Personal Communications
Version 5.0 for Windows® 95, Windows 98,
Windows NT®, and Windows 2000

Access Feature

SC31-8684-01

Personal Communications
Version 5.0 for Windows® 95, Windows 98,
Windows NT®, and Windows 2000

Access Feature

SC31-8684-01

Note

Before using this information and the product it supports, be sure to read the general information under “Appendix G. Notices” on page 79.

Third Edition (May 2000)

This edition applies to Version 5.0 of IBM Personal Communications and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About This Book

This book describes how to install, configure, and start the Personal Communications Access Feature. After you get the Personal Communications Access Feature up and running and begin to perform various tasks, use the online help whenever you need additional information. See “Personal Communications Access Feature Library” on page 29 for information about the Personal Communications library and related publications. See the glossary on page 83 for definitions of technical terms used throughout this book.

This book is for users of the IBM® ®Personal Communications Access Feature for Windows 95, Windows 98, Windows NT, and Windows 2000.

The windows in this book are representative of ones that you might see while working with Personal Communications. Minor variations could occur between what you see on your display and what is in the book.

In this book, *Windows* refers to Windows 95, Windows 98, Windows NT, and Windows 2000.

Conventions Used in This Book

This book uses the following text conventions:

UPPERCASE
: Uppercase indicates commands, program keywords, and default values. You can enter these values in uppercase or lowercase.

Bold
: Bold type indicates the names of window controls, such as lists, check boxes, entry fields, menu choices, and push buttons.

Italics
: Italics indicate:

- Special emphasis in text or a reference citation
- Variables that you supply a value for

`Example type`
: Example type indicates information that the user is instructed to type at a command prompt or in a window.

This book also uses *icons* (pictures) in the text to help you find different types of information.

This icon represents a note, important information that can affect the operation of Personal Communications or the completion of a task.

This icon represents a tip or additional information that can help you complete a task.

Part 1. Introduction

Chapter 1. Personal Communications Access Feature for Windows 95, Windows 98, Windows NT, and Windows 2000

Personal Communications Access Feature brings the power of personal networking to your workstation by exploiting the networking capabilities of Windows 95, Windows 98, Windows NT, and Windows 2000 to provide a variety of connectivity options supporting local area network (LAN) and wide area network (WAN) environments. Whether it is client/server applications or connectivity, Personal Communications offers a robust set of communication, networking, and administrative features.

Personal Communications supports a variety of SNA-based client application programming interfaces (APIs). You can create applications that use the peer-to-peer client APIs, which are based on LU 6.2 and provided by Personal Communications. These APIs let you simultaneously access and process information on peer workstations.

Personal Communications Access Feature supports Advanced Peer-to-Peer Networking® (APPN®) as an end node, and uses the advanced network features: high-performance routing (HPR) and dependent LU requester (DLUR).

Personal Communications Access Feature also includes AnyNet® SNA over TCP/IP, which allows client/server SNA applications to communicate over a TCP/IP network.

The AnyNet Sockets over SNA access node function enables TCP/IP application programs using either the WinSock 1.1 or WinSock 2.0 socket interface to communicate over an SNA network.

Personal Communications Access Feature Program

When you have installed Personal Communications Access Feature, the main functions that you can use are displayed as icons. On Windows 95, Windows 98, Windows NT, and Windows 2000, some are grouped in subfolders.

A brief explanation of each function follows:

Configuration Aid

SNA Node Configuration

Use this icon to configure an SNA node and associated resources or to change advanced configuration parameters. You will need to create an SNA node configuration to enable client/server and peer communication.

Administrative and Problem Determination (PD) Aids

Display SNA Sense Data

Use this icon to display SNA sense data and associated explanations. Sense

data is SNA error information located in internal SNA flows, error logs, error messages, and traces. Some Personal Communications error messages include sense data.

Log Viewer

Use this icon to view, merge, and sort the Personal Communications message and trace logs. Personal Communications logs errors and informational messages during initialization and operation. This can also be run from an active session by selecting **Launch > Log Viewer** from the **Actions** menu.

Information Bundler

Use this icon to gather system files and specific trace and log files, as well as registry information, such as the software installed or running on a machine. This can also be run from an active session by selecting **Launch > Information Bundler** from the **Actions** menu.

SNA Node Configuration Verification

Use this icon to verify that the ASCII editing changes that you have manually made to the SNA node configuration data are valid.

Trace Facility

Use this icon to turn trace functions on and off and to capture communication-protocol information that passes between your workstation and other host systems. You can use traces to resolve communication problems. This can also be run from an active session by selecting **Launch > Trace Facility** from the **Actions** menu.

Internet Service

Use this icon to send system and diagnostic data collected by the Information Bundler to an FTP server. Double-click this icon and follow the instructions for locating the service file in your system and establish a connection to a specified anonymous FTP site. The default site is the IBM service site for Personal Communications for Windows 95, Windows 98, Windows NT, and Windows 2000. This can also be run from an active session by selecting **Launch > Internet Service** from the **Actions** menu.

Personal Communications Migration

Use this icon to to migrate your configuration files to Personal Communications Version 5.0.

Product Information

Readme–Please!

The Readme file includes important information that became available too late to be included in this book or that might be of special interest.

Manuals

You can install the following books so they can be available online to you.

Access Feature

This book, which you are now reading, describes how to install, configure, and start the Personal Communications Access Feature.

Client/Server Communications Programming

This book describes support for the APPC and LUA application programming interfaces (APIs).

System Management Programming

This book describes how to use the Node Operations Facility (NOF) application programming interface (API).

Configuration File Reference

This book describes the advanced program-to-program communications (APPC) application programming interface (API) supported by Personal Communications.

Utilities

Check Connection (APING).

This is a sample CPI-C program that verifies network connections. You can use APING to establish network connections, diagnose problems, and take simple performance measurements. APING exchanges data packets with a partner computer and measures how long the data transfer takes.

Transfer File (AFTP).

This program provides a fast and efficient transfer of text and binary files.

Note

The APPC and CPI-C utilities are available in U.S. English only.

THESE PROGRAMS ARE PROVIDED ON AN AS-IS BASIS WITHOUT ANY WARRANTY OF ANY KIND, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE WHICH ARE EXPRESSLY DISCLAIMED.

Network Drives Configuration

Use this icon if you want to set up or change the configuration of your access to shared network drives.

Note: This function is not available for Windows 2000.

User Preference Manager

Use this icon if you want to set up or change the user preferences, such as LDAP, language selections, or default directory.

Personal Communications Sessions

The sessions that Personal Communications provides are logical connections enabling communication between your workstation and a host system. The following session type is available:

Client/server session

Establish connections that allow peer communication using any of the following methods: CPI-C; APPC (LU 6.2); and LU 0, 1, 2, and 3 using RUI.

Personal Communications Connections

The following sections describe Personal Communications connections.

Client/Server (Peer) Connections

Personal Communications provides APPN end node support for workstations, allowing them to communicate more flexibly with other systems in the network.

When your workstation is defined as an APPN end node, an APPN network node server provides the following directory and routing services for your workstation:

- Intermediate session routing
- Directory services
- Topology and route selection services

Using Personal Communications, you can establish client/server connections to a variety of computers, including System/390® (S/390®), AS/400®, and RS/6000® with AIX® SNA Services) and to workstations running OS/2®, Windows 3.x,

Windows 95, Windows 98, Windows NT, or Windows 2000. Another option is to connect the computers to a workstation running Communications Server. Available SNA client/server connection types are:

- LAN via IEEE 802.2
- Synchronous Data Link Control
- SNA-over-Async
- AnyNet SNA over TCP/IP
- Hayes AutoSync
- DLUR
- Twinaxial Data Link Control (APPC)
- SDLC
- WAC
- X25
- OEM Deep Adapter

APPN Networking

Advanced Peer-to-Peer Networking (APPN) is the underlying networking protocol that routes APPC traffic through intermediate nodes in the network. For instance, when Program A uses APPC to talk to Program B, APPN finds the node where Program B is located and directs the APPC traffic through the network.

APPN includes several features that help reduce the amount of configuration required to set up and maintain a network. These features automate many tasks that are time-consuming, complicated, and error prone. For example, if you are installing a new workstation that uses APPN, you do not have to set up configuration information for every workstation you want to communicate with. You simply provide the name of the computer and the address of the intermediate node that handles your traffic. APPN takes care of the rest of the information needed to route APPC traffic to and from your workstation.

If you connect to an APPN network, you simplify your own configuration and make it easier for other computers in the network to find you.

APPC

Advanced program-to-program communication (APPC), also known as LU 6.2, is software that enables high-speed communications between programs on different computers, from portables and workstations to midrange and host computers. APPC software is available for many different operating systems, either as part of the operating system or as a separate software package.

APPC is a communications protocol that enables programs on different computers to “talk to” each other. APPC provides the interface between the programs and the networking hardware and software and defines the rules that programs use to exchange information.

APPN Network Enhancements

By participating in an APPN network, Personal Communications workstations can also take advantage of two additional functions:

- High-performance routing
- Dependent LU requester

High-Performance Routing (HPR): Personal Communications supports high-performance routing (HPR) over token-ring and Ethernet connections, which increases data routing performance and reliability. HPR supports the Rapid

Transport Protocol to provide nondisruptive rerouting around network outages, efficient selective retransmission, and end-to-end data integrity and congestion control.

Dependent LU Requester (DLUR): 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 gateway) to be adjacent to the VTAM® host.

A DLUR is an APPN end node or network node that owns dependent LUs, but requests that a dependent LU server (DLUS) provides the system services control point (SSCP) for those dependent LUs. A DLUS controls conversion from a subarea environment to an APPN environment, allowing you to maintain central management of remote dependent LUs while benefiting from an APPN network.

Personal Communications uses 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 secondarylogical unit (SLU) support by establishing an LU 6.2 session between a dependent LU requester node (DLUR), and a dependent LU server node (DLUS).

AnyNet Function

The AnyNet function allows any application to communicate over any networking protocol and to extend the reach of applications over multiple networks. Using AnyNet function, you can reduce the number of installed network protocols and reduce operational complexity without modifying your existing applications or hardware. Personal Communications includes AnyNet SNA over TCP/IP support, which allows SNA emulator and client/server applications to communicate over a TCP/IP network.

See “Appendix D. Introducing AnyNet SNA over TCP/IP” on page 55 for examples of how to set up AnyNet connections.

What’s New in Personal Communications Access Feature

Personal Communications Access Feature Version 5.0 provides The following new enhancement:

- **Simplified Twinax Configuration**

  Personal Communications Version 5.0 Provides support for plug-and-play installation for those devices which support it. See “Using Twinax (5250) Adapters” on page 42 for details.

Part 2. Planning for and Installing Personal Communications

Chapter 2. Planning to Install Personal Communications

Personal Communications supports a wide range of workstations that use different bus architectures and peripheral components. There are hardware and software requirements, as well as memory and storage requirements,to consider when planning the installation of Personal Communications.

The following sections describe these requirements for installing Personal Communications and list support for monitors, adapters, and keyboards.

Workstation Hardware

Personal Communications supports workstations with the following hardware:

Table 1. Workstation Hardware Support

System units	The recommended system unit has an Intel® 80486, or faster, microprocessor and access to a CD-ROM drive.
Display Monitors	All display monitors supported by Windows 95, Windows 98, Windows NT, and Windows 2000.
Video Adapters	All video adapters supported by Windows 95, Windows 98, Windows NT, and Windows 2000.
Keyboards	• Enhanced keyboard (101-key, 102-key, 104-key) (Only the 101-key for Japanese) • Space-saving keyboard • Microsoft Natural Keyboard • 5576-001/002/003/A01/B01/C01/B05 (Japanese) • 5576-001 keyboard (Japanese) • 5576-002 keyboard (Japanese) • 5576-003 keyboard (Japanese) • 5576-A01 keyboard (Japanese)
Communication adapters	LAN, SDLC, coaxial, twinaxial, WAC, COM Port, X.25, OEM and multiprotocol communication adapters.
Modems	All asynchronous modems that use the Hayes AT command set and are supported by Windows 95, Windows 98, Windows NT, or Windows 2000. Synchronous (SDLC) modems attached to a multiprotocol adapter (MPA), SDLC, or WAC adapter.

Workstation Hard Disk Requirements

If you are running Windows 95 or Windows 98, Personal Communications files will need 700 KB of disk space on the same disk drive where Windows 95 or Windows 98 is installed.

If you are running Windows NT or Windows 2000, Personal Communications will need 9 MB of disk space on the same disk drive where Windows NT or Windows 2000 is installed.

Note

Refer to the Readme file for the most up to date workstation memory requirements.

Workstation Memory Requirements

The amount of memory you need to install Personal Communications depends on several factors, including the attachment type and the number of sessions. While Personal Communications will run on 16-MB workstations, a minimum of 32 MB of memory is recommended.

If you want to run Personal Communications and other applications simultaneously, you might need to add more memory to your workstation.

Workstation Software

Personal Communications requires that your workstation have Microsoft® Windows 95, Windows 98, Windows NT, or Windows 2000 installed.

Chapter 3. Installing Personal Communications

This chapter provides some tips about installing Personal Communications. Online help is also available. Press F1 in any window to obtain the help.

Types of Installation

Personal Communications allows you the following installation options:

- Install from CD-ROM or from a network server.
- Use the silent installation approach.

Considerations Before Installing

Before you begin installation, make sure all other applications are stopped. If any version of Personal Communications is already running, stop it and then start Setup. If you choose to install Personal Communications on a Windows NT or Windows 2000 workstation, you must be logged onto your workstation as a member of the Administrator group.

Using a Workstation with Multiple Operating Systems

There are some special considerations if you want to use machines that share hard disk space among multiple operating systems. If you want to put Personal Communications into a Personal Communications subdirectory that was originally installed under another operating system, you must remove the previous version first. Failure to do this may cause unpredictable results, including not being able to run Personal Communications from either operating system.

The following chart will help you find sections in this chapter that refer to the type of installation you want to do.

If you want to install Personal Communications...	Go to...
To view the online books	Page 15
To print the online books	Page 14
On a server for use in installing Personal Communications on workstations	Page 14

Note

If you are reinstalling Personal Communications Access Feature Version 5.0 for Windows 95, Windows 98, Windows NT, or Windows 2000, or are upgrading from Personal Communications for Windows 3.x or Windows 95 or Windows 98, make sure that Personal Communications is not running before you start setup.

Installing Personal Communications Access Feature

To start installing Personal Communications, do as follows:

1. Insert the medium on which you received Personal Communications Access Feature.
2. Change to the subdirectory that contains the Personal Communications Access Feature.
3. Change to subdirectory `disk1`, and enter **setup**.
4. Select **Next** from the Welcome to Personal Communications panel.
5. Do one of the following:
   - Type the drive and directory where you want to install Personal Communications.
   - Click **Browse** to display a dialog box for choosing the directory.
   - Accept the default drive and directory, **C:\Program Files\Personal Communications**.
6. Do one of the following:
   - Accept "IBM Personal Communications" as the name of the folder or program group.
   - Type another folder or group name.
   - Select another folder or group from the list.

   The Ready to Install window appears. Click **Next**. A progress bar will show the progress of the installation.
7. The next panel asks whether you want to install the IEEE 802.2 interface provided with Personal Communications. (The IEEE 802.2 interface is also known as the LLC2 protocol.) If you choose **Yes** (the default), IEEE 802.2 will install automatically. If you deselect the **Yes** box, it will not be installed.

   The same window asks whether you want automatic restart of your machine. **Yes** (the default) will result in automatic rebooting after installation is complete. Restarting is a necessary part of this installation procedure, but if you are installing several products, you may prefer to postpone restarting until all your installation activities are complete. Simply deselect the **Yes** box, but remember to reboot later, because you will not have another reminder.

Network Installation

Installing Images onto a Network Server for Remote Installation

You can copy Personal Communications installable images to a network server so that any workstations attached to the network can install the product from the server.

Copying Install Images to a Network Server

The following XCOPY command will copy the Personal Communications installable images to a network server:

```
XCOPY x:\<source>\*.* <destination>/S
```

where *x* is the CD-ROM drive that you are using and *source* is the pathname to the subdirectory (on the distribution medium) that contains the Access Feature install files such as setup.exe and setup.ins.

Removing Personal Communications Access Feature from a Workstation

For Windows NT and Windows 2000: You can remove the Personal Communications Access Feature program files, folder items, subdirectories, and registry entries. However, the configuration and other user files in your Personal Communications private subdirectory, as well as any corresponding entries in the Startup folder, are not removed.

To remove Personal Communications Access Feature from your workstation, you must be logged on as a member of the Administrator group if you are running Windows NT or Windows 2000.

1. Make sure that Personal Communications is not running before you begin your uninstallation.
2. From the Start menu, select:

   Settings
   Control Panel

3. From the Control Panel window, double-click **Add/Remove Programs**.
4. Select **IBM Personal Communications** and then click **Add/Remove**.
5. Click **Yes** to confirm that you want to delete Personal Communications.
6. Click **OK** until you return to the Control Panel window.

   If you have installed LLC2, you will be prompted to remove it.

Note

If you plan to reinstall Personal Communications, restart your workstation first.

Viewing Online Publications

Personal Communications publications can be viewed online using the Adobe Acrobat Reader.

Installing Personal Communications Using the Silent Installation Option

Personal Communications (utilizing the InstallShield Silent program) provides a method of installation that can allow for the automated electronic distribution of software. This type of installation (called silent installation) does not require a user to provide installation input via dialog boxes, but utilizes a response file to provide required input.

See “Appendix F. Using InstallShield Silent” on page 77 for information on usage examples.

Recording and Using Your Own Response File

You can record your own response file during an installation of Personal Communications; you can then use the recorded response file during all other future similar installations of Personal Communications.

Checking the Log

When you are running a silent installation, no messages appear, but a log file named **SETUP.LOG** captures the relevant installation process information. Check the log to see that the silent installation completed successfully. Failure is indicated by a ResultCode value that is less than zero.

An option is available to allow you to set up your own log file, if desired.

Part 3. Using Personal Communications

Chapter 4. Enabling Client/Server Applications

Personal Communications support for APPN end nodeslets you take advantage of APPN network nodes. An APPN network allows automatic locating of partner applications and dynamic, optimized route selection. These features allow workstations to communicate more flexibly with other systems in the network.

Personal Communications also provides support for 32-bit APPC and CPI-C APIs, which provide convenient ways for 32-bit client application programs on your workstation to communicate with those on other workstations.

Configuring a Node

Before you can take advantage of the client APIs and communicate with other application programs, you first need to configure Personal Communications as an SNA end node.

To configure the SNA node:

1. From the Start menu, select:

   Programs
   IBM Personal Communications
   SNA Node Configuration

   Or, from an active session, click **Launch > SNA Node Configuration** from the **Actions** menu.

   The Personal Communications SNA Node Configuration window appears.
2. Select the necessary Configuration Options in the order listed on the window.

