMTRACE
  - /dlc TN3270E
  - /event 41

9.14 Subsystem Management Panels

The new TN3270E Server function also contains panels for the Subsystem Management.

1. You can select the TN3270E Server Session selections added from the Details drop-down list.

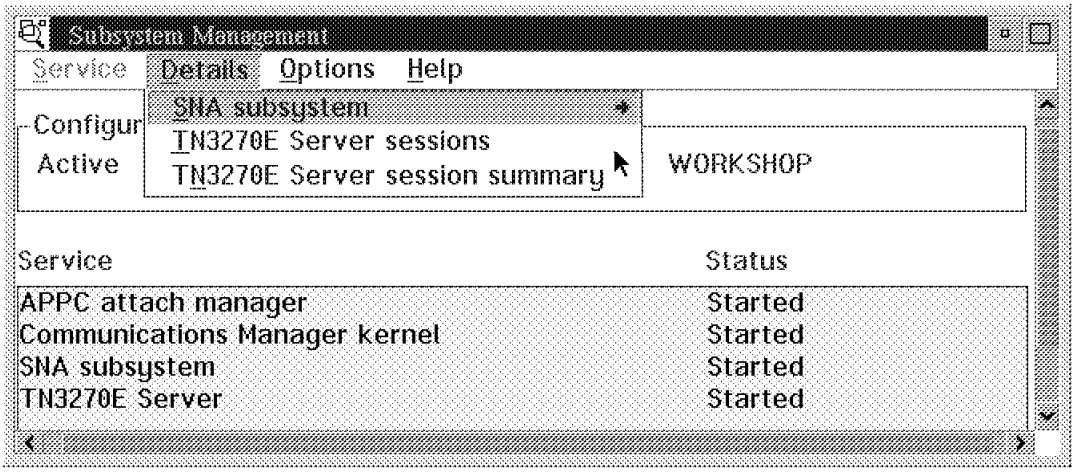


Figure 198. Subsystem Management

2. When you select **TN3270E Server sessions**, the related window appears, showing you all configured TN3270E Server sessions including LU Name, Client Connect Status, IP Address, Class, Idle Time and Associated LU Name. This basically shows the same things that you can see when you enter CMTN3270 on the command line.

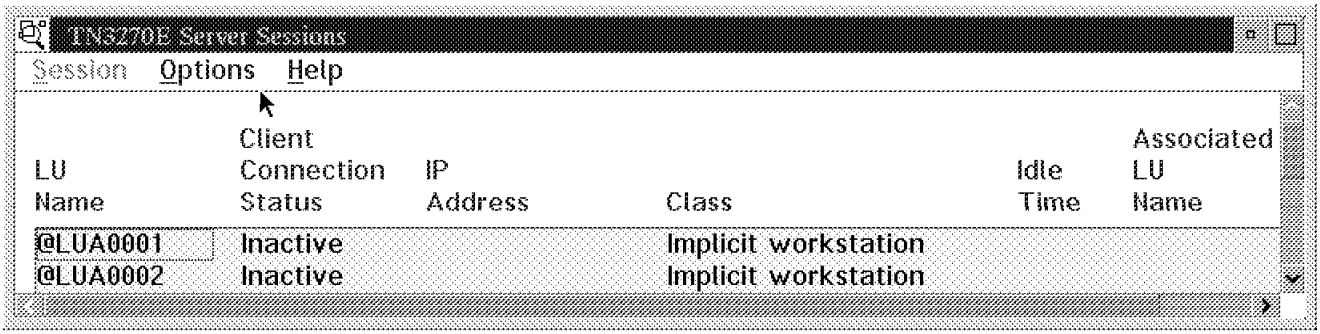


Figure 199. Subsystem Management

3. You also have the choice to display only certain kinds of sessions. Click on the **Option** drop-down box of the TN3270E Server Sessions window.

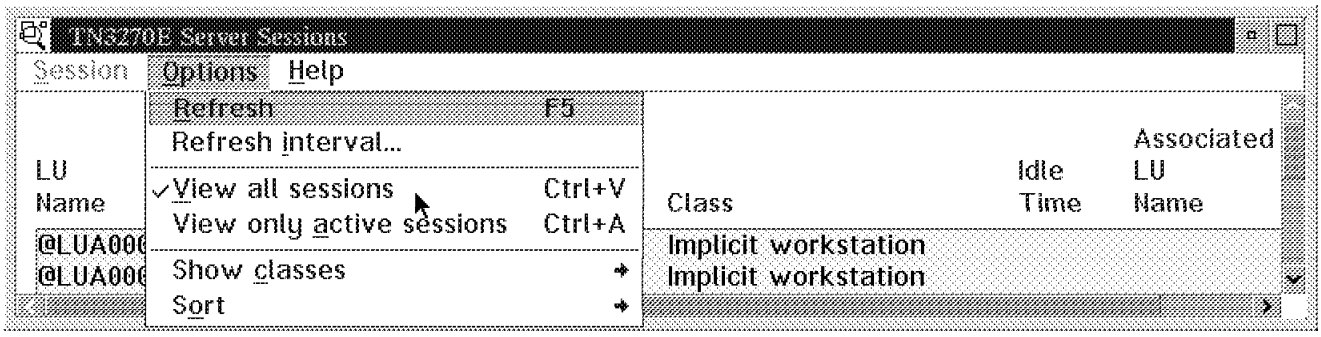


Figure 200. TN3270E Server Sessions

4. The Server Session Summary window displays a summary of the configured TN3270E Server Sessions. Here, mainly the Class Type of the LUs and the Number of LUs Configured as well as the Number of LUs in Use are displayed.

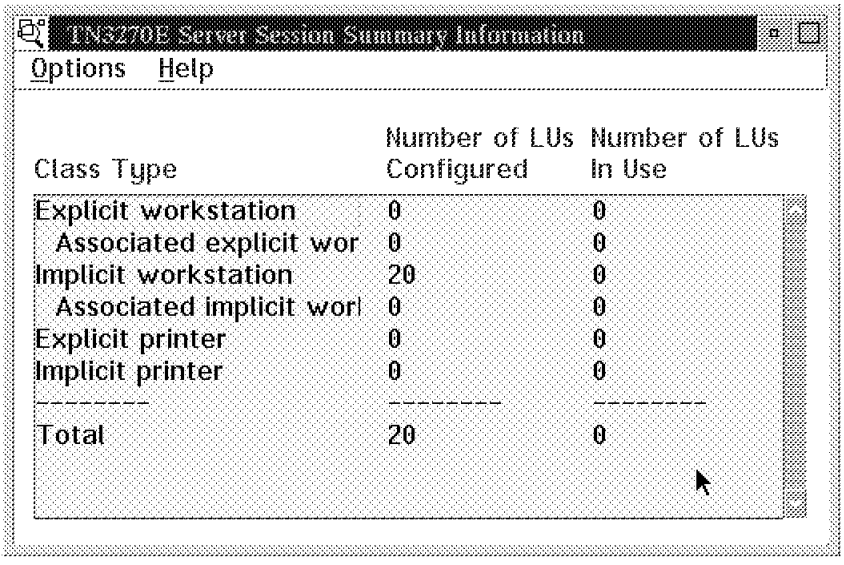


Figure 201. TN3270E Session Summary Information

9.15 Problem Determination

In this section, we show you how to set traces for problem determination of the TN3270E function.

9.15.1 Trace

For problem determination purposes, IBM Communications Server Release 4.1 provides appropriate trace options.

To take a TN3270E Server Trace, proceed as follows:

- Open the Communications Server folder.
- Open the Problem Determination Aids - Trace folder.
- Select:
  - API - **LUA_RUI**
  - DLC - your communications medium, **TN3270E Server**
  - Event -**LUA** and **TN3270E Server Internal**

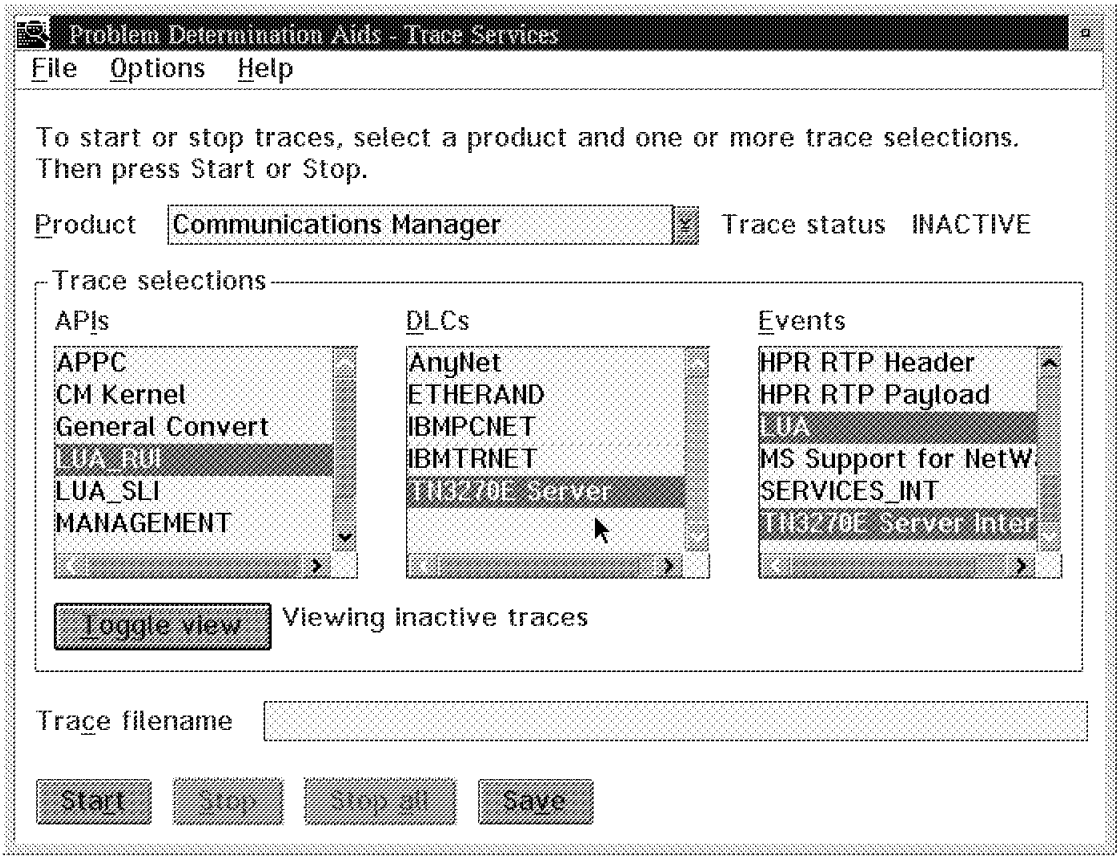


Figure 202. Problem Determination Aids - Trace Services

For complete problem determination activities, the standard APPC Traces and the related FFST/2 Log entries are recommended.

Chapter 10. DLUS-Served LU Registration

The DLUR (dependent LU requester) function in Communications Server has been enhanced in Communications Server to support DLUS-Served LU Registration. With this new support, DLUS/DLUR performance improves by eliminating the need to broadcast in order to search for LUs. In this chapter, we present an overview of this function.

10.1 Introduction

Communications Server enables the support in VTAM V4R2 for dependent LUs through APPN networks and combined subarea and APPN networks. The dependent LU server function (in VTAM) provides dependent secondary logical unit (SLU) support by establishing an LU 6.2 session between a dependent LU requester (DLUR) node and a dependent LU server (DLUS) node.

A DLUR is an APPN end node or network node that owns dependent LUs, but requests that a DLUS provide the system service control point (SSCP) for those dependent LUs. A DLUS controls conversation from a subarea environment to an APPN environment, allowing you to maintain management of remote dependent LUs while benefiting from an APPN network.

DLUR allows dependent LUs (LU 0, 1, 2, 3, and dependent LU 6.2) to benefit from an APPN network. It supports dynamic and multiple paths through the network and eliminates the need for dependent LUs (or their gateways) to be adjacent to the VTAM host.

10.2 DLUS-Served Registration Overview

This function allows an end node DLUR to register its LU so that the network node can locate these LUs without having to pass the locate requests to the DLUR. See Figure 203 on page 222.

That is, an end node is now able to register its *dependent LUs* at its preferred NN server. These dependent LUs become entries in the *Topology Database* of the NN server. Because the topology database of each NN server is known by the entire network, the DLU server (VTAM) now knows the location of the dependent LUs and is therefore able to send DLUR requests directly to them. There is no need for the DLU server to send broadcast searches to locate the dependent LUs. It simply does a locate search for the dependent LUs within its own topology database.

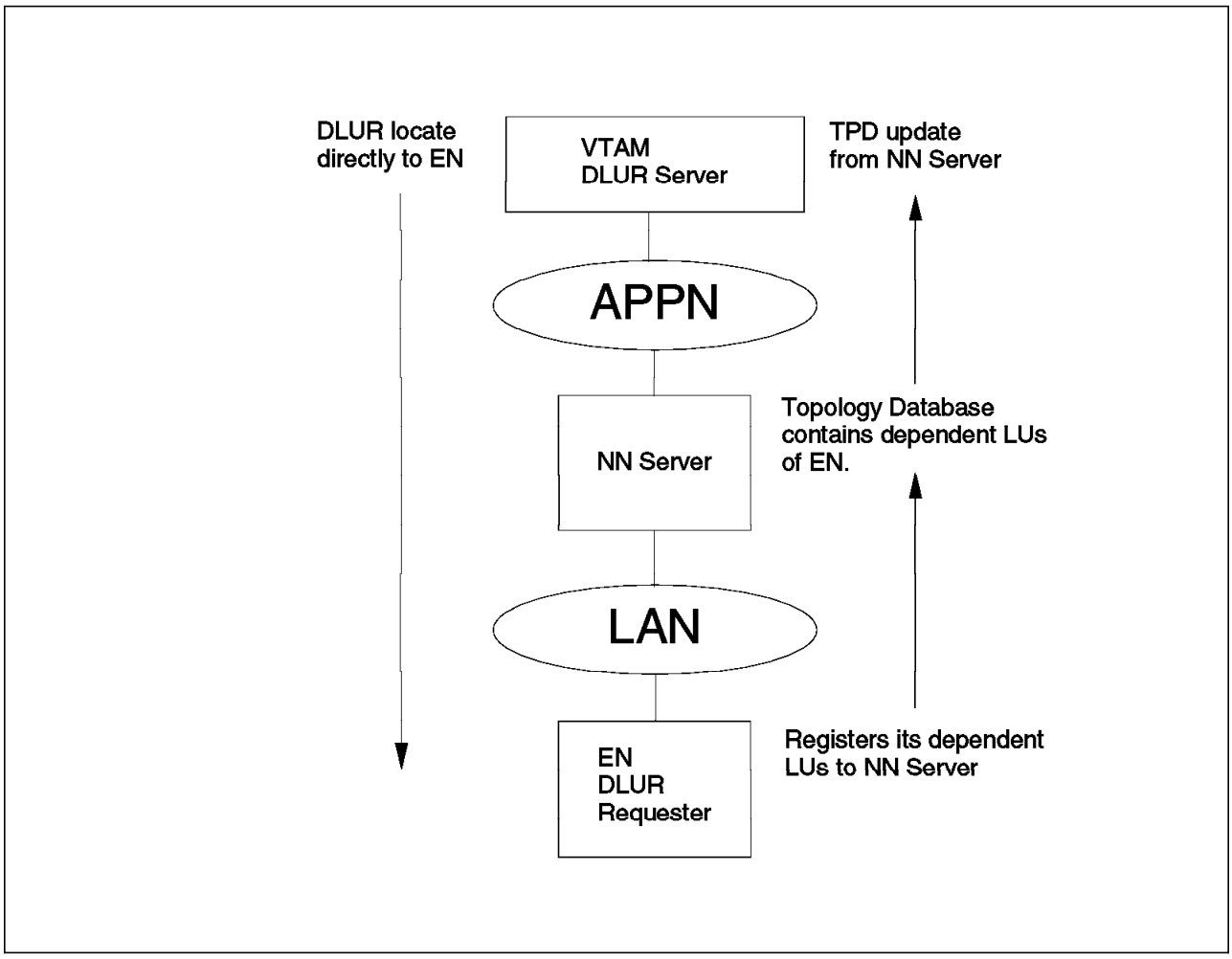


Figure 203. DLUR Schematic

10.3 DLUR Overview

In this section, we show you an overview on how the DLUS/DLUR pipe is established in order to encapsulate the SSCP-PU session and SSCP-LU sessions.

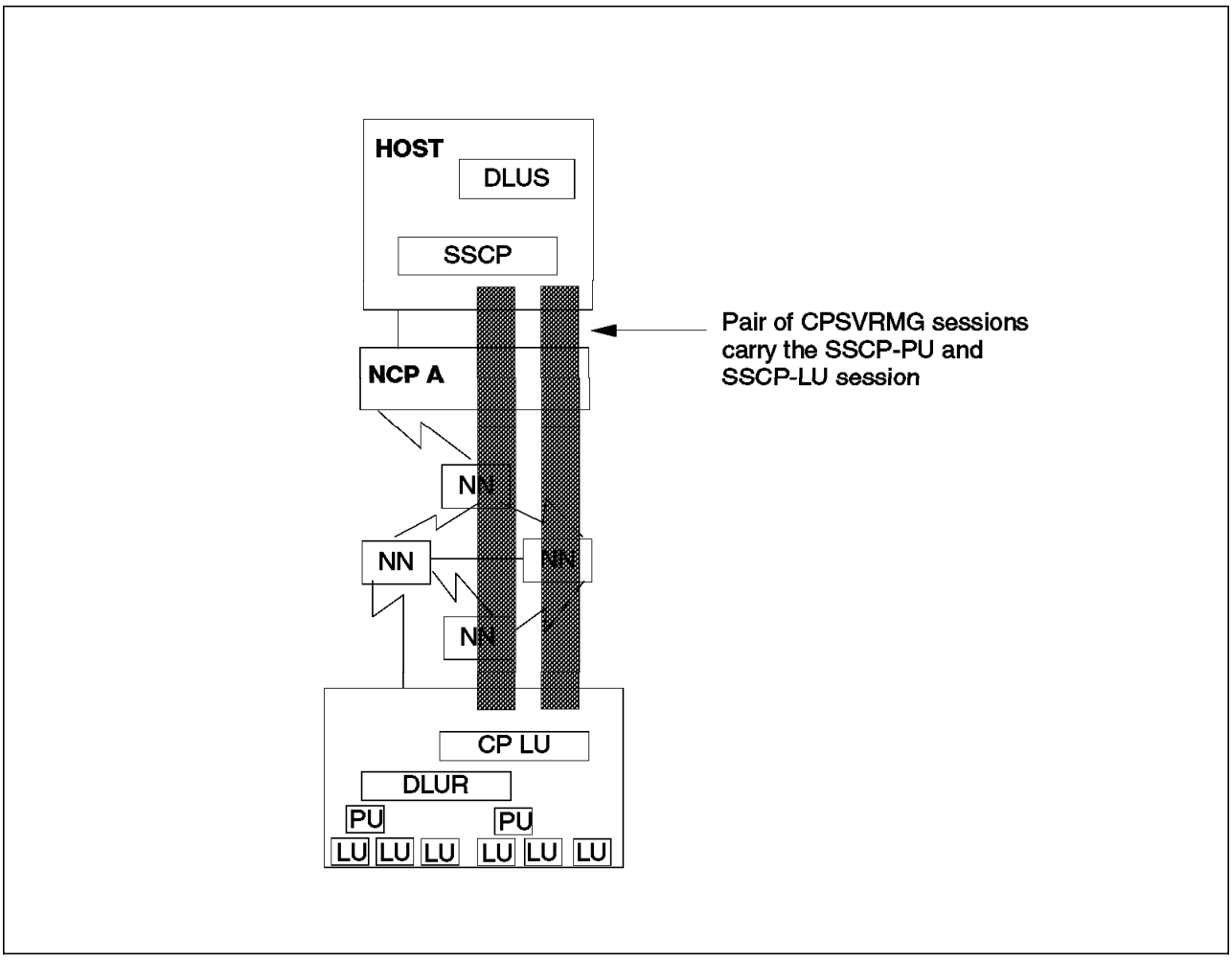


Figure 204. DLUR/DLUS Overview

Connection to the network using connectivity of you choice (for example, token-ring, SDLC, and AnyNet) must be configured and active before the DLUR-DLUS connection can be established. Once an APPN connection exists between the DLUR and DLUS a pair of control sessions are established between the DLUR and DLUS using a special mode, *CPSVRMGR*. (See Figure 204 on page 223.) For one of these sessions, the DLUR is the contention winner, and for the other session, it is the contention loser. This is similar to CP-CP sessions, except that these sessions are reserved exclusively for the use of DLUS/DLUR. This pair of control sessions is also referred to as the *CP-SVR pipe* and appears as a link to Communications Manager. It can therefore be activated, deactivated and displayed using existing utilities, such as SubSystem Management, and sample programs, such as PMdisplay and CMLinks. In the traditional subarea environment, only the SSCP (DLUS) could activate a PU. To support the DLUR initiated activation/deactivation, two new RUs were created, REQACTPU and REQDACTPU.

All data sent on the CPSVRMGR session pipe is contained in a new GDS variable, the *Encapsulation GDS* variable (*GDS X'*1500). The Encapsulation variable is used to carry SSCP-PU and SSCP-LU session traffic on the CPSVRMGR pipe. It contains the entire FID 2 PIU (TH, RH, RU). Once the pipe is activated, SSCP-to-PU and SSCP-to-LU support can be provided to PUs and LUs

that have defined the pipe as their host link. LU-to-LU sessions use the best path available through the network, and they do not use the pipe.

The DLUS node is the owning SSCP for the dependent secondary LUs. Note that the DLUS node *owns* the dependent LUs even though the resources are not physically within the SSCP's domain.

The session setup flows between the DLUS and the primary LU, but the BIND and session data flow directly between the PLU and the SLU. (See Figure 205 on page 224.)

Another way to look at this, for those that are more familiar with the subarea terminology of boundary function, is to say that the DLUR function extends the NCP boundary function right down to the node where the DLUR code resides. So think of this as taking a piece of the NCP which resides at the host site and extending it down to the node through the use of an LU 6.2 session pipe. Therefore, for all practical purposes, and from the perspective of the PU in the downstream node, the NCP appears to be adjacent since the DLUR pretends to be an NCP.

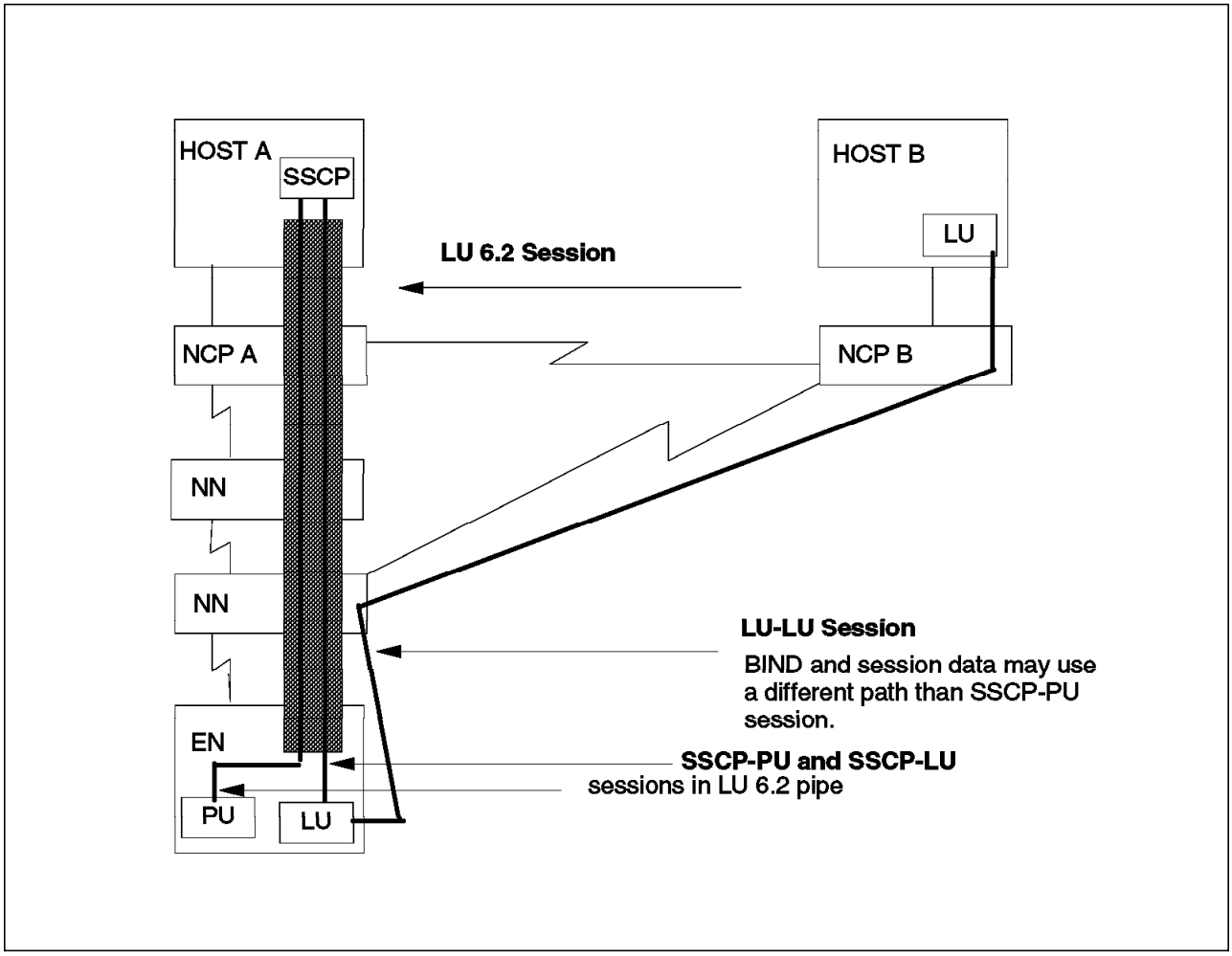


Figure 205. DLUR/DLUS Session Flow

10.3.1 Benefits and Values of DLUR

Dependent LU requester, as supported by Communications Server, gives you the following benefits/values:

- Provides a migration strategy for dependent LU devices from a subarea environment to APPN.
- Provides distributed dependent LU support without the need for SSCP function at all APPN nodes. This allows distribution of dependent LUs throughout the network.
- Allows LU-LU sessions to be routed independently of the location of the owning SSCP, thereby taking advantage of APPN search logic.
- Network management continues to be provided by the DLUS node, and the PUs that are not adjacent to an SSCP can still be visible to the host.

10.3.2 Scenario

Next, we show a sample configuration for the following scenario:

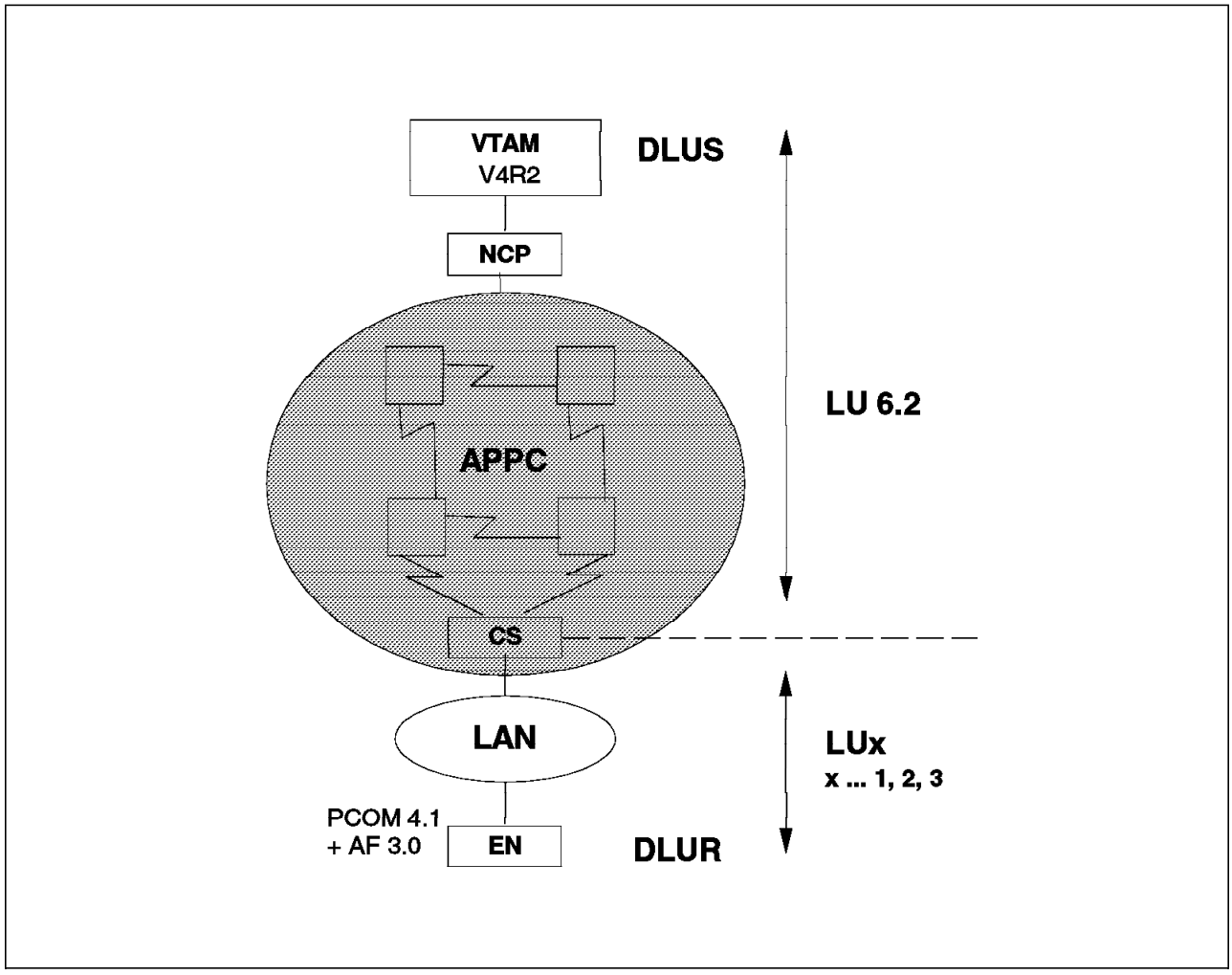


Figure 206. DLUR/DLUS Scenario

10.4 Configuration

To configure for DLUS-Served LU Registration from the scenario shown in Figure 206 on page 225 do the following:

1. Enter the Communications Manager setup by clicking on the appropriate icon within the Communications Server folder or enter CMSETUP on the command line.
2. In the Communications Manager Configuration Definition panel select the Workstation Connection Type you want to use and select **LUA APIs with DLUS**.

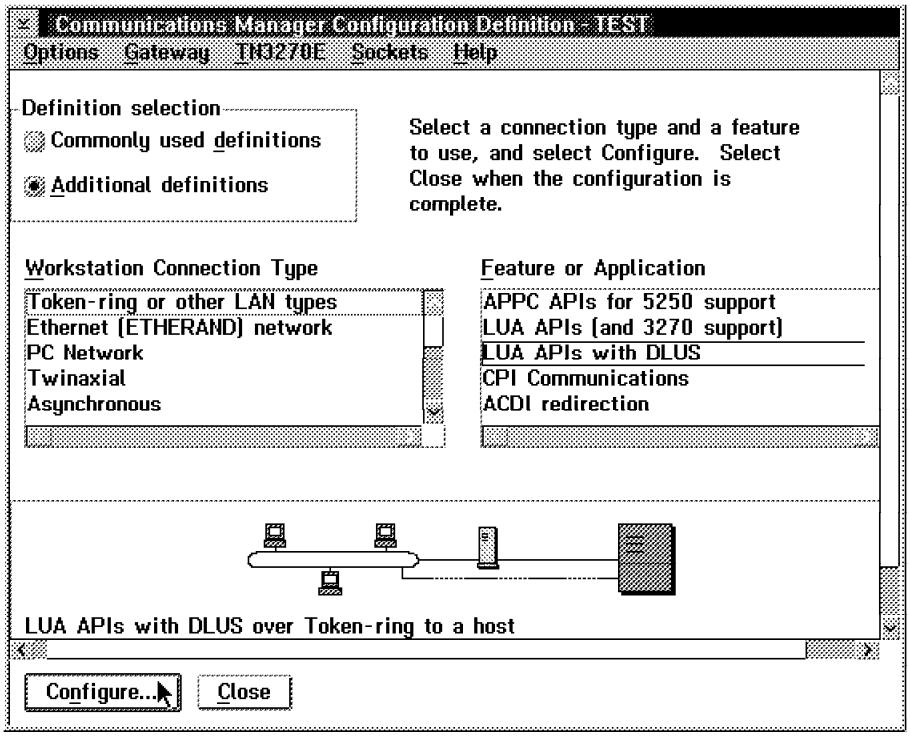


Figure 207. Communications Manager Configuration Definition

3. Configure the DLC - Token-ring or other LAN types and the SNA local node characteristics profiles to meet your needs (see Figure 208 on page 227).

