AnyNet: SNA over TCP/IP Installation and Interoperability

Document Number GG24-4395-00

March 1995

International Technical Support Organization
Raleigh Center

Take Note!

Before using this information and the product it supports, be sure to read the general information under "Special Notices" on page xv.

First Edition (March 1995)

This edition applies to:

- Version 4 Release 2 of ACF/VTAM, Program Number 5695-117 for use with the MVS/ESA Operating System
- Version 2 Release 2.1 of IBM TCP/IP for MVS/ESA, Program Number 5735-HAL for use with the MVS/ESA Operating System
- Version 3 Release 1 of IBM TCP/IP for MVS/ESA, Program Number 5655-HAL for use with the MVS/ESA Operating System
- Version 2.0 of IBM AnyNet/2, SNA over TCP/IP and Sockets over SNA, Program Number 5622-321 for use with the OS/2 Operating System
- Version 2 of IBM TCP/IP for OS/2, Program Number 5622-086 for use with the OS/2 Operating System
- Version 1.1 of Communications Manager/2, Program Number 5622-078 for use with the OS/2 Operating System
- Version 3 Release 1 of Operating System/400, Program Number 5763-SS1

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

An ITSO Technical Bulletin Evaluation Form for reader′s feedback appears facing Chapter 1. If the form has been removed, comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. 985 Building 657
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Abstract

This document describes the new implementations of the Multiprotocol Transport Networking (MPTN) architecture on the MVS, OS/2 and AS/400 platforms. It includes SNA over TCP/IP (MVS/ESA and OS/2) and AnyNet/400 APPC over TCP/IP. They are shipped with the AnyNet/MVS feature of VTAM Version 4 Release 2 for MVS/ESA and as AnyNet/2 OS/2 program products except AnyNet/400 APPC over TCP/IP. AnyNet/400 APPC over TCP/IP is shipped with OS/400 Version 3 Release 1.

This document provides working configurations, definitions and test results. It guides the readers in the planning, methodology, installation and implementation of their first MPTN network.

It is intended for systems engineers, network managers, application designers and technical support personnel who are interested in the implementations of the IBM MPTN architecture.

A basic working knowledge of SNA, TCP/IP and MPTN is assumed. However, this document also provides a brief introduction to these topics.

(236 pages)

Contents

Figures

Tables

Special Notices

This publication is intended to help systems engineers, network managers, application designers and technical support personnel, who are interested in implementing some Multiprotocol Transport Networks, by providing them with working configurations, definitions and test results. The information in this publication is not intended as the specification of any programming interfaces that are provided by VTAM Version 4 Release 2 AnyNet/MVS feature, OS/2 AnyNet/2 program product: SNA over TCP/IP, or Operating System/400. See the PUBLICATIONS section of the IBM Programming Announcement for the above products for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (VENDOR) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

ACF/VTAM	Advanced Peer-to-Peer Networking
AIX/ESA	AIX/6000
AIXwindows	AnyNet
Application System/400	APPN
AS/400	CICS
CICS OS/2	CICS/ESA
CICS/MVS	DB2
DB2/2	IBM
Operating System/2	Operating System/400
OS/2	OS/400
PS/2	RISC System/6000
S/370	S/390
SAA	VM/ESA
VM/XA	VSE/ESA
VTAM	WIN-OS/2

The following terms are trademarks of other companies:

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

DEC VT52, DEC VT100 and DEC VT220	Digital Equipment Corporation
EtherLink	3COM Corporation
SunOS, SPARCstation, Network File System, NFS	Sun Microsystems, Inc.
1-2-3, Lotus, Freelance, Freelance Graphics	Lotus Development Corporation

Other trademarks are trademarks of their respective companies.

Preface

This document describes the new implementations of the Multiprotocol Transport Networking (MPTN) architecture on the MVS, OS/2 and AS/400 platforms. It includes SNA over TCP/IP (MVS/ESA and OS/2) and AnyNet/400 APPC over TCP/IP. They are shipped with the AnyNet/MVS feature of VTAM Version 4 Release 2 for MVS/ESA and as AnyNet/2 OS/2 program products except AnyNet/400 APPC over TCP/IP. AnyNet/400 APPC over TCP/IP is shipped with OS/400 Version 3 Release 1.

This document provides working configurations, definitions and test results. It guides the readers in the planning, methodology, installation and implementation of their first MPTN network.

It is intended for systems engineers, network managers, application designers and technical support personnel who are interested in the implementations of the IBM MPTN architecture.

A basic working knowledge of SNA, TCP/IP and MPTN is assumed. However, this document also provides a brief introduction to these topics.

It is not the purpose of this document to explain in detail how to use products which are prerequisites for AnyNet/MVS, AnyNet/2 and AnyNet/400 A list of publications is included in this document where you can refer to for more information if required.

Please also understand that guidance on how to actually develop applications to be used with AnyNet/MVS or AnyNet/2 SNA over TCP/IP, or AnyNet/400 APPC over TCP/IP including samples, would be far beyond the scope of this document.

How This Document Is Organized

The document is organized as follows:

- Part 1, “Introduction”

  This part of the document provides an introduction to the network architectures that will be used by the products and scenarios within this publication.

- Part 2, “Platform Implementations”

  This part of the document provides an overview of the SNA over TCP/IP functions and explains how to install and configure them under the MVS/ESA, OS/2 and AS/400 operating Systems.

- Part 3, “Working Scenarios”

  This part of the document provides the installation scenarios that were conducted to test and document the SNA over TCP/IP functions between OS/2 and MVS systems, and APPC over TCP/IP function between OS/400 and MVS systems.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

- VTAM V4R2 publications
  - *VTAM Release Guide Version 4 Release 2 for MVS/ESA*, GC31-6492
  - *VTAM Messages and Codes Version 4 Release 2 for MVS/ESA*, SC31-6493
  - *VTAM Network Implementation Guide Version 4 Release 2 for MVS/ESA*, SC31-6494
  - *VTAM Operation Version 4 Release 2 for MVS/ESA*, SC31-6495
  - *VTAM Resource Definition Reference Version 4 Release 2 for MVS/ESA*, SC31-6498
  - *VTAM Diagnosis Version 4 Release 2 for MVS/ESA*, LY43-0065 (available to IBM-licensed customers only)
- AnyNet publications
  - *Program Directory ACF/VTAM AnyNet Feature for V4R2*, Program Number 5695-117
  - *VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA*, SC31-6526
  - *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP*, SC31-6527
  - *VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA Gateway for OS/2*, SC31-6528
  - *IBM AnyNet/2 Version 2.0: Guide to SNA over TCP/IP*, GV40-0375
  - *IBM AnyNet/2 Version 2.0: Guide to Sockets over SNA*, GV40-0376
  - *IBM AnyNet/2: NetBEUI over SNA Version 1.0 User's Guide*, GV40-0377
  - *IBM AnyNet/2: NetBEUI over SNA Version 1.0 Administrator's Guide*, GV40-0402
- TCP/IP publications
  - *IBM TCP/IP Version 2 Release 2.1 for MVS: Planning and Customization*, SC31-6085
  - *IBM TCP/IP Version 2 Release 2.1 for MVS: Programmer's Reference*, SC31-6087
  - *IBM TCP/IP Version 2 Release 2.1 for MVS: User's Guide*, SC31-6088
  - *IBM TCP/IP Version 2 for OS/2: Installation and Administration*, SC31-6075
  - *IBM TCP/IP Version 2 for OS/2: User's Guide*, SC31-6076
  - *IBM TCP/IP Version 2 for OS/2: Extended Networking Guide*, SC31-7071
  - *IBM TCP/IP Version 2 for OS/2: Network File System Guide*, SC31-7069
  - *IBM TCP/IP Version 2 for OS/2: X Window System Server Guide*, SC31-7070
  - *IBM TCP/IP Version 2 for OS/2: Domain Name Server Guide*, SC31-7174