   To select an option, click it and then click **New...**.

   Enter the required information, or select values for the session parameters on the Basic property sheet and then click **OK**. Click **Help** to display parameter details.
3. Click **Save** or **Save As** from the File menu.
4. Type the file name and then click **Save**. Note that you can choose the file type and where to save this file. The default file type is .ACG and the default directory is the Personal Communications private subdirectory.

   When you save an SNA configuration file, you are also makingit the default.
5. Click **Exit** from the File menu.

Tip

If you have more than one configuration file, you can apply additional configuration files to an active node. Refer to Help in SNA Node Operations for details.

Starting the SNA Node

To start the SNA node:

1. From the Start menu, select:

   Programs
   IBM Personal Communications
   Administrative and PD Aids
   SNA Node Operations

   Or, from an active session, select **Launch > SNA Node Operations** from the **Actions** menu.

   The Personal Communications SNA Node Operations window appears.
2. Click **Start Node...** from the Operations menu.

   The Select configuration file to use... window appears.
3. Double-click the configuration file (.ACG) that defines the node you want to start.

Stopping the SNA Node

To stop the SNA node:

1. From the Start menu, select:

   Programs
   IBM Personal Communications
   Administrative and PD Aids
   SNA Node Operations

   Or, from an active session, select **Launch > SNA Node Operations** from the **Actions** menu.

   The Personal Communications SNA Node Operations window appears.
2. Click a resource in the Alias column.
3. Click **Stop Node...** from the Operations menu.

Command Line Utilities

Another way to perform SNA Node operations is to use the command line. The following command line programs are available to perform basic operations for the SNA Node.

CSSTART [-p | -q | -a | -m | -d | -h][cfgfile]

The CSSTART command is used to start the SNA Node with a specified configuration. If the **-p** flag is used, then all messages are shown in a message box popup. Messages are written to stdout by default. The **-q** flag suppresses all output. The **-a** flag enables automatic start up after rebooting the machine. The **-m** flag disables automatic start up after rebooting the machine. The **-d** flag sets the default SNA node configuration file. The **-h** flag displays help for this command.

The **cfgfile** parameter can be:

- The full path name of the configuration file you want to use to start the product.
- fn.ext (Personal Communications private subdirectory is assumed).
- fn (.ACG ext and Personal Communications private subdirectory are assumed).

If no **cfgfile** parameter is specified, then the default configuration file is used. If no default configuration file has been set, then an error is reported. If the node is successfully started, then CSSTART returns zero, otherwise a non-zero value is returned.

The **-a**, **-m**, **-d**, and **-h** flags do not start the SNA node.

CSSTOP [-p | -q | -h]

The CSSTOP command is used to stop the SNA Node. The flags have the same meaning as for CSSTART. If the node is successfully stopped then CSSTOP returns zero, otherwise a non-zero value is returned.

The **-h** flag will not issue the command.

CSQUERY [-p | -q | -h]

The CSQUERY command is used to query the status of the SNA Node. The flags have the same meaning as for CSSTART. Along with the running or stopped status, the default and active configuration file names are shown. If the node is running, then CSQUERY returns zero; otherwise a non-zero value is returned.

The **-h** flag will not issue the command.

APING Requirements

If you wish to run the Check Connection (APING) utility provided with this package to an AS/400® running OS/400® V3R1, you must install PTF MF10808 on the AS/400 first. If this PTF is not installed, then errors with sense codes of 080F 6051 and 1008 201D will occur.

Chapter 5. SNA Node Operations

SNA Node Operations enable you to display information and manage the node for certain Personal Communications resources, which can be useful when operating the SNA node. The following tasks are supported:

- Starting and stopping resources
- Deleting resources
- Displaying resource information
- Changing session limits
- Initiating path switches

The configuration information for a SNA Node is stored in an ASCII file, with the extension `.ACG`. You can edit the file by launching SNA Node Configuration. To ensure that your new configuration is correct, use the Verification tool.

You can display information or take action on resources if you start client/server sessions with one of the following attachment types:

Client/Server Connections
LAN LAN via IEEE 802.2
Twinaxial
Twinaxial Data Link Control (APPC)
COM port
SNA-over-Async
Hayes AutoSync
X.25
SDLC Synchronous Data Link Control
IBM WAC
Synchronous Data Link Control
X.25
AnyNet® SNA over TCP/IP
APPC
Sockets over SNA
AnyNet IP to LU Mapping
AnyNet Sockets Connections
AnyNet Sockets Devices
AnyNet Sockets Routes

SNA resources that can be displayed or managed include the following:

Connection Networks
Allow APPN nodes in a LAN to have direct links with each other without requiring logical link definitions at each node.

Connections
Link stations to the adjacent nodes

CPI-C Side Information
Associate a set of parameters with a specified symbolic destination name.

Data Link Control
How data is formatted for transmission on the physical connection.

Devices
Parts of the workstation hardware, such as adapters, that are used to transmit and receive data.

DLUR PU
A dependent logical unit requester (DLUR) physical unit (PU) is a PU in an APPN end node that owns dependent LUs, but requests that a dependent LU server provide the SSCP services for those dependent LUs.

Focal Point
A system that provides centralized network management services.

Local LU 0 to 3
A local logical unit (LU) type 0, 1, 2, or 3 is a dependent LU on the workstation that provides services for 3270 terminal and printer emulation applications.

Local LU 6.2
A local logical unit (LU) type 6.2 is an independent or dependent LU that provides APPC services.

LU 6.2 Session
Transports data between two partner LU 6.2s. Conversations between transaction programs use LU 6.2 sessions.

Mode The name used by the initiator of a session to designate the characteristics desired for the session, such as traffic pacing values, message-length limits, sync point and cryptography options, and the class of service (COS) within the transport network.

Node The control point that manages the node and its associated resources. The local node provides APPN services.

Partner LU 6.2
A remote computer that communicates through an APPC session with local LU 6.2s.

RTP Connection
In high-performance routing (HPR), a rapid transport protocol (RTP) connection is the connection established between the endpoints of the route to transport session traffic.

Transaction Program
A transaction program (TP) is a program that processes transactions in an SNA network. There are two kinds of TPs: application transaction programs and service transaction programs.

Starting Node Operations

To start Node Operations, click on the **SNA Node Operations** icon in the Personal Communications folder.

Using the Menu Bar

From the menu bar, select one of the following:

Operations
To start or stop a SNA node.

To start a node, select the configuration file you want to use.

To change session limits (CNOS).

To apply a new configuration file, select the configuration file you want to use.

Launch

To launch to other Personal Communications programs, such as Log Viewer, Trace Facility, or SNA Node Configuration.

View To modify the layout of your Personal Communications window (Tool bar or Status bar).

To select resource attributes, which allows you to customize the display of SNA resource attributes.

To view frequently used resources, such as:
- Node
- Connections
- Local LU 0 to 3
- Partner LUs
- LU 6.2 sessions

Displaying a Resource

To display a resource, select a resource from the **Select SNA Resource** pull-down list.

Starting a Resource

To start a resource:

1. Select a resource from the **Select SNA Resource** pull-down list.
2. Select the first column of the item to display the pop-up.
3. Select **Start** to start the resource. A started (active) resource can be stopped, but not deleted.

If the resource to be started is a connection, an indicator in the form of a lightning bolt is displayed for the duration of the connection attempt.

Stopping a Resource

To stop a resource:

1. Select a resource from the **Select SNA Resource** pull-down list.
2. Select the first column of the item to display the pop-up.
3. Select **Stop** to suspend the resource. If a resource is stopped (inactive), it can be deleted or redefined.

Deleting a Resource

To delete a resource:

1. Select a resource from the **Select SNA Resource** pull-down list.
2. Select the first column of the item to display the pop-up.
3. Select **Delete** to delete the resource.

A resource to be redefined need not be deleted, but it should be stopped (inactive).

Changing an HPR Path Switch

To perform a high performance routing (HPR) path switch:

1. Select **RTP connections** from the Select SNA Resource pull-down list.
2. Select the first column of the item to display the pop-up.
3. Select **HPR path switch** to change the path for the data on this RTP connection.

Updating SNA Resources Dynamically

It is not necessary to delete an SNA resource before redefining it. A resource must be inactive, however, when you redefine it.

If a redefinition is rejected for a reason other than that the resource is active, a STATE_CHECK return code is returned, as well as a specific secondary return code as listed in the NOF specification. Most rejections are due to inconsistencies in the configuration, such as one of the following:

- Attempting to define an LS with the adjacent CP name set to the local CP name.
- Attempting to define two link stations on a non-switched port with the same destination address.

Such rejections occur regardless of whether you are defining a resource for the first time or redefining it.

Keep in mind the following when redefining resources:

- You may not change the ls_role of a port that has link stations defined on it. The link stations must be deleted before the port role can be changed (because an LS may default to the port ls_role).
- You may not change the type of a named DLC. The DLC definition must be deleted before the same name can be reused for a DLC of a different type.

Chapter 6. Problem Analysis

This chapter describes the information that will help you analyze problems with Personal Communications and ways to report a problem to IBM.

For information about Personal Communications and support, refer to the following Web sites:

- The Personal Communications home page provides access to general product information, and download services. To view this page, use an Internet browser and go to the following Web site:
  http://www.ibm.com/software/network/pcomm
- The Personal Communications support page provides links to code fixes, tips, newsgroups, support options, and services. To view this page or to submit a software defect report, use an Internet browser and go to the following Web site:
  http://www.ibm.com/software/network/pcomm/support

Administrative and Problem Determination Aids

Personal Communications provides several utilities to help you with problem analysis. The following sections describe these utilities and how to use them.

Log Viewer

The Personal Communications log viewer utility enables you to view, merge, sort, search, and filter information contained in message and trace logs. Use the log viewer during problem analysis to work with message and trace log entries. The default name of the message log output file is *PCSMSG.MLG*; its file extension must be *.MLG*. The file extension for trace logs must be *.TLG*.

To view message or trace logs:

1. From the Administrative and PD Aids subfolder, click **Log Viewer**.

   Or, from an active session, select **Launch > Log Viewer** from the **Actions** menu.
2. From the list of logged messages, double-click on a message to display the message text.

For more information about log viewer functions, refer to the *Personal Communications Reference Volume I*.

Trace Facility

The Personal Communications trace facility enables you to log trace information for certain Personal Communications functions.

To start a trace, perform the following steps:

1. From the Administrative and PD Aids folder, click **Trace Facility**.

   Or, from an active session, select **Launch > Trace Facility** from the **Actions** menu.

   The trace status on the title bar displays the current state:

   Active Trace data is being collected by the trace facility.

   Inactive
   No trace data is being collected.

2. From the main dialog box, click **Set Up** to set the desired trace system parameters.
3. Click **OK** to return to the main trace dialog box.
4. From the main trace dialog box, select the type of data you want to trace from the Function Name, Component Name, and Trace Option list boxes.

   Function Name
   A specific set of Personal Communications features, such as 3270/5250 Emulator or User Services.

   Component Name
   the name of a specific part of a function, such as API data (for the 3270/5250 Emulator function) or Node Initialization (for the User Services function).

   Trace Options
   The options associated with a particular component, such as EHLLAPI (for the API component) or API trace (for the Node Initialization component).
5. Start tracing data by clicking **Start**, or apply changes to the trace options by clicking **Apply**.
6. Run the operation that you want to trace.
7. Optionally, stop the trace by clicking **Stop**.
8. Save the trace data to your hard disk by clicking **Save**.
9. Click **Format** to specify a formatted trace file name and to format the trace data. The Information Bundler utility should be used immediately after the trace is complete to ensure that the correct information is gathered.

   Note: If you have changed the path for the fomatted trace file, the Information Bundler will not find the trace information. Copy the trace files to the \private subdirectory of the directory in which Personal Communications is installed.
10. Click **OK**.
11. Click **Clear** to clear the trace buffer where you saved a trace.
12. Use the log viewer to view the formatted trace log.

Information Bundler

The Personal Communications Information Bundler utility gathers system files, trace and log files, and registry information and creates a .RAM file. This .RAM file is sent to support personnel via FTP for problem resolution. This utility should be executed immediately after the trace is complete to ensure that the correct information is gathered. To use the information bundler utility:

1. From the RAS folder in the Personal Communications program group, double-click the **Information Bundler** icon.

   Or, from an active session, select **Launch > Information Bundler** from the **Actions** menu.
2. A .RAM file containing system and Personal Communications information is created in the Personal Communications directory. By default this file is called 12345.RAM.

Internet Service

The Internet Service utility enables you to send the .RAM file containing diagnostic data collected by the Information Bundler to an FTP server. This utility will not work unless TCP/IP is configured and you have a direct connection to the internet. To use Internet Service:

1. Double-click the **Internet Service** icon located in the Personal Communications program group.

   Or, from an active session, select **Launch > Internet Service** from the **Actions** menu.

   The Internet Service window contains four data fields that must have valid values before you can submit your problem report.
2. Verify that the FTP Address field contains the default address **testcase.software.ibm.com**. This is the service anonymous FTP server.
3. Enter your email address in the field provided.
4. In the Problem Determination Filename field, enter the file name and path of the .RAM file created with the Information Bundler. This file is located in the Personal Communications installation directory.
5. In the PMR Number field, enter the PMR number that you received as a result of contacting IBM support personnel.
6. When all fields in the window have been filled with valid values, click **Transmit** to submit your problem determination information.

Personal Communications Access Feature Library

The Personal Communications Access Feature library includes the following online publications:

- *IBM Personal Communications Access Feature, SC31-8684*
- *IBM Personal Communications Client/Server Communications Programming, SC31-8479*
- *IBM Personal Communications System Management Programming, SC31-8480*
- *IBM Personal Communications Configuration File Reference, SC31-8655*

Related Publications

For information about local area networks (LANs), refer to the following publications:

- *IBM Local Area Network Technical Reference*
- *AS/400 Communications: Local Area Network (LAN) Guide Version 2*

Contacting IBM

This section lists a number of ways you can reach IBM for various reasons. Depending on the nature of your problem or concern, we will ask you to be prepared to provide the following information to allow us to serve you better.

- The environment in which the problem occurs:
  - Personal Communications configuration
    - Personal Communications version and CSD level
    - The name of the workstation profile
    - The name of the SNA Node Configuration file
  - Workstation configuration

    - The machine type and model, the system memory, the video adapter
    - The communication adapter you are using
    - Other adapters (especially communication adapters) installed
    - The printer type and model
    - Other devices installed, such as sound cards, modems, or fax machines
  – Software configuration
    - Windows version and level
    - Communications and device-driver version and level
    - Other communication programs (such as NetWare, Microsoft SNA Server, or Microsoft® Data Link Control) that are running and using resources
    - Printer driver version and level
  – Host configuration
    - The upstream host connection and configuration
- Problem analysis information
  – Symptoms
  – Type of problem
  – OIA messages or error messages (if any)
  – Key factors related to the problem

If you have a technical problem, take the time to review and carry out the actions suggested here. Use your local support personnel before contacting IBM. You can also check the Hints and Tips on the WWW for more information. Only persons with in-depth knowledge of the problem should contact IBM; therefore, support personnel should act as the interface with IBM.

Support Options

If you determine that you need to contact IBM, you can do any of the following:

- Access the Personal Communications for Windows 95, Windows 98, and Windows NT Web page:

  `http://www.ibm.com/software/network/pcomm`
- To find the phone number for IBM Software Support, U.S. customers can call 1-800-IBM-4YOU. International customers that have access to the U.S. ″800″ toll free numbers can reach the International Support Center by calling 1-800-IBM-4YOU and asking to speak with the International Support Center (ISC) in Atlanta. International customers without access to the U.S. toll free numbers can call the ISC directly at 770-863-1234. The ISC's FAX number is 770-863-3030 and is available 24 hours a day.

Part 4. Appendixes

Appendix A. Directories Created by the Installation Program

The program creates the following subdirectories in the root of the main Personal Communications directory: INCLUDE, LIB, and (optionally) PRIVATE. With the exception of the PRIVATE subdirectory, the subdirectories might contain files only when you have selected to install them.

Note: In addition, the program creates subdirectories for each language selected for installation. (For example, en_US is created for US English.)

Personal Communications Private Subdirectory

Note: The location of this directory can be selected; by default, it is the PRIVATE subdirectory in the directory where you install Personal Communications.

This directory is almost empty when you first install Personal Communications (unless you have migrated from a previous version), but if you create and save configurations, the following types of files are stored here:

Personal files in the Personal Communications private subdirectory:
- SNA configuration files (*.ACG)
- Automatic-dial facility files (.ADU)
- Trace log files (.TLG)

Appendix B. Attachment Considerations

This appendix describes factors you should consider when configuring Personal Communications for the attachment types listed here.

LAN via IEEE 802.2 Attachment

Following is a description of preinstallation considerations for installing Personal Communications for use on a LAN. For more detailed information, refer to the *IBM Local Area Network Technical Reference*.

PIU Size

This value is negotiated between your workstation and the host system, unless `XID=No` is defined on the PU, or the Adjacent CP name has been changed to `Host - XID0` in SNA Node Configuration. If this value is not negotiated, the `Maximum PIU Size` must match the `MAXDATA` value defined on the host system.

Receive Window Count

This value is negotiated between your workstation and the host system, unless `XID=No` is defined on the PU, or the Adjacent CP name has been changed to `Host - XID0` in SNA Node Configuration. If this value is not negotiated, the `Receive Window Count` must match the `MAXOUT` value defined on the host system.

Using a PCMCIA Token-Ring or Ethernet Card

PC/3270 supports the IBM Token-Ring and Ethernet credit card adapters in the same way as it does the full-size adapters.

SDLC MPA Attachment

This section describes factors you should consider when configuring Personal Communications with an SDLC attachment.

Line Speed

For the SDLC attachment, a line speed up to 28.8 Kbps is supported. However, actual line speed will vary depending on the carrier speed the modem supports.

V.25bis Autodial

Personal Communications only supports the bit-oriented protocol for V.25bis call control messages. Modems must support an ASCII-coded character set consisting of 7-bit coded ASCII characters with odd parity.

Personal Communications supports the following adapters:

- Multiprotocol (MPA) interface compatible adapters, for example:
  - IBM Multiprotocol Communications Adapter (73G7099)
  - IBM Multiprotocol Adapter/A (645114 or later)
  - IBM SDLC/Asynch Communications Adapter (ASCA) (42H4332)
- IBM SDLC PCMCIA adapter (Japanese only)
- IBM SDLC PCMCIA modem (Japanese only)

- IBM PCI Multiprotocol Adapter (12J2981)

Performance Considerations

If you use multiple sessions in a high-speed and high-load configuration with an SDLC attachment, the connection to the host might fail. In this case, set the host's timeout value to be greater than that of Personal Communications.

- To change the host timeout value, modify the following NCP parameters:

  REPLYTO
  The maximum duration in seconds that the host waits for a reply from the terminal. The default is 1 second.