   Click on **SNA connection** to configure an APPN connection to the host system.

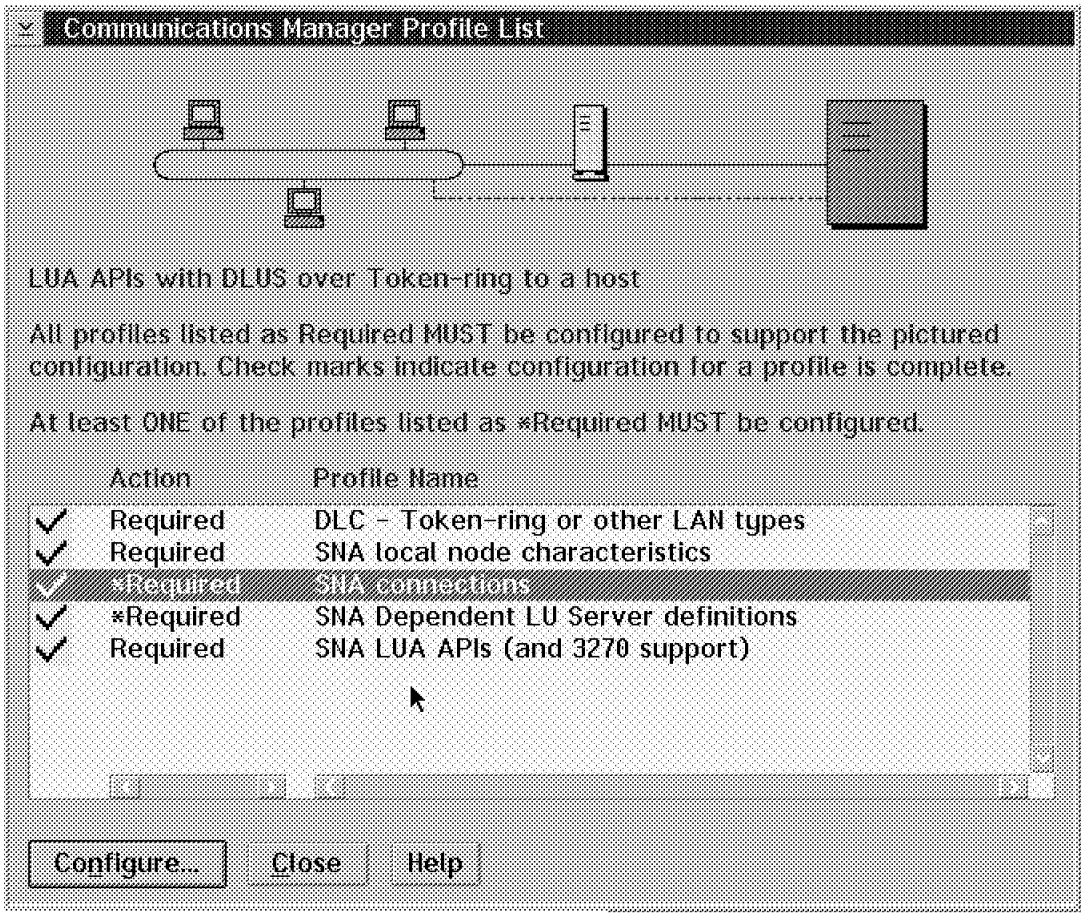


Figure 208. Communications Manager Profile List

4. In the Connection List panel select **To network node** as the partner type and click on **Create...** for further definitions.

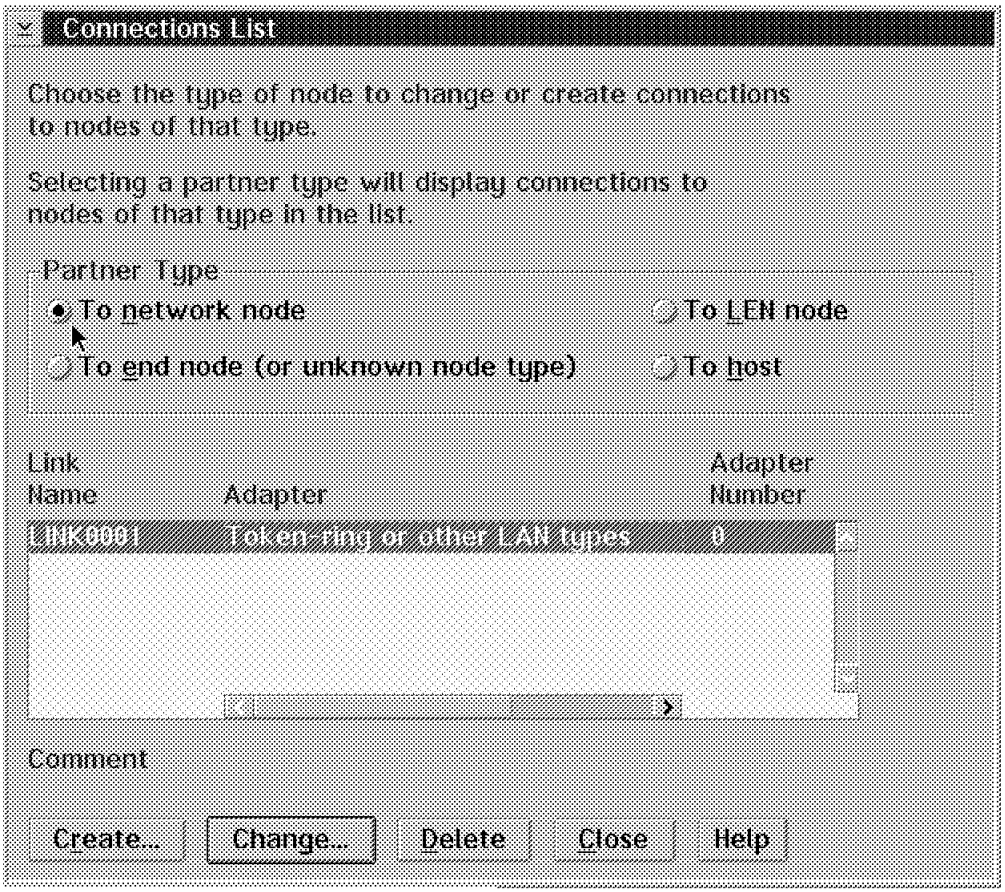


Figure 209. Connection List

5. In the Connection to a Network Node window enter the Link name, Partner LU definitions (optional) and the LAN destination address of the host system. You have the option of clicking on **Additional parameters...** Click on **OK** to save these definitions.

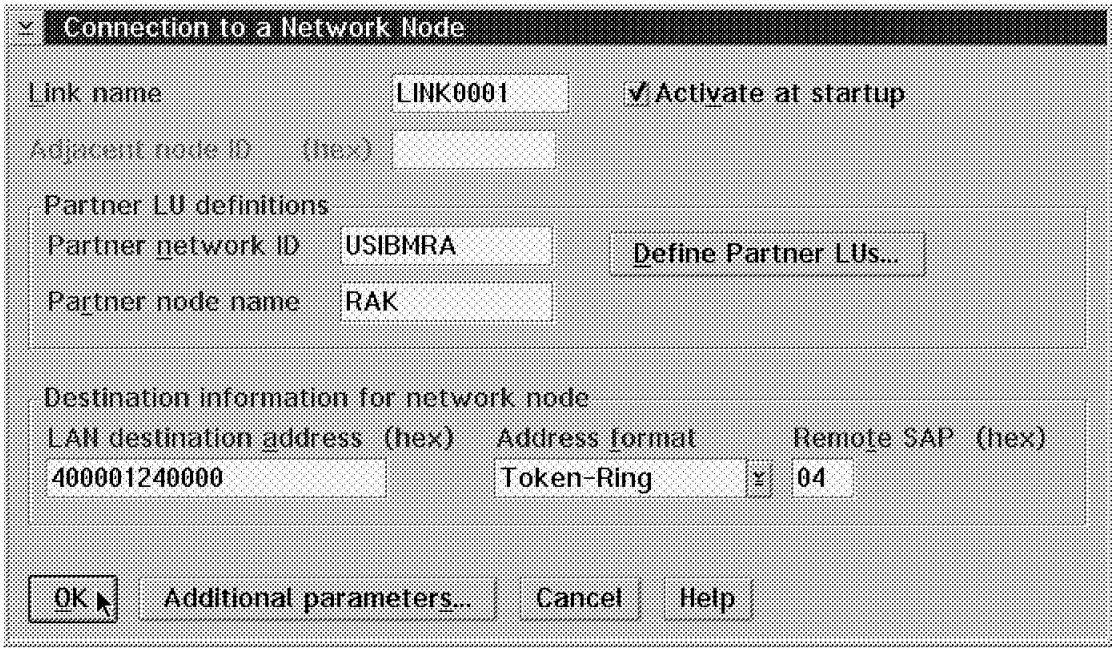


Figure 210. Connection to a Network Node

6. To configure the SNA Dependent LU Server definitions select the appropriate profile from the Communications Manager Profile List panel and click on **Configure...**

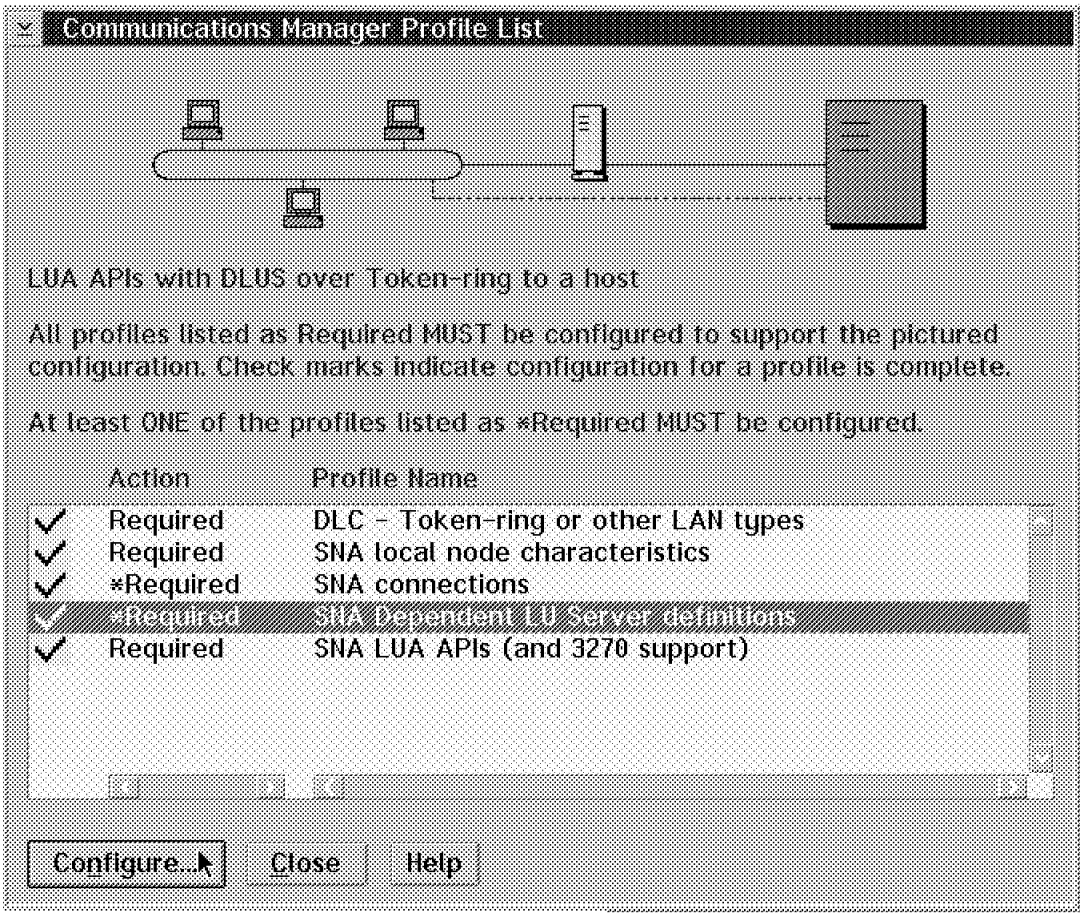


Figure 211. Communications Manager Profile List

7. In the Dependent LU Server Parameters window enter the Link name for the link to the host system, and the Fully-qualified dependent LU server name. The parameter MPU0x is a default parameter created by Communications Manager. You can change it to meet your needs. The parameter Node ID is the ID under which VTAM has generated the entries in the LOGMODE table for dependent LUs. Ask your VTAM coordinator for the proper ID. Finish these definitions by clicking on **OK**.

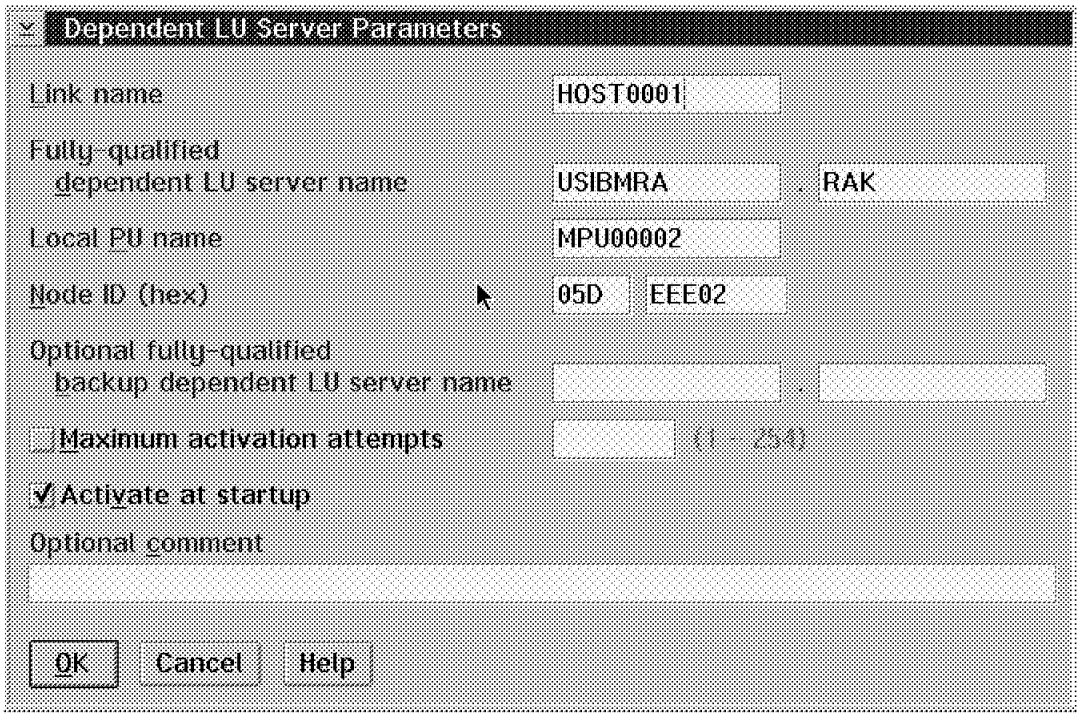


Figure 212. Dependent LU Server Parameters

8. The last required profile on the Communications Manager Profile List panel is SNA LUA APIs (and 3270 support). Click on **Configure...** to configure these parameters.

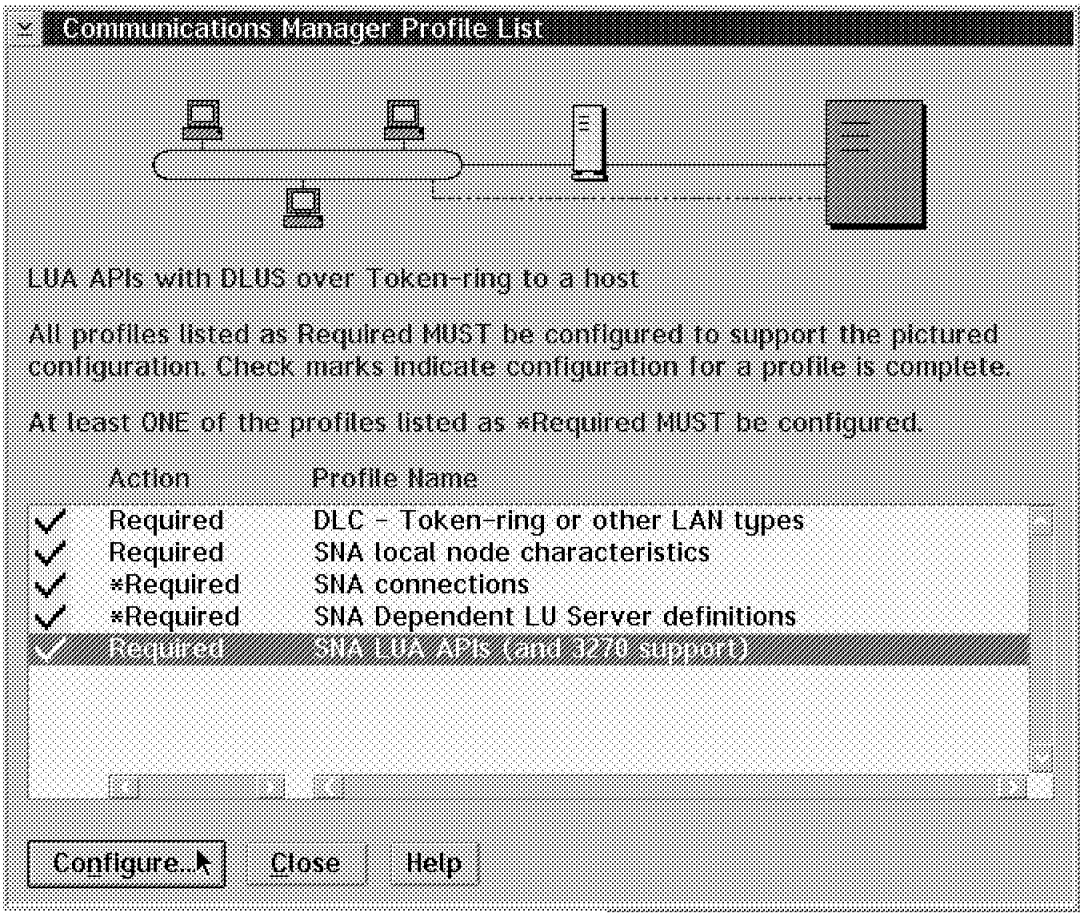


Figure 213. Communication Manager Profile List

9. In the LUA API Definitions window click on **Create** to create new LUA definitions or click on **Change** to change existing LUA definitions.

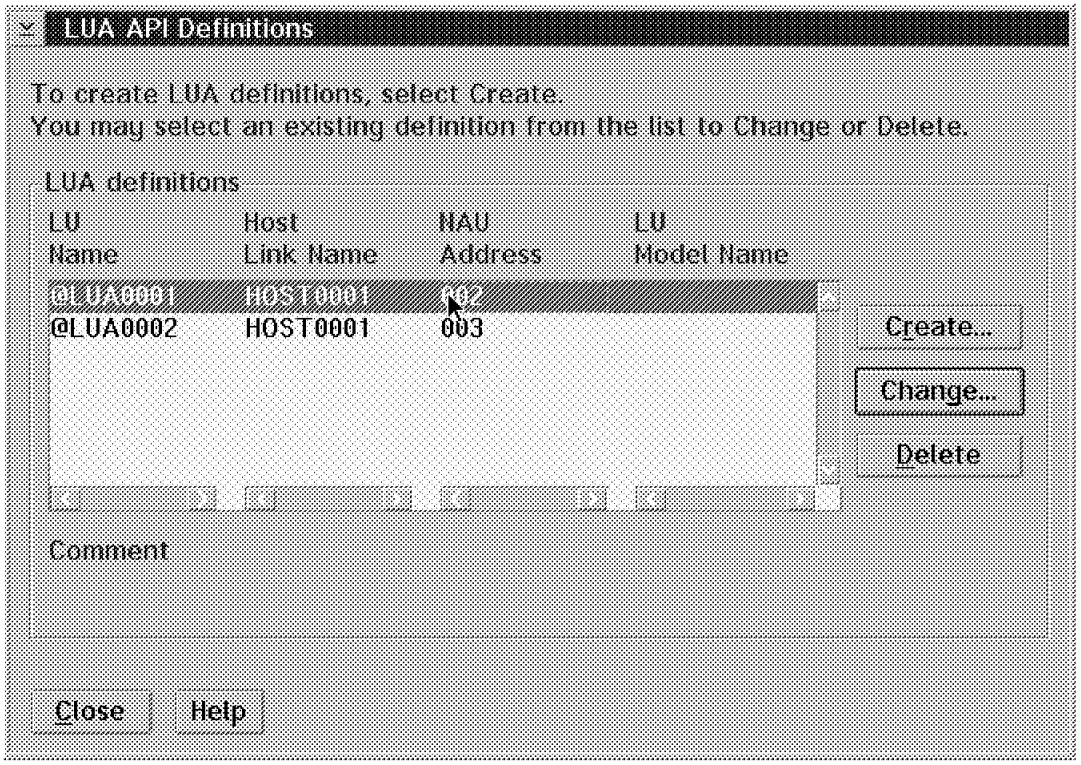


Figure 214. LUA API Definitions

10. Clicking on **Create...** displays the LUA API Parameters window.

    Select the Host link name from the list box and enter the Number of LUA definitions you wish to create. (In this example, we only discuss creation of multiple LUA definitions.)

    To finish these definitions, click on **OK** and return to the LUA API Definitions window.

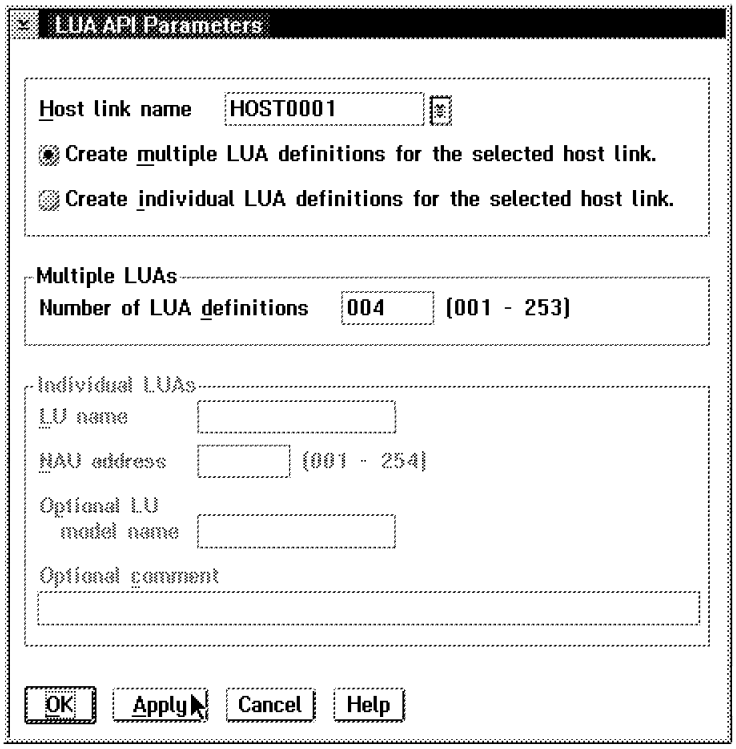


Figure 215. LUA API Parameters

11. The LUA API Definitions window now shows the LUA definitions you created.

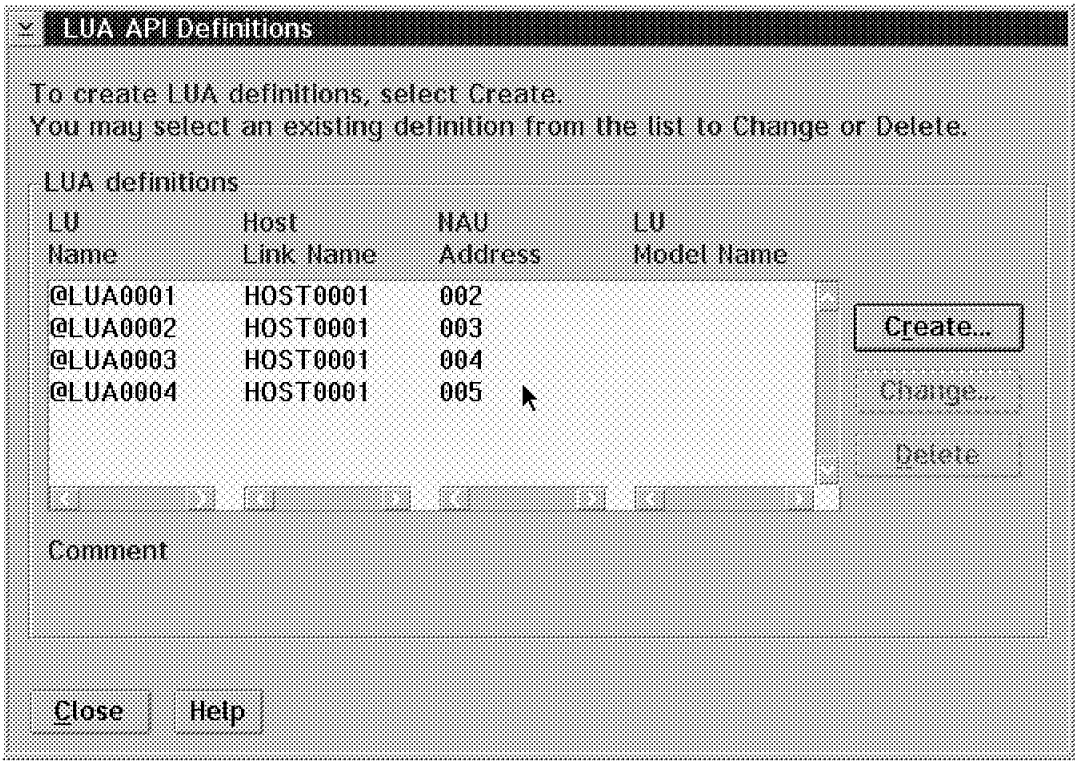


Figure 216. LUA API Definitions

12. Click on **Close** to finish these definitions and to go back to the Communications Manger Profile List to finish the configuration.

10.5 .NDF File Definitions

The .NDF file contains some new statements after configuring for DLUS-Served LU Registration.

These parameters are shown in Figure 217 on page 236.

```
DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR05137 )
                 CP_ALIAS(WTR05137)
                 NAU_ADDRESS(INDEPENDENT_LU)
                 NODE_TYPE(NN)
                 NODE_ID(X'05D05137')
                 NW_FP_SUPPORT(NONE)
                 HOST_FP_SUPPORT(YES)
                 DLUR_MULTISUBNET_SUPPORT(YES)
                 FREE_UNUSED_SESSIONS(NO)
                 FREE_UNUSED_SESSIONS_TIME(10)
                 HOST_FP_LINK_NAME(HOST0001)
                 MAX_COMP_LEVEL(NONE)
                 MAX_COMP_TOKENS(0);
                              :
                              :
  DEFINE_DEPENDENT_LU_SERVER  LINK_NAME(HOST0001)
             FQ_DLUS_NAME(USIBMRA.RAK      )
             PU_NAME(MPU00002)
             NODE_ID(X'05DEEE02')
             MAX_ACTIVATION_ATTEMPTS(INFINITE)
             ACTIVATE_AT_STARTUP(YES);

  DEFINE_LUA  LU_NAME(@LUA0001)
              HOST_LINK_NAME(HOST0001)
              NAU_ADDRESS(2);

  DEFINE_LUA  LU_NAME(@LUA0002)
              HOST_LINK_NAME(HOST0001)
              NAU_ADDRESS(3);

  DEFINE_LUA  LU_NAME(@LUA0003)
              HOST_LINK_NAME(HOST0001)
              NAU_ADDRESS(4);

  DEFINE_LUA  LU_NAME(@LUA0004)
              HOST_LINK_NAME(HOST0001)
              NAU_ADDRESS(5);

  DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                   ALIVE_TIMER(60)
                   PATH_SWITCH_TIMER_LOW(480)
                              :
                              :
              DISABLE_DLUR_REGISTRATION(NO);    <===========
                              :
                              :
  SET_DISCOVERY_SERVER  ADAPTER_NUMBER(0)
                        GROUP_NAMES(IROUTSNA)
                        ROUTING_CAPABILITIES(NN);
```

Figure 217. DLUS-Served LU Registration, NDF File Extract

10.6 Subsystem Management

The SNA Subsystem Management also mirrors the function of the DLUS-Served LU Registration.

The Detail Option LU 6.2 Sessions shows that two CPSVCMG and two CPSVRMGR sessions are online between the network node server WTR05137 and the DLUS (host) USIBMRA.RAK (see Figure 218 on page 237).

There are also two CPSVCMG sessions active between the network node server and the DLUR, the end node WTR01533.

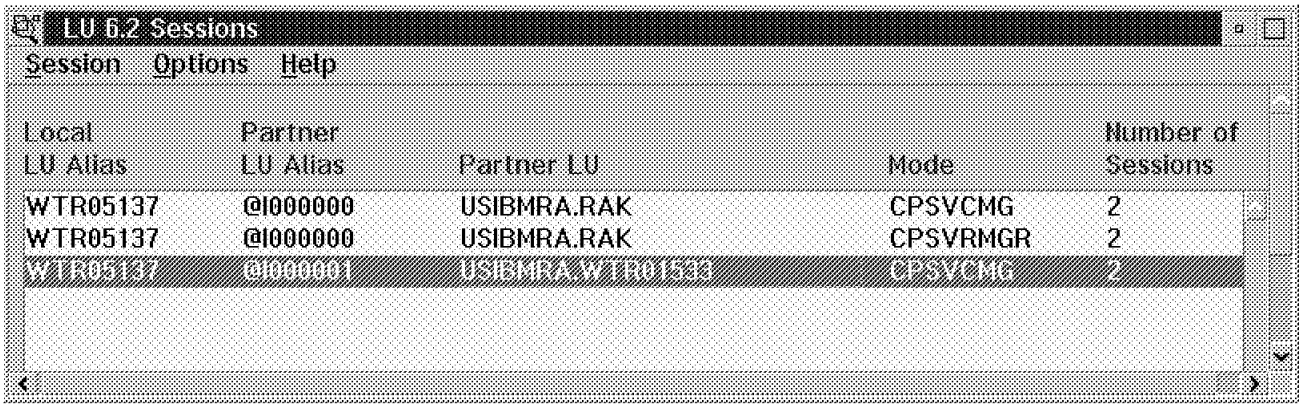


Figure 218. SNA Subsystem Management, Detail LU 6.2 Sessions

In the Detail Option of SNA Subsystem Management, the Logical Links panel shown in Figure 219 on page 237 shows that there are three different links active:

- @AAAAAAA, which is the link from the NN server to the DLUS requester WTR05133 as an EN, running 5 active sessions
- HOST0001, which is the link for the dependent LU sessions between the DLUS and the DLUR
- LINK0001, which is the link we defined for the APPN traffic from the NN server to the host system

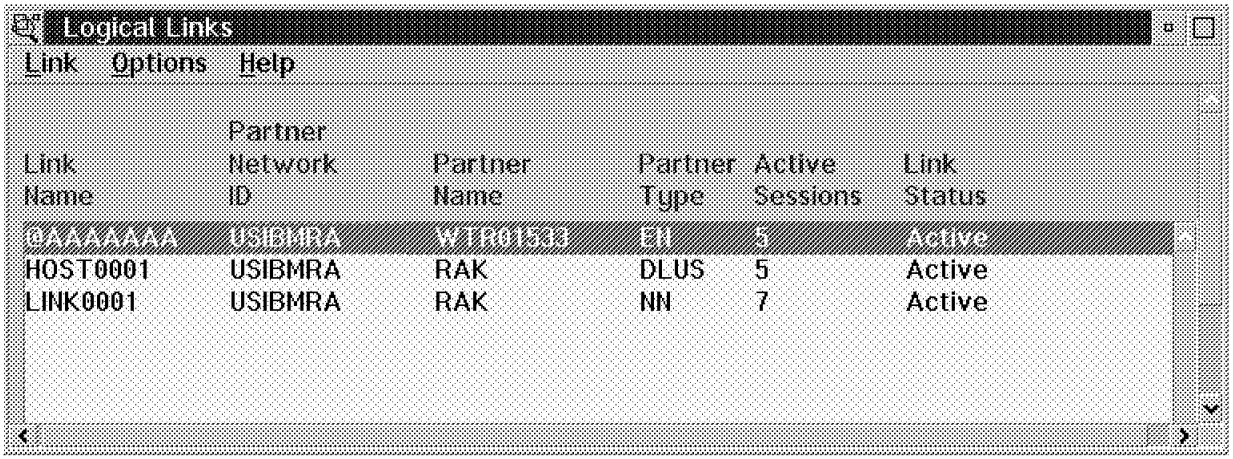


Figure 219. SNA Subsystem Management, Detail Logical Links

10.6.1 Display

A more detailed view of the functions imbedded when configuring and using DLUS-Served LU Registration is provided by the Display function of Communications Server.