- IBM Communications Manager/2 Version 1.1 publications
  - *IBM Communications Manager/2 Version 1.1 Workstation Installation and Configuration Guide*, SC31-7169
  - *IBM Communications Manager/2 Version 1.1 Network Administration and Subsystem Management Guide*, SC31-6168
- MPTN publications
  - *Networking Blueprint Executive Overview*, GC31-7057
  - *Multiprotocol Transport Networking: Technical Overview*, GC31-7073
- AS/400 publications
  - *AS/400 Communications Configuration*, SC41-3401
  - *AS/400 APPN Support*, SC41-3407
  - *AS/400 TCP/IP Configuration and Reference*, SC41-3420
  - *AS/400 Sockets Programming*, SC41-3442
  - *AS/400 APPC Programming*, SC41-3443
- Other useful IBM publications
  - *System Network Architecture Formats*, GA27-3136
  - *Multiplatform APPC Configuration Guide*, GG24-4485

International Technical Support Organization Publications

AnyNet SNA over TCP/IP: Installation and Interoperability, GG24-4395

AnyNet Sockets over SNA and NetBIOS over SNA: Installation and Interoperability, GG24-4396

AS/400 AnyNet Scenarios, GG24-2531 (available in April, 1995)

APPN Architecture and Product Implementations Tutorial, GG24-3669

TCP/IP Tutorial and Technical Overview, GG24-3376

TCP/IP V2.0 for OS/2: Installation and Interoperability, GG24-3531

A complete list of International Technical Support Organization publications, with a brief description of each, may be found in:

International Technical Support Organization Bibliography of Redbooks, GG24-3070.

To get a catalog of ITSO technical publications (known as “redbooks”), VNET users may type:

```
TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
```

How to Order ITSO Redbooks

IBM employees in the USA may order ITSO books and CD-ROMs using PUBORDER. Customers in the USA may order by calling 1-800-879-2755 or by faxing 1-800-284-4721. Visa and Master Cards are accepted. Outside the USA, customers should contact their local IBM office.

Customers may order hardcopy ITSO books individually or in customized sets, called GBOFs, which relate to specific functions of interest. IBM employees and customers may also order ITSO books in online format on CD-ROM collections, which contain redbooks on a variety of products.

Acknowledgments

The advisor for this project was:

Michael Li
International Technical Support Organization, Raleigh Center

The authors of this document are:

Martin W. Murhammer
IBM Austria, Open Systems Center, Vienna

John Purrington
IBM Germany, Munich

Thilo Teichtmann
IBM Germany, Education Center, Herrenberg

Susanne Hüsing
IBM Germany, Education Center, Herrenberg

This publication is the result of a residency conducted at the International Technical Support Organization, Raleigh Center.

Thanks to the following people for the invaluable advice and guidance provided in the production of this document:

Carla Sadtler, Georg Steinborn
and all others at the
International Technical Support Organization, Raleigh Center

Bud Gross
International Technical Support Organization, Raleigh Center

Mick Lugton
International Technical Support Organization, Raleigh Center

Alfred Christensen
International Technical Support Organization, Raleigh Center

Wendy White
IBM Research Triangle Park

Ronnie Sarkar
IBM Research Triangle Park

Alisa Morse
IBM Research Triangle Park

Mark Womack
IBM Research Triangle Park

Bert Voorham
IBM Netherlands

Part 1. Introduction

Chapter 1. A Short Introduction to SNA Networking

In the 1950s, electronic data processing became a reality, and in the first steps of this new science, many different kinds of machines were proposed, built and installed. They had different physical dimensions, processing power, technologies for storing and processing data, and ways of interacting with the users.

In the 1960s, the first computer architectures based on the experiences of the 1950s emerged. At this time also, standardization began to appear in the design of computer hardware. This was made possible largely through advances in electronics, in particular, developments in solid state device technology.

This increased power, along with a more widespread familiarity with computers, resulted in greater demand for communications both between users and the central computer system, and between computer systems themselves.

As in the 1950s, for the hardware, the solutions for data exchange were particularly different for communications to different terminal devices. In the 1970s the first standards for communication were published, and in 1974 IBM announced Systems Network Architecture (SNA).

At its announcement, this communication architecture reflected the way information systems were designed and implemented. That meant a centralized and hierarchical architecture in which the hosts had all the processing power.

This design was due primarily to the cost of the processors. At that time, there was an informal formula (called Grosches law) that stated that the cost of having a given amount of processing power split into two systems was about 50% more than the cost of providing the same amount of power in a single computer.

Moreover, communication lines were expensive and their capacity to transfer large amounts of data was limited.

Distributed environments were still only in the design stage. They could become reality only in the 1980s, with the decreasing costs of electronic devices and communication lines.

In the 1980s, there was also widespread personal computing, which has brought very powerful workstation accessibility to most users. The need to have personal computers, as well as midrange systems, connected in various and complex ways to each other and to large hosts, was met by IBM in two further updates of Systems Network Architecture.

In 1983, IBM announced the network node type 2.1 and Advanced Program-to-Program Communication (APPC) or the LU 6.2 protocol (that is, a new method to make applications located in different systems communicate with each other). In 1985, IBM introduced a new kind of systems interconnection called the Advanced Peer-to-Peer Networking (APPN). These new protocols are supported or can be recognized by most of the IBM systems today, and with the latest releases of host communications software, the traditional “host controlled” networks can integrate, in a full function way, “peer-to-peer” portions of the network.

1.1 SNA Architecture

This short introduction to Systems Network Architecture is intended to give you a basic understanding and a common terminology reference for the structure and the components of IBM SNA. If you are familiar with the SNA environment, you can skip this section.