  RETRIES=(m,t,n)
  The number of recovery attempts for errors occurring during transmission over the link. Repetitive retries are called a retry sequence. The maximum number of retries is specified as m, which can be from 0 to 128. NCP pause (the time between the retry sequences) is specified as *t* and can be from 1 to 255 seconds. The maximum number of retry sequences is specified by *n* and can be from 1 to 127. For example:

  ```
  REPLYTO=1, RETRIES=(3,4,5)
  (( 1(seconds) x 3(times) + 4(seconds)) x 5(times)
  = 35 (seconds)
  ```

  Note

  

  In this example, the host will wait 35 seconds for the terminal to reply.

  For more details, ask your system personnel.

- To change the Personal Communications timeout value, type a new value in the Inactivity timer field on the Performance page of the Connections property sheet in SNA Node Configuration.

  The value is in seconds and can be from 40 to 160. The default is 80.

PIU Size

This value is negotiated between your workstation and the host system, unless `XID=No` is defined on the PU, or the Adjacent CP name has been changed to `Host - XID0` in SNA Node Configuration. If this value is not negotiated, the `Maximum PIU Size` must match the `MAXDATA` value defined on the host system.

Receive Window Count

This value is negotiated between your workstation and the host system, unless `XID=No` is defined on the PU, or the Adjacent CP name has been changed to `Host - XID0` in SNA Node Configuration. If this value is not negotiated, the `Receive Window Count` must match the `MAXOUT` value defined on the host system.

Preventing Conflicts with Non-PCI MPA Adapters

The multiprotocol adapters use system resources that, in some cases, cannot be shared.Therefore, you must make sure there are no conflicts with the following resources:

- DMA level
- I/O addresses (MPA0 uses X'380'–X'38F' and MPA1 uses X'3A0'–X'3AF')
- IRQ level (ISA bus only)

The IBM Multiprotocol interface adapters use the following system resources:

- The AT-bus SDLC adapters require the use of IRQ3 and DMA channel 1 and only the MPA0 communication port is supported.
- The MPA for ISA can be supported using mode 4 (IRQ3 and DMA1 channel 1) or mode A (IRQ4 and DMA channel 1). IRQ level send/receive of 3 or 4 is supported. The ASCA can be supported using mode 4 (IDQ3 or IRQ5 or IRQ4 or IRQ7 and DMA channel 1) or mode A (IRQ4 or IRQ7 and DMA channel 1. IRQ level send/receive or 3, 4, 5, or 7 is supported. The interrupt level setting on the configuration panel in Personal Communications must match the value on the adapter card.

Note

When the MPA is configured for mode 4, the device will not be allowed to use the other IRQ even though only one interrupt is claimed.

Personal Communications supports the MPA0 or MPA1 communication ports. Some sound cards use an I/O address within the range X'380'–X'38F'. MPA1 should be used to avoid a conflict.

- The MPA/A for a microchannel-bus workstation uses IRQ3, and the DMA level is configurable. Both the MPA0 and MPA1 communication ports can be active at the same time and the interrupt is claimed as sharable.

IBM Global Network—SNA over Async Attachment

IBM Global Network - SNA over Async is supported using 8-bit no parity; SNA session-level compression and decompression is not supported.

Line Speed

Line speeds up to115.2 kbps are supported. The maximum usable line speed depends on the speed of the processor or communication port, however. If communication stops or becomes exceedingly slow, reduce the line speed.

Updating Phone List

To update the phone number list for IBM Global Network SNA over Async, do the following:

1. Click the **Used Phone Book** button on the **Define a Com Port Connection** panel.
2. Click the **Download Phone Numbers** button.

   Personal Communications FTPs to the IGN server and downloads the phone number file and converts it to the correct format.

Note

Sockets over SNA or Proxy must be configured in order for this procedure to work.

IBM PC720 Modem Initialization (Japan Only)

In the Japanese version of Personal Communications, if you use an IBM PC720 and connect it via a dial connection, the error message "COMDIA15 Unable to initialize modem" may appear at startup. To correct this condition, increase the value of **Retries** in **Initialization** to more than 1 by using the **Modem Information** panel.

Practical Peripherals 14400MT Modem Considerations

The Practical Peripherals 14400MT modem does not always work properly when receiving incoming connections. If it appears that the modem is not answering an incoming call, then do the following:

1. Using the Node Operations panel, stop the Device associated with the modem.
2. Turn off power to the modem and then back on.
3. Using the Node Operations panel, start the Device associated with the modem.

Twinax Attachments

Personal Communications classifies twinax adapters under five adapter types:

- IBM 5250 Plug-and-Play Adapters
  - IBM 5250 Emulation PCMCIA Adapter
  - IBM 5250 PCMCIA Adapter (DBCS-unique adapter)
  - IBM 5250 Emulation PCI Adapter
  - IBM 5250 Express ISA Adapter, *when the adapter's Plug-and-Play BIOS is enabled.*
  - IBM 5250 Express PC Card
  - IBM 5250 Express PCI Adapter
- IBM 5250 MicroChannel Adapters
  - IBM System36/38 Workstation Emulation Adapter /A (DBCS-unique adapter)
  - IBM 5250 Emulation Adapter /A
- IBM 5250 ISA Adapters
  - IBM Enhanced 5250 Display Station Emulation Adapter
  - IBM 5250 Express ISA Adapter, *when the adapter's Plug-and-Play BIOS is disabled.*
- IBM 5250 ISA Adapter (Asia-Pacific)
  - IBM AT-Bus Communication Adapter (DBCS-unique adapter)
- Other Twinax Adapter
  - OEM 5250 adapters

See "Using Twinax (5250) Adapters" on page 42 for details on these adapters.

IBM SDLC PCMCIA Adapter and Modem (Japan Only)

For Windows 95/98: The support uses the Configuration Manager for Plug and Play support, which will automatically configure IRQ number, I/O ports and memory addresses as required. IBMSDLC.INF (installed in the Personal Communications installation directory) describes the SDLC PCMCIA cards to the operating system. When you insert the PCMCIA card, Windows 95/98 prompts you for the adapter drivers. You must specify the `Personal Communications` installation directory and select **OK**. To verify that the adapter is functioning properly, select the appropriate PCMCIA adapter under **Other Device** on the System Property page's Device Manager property sheet.

For Windows NT and Windows 2000: Plug and Play is not supported. During installation the registry is updated, using default parameters. When the configuration is started and an incoming or outbound connection is enabled, the device attempts to load, using the stored parameters from the registry. If a card is found and a conflict is detected, the registry is updated with values that do not conflict and you are notified to restart the system.

Note

The system must be restarted for these values to take affect.

No more than one SDLC PCMCIA card can be used concurrently. Because a common driver supports both SDLC PCMCIA and MPA interface cards, an MPA cannot be used concurrently with SDLC PCMCIA.

IBM SDLC PCMCIA Adapter

This card supports:

- Leased and switched connections
- Speeds up to 19.2 Kbps
- Switched and Constant Request To Send (RTS)
- NRZ and NRZI encoding schemes

The IRQ number on the device page's adapter property sheet is ignored.

IBM SDLC PCMCIA Modem

This card supports:

- Switched V.26bis connections only
- Speeds up to 2400 baud
- Switched RTS only

The configured value on the constant RTS and IRQ level send/receive device page's adapter property sheet are ignored.

IBM PCI Multiprotocol Adapter

Refer to the *Quick Installation Guide* in the *PCI Multiprotocol Adapter User's Guide* for instructions on installing this adapter under Windows 95, Windows 98, Windows NT, and Windows 2000.

Two IBM PCI Multiprotocol Adapter's can operate concurrently. The configuration specifies one adapter with Communication Port MPA0 and the second adapter with Communication Port MPA1. When the link is activated, MPA0 is the first PCI Multiprotocol Adapter that the operating system finds and MPA1 is the second adapter found. If more than one PCI Multiprotocol adapter is installed on a machine and you are unsure which adapter the operating system will find first, you may want to perform an isolated test to determine which adapter is connected to which Communication Port. One suggestion is to disconnect the modems connected to the adapter from the telephone line and then try to start each link individually. By watching the modems, you should be able to determine which modem is connected to which Communication Port. Then the modems can be attached to the proper network. If a single PCI Multiprotocol Adapter is used, the configuration file must specify MPA0 as the Communication Port.

If a Personal Communications configuration is using this adapter, and a PCI MPA adapter is added or removed from the system, Personal Communications may have to be reconfigured.

An ISA-bus MPA and IBM PCI Multiprotocol Adapter can be used concurrently, as long as the ISA-bus MPA adapter is configured as MPA1.

This adapter supports:

- Leased and switched connections
- Speeds up to 56 Kbps
- Switched and Constant Request To Send (RTS)
- NRZ and NRZI encoding schemes

The IRQ number on the device page adapter property sheet is ignored.

WAC Attachments

This section describes factors you should consider when configuring Personal Communications to communicate via a Wide Area Connector (WAC) adapter.

EIB Support

All available Electrical Interface Boards (EIBs) are supported for the WAC adapter.

Supported Adapters

The IBM WAC adapters for bus types ISA and MCA are supported. **Note:** At this time, the new IBM PCI bus WAC adapter is NOT supported by Personal Communications.

Potential Configuration Issues

When configuring both ports (upper and lower) on a single WAC adapter, you must create two logical devices, using the configuration utility. Be careful to configure the same Shared Ram Address for both devices, using the same adapter number. If you do not do this, the Shared Ram Address for the first device activated will be used for the second device.

For ISA systems, the adapter number is obtained from the interpretation of the on-board DIP switch settings. Switches 1-3 determine the Interrupt level, and switches 4-7 (read in reverse order) determine the adapter number. Consult the adapter installation publication for additional information. Adapter numbers 0 and 1 are logically the same. This is also true for adapter numbers 8 and 9.

On some IBM ValuePoint systems, intermittent problems can occur. If you are using one of these systems and are having problems, ensure that the adapter's DIP switch settings are such that Interrupt Level 9 is being used.

If you intend to use the WAC adapter in a system and you also plan on using Hayes AutoSync function, then you might experience problems with your async connection. To find appropriate Shared Memory values, you should use the adapter diagnostic disk provided with the adapter.

For MCA systems, the adapter number is the slot number in which the adapter physically resides. You can easily obtain this number via the MCA Configuration utility that was provided with your system. For IBM systems, you can determine this number by looking at the back of the system unit and providing the number displayed for the selected slot.

System Resources

The Wide Area Connector adapter utilizes the following system resources per each adapter installed:

- 32 bytes of IO addresses (differs depending upon the adapter number used on ISA; see help on the configuration panel for additional information, or via MCA adapter config.)
- Interrupt level (1 level can be used on MCA if all adapters share the same, ISA adapters must have unique interrupt numbers.)
- 16KB of shared memory

For ISA adapters, you should consult your WAC installation publication for information concerning the IRQ used and the adapter number configured, both of which are configured via the adapter's DIP switches.

3174 Peer Communications Support (Lan over Coax)

The Lan over Coax function provides peer-to-peer communication for workstations connected to an IBM 3174 Control Unit via an IBM 3270 communication adapter. It provides support for any NDIS protocol stack capable of utilizing the IEEE 802.5 (Token Ring) frame format such as IEEE 802.2 (SNA), Netbios, TCP/IP, or LLC2.

Installation

To install the 3174 Peer Communications Adapter Support, from the Network Control Panel, choose to install a new adapter. When prompted, select the **Have Disk** option, and direct the installation program to the `XLNCOAX` subdirectory in the `Personal Communications` installation directory.

Attachment Considerations

The `.INF` file will install the new adapter support and bind the adapter to all valid protocol stacks. To utilize the new adapter, after installing and rebooting, configure Personal Communications to utilize a LAN adapter, and enter the adapter number that corresponds to the 3174 Peer Communications adapter number found under the LLC Protocol for this adapter. The 3174 Peer Communications RPQ is also required on the 3174.

Configuration

Configure by using the Network Control Panel's **Properties** button for the selected adapter.

Options

Network Address
Overrides the burned in adapter address.

Queued Transmits
Number of queued transmit elements to support.

Slot Number
Required for MCA bus systems, not applicable to others; indicates the adapter to use in a specific MCA slot.

On Micro Channel workstations, you can use up to 4 type-B 3270 Connection Adapters (but only 1 type A). When you configure Personal Communications, you must specify the physical slot in which each adapter is installed.

System Resources

On ISA Bus systems, the following system resources are used:

- Interrupt Level 2
- IO address 0x2d0 through 0x2df
- Shared memory address 0xce000

These resources are not changeable. On MCA systems, these resources are configured via the MCA adapter configuration, with the exception for the Interrupt number, which is Interrupt 2.

Multiple Adapter Coexistence

If you use type-B adapters, you can use one or more for DFT connections concurrently with 3174 Peer Communications. If you want to do this, the Peer Communications adapter(s) must be in higher-numbered slots than the DFT adapter(s) because DFT connections will, by design, use the first adapter(s) they find, searching from slot 1.

Adapter Setup Hints and Tips

This section describes how to configure adapters, including PCMCIA cards.

Using Twinax (5250) Adapters

Personal Communications classifies twinax adapters under five adapter types:

- IBM 5250 Plug-and-Play Adapters
  - IBM 5250 Emulation PCMCIA Adapter
  - IBM 5250 PCMCIA Adapter (DBCS-unique adapter)
  - IBM 5250 Emulation PCI Adapter

  - IBM 5250 Express ISA Adapter, *when the adapter's Plug-and-Play BIOS is enabled.*
  - IBM 5250 Express PC Card
  - IBM 5250 Express PCI Adapter
- IBM 5250 MicroChannel Adapters
  - IBM System36/38 Workstation Emulation Adapter /A (DBCS-unique adapter)
  - IBM 5250 Emulation Adapter /A
- IBM 5250 ISA Adapters
  - IBM Enhanced 5250 Display Station Emulation Adapter
  - IBM 5250 Express ISA Adapter, *when the adapter's Plug-and-Play BIOS is disabled.*
- IBM 5250 ISA Adapter (Asia-Pacific)
  - IBM AT-Bus Communication Adapter (DBCS-unique adapter)
- Other Twinax Adapter
  - OEM 5250 adapters

The configuration of each IBM adapter type is described in the following sections. All the IBM adapters are supported; OEM adapters are supported if their manufacturer has created a device driver that is compatible with the IBM driver.

IBM 5250 Plug-and-Play Adapter

In Personal Communications, set the **Adapter Type** to IBM 5250 Plug-and-Play Adapter. You do not have to set the **Interrupt Level**, **I/O Address**, or **Memory Address** fields; these fields are grayed out. Personal Communications automatically reads the interrupt level, I/O address, and memory address from the adapter card.

Although you will probably not need to do so, you can change the interrupt level, I/O address, or memory address using the adapter configuration utility provided with the adapter.

If you experience problems getting a host connection with the 5250 Emulation PCI Adapter and you have an IBM personal computer, upgrading the BIOS might solve the problem. You can find the date of your system BIOS by using your system setup when you boot your personal computer. Then you can look on the Internet to see if there is a later version for your personal computer. The IBM File Library Web site that has the latest BIOS upgrade files is located at:
www.pc.ibm.com/listfiles.html

From there you can search for your specific IBM personal computer. If there is a flash BIOS upgrade date later than that on your personal computer, download the file to your personal computer and follow the instructions that come with the file. For non-IBM personal computers, contact the manufacturer of your personal computer for BIOS upgrade information.

IBM 5250 MicroChannel Adapter

In Personal Communications, set the adapter type to IBM 5250 MicroChannel Adapter. You do not have to set the **Interrupt Level**, **I/O Address**, or **Memory Address** fields; these fields are grayed out. Personal Communications automatically reads the interrupt level, I/O address, and memory address from the adapter card.

Although you will probably not need to, you can change the adapter's interrupt level, I/O address, or memory address using the system configuration program and the reference diskette provided with the adapter.

IBM 5250 ISA Adapter: In Personal Communications, set the adapter type to IBM 5250 ISA Adapter. You must then set the interrupt level, I/O address, and Memory address to match the values set on the adapter card. For a new configuration, Personal Communications shows the default settings.

To view or change the settings on the adapter card, use the adapter configuration utility provided with the adapter. For the IBM Enhanced 5250 Display Station Emulation Adapter only, change the I/O address using switches on the adapter itself.

IBM 5250 ISA Adapter (Asia-Pacific)

In Personal Communications, set the adapter type to IBM 5250 ISA Adapter (Asia-Pacific). You must then set the interrupt level and memory address to match the values set on the adapter card. For a new configuration, Personal Communications shows the default settings. The **I/O Address** field is grayed out; you do not need to set this field. Personal Communications automatically reads the I/O address from the adapter card.

To change the I/O address on the adapter card, use the switches on the adapter itself. To view the I/O address, or to view or change the interrupt level or memory address on the adapter card, use the adapter configuration utility provided with the adapter.

Twinax Connections through a 5494 Controller to an AS/400

To set up a twinaxial connection to an AS/400 through a 5494 controller:

1. Click **Programs>IBM Personal Communications>SNA Node Configuration** from the Windows **Start** menu.
2. Click **Configure Connections** from the list of **Configuration options** and then click **Twinaxial** from the list of DLCs.
3. Click an entry from the list of configured twinaxial connections and click **View>Change>Add**
4. Check **Link to preferred NN server**.
5. Click the Security tab.
6. In the **Adjacent CP name** field, enter the network ID and CP name of the 5494 controller.
7. Click **OK** to save the connection information.
8. Click **Configure Partner LU 6.2** from the list of **Configuration options**.
9. Click an entry from the list of **Partner LU 6.2** definitions and click **View>Change>Add**
10. In the **Partner LU name** field, enter the CP name of the AS/400 system. The CP name is also known as the system location name.
11. In the **Fully qualified CP name** field, enter the network ID and CP name of the 5494 controller.
12. Click **OK** to save the Partner LU 6.2 information.
13. Click **Save As...** from the **File** menu.
14. Enter a file name with an .ACG file extension, such as `CFG5494.ACG`. Click **Yes** to make this configuration the default.

Note: To enable this configuration, configure the 5494 controller as a network node (NN).

Hints and Tips

Appendix C. SNA Client/Server Concepts

Terminology

Advanced Program-to-Program Communications (APPC)
An implementation of the SNA LU 6.2 protocol that allows interconnected systems to communicate and share the processing of programs.

Advanced Peer-to-Peer Networking (APPN)
An enhancement for Systems Network Architecture (SNA) networks featuring:

- Dynamic exchange of network topology information that simplifies connections, route selection, network definition, and reconfiguration
- Automated resource registration and directory lookup
- Greater distributed network control that helps to isolate the effects of single points of failure

Common Programming Interface for Communications (CPI-C)
Personal Communications provides support for the Common Programming Interface for Communications (CPI-C) 2.0 industry-standard interface from X/Open. The CPI-C interface enables greater application portability across different platforms. By using CPI-C 2.0, APPC programming is simplified, resulting in reduced cycle time, and enhanced client-server computing capability. This support provides the capability for distributed parts of an application to ″converse″ with one another. The implementation is consistent with the Conversational function described in IBM's Open Blueprint.