Depending on the option you enter when starting display, this function shows, for example, the System Default Information or the Directory Information of the workstation you configured.

The following example of a system configuration with DLUS-Served LU Registration was taken with the options -D and -SYS of the display command (see Figure 220 on page 239).

- -D displays the Directory Information
- -SYS displays the System Default Information

In the Directory Information of this example, note the LU entry type of the LU name. You will find some LU entry types with Register (DLUR entry). These are the dependent LUs of the DLUR, registered at the network node server.

The System Default Information shows the local node capabilities. Note the new capability DLUR registration support. This entry shows you if the local node supports DLUR registration or is configured to support DLUR registration.

```
************************************
*       Communications Server      *
*       DISPLAY Sample Program     *
************************************

************************************
*       Directory Information       *
************************************
Total directory entries                    4
Network node entries                       1

1>Network node CP name                     USIBMRA.WTRO5137
  Number of associated LUs                 4

1.1>LU name                                USIBMRA.WTRO5137
    Owning CP name                         USIBMRA.WTRO5137
    LU entry type                          Home

1.2>LU name                                USIBMRA.WTRO1533
    Owning CP name                         USIBMRA.WTRO1533
    LU entry type                          Register

1.3>LU name                                USIBMRA.WEEE0302
    Owning CP name                         USIBMRA.WTRO1533
    LU entry type                          Register (DLUR entry)
1.4>LU name                                USIBMRA.WEEE0303
    Owning CP name                         USIBMRA.WTRO1533
    LU entry type                          Register (DLUR entry)
Local and adjacent node entries            0
************************************
*       Communications Server      *
*       DISPLAY Sample Program     *
************************************

************************************
*     System Default Information     *
************************************
Default mode name
Default local LU name                      WTRO5137
Implicit partner LU support                Yes
Maximum held alerts                        10
Conversation security required             No
Maximum logical record send size           32767
Default TP directory                       *
Default TP operation                       Non-queued attach manager started
Default TP program type                    Background
Implicit link HPR support                  Yes
Retry count                                6
Alive timer                                60
Path switch timer (Low)                    480
Path switch timer (Medium)                 240
Path switch timer (High)                   120
Default routing preference                 Native first
DLUR registration support                  Yes
```

Figure 220. Example of Display Output

10.7 Trace

Taking a trace when running Communications Manager with a configuration containing DLUS-Served LU Registration provides more information of how registration is negotiated and performed.

The following are trace examples showing an XID exchange (see Figure 221 on page 241) and an FMH-5 Attach command (see Figure 222 on page 242) in which registration is negotiated:

- The XID portion of the trace shows that negotiation takes place between the host system and the network node server (Destination address=40005200513704).

  Note that DLUS-Served LU Registration is negotiated during the XID exchange using the term DLUS-Served LU Registration = Supported.

```
DLC type: IBMTRNET
Adapter number: 00
Destination address: 40005200513704
ALS ID: 8C01177108B307A4
XID
|| (0x0000) Format = 3
||          Node type = 2
|| (0x0001) Total length = 101
|| (0x0002) Node ID = 0x05D05137
|| (0x0008) Init-self = Cannot receive
||          Stand-alone BIND = Can receive
||          BIND segment generation = Can generate
||          BIND segment receipt = Can receive
||          FID type = FID 2
|| (0x0009) ACTPU suppression = ACTPU not requested
||          APPN network node = Yes
||          CP-CP sessions requested = Yes
||          CP-CP sessions supported = Yes
||          Exchange state = Pre-negotiation
||          Secondary initiated non-activation exchange = Supported
||          CP name change = Supported
|| (0x000A) Adaptive BIND pacing sender = Supported
||          Adaptive BIND pacing receiver = Supported
||          Quiesce TG requested = No
||          PU Capabilities control vector = Not supported
||          Sender is an APPN peripheral border node = No
||          Adaptive BIND pacing = Supported for all LUs, unless overridden
|| (0x000F) Parallel TGs = Supported
||          DLUR XID sender prefers ACTPU over CP-SVR pipe = No
||          DLUS-Served LU Registration = Supported
|| (0x0010) TG number = 0
|| (0x0011) DLC type = SDLC
|| (0x0012) DLC data length = 11
|| (0x0013) ABM = Supported
||          Link station role = Negotiable
||          Short hold mode status = Not reconnection
||          Short hold mode capability = Not supported
||          Transmit/receive capability = Primary
|| (0x0014) ABM nonactivation XID exchange initiator = No
|| (0x0015) Maximum receive BTU length = 2224
|| (0x0017) SDLC CR profile = SNA link profile
|| (0x0018) SIM/RIM options = Not supported
|| (0x001B) Maximum I-frames before ACK = 4
|| (0x001D) Network name control vector:
||           (0x0002) Network name type = CP
||           (0x0003) Name = USIBMRA.WTR05137
|| (0x0030) Network name control vector:
||           (0x0002) Network name type = LS
||           (0x0003) Name = @AAAAAAA
|| (0x003B) Product set ID control vector:
||             Hex dump:
||                10280011 11040E02 F5F6F2F1 F2F5F4F0   <.(.............> <........56212540>
||                F0F2F2F0 16110313 0011F9F5 F4F5D3F9   <................> <0220......9545L9>
||                C5F2F3E6 C2F3F2F4 4040                <........@@      > <E23WB324        >
```

Figure 221. Trace Example, XID

- The FMH-5 Attach is issued by the host system and addressed to the DLUR (EN WTR05133). Note the entry LUs are DLUS-served in the Register portion of the attach command (Command parameters control vector).

```
DLC type: IBMTRNET
Adapter number: 00
Destination address: 40005200513704
ALS ID: 8C01177108B307A4
Transmission priority: Network
||TH: FID2, OIS, LFSID=0x10102, SNF=0x0003
||RH: RQ, FMD, FI, OIC, RQD3, PI, BB, CEBI
||FMH-5
||   Command code = Attach
||   User ID already verified = No
||   Password is substituted = No
||   PIP present = No
||   Conversation type = Basic
||   Synchronization level = Confirm
||   Transaction program name = ?002 (APPN Resource registration)
||   Logical unit of work identifier:
||      LU name = USIBMRA.WTR01533
||      Instance number = 0xAD962528D494
||      Sequence number = 0x0001
||   Conversation correlator = 0xE8200F713AB307A4
||Register
||   Command parameters control vector:
||    (0x0002) Request/reply = Request
||             Central resource registration requested = No
||             Entry type = Register
||             LUs are DLUS-served = Yes
||   Associated resource entry control vector:
||    (0x0002) Resource type = ENCP
||    (0x0004) Resource name = USIBMRA.WTR01533
||   Directory entry correlator control vector:
||    (0x0002) Correlator = 0x00000001
||   Directory entry control vector:
||    (0x0002) Resource type = LU
||    (0x0004) Resource name = USIBMRA.WEEE0302
```

Figure 222. Trace Example, FMH-5 Attach

Part 7. Data Link Control

Chapter 11. Frame Relay Support

IBM Communications Server Release 4.1 has implemented frame relay data link control (DLC) to support SNA and TCP/IP protocols. In previous releases, the RXR/2 product was required in order to connect to a frame relay network. In this chapter, we review the newly integrated frame relay implementation, which includes BAN (border access node) and BNN (border network node) support.

Note

IBM Communications Server Release 4.1 has integrated frame relay support as a DLC only. Other frame relay functions such as source route bridging (SRB) are not supported.

11.1 Overview

In this section, we present a frame relay overview in order to review some of the main features of this network. If you require more information about frame relay, please see *IBM Communications Server Release 4.1 Frame Relay*, GC31-8319-01.

11.1.1 Frame Relay Overview

The frame relay (FR) protocol is a method of transmitting internetworking packets by combining the packet switching and port sharing of X.25 with the high-speed and low delay of time division multiplexing (TDM) circuit switching. Frame relay allows you to connect multiple LANs to a single high-speed (1.54 Mbps) WAN link with multiple point-to-point permanent virtual circuits (PVCs).

Frame relay is a layer-2 packet-switching protocol that provides a more efficient end-to-end transmission mechanism than X.25 and other upper-layer protocols. It accomplishes this by eliminating almost all internodal routing processing such as congestion control and error correction mechanisms. As a result, this lightweight protocol is able to take advantage of the highly reliable, high-speed circuits that are currently available. This enables user devices to better realize the potential speeds of the transmission interface.

This efficiency requires the following:

- End devices must be intelligent and there must be a high-level protocol that guarantees the integrity of the communications between them.
- The layer one, physical and electrical connections, must have a very low error rate. In the case of an error, frame relay is able to identify it, but it does not make retransmissions of the frames in error. When this occurs, frames are discarded expecting that the end devices will detect the missing frames in order to retransmit.

Frame relay offers the following features:

1. Fast and simple packet switching

   Frame relay is based upon fast packet switching (FPS) technology in which a continuous data stream is segmented into packets and transported in a packet switching network. FPS is different from traditional packet switching (such as X.25) in that the packet switches only perform low-level functions

such as routing, congestion management, and CRC checking. Higher-level functions such as flow control, error correction, and acknowledgments are performed by the terminal equipment. Because of the lack of ″hop-by-hop″ error correction, frame relay is designed to carry data over good quality, high-speed lines.

2. Intended for high bandwidth and low network delay

   The reduced complexity of the packet switches results in low network delays and high bandwidth. This is the benefit of frame relay and FPS, in general, when compared to traditional packet switching techniques such as X.25.

3. Layer 2 multiplexing and single-port access to network

   As the network is responsible for routing packets to multiple destinations, the frame relay user needs only a single physical port to connect to the network. A layer 2 identifier is used to specify a unique logical connection over a port. This single logical connection can be used to transport multiple protocols to a single destination. Also, multiple logical connections to different destinations can be made over a single physical connection.

4. Bandwidth on demand

   As frame relay networks are based upon packet switching, they allocate only the required amount of bandwidth necessary for a transmission.

5. High throughput and low delay

   Utilizing the core aspects (error detection, addressing, and synchronization) of the link access protocol - D-Channel (LAPD) datalink protocol, frame relay eliminates all network layer (Layer 3) processing. By using only the core aspects, frame relay reduces the delay of processing each frame.

6. Congestion detection

   Upon receiving Backward Explicit Congestion Notification (BECN), the router initiates a controlled slowdown of traffic, thereby avoiding a complete frame relay network shutdown.

7. Circuit access and control

   As the router dynamically learns about the availability of non-configured circuits (orphan circuits), you can control access to those new circuits.

8. Network management option

   As your network requires, the frame relay protocol can operate with or without a local network management interface.

9. Multiplexing protocols

   Using one PVC to pass multiple protocols, frame relay provides no error correction or retransmission function. To provide error free end-to-end transmission of data, frame relay relies on the intelligence of the host devices.

   The virtual circuit between a pair of frame relay network users called frame relay terminal equipment (FRTE) is set up at subscription time and is called a permanent virtual circuit (PVC).

   Frame relay network users employ a set of operational procedures called local management interfaces (LMIs) to perform physical link integrity verification and obtain status of the logical connections on the physical interface. By using LMI, a user can detect frame relay PVC outages and

their recovery. The frame relay user-network interface, which includes definition of LMI, is defined by:

CCITT Q.933 Annex A

ANSI T1.617 Annex D

11.1.2 Frame Relay Network

The frame relay network consists of the frame relay backbone (consisting of frame relay switches provided by the frame relay carrier) providing the frame relay service. The router functions as the frame relay connection device. The router encapsulates frame relay frames and routes them through the network based on a Data Link Connection Identifier (DLCI). The DLCI is the medium access control (MAC) address that identifies the PVC between the router and the frame relay destination device. For example, in Figure 223 a packet destined to go from router B to router D would have a DLCI of 19 to reach router D; however, a packet destined to go from router D to router B would have a DLCI of 16.

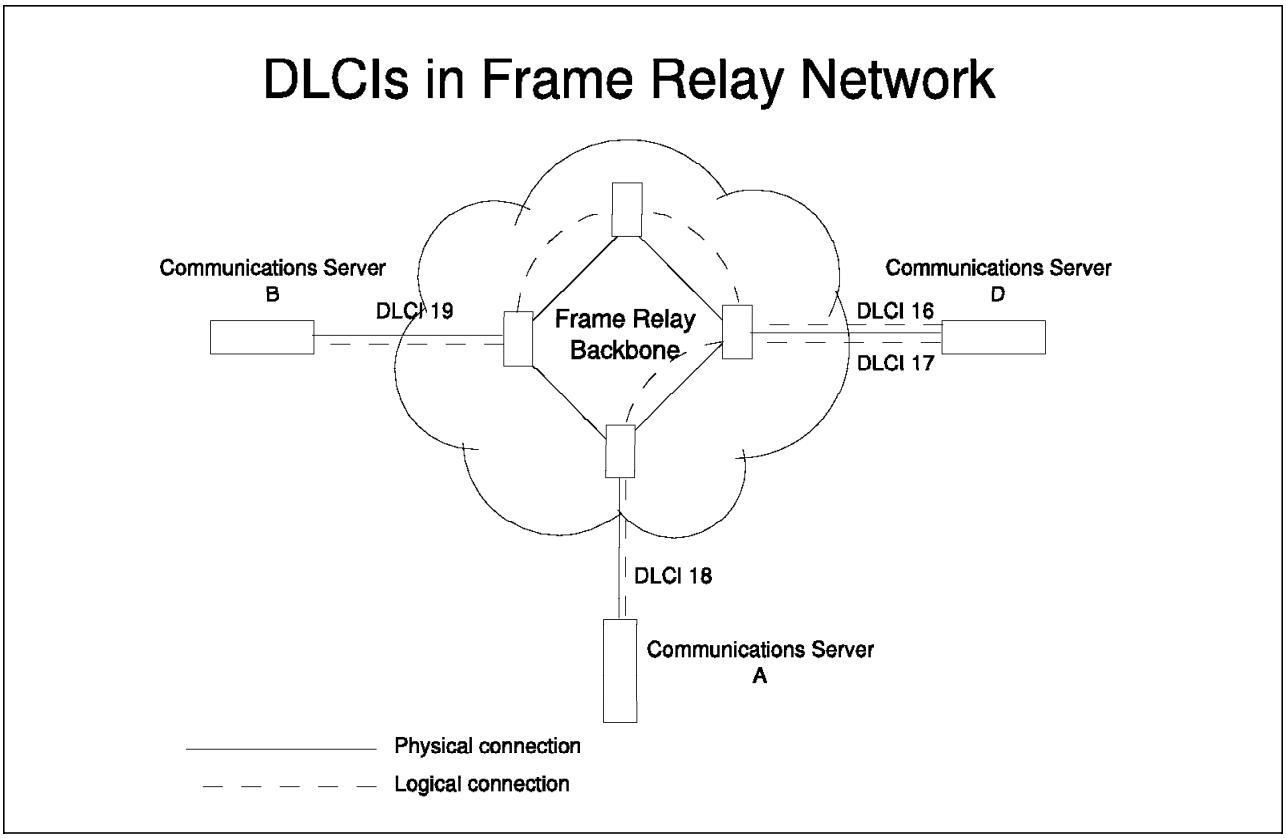


Figure 223. DLCI in Frame Relay Network

The data link connection (DLC) is a type of virtual circuit that is the main part of the frame relay network.

The majority of providers use permanent virtual circuits (PVCs) predefined at both sides of the connection.

A DLC is determined by a number called a Data Link Connection Identifier (DLCI), which has only local significance. This means that a DLCI can be different at the local and remote ends of a frame relay connection, so you need

to know both. The carrier is the one that takes care of the routing tables in each switch that routes the frames to the correct destination.

There are three types of logical circuits:

1. Permanent virtual circuits (PVCs) are used primarily.
2. Switched virtual circuits (SVCs) are not commonly available.
3. Multicast is proprietary.

PVCs are permanently set up by the carrier between two points. They have an assigned bandwidth that can also be used by other PVCs that use the same physical line.

A frame relay network has the following characteristics:

1. Transports frames transparently. The network can modify only the DLCI, congestion bits, and frame check sequence. High-level data link control (HDLC) flags and zero bit insertion provide frame delimiting, alignment, and transparency.
2. Detects transmission, format, and operational errors (frames with an unknown DLCI).
3. Preserves the ordering of frame transfer on individual PVCs.
4. Does not acknowledge or retransmit frames.

11.1.3 Frame Relay Frame

A frame relay frame consists of a fixed size address field with variable sized encapsulated user data. The following figure illustrates a frame relay frame format.

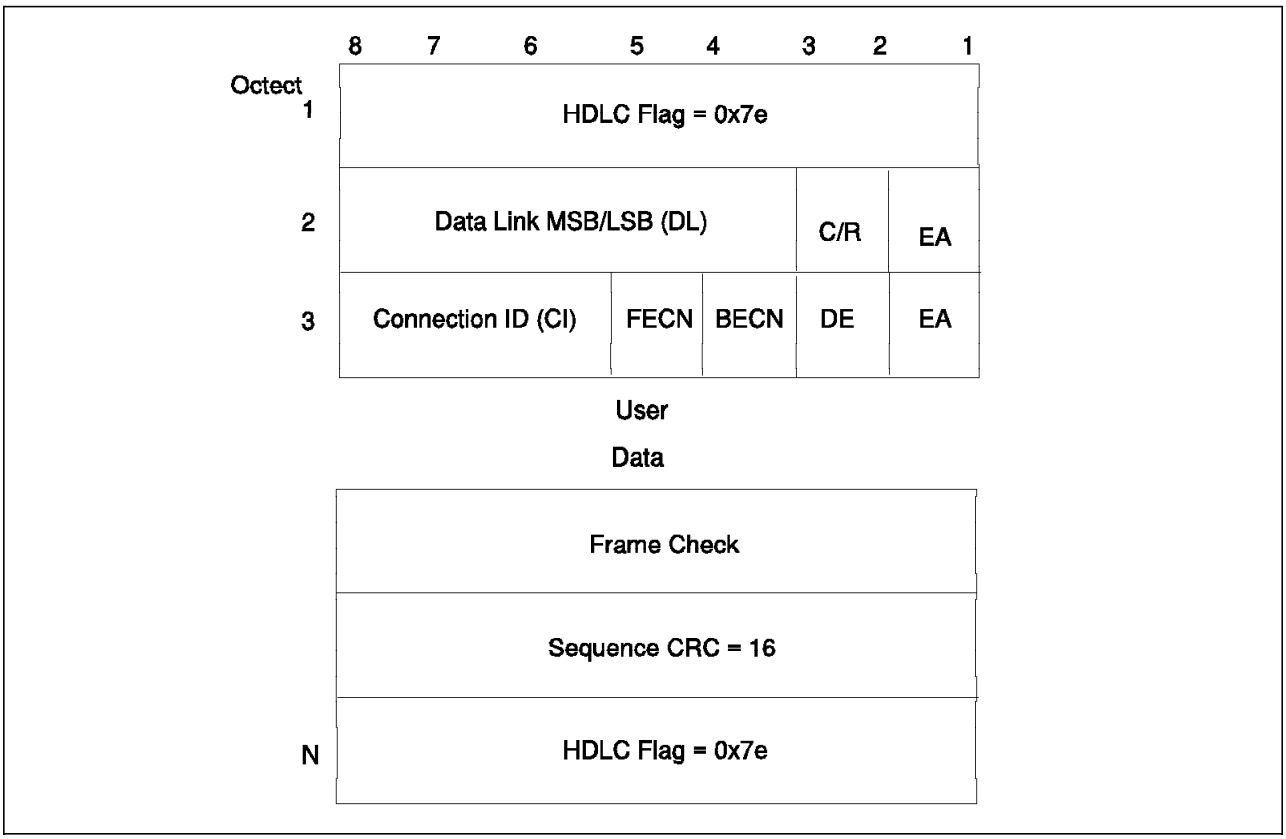


Figure 224. Frame Relay Frame

11.1.3.1 HDLC Flags

Located in the first and last octet, these flags indicate the beginning and end of the frame.

11.1.3.2 Data Link Connection Identifier (DLCI)

This 10-bit routing ID resides in bits 3-8 of octet two and bits 5-8 of octet three.

The DLCI is the MAC address of the circuit. The DLCI allows the user and network management to identify the frame as being from a particular PVC. The DLCI enables multiplexing of several PVCs over one physical link.

11.1.3.3 Command/Response (C/R)

This field's use is not defined within the frame relay standards and the field is passed transparently across the network.

11.1.3.4 Extended Address

This version of frame relay does not support extended addressing.

11.1.3.5 Forward Explicit Address Notification (FECN)

The frame relay backbone network sets this bit to 1 to notify the user receiving the frame that congestion is occurring for the PVC in the direction the frame is being sent.

11.1.3.6 Backward Explicit Congestion Notification (BECN)

The frame relay backbone network sets this bit to 1 to notify the user that congestion is occurring when the user sends a frame for this PVC. The router then initiates a throttle down to a rate equal to or less than the user-defined CIR. The CIR for a PVC is supplied by the frame relay service provider and is configured using the add permanent-virtual-circuit command.

11.1.3.7 Discard Eligibility (DE)

The network may discard transmitted data exceeding the CIR on a PVC. The DE bit is set by the end station to indicate discard eligibility. This version of frame relay does not set the DE bit, but does log the exception if the DE bit is not set. If the DE bit is set, the exception is not logged.

11.1.3.8 User Data

This field contains the protocol packet being transmitted. This field can contain a maximum of 8188 octets; however, the frame check sequence (FCS) can effectively detect errors only on a maximum of 4096 octets of data. The protocol data is preceded by a frame relay encapsulation header as defined in RFC 1490.

11.1.3.9 Frame Check Sequence

This field is the standard 16-bit cyclic redundancy check (CRC) that HDLC and LAPD frames use. This field detects errors occurring in the bits of the frame between the opening flag and FCS.

11.1.4 Local Management Interface

The LMI protocol is a set of operational procedures employed for controlling the connection between the user and the network.

11.1.4.1 LMI Responsibilities

LMI has the following responsibilities:

- Ensuring that the link between the user and the network is active
- Notifying the addition and deletion of PVCs
- Delivering status messages regarding the availability of the circuits

Communications Server supports two LMI standards:

- ANSI T1.617 Annex D
- Frame Relay Forum′s LMI Revision 1 (LmiRev1)

Each standard supported uses a special DLCI to identify its LMI functions. ANSI T1.617 Annex D uses DLCI 0, and LmiRev1 uses DLCI 1023.

Communications Server frame relay does not support the ITU-T Q.933 (international) standards. However, ITU-T and ANSI are closely aligned with the notable exception being the LMI frame formats.

11.1.4.2 LMI Signaling Mechanisms

With LMI we have three signalling mechanisms:

1. Unidirectional (required)

   Includes link integrity verification, PVC status notification, and full status polling.

2. Bidirectional

   Allows both sides of a physical link to support the user side and network side LMI procedures concurrently.

3. Asynchronous Update

   Allows the network side to send unsolicited PVC status using the Status message with a special report type. This is done because using either unidirectional or bidirectional signalling can cause significant delay in informing that changes have occurred in PVC status.

11.1.4.3 LMI Features

With LMI we have the following features:

1. Link Integrity Verification (LIV)

   LIV procedures require the user side of a physical link to exchange sequence numbers periodically on a defined polling interval so each side can determine whether the physical connection to the adjacent node is still functional.

2. Notification of PVC status
3. Full Status Polling
4. Unsolicited Status Messages

11.1.5 SNA over Frame Relay

Communications Server supports the SNA extensions to RFC 1490 that define routed format frames used for SNA traffic. ACF/NCP and 3174 support these extensions. If the remote end does not support the extensions, Communications Server can use bridged format frames to communicate.

The Frame Relay Forum document FRF.3 describes how SNA can be carried directly over frame relay.

Frame relay does not participate in windowing or retransmissions, so for SNA Communications Server use Logical Link Layer Type II (LLC2 IEEE 802.2) protocol over frame relay.

The Communications Server frame relay support was extracted from the RouteXpander/2 product. Communications Server's frame relay consists of a network driver and a protocol module. This allows the association of 802.2 or IP protocols with the frame relay WAN access for 802.5. This software lets the workstation communicate through the frame relay WAN access as though it were a token-ring LAN adapter device driver. Software that ordinarily communicates over token-ring LANs can communicate through the WAN access for 802.5 without any knowledge of the network it is traversing. This allows LAN-based protocols and the programs that they support to run without being changed. The protocols are bound to the WAN access for 802.5 and the network handles the communication.

Note

IBM Communications Server Release 4.1 supports SNA over frame relay and IP over frame relay.

RFC 1490 defines the encapsulation method for carrying multiprotocol traffic over a frame relay network.

Protocols encapsulate their packets within a Q.922 Annex A frame. In comparison, X.25 uses the link access procedure - balanced (LAP-B) protocol while frame relay uses LAP-D, ITU's Q.922 protocol.

The frames contain information necessary to identify the protocol carried within the protocol data unit (PDU), thus allowing the receiver to properly process incoming packets.

Figure 225 shows the Q.922 Annex A frame format.

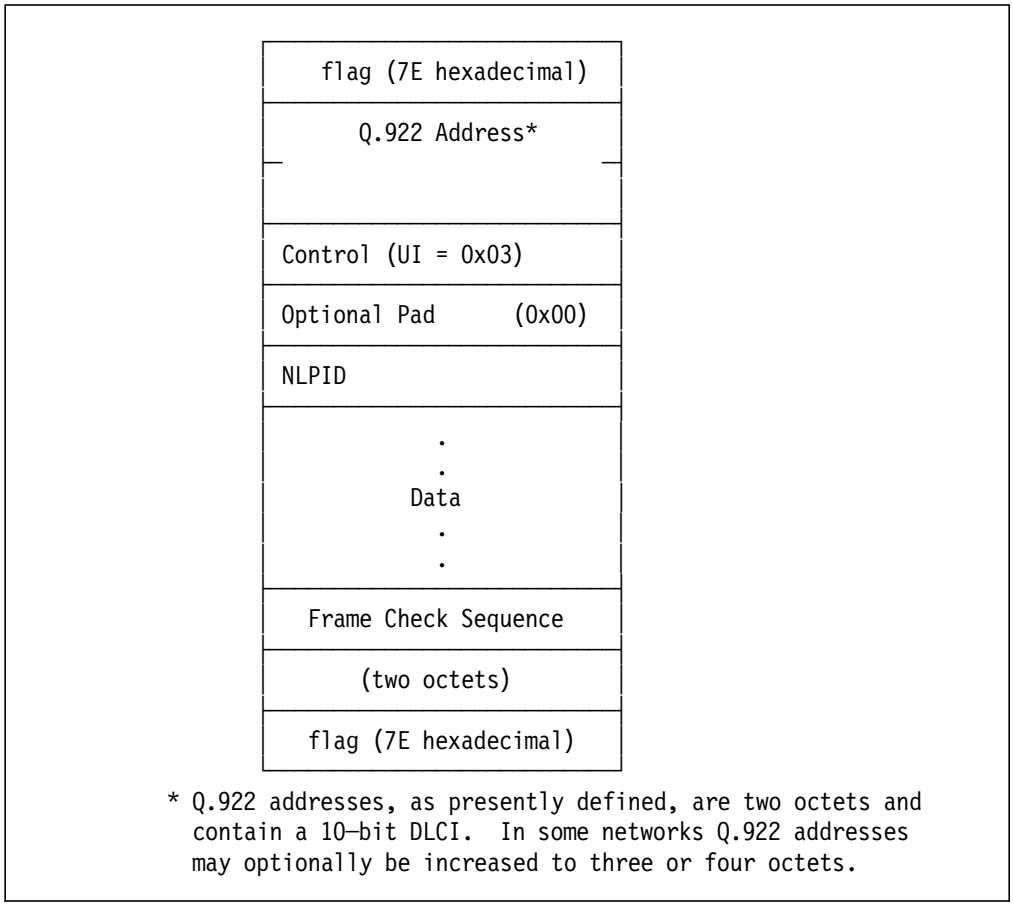


Figure 225. Q.922 Annex A Frame Format

Q.222 addresses are two octects and contain a 10-bit DLCI.

The control field is the Q.922 control field. The UI (0x03) value is used unless it is negotiated otherwise.

Pad field is an optional field used to align the remainder of the frame to a convenient boundary for the sender.

Network Level Protocol ID (NLPID) contains the values for many different protocols. This field tells the receiver what encapsulation or protocol follows. The most commonly used NLPID values are:

- 0x80 IEEE subnetwork access protocol (SNAP)

- 0x81 ISO CLNP
- 0x82 ISO ESIS
- 0x83 ISO ISIS
- 0xCC IP
- 0xCE EtherType

For a more detailed explanation of the protocol format, refer to Appendix G, “RFC 1490 Extract” on page 319 where there is an extract of RFC 1490.

There are two formats used to transport LLC2 over frame relay:

- Routed frame format
- Bridged frame format

Both encapsulate LLC2 frames directly into frame relay, but the bridged frame format also uses a MAC address.

The bridged frame format is less restrictive and easier to configure but has more overhead.

We could use either frame format; it depends on the capabilities of the partner node that we want to connect.

You must configure the destination MAC address when defining an SNA connection.

The specified MAC address can be either real or virtual:

- A real address results in bridged format frames. For more information, see 11.1.5.1, “Bridged Format with Real MAC Addresses.”
- A virtual address results in routed format frames. For more information, see 11.1.5.2, “Routed Format with Virtual MAC Addresses” on page 255.

Note

1. Frame relay peripheral node functions introduced with NCP V7R1 only support routed format frames. If Communications Server is connected to an NCP as a peripheral node, you must configure Communications Server to use virtual MAC addresses. If the boundary access node (BAN) function has been added to the NCP, bridged format can also be used.
2. Connections to an IBM AS/400 can be established using routed format frames.
3. Network routers do not typically support routed format frames. Configure Communications Server to use real MAC addresses when it is connected to a network router for routing SNA traffic.

11.1.5.1 Bridged Format with Real MAC Addresses

Real MAC addresses are either addresses of stations physically attached to a token-ring or Ethernet or MAC addresses assigned to the frame relay interface of a frame relay attached station. These addresses can be used as the LAN destination address when defining the links between two Communications Servers. Figure 226 on page 254 shows you an example.

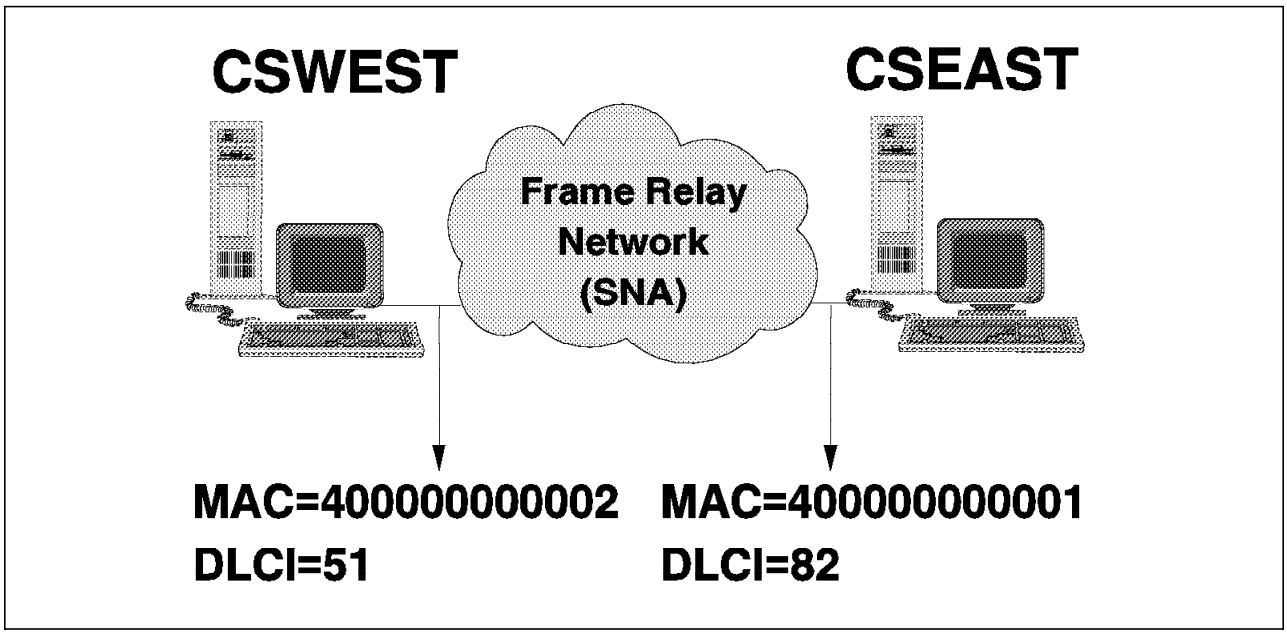


Figure 226. Example Using MAC Addresses in a Frame Relay Network

- CSWEST and CSEAST are connected to each other via a frame relay network. The network has assigned DLCI number 51 to CSWEST for its virtual circuit to CSEAST, and DLCI number 82 to CSEAST for the same virtual circuit.
- The real MAC address for the frame relay interface of CSEAST is 400000000001. For CSEAST, this MAC address is defined when configuring the Communications Server WAN access for the 802.5 network adapter. Similarly, the real MAC address for CSWEST is 400000000002.
- In CSWEST, a logical link is configured to access CSEAST. In that link definition, the destination address is 400000000001. In CSEAST, a logical link for CSWEST is defined with a destination address of 400000000002.