As discussed in the previous section, IBM introduced SNA in order to have a reference model for communications among data processing systems, to standardize communication formats and protocols, and to make it easier to connect different systems.

A layered approach was considered essential in order to have simple interfaces among the different functions that a network must accomplish.

Systems Network Architecture consists of three logical groups of layers. The total number of layers are seven. The groups of layers, which go from the entry point for an SNA end user (defined below) to the physical communication medium, are:

- Application layer
- Function management layers
- Transmission subsystem layers

The *application layer* is related to the actual data processing and the interaction of the users with the system. In this sense, another SNA term is used: *end user*. An end user in an SNA network is typically a person working at a terminal or an application program executing in a processor. The end users are the ultimate sources and destinations of information. This application layer is the seventh layer of SNA.

The *function management layers* are:

- The presentation services layer (the sixth layer)
- The data flow control layer (the fifth layer)

They control the logical connections among the end users. More detail about these layers will be given later in this chapter.

There are four *transmission subsystem* layers related to the establishment, the management and the termination of the links between network nodes and the routing of the information between end users. They are:

- The transmission control layer (the fourth layer)
- The path control layer (the third layer)
- The data link control layer (the second layer)
- The physical layer (the first layer)

This physical layer is not described by Systems Network Architecture because several different media can be used in an SNA network.

For each layer IBM has defined, there are formats and protocols to pass information to the upper and lower layers. See Figure 1 on page 5 for a complete picture of the SNA layers.

Figure 2 on page 5 shows the relationship between the SNA network layers and the end users. The combination of the first three layers is also called *the transport network* because these are the layers that enable actual routing and conveying of data. The transmission control layer together with the transport network, as already introduced, is called the *Transmission Subsystem*. network addressable units (NAU), which will be introduced later, interface the transmission subsystem on one side and the end users on the other. They are the access port to the network for the end users. All the layers, apart from the application layer, form the *communication subsystem*.

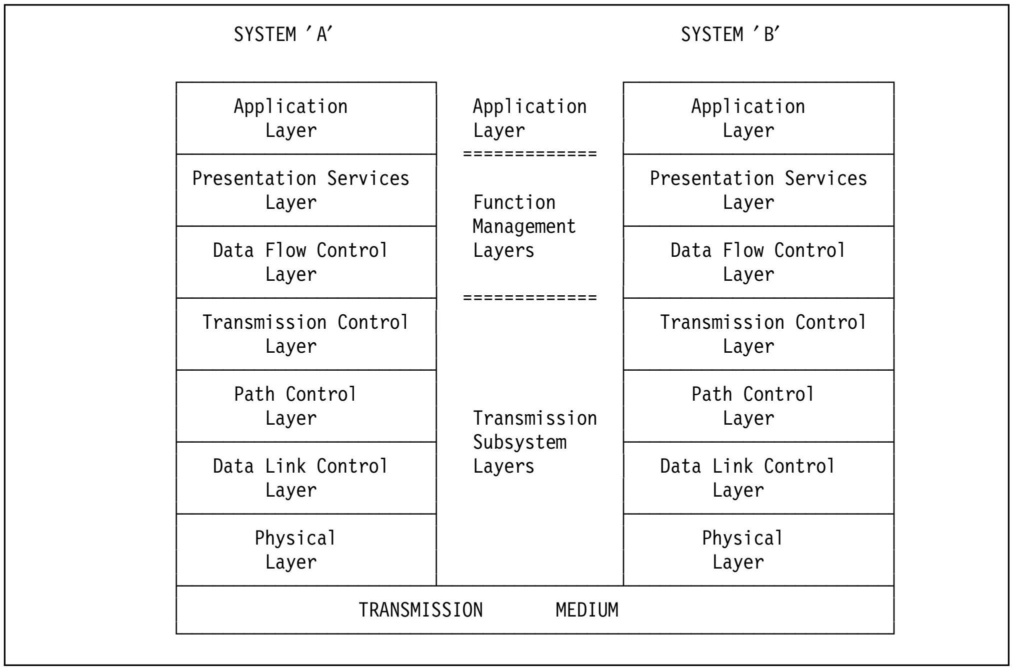


Figure 1. SNA Communication Layers Structure

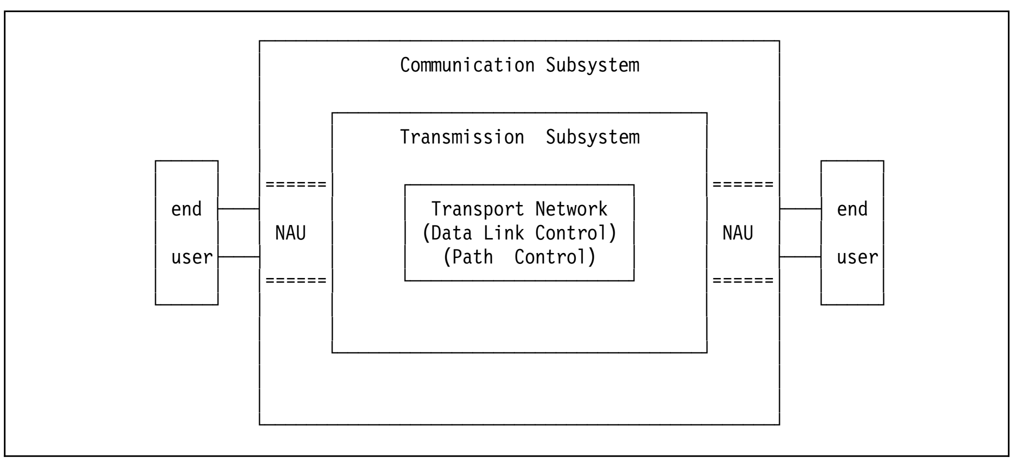


Figure 2. Relationship between End Users and SNA Network

Before entering into more detail about the way the information is packed and transmitted in an SNA network, more elements and entities have to be introduced. The end users have already been discussed, but the physical components that make up the network have not yet been defined.

The physical components of an SNA data communications network are called *nodes*, and nodes are interconnected by *data links*.

A host processor that contains a telecommunications access method (for example, IBM ACF/VTAM) is a *host subarea node*. The telecommunications access method allows the end users located in a host subarea node to access the network. Moreover, it controls the network, including all resources in all nodes.

A communication controller with its network control program (for example, IBM ACF/NCP) is a *communication controller subarea node*. This node, under direction from the host subarea node, controls the links and terminals attached to it. A communication controller can be connected to a host node or to another communication controller. In the former case, the connection is *local*, in the latter one, it is *remote*. In both cases, it can provide *intermediate network node (INN)* functions and/or *boundary network node (BNN)* functions. INN functions are for communication controller to communication controller connection; BNN functions are for connection with the network periphery.

All other nodes are called *peripheral nodes*. A peripheral node could consist of a controller with a single display device, a controller that supports several display and printer devices, or a processor.