Logical Unit Address (LUA)
System software and interfaces that supply input/output (I/O) service routines to support communications that use LU types 0, 1, 2, and 3 SNA protocols. These protocols support user-defined data streams, SNA character streams, and SNA 3270 data streams. LUA services include only those services that support data communications. LUA does not supply any device emulation facilities.

Node An endpoint of a link, or a junction, common to two or more links in a network. Nodes can be linked to host processors, communication controllers, cluster controllers, terminals, or workstations.

End Node
Provides directory and routing services for a workstation on an APPN network.

If the workstation will not be connecting from an end node to a network node server, you need to define an SNA connection.

APPC Concepts

Personal Communications provides Advanced Peer-to-Peer Networking (APPN) end-node support for workstations, allowing them to communicate more flexibly with other systems in the network.

Personal Communications provides advanced program-to-program communications (APPC) to support communications between distributed

processing programs, called *transaction programs* (TPs). APPN extends this capability to a networking environment. The TPs can be located at any node in the network that provides APPC.

Introducing APPC and CPI-C

Advanced Program-to-Program Communication (APPC), also known as LU 6.2, is software that enables high-speed communications between programs on different computers, from portables and workstations to midrange and host computers. APPC software is available for many different IBM and non-IBM operating systems, either as part of the operating system or as a separate software package.

APPC serves as an interface between application programs and the network. When the communications application on your workstation passes information to the APPC software, APPC takes the information and sends it on to a network interface, such as a token-ring adapter card. The information travels across the network to another computer, where the APPC software receives the information from the network interface. APPC puts the information back into its original format and passes it to the corresponding communications application.

APPC provides a consistent set of functions for program-to-program communications across different platforms. The APPC architecture did not, however, specify a common application programming interface (API) for implementing these functions. As a result, each operating system that supports APPC has its own API, a set of verbs that closely resemble the operating system itself. These differences do not pose a problem if you are writing programs for two computers that use the same operating system. But when you design client/server applications, split across different types of computers (each with its particular strengths and its particular kind of operating system), you must learn how to use two or more different sets of verbs.

The Common Programming Interface for Communications (CPI-C) eliminates this problem. CPI-C provides one standard set of verbs, known as CPI-C calls, for all systems that support CPI-C. As a result, you learn only one set of calls to write client/server applications for different systems.

What Is a Transaction Program?

The part of the communications application that initiates or responds to APPC communications is called a transaction program. A transaction program is not an entire, stand-alone program. Instead, it is the part of the program that handles transactions (exchanges of data) with another program.

A conversation between two programs is similar to a conversation between two people. In a conversation with another person, you follow unwritten rules that govern how you begin and end the conversation, take turns speaking, and exchange information. Similarly, the APPC protocol provides the rules that govern conversations between transaction programs: how conversations start and stop, which program “speaks” first, and how data is exchanged. Computers need complete and rigid rules for conversations between programs. For that reason, APPC consists of a set of well-defined and thorough rules to cover all possible communications situations.

An APPC program may have several conversations active at one time, with the same transaction program or with different transaction programs.

Every transaction program needs a partner to communicate with. Thus, transaction programs are developed in pairs called partner transaction programs.

People use different parts of speech to communicate with each other. Transaction programs are more limited in their communications; they use only verbs to communicate with each other. The verbs are the programming language you use to start, stop, and control conversations.

A transaction program consists of APPC and CPI-C verbs such as:

ALLOCATE (CMALLC in CPI-C)
Starts a conversation with another transaction program

SEND_DATA (CMSEND in CPI-C)
Sends data to the partner transaction program

RECEIVE (CMRCV in CPI-C)
Receives data from the partner transaction program

DEALLOCATE (CMDEAL in CPI-C)
Ends a conversation with another transaction program

The APPC verbs make up the application programming interface (API) for APPC and CPI-C. In other words, these verbs represent the interface between the transaction program and the APPC software.

What Is the Difference between APPC and APPN?

APPC is a communications protocol that enables programs on different computers to “talk to” each other. APPC provides the interface between the programs and the networking hardware and software and defines the rules that programs use to exchange information.

Advanced Peer-to-Peer Networking (APPN) is the underlying networking protocol that routes APPC traffic through intermediate nodes in the network. For instance, when Program A uses APPC to talk to Program B, APPN finds the node where Program B is located and directs the APPC traffic through the network.

APPN includes several features that help reduce the amount of configuration required to set up and maintain a network. These features automate many tasks that are time-consuming, complicated, and error-prone. For example, if you are installing a new workstation that uses APPN, you don’t have to set up configuration information for every workstation you want to communicate with. You simply provide the name of the computer and the address of the intermediate node that handles your traffic. APPN takes care of the rest of the information needed to route APPC traffic to and from your workstation.

If you connect to an APPN network, you simplify your own configuration and make it easier for other computers in the network to find you.

Side Information Definitions

Before starting a conversation with a partner program, a CPI-C program requires the following information:

- The name of the partner program
- The name of the LU on the remote computer
- The type of session used by the program
- Security information

If the CPI-C partner program uses a blank symbolic destination name, a side information definition is not necessary.

APPC Issues and Considerations

Why is APPC the protocol of choice for many client/server applications? Many users cite its openness, its advanced features, its common programming interface, and its exceptional performance. IBM and other vendors are constantly improving both the architecture and the products to ensure greater usability and performance in your network.

Improving Productivity Using APPC

The advanced features of APPC help speed the development of robust client/server applications and keep development costs low.

Client/server applications often prove to be especially challenging to the programmer. It is not enough to understand the details about the computer and operating system that your application uses. If you are designing client/server applications, you have to deal with computers used by both sides of the application, as well as the intervening network. For instance, implementing effective security measures is more important and more complicated, since you must ensure security throughout the network. In addition, synchronization and error reporting are often problematic in a distributed environment, because both sides of the application must coordinate their work.

One reason that APPC has become so widely used in client/server applications is that it furnishes a complete set of useful functions like security, synchronization, and error reporting. If you use other protocols, you must build these functions into every application you write. For example, if you use NetBIOS, you must design and implement a strategy for security for each application. Not only does this approach require additional work, it may result in a number of incompatible security systems in your network. By contrast, APPC includes a common set of security services that are consistent across all APPC platforms.

Similarly, most protocols do not include synchronization functions. Synchronization is required by any program that cannot continue processing data until the data sent to the partner program has been received and processed. Rather than writing your own synchronization routine, APPC provides a Confirm call that you can use to handle synchronization between two programs.

Improving Productivity Using CPI-C

Because many client/server applications are distributed between mainframe systems and workstations, host programmers and workstation programmers must work together to develop partner applications. CPI-C bridges the gap between these programmers by providing a common “language” for designing the communications part of the applications. CPI-C also enables you to write the communications portion of an application without knowing the details of the operating system. After you write a CPI-C application, you can easily move the programs from one operating system to another with few changes. So, by using the CPI-C interface, you can port both your applications and your programming skills from one system to another, quickly and cost-effectively.

What Is a Logical Unit?

Every TP gains access to an SNA network through a *logical unit* (LU). An LU is SNA software that accepts verbs from your programs and acts on those verbs. A

TP issues APPC verbs to its LU. These verbs cause commands and data to flow across the network to a partner LU. An LU also acts as an intermediary between the TPs and the network to manage the exchange of data between TPs. A single LU can provide services for multiple TPs. Multiple LUs can be active in the node simultaneously.

LU Types

Personal Communications supports LU types 0, 1, 2, 3, and 6.2. LU types 0, 1, 2, and 3 support communication between host application programs and different kinds of devices, such as terminals and printers.

LU 6.2 supports communications between two programs located at type 5 subarea nodes, type 2.1 peripheral nodes, or both, and between programs and devices. APPC is an implementation of the LU 6.2 architecture.

Communication occurs only between LUs of the same LU type. For example, an LU 2 communicates with another LU 2; it does not communicate with an LU 3.

When Personal Communications for Windows NT is configured, your local and partner LUs are defined once for each machine. When you design programs, you must understand the capabilities supported by the LU in the particular machines you plan to use.

Dependent and Independent LUs

A *dependent LU* depends on a system services control point (SSCP) to activate a session. A dependent LU needs an active SSCP-LU session, which the dependent LU uses to start an LU-LU session with an LU in a subarea node. A dependent LU can have only one session at a time with the subarea LU. For communications with a TP at a subarea node, each dependent LU can have only one conversation at a time, and each dependent LU can support communications for only one TP at a time.

An *independent LU* does not depend on an SSCP to activate a session. An independent LU supports multiple concurrent sessions with other LUs in a subarea node, so you can have multiple conversations and support multiple TPs for communications with subarea TPs. LUs between peripheral nodes also use this support.

The distinction between a dependent LU and an independent LU is meaningful only when discussing a session between an LU in a peripheral node and an LU in a subarea node. Otherwise, dependent and independent LUs both support multiple concurrent sessions and conversations when communicating between type 2.1 peripheral nodes, for example, between two Windows NT workstations. A Personal Communications for Windows NT LU can support a single session with a dependent LU or multiple sessions with an independent LU.

What Is a Session?

Before TPs can communicate with each other their LUs must be connected in a mutual relationship called a *session*. A session connects two LUs, so it is called an *LU-LU* session. Figure 1 on page 52 illustrates this communication relationship. Multiple, concurrent sessions between the same two LUs are called *parallel* LU-LU sessions.

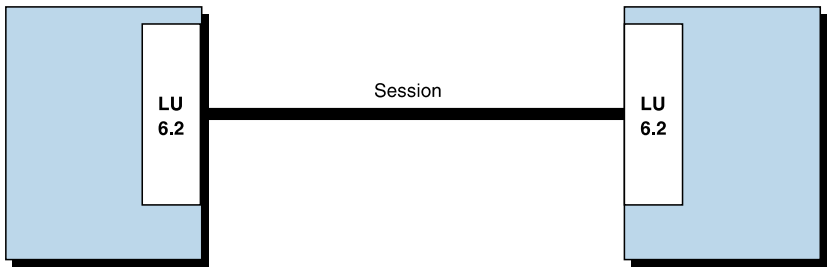

Figure 1. A Session between Two LUs

Sessions act as conduits that manage the movement of data between a pair of LUs in an SNA network. Specifically, sessions deal with things such as the quantity of data transmitted, data security, network routing, and traffic congestion.

Sessions are maintained by their LUs. Normally, your TPs do not deal with session characteristics. You define session characteristics when you:

- Configure your system
- Use the Session Services function in the Subsystem Management windows
- Use the management verbs

What Is a Conversation?

The communication between TPs is called a *conversation*. Conversations occur across LU-LU sessions. A conversation starts when a TP issues an APPC verb or CPI-C call that allocates a conversation. The conversation style associated with the conversation indicates the style of data transfer to be used, two-way alternate or two-way simultaneous. A conversation that specifies a two-way alternate style of data transfer is also known as a *half-duplex* conversation. A conversation that specifies a two-way simultaneous style of data transfer is referred to as a *full-duplex* conversation.

When a half-duplex conversation is allocated to a session, a send-receive relationship is established between the TPs connected to the conversation, and a two-way alternate data transfer occurs where information is transferred in both directions, one direction at a time. Like a telephone conversation, one TP calls the other, and they “converse”, one TP talking at a time, until a TP ends the conversation. One TP issues verbs to send data, and the other TP issues verbs to receive data. When it finishes sending data, the sending TP can transfer send control of the conversation to the receiving TP. One TP decides when to end the conversation and informs the other when it has ended.

When a full-duplex conversation is allocated to a session, both TPs connected to the conversation are started in send-and-receive state, and a two-way simultaneous data transfer occurs where information is transferred in both directions at the same time. Both TPs may issue verbs to send and receive data simultaneously with no transfer of send control required. The conversation ends when both TPs indicate they are ready to stop sending data, and each TP has received the data sent by the partner. If an error condition occurs, one TP may decide to end both sides of the conversation abruptly.

Conversations can exchange control information and data. The TP should select the conversation style best suited for its application.

A Conversation between Two TPs

Figure 2 shows a conversation between two TPs as it occurs over a session.

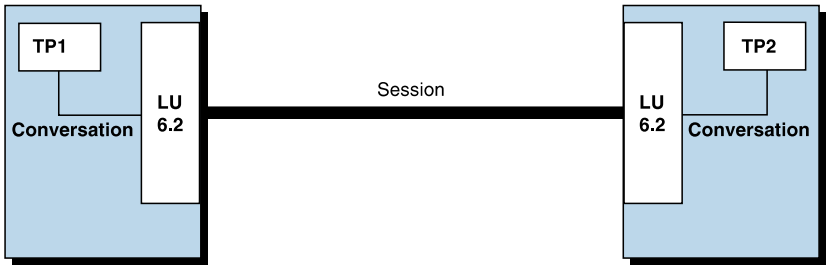

Figure 2. A Conversation between Two TPs

A session can support only one conversation at a time, but one session can support many conversations in sequence. Because multiple conversations can reuse sessions, a session is a long-lived connection compared to a conversation. When a program allocates a conversation and all applicable sessions are in use, the LU puts the incoming Attach (allocation request) on a queue. It completes the allocation when a session becomes available.

Two LUs can also establish parallel sessions with each other to support multiple concurrent conversations. A parallel session occurs when either TP allocates a conversation, and a session exists but is being used by a conversation. The LU can request a new session to satisfy the allocation.

Parallel Sessions between LUs

Figure 3 shows three parallel sessions between two LUs; each session carries a conversation.

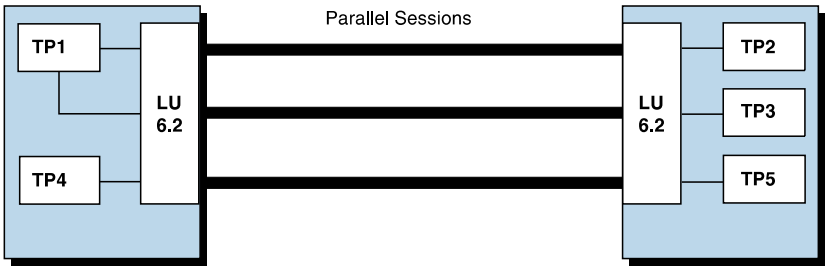

Figure 3. Parallel Sessions between LUs

SNA Communications

Personal Communications supports Systems Network Architecture (SNA) type 2.1 nodes (including SNA type 2.0 and SNA type 2.1 support for logical units, “LUs”, other than SNA LU 6.2). This support lets you write programs to communicate with many other IBM SNA products.

You can write programs without knowing the details of the underlying network. All you need to know is the name of the partner LU; you do not need to know its location. SNA determines the partner LU location and the best path for routing data. A change to the underlying network, such as a physical address change, the

addition of a new adapter, or the relocation of a machine, does not affect APPC programs. A program might, however, need to establish link connections over switched SDLC connections.

When Personal Communications for Windows NT starts, it establishes local LU and logical link definitions, which are stored in a configuration file. The system management application programming interface (API) provides functions that control configuration definition and adapter and link activation.

SNA Node Operations

With SNA Node Operations, either an end-user or system administrator can display information or take action on node resources that are known to the active node.

The following tasks are supported:
- Starting and stopping resources
- Deleting resources
- Displaying resource information
- Changing session limits
- Initiating path switches

See “Chapter 5. SNA Node Operations” on page 23 or the online help for information on using SNA Node Operations.

Appendix D. Introducing AnyNet SNA over TCP/IP

This appendix introduces the AnyNet SNA over TCP/IP function of Personal Communications.

Notes:

1. In the following examples, Windows 95 is shown; Windows 98 may be used in those situations, as well.
2. In the following examples, Windows NT is shown; Windows 2000 may be used in those situations, as well.

What Does AnyNet SNA over TCP/IP Do?

SNA over TCP/IP 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 therefore reduce operational complexity. These benefits are gained without modification to your existing application programs or hardware.

The AnyNet function in Personal Communications enables SNA applications to communicate over an IP network. This includes APPC and CPIC applications.

The following network configurations illustrate how SNA over TCP/IP access nodes and gateways can be used.

Running APPC or CPI-C Applications over a TCP/IP Network

Figure 4 illustrates how you can use the AnyNet SNA over TCP/IP function of Personal Communications to enable communication between SNA applications over an IP network.

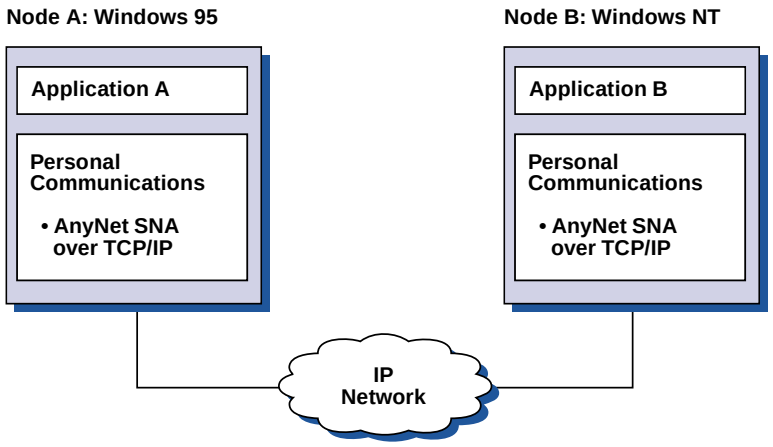

Figure 4. Running APPC or CPI-C Applications over a TCP/IP Network

Using AnyNet to Connect SNA Applications on Different Platforms

Figure 5 illustrates some of the many products with AnyNet SNA over TCP/IP function. AnyNet gateways (Node F) allow SNA applications to communicate across a combination of TCP/IP and SNA networks.