Bridged format is always used if the destination MAC address is a real MAC address. The bridged format frame contains an entire 802.5 frame.

When establishing the connection, CSWEST uses test frame broadcasts to locate the remote station, specifying the desired destination MAC address. A response should be received from only one other device because the MAC address must be unique. The DLCI associated with the response is then used for the traffic on the connection.

The following figure shows a scheme of the bridged frame format.

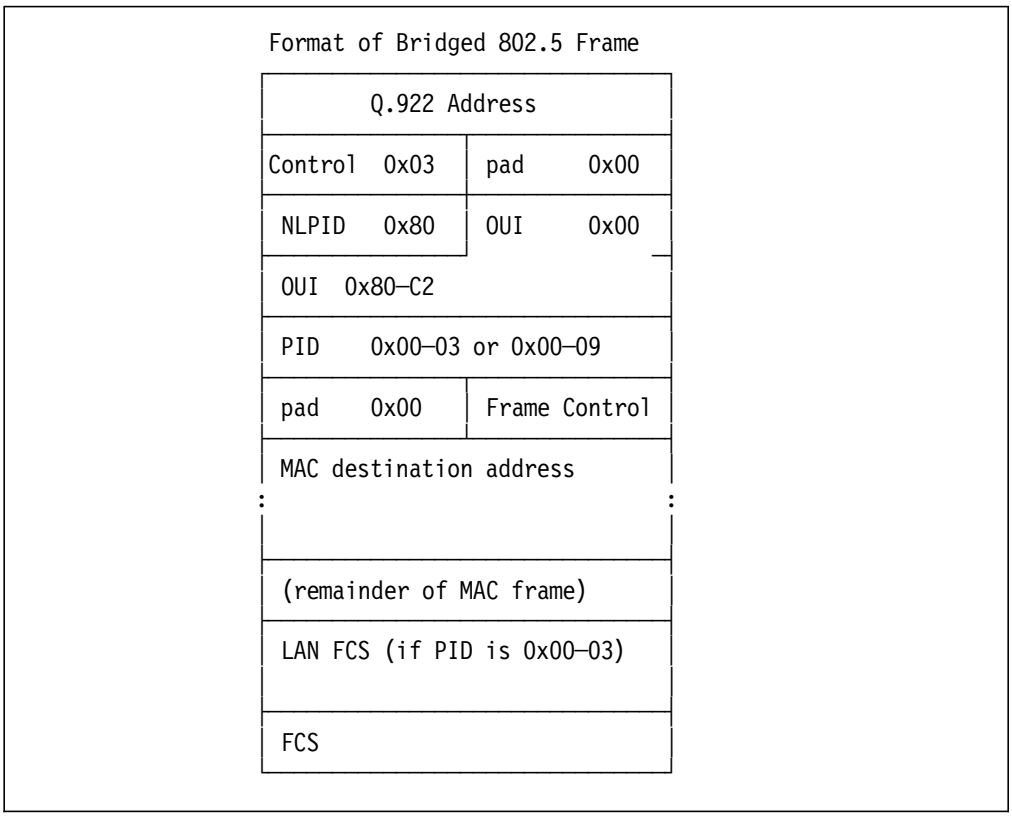


Figure 227. Bridged Frame Format Scheme

11.1.5.2 Routed Format with Virtual MAC Addresses

The virtual MAC addresses assigned by a local node to identify a remote station at the other end of a virtual circuit have the format of mmmmmmFFnnnn, where:

- Digits 1-6 (mmmmmm) are the virtual MAC address mask.

  The virtual MAC address mask is a parameter that is defined when configuring frame relay on the Communications Server WAN access for 802.5 network adapter. The default value for this parameter is 4000FF.

- Digits 7 and 8 are always FF.
- Digits 9-12 are the DLCI numbers assigned at the local interface for the virtual circuit to the remote device.

To use routed SNA frames in the configuration shown in Figure 226 on page 254:

- Configure a logical link in CSWEST to access CSEAST over frame relay. The destination address is a virtual MAC address of 4000FFFF0051.
- Configure a logical link in CSEAST to access CSWEST over frame relay. The destination address is a virtual MAC address of 4000FFFF0082.

The Communications Servers send messages as if they are sending them on a token-ring LAN. These frames are passed to the frame relay device driver, which converts them to RFC 1490 format using routed frame format. The virtual MAC address is not contained in the resulting frame.

Routed format is always used if the destination MAC address is a virtual MAC address. Routed format frames are sent on the DLCI identified with the nnnn portion of the virtual MAC address. To send SNA routed frames the remote

node at the other end of the frame relay network must support the SNA extensions to RFC 1490.

If two Communications Servers are communicating with each other over a point-to-point connection (DLCI number 32) and using the default virtual MAC address mask (4000FF), the virtual MAC address for both Communications Servers is 4000FFFF0032. Use 4000FFFF0032 as the destination MAC address when defining the link between the Communications Servers.

The following figure shows a scheme of the routed frame format.

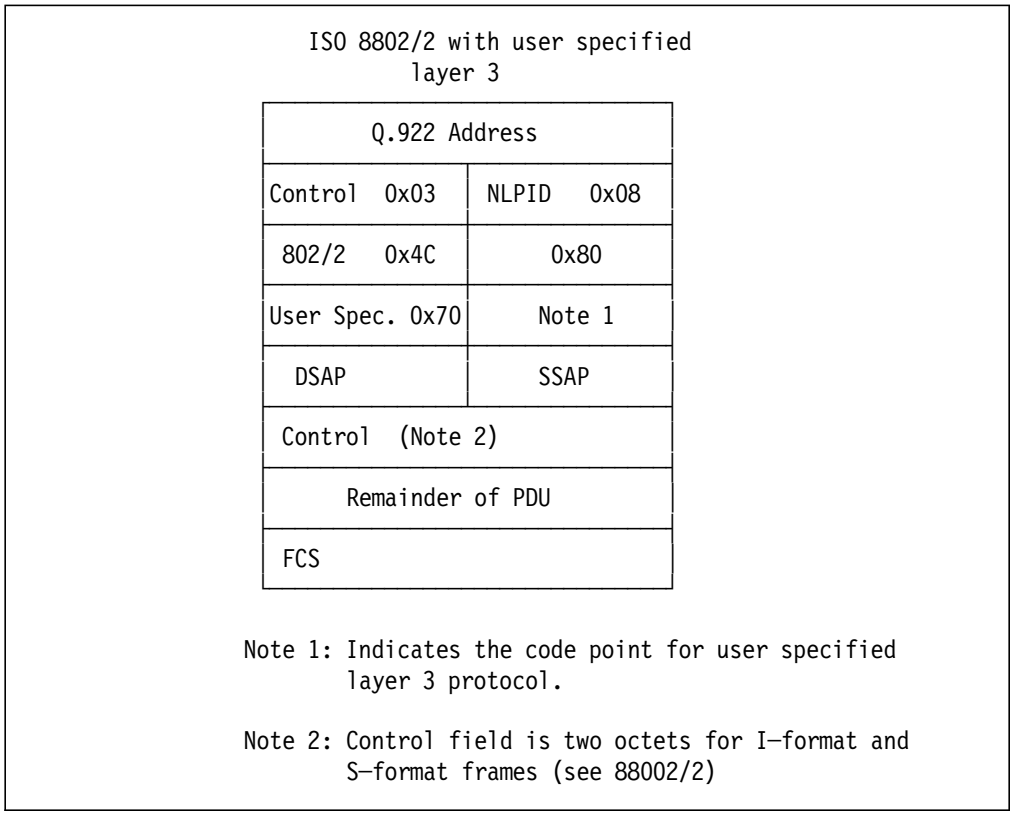


Figure 228. Routed Frame Format Scheme

Note

We always refer to the bridged or routed frame format. A bridged frame format does not mean that we are bridging all the protocol frames from one end to the other side of the communication link. It only means that we are using a certain type of standard frame format.

There is another way to send SNA over frame relay. This method is commonly used in the routers world but not implemented in Communications Server. It is called data link switching (DLSW - RFC 1795) and was developed by IBM. It provides SNA *encapsulation* in IP at level 3, making local acknowledgment of polls. However, it adds latency due to TCP/IP processing and requires extra overhead. It puts extra processing overhead in the routers and also requires an LLC or SDLC session between the end stations and the DLSW node. This could limit the scalability.

11.1.6 TCP/IP over Frame Relay

Frame relay in Communications Server can also be used to transport TCP/IP traffic. This support uses RFC 1490 in frame relay. For more details about the protocol, refer to Appendix G, “RFC 1490 Extract” on page 319.

In this case, TCP/IP uses the routed frame format. The ARP requests are broadcasting in order to learn the correct DLCI. Then the flows are broadcast as if this was a token-ring LAN.

You could use a Communications Server machine as an IP router between a LAN network and a frame relay network. In this case you only have to define the TCP/IP support, enable the routing function (ipgate on), and, if you want, start the routed program in order to propagate the routing table in both networks.

If you also have defined Sockets over SNA Gateway you could start the gated program making a check in the RIP box of the backup and load balancing. If you are running in a Sockets over SNA parallel gateway, you already have the gated program started, so you must not start any other routed program.

11.2 Configuration

In order to explain the frame relay support feature configuration in IBM Communications Server Release 4.1, we set up a scenario that consists of two workstations running IBM Communications Server Release 4.1 connected through a frame relay network. Each computer has a wide area connector (WAC) adapter and is connected to a modem eliminator (V.35). See Figure 229.

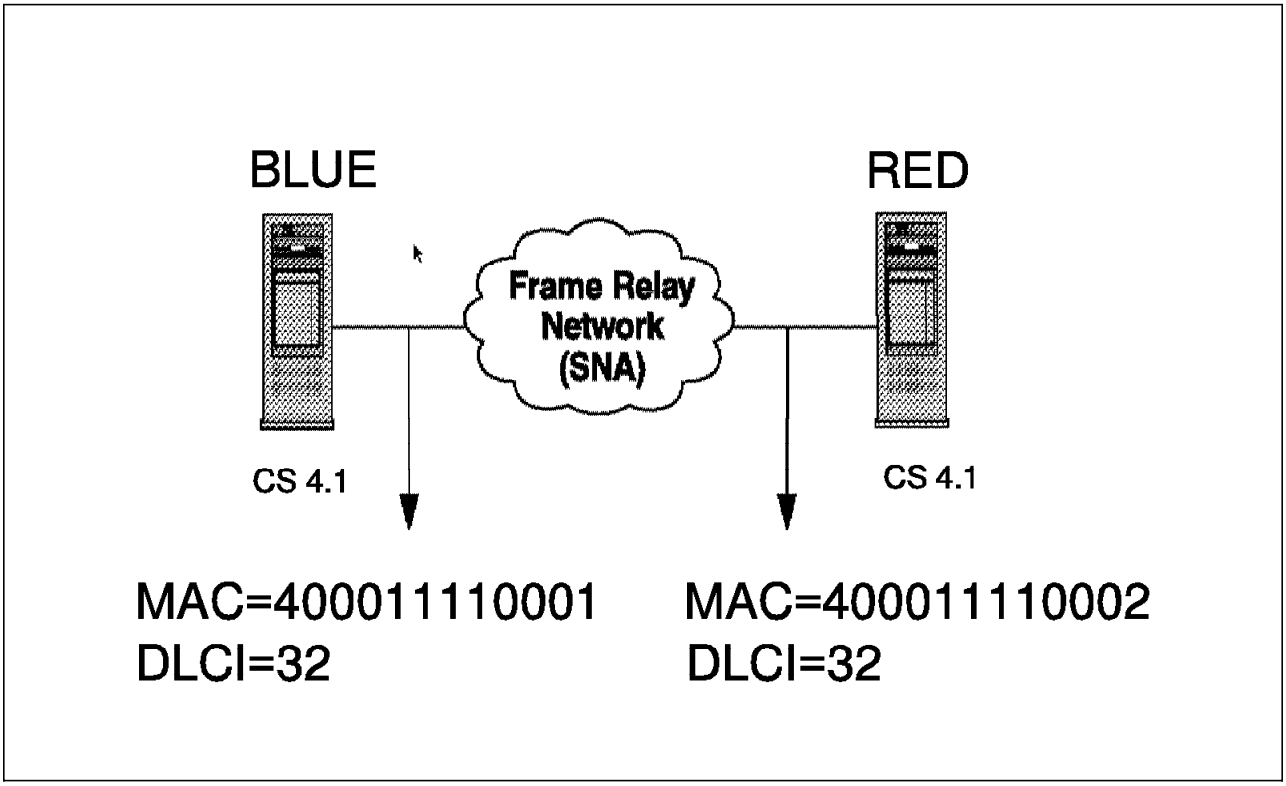


Figure 229. Frame Relay Scenario

The recommended sequence of frame relay configuration tasks is:

- MPTS:
  1. Configuring the WAN adapter for frame relay
  2. Binding frame relay WAN access for 802.5 protocol
  3. Configuring frame relay WAN access for 802.5 network adapter
  4. Binding frame relay WAN access for 802.5 protocol
     - Configuring IBM IEEE 802.2
     - Configuring TCP/IP
- CMSETUP:
  1. Configuring Communications Server's frame relay SNA support
     - Token-ring DLC
     - Any explicit SNA logical link

11.2.1 Configuring MPTS

Each of the components mentioned above can be configured using MPTS. Frame relay WAN access for 802.5 support consists of a network driver and a protocol module.This allows you to associate 802.2 or TCP/IP protocols with the frame relay WAN access for 802.5. This software lets the workstation communicate through it as though it were a token-ring (also known as 802.5) LAN adapter device driver. Software that ordinarily communicates over token-ring LANs can communicate through the WAN access for 802.5 to the Communications Server's frame relay, then through the associated WAN adapter, and on to destinations across the WANs. All of this can be done without any knowledge of the network being traversed. This makes a vast number of connections possible and allows LAN-based routing protocols and the programs they support to run without being changed. Bind the protocols to the WAN access for 802.5 and the network handles the communications. Figure 230 on page 259 is used throughout this document when referring to specific components of frame relay.

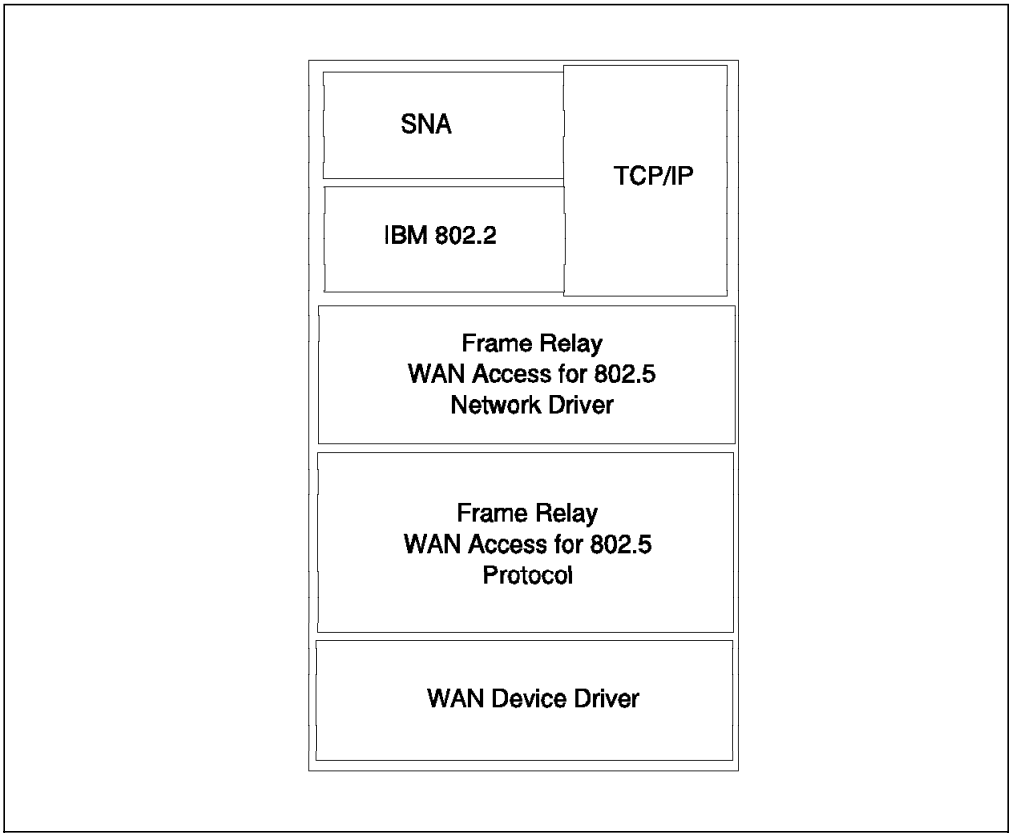


Figure 230. Frame Relay Components in IBM Communications Server Release 4.1

11.2.1.1 Configuring the WAN Adapter for Frame Relay

To configure the MPTS correctly, you must use the MPTS version packaged in the IBM Communications Server Release 4.1 compact disk or a later version.

First run the MPTS configuration program, and then select the **Configure Option**. Select the **LAN Adapter and protocols** option in the configuration panel.

Select **Additional Network Drivers** from the Installation menu. Select **Continue...** and press Enter.

Modify the screen to match your CD-ROM drive letter and the path to the OS/2 MAC driver. In our case this is E:\DRIVERS\IBMWAC.

Select **OK** and press Enter. MPTS will now copy the OS/2 device driver onto your hard drive:

E:\DRIVERS\IBMWAC\IBMWAC.OS2 into C:\IBMCOM\MACS directory

E:\DRIVERS\IBMWAC\IBMWAC.NIF into C:\IBMCOM\MACS directory

E:\DRIVERS\IBMWAC\WAC.MSG into C:\IBMCOM directory

E:\DRIVERS\IBMWAC\WACH.MSG into C:\IBMCOM directory

After the device driver files have been successfully selected, choose the **Frame Relay WAN Access for 802.5 Protocol** for the IBM Wide Area Connector and **IBM 802.2** for the Frame Relay WAN Access for 802.5 Network Adapter.

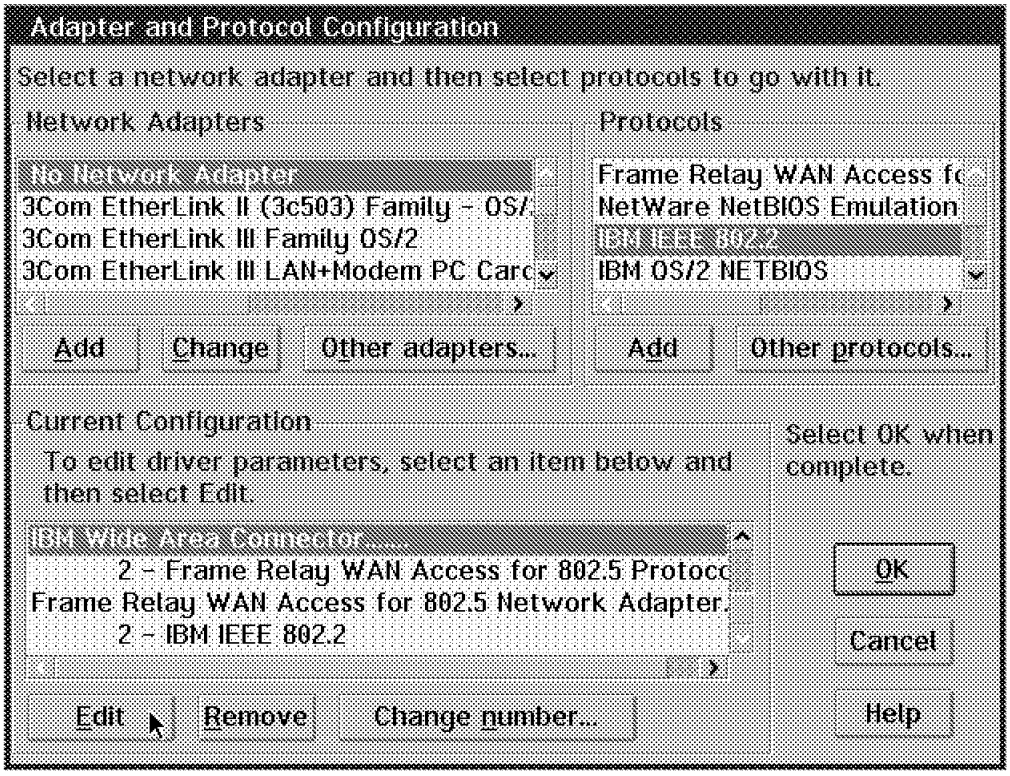


Figure 231. MPTS Adapter and Protocol Configuration

Select **IBM Wide Area Connector...** from the Current Configuration list and click on **Edit** to supply the IBM WAC communication parameters.

Enter the Slot or Card number to match the WAC Adapter installation in your system. For MicroChannel machines, this corresponds to the physical card slot number.

Attention

With ISA bus machines, this is a LOGICAL number that corresponds with the adapter's switch settings and has no relation to the physical card location. For ISA systems, refer to your adapter documentation to obtain the card number.

Set the Port number to that which corresponds to the link over which communications will occur (that is, 0 = Link A; 1 = Link B). Physically, the 0 refers to the daughter board, which is located farthest from the bus tab pins on the adapter.

Select a memory window (RAMADDRESS) for the adapter. MicroChannel machine users may ignore this step, since this is automatically resolved using the .ADF file.

For ISA bus machines, the default value of the 16-KB shared memory address is D0000. This may or may not be acceptable depending on the specific ISA system and the other adapters that may be installed.

In many cases you will be able to accept the remaining default values. However, your application, specific system or network configuration may require other values to be altered. Online help is available for the remaining default values.

APPLICATION NOTES

If your application supports switched DCEs (via V.25bis dialing), you should set CONNTYPE = 1. See Figure 232 for the Link connection type parameter.

Applications that use port connection manager (PCM) to manage the ports should also set PCMSUPPORT = 1 on the MPTS configuration screen. See Figure 232 for the ANDIS PCM support parameter.

When using the WAC in a multi-port arrangement, such as attaching to a port sharing device, NRZI must be configured the same for all. All devices on a given line must have NRZ set identically in order to communicate. See Figure 232 for the NRZI option.

Select **OK** and press Enter to save these parameters.

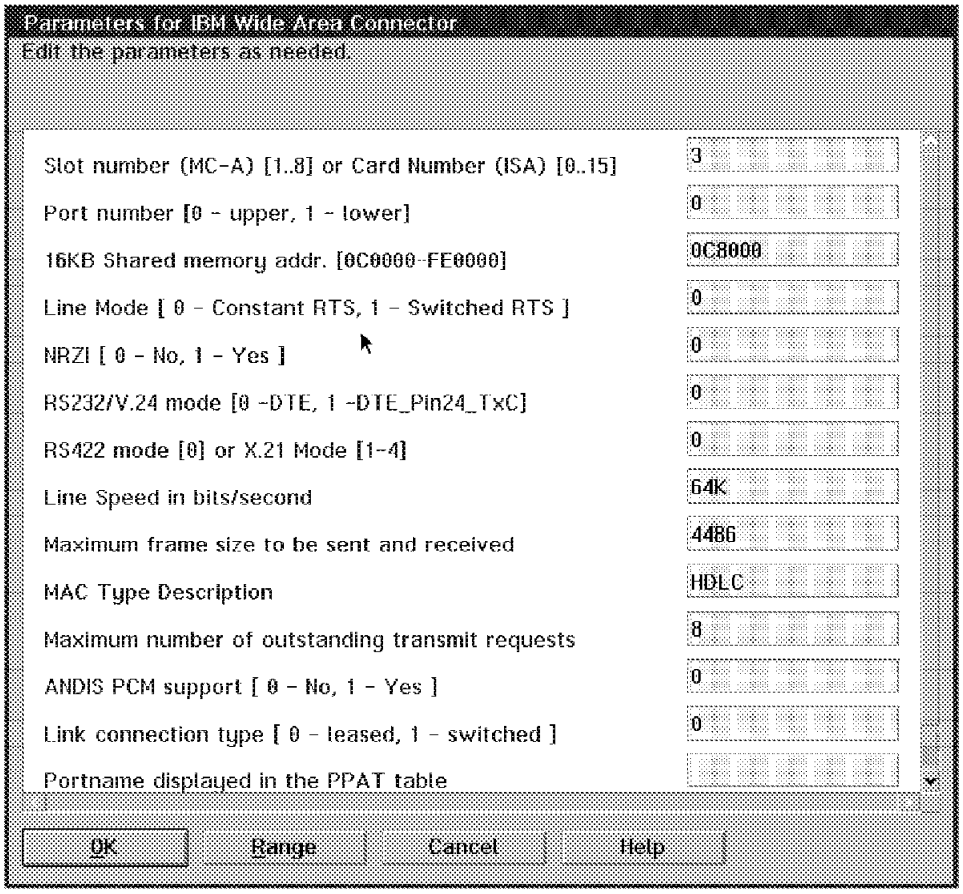


Figure 232. Wide Area Connector Parameters

11.2.1.2 Binding Frame Relay WAN Access for 802.5 Protocol

Binding indicates which network adapters and protocols are supposed to interact. Binding allows network adapters and protocols to exchange data and control information.

This capability to mix-and-match network adapters and protocols is feasible, because all of these modules implement a standardized interface known as the network driver interface specification (NDIS).

1. Scroll through the Protocols list box and select **Frame Relay WAN Access for 802.5 Protocol**. Add the Frame Relay WAN Access for 802.5 Protocol from the Protocols list box to the Current Configuration list box by either double-clicking on it or by clicking on the **Add** button. This copies the Frame Relay WAN Access for 802.5 Protocol into the Current Configuration list box. It is indented under the IBM WAN to show that they are bound together.
2. In the Current Configuration list box, select the **Edit** button to display the edit window for the Frame Relay WAN Access for 802.5 Protocol.
3. When you are satisfied with all of the values on the edit window, click on **OK**. Continue on to the next section.

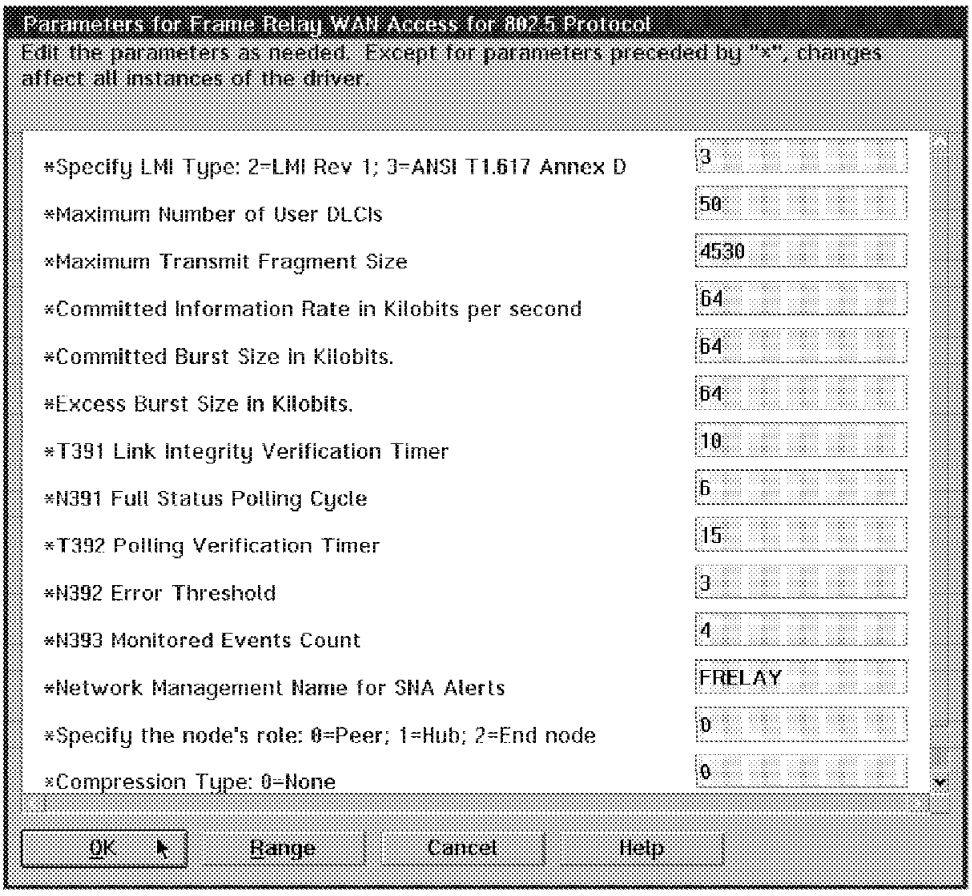


Figure 233. Frame Relay WAN Access for 802.5 Protocol

11.2.1.3 Configuring Frame Relay WAN Access for 802.5 Network Adapter

1. Scroll through the Network Adapters list box to the Frame Relay WAN Access for 802.5 Network Adapters, then select it. Add the Frame Relay WAN Access for 802.5 Network Adapter from the Network Adapters list box to the Current Configuration list box by either double-clicking on it or by clicking on the **Add** button. This copies the IBM Communications Server Release 4.1 WAN Access for 802.5 Network Adapter into the Current Configuration list box.
2. Edit the following required parameters for the Communications Server′s Frame Relay WAN Access for 802.5 Network Adapter:
   - **Locally Administered MAC Address:**

     The locally administered MAC address should be a network-wide, unique, locally administered MAC address. It can be used by upper layer protocols to identify this Communications Server′s frame relay node to other nodes (for instance, in the Source Address field of frames transmitted by this Communications Server). The locally administered MAC address should not include the virtual MAC address mask.
   - **Virtual MAC Address Mask:**

     The virtual MAC address mask is only needed when SNA or APPN traffic needs to be sent in RFC-1490 or RFC-1356 routed format (for instance, to a 3174 but not necessarily to a 6611). The virtual MAC address mask can be used to construct a DLCI-specific artificial address that is to be stored in Communications Server′s APPN or SNA Destination Address field. For additional information refer to *Routed Format with Virtual MAC Addresses*.

     The default ring number specifies the ring number of the frame relay network when viewed as a logical LAN. Communications Server′s Frame Relay views the entire frame relay network as one ring and each DLCI as a station on the ring. Ring numbers used in this frame relay network should be different than ring numbers used on LANs and other networks that communicate with this frame relay network.
3. When you are satisfied with all the values on the edit window, click on **OK**. You are returned to the Configure Workstation window.

Remember the numbers of the local MAC address and the virtual MAC mask; both will be used in the Communications Server setup process.

In the two machines there is only one difference in the MPTS configuration. The difference is the locally administrated MAC address. The machine on the left has 400011110001 as its MAC address. The machine on the right has 400011110002 as its MAC address.

The ring number is the number of the ring that is represented by the frame relay network when viewed as a logical LAN. The ring numbers used in this frame relay network should be distinct from all ring numbers used on LANs and other networks reachable from this one.

This ring number should be the same in both machines that want to communicate.

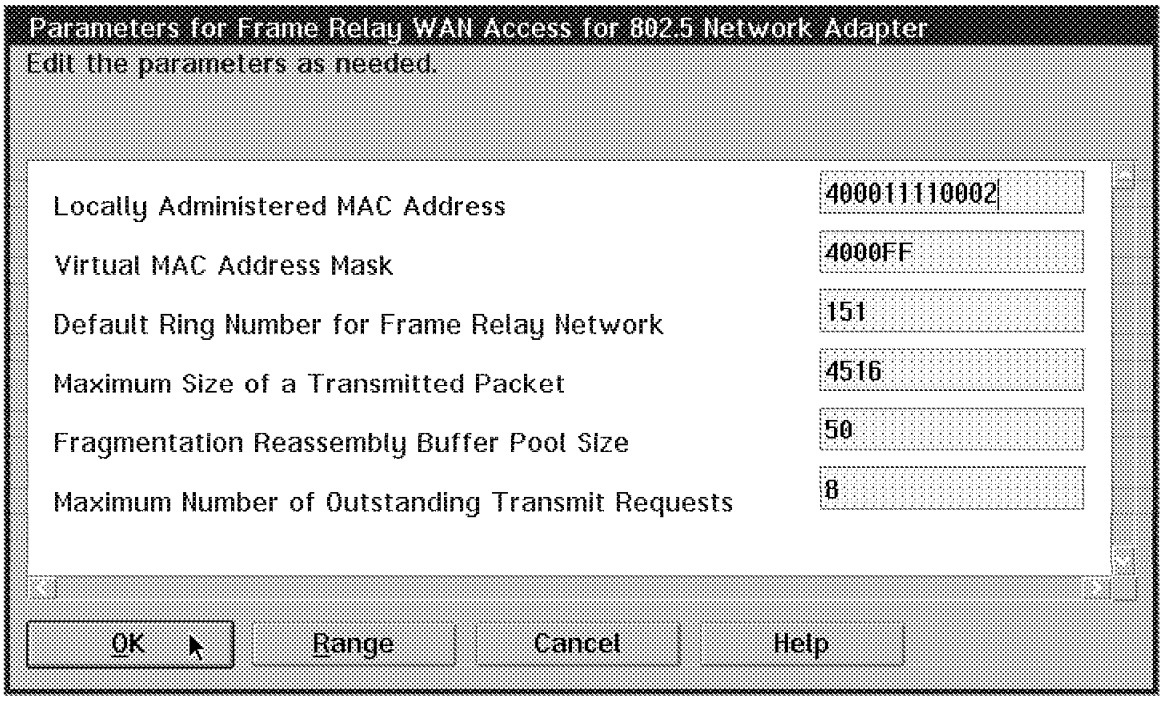


Figure 234. Parameters for Frame Relay WAN Access for 802.5 Network Adapter

11.2.1.4 Configuring IBM IEEE 802.2

Frames are passed to and from the SNA/APPN router using the IBM IEEE 802.2 Protocol.