End users report to peripheral nodes for being connected to the network, and usually they communicate with the end users located in host nodes.

A *logical unit (LU)* is the SNA component that handles communication between end users. A *physical unit (PU)* is the component that manages physical resources in each SNA node. The *system services control point (SSCP)* is the central point of control for the network.

Other types of control points have been designed for the new kinds of SNA networking. These are the *single-node control point (SNCP)* for low-entry networking (LEN) nodes, the *end-node control point (ENCP)* and the *network-node control point (NNCP)* for APPN nodes. The peripheral nodes owning these kinds of control points are particularly useful in the implementation of distributed environments.

Details about these SNA components, and LEN and APPN architectures will be covered later in this chapter.

The combination of the access method plus the communication controllers and the lines among them, plus the peripheral nodes and the lines between these nodes and the communication controllers, are also referred to as a *domain*. A domain is the set of resources “owned” (and controlled) by a single SSCP.

An example of a domain is depicted in Figure 3 on page 7. This figure shows also the interconnection of a small APPN network (the three peripheral nodes T2.1) to a domain. The peripheral node T2.0 connected to a node T4 and to the token-ring network can be both an IBM 3174 and an IBM Personal System/2* running OS/2EE Communications Manager Gateway feature. This is the subject of a following chapter. Node types will be discussed later in this chapter.

A *subarea* is a portion of domain. A subarea must contain either a host or a communication controller. Subareas are numbered in a domain and numbers assigned to them are used as the first part of the address for the resources located in that subarea.

Two or more domains can be interconnected into one consolidated network. This interconnection is called *multidomain* or *multisystem networking*.

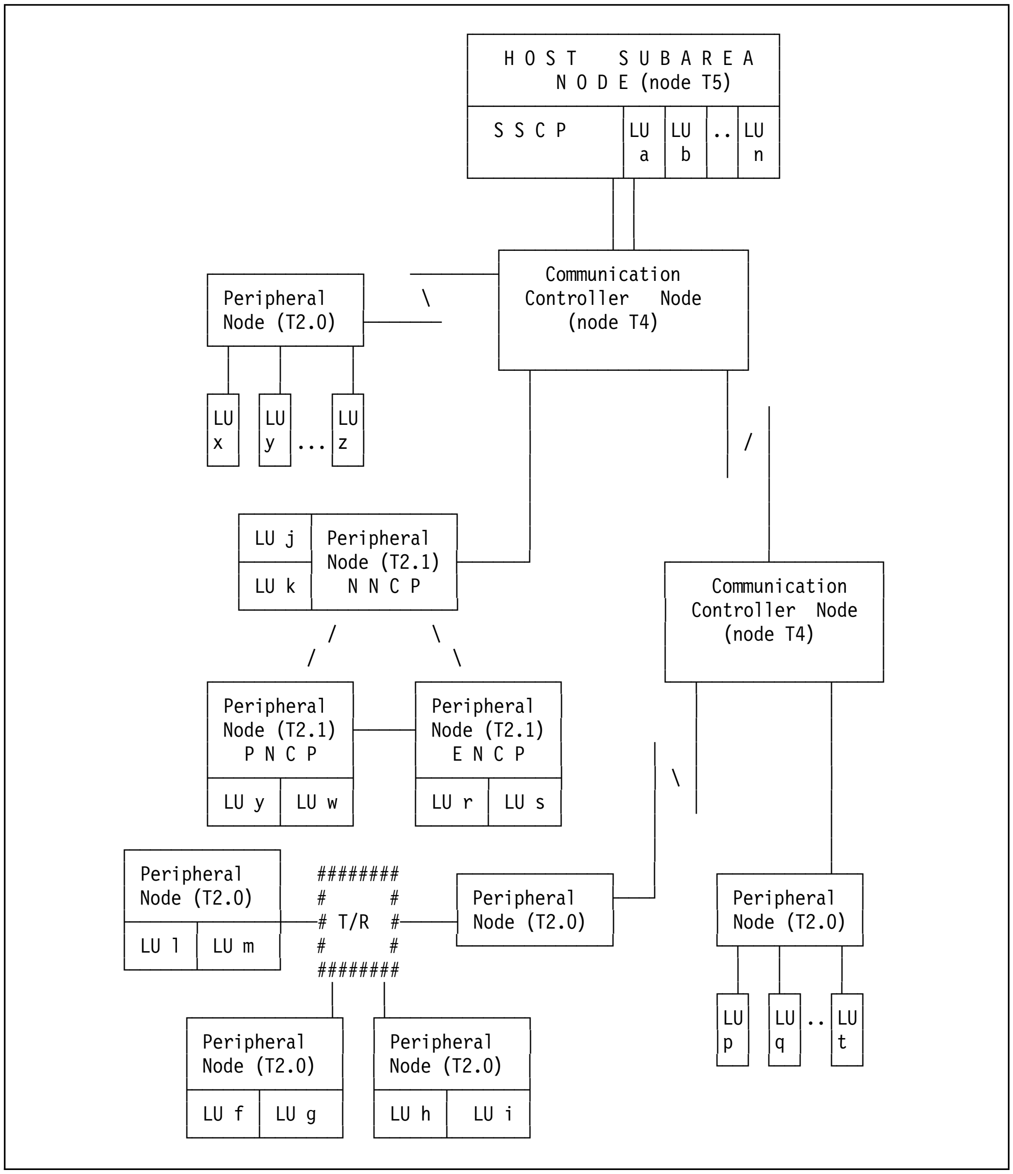


Figure 3. SNA Network Sample

LUs, PUs, SSCPs, SNCPs, ENCPs and NNCPs are called *network addressable units (NAU)*. NAUs are assigned network addresses so that data can be transmitted from one NAU to another NAU. A logical connection must exist between two NAUs in order for them to communicate with each other. This logical connection is called a *session*.

Different kinds of sessions can be established. Not all the combinations between two NAUs are possible sessions. The feasible ones are:

- SSCP-SSCP
- SSCP-PU
- SSCP-LU
- PU-PU
- LU-LU
- CP-CP (where CP stands for SNCP, ENCP, NNCP).

The basic mechanisms used to exchange data on a session are founded on:

- Request/Response Unit
- Request/Response Header
- Transmission Header

Starting from the end-user point of view and going into the physical network, at each layer some data is added to the application data. This data is necessary to the SNA network components to convey application data and to manage the network. The following list contains the names of the main parts of an SNA data frame:

Request Unit (RU):
Normally contains user data but may contain control information to assist in the routing of particular messages or in the management of the network.

Function Management Header (FMH):
Contains function management data which comes first in the RU, before user data. Not all the RUs contain an FMH. If these bytes are present, a bit will be set in the RH. In fact, these bytes can be seen as an extension of the RH.