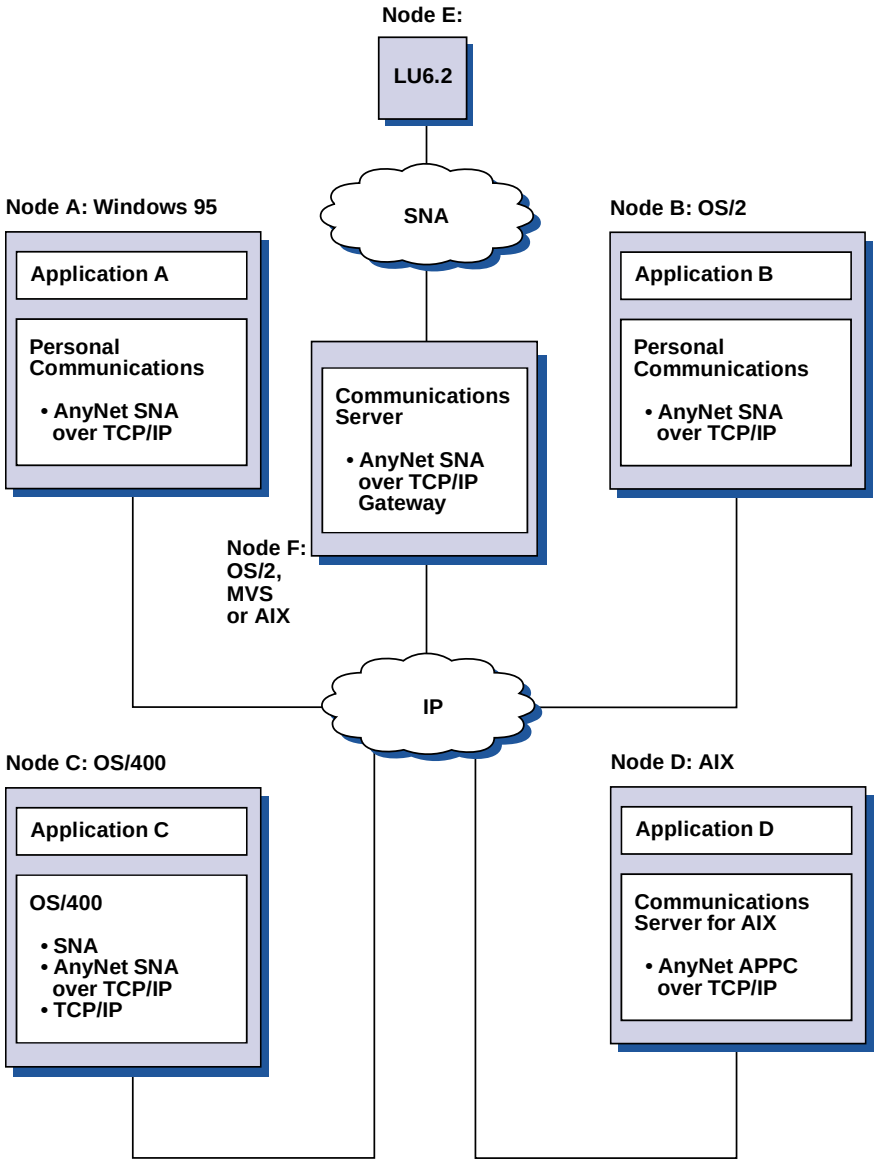

Figure 5. Using AnyNet to Connect SNA Applications on Different Platforms

Chaining Gateways to Connect Multiple SNA and IP Networks

Chained gateways enable the SNA network to connect multiple IP networks.

Note: While Figure 6 on page 57 shows three networks connected, the number of networks that can be concatenated is unlimited.

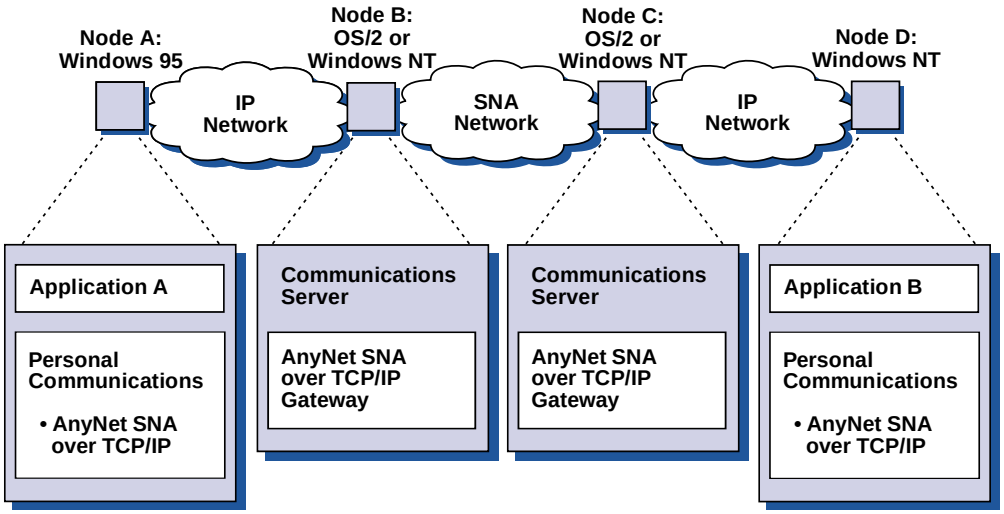

Figure 6. Chaining Multiple Gateways to Connect Multiple IP Networks

How Does SNA over TCP/IP Work?

SNA over TCP/IP uses protocols that bypass the lower transport layers of the SNA architecture. This process is transparent to the SNA application program. Using a supported API, an application program passes data to the presentation services of Personal Communications. This data, in turn, is passed through the SNA architectural layers and presented to SNA over TCP/IP at the data link control level. When Personal Communications initiates a session for an application program, SNA over TCP/IP translates the SNA routing information (network-qualified name) into IP routing information (IP address) and uses the IP address to create a TCP connection to the appropriate system.

Figure 7 on page 58 illustrates how an SNA over TCP/IP access node enables SNA application programs to communicate over IP networks.

AnyNet SNA over TCP/IP

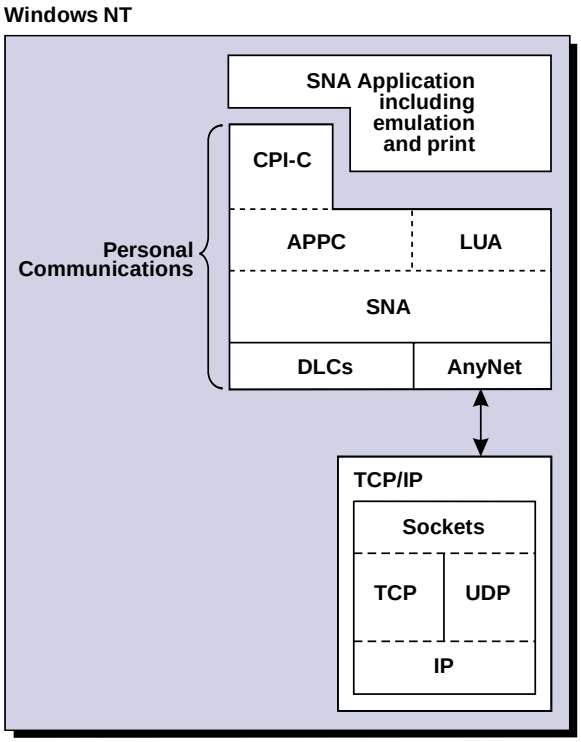

Figure 7. Structure of SNA over TCP/IP for Personal Communications

The configuration information you define enables SNA over TCP/IP to determine:

- Whether to route the data using SNA or to route the data using IP
- The IP address associated with the network-qualified LU name

SNA over TCP/IP uses both stream (TCP) and datagram (UDP) sockets that are bound to the well-known port (397). Any information received over this port is routed to AnyNet.

Mapping SNA Resources to IP Addresses

SNA over TCP/IP access nodes must determine the IP address of the partner before an SNA session can be established. The SNA identifier of the partner is mapped to an IP address using the following steps:

1. SNA over TCP/IP receives the SNA identifier from Personal Communications in one of the following formats:
   - For LU names, `netid.luname`
   - For CP names, `netid.cpname`
2. SNA over TCP/IP takes the identifier and generates a domain name:
   - For LU names, `luname.netid.snasuffix`
   - For CP names, `cpname.netid.snasuffix`

   Note: The default value for `snasuffix` is **SNA.IBM.COM**. For additional information on the SNA domain name suffix, refer to the on-line help.

   Figure 8 on page 59 shows examples of domain names generated by SNA over TCP/IP.

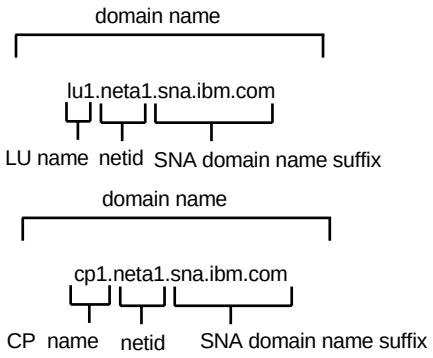

Figure 8. Formats of the Domain Names That SNA over TCP/IP Builds

3. SNA over TCP/IP requests that the domain name be translated into an IP address.
4. TCP/IP uses the HOSTS file or domain name server to translate the domain name into an IP address (for example, 9.67.192.28).

When the IP network includes SNA over TCP/IP gateways, consider the following additional address mapping issue:

If the partner LU can be reached through a gateway, the domain name of the partner must be mapped to the IP address of the gateway. If there are parallel gateways, the domain name should be mapped to each gateway IP address.

Defining Domain Names and IP Addresses

This section describes the TCP/IP name resolution function, used by AnyNet to map SNA resources to IP addresses. This function queries both the local HOSTS file and any domain name servers to convert a domain name (for example, lu1.neta1.sna.ibm.com) into an IP address (for example, 10.1.1.1).

HOSTS File

You can use TCP/IP HOSTS files to define domain names and IP addresses for your network. However, as your network becomes larger and maintaining the HOSTS file on each end-user workstation becomes too time-consuming, it is recommended that you use a domain name server.

The HOSTS file (in the drivers\etc subdirectory of your NT system directory) lists:

- IP address
- Domain name
- Other aliases for the domain name

For example, if your IP address is 10.1.1.1, network ID is NETA1, SNA resource name is LUA1, and SNA domain name suffix is the default (sna.ibm.com), enter the following in your HOSTS file:

```
10.1.1.1      lua1.neta1.sna.ibm.com
```

Domain Name Server

Domain names and IP addresses can also be defined in the domain name server database.

Each SNA identifier is mapped to a corresponding IP address by a domain name server. The location of these servers is configured in the Network section of the Control Panel.

For more information on HOSTS files and domain name servers, refer to your TCP/IP documentation. If your workstation is using the TCP/IP support in Windows 95, Windows 98, Windows NT, or Windows 2000, refer to the on-line TCP/IP documentation that is included with the Windows 95, Windows 98, Windows NT, or Windows 2000 product.

Defining Unique CP Names and Connection Network Names

For configurations that have two or more SNA over TCP/IP gateways connecting an SNA network with two or more IP networks, you must define a unique SNA control point (CP) name and a unique SNA connection network name for each IP network.

All LUs that reside on access nodes in the IP network appear to reside on a node with this CP name.

Use the reverse data file of the domain name server or the HOSTS file to define the CP name and the connection network name for a given IP network. Map the IP address 127.0.0.3 to the CP name and map the IP address 127.0.0.4 to the connection network name.

The following example shows entries in the reverse data file. For an IP network with SNA network ID NETA, CP name MYCPNAME, and connection network name MYCNET, you would define the following entries:

```
127.0.0.3       NETA.MYCPNAME.
127.0.0.4       NETA.MYCNET.
```

Notes:

1. A period is required at the end of the name only if the definition is in the DNS reverse data file. No period is used in HOSTS file definitions.
2. Do not include the SNA domain name suffix.

The AnyNet SNA over TCP/IP function of Personal Communications provides a default CP name ($ANYNET.$GWCP) and a default connection network name ($ANYNET.$GWCNET). In configurations with one IP network, you can use the default by not defining a CP name or a connection network name. In configurations with multiple gateways connecting multiple IP networks, one IP network can use the default. You must, however, define a unique CP name and connection network name for all other IP networks.

Figure 9 on page 61 shows how to define the CP name and the connection network name for a configuration with two IP networks.

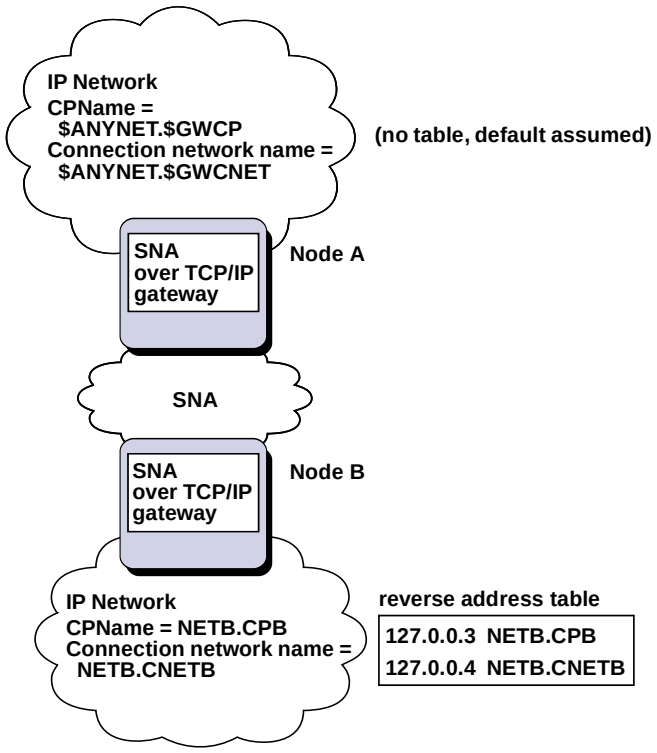

Figure 9. Defining a CP Name and a Connection Network Name

Setting and Modifying the Routing Preference Protocol

When an SNA application initiates a session, Personal Communications must first determine which transport to use, either SNA, IP, or a combination of the two.

Personal Communications uses the default routing preference for the session unless there is a specific routing preference set for the partner LU. Personal Communications then sets up a session for the partner LU using the preferred transport.

If the partner is not available through the preferred transport, Personal Communications can reroute the session request over the other available transport, depending on the setting in the routing preference table. For example, if the setting in the routing preference table for the destination LU is non-native first, Personal Communications will attempt to set up the session over TCP/IP. If the partner is not available through TCP/IP, Personal Communications will route the session over SNA. However, if the setting in the routing preference table for the destination LU is non-native only, Personal Communications will attempt to set up the session over TCP/IP. If TCP/IP is not available, Personal Communications will not route the session over SNA, because the setting allows only Personal Communications to use non-native transport.

The routing preference table is used only for new sessions. Previously existing sessions use the same transport; they are not brought down and rerouted if the routing preference table is changed.

You can set or modify the default routing preference to one of the following:

Native first
Requests are routed over SNA. If no SNA route is available, requests are routed over TCP/IP.

Non-native first
Requests are routed over TCP/IP. If no TCP/IP route is available, requests are routed over SNA.

Native only
Requests are routed only over SNA. If no SNA route is available, the connection will fail.

Non-native only
Requests are routed only over TCP/IP. If no TCP/IP route is available, the connection will fail.

Helpful Hints

This section contains helpful hints on tuning, TCP/IP connectivity via SLIP or PPP, and dynamic IP addresses.

Tuning

If you can access an LU through multiple SNA over TCP/IP gateways, and you have mapped that LU name to multiple IP addresses, increase the Connection retry duration and Connection wait time limit when configuring the AnyNet base parameters. This ensures that TCP attempts to connect to all possible adapters and gateways. It can take as long as 90 seconds for a TCP connection to fail to an inactive IP address.

Dynamic IP Addresses

Generally, AnyNet SNA over TCP/IP depends on SNA resources (for example, LU names, CP names or idblk/num) being statically mapped to IP addresses. However, depending on your configuration and how your connections are initiated, you might be able to use AnyNet SNA over TCP/IP in environments where IP addresses are dynamically assigned (for example, DIICP).

APPC or CPIC Applications

An SNA over TCP/IP access node with a dynamically assigned IP address may always initiate sessions to another SNA over TCP/IP access node or gateway with a static IP address.

The only way an SNA over TCP/IP access node or gateway with a static IP address (node A) can initiate a session to a partner with a dynamic IP address (node B) is:

- Node B initiated a session to or through node A first.
- The session initiated in Step 1 is still active.

Dependent LU Applications

Note: The information in this section applies to LU 0, 1, 2, 3, or dependent 6.2 applications.

SNA over TCP/IP access nodes with dynamically assigned IP addresses can support dependent LU communications if the following criteria are met.

- Dependent LU communication is occurring through an SNA gateway either through:

  - Communications Server/2 V4.1 with APAR applied
  - Communications Server/2 V5
  - Communications Server for Windows NT 5.0 or later
- The SNA over TCP/IP access node with a dynamically assigned IP address initiates connections to an SNA gateway with a static IP address.

Dependent LU communication through DLUS/DLUR over AnyNet SNA over TCP/IP is not supported if the DLUR node has a dynamically assigned IP address.

Appendix E. AnyNet Sockets over SNA

This chapter contains information about how to configure Sockets over SNA. The Sockets over SNA access node function of Personal Communications enables WinSock compliant applications to communicate over SNA networks. .

Notes:

1. In the following examples, Windows 95 is shown; Windows 98 may be used in those situations, as well.
2. This function is not provided on Windows.

AnyNet Sockets over SNA Access Node

The Sockets over SNA access node function enables TCP/IP application programs using either the WinSock 1.1 or WinSock 2.0 socket interface to communicate over an SNA network.

Socket Applications Communicating over an SNA Network

Figure 10 illustrates a network where Sockets over SNA is configured on each node, but no Sockets gateway function is needed. Socket applications A and B can exchange information as long as they are connected through an SNA network. The SNA transport can take full advantage of APPN and HPR for enhanced reliability and performance.

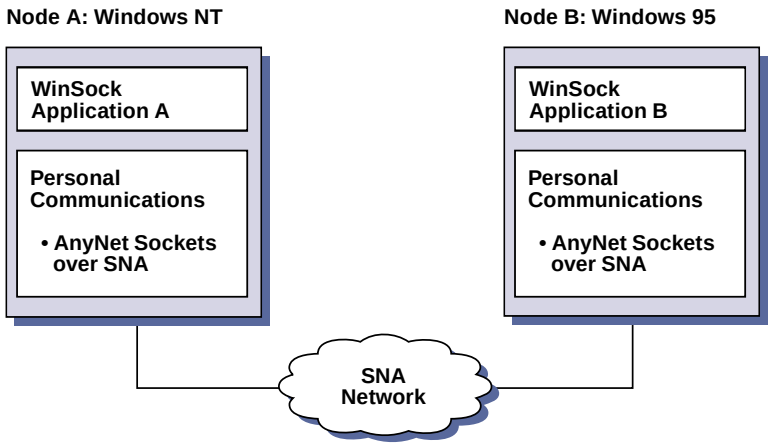

Figure 10. Sockets Applications Communicating over an SNA Network

Connecting Sockets over SNA Access Nodes on Different Operating Systems

Figure 11 on page 67 illustrates an SNA network in which socket applications on multiple operating systems are communicating over an SNA network. In this example:

- Node A is configured with Sockets over SNA which is provided as part of Personal Communications for either Windows 95 or Windows NT.
- Node B is configured with Sockets over SNA for OS/2 which is also provided with Communications Server for OS/2.

- Node C is configured with Sockets over SNA for MVS which is provided with the VTAM Multiprotocol Transport Feature (MPTF) or the VTAM AnyNet feature.
- Node D is configured with Sockets over SNA for AIX which is provided as part of the Communications Server for AIX.
- Node E is configured with OS/400, which includes the Sockets over SNA function.
- Node F is configured with Windows NT and Sockets over SNA, which is provided with Communications Server for Windows NT.