From the LAPS Configure Workstation window:

1. Bind the IBM IEEE 802.2 Protocol to the Frame Relay WAN Access for 802.5 Network Adapter.
2. Notice the number that LAPS inserted next to IBM IEEE 802.2 in the Current Configuration list box. This is the logical adapter number. Compare the IBM IEEE 802.2 Protocol's logical adapter number with the logical adapter that LAPS assigned to the Frame Relay WAN Access for 802.5 Protocol (under either the WAC or ARTIC Network Adapter). Set these logical adapter numbers to the same value:
   - With the IBM IEEE 802.2 Protocol selected in the Current Configuration list box, click on the **Change Number** button.
   - When the Change Logical Adapter Number window appears, scroll to the logical adapter number assigned to the Frame Relay WAN Access for 802.5 Protocol. Click on it, and then click on the **Change** button.

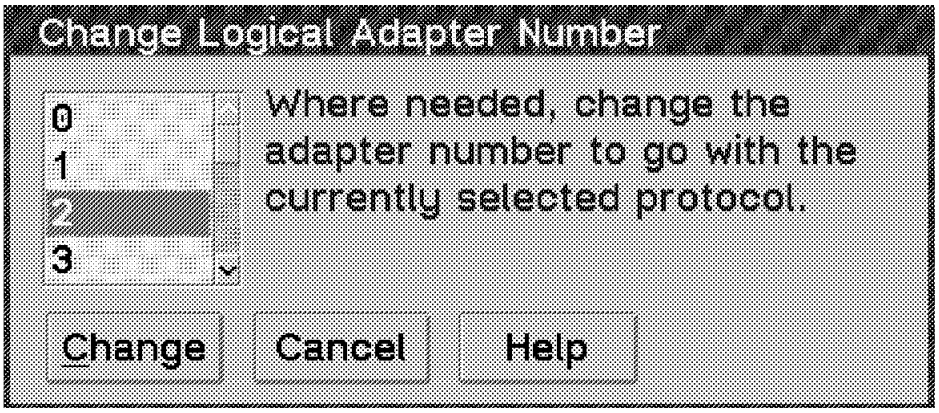


Figure 235. Change the Number of Adapter

Then you could edit the 802.2 parameters to make any changes that you need. Remember that you need one link station for each machine that you want to have a link.

For example, you may want to connect your machine using a frame relay network to ten other machines directly. This means you must use a DLCI for every machine, each having a connection definition. Thus, you need to have ten link stations in the Maximum Link Stations parameter.

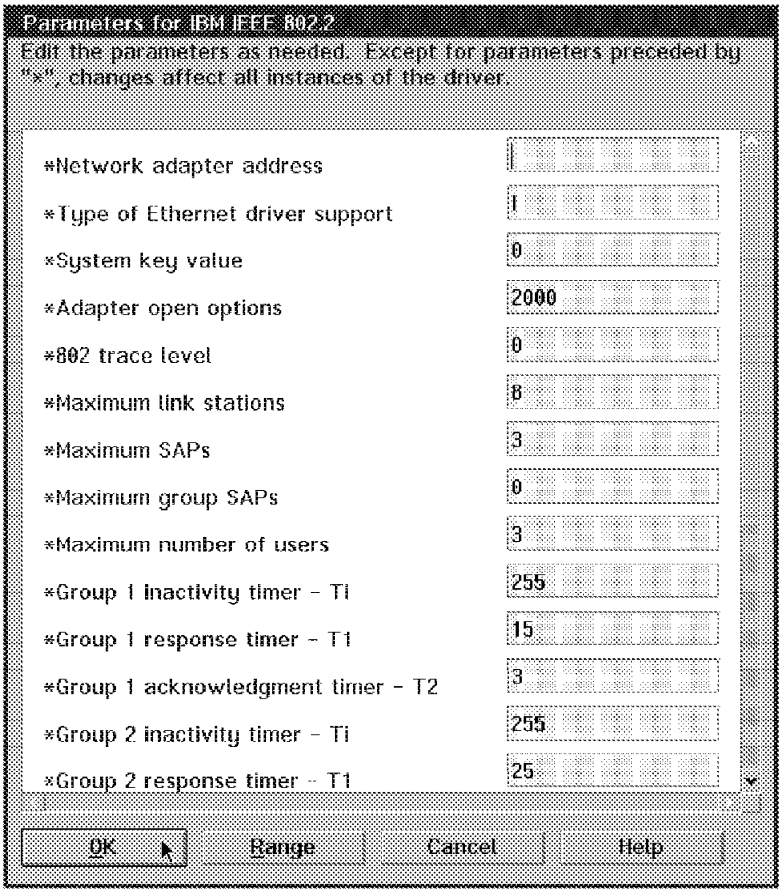


Figure 236. Parameters for IBM IEEE 802.2

When you have finished this procedure, the Current Configuration list box will look like Figure 231 on page 260.

11.2.2 Configuring IBM Communications Server Release 4.1

Configure your DLC adapter parameters as if this network was a token-ring network.

Remember to match the adapter number here with the adapter number that you use in the MPTS configuration panels for the WAN adapter for frame relay. In this case we are using the WAN adapter as adapter number 2.

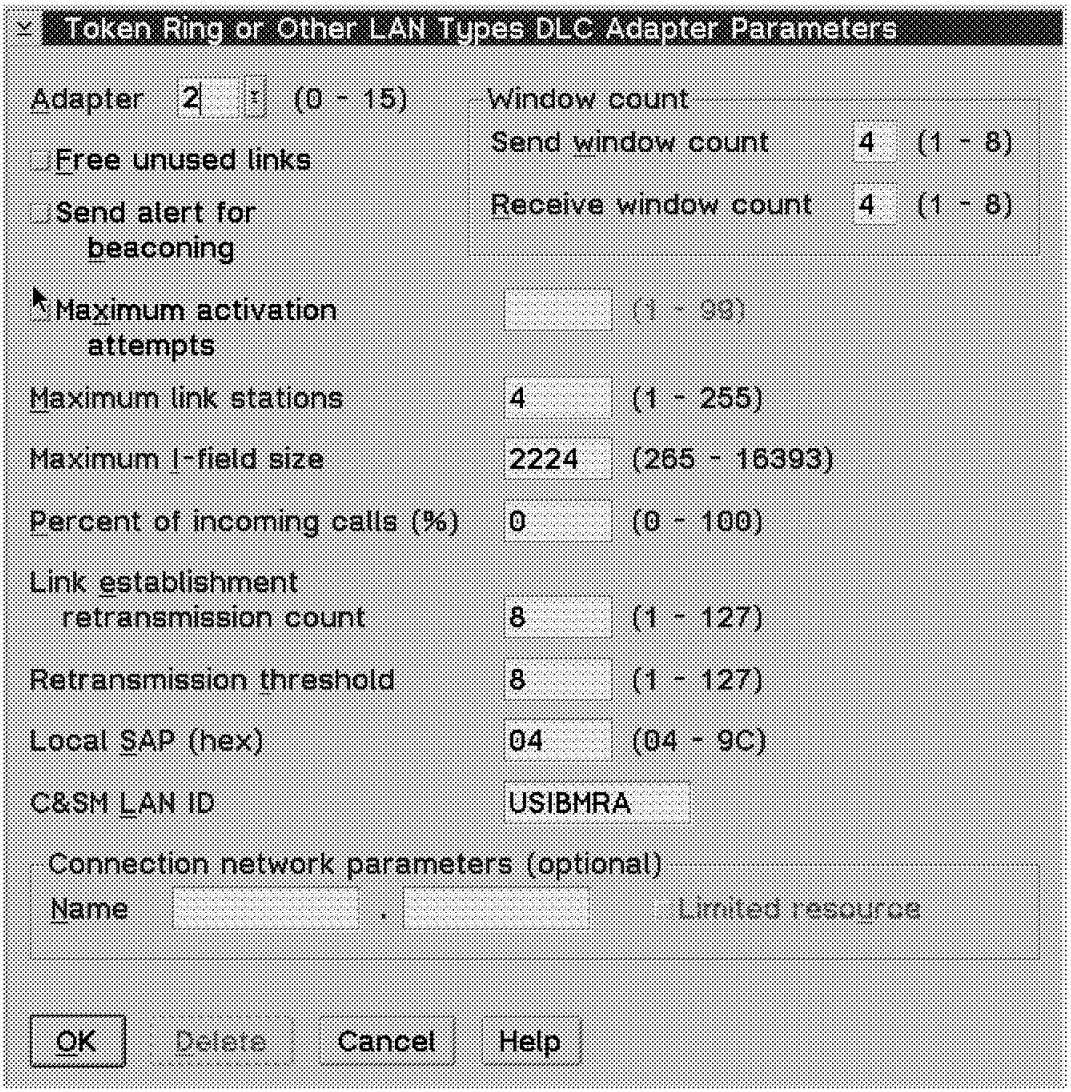


Figure 237. DLC Configuration in IBM Communications Server Release 4.1

Then configure the SNA characteristics in the way you normally do (no additional considerations).

In the connections list panel be sure to select the **Token-Ring or other LAN Types** adapter in the frame relay link.

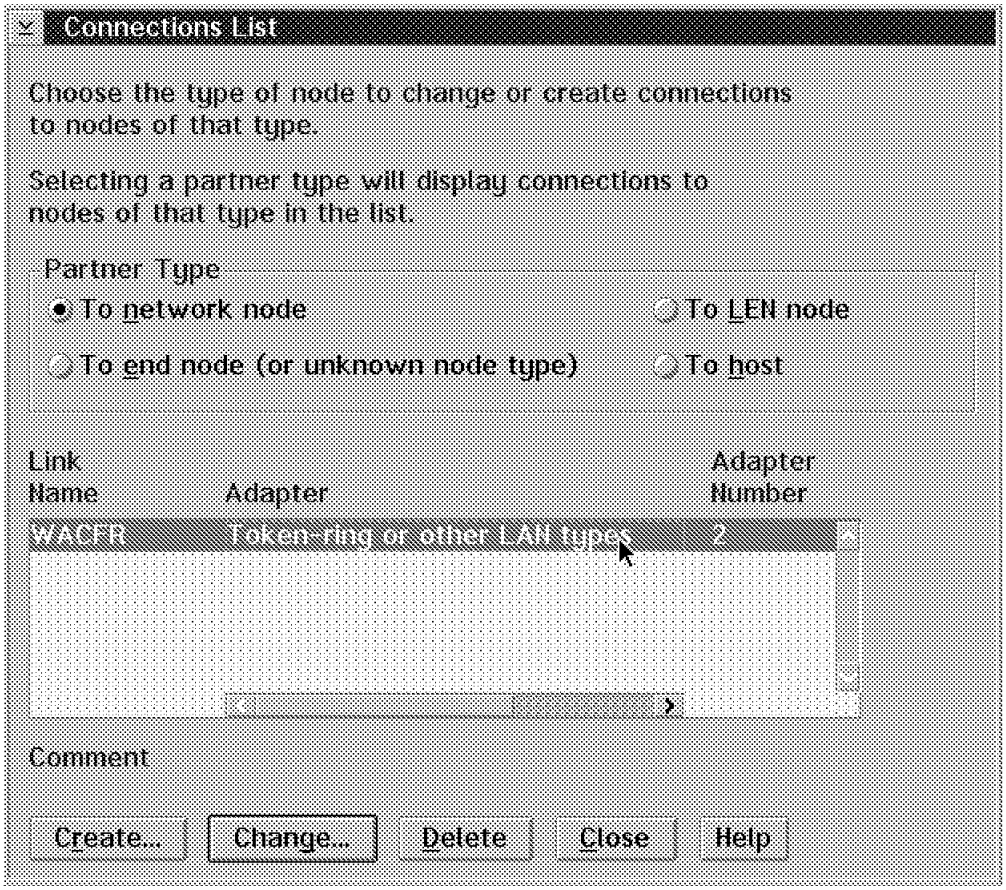


Figure 238. Connection Configuration in IBM Communications Server Release 4.1

The adapter number used in this connection is 2; remember the WAC adapter in the MPTS configuration. See Figure 231 on page 260.

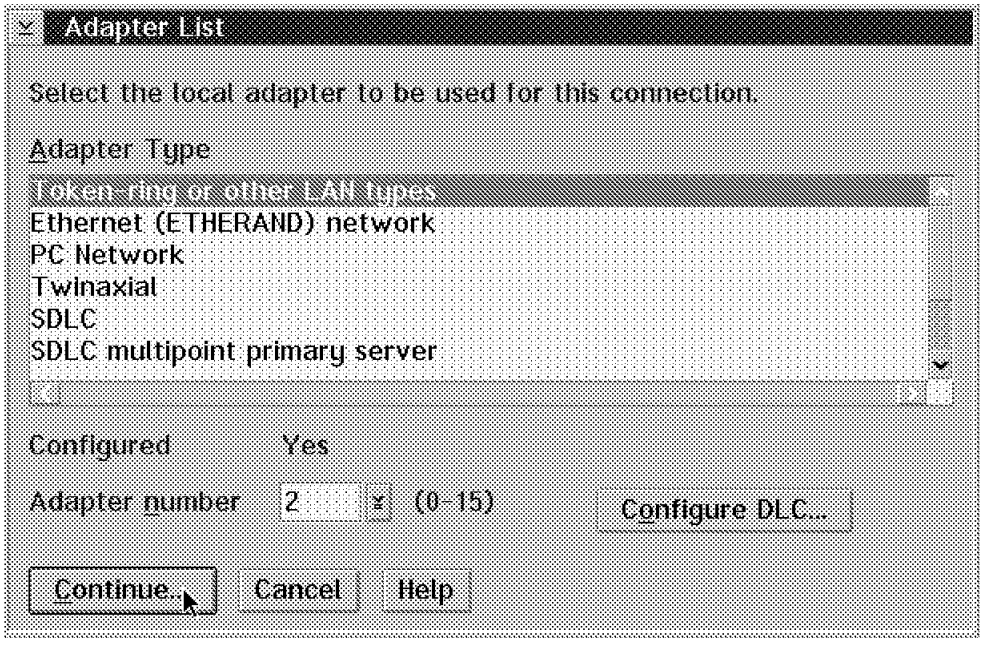


Figure 239. Adapter List

Be sure to specify the correct LAN destination address and address format for the network node in the frame relay link.

As in the MPTS configuration there are no differences between both machines up to this point. Now each machine has to point to the other.

In this example we are using (at the blue machine) the real address of the partner (red machine) adapter: 400011110002. Thus, we are using the bridged frame format. By default, in a point-to-point connection, Communications Server uses the DLCI number 32. In a real frame relay network connection, Communications Server learns the DLCI numbers and finds the correct DLCI to use from the TEST response command from the partner machine.

The red machine must have a connection with a LAN destination address of 400011110001.

In the same way we could use the routed frame format and (at the blue machine) define the LAN destination address as 4000FFFF0032 in order to use the DLCI number 32. Also in this case, we are using a direct connection, and so we chose the default DLCI. In a real frame relay network connection you must get the DLCI numbers from your frame relay network provider.

Using routed frame format, the blue machine also has a LAN destination address of 4000FFFF0032, because it is also using the DLCI number 32 (the default DLCI).

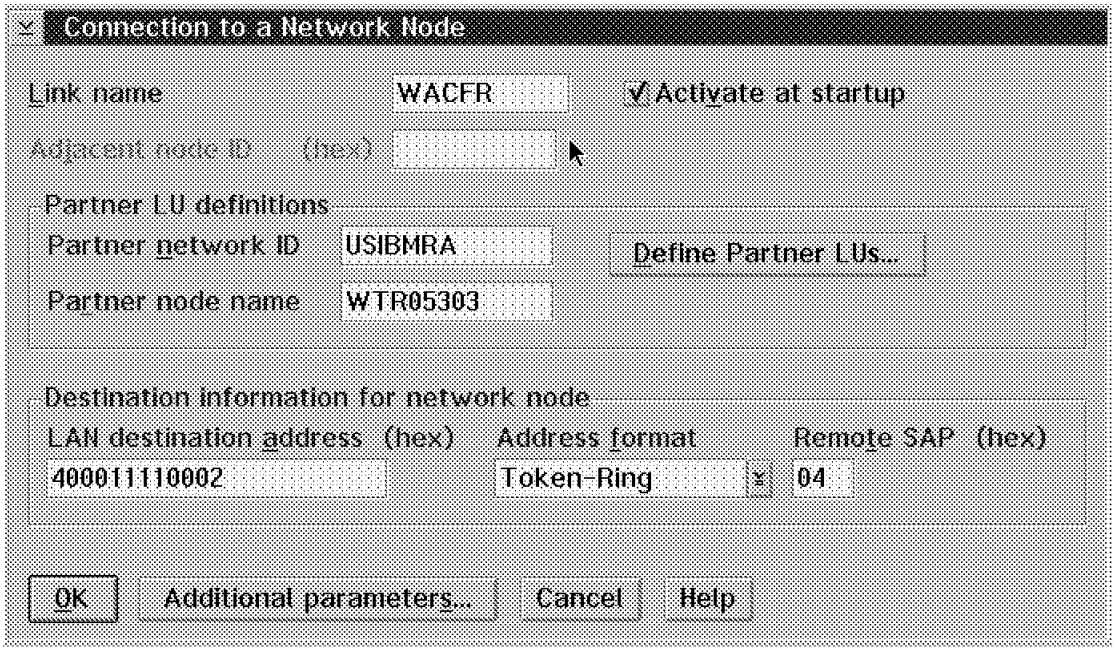


Figure 240. Connection Panel

11.2.3 Configuring TCP/IP over Frame Relay

In IBM Communications Server Release 4.1, you can also configure TCP/IP over frame relay. If this is required, you only need to add the TCP/IP protocol support to the frame relay adapter. Then, you will configure the TCP/IP product or MPTS as if the frame relay adapter is just another LAN adapter.

11.2.3.1 Configuring TCP/IP Protocol Support

Frames are passed to and from the TCP/IP router using the TCP/IP protocol.

From the LAPS Configure Workstation window:

1. Bind the TCP/IP Protocol to the Frame Relay WAN Access for 802.5 Network Adapter.
2. As in the 802.2 configuration, change the logical adapter number to match the logical adapter number of the Frame Relay WAN Access for 802.5 Protocol (under either the WAC or ARTIC Network Adapter).

11.2.3.2 Configuring TCP/IP Product or MPTS TCP/IP

For TCP/IP you need two things: the TCP/IP stack (or protocol support) and the TCP/IP itself.

The protocol support is always defined using the MPTS panel as in the previous step. This actualizes the \IBMCOM\PROTOCOL.INI file.

But for the TCP/IP itself you could have two alternatives: *Do* you have the TCP/IP product installed or *not*.

If you have the TCP/IP product, configure it using the TCP/IP configuration panels that could be accessed using the TCPCFG command. If you do not have the TCP/IP product, you should define TCP/IP using the MPTS panels.

In any case both methods update a file named SETUP.CMD that resides in the ..\BIN subdirectory where the SET ETC environment variable points to and is called by the TCPSTART.CMD. In the case of the TCP/IP product, it is always called by the MPTSTART.CMD at boot time.

11.3 LANTRAN.LOG

The LANTRAN.LOG file should look like Figure 241 on page 270 if all of the drivers were loaded correctly.

```
IBM OS/2 LANMSGDD [07/31/96] 5.05 is loaded and operational.
IBM OS/2 LANDD [07/31/96] 5.00.03
IBM OS/2 LANDLLDD 2.01
IBM OS/2 LANDLLDD is loaded and operational.
Card: 4  Port: 0   Hardware Interface: V35
I/O Port: 0400   Irq: 5   Shared Ram: 000DC000
Rx Clock: External          Tx Clock: External
IBMWAC 2.01.00 successfully loaded.
IBM OS/2 WAN Access for 802.5 V2.00 successfully loaded using adapter 2.
IBM Token-Ring Shared RAM Family NDIS2 Driver, Version 3.2
IBM Token-Ring Shared RAM Family NDIS2 Driver, Version 3.2
IBM LANVDD is loaded and operational.
IBM OS/2 LAN Netbind
Drivername IBMTOK, UAA=10005AEC13A2, ROM=C8000, RAM=C0000, IO=0A20, IRQ=10
Token-Ring adapter data rate is 4 Mbps.
IBM LANDD is accessing IBM 802.5 LAN Interface.
Token-Ring adapter opened in half duplex mode successfully.
Adapter 0 was initialized and opened successfully.
Adapter 0 is using node address 400052005184.
IBM LANDD was successfully bound to MAC: IBMTOK_nif->VECTOR.
Drivername IBMTOK2, UAA=10005AAD94DA, ROM=CA000, RAM=C4000, IO=0A24, IRQ=09
Token-Ring adapter data rate is 4 Mbps.
IBM LANDD is accessing IBM 802.5 LAN Interface.
Token-Ring adapter opened in half duplex mode successfully.
Adapter 1 was initialized and opened successfully.
Adapter 1 is using node address 400052005183.
IBM LANDD was successfully bound to MAC: IBMTOK_nif2->VECTOR.
IBM LANDD is accessing IBM 802.5 LAN Interface.
Adapter 2 was initialized and opened successfully.
Adapter 2 is using node address 400011110002.
IBM LANDD was successfully bound to MAC: FREFRM_nif.
```

Figure 241. LANTRAN.LOG in Frame Relay

11.4 Problem Determination

We hope this section will be a first source of help when you are troubleshooting problems related to frame relay support in IBM Communications Server Release 4.1. We include a frame relay trace obtained with the tools described below.

11.4.1 Trouble Indicators

Communications Server′s frame relay does not seem to work:

1. Are there any error messages in the \IBMCOM\LANTRAN.LOG?
2. Did messages indicating correct operation appear in \IBMCOM\LANTRAN.LOG?
3. Are you using an unsupported network adapter?
4. Are your network adapters set up correctly?

Check the IRQ level and shared RAM with other adapters that are installed.

The system′s performance is not adequate:

1. Does your workstation have enough memory and a fast enough processor

for all the work you have given it?

2. Is your WAN line speed fast enough?
3. Are your packet and frame sizes optimized for your WAN line speed?
4. Do your queue depths accommodate your overall latency needs?

11.4.2 Troubleshooting Tips for Frame Relay

Over frame relay, link failures (DLCI inactive) or network loss can occur for the following reasons:

- Loss of data set ready (DSR)
- Excessive delays in sending or receiving status messages
- Errors in status sequences

If you experience frame relay link failure or network loss intermittently and unexpectedly (for instance, during periods of sustained WAN traffic), there are two likely reasons:

- Physical connectivity problems
- Problems communicating status information across the WAN due to high data traffic volumes

If you experience these problems across a frame relay network or a leased-line running frame relay, you should:

1. Check the physical connection to the frame relay network. If the physical connection seems good, check out the WAN adapter (either by running diagnostics or swapping it with another WAN adapter). If the physical connection and adapter appear to be working, continue with the following steps.
2. Use MPTS to edit the following WAN access for 802.5 protocol parameters (see Figure 233 on page 262):

   N392 Error Threshold -- increase, up to 10 (events)

   T392 Polling Verification Timer -- increase, up to 30 (seconds)

   N393 Monitored Events Count -- increase, up to 10 (status messages)

   Increasing these parameters allows for more flexible status handling during periods of peak data traffic. If you are connected to a frame relay network, you might have to check with your frame relay provider for the acceptability of these parameter settings.
3. If the problem persists, contact your frame relay service provider and report the problem.

11.4.3 Tools

There are some tools available to obtain status and statistics regarding the frame-relay device driver and WAC adapter, as well as a trace formatting tool. These tools will be located in the CMLIB directory and installed when the frame-relay additional function is selected in IBM Communications Server Release 4.1.

11.4.3.1 FRNSTAT.EXE

This displays the frame-relay device driver's status and statistics.

Usage: frnstat -i -c -e
 -i = interface
 -c = circuits
 -e = last line error
 -a = display all

The following figure is the output from FRNSTAT.EXE:

```
LA LMI T391 N391 N392 N393 Max VCs Multicast
== === ===  ==== ==== ==== === === ========
01   3 0010 0006 0003 0004    0050 Non-BCast

LA LINK S Sent Pkts  Rcv Pkts   Sent Bytes Rcv Bytes   Rcv FECNsRCV BEC
== ==== = ========== ========== ========== ==========  ================
01 0032 A 0003723300 0003734182 0787168997 0909818286  0000000000000000

LA Error Type     Frame Data (in Hex)
01 No Errors      No Frame Data
```

Figure 242. FRNSTAT Output

Of special interest are the LINK number (that is, the DLCI number) that is in use and the S status of the link, which in this case is A (active).

Also you could review the timers and the statistics of packets and bytes that are sent and received.

11.4.3.2 WACSTATF.EXE

This displays the WAC adapter's status and statistics.

It provides a simple display of the WAC control lines and frame errors. This information can be especially useful for diagnosing DSU/CSU or line quality related problems.

```
Number of WAC ports = 1

Port Type  In Speed   Out Speed
==== ===== ========== ==========
 01  V35   0000065536 0000065536

Port Clock FCS Errors Tx Underruns Rcv Overruns Int Frames Abort Frames
==== ===== ========== ============ ============ ========== ============
 01    Ext 0000000000   0000000000   0000000000 0000000000   0000000000

Port Bits Stop  Parity Auto Baud Parity Errs Frame Errs Overrun Errs
==== ==== ===== ====== ========= =========== ========== ============

Port  RTS  CTS  DSR  DTR  RI   DCD  SQ   SRS  SRTS SCTS SDCD
====  ==== ==== ==== ==== ==== ==== ==== ==== ==== ==== ====
 01    On   On   On   On  ---   On  ---  ---  ---  Off  Off
```

Figure 243. WACSTAT Output

11.4.4 Frame Relay Tracing and Troubleshooting

Tracing can be used to troubleshoot network-based problems. The following sections describe how to activate frame relay tracing and how to interpret the trace information.

11.4.4.1 Enabling Frame Relay Tracing

To enable frame relay tracing, do as follows:

1. Make sure that you have the following OS/2 programs in the \OS2 directory of your boot drive:

   TRACE.EXE in \OS2

   TRACEFMT.EXE in \OS2

   TRACEFMT.DLL in \OS2\DLL

   TRACEFMT.HLP in \OS2\HELP

   If you do not find these programs, then you will need to install them. To install the preceding OS/2 programs:

   - At an OS/2 prompt, type `INSTALL`.
   - At the first panel (System Configuration), click on the **OK** button.
   - On the OS/2 Setup and Installation window, click on **Serviceability and Diagnostics Aids**. Then click on **Install** to proceed with the installation. (You might be asked to insert different OS/2 installation diskettes into a diskette drive.)

2. Edit \CONFIG.SYS to include the following statements (anywhere):

   TRACEBUF=63

   TRACE=ON 164

   For more information on these statements, you can use the OS/2 HELP facility (for instance, type at an OS/2 prompt HELP TRACEBUF and HELP TRACE).

3. Run LAPS and double-click on **Frame Relay WAN Access for 802.5 Protocol** in your current Configuration window. At the bottom of the list of parameters you will see: Allow tracing for this port? This parameter defaults to NO. Change this parameter′s value to YES for each port (that is, in each Frame Relay WAN Access for 802.5 Protocol entry) on which you desire tracing. After updating this parameter, exit LAPS.

4. Reboot your frame relay machine.

   You will now be tracing every frame sent through the frame relay device driver. To turn tracing off at any time, use the following OS/2 command:

   `TRACE OFF`

   To turn tracing (back) on, type:

   `TRACE ON 164`

   The parameter 164 specifies that tracing should be turned on for certain data communications device drivers, including frame relay driver.

   For more information on OS/2 trace commands, type `HELP TRACE` at any OS/2 prompt.

   There is a /C option that clears the trace buffer that could be more useful for problem analysis than isolating a single problem at a time.

5. When you want to gather and store trace data (for instance, after a failure occurs that you want traced), type the following command at an OS/2 prompt:

   TRACEFMT

   This will bring up a PM session filled with a variety of data. An example of the data you might see may look like Figure 244:

```
Sample TRACEFMT Display
System Trace Formatter Output
Formatted Events

Buffer size = 63 K Bytes

EVENT# [0008]  PID [0000]  MAJOR [A4]  MINOR [004A]  LENGTH [003C]  TIME: ......
10 40 40 00 00 00 00 02 40 00 00 00 00 01 04 04 EC 44 2C 00 02 02 03 F2 00 90
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00

EVENT# [0009]  PID [0000]  MAJOR [A4]  MINOR [004A]  LENGTH [0006]  TIME: ......

80 00 80 C2 00 09

EVENT# [000A]  PID [0000]  MAJOR [A4]  MINOR [004A]  LENGTH [0004]  TIME: ......

08 01 03 00
```

Figure 244. TRACEFMT Output

   Store this data for subsequent processing. To store the data, click on the **File** pull-down menu or type ALT-F, and then click on the **Save as** menu or type S. Next, click on the submenu **Save formatted trace data** or type F. This will present a pop-up menu that will allow you to specify a trace-file name. Choose any name you want (for example, FREL.TRC) and save this file. When the trace-file save has been completed (indicated by another pop-up), exit the OS/2 TRACEFMT program.

6. You now have two options. You can analyze the trace data yourself, or you can contact IBM support to assist you in analyzing your trace, or both. If you choose to contact IBM support, send them the formatted trace data you have just saved. Run Gather Problem Determination (CMPD.CMD) located in the Problem Determination Folder. Running CMPD will gather the following files:

   \CONFIG.SYS

   \IBMCOM\PROTOCOL.INI

   \IBMCOM\LANTRAN.LOG

   If you choose to analyze the formatted trace data yourself, then copy (or move) the formatted trace files to the \CMLIB directory (where the frame relay FRETRACE.EXE program is located) and issue one of the following commands:

   FRETRACE <filename> - for a simplified ladder diagram

   or

   FRETRACE <filename> /c - for a more detailed ladder diagram

Where <filename> is the name of the file you created with the OS/2 TRACEFMT program in step (5). (For example, you can issue the command FRETRACE FRELAY.TRC /C.)

Note

This tool only runs on the English version of OS/2.

11.4.4.2 Interpreting Frame Relay Trace Output

FRETRACE will generate several output files. The output files all use the same base file name as the input file name, but have different file name extensions. The output files and their extensions are as follows:

1. The .ANA file protocol-analyzes each PDU.
2. The .LDR file ladder-diagrams each PDU.
3. The .DLC file provides data link connection statistics.

The FRETRACE output file that is most likely to be of use for troubleshooting ends with the suffix .LDR. In addition, the .LDR file can be used in tandem with the .ANA file, because the .LDR provides a high-level ladder diagram of network traffic, whereas the .ANA file provides a more low-level data analysis. The .LDR frames can be correlated to the appropriate .ANA frame by referencing their common PDU number. The following figure shows sample snippets from .LDR (ladder diagram) and .ANA (protocol analysis) files.