The length of an RU depends on network products. Usually, each SNA product allows the users to configure a maximum and a minimum length for this field.

Request Header (RH):
Contains control information about the format and the type of the associated RU. RH field length is fixed at 3 bytes.

Basic Information Unit (BIU):
It is formed by the combination of RH and RU. It is the fundamental data unit handled by path control.

Transmission Header (TH):
Contains control information required by path control to manipulate BIUs (routing information, type of flow, sequence number of the BIU).

Format Identifier (FID):
The first byte of the TH. It is always present in an SNA frame. It specifies which of the five types of TH format has been used. Each format causes the TH field to have a different length. The following are the FID types:

FID type 0: used for communication between nodes T5 and nodes T4, which handle communications with pre-SNA devices. The related TH field is 10 bytes long.

FID type 1: used for communication between nodes T5 and nodes T4 or between two nodes T4 (it is used in the early SNA networks). The related TH field is 10 bytes long.

FID type 2: used for communication between nodes T5 or nodes T4 and an adjacent node T2. The related TH field is 6 bytes long.

FID type 3: used for communication between nodes T5 or nodes T4 and an adjacent node T1. The related TH field is 2 bytes long.

FID type 4: used in the same communications in which FID type 1 can be used (FID type 4 is used in the latest SNA networks). The related TH field is 26 bytes long.

Currently, the most used FIDs in SNA networks are FID2 and FID4. For the OS/2EE Communications Manager, only FID2 will be discussed. See the section on type 2.1 node, later in this chapter, for details about the format of TH type FID2.

Path Information Unit (PIU):
Consists of TH and relative BIU.

Basic Transmission Unit (BTU):
May contain more than one PIU depending on the use of blocking or not. This is the data exchanged between the data link control layer and the path control layer.

Basic Link Unit (BLU):
The basic unit of transmission on the link. For instance, for SDLC the BLU is a frame of this format:

FLAG, ADDRESS, CONTROL, BTU, Frame Check Sequence, FLAG

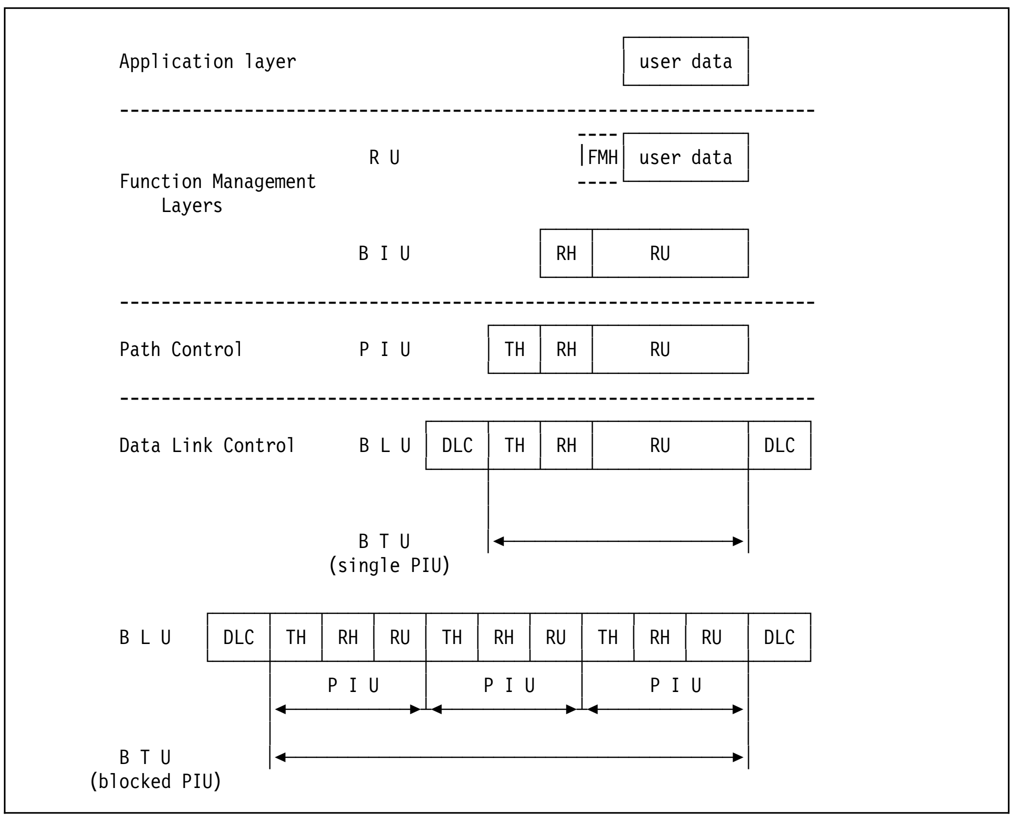


Figure 4. SNA Frame Format

1.1.1 Logical Units

SNA products are classified according to the SNA functions that they support. The logical unit(s) that reside in each product assumes that classification. The classification type of each LU designates a particular subset of SNA functions that the product can perform when its LU is in session with an LU in another product.

SNA defines LU types 0 through 7. The logical unit type designation is a convenient way of classifying SNA products according to the subsets of SNA functions they perform. It is useful in selecting a set of SNA products to use in a particular application.

Some SNA products support more than one LU type. For example, the logical unit provided by IBM CICS can participate in LU-LU sessions with LU types 0 through 6. LU types are characterized by their associated transmission services (TS) profile, function management (FM) profile and presentation services (PS) profile.

Before discussing these profiles, a brief description of the characteristics of each LU type will be given.

LU types are grouped in three main categories:

Host-to-terminal LU types: They define generic devices so that different terminal implementations are masked to the host applications. This allows a reduction in host application development.

Program-to-program LU types: They define the protocols for the communication between processes that can be single programs or application subsystems.

User-defined Session Protocol LU type: This is the LU 0 that allows users to define their own profiles. Usually it is used for program-to-program communication. Also some IBM products such as NetView and file transfer program (FTP) use this LU type. Additionally, the IBM database product IMS defines a version of LU 0 for user access from intelligent devices. This definition is called SLU type P.

This is the complete list of LU types (note that LU type 5 has never been defined in SNA architecture):

LU type 0: These are implementation-dependent and do not fall within the groupings of profiles defined by SNA. IBM 4700 and IBM 8100 connected to IMS systems use this LU type.

LU type 1: Type 1 logical units are for application programs that communicate with terminals. They support single or multiple devices in an interactive, batch, or distributed processing environment. Devices may include consoles, printers with keyboards, normal printers, diskettes, disks and card units. This session normally uses the SNA Character String (SCS). Also LU 1 is sometimes used for processor (program)-to-intelligent node communication. This is the case for some program products that run in an IBM communication controller, such as network control program packet switching Interface (NPSI) or network terminal option (NTO).