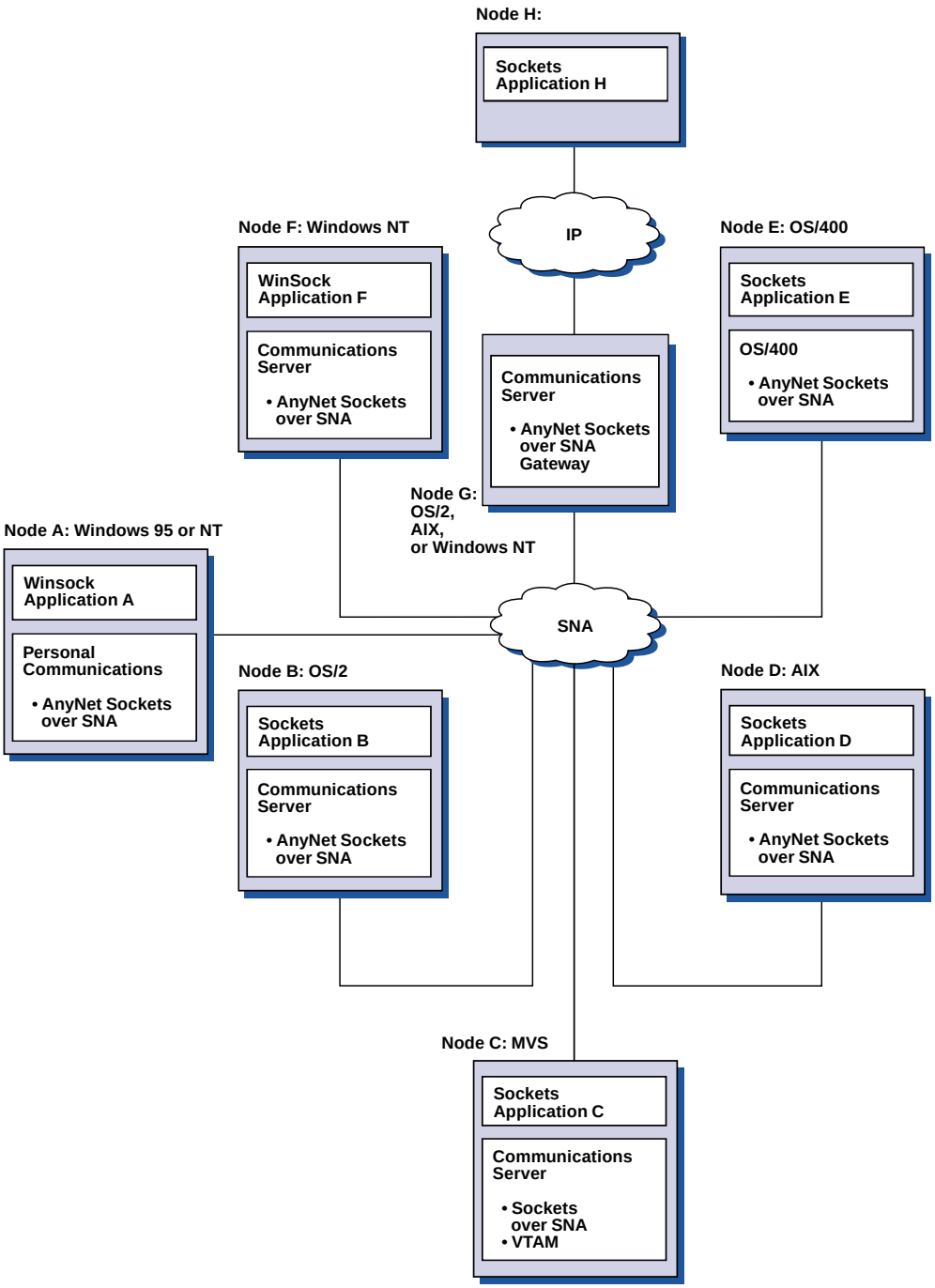

Figure 11. Socket Applications on Multiple Operating Systems Communicating over an SNA Network

Connecting Remote TCP/IP Networks Using an SNA Backbone and Sockets over SNA Gateways

Sockets over SNA access nodes can communicate with a native TCP/IP network through a Sockets over SNA gateway (as provided by IBM Communications Server products).

AnyNet Sockets over SNA

In Figure 12, Node A is configured with TCP/IP and runs on a remote, native TCP/IP network. Node C is configured as a Sockets over SNA Gateway. Node B is an access node configured with Sockets over SNA and runs on an SNA network.

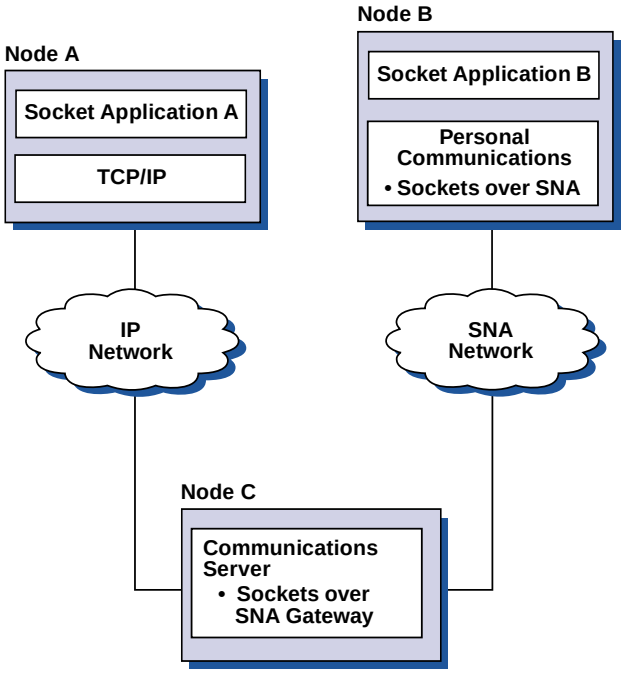

Figure 12. Socket Applications Communicating through Multiple Sockets Gateways

How Does Sockets over SNA Work?

Figure 13 on page 69 shows the structure of a Windows NT node that is running Sockets over SNA and illustrates how socket application programs and Sockets over SNA operate on a Windows NT node. Similarly, Figure 14 on page 70 shows the structure of a Windows 95 node that is running Sockets over SNA.

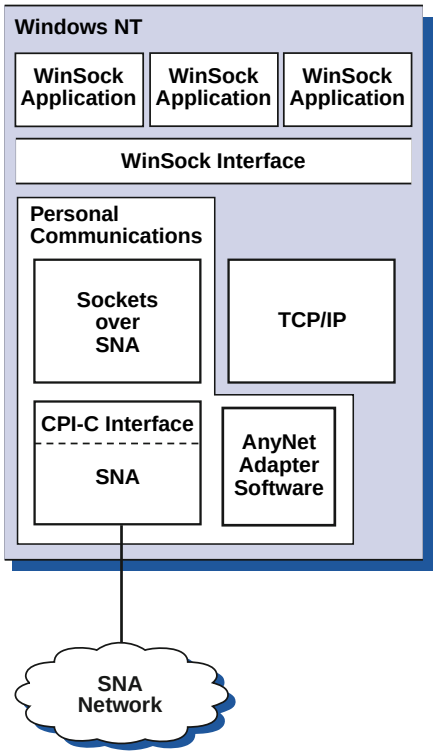

Figure 13. Structure of a Windows NT Node Running Sockets over SNA

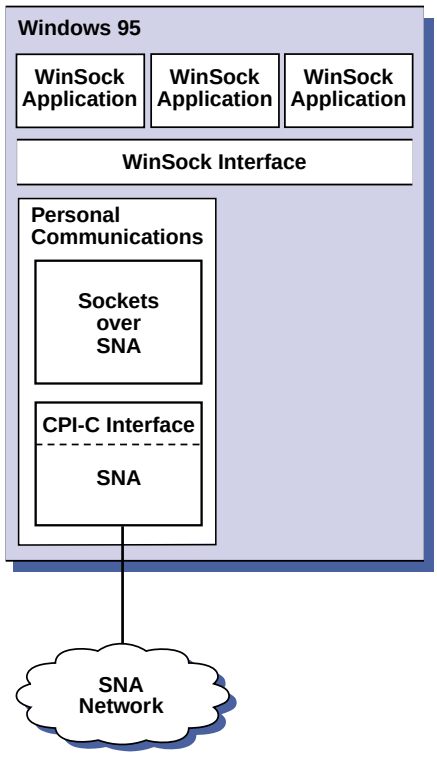

Figure 14. Structure of a Windows 95 Node Running Sockets over SNA

WinSock is an API that allows socket applications to run in a Windows environment. Windows NT 4.0 supports Version 2 of the WinSock API. Windows 95 supports version 1.1 of WinSock. WinSock 2.2 is available from Microsoft.

Sockets over SNA does not provide a WinSock interface and does not process socket calls. Instead, WinSock applications use the WinSock interface. The Sockets over SNA code then allows these applications to communicate across the SNA network.

Note: As a part of the installation of the AnyNet® Sockets over SNA component, AnyNet Sockets over SNA will install its own layered service provider (LSP). The AnyNet LSP will form a new service provider protocol chain which will exclude all other currently installed LSPs. This means applications using an existing LSP may no longer work.

Likewise, if another service provider is installed after installation of AnyNet Sockets over SNA, it is possible that the AnyNet Sockets over SNA LSP will be left out of the new protocol chain. It is also possible for a new service provider to insert its LSP into the protocol chain in such a manner as to cause the AnyNet Sockets over SNA code to fail.

This is a known problem with the current WinSock 2 Specification. At this time there is still no architected solution to the problem of managing the ordering of multiple service providers in protocol chains. Until such a solution is made available for the WinSock 2 Specification, there will be continued problems with multiple service providers.

Personal Communications users who are experiencing this problem should try the following workarounds:

If you do not need to run the AnyNet Sockets over SNA function:

- At installation time use the **Custom Install** option and uncheck the **AnyNet Sockets over SNA** box.
- If install has already been performed, using **Full Install** or **Custom Install** with AnyNet Sockets over SNA:
  1. Go to a DOS command prompt and switch to the directory where Personal Communications is installed.
  2. Issue the command **ANYWS2 -h**.
  3. Follow the help information to uninstall the AnyNet LSP.
  4. If you later need to use AnyNet Sockets over SNA, issue the **ANYWS2 -h** command and follow the directions to install and reorder the AnyNet LSP.

If you need to run with the AnyNet Sockets over SNA function and another service provider:

- In some cases, it will be possible to run with both service providers.
  1. Ensure that AnyNet Sockets over SNA is installed first.
  2. Install the new service provider.

  If the new service provider is of the correct type *and* creates a new protocol chain correctly, both AnyNet Sockets over SNA and the new service provider will work
- In most cases this will not work, and you will have to determine which service provider you want to work with for a given work scenario; to accomplish this:
  1. Go to a DOS command prompt and switch to the directory where PCOMM is installed.
  2. Issue the command **ANYWS2 -h**.
  3. Follow the help information to uninstall or install and reorder the AnyNet LSP.

.

Generating an LU 6.2 Call from a Socket Call

To enable TCP/IP-formatted information to route over SNA, Sockets over SNA maps IP addresses to SNA network-qualified LU names. When an application program invokes Sockets over SNA to establish a stream connection with another application program, Sockets over SNA establishes two half-duplex LU 6.2 conversations for the stream connection.

Sockets over SNA establishes one LU 6.2 conversation for all datagrams sent to a single destination. Conversations dedicated to datagram traffic are deallocated if they are unused for some specified period of time.

Mapping an IP Address to an SNA Network-Qualified Name

When an application program invokes Sockets over SNA to communicate with another application program, it supplies the IP address of the destination node. Sockets over SNA must map the IP address to an SNA address to issue an appropriate LU 6.2 call. For every IP address that identifies a node, there will be a corresponding SNA network-qualified name.

"Routing and Mapping Overview" explains how address mapping works and provides guidelines and requirements for setting up IP-LU address mapping.

Application Program Support Provided by Sockets over SNA

Sockets over SNA supports WinSock 1.1 and WinSock 2.0 applications that use AF_INET sockets. Sockets over SNA does not support applications that use broadcasting.

Planning for Sockets over SNA

This section describes what the network planner should consider before configuring a network with Sockets over SNA.

Routing and Mapping Overview

This section explains basic concepts of Internet addressing and how those concepts relate to routing and mapping. It includes the following information:

Internet Addressing

Every host is assigned at least one unique Internet Protocol (IP) address, which is used to route data through the network.

Note: In the IP suite of protocols, *host* refers to an end system and can be any workstation; it does not have to be a mainframe.

The IP address assigned to the host does not define a host on the network; it defines a network interface on that host to a network. For example, the address of the SNA network interface identifies a node's connection to the SNA network. You must configure a unique IP address for the SNA interface.

The following section describe the IP address format, address classes, and network masks. For more detailed information, see your TCP/IP documentation.

IP Address Format and Classes

An IP address consists of a 2-part, 32-bit address field:

- The first part of the address field contains the network address; the second part contains the host address.
- The number of bits used for the network and host portions of an IP address are variable, depending on the address class of the IP address.
- A network mask enables you to use part of the host portion of the IP address as a subnetwork address.

Default network masks are shown in Table 2.

Table 2. IP Address Masks Supported by Sockets over SNA

For a dotted decimal IP address in the form *a.b.c.d*, the range of values for *a* is:	**Default Network Mask**
1–127	255.0.0.0
128–191	255.255.0.0
192–223	255.255.255.0

Masks Used by Sockets over SNA

Sockets over SNA uses two types of masks:

- **Subnet mask**

  The subnet mask is used in routing and is specified during the configuration of the local node and routes. You can accept the default subnet mask or specify a value other than the default to define subnetwork addresses.

- **Address mask**

  The address mask is used for generated IP-LU address mapping and is specified during configuration.

IP Routing Table

Each host has an IP routing table that stores information about possible destinations and how to reach them. Route entries are added when:

- The local IP address is defined. For more information, see the online help.
- Routes are defined from the SNA Node Configuration window. Click **Configure AnyNet Sockets over SNA** and then click **Routes**. For more information, see the online help.
- A route with fewer hops is found. This occurs when Sockets over SNA receives an ICMP redirect message.

For a routing table example, see Figure 15 on page 74.

SNA Network ID Used by Sockets over SNA

For each route you define through the SNA interface (sna0), there must be a corresponding SNA network ID to which the destination 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 IP 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 a unique one-to-one mapping between an IP network address and an SNA network ID.

How an IP Address Is Mapped to an LU Name

You can use either explicit or generated mapping to map IP addresses to SNA LU names:

- In explicit mapping, all LU names are defined, and you assign each IP host address to a specific SNA LU name during configuration.
- In generated mapping, LU names are automatically generated for you. During configuration, you specify a network IP address, address mask, network ID, and LU template value.

Sockets over SNA uses the address mask to map the network portion of the IP address to the SNA network ID and the host portion to the SNA LU name. The LU template value is used to determine the characters and the positions of characters used in the LU name.

You can display the generated LU name for a given IP address using the **sxmap** command line utility. The syntax for this utility is:

```
sxmap convert <IP address> <address mask> <LU template>
```

How Sockets over SNA Routes and Maps Data

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

1. Sockets over SNA searches its own 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, and if the node is configured for native access, the connection request is forwarded to the native TCP/IP stack.
2. If Sockets over SNA finds a matching route, the route entry indicates how the destination can be reached:
   a. If the router address is the address of a local network interface, such as sna0, the destination network, subnetwork, or host address can be reached directly.
   b. 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 15 shows an example of an IP routing table.

Destination IP Address	Destination Mask	Gateway IP Address	Use Count
1.2.3.4	255.255.255.255	199.245.253.1	10
10.0.0.0	255.0.0.0	199.245.253.2	0
10.11.0.0	255.255.0.0	199.245.253.113	37
127.0.0.1	255.255.255.255	127.0.0.1	8
128.1.0.0	255.255.0.0	199.245.253.3	0
199.245.253.0	255.255.255.0	199.245.253.113	368

Figure 15. Example of an IP Routing Panel

3. If no route is found in the Sockets over SNA routing table, and native TCP/IP access is also configured, then Sockets over SNA assumes that the TCP/IP destination 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:
   a. If Sockets over SNA finds a matching entry, an LU 6.2 connection is established.
   b. If Sockets over SNA does not find a matching entry, the connection attempt fails and Sockets over SNA returns completion error code WSAEHOSTUNREACH.
   c. Sockets over SNA passes the destination SNA address and data to Personal Communications.

Defining Sockets over SNA Modes

Sockets over SNA uses LU 6.2 conversations to enable communication between socket application programs. When an LU 6.2 conversation is established, Sockets

over SNA defines the mode and associated session characteristics of the connection. Personal Communications uses the mode name to identify characteristics of the connection between the two Sockets over SNA nodes.

The default Sockets over SNA mode is BLANK. You can use the default mode for Sockets over SNA or define your own. To change the default Sockets over SNA mode, from the SNA Node Configuration window, click **Configure AnyNet Sockets Over SNA** and then click **Modes**. You can define another default mode for all TCP/IP traffic and assign a specific mode to a specific TCP/IP port.

If you specify an alternative mode that is not defined by Personal Communications, you must define the session characteristics associated with that mode to Personal Communications.

Changing the Idle-Timeout Interval

The Idle Timeout start option allows you to adjust the number of idle seconds before Sockets over SNA deallocates a datagram conversation. This interval enables you to balance between using system resources to maintain an existing datagram conversation and taking longer to reestablish a new datagram conversation. For example, if you set this value low, unused datagram conversations end faster, but it takes longer to send the next datagram. The default idle-timeout interval is 90 seconds.

To modify the start option, from the SNA Node Configuration window, click **Configure AnyNet Sockets over SNA**, then click **View/Change/Add**, click **Advanced** and select a new value for this option.

Appendix F. Using InstallShield Silent

InstallShield Silent allows for the automated installation of software, also known as silent installation. With InstallShield Silent, you do not need to monitor the installation or provide input via dialog boxes. An InstallShield Silent installation runs unattended, without displaying any indication that it is running.

Setting Up InstallShield Silent

The file SETUP.EXE is the main program file for installing Personal Communications. It uses the following optional command line parameters, which must be preceded by either a slash (/) or a dash (–). When you use long path and file name expressions, enclose the expressions in double quotes.

Parameter
Description

–f1<path\ResponseFile>
Use this option to specify the alternate location and name of the response file (.ISS file). If you use this option when running Installshield Silent, the response file is read from the directory or file specified by <path\ResponseFile>.

The argument to –f1 currently cannot be longer than 43 characters. If your response file is a pathname longer than 43 characters, you can move the response file to a directory that results in a shorter pathname.

–f2<path\LogFile>
Use this option to specify the alternate location and name of the log file created by Installshield Silent. By default, SETUP.LOG is created and stored in the same directory as SETUP.EXE. You must have write access to the location where SETUP.LOG will be created.