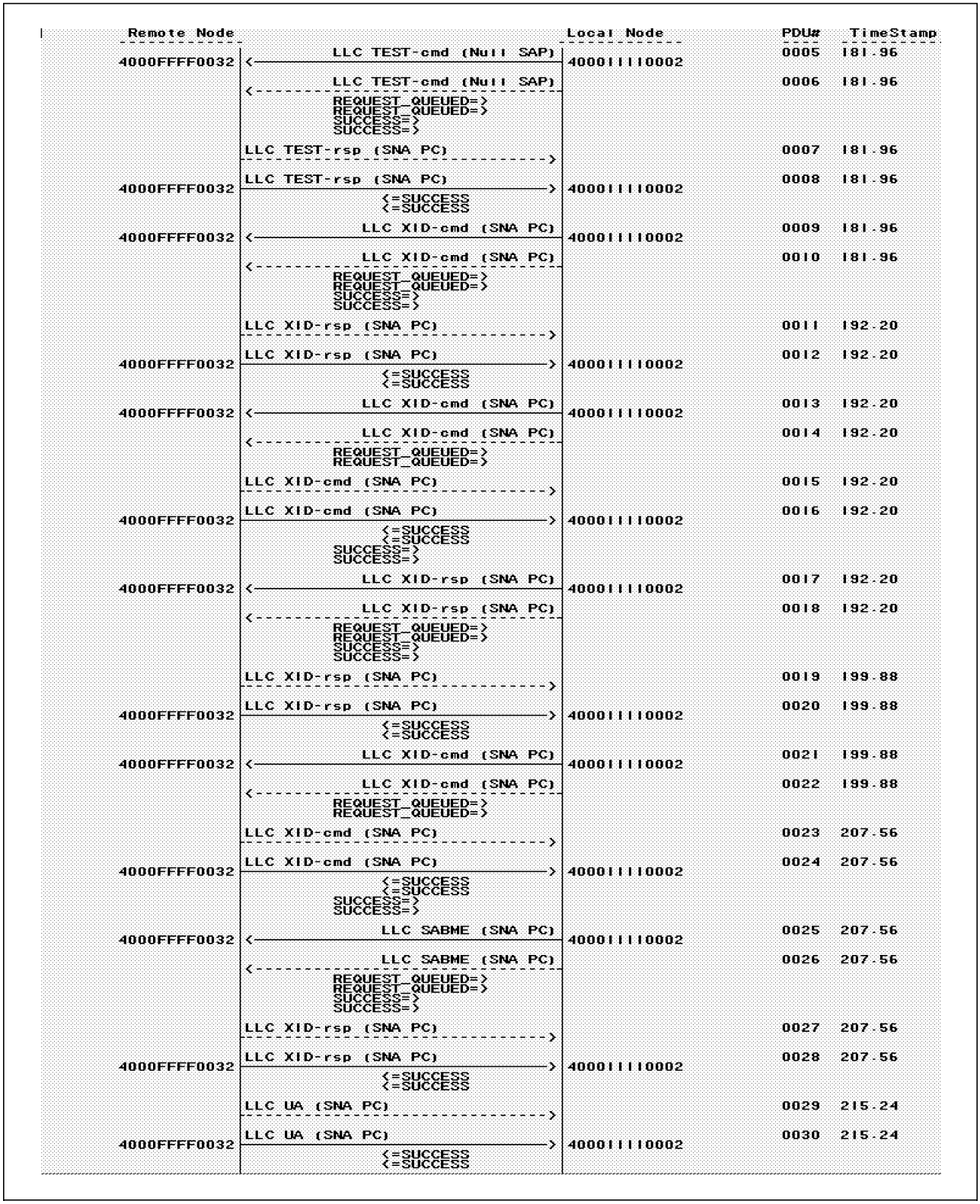


Figure 245. .LDR Routed Format Output of Link Establishment

There are some details to be considered:

- Note that the left column has the virtual MAC address of the destination node. It uses the DLCI number 32.
- There are always two PDUs for each packet.

The order is based on the time of arrival.

The dotted line (----) corresponds to the real frame that is sending or receiving by the line. Note that this does not have the MAC addresses in it, because it is a routed format frame.

The continuous line (____) corresponds to the upper protocol frame (802.2 in this case). This is the one that always has the MAC addresses in it, because it simulates a token-ring network and matches the MAC addresses that are supposed to be used.

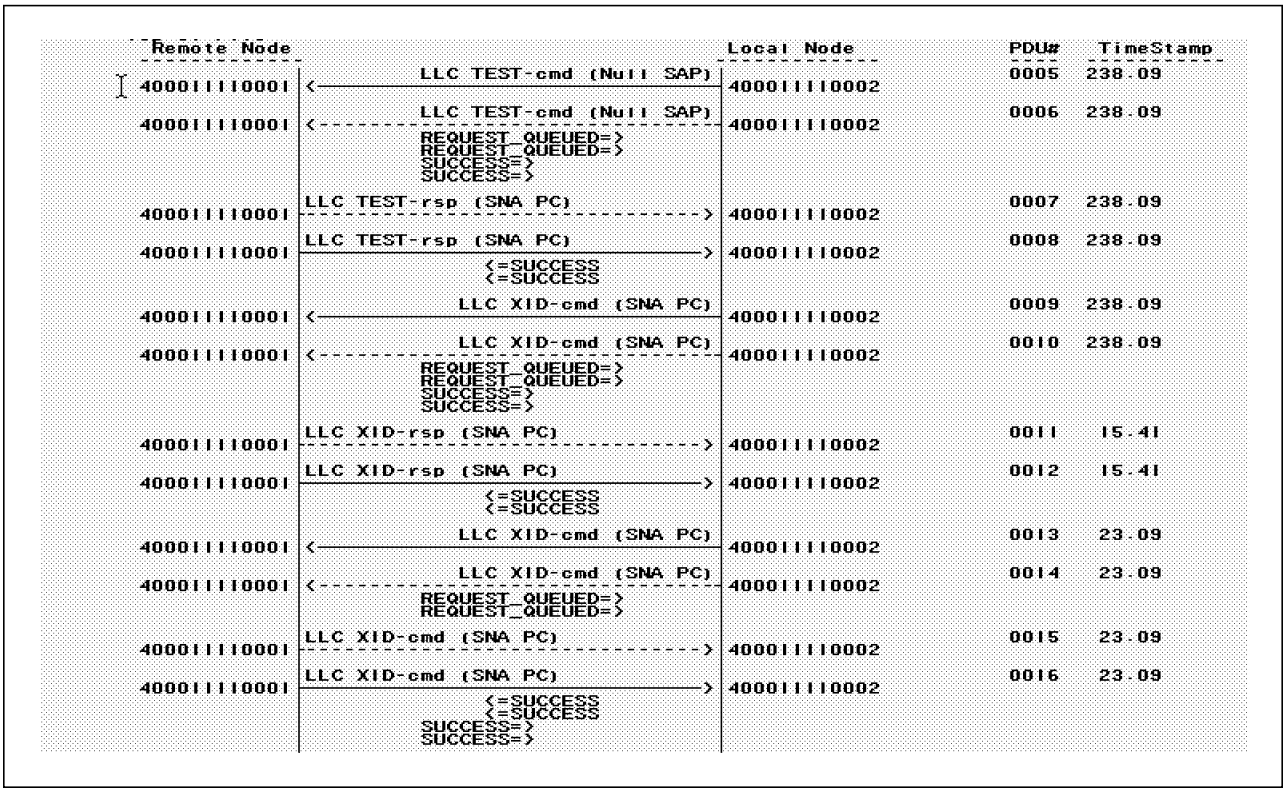


Figure 246. .LDR Bridged Format Output of Link Establishment (Extract)

This is an extract of the link establishment process in order to illustrate a comparison to the routed format.

Note that both PDUs always have the MAC addresses, because it is a bridged format frame.

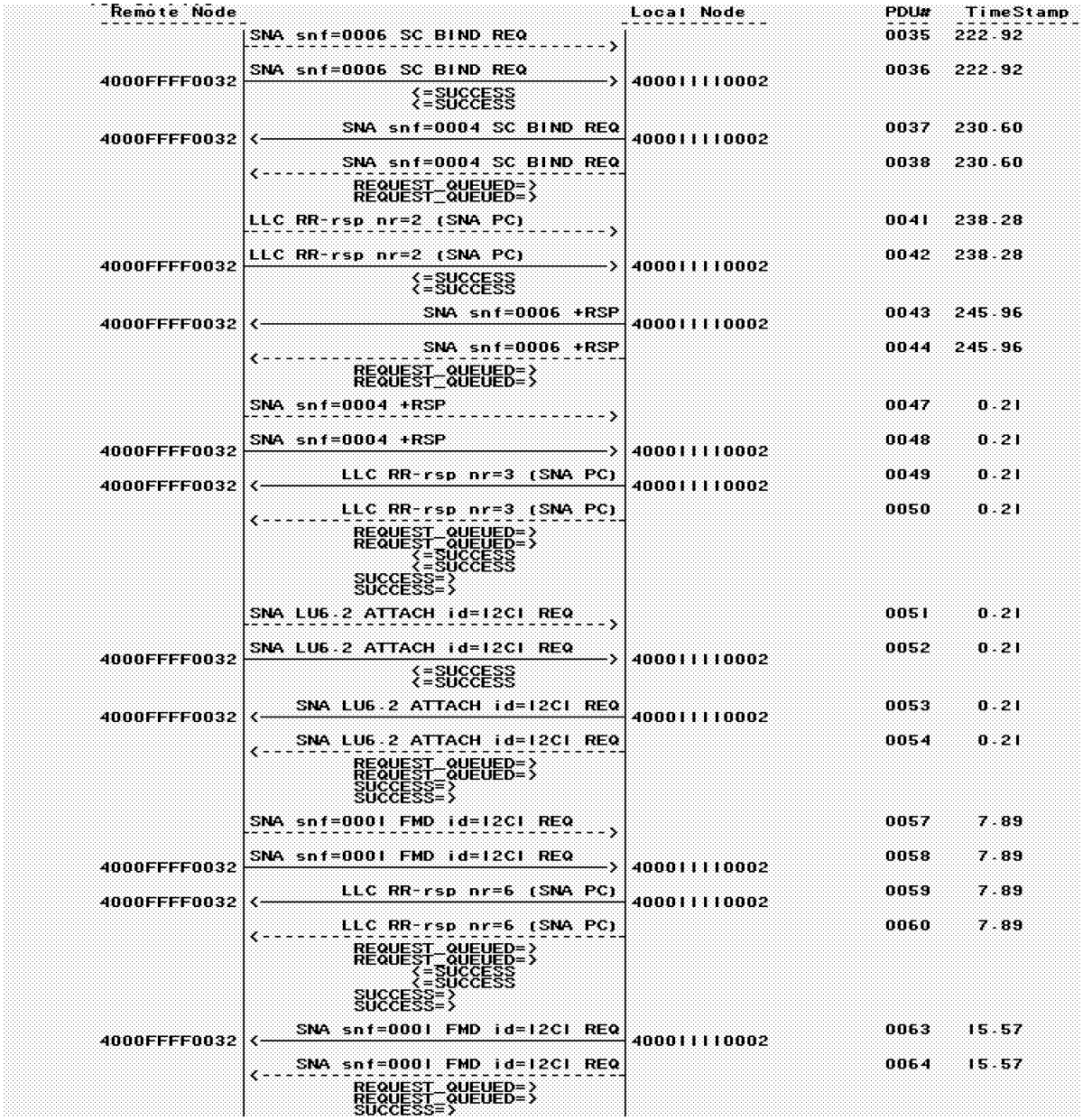


Figure 247. .LDR Routed Format Output of Session Establishment

Figure 247 shows you a normal session establishment. Note that the program that formats the trace gives you good detail on the frames. You can see that it identifies BINDs, ATTACHs and FMDs. Some PDUs are removed to reduce the output size.

```
================================================================================
                              *** PDU #0014 ***
------------------------------Original Trace Data-----------------------
08010308 4C807083 0404BF32 6505D053 03000080 FAC10000 0000A001 010B7100
08B00000 00000400 0E11F4E4 E2C9C2D4 D9C14BE6 E3D9F0F5 F3F0F30E 09F7E6C1
C3C6D940 40401028 00111104 0E02F5F6 F2F1F2F5 F4F0F0F2 F2F01611 03130011
F8F5F9F5 F0D2C6F2 F3F0F0D4 C3F5F6F3
---------------------------------PDU Summary----------------------------
192.20  PDU #0014. LLC XID-cmd (SNA PC) ◄==
             Total bytes = 112
             Analysis of protocol-specific overhead in bytes:
                Frame Relay    = 2   (DLCI = 32) [FCS bytes not included]
                RFC-1490       = 2   (LLC XID-cmd (SNA PC))
                Other          = 108
================================================================================
================================================================================
                              *** PDU #0035 ***
------------------------------Original Trace Data-----------------------
08010308 4C807083 04040000 2F000102 00066B81 00310013 07B0B050 33018086
86800106 02000000 00000040 1C234000 10E4E2C9 C2D4D9C1 4BE6E3D9 F0F5F1F8
F4310008 02C3D7E2 E5C3D4C7 090301CD D299FDCD D2991104 E4E2C9C2 D4D9C14B
E6E3D9F0 F5F1F8F4 0A130071 EB7DC8E7 8AA15B00 10E4E2C9 C2D4D9C1 4BE6E3D9
F0F5F3F0 F36019D9 6FE03AD3 FF1FC310 E4E2C9C2 D4D9C14B E6E3D9F0 F5F1F8F4
2C090407 C3D7E2E5 C3D4C7
---------------------------------PDU Summary----------------------------
222.92  PDU #0035. SNA snf=0006 SC BIND REQ ==►
             Total bytes = 171
             Analysis of protocol-specific overhead in bytes:
                Frame Relay    = 2   (DLCI = 32) [FCS bytes not included]
                RFC-1490       = 2   (LLC INFO (SNA PC))
                SNA FID2/APPN  = 167  (TH[0] = 2F)
                                      (DAF' = 01, OAF' = 02, SNF = 0006)
                                      (RH: 6B 81 00)
================================================================================
```

Figure 248. .ANA Routed Format Output of XID and BIND

In the previous figures, you could see an XID and a BIND as they are received by the frame relay protocol. Remember that there are always two PDUs by frame.

Here we interpret the TH and identify the RH headers of the SNA frame.

Also you could identify the routed format as follows:

- Bytes one and two (0801) are the Q.922 address. Identify the DLCI 32.
- Byte three (03) is the Control.
- Byte four (08) is the NLPID that identifies the Q.933 format.
- Byte five and six (4C80) are the L2 ID that identifies the 802.2.
- Bytes seven and eight (7083) are the L3 ID.

For more details, refer to the RFC 1490 specification in Appendix G, “RFC 1490 Extract” on page 319

```
================================================================================
                              *** PDU #0037 ***
------------------------------Original Trace Data-------------------------
08010300 800080C2 00091040 40001111 00024000 11110001 04040002 2F000102
00046B81 00310013 07B0B050 33018086 86800106 02000000 00000040 1C234000
10E4E2C9 C2D4D9C1 4BE6E3D9 F0F5F1F8 F4310008 02C3D7E2 E5C3D4C7 090301CE
D299FCCE D2991104 E4E2C9C2 D4D9C14B E6E3D9F0 F5F1F8F4 0A13000B DE63D57E
8375A000 10E4E2C9 C2D4D9C1 4BE6E3D9 F0F5F3F0 F36019D9 6FE03ACF 0AE15F10
E4E2C9C2 D4D9C14B E6E3D9F0 F5F1F8F4 2C090407 C3D7E2E5 C3D4C7
---------------------------------PDU Summary----------------------------
143.41  PDU #0037. SNA snf=0004 SC BIND REQ ==►
          Total bytes = 187
          Analysis of protocol-specific overhead in bytes:
             Frame Relay    = 2   (DLCI = 32) [FCS bytes not included]
             RFC-1490       = 8   (RFC-1490 "Bridged token-ring w/o FCS")
                                  (OUI = 0080C2, PID = 0009)
             802.5 MAC      = 14  (DA = 4000 1111 0002)
                                  (SA = 4000 1111 0001)
             802.2 LLC      = 4   (DSAP = 04, SSAP = 04)
             SNA FID2/APPN  = 159  (TH[0] = 2F)
                                  (DAF' = 01, OAF' = 02, SNF = 0004)
                                  (RH: 6B 81 00)
================================================================================
```

Figure 249. .ANA Bridged Format Output of BIND

There are some differences between this bridged format and the previous routed format frames.

There are 16 extra bytes in any bridged format frame. It is an overhead.

Also you could identify the bridged format as follows:

- Bytes one and two (0801) are the Q.922 address. Identify the DLCI 32.
- Byte three (03) is the Control.
- Byte four (00) is a PAD.
- Byte five (80) is the NLPID for SNAP.
- Bytes six, seven and eight (0080C2) are OUI.
- Bytes nine and ten (0009) are the PID (802.5 in this case).

For more details, refer to the RFC 1490 specification in Appendix G, "RFC 1490 Extract" on page 319

In the PDU Summary, the parameter RFC-1490=8 points to a bridged format frame that is easy to see.

```
Total  +-----Node1-----+ +-----Node2-----+
Frames [MAC Address SAP] [MAC Address SAP]   Route Designators (Ring-Bridge)
------ ------------ ---- ------------ ---- ------------------------------
    50 400011110002  04  4000FFFF0032  04
------ ------------ ---- ------------ ---- ------------------------------
          Lower-Layer Protocols:
             802.X bytes     =   889 (22%)
          SNA Protocols:
             APPN/HPR bytes  =  2280 (58%)
          Miscellaneous:
             'Other' bytes   =   742 (18%)
          Total Bytes        =  3911
------ ------------ ---- ------------ ---- ------------------------------
    46 400011110002  C8  4000FFFF0032  C8
------ ------------ ---- ------------ ---- ------------------------------
          Lower-Layer Protocols:
             802.X bytes     =   782 (17%)
          SNA Protocols:
             APPN/HPR bytes  =  3633 (82%)
          Total Bytes        =  4415
------ ------------ ---- ------------ ---- ------------------------------
     2 400011110002  04  4000FFFF0032  00
------ ------------ ---- ------------ ---- ------------------------------
          Lower-Layer Protocols:
             802.X bytes     =    34 (94%)
          Miscellaneous:
             'Other' bytes   =     2 ( 5%)
          Total Bytes        =    36
```

Figure 250. .DLC Output

This information could be very useful for statistics of the traffic that certain pairs of nodes have, or you could isolate the time of your trace and determine the real amount of WAN traffic that certain applications produce.

This output was taken from a link connect process. The preceding trace was generated by FRETRACE without using FRETRACE's /C option. If the /C option had also been used, then control information related to OS/2 NDIS interfaces would be displayed. This type of control information is particularly useful if your trace indicates that you are only transmitting frames but are not receiving any frames, or if you believe that there might be a problem between the frame relay component and the IBMWAC device-driver to which it is bound.

For related information on the output created by FRETRACE, go to the directory where FRETRACE.EXE is stored. At the OS/2 command prompt type FRETRACE.

Note

FRETRACE.EXE help message is formatted for a full-screen OS/2 session. If FRETRACE is invoked from an OS/2 window, its window output might be displayed too quickly for viewing.

If you take your own traces you could also see certain frames that correspond to the LMI traffic. The LMI traffic may also be useful is some cases when troubleshooting problems related to the frame relay network appears.

For example, you could see if the timers are working correctly or if certain DLCIs are available.

11.4.5 Tracing IP Networks

Tracing can be used to troubleshoot network-based TCP/IP problems. The following sections describe how to activate TCP/IP tracing and how to interpret the trace information.

11.4.5.1 Enabling IP Tracing

To activate an IP trace you only have to use the IPTRACE command that is located in the \MPTN\BIN directory.

```
[C:\mptn\bin]iptrace -?

 IPTRACE.EXE: Turn on packets tracing and save log in iptrace.dmp.
              Stop tracing by pressing ◄enter► or ◄ctrl-brk►.
              IPFORMAT.EXE can later be used to interpret the log file.

 USAGE:
          iptrace [-i] [interface]

 EXAMPLES:
          iptrace         (traces all packets on all interfaces)
          iptrace lan0    (traces all packets on lan0 interface)
          iptrace -i      (traces IP packets on all interfaces)
          iptrace -i lan0 (traces IP packets on lan0 interface)
```

Figure 251. IPTRACE Command - Help

11.4.5.2 Interpreting IP Trace Output

The IPTRACE command produces output that resides in IPTRACE.DMP, which should be formated using the IPFORMAT.EXE command.

```
[C:\mptn\bin]ipformat -?
 IP Trace Formatter (IPFORMAT)
 (C) Copyright IBM Corp. 1993, 1996
 Release: V1.40 07/15/96
  IPFORMAT
[-a][-d][-e<opt>][-f<file>][-h][-n][-s<hwaddress>][-x]
   (data can be piped to a file (►filename) for browsing using an editor)
     -a Don't format ARP/RARP packets
     -d Don't display data portion of packet
     -e Exclude ◄opt► packets.  opt can be:
          a (ARP), t (TCP), u (UDP), i (ICMP), g (IGMP)
     -f ◄input file► (default IPTRACE.DMP)
     -h display raw data packet after formatted info
     -n Don't display hex data for unknown data type
     -s ONLY format data for ◄hwaddress► (Source or Destination)
         where hwaddress is the 12 digit hex address
             ie: 123456789ABC
     -x Convert datafile to "Sniffer"  format file
     -? Help (This screen)
```

Figure 252. IPFORMAT Command - Help

The following figures are examples of IP traces of an ARP flow and a PING flow.

```
------------------------- #:10 -------------------------
 Delta Time:  0.000sec   Packet Length: 52 bytes (34 hex)
 802.5:  Dest: FF: FF: FF: FF: FF: FF   Source: C0: 00: 11: 11: 00: 02
------------------------- ARP -------------------------
 ARP:  Hardware Type:6     (IEEE 802)
 ARP:  Protocol Type:0800 (IP Address)
 ARP:  Hardware Len:6
 ARP:  Protocol Len:4
 ARP:  Operation:1  (ARP Request)
 ARP:  Sender HW address: 400011110002
 ARP:  Sender PA: 130.001.001.001.
 ARP:  Target HW address: 000000000000
 ARP:  Target PA: 130.001.001.002.

------------------------- #:11 -------------------------
 Delta Time:  0.000sec   Packet Length: 52 bytes (34 hex)
 802.5:  Dest: 40: 00: 11: 11: 00: 02   Source: C0: 00: FF: FF: 00: 32
------------------------- ARP -------------------------
 ARP:  Hardware Type:6     (IEEE 802)
 ARP:  Protocol Type:0800 (IP Address)
 ARP:  Hardware Len:6
 ARP:  Protocol Len:4
 ARP:  Operation:2  (ARP Response)
 ARP:  Sender HW address: 4000FFFF0032
 ARP:  Sender PA: 130.001.001.002.
 ARP:  Target HW address: 400011110002
 ARP:  Target PA: 130.001.001.001.
```

Figure 253. IPFORMAT Output ARP

First of all, the local machine from where we took the IPTRACE has a real MAC address of 400011110002 for its frame relay adapter, and its IP address is

130.1.1.1. The remote machine has a real MAC address of 400011110001 for its frame relay adapter, and its IP address is 130.1.1.2.

Then, as you can see, the local machine is sending a broadcast ARP request. Also, not shown in the trace, this machine is also receiving an ARP request, because it is a broadcast.

Then the remote machine responds with an ARP response. See that the sender HW address of this response is 4000FFFFF0032, so it is using the routed frame format.

```
------------------------- #:12 -------------------------
 Delta Time:  0.031sec   Packet Length: 108 bytes (6C hex)
 802.5:  Dest: 40: 00: FF: FF: 00: 32   Source: C0: 00: 11: 11: 00: 02
 802.5:  Dest: 130.001.001.002    Source:  130.001.001.001
---------------------- IP HEADER -----------------------
 IP:  Version: 4 Correct    Header Length: 20 bytes
 IP:  Type Of Service: 00
 IP:     000. ....  Routine
 IP:     ...0 ....  Normal Delay
 IP:     .... 0...  Normal Throughput
 IP:     .... .0..  Normal Reliability
 IP:  Total Len: 84 (x54) bytes          Id: FF41
 IP:  Flags: 0
 IP:     .0..       May Fragment
 IP:     ..0.       Last Fragment
 IP:  Fragment Offset: 000
 IP:  Time To Live: 255 sec    Protocol: 1  ICMP
 IP:  Header Checksum: B661    (Correct)
 IP:  No Options
--------------------- ICMP HEADER ----------------------
 ICMP:  Type: 08  Echo
 ICMP:  Checksum: F492    (Correct)
 ICMP:  Identify: 29A9
 ICMP:  Sequence #: 0000
--------------------------------- DATA ---------------------------------
0000 4E BD 53 32 40 D1 0C 00    08 09 0A 0B 0C 0D 0E 0F   N.S2@.........
0010 10 11 12 13 14 15 16 17    18 19 1A 1B 1C 1D 1E 1F   ..............
0020 20 21 22 23 24 25 26 27    28 29 2A 2B 2C 2D 2E 2F    !"#$%&'()*+,-.
0030 30 31 32 33 34 35 36 37                              01234567
```

Figure 254. IPFORMAT Output PING

Here we see an ICMP Echo that is the request that the local machines send and that is going to be responded to by an ICMP Echo Reply by the remote machine.

As you can see in the 802.5 row, it is also a routed frame format as defined in RFC 1490.

The last thing is a frame relay trace of the PING.

```
================================================================================
                              *** PDU #0009 ***
------------------------------Original Trace Data-------------------------
080103CC 45000054 30B10000 FF0184F2 82010101 82010102 0800218C 1CD90000
8B0D5532 E0570E00 08090A0B 0C0D0E0F 10111213 14151617 18191A1B 1C1D1E1F
20212223 24252627 28292A2B 2C2D2E2F 30313233 34353637
---------------------------------PDU Summary----------------------------
241.11  PDU #0009. ICMP Echo Request (0000) ◄==
             Total bytes = 88
             Analysis of protocol-specific overhead in bytes:
                Frame Relay    = 2   (DLCI = 32) [FCS bytes not included]
                RFC-1490       = 2   (RFC-1490 "Internet IP")
                IP             = 20  (Source      = 82.01.01.01)
                                     (Destination = 82.01.01.02)
                Other          = 64
================================================================================
```

Figure 255. Frame Relay Trace of PING

Note the NLPID of '0xCC' that corresponds to the IP traffic and that the IP addresses are in hex format.

Part 8. Appendixes

Appendix A. Example Configuration Files

This appendix presents several configuration files.

A.1 Local Configuration File Example

Figure 256 shows an example of the local configuration file (AXSCONF.INI) generated by this sample scenario. The local configuration file specifies the name of the global configuration file, LANGWA.RSP.

```
RSPFILE = LANGWA.RSP
```

Figure 256. Example of Local Configuration File (AXSCONF.INI)

A.2 Global Configuration File Examples

This section contains the following global configuration files generated for the LANGWa, LANGWx and LANGWy gateways:

- A.2.1, “Global Configuration File for LANGWx”
- A.2.2, “Global Configuration File for LANGWy” on page 293
- A.2.3, “Global Configuration File for LANGWa” on page 297

A.2.1 Global Configuration File for LANGWx

Figure 257 on page 290 shows the global configuration file generated for the LANGWx LAN Gateway.

```
LAN_GATEWAY=(
  NAME=LANGWX
  GATEWAY=(
    NAME=LANGWX
    IPX=0
    NETBIOS=1
    USE_SNA=1
    USE_TCP=0
    QUALIFY_IPX=0
    REGION=LANX
    LU=USIBMRA.LANGWX
    HOSTNAME=
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x00000000
    NETWARE_SNAP=0x00000000
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=8
    LINK=(
      NAME=LOOOOOO1
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 257 (Part 1 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWx

```
  GATEWAY=(
    NAME=LANGWY
    IPX=1
    NETBIOS=0
    USE_SNA=0
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANY
    LU=
    HOSTNAME=129.1.1.25
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x00000000
    NETWARE_SNAP=0x00000000
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=25
    LINK=(
      NAME=LO0000002
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 257 (Part 2 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWx

```
  GATEWAY=(
    NAME=LANGWA
    IPX=1
    NETBIOS=1
    USE_SNA=1
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANA
    LU=USIBMRA.LANGWA
    HOSTNAME=129.1.1.1
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x0000000a
    NETWARE_SNAP=0x000000aa
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=46
    NAMEQUALIFIER=(
      NAME=NBFILSRV
)
```

Figure 257 (Part 3 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWx

```
    LINK=(
      NAME=LOOOOOO3
      GATEWAY=LANGWX
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    LINK=(
      NAME=LOOOOOO4
      GATEWAY=LANGWY
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
)
```

Figure 257 (Part 4 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWx

A.2.2 Global Configuration File for LANGWy

Figure 258 on page 294 shows the global configuration file generated for the LANGWy LAN Gateway.

```
LAN_GATEWAY=(
  NAME=LANGWY
  GATEWAY=(
    NAME=LANGWX
    IPX=0
    NETBIOS=1
    USE_SNA=1
    USE_TCP=0
    QUALIFY_IPX=0
    REGION=LANX
    LU=USIBMRA.LANGWX
    HOSTNAME=
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x00000000
    NETWARE_SNAP=0x00000000
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=8
    LINK=(
      NAME=LOOOOOO1
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 258 (Part 1 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWy

```
  GATEWAY=(
    NAME=LANGWY
    IPX=1
    NETBIOS=0
    USE_SNA=0
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANY
    LU=
    HOSTNAME=129.1.1.25
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x0000000a
    NETWARE_SNAP=0x000000aa
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=25
    LINK=(
      NAME=LO0000002
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 258 (Part 2 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWy

```
  GATEWAY=(
    NAME=LANGWA
    IPX=1
    NETBIOS=1
    USE_SNA=1
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANA
    LU=USIBMRA.LANGWA
    HOSTNAME=129.1.1.1
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x0000000a
    NETWARE_SNAP=0x000000aa
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=46
    NAMEQUALIFIER=(
      NAME=NBFILSRV
)
```

Figure 258 (Part 3 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWy

```
    LINK=(
      NAME=LOOOOOO3
      GATEWAY=LANGWX
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    LINK=(
      NAME=LOOOOOO4
      GATEWAY=LANGWY
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
)
```

Figure 258 (Part 4 of 4). Example of Global Configuration File (LANGWA.RSP) for LANGWy

A.2.3 Global Configuration File for LANGWa

Figure 259 on page 298 shows the global configuration file generated for the LANGWa LAN Gateway.

```
LAN_GATEWAY=(
  NAME=LANGWA
  GATEWAY=(
    NAME=LANGWX
    IPX=0
    NETBIOS=1
    USE_SNA=1
    USE_TCP=0
    QUALIFY_IPX=0
    REGION=LANX
    LU=USIBMRA.LANGWX
    HOSTNAME=
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x00000000
    NETWARE_SNAP=0x00000000
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=8
    LINK=(
      NAME=LOOOOOO1
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 259 (Part 1 of 4). Example of Global Configuration File (SAMPLE.RSP) for RALEIGH

```
  GATEWAY=(
    NAME=LANGWY
    IPX=1
    NETBIOS=0
    USE_SNA=0
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANY
    LU=
    HOSTNAME=129.1.1.25
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x00000000
    NETWARE_SNAP=0x00000000
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=25
    LINK=(
      NAME=LO0000002
      GATEWAY=LANGWA
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
```

Figure 259 (Part 2 of 4). Example of Global Configuration File (SAMPLE.RSP) for RALEIGH

```
  GATEWAY=(
    NAME=LANGWA
    IPX=1
    NETBIOS=1
    USE_SNA=1
    USE_TCP=1
    QUALIFY_IPX=0
    REGION=LANA
    LU=USIBMRA.LANGWA
    HOSTNAME=129.1.1.1
    ADAPTER=0
    LOOPBACK=0
    ACCEPT_NEW_PARTNERS=0
    REMEMBER_NEW_PARTNERS=0
    RETRY_INTERVAL=2
    DYNAMIC_TIMEOUT=15
    BUFFERS=200
    BUFFER_THRESHOLD=80
    IPX_SOURCE_ROUTING=1
    NETWARE_802_2=0x0000000a
    NETWARE_SNAP=0x000000aa
    RIP_ENTRIES=100
    RIP_THRESHOLD=80
    SAP_ENTRIES=100
    SAP_THRESHOLD=80
    CIRCUITS=255
    CIRCUIT_THRESHOLD=80
    STATIONS=50
    STATION_THRESHOLD=80
    LOCAL_BUSY=50
    LOGENABLE=1
    EVENTFILE=0
    EVENTWRAP=1
    EVENTWRAPSIZE=3000
    MESSAGEFILE=0
    MESSAGEWRAP=1
    MESSAGEWRAPSIZE=3000
    AUTOSTART=1
    SNMP=0
    FILTER=0
    FILTERFILE=
    FILTERINPUTFILE=
    POSITION=46
    NAMEQUALIFIER=(
      NAME=NBFILSRV
)
```

Figure 259 (Part 3 of 4). Example of Global Configuration File (SAMPLE.RSP) for RALEIGH