LU type 2: Type 2 logical units are for application programs that communicate with single display devices in an interactive environment. The session uses the SNA 3270 data stream and operates in half-duplex flip-flop mode. The IBM 3278 is an example of a type 2 logical unit.

LU type 3: Type 3 logical units are for communication between application programs and a single printer using the SNA 3270 data stream. The LU in the IBM 3274 that is associated with a 3287 printer can be bound as a type 3 logical unit.

LU type 4: Type 4 logical units are similar to type 1 LUs, but they also support terminal-to-terminal communication.

LU type 6 and LU type 6.1: Type 6 and 6.1 logical units are for data communication between application subsystems. IBM CICS and IBM IMS are examples of implementations of type 6 and 6.1 logical units. An IBM CICS logical unit can establish a session with another IBM CICS logical unit or with an IBM IMS logical unit.

LU type 6.2 (also called APPC): Type 6.2 logical units use a device-independent protocol for process-to-process (or program-to-program) communication in a distributed environment. Their operation is symmetric (peer-to-peer), with both partners having equal control over the resources allocated by the SNA session over which they communicate.

LU 6.2 is significant in the sense that it changes many concepts in SNA. For example, with LU 6.2, the end user is a program (called a transaction program), not the LU itself.

LU type 7: A type of logical unit for an application program that communicates with a single display terminal in an interactive environment, for example, a session involving an application program in an IBM System/36* and an IBM 5250 display terminal.

Other important classifications concern the method to establish a session between two LUs and how the LUs network access is controlled. In fact, an LU can be:

- *Primary* (in this case, it can start a session or, with a more appropriate term, bind another LU on a session).

or

- *Secondary* (in this case, it can only ask the LU that would be its partner to start a session).

The primary LU will also be responsible for the session management.

The other classification is between:

- Independent Logical Unit

and

- Dependent Logical Unit

If the LUs are located in a network where a host subarea node is present, they are usually connected to the SSCP that manages that network. An LU that has a session with an SSCP is a *dependent LU*. It will use the SSCP-LU session to send or to receive requests to establish sessions with other LUs, according to the role that it plays. An LU that has no SSCP-LU session is an *independent LU*.

Independent LUs are very important when implementing a low entry network because they do not need a central control point for the establishment of the sessions. In fact, independent LUs are usually housed in peripheral nodes and can talk to or through both a subarea network and an APPN network. Type 6.2 LUs can exploit all their capabilities only if they are independent LUs. In this case, they can have parallel and multiple sessions.

An LU is said to have parallel sessions, when it has more than one session, simultaneously, with the same partner LU.

An LU, instead, has multiple sessions when it is in session with more than one partner LU.

At the beginning of this section, the three types of profiles that define the capabilities of an LU were introduced. Each profile is related to one of the three SNA layers that are covered by LU functions:

- *Presentation Services Profile (PSP)*: This profile is related to the sixth SNA layer and defines the rules about the interface between the end users and the LU. This interface varies depending on whether the end user is a device (for example, a display terminal) or a program. For LU 6.2, the presentation devices are defined via the LU 6.2 verbs. For the LU 2 they are defined via the IBM 3270 data stream.

- *Function Management Profile (FMP)*: This profile is related to the fifth SNA layer and defines the SNA commands that can be used by an LU. There are three main categories of SNA commands:

  – Network control commands
  – Data flow control commands
  – Session control commands

  The function management profiles are numbered. Valid numbers are 0, 2 through 5, 17 and 18. All the other numbers are reserved.

```
* Primary and secondary half-session use immediate control mode
  and immediate response mode
* Only single RU chains allowed
* No compression
* Primary half-session sends no Data Flow Control (DFC) RUs
* Secondary half-session may send LUSTAT
* NS headers are allowed
* No FMHs
* No brackets
* No alternate code
* Normal flow send/receive mode is HDX-CONT
* Secondary half-session wins contention
* Primary half-session is responsible for recovery
```

Figure 5. Sample of FMP: Function Management Profile 0

- *Transmission Subsystem Profile (TSP)*: This profile is related to the fourth SNA layer. It specifies the transmission characteristics of the session. Like the function management profile, it is numbered and valid numbers are 1 through 5, 16 and 17. Parameters defined in these profiles are the use of pacing, of sequence number or identifier, maximum RU size and which type of data flow are supported. A secondary dependent LU has four types of data flow if it is in session with another LU:

  – LU-LU normal flow
  – LU-LU expedited flow
  – SSCP-LU normal flow

– SSCP-LU expedited flow

```
* No pacing.
* Identifiers rather than sequence numbers are used on the
  normal flow.
* SDT,CLEAR, RQR, STSN are not supported.
* No maximum RU sizes for the normal flow are specified.
```

Figure 6. Sample of TSP: Transmission Subsystem Profile 1

The choice of PSPs, FMPs and TSPs is determined by the LU type. Each LU type has a subset of allowable profiles.

1.1.2 Physical Units and/or Node Types

As SNA has evolved in function, new concepts have been added and old ones changed. This process has caused many changes in the terminology used to describe SNA. One of the terms that has been modified is *physical unit*. The new term to refer to this SNA entity is *node type*. This terminology change also reflects the updates in function that have been brought to the architecture.

The subarea and peripheral nodes are also referred to as node types 5, 4, 2.1 and 2. *Type 5* is for host subarea node; *type 4* is for communication controller subarea node; and *types 2.1 and 2* are used to refer to peripheral nodes.

The PU type or node type reflects the functional capabilities of the node. Thus, type 5 nodes have more functional capability than the other node types. A type 2.1 node is very similar to a type 2 node except there is a *single-node control point (SNCP)*, or an *end-node control point (ENCP)*, or a *network-node control point (NNCP)* in the type 2.1 node. These CPs provide a subset of SSCP functions. This capability allows type 2.1 nodes to communicate with each other on a *peer-to-peer* basis. On the other hand, type 2 nodes can only communicate with subarea nodes.

There is a third type of peripheral node used in earlier products: That is type 1 or *terminal node*. There are a few devices that fall into this category, such as an IBM 6670 Information Distributor and an IBM 3767.

Nodes are part of the *transport network* (refer to Figure 2 on page 5). A node type is characterized by its capability to manage links and to route information on different links. Node features depend on the CP (SSCP or one of the PNCPs) that coordinates its activity.

The following is the complete list of the SNA node types (note that node type 3 has never been defined in SNA architecture):

Node type 1:
The only implementations designed for this node type provide single LU support. This means a limited set of transmission services and the absence of PU services into the node.

Node type 2 (or type 2.0):
They provide services for the LUs located in the node and have no capabilities to route information as intermediate nodes.

Node type 4:
These nodes are usually managed by NCP and essentially provide routing services as intermediate nodes. No session services are available.

Node type 5:
These nodes are IBM System/370* host and provide session services to SSCP and the local LUs. They can also be intermediate nodes.