–noinst
Use this option to cause setup to prompt you for installation choices and then to terminate without performing an actual installation. If you use –noinst it must be the first parameter after setup. Normally you will use –noinst with –r.

–r Use this option to automatically record a silent installation response file (.ISS file) from an attended install. This is a record of the installation input and is stored, by default, in the windows directory.

–s Use this option to run a silent installation.

Usage Examples

The following examples illustrate possible uses of SETUP.EXE for silent installation:

setup –s
Launches Installshield Silent and loads SETUP.ISS from the directory containing SETUP.EXE. The log file SETUP.LOG is created in the same directory.

setup –s –f1c:\mydir\mydir.iss
Launches Installshield Silent and uses MYDIR.ISS (from the C:\MYDIR directory) as the response file. This example also creates the log file SETUP.LOG in the same directory as SETUP.EXE .

setup –s –f1c:\mydir\mydir.iss –f2c:\mydir\mydir.log
Launches Installshield Silent, uses MYDIR.ISS from the C:\MYDIR directory, and generates the log file MYDIR.LOG in the C:\MYDIR directory.

setup –noinst –r
Records a response file without installing the product. The purpose is to generate a response file for subsequent installations without doing an actual installation on the current system. By default, the response file is named SETUP.ISS and is located in the Windows directory.

setup –noinst –r –f1c:\mydir\mydir.iss
Records a response file without installing the product and generates a response file MYDIR.ISS in the C:\MYDIR directory

Usage Notes

- The parameters f1 and f2 cannot be followed by a space.
- When you run Installshield Silent, a log file is created in the directory where SETUP.EXE is located. By default, the log file has the default name SETUP.LOG if the –f2 parameter is not provided along with –f1.
- If you do not use the –f1 parameter when running Installshield Silent, the response file SETUP.ISS is assumed to be in the same directory as SETUP.EXE. A log file is created in the same directory.
- SETUP.EXE command line parameters are not case-sensitive.

Appendix G. Notices

This information was developed for products and services offered in the United States. IBM may not offer the products, services, or features discussed in this information in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this information. The furnishing of this information does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
U.S.A

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS INFORMATION AS IS WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the information. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this information at any time without notice.

Any references in this information to non-IBM documentation or non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those documents or Web sites. The materials for those documents or Web sites are not part of the materials for this IBM product and use of those documents or Web sites is at your own risk.

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
Department T01
Building 062
P.O. Box 12195
Research Triangle Park, NC 27709-2195

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries, or both:

400
ACF/VTAM
AFP
AIX
AIXwindows
APL2
APPN
AS/400
AT
Advanced Peer-to-Peer Networking
AnyNet
Application System/400
BookManager
CICS
CICS/MVS
CICS/VSE
COBOL/2
CUA
Common User Access
Client Access
Client Access/400
Current
DB2
DB2/400
DRDA
DisplayWrite
Distributed Relational Database Architecture
ExecJet
GDDM
Global Network
IBM
IBMLink
IIN
IMS
InfoWindow
MVS/ESA
MVS/XA

Microsoft, Windows, Windows NT©, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

PC Direct is a registered 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.

C-bus is a trademark of Corollary, Inc.

Trademarks

ActionMedia, LANDesk, MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Glossary

The complete *IBM Dictionary of Computing* is available on the World Wide Web at **www.networking.ibm.com/nsg/nsgmain.htm**.

A

access control. In computer security, ensuring that the resources of a computer system can be accessed only by authorized users in authorized ways.

access control profile. A list of the access privileges assigned to users and groups for a particular network resource in a domain.

access priority. In the IBM Token-Ring Network, the maximum priority a token can be given by the adapter for transmission.

adapter. A part that electrically or physically connects a device to a computer or to another device.

Advanced Communications Function (ACF). A group of IBM licensed programs, principally VTAM, TCAM, NCP, and SSP, that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing.

advanced program-to-program communication (APPC). (1) The general facility characterizing the LU 6.2 architecture and its various implementations in products. (2) Sometimes used to refer to the LU 6.2 architecture and its product implementations as a whole, or to an LU 6.2 product feature in particular, such as an APPC application programming interface.

AID key. A control key that generates a host attention interrupt.

alert. (1) A message sent to a management services focal point in a network to identify a problem or an impending problem. (2) In SNA management services (SNA/MS), a high priority event that warrants immediate attention.

alias name. A name that is defined in one network to represent a logical unit name in another interconnected network. The alias name does not have to be the same as the real name; if these names are not the same, translation is required.

American National Standard Code for Information Interchange (ASCII). A character set standard established by the American National Standards Institute, using a set of 7-bit binary-coded characters (8 bits including parity check). ASCII is used for information interchange among data processing systems, data communication systems, and associated equipment. An extended, 8-bit version of this code is used in the personal computer. It consists of control characters and graphic characters in addition to the 7-bit ASCII characters. See also *extended binary-coded decimal interchange code.*

American National Standards Institute (ANSI). An organization consisting of producers, consumers, and general interest groups, that establishes the procedures by which accredited organizations create and maintain voluntary industry standards in the United States. (A)

ANSI. American National Standards Institute.

AnyNet product family. A group of IBM products that implement the multiprotocol transport networking (MPTN) architecture, thus enabling application programs to communicate independently of the underlying network transport protocol.

API. Application programming interface.

APL. A programming language requiring the use of a special keyboard to represent various operator symbols.

APPC. Advanced program-to-program communications.

application. A collection of software components used to perform specific types of user-oriented work on a computer.

application program. (1) A program written for or by a user that applies to the user's work, such as a program that does inventory control or payroll. (2) A program used to connect and communicate with stations in a network, enabling users to perform application-oriented activities.

application programming interface (API). (1) The set of programming language constructs or statements that can be coded in an application program to obtain the specific functions and services provided by an underlying operating system or service program. (2) In VTAM, the language structure used in control blocks so that application programs can reference them and be identified to VTAM.

application transaction program. A program written for or by a user to process the user's application; in an SNA network, a user of a type 6.2 logical unit. Contrast with *service transaction program*.

APPN. Advanced Peer-to-Peer Networking.

APPN end node. See *Advanced Peer-to-Peer Networking (APPN) end node.*

APPN network node. See *Advanced Peer-to-Peer Networking (APPN) network node.*

ASCII (American National Standard Code for Information Interchange). The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), that is used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

ASYNC. Asynchronous.

asynchronous (ASYNC). (1) Pertaining to two or more processes that do not depend upon the occurrence of specific events such as common timing signals. (T) (2) Without regular time relationship; unexpected or unpredictable with respect to the execution of program instructions.

Asynchronous Data Link Control (ASYNCH). A discipline for managing asynchronous, serial-by-bit information transfer over a line connection.

asynchronous transmission. A data communication method wherein character strings of variable length are transmitted serially at a variable transmission rate. Each character or defined block of characters has a constant transmission duration.

atom. Represents a character string that has been reduced to a unique integer value.

attention identifier. See *AID key*.

attachment. A communication link used by Personal Communications to connect a personal computer to a host system, consisting of an adapter and controlling software.

audit trail. A history of network activity on a server.

AUTOEXEC.BAT file. In the DOS operating system, a batch file that resides in the root directory of the boot drive. AUTOEXEC.BAT contains commands that DOS executes every time the PC is booted.

automatic access utility. A utility that allows you to create and edit the files that control the automatic-dial, automatic-logon, and modem control programs.

autoskip. A field defined as protected **and** numeric. Causes the cursor to skip to the next unprotected field.

B

backup diskette. A diskette that contains information copied from a Personal Communications diskette.

base-color mode. The mode in which a terminal operates if the host system does not supply color information.

Basic Input/Output System (BIOS). Code that controls basic hardware operations, such as interactions with diskette drives, hard disk drives, and the keyboard.

batch file. In IBM DOS, a file containing commands to be processed sequentially. A DOS batch file must have a **.BAT** extension in order to be executable.

binary. Pertaining to the base two system of numbers. The binary digits are 0 and 1. Executable files are generally in binary format rather than the character string format that text files are composed of.

BIOS. (1) Basic Input/Output System. (2) See also *NetBIOS*.

bis. A suffix used to designate the second version of a standard, as in ITU-T Recommendation V.25 bis.

bit map. A representation of an image by an array of bits.

C

Caps Lock. The personal computer keyboard mode entered after the **Caps Lock** key is pressed. This mode is indicated by a capital A in the operator information area (OIA). When the **Shift key** is pressed while in this mode, **Caps Lock** is temporarily discontinued and the OIA indicator changes. Pressing the **Shift** key again returns the Caps Lock mode.

character cell. The maximum physical boundary of a character on a display screen.

character set. A finite group of characters defined for a keyboard or output device.

code page. (1) An assignment of graphic characters and control function meanings to all code points; for example, assignment of characters and meanings to 256 code points for an 8-bit code, assignment of characters and meanings to 128 code points for a 7-bit code. (2) A particular assignment of hexadecimal identifiers to graphic characters.

Common Programming Interface for Communications (CPI-C). An evolving application programming interface (API), embracing functions to meet the growing demands from different application environments and to achieve openness as an industry standard for communications programming. CPI-C provides access to interprogram services such as (a) sending and receiving data, (b) synchronizing processing between programs, and (c) notifying a partner of errors in the communication.

configuration. (1) The manner in which the hardware and software of an information processing system are organized (2) In Personal Communications, the

arrangement of personal computers connected to one or more host systems by one or more attachment types.

configuration file. A file that specifies the characteristics of a system device or network.

continuous carrier. A characteristic of the modem used in an SDLC type of attachment; frequency capable of being modulated or impressed with a second (informative carrier) signal.

control unit. A device that manages the flow of data between personal computers used with Personal Communications and a host system.

conversation. A logical connection between two transaction programs using an LU 6.2 session. Conversations are delimited by brackets to gain exclusive use of a session.

CP. Control point.

CPI-C. Common Programming Interface for Communications.

CP name. A network-qualified name of a control point (CP), consisting of a network ID qualifier identifying the network (or name space) to which the CP's node belongs, and a unique name within the scope of that network ID identifying the CP. Each APPN or LEN node has one CP name, assigned to it at system-definition time.

CSD. Corrective service diskette.

customization. The process of configuring one or more personal computers connected to a host system by the SDLC, LAN, ASYNCH, X.25, DFT, or other attachment. Customization precedes Personal Communications installation and includes the definition of the sessions to be carried out by Personal Communications.

D

DBCS. Double-byte character set

default. A value, attribute, or option that is assumed when no differing selection is made.

definition file. A file loaded by default that contains keyboard and screen color characteristics that can be adapted for a particular session. The lines of a definition file must be coded according to a strict layout.

delimiter. In the IBM Token-Ring Network, a bit pattern that defines the limits of a frame or token.

dependent LU. See *SSCP-dependent LU*.

dependent LU requester (DLUR). An APPN end node or an APPN network node that owns dependent LUs, but requests that a dependent LU server provide the SSCP services for those dependent LUs.

dependent LU server (DLUS). An APPN network node that provides SSCP services for a dependent LU in its own or another APPN network. Contrast with *dependent LU requester*.

diacritic key. A key that allows a character to be entered without the cursor position being changed. When you enter an accent mark or a tilde, for example, the cursor does not advance to the next position until you enter the corresponding alphabetic character.

dial-up line. See *switched line*.

direct memory access (DMA). The system facility that allows a device on the Micro Channel bus to get direct access to the system or bus memory without the intervention of the system processor.

directory. A named hierarchical grouping of files in a file system.

DLUR. Dependent LU requester.

DLUS. Dependent LU server.

DMA. See *direct memory access.*

domain. A set of servers that allocate shared network resources within a single logical system.

DOS session. A session in which a personal computer operates as a stand-alone computer, running under Disk Operating System (DOS). See *host session*.

download. The transfer of a file from a host system to a personal computer.

double-byte character set (DBCS). A set of characters in which each character is represented by 2 bytes. Languages such as Japanese, Chinese, and Korean, which contain more symbols than can be reqpresented by 256 code points, require double-byte character sets. Because each character requires 2 bytes, the typing, display, and printing of DBCS characters requires hardware and programs that support DBCS. Contrast with *single-byte character (SBCS).*

dynamic data exchange (DDE). A protocol that allows applications to exchange data.

E

EBCDIC. See *extended binary-coded decimal interchange code*.

ECF. See *Enhanced Connectivity Facilities.*

EGA. See *enhanced graphics adapter*.

EHLLAPI. See *Emulator High-Level Language Application Programming Interface*.

EIA 232. A standard established by the Electronic Industries Association for direct linkage for serial binary data transfer between a computer and peripheral equipment, such as a modem or printer. See *V.24*.

EMM. See *Expanded Memory Manager*.

EMS. See *Expanded Memory Specification*.

emulator. A program that allows a device to operate as if it were a different type of device. Personal Communications, for example, allows supported personal computers and printers to operate as if they were 3270-series or 5250-series workstations.

Emulator High-Level Language Application Programming Interface (EHLLAPI). A programming interface that permits workstation-based programs to access host computer presentation space.

Enhanced Connectivity Facilities (ECF). An IBM communication utility that provides the ability to share resources between workstations and host systems.

enhanced graphics adapter (EGA). An adapter, such as the IBM Enhanced Graphics Adapter, that provides high-resolution graphics, allowing the use of a color display for text processing as well as graphics applications.

EOF. End of Field

EOT. End of Text

error log. A file that stores error information for later access.

Ethernet. A 10-Mbps baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and delayed retransmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

exit function. A function that allows the user to switch from a host session to a DOS session. The key sequence for exit is **Ctrl+End**.

Expanded Memory Manager (EMM). A standard device driver that manages expanded memory in 16-KB logical pages. The EMM maps a set of logical pages into a page frame using the physical address space that is available in memory above 640 KB, storing the map into conventional memory. The application then accesses the expanded memory thus created as though it were conventional memory.

Expanded Memory Specification (EMS). A specification for accessing memory above the conventional DOS memory address space, 0–640 KB. See *Expanded Memory Manager*, *LIM EMS 3.2*, and *LIM EMS 4.0*.

extended binary-coded decimal interchange code (EBCDIC). The standard code, using a character set consisting of 8-bit coded characters, used by Personal Communications for information interchange between personal computers and a host system. See also *American National Standard Code for Information Interchange*.

extended-color mode. The mode in which a terminal operates if the host system supplies color information.

extended error code. An 8-byte data string returned by **Query System** generated by an internal system error that is used by service personnel for diagnosis.

extended memory. Memory above 1 MB that can be accessed and used by 80286 and 80386** (or greater) CPUs to run applications in protected mode. PC DOS does not normally access memory above 640 KB. This memory is useful only for applications that are written specifically to use extended memory. Successful use of extended memory by an application usually requires the use of an 80386 or greater CPU, which provides machine instructions to handle the use of extended memory.

external resource. A file, directory, or device supplied by a server outside the current domain.

F

field. (1) An area in a record or panel used to contain data. (2) In the IBM 3270 data stream, a group of consecutive positions on a presentation space having similar characteristics that are defined by a field attribute byte at the beginning of the field.

field delimiters. Symbols (often brackets) that indicate the limits of a data entry field.

file mode. In VM/CMS, the third field of the file identifier, which indicates the assigned virtual disk wherein the file resides.

file type. In VM/CMS, the second field of the file identifier. The file type can be arbitrary, or of significance to the system (for example, SCRIPT, LIST3270, and NOTEBOOK file types).

fixed disk. See *hard disk*.

flag. A character or bit sequence that marks an occurrence or boundary, such as the end of a word or the beginning or end of a data transmission block.

frame. A data structure (data frame) composed of fields meeting the field specifications of a type of communication protocol. Frames are used to control data transfer across a data link.

In SDLC, a frame is a sequence of bits delimited by an opening and closing flag. In X.25 packet switching data networks, frames are composed of 8-bit byte sequences delimited by beginning and ending flags; the frames in X.25 control various functions, data transfer, and transmission checking.

G

gateway. A station in a local area network through which a connection to a host system or a separate network is established.

gateway status utility. A utility that allows a user to check the status of sessions in a gateway network.

H

hard disk. A rigid-magnetic disk such as the internal disks used in the system units of personal computers and in external hard disk drives (also known as fixed disks).

High-Level Language Application Programming Interface (HLLAPI). A software product that supports interaction between a host program and an application program running on a personal computer in terminal emulation (usually 3270 terminal emulation).

High-Performance Routing (HPR). An addition to APPN that enhances data-routing performance and session reliability.

HLLAPI. See *High-Level Language Application Programming Interface*.

host address. A LAN address used by the host controller.

host code page. The character code page used by the host system to which a gateway connects.

host print. A printer session on a local personal computer directed by the host system.

host profile. A configuration profile that identifies a gateway by name, and establishes LU addresses and types for connected logical units.

host session. A logical connection that enables a personal computer to communicate with a host system. A session can be identified by LU address, LT number, or session ID. See *DOS session*. See also *logical terminal*.

host short name. See *short name*.

host system. In Personal Communications, the computer linked to one or more personal computers by the SDLC, LAN, ASYNCH, X.25, or DFT attachment.

hot key. A keystroke or keystroke combination that initiates the hotkey function. See *hotkey function*.

hotkey function. A function that allows you to switch to the next logical terminal or DOS, on whichever screen it last appeared, without exiting the current application.

HPR. High-Performance Routing.

I

IBM PC Network. A local area network providing peer-to-peer communication among IBM personal computers, allowing them to share resources. It utilizes carrier sense multiple access/collision detect (CSMA/CD) protocol and supports a data transmission rate of 2 million bits per second. Broadband and baseband versions are available.

IBM Token-Ring Network. A general-purpose baseband token-ring local area network for interconnecting information processing equipment at a local site. It uses a token-ring access protocol and supports a data rate of either 4 or 16 million bits per second. It conforms to the IEEE 802.5 (token-ring) and IEEE 802.2 (logical link control) standards.

IBM NCP Packet Switching Interface Program (NPSI). A network host communication product that runs in an IBM 3745 communications controller and enables it to connect to an ITU-TS X.25 recommendation packet-switching data network.

idles. In a token-ring network, the signals sent when neither frames nor tokens are being transmitted.

IEEE. See *Institute of Electrical and Electronics Engineers*.

IEEE 802.2. An IEEE standard for logical link control in local area networks.

initial program load (IPL). The initialization procedure that starts an operating system, or the process of loading programs and preparing a system to run jobs.

installation. In Personal Communications, the process of loading microcode from the Personal Communicationsdiskettes.

Institute of Electrical and Electronics Engineers (IEEE). An organization that establishes and promotes standards in related industries.

integrated services digital network. A digital end-to-end telecommunication network that supports multiple services including, but not limited to, voice and data.

interleave. In Personal Communications, to arrange parts of a transmission sequence so that they are interspersed with parts of one or more other transmission sequences of the same nature, with each

sequence retaining distinct meaning and identity. See also *packet-switching data network* and *X.25 Recommendation*.