```
    LINK=(
      NAME=LOOOOOO3
      GATEWAY=LANGWX
      TYPE=AUTOLINK
      PROTOCOL=SNA
      MODE=#BATCH
)
    LINK=(
      NAME=LOOOOOO4
      GATEWAY=LANGWY
      TYPE=AUTOLINK
      PROTOCOL=IP
)
    COLORS=15 -2 6 -2 6 2 -2 15 -1
)
)
```

Figure 259 (Part 4 of 4). Example of Global Configuration File (SAMPLE.RSP) for RALEIGH

Appendix B. Filter Program Application Program Interface

This chapter describes the application program interface (API) for the filter program and includes the following sections:

- B.1, “Overview of the Filtering Process”
- B.2, “Supplied Filter (AXSFILTR.DLL)”

B.1 Overview of the Filtering Process

Because the LAN Gateway is an efficient user of your WAN, filtering of WAN traffic might not be needed in your network. However, because every IPX frame, NetBIOS datagram, and NetBIOS session initiation frame passes through the filter, the filtering mechanism can be useful in gathering detailed information about network usage without actually filtering.

If the normal mechanisms do not provide enough traffic control, filtering can be used to further restrict WAN traffic. For example, a user filter can exclude specific users from certain resources, or protect a resource from access except from specified users. Filtering can also vary depending on the time of day so that noncritical resources do not use too much WAN bandwidth during high traffic periods.

Use the LAN Gateway configuration tool to set up the LAN Gateway to use a frame filtering program. The LAN Gateway ships a default filter program, AXSFILTR.DLL, or you can write your own filter program. Refer to B.2, “Supplied Filter (AXSFILTR.DLL)” for more information.

B.2 Supplied Filter (AXSFILTR.DLL)

The LAN Gateway ships with a default filter program, AXSFILTR.DLL. You can use the default program or write your own filter program. The source code, AXSFILTR.C, is supplied for reference if you choose to write your own program. This section describes the default filter program and includes descriptions of filter file keywords.

B.2.1 Overview of the Default Filter Program

The LAN Gateway ships with a default filter program, AXSFILTR.DLL, which uses an ASCII text file as input. This text file can specify the following information:

- For an individual NetBIOS LAN, a list of source and distribution NetBIOS names
- For an individual IPX LAN, a list of source and destination IPX addresses
- Optionally, the action to be taken if an incoming frame matches the source and distribution lists

A nonfunctioning filter input file, AXSFILTR.DAT, is shipped with the LAN Gateway software. Before using this input file, replace the names and addresses in the file with names and addresses from your own network.

When the gateway starts, it calls the filter program to initialize it. The specified file name and path of the input file is passed as a parameter on the call. AXSFILTR.DLL reads in and parses the input file. If errors are detected,

AXSFILTR.DLL returns an error message and return code to the gateway. If logging is enabled, the gateway logs the error message in the message log file.

If the input file is valid, the gateway calls the filter on each NetBIOS Name_Query, Status_Query and Datagram, and each IPX frame to determine whether the frame should be passed on to the target region.

Note: Each frame that passes through the filter might degrade the gateway's performance.

The gateway sends, as parameters in the call, pointers to the target gateway region name and to the beginning of the frame. The filter compares the region name to the region name in the filter input file. Then, the filter compares the frame source and destination information to the names and addresses listed for that LAN. If there is a match, the filter sets the return code according to the action specified in the filter input file.

During shutdown, the gateway calls AXSFILTR.DLL; however, it does not pass any parameters except for the action code. The filter cleans up, exits, and passes nothing to the gateway.

B.2.2 Input File Format

The input for the filter input file is given as a set of keywords with corresponding values that follow these syntax rules:

- The valid keywords and parameters are:

  ```
  REGION=value
  ACTION=value
  SNETW=value
  DNETW=value
  SNAME=value
  DNAME=value
  ```

- Keywords are capitalized and entered in full length as shown by the preceding bullet.
- Each keyword is followed by a value used to match names in the frame. The value is enclosed in double quotation marks.
- The keyword and its value must be on the same line of the input file; they cannot span lines.
- The filter input file is comprised of individual sections defined by REGION keywords. The keywords that follow a REGION keyword are associated with that region.
- The value does not need to follow the keyword immediately. All characters between a keyword and its value are ignored (including the asterisk (*)).
- A line is terminated by a new line character (X'0D') or the null character (X'00').
- An asterisk (*) indicates a comment. The comment extends from the asterisk to the end of the current line of the input file. The exception is an asterisk between a keyword and its value or an asterisk within the value.

For a closer description of the keywords see the *LAN Gateway User's Guide*.

B.2.2.1 Example of ASCII Filter Input File Without Errors

```
* Region NEWYORK definitions
REGION="NEWYORK"   ACTION="DISCARD"
SNAME="JOE*"
DNAME="APPL_FILESRV01 !"
SNAME="OEM            " DNAME="*FILESRV*"
* Region CHICAGO definitions
REGION="CHICAGO"   ACTION="FORWARD"
SNAME="REQ*"
* Region SANFRAN definitions
REGION = "SANFRAN"            * no action specified, defaults to discard
DNAME       "*PRTSRV2*"
* Region DALLAS definitions
REGION="DALLAS"    ACTION="DISCARD"
DNETW="3040506A"              * Discard if to this IPX network
SNETW="000000AB.100040005032" * Discard if from this IPX network
```

B.2.2.2 Example of Filter Input File Actions

This example shows the actions taken based on the input in B.2.2.1, "Example of ASCII Filter Input File Without Errors."

Frame Source Name/Address	Frame Destination Name/Address	Target Region	Action
"BILL "	"XXXX_FILESRV02 "	NEWYORK	forward
"JOESMITH "	"XXXX_FILESRV02 "	NEWYORK	discard
"joesmith "	"XXXX_FILESRV02 "	NEWYORK	forward
"SMITHJOE "	"XXXX_FILESRV02 "	NEWYORK	forward
"ANYONE "	"APPL_FILESRV01 !"	NEWYORK	discard
"ANYONE "	"APPL_FILESRV01 @"	NEWYORK	forward
"OEM "	"XXXX_FILESRV01 "	NEWYORK	discard
"OEM01 "	"XXXX_FILESRV01 "	NEWYORK	forward
"OEM "	"XXXX_PRTSRV01 "	NEWYORK	forward
"BILL "	"XXXX_FILESRV02 "	CHICAGO	discard
"REQ "	"XXXX_FILESRV02 "	CHICAGO	forward
"REQ01 "	"XXXX_FILESRV02 "	CHICAGO	forward
"AUSREQ "	"XXXX_FILESRV02 "	CHICAGO	forward
"APPL_REQ01 "	"APPL_FILESRV01 !"	CHICAGO	forward
"OEM "	"XXXX_FILESRV01 "	CHICAGO	discard
"APPL_REQ01 "	"XXXX_PRTSRV20 "	SANFRAN	discard
"OEM "	"XXXX_PRTSRV12 "	SANFRAN	forward
"BILL "	"XXXX_FILESRV02 "	DETROIT	forward
4321ABCD.10005A5A4344	3040506A.100045321ABC	DALLAS	discard
000000AB.100040005032	00001234.000000000001	DALLAS	discard
000000AB.100040005000	00001234.000000000001	DALLAS	forward

B.2.2.3 Example of Common Filter Input File Errors

```
* Keyword not valid
REG = "NEWYORK"         * cannot abbreviate keywords
region = "NEWYORK"      * keywords must be capitalized
"DISCARD"               * keyword must be present

* Region value not valid
REGION = CHARLOTT       * value not enclosed in double quotes
REGION = "CHARLOTTE"    * REGION value longer than 8 characters

* Action value not valid
ACTION = DISCARD         * value not enclosed in double quotes
ACTION = "discard"       * ACTION value must be capitalized
ACTION = "DISC"          * ACTION value cannot be abbreviated
ACTION = "FILTER"        * ACTION value must be "DISCARD" or
                                "FORWARD"

* SNETW value not valid
SNETW = "00192.100024003000"   * IPX network number not valid, must
                                  be 8 characters
SNETW = "345ABCDE."            * Must have node address after "."

* DNETW value not valid
DNETW = "00192.100024003000"   * IPX network number not valid, must
                                  be 8 characters
DNETW = "345ABCDE."            * Must have node address after "."

* General syntax problems
REGION =                 * keyword and its value must be
"NEWYORK"                   on the same line of the input file
* REGION = "NEWYORK"     * this entire line is commented out
```

Appendix C. Loopback Mode

This appendix describes the LAN Gateway loopback mode option. By enabling the LAN Gateway loopback mode, a LAN Gateway can be used as a WAN access node.

In this configuration, the LAN Gateway can connect NetBIOS and IPX applications that run locally on the LAN Gateway workstation to remote LANs over an IP or SNA WAN. The loopback driver, which looks like an IBM token-ring MAC driver to the local application programs, supports the capability of linking NetBIOS, IPX, and gateway frames over the WAN.

This appendix includes the following sections:

C.1 Overview of the Loopback Driver

When local NetBIOS or IPX applications send frames to the loopback driver, the driver converts the NetBIOS or IPX logical link control (LLC) information contained in the frame to LAN Gateway information. The driver then routes the frame back to the gateway. The gateway, the local application program, and OS/2 view the frame as having been sent from a local workstation on a LAN. The loopback driver is configured the same as any IBM token-ring or MAC layer driver.

C.2 Installing and Enabling the Loopback Mode

To install the loopback driver on the local LAN Gateway:

1. Verify that Multiprotocol Transport Services (MPTS) is installed.
2. Verify that the LAN Gateway software is installed.
3. In MPTS, install the loopback driver and set the loopback protocol support and adapter numbers. Specify x:\langw\misc as the source directory, where x:\langw\ is the directory where you installed the LAN Gateway.

   From the LAPS Configuration window in MPTS:

   a. Select **AnyNet LAN Gateway Loopback Driver** from the Network Adapters list.

   b. Click on the **Add** push button under the Network Adapters list.

   c. Select **IBM IEEE 802.2** from the Protocols list.

   d. Click on the **Add** push button under the Protocols list.

   e. Select the appropriate application protocols from the Protocol list. Take either or both of the following actions:

      - If the LAN Gateway supports NetBIOS applications, select **IBM OS/2 NetBIOS** from the Protocols list. Click on the **Add** push button under the Protocols list.

- If the LAN Gateway supports IPX applications, select **IBM NetWare Requester Support** from the Protocols list. Click on the **Add** push button under the Protocols list.

f. Verify the adapter numbers in the Current Configuration list. Modify the adapter numbers as needed by selecting a protocol from the list and click on the **Change Numbers** push button.

g. Click on the **OK** push button.

4. Exit MPTS. The workstation updates the CONFIG.SYS and IBMCOMPROTOCOL.INI files with the loopback driver.
5. To ensure the adapter driver is loaded before it is called by a local application program or the gateway, it is recommended that you edit the device statements in the workstation's CONFIG.SYS file. Move the DEVICE=C:IBMCOMMACSAXSLOOP.OS2 statement before the gateway AXSLANDD.SYS and the NetBIOS or IPX device driver statements.
6. Verify that the LAN gateway adapter number, defined in the global configuration file, is the same adapter number specified for MPTS in Step 3f.
7. Restart the workstation.

   Note: In IPX configurations, the Novell requester issues an error message. Click on the **OK** push button and start the gateway.

8. Use the LAN Gateway configuration tool to set the loopback mode. Use the System Setup function to select the loopback mode on the appropriate window.

C.3 Recommendations for Configuring the LAN Gateway in Loopback Mode

Consider the performance recommendations in Table 20 when configuring the loopback mode.

Table 20. Recommended Settings for Configuring the LAN Gateway in Loopback Mode

Parameter	Recommended Setting
Maximum Number of Link Stations	Set the value to 3, with one NetBIOS workstation available.
Maximum Number of Circuits	Set the value to 6 circuits for each active application on the gateway. If the gateway supports a LAN server, set this value to 6 circuits for each remote user. Reserve several extra circuits if the gateway accesses and acquires adapter status from remote servers.
Maximum Number of Buffers	If gateway links to one or two partner gateways, consider setting this value to less than the default. A value of 100 is recommended. The gateway in loopback mode requires additional memory for local applications.

C.4 Starting the LAN Gateway in Loopback Mode

When the loopback mode is enabled, links to partner gateways must be started before the local NetBIOS or IPX application programs can run successfully. If the links are specified as autolinks, the links are started automatically whenever the gateway starts.

Appendix D. Migration Considerations and Procedures for LTLW and IPX over SNA Gateway

If you are currently using the LAN-to-LAN wide area network (LTLW) program or AnyNet IPX over SNA Gateway, you can migrate existing configurations to the LAN Gateway.

This appendix explains how to migrate your existing configurations to the LAN Gateway and includes the following sections:

- D.1, “Migrating LTLW Configurations”
- D.2, “Migrating IPX over SNA Gateway Configurations”
- D.3, “Converting Existing Configuration Files to the LAN Gateway Format” on page 312

D.1 Migrating LTLW Configurations

If the LTLW program is currently installed on your system, use the standard installation procedure for installing the LAN Gateway feature of Communications Server.

When the LAN Gateway is installed, the installation program comments out any LTLW device drivers in your CONFIG.SYS file. No other file changes are made during the installation process, and LTLW product or configuration files are not deleted.

Note: If the LAN Gateway feature is not installed in the same directory or on the same machine as your existing configuration files, copy your .SCF files from the existing directory to the directory where the LAN Gateway feature is installed.

D.2 Migrating IPX over SNA Gateway Configurations

If the IPX over SNA Gateway is currently installed on your system, delete IPX over SNA files before installing the LAN Gateway feature of Communications Server:

1. To delete IPX over SNA files using the Communications Server CD, go to the SERVERLANGW subdirectory on the CD and type `install`.
2. The LAN Gateway README file is displayed. To continue, click on the **Continue** push button.
3. In the Installation and Maintenance window, select **View** on the menu bar and click on the **Installed Products** option to view the list of installed products.
4. Select the **AnyNet Workstation Setup** option.
5. Select **Action** on the menu bar and click on the **Delete** option.
6. In the Delete window, select all of the IPX over SNA components that appear in the list.

   Possible components are:

   - IPX over SNA Gateway for OS/2 Product

- IPX over SNA Gateway for OS/2 Online Documentation
- IPX over SNA Gateway for OS/2 Printable Documentation

7. Click on the **Delete** push button to delete the selected components. After the Delete action completes, the Installation and Maintenance window displays.
8. After deleting the IPX over SNA Gateway files, install the LAN Gateway.

   You can begin the LAN Gateway installation procedure by selecting **View** on the menu bar and clicking on the **Current Catalog** option. Sample LAN Gateway installation procedures can be found later in this section.

Note: If the LAN Gateway is not installed in the same directory or on the same machine as your existing configuration files, copy your .SCF files from the existing directory to the directory where the LAN Gateway is installed.

D.3 Converting Existing Configuration Files to the LAN Gateway Format

Before you can use existing LTLW or IPX over SNA Gateway configuration files with the LAN Gateway, you must convert them to the LAN Gateway format. The configuration program will do most of this automatically for you.

To complete the conversion process, use the LAN Gateway configuration tool:

1. Start the LAN Gateway configuration tool using either of the following methods:
   - Enter `axsconf` at an OS/2 command prompt.
   - Double-click on the **LAN Gateway Configuration Tool** icon in the LAN Gateway folder.
2. Enter the name of the existing configuration file and click on the **OK** push button.
3. To use the new configuration file extension of .RSP, click on the **File** menu bar option and select the **Save as...** option to save the configuration using this file extension.
4. Click on the **Setup System** push button.
5. Enter all system setup values.
6. Save the configuration.

Note: After finishing the migration process and verifying that you have a working configuration, you might want to manually delete the LTLW product files in order to free disk space.

Appendix E. WAN TCP Port Number Used by LAN Gateways

All stream socket calls transferred between LAN Gateways over IP WANs are bound to TCP port number 1491. This number was assigned to AnyNet LAN Gateway by the Internet Assigned Numbers Authority (IANA).

Important

All LAN Gateways on the same IP WAN that need to communicate with each other must use the same TCP port number. It is recommended that your WAN configuration use the assigned TCP port number of 1491. If you change the port number for one LAN Gateway, you must change the port number for all other LAN Gateways attached to that IP WAN.

It is strongly recommended that you do not change this port number.

If another application requires port 1491, you might need to reconfigure the port used by the LAN Gateways. This section explains how to change the TCP port number, if necessary.

LAN Gateway uses the SERVICES file to set the port number. This file is provided by IBM TCP/IP for OS/2. The SET ETC statement of your CONFIG.SYS file points to the directory where the SERVICES file is located. These files are usually in the C:TCPIPETC or the C:MPTNETC directory.

The port number is set by the SERVICES file line entry shown in Figure 260.

```
     anynetgateway    1491/tcp       # comment
```

Figure 260. Example of the Port Number Entry in the SERVICES File

If the SERVICES file does not contain this line entry, you can create one. The comment is optional.

After starting the LAN Gateway, you can use the TCP/IP netstat -s command to verify that port 1491 is active. The command output looks similar to the example in Figure 261.

```
                  FOREIGN    LOCAL     FOREIGN
SOCK     TYPE       PORT     PORT       HOSTSTATE
==== ========   =========   ======   =========   =======
28     STREAM           0     1491     0.0.0.0    LISTEN
17      DGRAM           0        0     0.0.0.0    UDP
8      STREAM           0     1024     0.0.0.0    LISTEN
```

Figure 261. Example of netstat -s Output Showing an Active TCP Port

Successful output shows a stream socket bound to port 1491 that is in the LISTEN state.

To modify the port number:

1. Edit the SERVICES file.

2. Change 1491 to the new port number.
3. Reboot the workstation.

For example, to use port number 1703, change the line entry in Figure 260 on page 313 to the entry shown in Figure 262.

```
anynetgateway    1703/tcp       # AnyNet LAN Gateway changed to 1703
```

Figure 262. Example of a Revised Port Number Entry in the SERVICES File

To verify the change, use the TCP/IP netstat -s command. You should see a stream socket bound to port number 1703 that is in the LISTEN state.

If two LAN Gateways are using different TCP port numbers, they cannot connect to each other. Both LAN Gateways issue message AXS0514E, indicating that the other gateway is not running.

Appendix F. VTAM Line Description for Multiple PU over a Single SDLC Line

```
LO7161   LINE ADDRESS=(161,HALF),   HALF DUPLEX                          *
               ANS=CONTINUE,       DON'T BREAK CROSS DOMAIN SESSIONS     *
               CLOCKNG=EXT,        DTE CABLE 7837395 ATTACHED            *
               ISTATUS=ACTIVE,                                           *
               NRZI=YES,           NRZI                                  *
               LSPRI=PU,           TP PRIORITY ALSO FOR BNN              *
               LPDATS=LPDA2,                                             *
               DUPLEX=FULL,        REQUEST TO SEND ALWAYS UP             *
               ETRATIO=30,         DEFAULT                               *
               SERVLIM=10,                                               *
               SPEED=9600,         NPA AND SCANNER USE       ####        *
               SRT=(,64)
*              STATOPT='CM/2 SERVER')                                     05950000
         SERVICE ORDER=(PO7161A,PO7161B,PO7161C,PO7161D),MAXLIST=9
PO7161A  PU ADDR=C1,                 CLUSTER ADDRESS = 01                *
               RETRIES=(,4,5),       7 RETRY PER SECOND FOR 5 TIMES      *
               MAXDATA=521,          MAXIMUM AMOUNT OF DATA              *
               MAXOUT=7,             MAX SDLC FRAMES BEFORE RESPONSE     *
               PACING=0,             PACING SET BY BIND IMAGE            *
               PASSLIM=7,                                                *
               PUTYPE=2,                                                 *
               DISCNT=(NO),          (V) VTAM                            *
               ISTATUS=ACTIVE,       (V) VTAM                            *
               SSCPFM=USSSCS,        (V) VTAM                            *
               USSTAB=US327X,        (V) VTAM                            *
               VPACING=0             (V) VTAM
*              STATOPT='CM/2',NOACTY)                                     06150000
TO7161A2 LU LOCADDR=2,               FIRST LU MUST BE LOCADDR=2          *
               MODETAB=MODEVR,DLOGMOD=VR03270,                           *
               ISTATUS=ACTIVE        (V) VTAM
TO7161A3 LU LOCADDR=3,                                                   *
               MODETAB=MODEVR,DLOGMOD=VR03270,                           *
               ISTATUS=ACTIVE        (V) VTAM
TO7161A4 LU LOCADDR=4,                                                   *
               MODETAB=MODEVR,DLOGMOD=VR03270,                           *
               ISTATUS=ACTIVE        (V) VTAM
TO7161A5 LU LOCADDR=5,                                                   *
               MODETAB=MODEVR,DLOGMOD=VR03270,                           *
               ISTATUS=ACTIVE        (V) VTAM
*                                                                        06280000
PO7161B  PU ADDR=C2,                 CLUSTER ADDRESS = 01                *
               RETRIES=(,4,5),       7 RETRY PER SECOND FOR 5 TIMES      *
               MAXDATA=521,          MAXIMUM AMOUNT OF DATA              *
               MAXOUT=7,             MAX SDLC FRAMES BEFORE RESPONSE     *
               PACING=0,             PACING SET BY BIND IMAGE            *
               PASSLIM=7,                                                *
               PUTYPE=2,                                                 *
               DISCNT=(NO),          (V) VTAM                            *
               ISTATUS=ACTIVE,       (V) VTAM                            *
               SSCPFM=USSSCS,        (V) VTAM                            *
               USSTAB=US327X,        (V) VTAM                            *
               VPACING=0             (V) VTAM
*              STATOPT='CM/2',NOACTY)                                     06410000
TO7161B2 LU LOCADDR=2,               FIRST LU MUST BE LOCADDR=2          *
```