Node type 2.1:
This is the latest node type and has the same capabilities of the node type 5, but can be an intermediate node both for subarea networks and APPN networks.

1.1.3 Control Points

Control points are located both in type 5 nodes and in type 2.1 nodes. A control point is what allows an SNA network to achieve its service of information delivery. The main tasks of an SNA control point are:

- Connectivity services (links activation)
- Directory services (definition of all the resources in the network)
- Session services (establishment of the sessions among NAUs)
- Routing services (control of the data flow)
- Management services (maintenance and tracking of the network operations)

The control point housed in a type 5 node is called system services control point (SSCP). It is contained in the telecommunication access method (IBM ACF/VTAM) installed in all the IBM S/370s that need to be connected to a network. This was the first kind of control point to be designed. It reflects the early SNA architecture. This means that as the network was supposed to be hierarchical, an SSCP has to centrally manage all the links, all the nodes, and generally all the network resources. For this reason, the portion of the network under SSCP control is called a domain.

There are different implementations of *peripheral node control point (PNCP)* (that is, a control point housed in type 2.1 nodes). So far, these CPs have been introduced:

- Single node control point (for LEN nodes)
- End node control point (for APPN nodes)
- Network node control point (for APPN nodes)

All these control points are PNCPs. Usually, there is some confusion among these terms, because the first implementation of a type 2.1 node is for low entry networking, and so to talk about PNCP or SNCP is the same thing. With the introduction of Advanced Peer-to-Peer Networking, this is no longer true.

As for the SSCP, these kinds of CPs reflect the structure of the network in which they are inserted. The differences between LEN and APPN is discussed in 1.2.2, “Low Entry Networking (LEN)” on page 22.

1.2 Distributed Environments

In recent years, small processors have become more attractive for a large number of organizations. This has depended on the increased capabilities and the reduced cost of such systems. Moreover, the diffusion of new kinds of networks (local area network, X.25, ISDN) and the capillary spread of the traditional networks has made it possible to have cooperation between different systems, using the connectivity provided by the networks.

The new approach to data processing is to distribute processing power where needed, as well as to distribute data in the closest location to the user of this data. This approach is also known as *cooperative processing*. In order to build a real cooperative and distributed environment, it is necessary to enable a full connectivity as well as, with a new terminology, the *interoperability* among systems.

Moreover, it is important in the design of a distributed environment, to consider the high-level services that may be provided by the networks. Examples of these services are the distribution of data or files, the distribution of documents under several different forms (text, image, voice, etc.), and the network management. For SNA networks, these services are the *application services*. These services will be discussed later in this chapter.

A big challenge, today, is the design of an integrated network for an enterprise. The requirements for such a network can be listed as follows:

- Handling of traditional wide area network traffic
- Integration of local area and wide area networks
- Support for any-to-any communications
- Use of standardized interfaces available directly to the users
- Management of the network as a single integrated entity
- Availability to support industry standards such as X.25, and platform for new standards such as ISDN
- Provision of application services

SNA node type 2.1 can be considered a major step toward the fulfillment of these requirements.

1.2.1 Node Type 2.1

The introduction of the type 2.1 node has been revolutionary for the SNA architecture. In fact, until the introduction of type 2.1 nodes, peripheral nodes were terminal or cluster controllers, or minicomputers accessing the network in the same way. These nodes are type 2.0 and type 1 nodes, and they can communicate only with a host node. They interface the boundary functions of the network, and they need to be controlled by the host. This hierarchical design of early SNA completely met the requirements of the 1970s information systems for optimized communications between a host and its periphery. But this design presents limitations if distributed data processing is required in our environment.

In fact, the characteristics of the LUs located in type 2.0 and type 1 nodes can be summarized as follows:

- They are always *secondary LUs*: This means they can communicate only with a primary LU, to which, eventually, they can request a session with them (this request must be presented to their SSCP).

- They are always *dependent LUs*, that is, they need to have a session with their SSCP.
- They can have *only* two sessions, one with a primary LU on a host and the other with the host SSCP. Between these sessions, only the LU-LU can carry application data.
- They must be activated by the host SSCP.

Only 255 LUs can reside in a type 2.0 node because type 2.0 nodes use an FID 2 transmission header. This TH is formed by 6 bytes divided into the following fields:

FID, reserved, DAF, OAF, Sequence number

- Format identifier (FID): 1 byte, with the first 4 bits representing the type of FID, the following 2 bits mapping field (00 if middle segment, 01 if last segment, 10 if first segment, 11 if only segment); then there is a reserved bit, and the last bit indicates the type of flow (0 normal or 1 expedited).
- Reserved field: 1 byte.
- Destination address field (DAF): 1 byte.
- Origin address field (OAF): 1 byte.
- Sequence number (of the frame): 2 bytes.

The address fields represent the *local address* for the LUs; that is, the addresses are valid only within their own node. With a byte, only 256 NAUs can be addressed. Address 00 hex is reserved for the local PU, while the LUs can take any other address.

Devices implementing a type 2.0 node cannot be connected end-to-end, but even if this were feasible, LUs housed in this device could not communicate, as they are only secondary and dependent.

For this reason, most of the small IBM systems presented the capability to be connected directly, outside of the SNA architecture. An example of early direct connections end-to-end are the links between two IBM 8100s, or two IBM S/36s, or two IBM 5520s. These connections were largely incompatible and could be achieved only between the same model of system. They used a subset of SNA protocol and formats. For example, two IBM 8100s used FID type 1, and two IBM S/36s used FID type 3, while two IBM 5520s used a modified FID type 3.

Refer to Figure 7 on page 17 for an illustration of this scenario.