International Telecommunication Union (ITU). The specialized telecommunication agency of the United Nations, established to provide standardized communication procedures and practices, including frequency allocation and radio regulations worldwide.

interrupt level. A number assigned to an interrupt to indicate priority. The interrupt with the lowest number has the highest priority.

IPL. See *initial program load*.

ISDN. Integrated services digital network.

ITU-TS. See *ITU-T*.

ITU-T. International Telecommunication Union - Telecommunication Standardization Sector. The part of the International Telecommunication Union (ITU) that is reponsible for developing recommendations for telecommunications. (Formerly known as the CCITT.)

J

jump. To start the jump function.

jump function. A switch to the logical terminal session specified.

K

key assignment. The character, numeral, symbol, or function generated when a keyboard key is pressed. All keys have default assignments that can be customized.

keyboard definition. Definitions of keyboard keys as stored on one of the Personal Communications diskettes.

keyboard template. An overlay placed on a keyboard to explain the function of each key.

L

LAN. See *local area network*.

leased line. See *nonswitched line*.

LIM EMS 3.2. An expanded memory specification agreed to in 1984 by the computer software manufacturers Lotus**, Intel**, and Microsoft** to access memory above the DOS conventional memory area, 0–640 KB, using an EMM device driver. LIM EMS 3.2 allows an application to store up to 8 MB of data in expanded memory, in 512 logical pages of 16 KB. The EMS 3.2 expanded memory manager maps up to 64 KB of expanded memory in the PC BIOS memory area. See also *LIM EMS 4.0* and *Expanded Memory Manager*.

LIM EMS 4.0. An extended expanded memory specification (EEMS) pioneered by AST** Research Corporation as an extension to LIM EMS 3.2 to permit expanded memory to be used for directly executing programs through multitasking, and to expand the areas accessible for mapping expanded memory beyond the conventional memory barrier, up to 1024 KB. The EEMS was further improved and defined by agreement of Lotus, Intel, and Microsoft corporations to better support windowing and multitasking operating environments. With appropriate hardware support, LIM EMS 4.0 supports up to 32 MB of expanded memory. Its EMM can create page frames up to 896 KB, and up to 2048 logical pages of 16-KB memory.

line speed. The speed at which data is transferred over an analog (telephone) line. Line speeds depend on the quality of the line.

local area network (LAN). (1) A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation. (T) (2) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network. (3) See also *Ethernet* and *token ring*.

local copy. A printer session initiated by a personal computer user or host that reproduces a presentation space with all supported graphics.

local LU. A logical unit not distributed over the LAN, but controlled by a gateway personal computer. This is normally a physical device such as a workstation, printer, or terminal.

logical terminal (LT). (1) A terminal with a name related to one or more physical terminals. (2) The definition of a specific 3270 terminal emulation session. See also *host session*.

logical terminal ID. See *logical terminal identifier*.

logical terminal identifier (LT ID). An identifier for the LU type specified in the gateway host profile. In Personal Communications, up to 8 (unchangeable) IDs can be displayed. See *session ID*.

logical terminal number. A number used to identify DFT host sessions. A 3270 control unit controls some number of sessions with the host. The control unit then assigns from 1 to 5 of these sessions to a particular coaxial connection and assigns each session an LT number from 1 to 5.

logical unit (LU). A type of network accessible unit that enables users to gain access to network resources and communicate with each other.

logon. The process of accessing a domain and assuming a user ID.

LT. See *logical terminal*.

LU. Logical unit.

LU 6.2. A type of LU that supports sessions between two applications in a distributed data processing environment using the SNA general data stream, which is a structured-field data stream, or a user-defined data stream.

LU address. In Personal Communications, a number from 02 to 254 assigned by the host to represent the remote partner in an SNA session.

M

MAN. Metropolitan area network.

mantissa. The decimal part of a common logarithm.

mark. A symbol or sequence of symbols that indicate the beginning or end of a field, word, data item, or data set.

menu. A list of available operations. You select which operation you want from the list.

messaging name. A name under which messages can be received.

metropolitan area network (MAN). A network formed by the interconnection of two or more networks which may operate at higher speed than those networks, may cross administrative boundaries, and may use multiple access methods. (T) Contrast with *local area network (LAN)* and *wide area network (WAN)*.

modem. A serial data translator that converts digital signals into analog signals at transmission, and on reception, converts analog signals back into digital signals. Modems are used primarily for telephonic computer communication. Synonymous with modulator/demodulator.

multiplexer (mux). A device capable of interleaving events of two or more activities, or of spreading the events of an interleaved sequence of the multiple activities.

mux. See *multiplexer*.

N

NA. Not applicable. When this appears in a calling parameter position, it means that PC400 EHLLAPI does not require this parameter to perform the specific function.

NCB. See *network control block*.

NetBIOS. Network Basic Input/Output System. In Personal Communications, an interface for application programs (API) between a LAN adapter and a token-ring network.

netname. The name used to identify a shared resource on the network.

network. (1) An arrangement of nodes and connecting branches. (T) (2) A configuration of data processing devices and software connected for information interchange. (3) A group of nodes and the links interconnecting them.

network accessible unit (NAU). A logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is the origin or the destination of information transmitted by the path control network. Synonymous with *network addressable unit*.

network address. (1) According to ISO 7498-3, a name, unambiguous within the OSI environment, that identifies a set of network service access points. (2) In a subarea network, an address, consisting of subarea and element fields, that identifies a link, link station, physical unit, logical unit, or system services control point. Subarea nodes use network addresses; peripheral nodes use local addresses or local-form session identifiers (LFSIDs). The boundary function in the subarea node to which a peripheral node is attached transforms local addresses or LFSIDs to network addresses and vice versa. Contrast with *network name*.

network addressable unit (NAU). Synonym for *network accessible unit*.

network administrator. A person who manages the use and maintenance of a network.

network application program. In the IBM Token-Ring Network, a program used to connect and communicate with adapters on a network, enabling users to perform application-oriented activities and to run other application programs.

network control block (NCB). A part of the network control program that controls the resources used by the communication network in a LAN attachment.

network name. The symbolic identifier by which users refer to a network accessible unit, a link, or a link station within a given subnetwork. In APPN networks, network names are also used for routing purposes. Contrast with *network address*.

network node (NN). See *Advanced Peer-to-Peer Networking (APPN) network node*.

network node server. An APPN network node that provides network services for its local LUs and client end nodes.

non-return-to-zero inverted encoding (NRZI). A data encoding method that ensures correct synchronization of information in an analog data stream transmission.

nonswitched line. A communication line (such as is used in the SDLC attachment) that is permanently connected and does not require dialing to establish communication. Contrast with *switched line*.

NPSI. See *IBM NCP Packet Switching Interface Program*.

NRZI. See *non-return-to-zero inverted encoding*.

NUL. A transmission control character used as a filler or for sequence control in network communication.

Num Lock. The keyboard mode entered after the **Num Lock** key is pressed. When the **Shift** key is pressed while in this mode, Num Lock is temporarily discontinued.

O

ODBC. Open Data Base Connectivity. An interface that allows applications to access data from many sources using Structured Query Language (SQL) as a standard for accessing data.

OIA. See *operator information area*.

OIA indicator. Characters or symbols in the operator information area providing information about network, workstation, gateway, or interface status. See also *operator information area*.

operator information area (OIA). The bottom line of a session window that shows information on status and Personal Communications connections.

OLE. Object Linking and Embedding; OLE is a COM-based technology that allows for the creation of compound documents.

P

packet. A discrete unit of information including addressing and sequence information that can be sent interleaved with other packets of the same and other transmission sequences. Packets sent over X.25 virtual circuits always follow the same route and arrive in the order they were sent.

packet assembler/disassembler (PAD). A functional unit that enables data terminal equipment (DTEs) not equipped with an X.25 interface to use a packet-switching data network. On the most common PAD, data from ASCII start/stop DTEs is tranformed to and from X.25 packets. See also *packet, packet-switching data network*.

packet-switching data network (PSDN). A data transmission system consisting of one or more host systems and one or more network stations that communicates using packets of information. The data is sent in interleaved packets to distribute the flow of information and maximize the data transmission capabilities of the network. X.25 is a packet-switching data network communication protocol. See *X.25 network*.

PAD. See *packet assembler/disassembler*.

parity bit. A binary digit appended to a group of binary digits as needed to make the sum of all the digits either odd (odd parity) or even (even parity).

PC Network. See *IBM PC network*.

PCSDIAL. A program that automatically dials a host number, permitting a fast connection.

permanent virtual circuit (PVC). An X.25 virtual circuit that is permanently established (by subscription) and, therefore, needs no call-establishment protocol. Contrast with *switched virtual circuit*.

physical unit (PU). The component that manages and monitors the resources (such as attached links and adjacent link stations) associated with a node, as requested by an SSCP via an SSCP-PU session. An SSCP activates a session with the physical unit in order to indirectly manage, through the PU, resources of the node such as attached links. This term applies to type 2.0, type 4, and type 5 nodes only.

physical unit identifier (PUID). In the gateway status utility, five hexadecimal digits used to identify the network station to the gateway.

PIU. Path information unit.

pooled LUs. Logical units that share host sessions in a common pool to save system resources.

POR. See *power-on reset function*.

port number. In DFT attachment, the logical ports assigned to a terminal by a control unit.

power-on reset (POR) function. A function that allows the user to reset all logical terminals. The key sequence for POR is **Ctrl+Pause** using the Enhanced or Space Saving keyboards. Otherwise, the sequence is **Ctrl+F2**.

preallocated LUs. Logical units only available to a specified network station.

presentation space. The host display space for the window.

presentation space position parameter. One of the four parameters that you must specify for some EHLLAPI functions. A position in the presentation space from 1 to 3564.

printer utility. A utility that allows users to check the status of the printers in their network.

private application. An application maintained for use only on one computer or user, that is not available on the network for other users.

program operator. The EHLLAPI program that performs and monitors activities in a workstation presentation space.

programmed symbol set (PS). A set of fonts that can be system-defined or defined by the user and to which a code can be assigned.

protected field. A field that holds data for viewing; the data cannot be changed (at the current screen) by the user.

protocol. A specification for the format and timing of data exchanged between communicating systems. Also, the set of rules governing the operation of units of the communication system to facilitate communication.

PSDN. See *packet-switching data network.*

PSID. Presentation Space Identifier; a short one-character or one-letter name of the presentation space.

PU. Physical unit. See also *physical unit identifier*.

public application. An application shared among users on a network.

PUID. See *physical unit identifier*.

PVC. See *permanent virtual circuit*.

Q

queue. A list of pending tasks for a computer or server to execute in order.

R

redefinition file. See *definition file.*

request. A message unit that signals initiation of a particular action or protocol. For example, Initiate-Self is a request for activation of an LU-LU session.

requester. A computer that accesses shared network resources through a server.

resource. A directory (files resource) or device used by computers on a network.

response time monitor. A function that measures and displays the host response time

RS-232C. See *EIA 232* and *V.24.*

RSAP. Remote service access point. See *service access point*.

run book. A supplemental guide, controlled by a system administrator, that describes the implementation and usage of a specific system configuration.

S

SAP. See *service access point*.

SAS. See *session active screen*.

SBCS. See *single-byte character set*.

SDLC. See *Synchronous Data Link Control.*

SDLC command. The control information in the C-field of the link header, sent from a primary station to a secondary station.

Server-Requester Programming Interface (SRPI). An IBM application programming interface (API) used by requester and server programs to communicate with the personal computer or host routers.

service access point (SAP). The gateway address of the controller. An SAP provides a point to link the controller to the host system.

service coordinator. The person in your organization responsible for answering hardware and software computing questions.

service transaction program. Any IBM-supplied transaction program running in a network accessible unit. Contrast with *application transaction program.*

session. (1) A logical connection between a server and a requester that was initiated by a successful request for a shared resource. See also *host session* and *DOS session*. (2) A logical connection between two network addressable units that can be activated, tailored to provide various protocols, and deactivated, as requested.

session active screen (SAS). The display that is active for the current communication session. See also *next-session-active-screen function.*

session ID. An alphabetic ID (*a* through *h*) assigned by Personal Communications to each session or screen. This applies to all types of host sessions and is used in file transfers. See also *short name.*

shared resource. A directory or device made available to users on a network. A shared resource is attached to a server, but it is not attached to the requesters that use them.

Shift Lock. The personal computer keyboard mode entered after the **Shift Lock** key is pressed. In this mode, the keyboard performs the function printed on

the upper half of the keytop. Press the **Shift Lock** key to cancel the Shift Lock mode.

short name. In Personal Communications, a character displayed in column 7 of the operator information area that shows the session ID. See also *session ID* and *operator information area*.

single-byte character set. A character set in which each character is represented by a one-byte code.

SMS Install. The Microsoft Systems Management Server (SMS) product provides for the remote installation and uninstallation of Personal Communications.

SNA. See *Systems Network Architecture*.

SNA network. The part of a user-application network that conforms to the formats and protocols of Systems Network Architecture. It enables reliable transfer of data among users and provides protocols for controlling the resources of various network configurations. The SNA network consists of network accessible units (NAUs), boundary function, gateway function, and intermediate session routing function components; and the transport network.

Sockets over SNA. The AnyNet Sockets over SNA access node function of Personal Communications enables TCP/IP application programs using the WinSock interface to communicate over an SNA network.

source diskette. In a diskette copying procedure, the diskette from which information is read.

SRPI. Server-Requester Programming Interface.

SSCP. System services control point.

SSCP-dependent LU. An LU that requires assistance from a system services control point (SSCP) in order to initiate an LU-LU session. It requires an SSCP-LU session.

suspend-DOS function. The function that stops the running of the DOS session. The key sequence for suspend-DOS is **Alt+S**. (In the Netherlands, the sequence is **Ctrl+S**.)

SVC. See *switched virtual circuit*.

switched line. A telecommunication line in which the connection is established by dialing. See also *switched virtual circuit*. Contrast with *nonswitched line*.

switched virtual circuit (SVC). An X.25 virtual circuit that is dynamically established when needed. The X.25 equivalent of a switched line. Also known as a virtual call. Contrast with *permanent virtual circuit (PVC)*.

switch-screen function. A function that switches keyboard control to the other screen, if two screens are used. The key sequence for switch-screen is **Alt+Tab**.

Synchronous Data Link Control (SDLC). A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a line connection. Transmission exchanges can be duplex (in both directions at the same time) or half-duplex (in either direction, one direction at a time) over switched lines or direct lines.

system services control point (SSCP). A component within a subarea network for managing the configuration, coordinating network operator and problem determination requests, and providing directory services and other session services for users of the network. Multiple SSCPs, cooperating as peers with one another, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its own domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks. The layered structure of SNA allows the ultimate origins and destinations of information, that is, the users, to be independent of and unaffected by the specific SNA network services and facilities used for information exchange.

T

target diskette. In a diskette or storage copying procedure, the diskette onto which information is written.

template. See *keyboard template*.

terminal emulator. See *emulator*.

terminal operator. The human user of a EHLLAPI application program. Contrast with *program operator*.

token ring. (1) According to IEEE 802.5, network technology that controls media access by passing a token (special packet or frame) between media-attached stations. (2) A FDDI or IEEE 802.5 network with a ring topology that passes tokens from one attaching ring station (node) to another. (3) See also *local area network (LAN)*.

token-ring network. (1) A ring network that allows unidirectional data transmission between data stations, by a token passing procedure, such that the transmitted data return to the transmitting station. (T) (2) A network that uses a ring topology, in which tokens are passed in a circuit from node to node. A node that is ready to send can capture the token and insert data for transmission.

TOKREUI. A low-level personal computer extended user interface program to access the token-ring adapter. TOKREUI is an acronym that stands for *token-ring extended user interface.*

TQDOS. A multitasking DOS interface that is called automatically by Personal Communications to handle service tasks.

trace size. See *trace table.*

trace table. A storage area that contains the records of all Personal Communications operations. The trace information can either be directed to the standard output destination, or it can be redirected to a file.

transaction program (TP). A program that processes transactions in an SNA network. There are two kinds of transaction programs: application transaction programs and service transaction programs. See also *conversation.*

transferring files. Sending files from a personal computer to a host system or vice versa. See *download* and *upload.*

U

UID. See *user identifier.*

unprotected field. A field that is available for the user to enter or modify data.

upload. The transfer of a file from a personal computer to a host system.

user ID. See *user identifier.*

user identifier (UID). A name that uniquely identifies a user on a network or system.

user profile. A file containing values for setting up the user environment, and that can be automatically accessed when a user logs on.

utility programs. Programs that support operations of Personal Communications, such as file transfer and keyboard redefinitions.

V

V.24. An ITU-T recommendation that defines interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE). When combined with the specification of electrical characteristics given in ITU-T recommendation V.28, it specifies a modem interface for data rates below 20 kilobits per second (which is equivalent to EIA 232).

V.35. An ITU-T recommendation that defines interchange circuits and electrical characteristics between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) at various data rates. Originally defined for wideband modems, but now commonly used for various data network services.

valid key. A key that is recognized by the host session.

VB Scripting. An interface used with Personal Communications to define and execute Visual Basic scripts.

VGA. See *video graphics adapter.*

video graphics adapter (VGA). A computer adapter that provides high-resolution graphics and a total of 256 colors. See also *enhanced graphics adapter (EGA).*

virtual route (VR). In SNA, either (a) a logical connection between two subarea nodes that is physically realized as a particular explicit route or (b) a logical connection that is contained wholly within a subarea node for intranode sessions. See also *Systems Network Architecture.*

VR. See *virtual route.*

VT terminal. A series of general purpose ASCII terminals that were introduced by Digital Equipment Company.

W

WAN. See *wide area network.*

wide area network (WAN). (1) A network that provides communication services to a geographic area larger than that served by a local area network or a metropolitan area network, and that may use or provide public communication facilities. (T) (2) A data communication network designed to serve an area of hundreds or thousands of miles; for example, public and private packet-switching networks, and national telephone networks. (3) Contrast with *local area network (LAN)* and *metropolitan area network (MAN).*

X

X.21. An ITU-T recommendation for a general-purpose interface between data terminal equipment and data circuit-terminating equipment for synchronous operation on a public data network.

X.25 network. A packet-switching data network linked in accordance with ITU-T recommendation X.25.

X.25 Recommendation. An ITU-T recommendation for the interface between data terminal equipment and packet-switched data networks.

X.400. The message-handling services standard of ITU-T and ISO.

XID. SNA exchange identification command. See also *Systems Network Architecture.*

Index

V

W

Readers' Comments — We'd Like to Hear from You

Personal Communications
Version 5.0 for Windows® 95, Windows 98,
Windows NT®, and Windows 2000
Access Feature

Publication No. SC31-8684-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC31-8684-01

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
Department CGMD / Bldg 500
P.O. Box 12195
Research Triangle Park, NC
27709-9990

Fold and Tape **Please do not staple** Fold and Tape

SC31-8684-01

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC31-8684-01