```
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161B3 LU LOCADDR=3,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161B4 LU LOCADDR=4,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161B5 LU LOCADDR=5,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
*                                                                        06540000
*                                                                        06550000
P07161C  PU ADDR=C3,               CLUSTER ADDRESS = 01                 *
               RETRIES=(,4,5),     7 RETRY PER SECOND FOR 5 TIMES       *
               MAXDATA=521,        MAXIMUM AMOUNT OF DATA               *
               MAXOUT=7,           MAX SDLC FRAMES BEFORE RESPONSE      *
               PACING=0,           PACING SET BY BIND IMAGE             *
               PASSLIM=7,                                               *
               PUTYPE=2,                                                *
               DISCNT=(NO),        (V) VTAM                             *
               ISTATUS=ACTIVE,     (V) VTAM                             *
               SSCPFM=USSSCS,      (V) VTAM                             *
               USSTAB=US327X,      (V) VTAM                             *
               VPACING=0           (V) VTAM
*              STATOPT='CM/2',NOACTY)                                    06680000
T07161C2 LU LOCADDR=2,             FIRST LU MUST BE LOCADDR=2           *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161C3 LU LOCADDR=3,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161C4 LU LOCADDR=4,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161C5 LU LOCADDR=5,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
*                                                                        06810000
*                                                                        06820000
P07161D  PU ADDR=C4,               CLUSTER ADDRESS = 01                 *
               RETRIES=(,4,5),     7 RETRY PER SECOND FOR 5 TIMES       *
               MAXDATA=521,        MAXIMUM AMOUNT OF DATA               *
               MAXOUT=7,           MAX SDLC FRAMES BEFORE RESPONSE      *
               PACING=0,           PACING SET BY BIND IMAGE             *
               PASSLIM=7,                                               *
               PUTYPE=2,                                                *
               DISCNT=(NO),        (V) VTAM                             *
               ISTATUS=ACTIVE,     (V) VTAM                             *
               SSCPFM=USSSCS,      (V) VTAM                             *
               USSTAB=US327X,      (V) VTAM                             *
               VPACING=0           (V) VTAM
*              STATOPT='CM/2',NOACTY)                                    06950000
T07161D2 LU LOCADDR=2,             FIRST LU MUST BE LOCADDR=2           *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
               ISTATUS=ACTIVE      (V) VTAM
T07161D3 LU LOCADDR=3,                                                  *
               MODETAB=MODEVR,DLOGMOD=VR03270,                          *
```

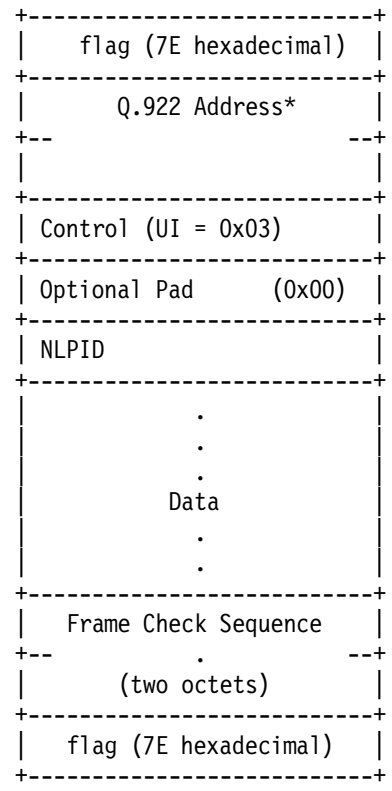
```
               ISTATUS=ACTIVE      (V) VTAM
T07161D4 LU LOCADDR=4,                                                 *
               MODETAB=MODEVR,DLOGMOD=VR03270,                         *
               ISTATUS=ACTIVE      (V) VTAM
T07161D5 LU LOCADDR=5,                                                 *
               MODETAB=MODEVR,DLOGMOD=VR03270,                         *
               ISTATUS=ACTIVE      (V) VTAM
*                                                                       07080000
```

Appendix G. RFC 1490 Extract

This appendix is an extract from the original RFC 1490. We only took some part of it. For formal references please refer to the complete version of RFC 1490.

3. Frame Format

All protocols must encapsulate their packets within a Q.922 Annex A frame (1,2). Additionally, frames shall contain information necessary to identify the protocol carried within the protocol data unit (PDU), thus allowing the receiver to properly process the incoming packet. The format shall be as follows:



* Q.922 addresses, as presently defined, are two octets and contain a 10-bit DLCI. In some networks Q.922 addresses may optionally be increased to three or four octets.

The control field is the Q.922 control field. The UI (0x03) value is used unless it is negotiated otherwise. The use of XID (0xAF or 0xBF) is permitted and is discussed later.

The pad field is used to align the remainder of the frame to a two octet boundary. There may be zero or one pad octet within the pad field and, if present, must have a value of zero.

The Network Level Protocol ID (NLPID) field is administered by ISO and CCITT. It contains values for many different protocols including IP, CLNP and IEEE Subnetwork Access Protocol (SNAP)(10). This field tells the receiver what encapsulation or what protocol follows. Values for this field are defined in ISO/IEC TR 9577 (3). A NLPID value of 0x00 is defined within ISO/IEC TR 9577 as the Null Network Layer or Inactive Set. Since it cannot be distinguished from a pad

field, and because it has no significance within the context of this encapsulation scheme, a NLPID value of 0x00 is invalid under the Frame Relay encapsulation. The Appendix contains a list of some of the more commonly used NLPID values.

There is no commonly implemented minimum maximum frame size for Frame Relay. A network must, however, support at least a 262 octet maximum. Generally, the maximum will be greater than or equal to 1600 octets, but each Frame Relay provider will specify an appropriate value for its network. A Frame Relay DTE, therefore, must allow the maximum acceptable frame size to be configurable.

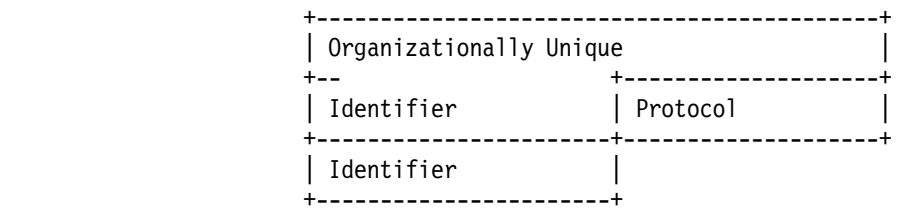
The minimum frame size allowed for Frame Relay is five octets between the opening and closing flags assuming a two octet Q.922 address field. This minimum increases to six octets for three octet Q.922 address and seven octets for the four octet Q.922 address format.

4. Interconnect Issues

There are two basic types of data packets that travel within the Frame Relay network: routed packets and bridged packets. These packets have distinct formats and therefore, must contain an indicator that the destination may use to correctly interpret the contents of the frame. This indicator is embedded within the NLPID and SNAP header information.

For those protocols that do not have a NLPID already assigned, it is necessary to provide a mechanism to allow easy protocol identification. There is a NLPID value defined indicating the presence of a SNAP header.

A SNAP header is of the form:



All stations must be able to accept and properly interpret both the NLPID encapsulation and the SNAP header encapsulation for a routed packet.

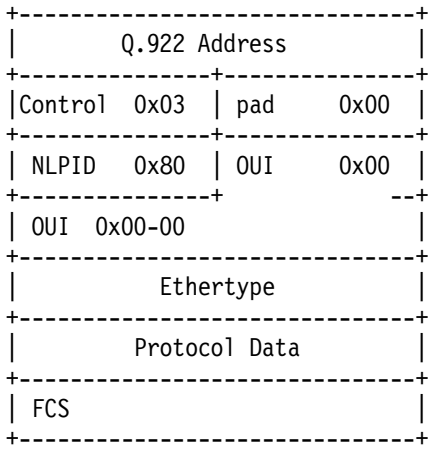
The three-octet Organizationally Unique Identifier (OUI) identifies an organization which administers the meaning of the Protocol Identifier (PID) which follows. Together they identify a distinct protocol. Note that OUI 0x00-00-00 specifies that the following PID is an Ethertype.

4.1. Routed Frames

Some protocols will have an assigned NLPID, but because the NLPID numbering space is so limited, not all protocols have specific NLPID values assigned to them. When packets of such protocols are routed over Frame Relay networks, they are sent using the NLPID 0x80 (which indicates a SNAP follows) followed by SNAP. If the protocol has an Ethertype assigned, the OUI is 0x00-00-00 (which indicates an Ethertype follows), and PID is the Ethertype of the protocol in use.

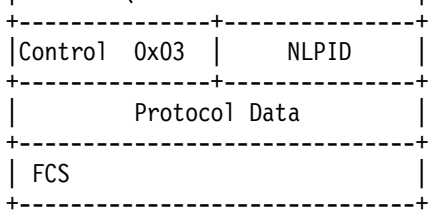
There will be one pad octet to align the protocol data on a two octet boundary as shown below.

Format of Routed Frames
with Ethertypes



In the few cases when a protocol has an assigned NLPID (see appendix), 48 bits can be saved using the format below:

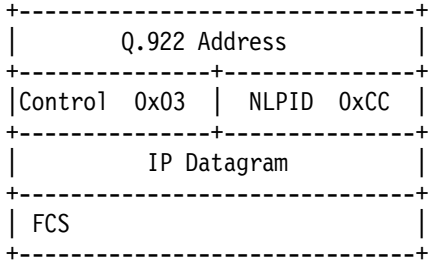
Format of Routed NLPID Protocol



The NLPID encapsulation does not require a pad octet for alignment, so none is permitted.

In the case of ISO protocols, the NLPID is considered to be the first octet of the protocol data. It is unnecessary to repeat the NLPID in this case. The single octet serves both as the demultiplexing value and as part of the protocol data (refer to "Other Protocols over Frame Relay for more details). Other protocols, such as IP, have a NLPID defined (0xCC), but it is not part of the protocol itself.

Format of Routed IP Datagram



4.2. Bridged Frames

The second type of Frame Relay traffic is bridged packets. These packets are encapsulated using the NLPID value of 0x80 indicating SNAP. As with other SNAP encapsulated protocols, there will be one pad octet to align the data portion of the encapsulated frame. The SNAP header which follows the NLPID identifies the format of the bridged packet. The OUI value used for this encapsulation is the 802.1 organization code 0x00-80-C2. The PID portion of the SNAP header (the two bytes immediately following the OUI) specifies the form of the MAC header, which immediately follows the SNAP header. Additionally, the PID indicates whether the original FCS is preserved within the bridged frame.

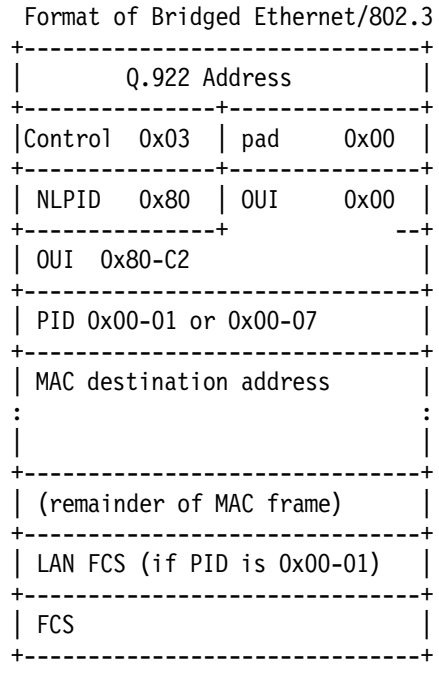
The 802.1 organization has reserved the following values to be used with Frame Relay:

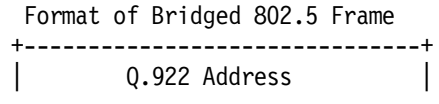
PID Values for OUI 0x00-80-C2

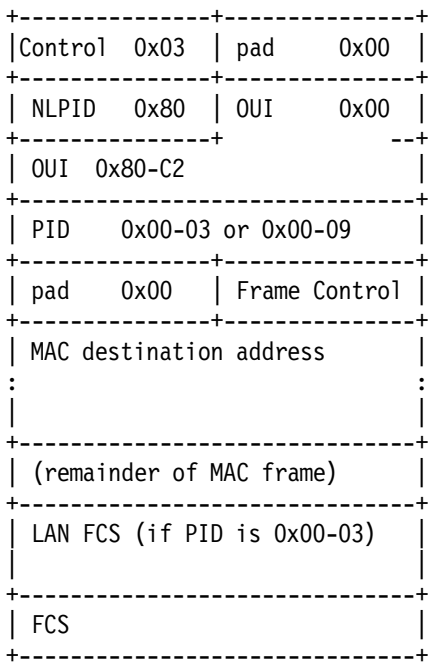
with preserved FCS	w/o preserved FCS	Media
0x00-01	0x00-07	802.3/Ethernet
0x00-02	0x00-08	802.4
0x00-03	0x00-09	802.5
0x00-04	0x00-0A	FDDI
	0x00-0B	802.6

In addition, the PID value 0x00-0E, when used with OUI 0x00-80-C2, identifies bridged protocol data units (BPDUs) as defined by 802.1(d) or 802.1(g) (12).

A packet bridged over Frame Relay will, therefore, have one of the following formats:







4. Data Link Layer Parameter Negotiation

Frame Relay stations may choose to support the Exchange Identification (XID) specified in Appendix III of Q.922 (1). This XID exchange allows the following parameters to be negotiated at the initialization of a Frame Relay circuit: maximum frame size N201, retransmission timer T200, and the maximum number of outstanding Information (I) frames K.

A station may indicate its unwillingness to support acknowledged mode multiple frame operation by specifying a value of zero for the maximum window size, K.

If this exchange is not used, these values must be statically configured by mutual agreement of Data Link Connection (DLC) endpoints, or must be defaulted to the values specified in Section 5.9 of Q.922:

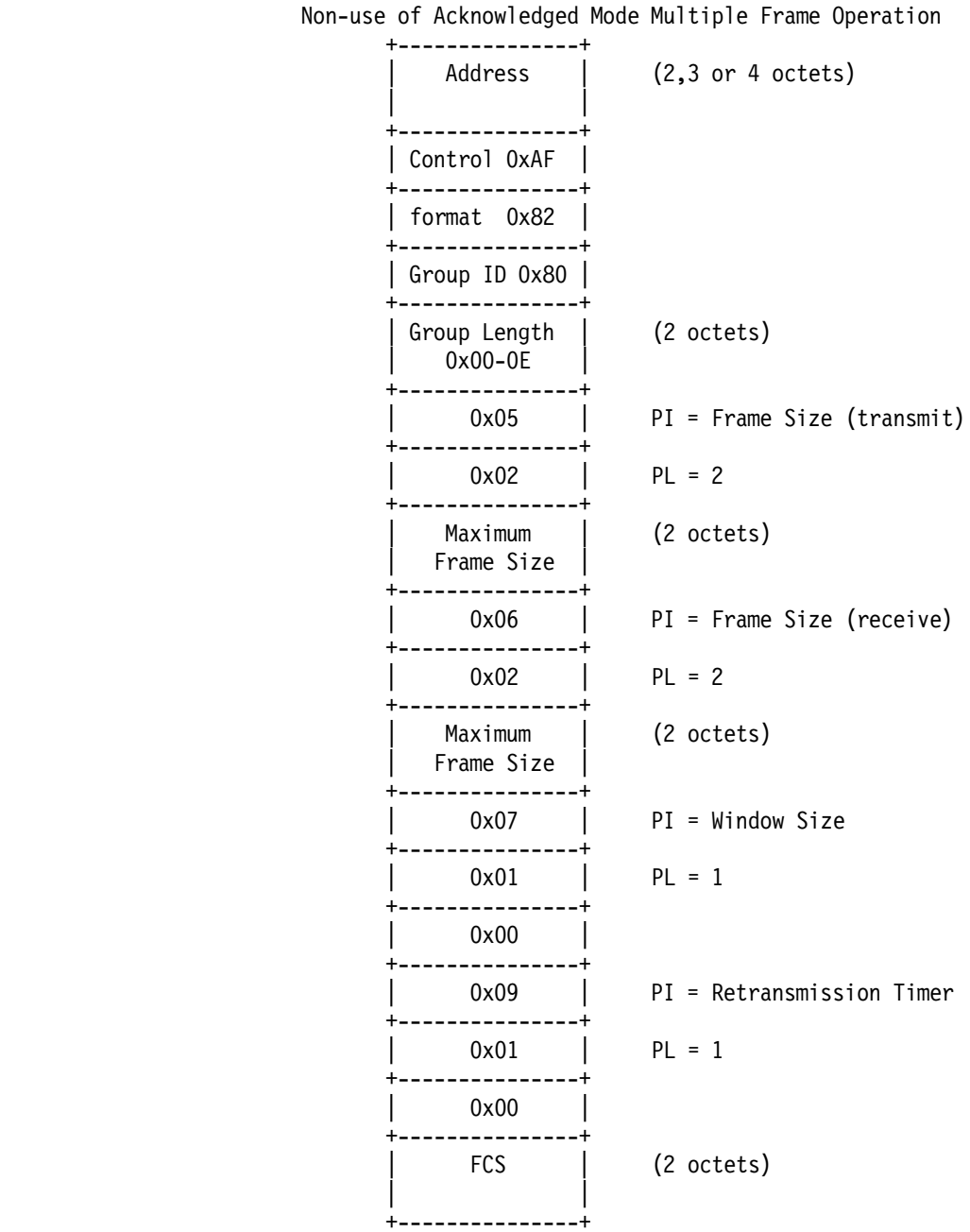
N201: 260 octets

K: 3 for a 16 Kbps link,
7 for a 64 Kbps link,
32 for a 384 Kbps link,
40 for a 1.536 Mbps or above link

T200: 1.5 seconds (see Q.922 for further details)

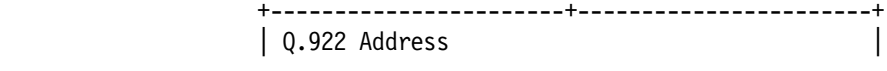
If a station supporting XID receives an XID frame, it shall respond with an XID response. In processing an XID, if the remote maximum frame size is smaller than the local maximum, the local system shall reduce the maximum size it uses over this DLC to the remotely specified value. Note that this shall be done before generating a response XID.

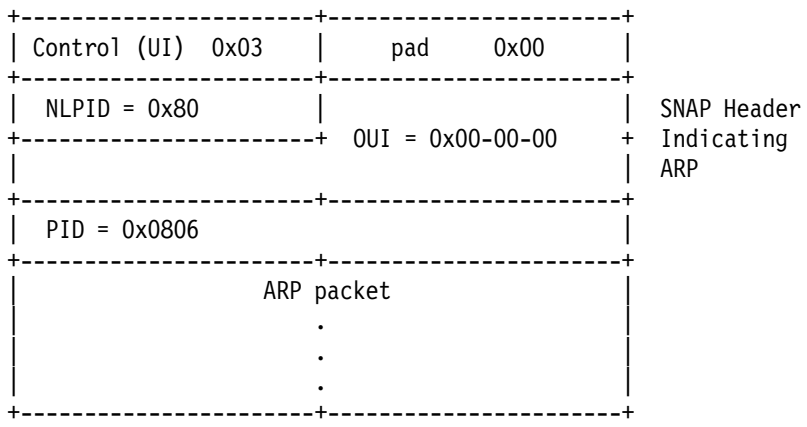
The following diagram describes the use of XID to specify non-use of acknowledged mode multiple frame operation.



7. Address Resolution

There are situations in which a Frame Relay station may wish to dynamically resolve a protocol address. Address resolution may be accomplished using the standard Address Resolution Protocol (ARP) (6) encapsulated within a SNAP encoded Frame Relay packet as follows:





Where the ARP packet has the following format and values:

Data:

ar$hrd	16 bits	Hardware type
ar$pro	16 bits	Protocol type
ar$hln	8 bits	Octet length of hardware address (n)
ar$pln	8 bits	Octet length of protocol address (m)
ar$op	16 bits	Operation code (request or reply)
ar$sha	noctets	source hardware address
ar$spa	moctets	source protocol address
ar$tha	noctets	target hardware address
ar$tpa	moctets	target protocol address

ar$hrd - assigned to Frame Relay is 15 decimal (0x000F) (7).

ar$pro - see assigned numbers for protocol ID number for the protocol using ARP. (IP is 0x0800).

ar$hln - length in bytes of the address field (2, 3, or 4)

ar$pln - protocol address length is dependent on the protocol (ar$pro) (for IP ar$pln is 4).

ar$op - 1 for request and 2 for reply.

ar$sha - Q.922 source hardware address, with C/R, FECN, BECN, and DE set to zero.

ar$tha - Q.922 target hardware address, with C/R, FECN, BECN, and DE set to zero.

Because DLCIs within most Frame Relay networks have only local significance, an end station will not have a specific DLCI assigned to itself. Therefore, such a station does not have an address to put into the ARP request or reply. Fortunately, the Frame Relay network does provide a method for obtaining the correct DLCIs. The solution proposed for the locally addressed Frame Relay network below will work equally well for a network where DLCIs have global significance.

The DLCI carried within the Frame Relay header is modified as it traverses the network. When the packet arrives at its destination,

the DLCI has been set to the value that, from the standpoint of the receiving station, corresponds to the sending station. For example, in figure 1 below, if station A were to send a message to station B, it would place DLCI 50 in the Frame Relay header. When station B received this message, however, the DLCI would have been modified by the network and would appear to B as DLCI 70.

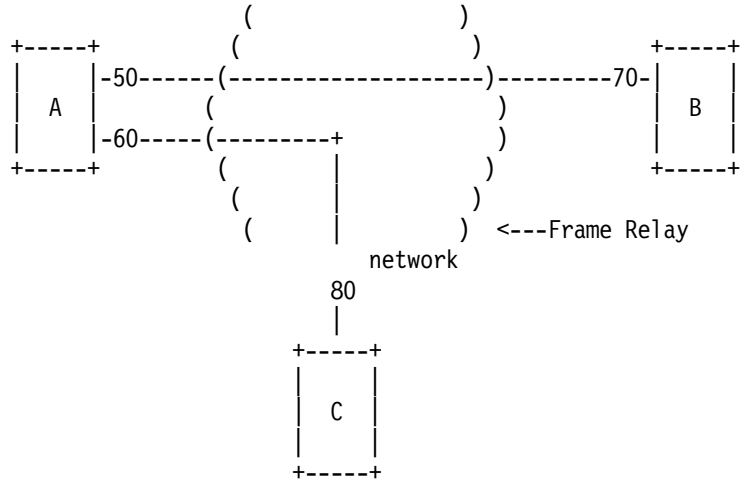


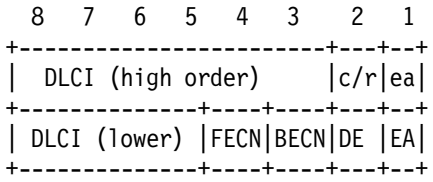
Figure 1

Lines between stations represent data link connections (DLCs). The numbers indicate the local DLCI associated with each connection.

DLCI to Q.922 Address Table for Figure 1

DLCI (decimal)	Q.922 address (hex)
50	0x0C21
60	0x0CC1
70	0x1061
80	0x1401

If you know about frame relay, you should understand the correlation between DLCI and Q.922 address. For the uninitiated, the translation between DLCI and Q.922 address is based on a two byte address length using the Q.922 encoding format. The format is:



For ARP and its variants, the FECN, BECN, C/R and DE bits are assumed to be 0.

When an ARP message reaches a destination, all hardware addresses will be invalid. The address found in the frame header will, however, be correct. Though it does violate the purity of layering, Frame Relay may use the address in the header as the sender hardware

address. It should also be noted that the target hardware address, in both ARP request and reply, will also be invalid. This should not cause problems since ARP does not rely on these fields and in fact, an implementation may zero fill or ignore the target hardware address field entirely.

As an example of how this address replacement scheme may work, refer to figure 1. If station A (protocol address pA) wished to resolve the address of station B (protocol address pB), it would format an ARP request with the following values:

```
ARP request from A
  ar$op     1 (request)
  ar$sha    unknown
  ar$spa    pA
  ar$tha    undefined
  ar$tpa    pB
```

Because station A will not have a source address associated with it, the source hardware address field is not valid. Therefore, when the ARP packet is received, it must extract the correct address from the Frame Relay header and place it in the source hardware address field. This way, the ARP request from A will become:

```
ARP request from A as modified by B
  ar$op     1 (request)
  ar$sha    0x1061 (DLCI 70) from Frame Relay header
  ar$spa    pA
  ar$tha    undefined
  ar$tpa    pB
```

Station B's ARP will then be able to store station A's protocol address and Q.922 address association correctly. Next, station B will form a reply message. Many implementations simply place the source addresses from the ARP request into the target addresses and then fills in the source addresses with its addresses. In this case, the ARP response would be:

```
ARP response from B
  ar$op     2 (response)
  ar$sha    unknown
  ar$spa    pB
  ar$tha    0x1061 (DLCI 70)
  ar$tpa    pA
```

Again, the source hardware address is unknown and when the request is received, station A will extract the address from the Frame Relay header and place it in the source hardware address field. Therefore, the response will become:

```
ARP response from B as modified by A
  ar$op     2 (response)
  ar$sha    0x0C21 (DLCI 50)
  ar$spa    pB
  ar$tha    0x1061 (DLCI 70)
  ar$tpa    pA
```

Station A will now correctly recognize station B having protocol address pB associated with Q.922 address 0x0C21 (DLCI 50).

Reverse ARP (RARP) (8) will work in exactly the same way. Still using figure 1, if we assume station C is an address server, the following RARP exchanges will occur:

```
RARP request from A              RARP request as modified by C
   ar$op  3 (RARP request)          ar$op  3  (RARP request)
   ar$sha unknown                   ar$sha 0x1401 (DLCI 80)
   ar$spa undefined                 ar$spa undefined
   ar$tha 0x0CC1 (DLCI 60)          ar$tha 0x0CC1 (DLCI 60)
   ar$tpa pC                        ar$tpa pC
```

Station C will then look up the protocol address corresponding to Q.922 address 0x1401 (DLCI 80) and send the RARP response.

```
RARP response from C             RARP response as modified by A
        ar$op  4  (RARP response)        ar$op  4 (RARP response)
        ar$sha unknown                   ar$sha 0x0CC1 (DLCI 60)
        ar$spa pC                        ar$spa pC
        ar$tha 0x1401 (DLCI 80)          ar$tha 0x1401 (DLCI 80)
        ar$tpa pA                        ar$tpa pA
```

This means that the Frame Relay interface must only intervene in the processing of incoming packets.

In the absence of suitable multicast, ARP may still be implemented. To do this, the end station simply sends a copy of the ARP request through each relevant DLC, thereby simulating a broadcast.

The use of multicast addresses in a Frame Relay environment is presently under study by Frame Relay providers. At such time that the issues surrounding multicasting are resolved, multicast addressing may become useful in sending ARP requests and other "broadcast" messages.

Because of the inefficiencies of broadcasting in a Frame Relay environment, a new address resolution variation was developed. It is called Inverse ARP (11) and describes a method for resolving a protocol address when the hardware address is already known. In Frame Relay's case, the known hardware address is the DLCI. Using Inverse ARP for Frame Relay follows the same pattern as ARP and RARP use. That is the source hardware address is inserted at the receiving station.

In our example, station A may use Inverse ARP to discover the protocol address of the station associated with its DLCI 50. The Inverse ARP request would be as follows:

```
InARP Request from A (DLCI 50)
ar$op   8       (InARP request)
ar$sha  unknown
ar$spa  pA
ar$tha  0x0C21  (DLCI 50)
ar$tpa  unknown
```

When Station B receives this packet, it will modify the source

hardware address with the Q.922 address from the Frame Relay header. This way, the InARP request from A will become:

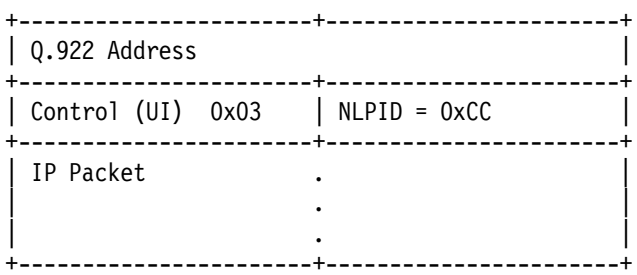
```
ar$op   8       (InARP request)
ar$sha  0x1061
ar$spa  pA
ar$tha  0x0C21
ar$tpa  unknown.
```

Station B will format an Inverse ARP response and send it to station A as it would for any ARP message.

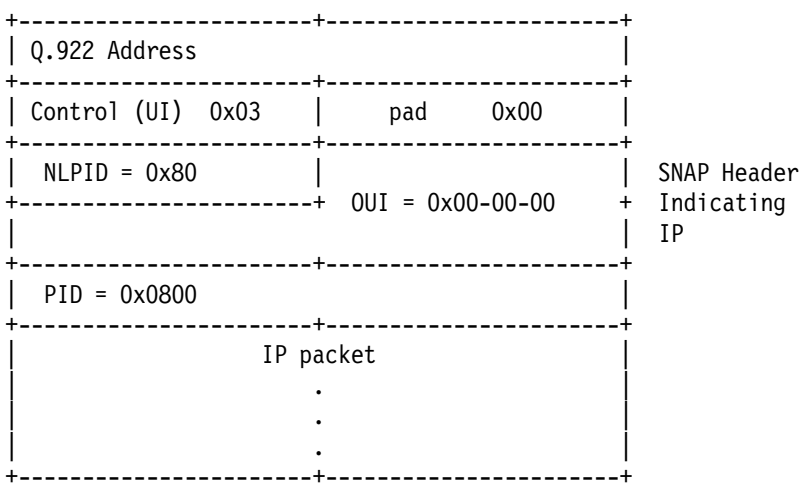
8. IP over Frame Relay

Internet Protocol (9) (IP) datagrams sent over a Frame Relay network conform to the encapsulation described previously. Within this context, IP could be encapsulated in two different ways.

1. NLPID value indicating IP



2. NLPID value indicating SNAP



Although both of these encapsulations are supported under the given definitions, it is advantageous to select only one method as the appropriate mechanism for encapsulating IP data. Therefore, IP data shall be encapsulated using the NLPID value of 0xCC indicating IP as shown in option 1 above. This (option 1) is more efficient in transmission (48 fewer bits), and is consistent with the encapsulation of IP in X.25.

11. Appendix A

List of Commonly Used NLPIDs

0x00	Null Network Layer or Inactive Set (not used with Frame Relay)
0x80	SNAP
0x81	ISO CLNP
0x82	ISO ESIS
0x83	ISO ISIS
0xCC	Internet IP

List of PIDs of OUI 00-80-C2

with preserved FCS	w/o preserved FCS	Media
0x00-01	0x00-07	802.3/Ethernet
0x00-02	0x00-08	802.4
0x00-03	0x00-09	802.5
0x00-04	0x00-0A	FDDI
	0x00-0B	802.6
	0x00-0D	Fragments
	0x00-0E	BPDUs as defined by 802.1(d) or 802.1(g)(12).

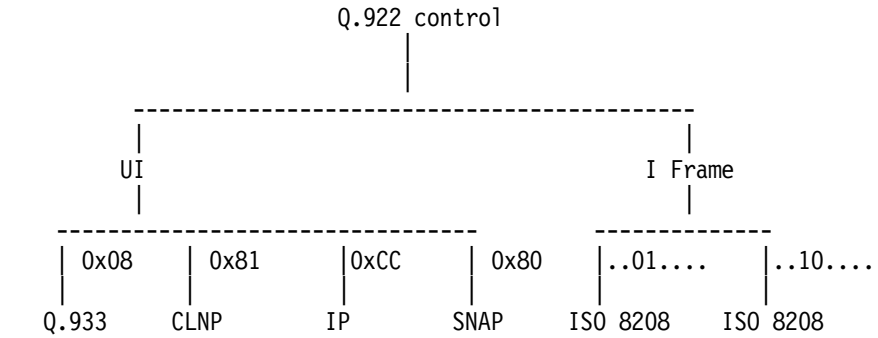
12. Appendix B - Connection Oriented procedures.

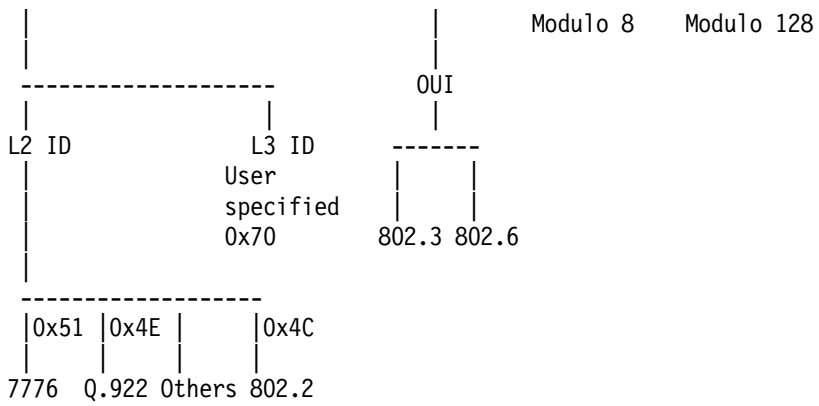
This appendix contains additional information and instructions for using CCITT Q.933 and other CCITT standards for encapsulating data over frame relay. The information contained here is similar (and in some cases identical) to that found in Annex F to ANSI T1.617 written by Rao Cherukuri of IBM. The authoritative source for this information is in Annex F and is repeated here only for convenience.

The Network Level Protocol ID (NLPID) field is administered by ISO and CCITT. It contains values for many different protocols including IP, CLNP (ISO 8473) CCITT Q.933, and ISO 8208. A figure summarizing a generic encapsulation technique over frame relay networks follows. The scheme's flexibility consists in the identification of multiple alternative to identify different protocols used either by

- end-to-end systems or
- LAN to LAN bride and routers or
- a combination of the above.

over frame relay networks.





For those protocols which do not have a NLPID assigned or do not have a SNAP encapsulation, the NLPID value of 0x08, indicating CCITT Recommendation Q.933 should be used. The four octets following the NLPID include both layer 2 and layer 3 protocol identification. The code points for most protocols are currently defined in ANSI T1.617 low layer compatibility information element. There is also an escape for defining non-standard protocols.

Format of Other Protocols using Q.933 NLPID

Q.922 Address	
Control 0x03	NLPID 0x08
L2 Protocol ID	
octet 1	octet 2
L3 Protocol ID	
octet 2	octet 2
Protocol Data	
FCS	

ISO 8802/2 with user specified layer 3

Q.922 Address	
Control 0x03	NLPID 0x08
802/2 0x4C	0x80
User Spec. 0x70	Note 1
DSAP	SSAP
Control (Note 2)	
Remainder of PDU	
FCS	

+------------------------------+

Note 1: Indicates the code point for user specified layer 3 protocol.

Note 2: Control field is two octets for I-format and S-format frames (see 88002/2)

13. References

(1) International Telegraph and Telephone Consultative Committee, "ISDN Data Link Layer Specification for Frame Mode Bearer Services", CCITT Recommendation Q.922, 19 April 1991.

(2) American National Standard For Telecommunications - Integrated Services Digital Network - Core Aspects of Frame Protocol for Use with Frame Relay Bearer Service, ANSI T1.618-1991, 18 June 1991.

(3) Information technology - Telecommunications and Information Exchange between systems - Protocol Identification in the Network Layer, ISO/IEC TR 9577: 1990 (E) 1990-10-15.

(4) Baker, F., Editor, "Point to Point Protocol Extensions for Bridging", RFC 1220, ACC, April 1991.

(5) International Standard, Information Processing Systems - Local Area Networks - Logical Link Control, ISO 8802-2: 1989 (E), IEEE Std 802.2-1989, 1989-12-31.

(6) Plummer, D., "An Ethernet Address Resolution Protocol - or - Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, RFC 826, MIT, November 1982.

(7) Reynolds, J. and J. Postel, "Assigned Numbers", STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.

(8) Finlayson, R., Mann, R., Mogul, J., and M. Theimer, "A Reverse Address Resolution Protocol", STD 38, RFC 903, Stanford University, June 1984.

(9) Postel, J. and Reynolds, J., "A Standard for the Transmission of IP Datagrams over IEEE 802 Networks", RFC 1042, USC/Information Sciences Institute, February 1988.

(10) IEEE, "IEEE Standard for Local and Metropolitan Area Networks: Overview and architecture", IEEE Standards 802-1990.

(11) Bradley, T., and C. Brown, "Inverse Address Resolution Protocol", RFC 1293, Wellfleet Communications, Inc., January 1992.

(12) IEEE, "IEEE Standard for Local and Metropolitan Networks: Media Access Control (MAC) Bridges", IEEE Standard 802.1D-1990.

(13) PROJECT 802 - LOCAL AND METROPOLITAN AREA NETWORKS, Draft Standard 802.1G: Remote MAC Bridging, Draft 6, October 12, 1992.

Appendix H. Special Notices

This publication is intended to help system engineers, system planners, system programmers and network administrators who need to understand and provide the new architectures and functions implemented in the new release of IBM Communications Server 4.1. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM Communications Server 4.1. See the PUBLICATIONS section of the IBM Programming Announcement for IBM Communications Server 4.1 for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (″vendor″) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX	AnyNet
APPN	AS/400
BookManager	CICS
Client Access/400	Client Access
Extended Services	FFST/2
IBM	IMS
InfoWindow	LAN Distance
LANDP	Micro Channel
NetView	Nways
Operating System/2	OS/2
Personal System/2	Portmaster
RXR/2	VTAM
WIN-OS/2	400

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Inc.

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Attachmate	Attachmate Corporation
HyperACCESS	Hilgraeve Incorporated
Internetwork Packet Exchange	Novell, Incorporated
IPX	Novell, Incorporated
NDIS	3Com Corporation and Microsoft Corporation
NetWare	Novell, Incorporated
Novell	Novell, Incorporated
PostScript	Adobe Systems, Incorporated
RUMBA	Wall Data Incorporated
Visual Basic	Microsoft Corporation
Wellfleet	Wellfleet Communications, Incorporated
386SX	Intel Corporation
486	Intel Corporation

Other trademarks are trademarks of their respective companies.

Appendix I. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

I.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see “How to Get ITSO Redbooks” on page 339.

- *IBM Communications Server for OS/2 Warp – Version 4.0 Enhancements* SG24-4587

I.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

I.3 Other Publications

These publications are also relevant as further information sources:

- *IBM Communications Server for OS/2 Warp, Version 4.1 - Guide to AnyNet LAN Gateway*, GC31-8320
- *IBM Communications Server for OS/2 Warp, Version 4.1 - Up and Running!*, GC31-8189
- *IBM Communications Server for OS/2 Warp, Version 4.1 - Network Administration and Subsystem Management Guide*, SC31-8181
- *IBM Communications Server for OS/2 Warp, Version 4.1 - Problem Determination Guide*, SC31-8186
- *IBM Communications Server for OS/2 Warp, Version 4.1 - Guide to Anynet Sockets over SNA*, GC31-8191-01
- *IBM Communications Server for OS/2 Warp, Version 4.1 - Frame Relay User's Guide and Reference*, GC31-8319
- *IBM Communications Server for OS/2 Warp, Version 4.1 - CPI-C Reference Supplement*, SC31-8153

I.3.1 Frame Relay Publications

- *White Paper: Switched Access To Frame Relay Services and Frame Relay Switched Virtual Circuits*.

  January 1996, Frame Relay Forum

  WWW: http://www.frforum.com

- *SNA Over Frame Relay. ATG's Communications & Networking Technology Guide Series*.

  1996, The Applied Technologies Group ATG.

  WWW: http://www.techguide.com

- *Frame Relay Service. In Today's Enterprise Network Environment ATG's Communications & Networking Technology Guide Series*.

  1996, The Applied Technologies Group ATG.

  WWW: http://www.techguide.com

I.3.2 LAN Publications

- *The Ethernet: A Local Area Network, Data Link Layer and Physical Layer Specifications, Digital, Intel, XEROX, Version 2.0*, November 1982, Digital Equipment Corporation, Intel Corporation, Xerox Corporation
- *IBM Local Area Network Technical Reference: IEEE 802.2 and NetBIOS Application Program Interfaces*, SC30-3587
- *IBM Token-Ring Network Problem Determination Guide*, SX27-3710 (required for IBM token ring problem determination)
- *IEEE 802.2 Local Area Networks Standard, 802.2 Logical Link Control*, ANSI/IEEE Standard, October, 1985.
- *IEEE 802.2 Local Area Networks Standard, 802.3 Carrier Sense Multiple Access*, ANSI/IEEE Standard, October 1985.
- *Network Transport Services/2 LAN Adapter and Protocol Support Configuration Guide*, S96F-8489

I.3.3 IPX Publications

- *IPX Router Specification*, Part Number 107-000029-001, Document Version 1.20; copyright April 1993 by Novell, Inc.

I.3.4 NetView Publications

- *IBM NetView Distribution Manager/2: Change Distribution Manager User's Guide*, SH19-5048

I.4 Requests for Comments (RFCs)

This section describes how to obtain electronic and printed copies of Internet RFCs.

I.4.1 Obtaining Electronic Copies through FTP

If you have FTP running on a workstation that is connected to the Internet, you can retrieve RFCs from the Network Information Center by using FTP to connect to ds.internic.net.

1. Use FTP to connect to host nic.ddn.mil.
2. Issue the command `user anonymous` to identify yourself to the host.
3. When prompted for a password, type `guest`.
4. Type the command `cd rfc` to change to the RFC directory.
5. Type the command `get rfcnnn.txt`, where `nnn` is the number of the RFC you are requesting.

I.4.2 Obtaining Electronic Copies through Electronic Mail

SERVICE@NIC.DDN.MIL is an automated service provided by the Network Information Center. It allows access to RFCs (and other documents) through ordinary electronic mail. This is especially useful for users who do not have access to the Network Information Center through a direct Internet link. Follow this procedure to obtain an RFC through electronic mail:

1. Send a mail message to SERVICE@NIC.DDN.MIL.
2. In the Subject field, type `RFC.nnn.`, where `nnn` is the RFC number. To obtain a list of all of the RFCs available, substitute the word index for nnn.

Large files are broken into smaller separate messages. The information you request is sent back to you as soon as possible.

I.4.3 Obtaining Printed Copies

Printed copies of RFCs are available for a fee from:

SRI International, Room EJ291
333 Ravenswood Avenue
Menlo Park, CA 94025
(415) 859-3695
(415) 859-6387
FAX (415) 859-6028

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at URL http://www.redbooks.ibm.com.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

  To get LIST3820s of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
  TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
  ```

  To get BookManager BOOKs of redbooks, type the following command:

  ```
  TOOLCAT REDBOOKS
  ```

  To get lists of redbooks:

  ```
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
  ```

  To register for information on workshops, residencies, and redbooks:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
  ```

  For a list of product area specialists in the ITSO:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
  ```

- **Redbooks Home Page on the World Wide Web**

  http://w3.itso.ibm.com/redbooks

- **IBM Direct Publications Catalog on the World Wide Web**

  http://www.elink.ibmlink.ibm.com/pbl/pbl

  IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

  With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** (Do not send credit card information over the Internet) — send orders to:

	IBMMAIL	**Internet**
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
(+45) 48 14 2207 (long distance charge)	Outside North America

- **1-800-IBM-4FAX (United States)** or **(+1)001-408-256-5422 (Outside USA)** — ask for:

  Index # 4421 Abstracts of new redbooks
  Index # 4422 IBM redbooks
  Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com
- **On the World Wide Web**

Redbooks Home Page	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Internet Listserver**

  With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

- Invoice to customer number
- Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

DO NOT SEND CREDIT CARD INFORMATION OVER THE INTERNET.

List of Abbreviations

IBM	International Business Machines Corporation
ITSO	International Technical Support Organization
LU	Logical Unit
PU	Physical Unit
PDU	Protocol Data Unit
BECN	Backward Explicit Congestion Notification
FECN	Forward Explicit Address Notification
CRC	Cyclic Redundancy Check
PVC	Permanent Virtual Circuit
FCS	Frame Check Sequence
LMI	Local Management Interface
DLCI	Data Link Connection Identifier
LLC2	Logical Link Layer Type 2
SNA	System Network Architecture
TCP/IP	Transport Control Protocol / Internet Protocol
SAP	Service Access Point
SNAP	IEEE Subnetwork Access Protocol
NLPID	Network Level Protocol ID
XID	Exchange Identification
ARP	Address Resolution Protocol
CPI-C	Common Programming Interface for Communications
RIP	Routing Information Protocol

Index

Special Characters

Numerics

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

ITSO Redbook Evaluation

IBM Communications Server for OS/2 Warp Version 4.1 Enhancements
SG24-4916-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redeval@vnet.ibm.com

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes____ No____

If no, please explain:

_____

_____

_____

_____

What other redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

_____

_____

_____

_____

_____

Printed in U.S.A.