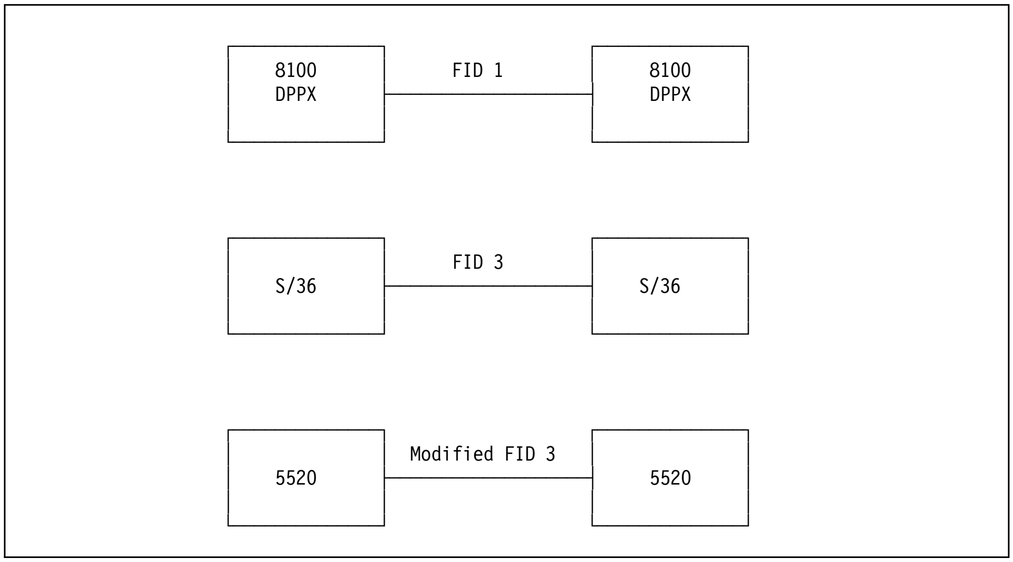


Figure 7. Early End-to-End Connection between Small Systems

The answer to the need to have small systems communicating with each other with a common protocol was provided with the introduction of the type 2.1 node. This extension to SNA architecture defines a protocol for *peer-to-peer* connections. This node type overcomes the main limitation of type 2.0 nodes by means of the introduction of an internal control point: the *peripheral node control point*.

The presence of a CP allows type 2.1 node to have:

- Dependent and independent LUs
- Primary and secondary LUs
- LUs having more than one partner LU simultaneously (multiple sessions)
- LUs having more than one session with the same partner LU (parallel sessions)
- No need to have SSCP-LU sessions
- No use of VTAM unformatted system services (USS)
- No need to have an SSCP-PU session
- PU and LUs can be activated locally

A type 2.1 node can have any type of LU, but it only uses all its features when using LU 6.2 protocols. Later in this chapter, we explain that in SNA application services, network services in any kind of connection are based on the LU 6.2 protocol.

The FID used by the LUs located within type 2.1 nodes is still a FID 2 format. That is, the associated TH is 6 bytes long, but the meaning of the DAF and OAF fields is no longer the destination LU and the origin LU address. The two bytes of the DAF and OAF fields are considered a single field, representing an identifier for the sessions between the LUs on that node and the partner LUs.

The combination of DAF and OAF field, plus the DAF/OAF assignment Indicator bit in the second byte of the TH, are now called *local form session identifier (LFSID)*. An LFSID has meaning only in the node that has generated it, and must be unique for each session that traverses a link. A pool of LFSIDs is assigned for each link from a type 2.1 node.

The use of LFSIDs allows a type 2.1 node to have up to 128K sessions between LUs. This number, representing the session limit of a type 2.1 node, is much greater than the session limit for a type 2.0 node (255 sessions). This upper limit gives huge communication capabilities to a type 2.1 node.

Moreover, the use of the LFSIDs frees SNA architecture from any static relationship between the LUs and the network addresses. LFSIDs are dynamically allocated[1], and an LU can have as many sessions as desired with any other LU. That is, an LU can have both multiple and parallel sessions.

Because a type 2.1 node can have both dependent and independent LUs, and these LUs use LFSIDs to identify their sessions, if dependent LUs existed in a node, they have assigned the space X′0101′ through X′01FF′ as LFSIDs for their LU-LU sessions (note that a dependent LU can have only a single session). See Figure 8 for a complete reference of the subdivision of session identifiers.

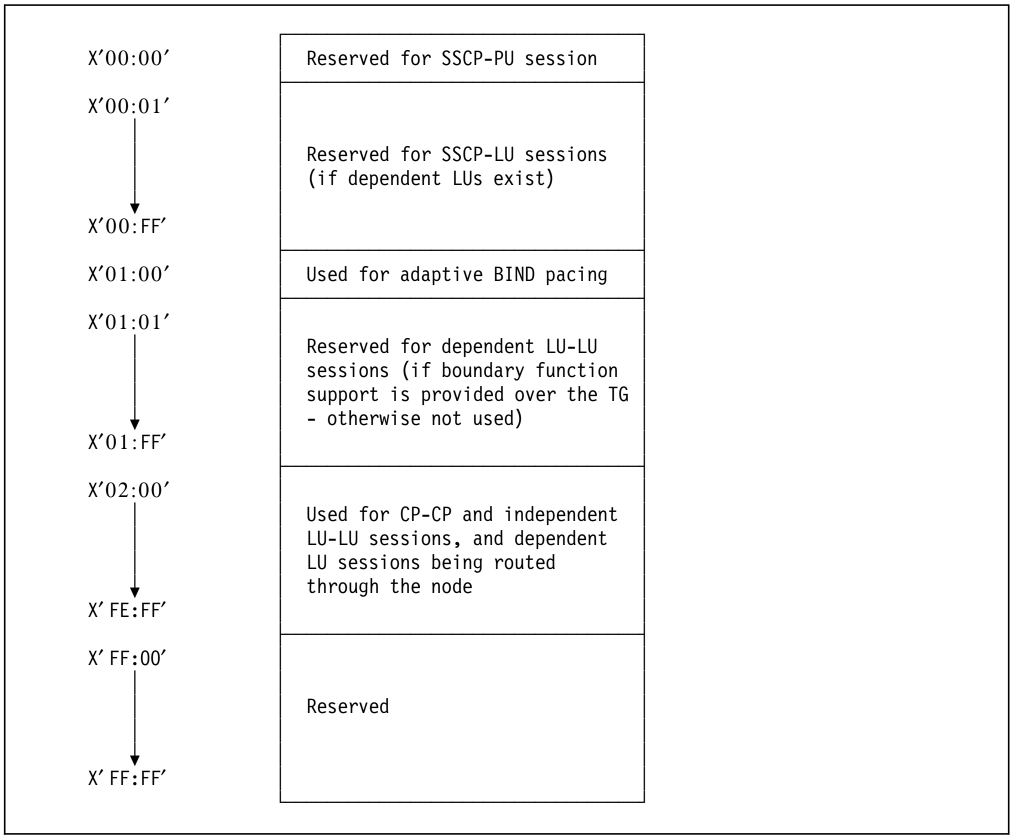


Figure 8. Session Identifier Address Space Subdivision

If a type 2.1 node is connected to a subarea network, because LFSIDs cannot be handled by LUs in this network, IBM ACF/VTAM must translate them. IBM ACF/NCP maintains a directory of the LFSIDs and relative subarea LU addresses. In case of parallel sessions, additional addresses are reserved for the LUs to preserve session identifier uniqueness. There are other cases for which it is necessary to have a unique identifier for a session, different from the addressing sequence formed by end addresses (or LFSID), nodes, and links. This is when information about a session is carried on another session, or to

1 The allocation of LSFIDs, as local addresses on a link, is similar to the allocation of logical channel addresses to virtual circuits in X.25.

provide a long term session identifier (for example, for problem determination purposes or accounting services).

This is achieved with the use of the *fully qualified procedure correlation identifier (FQPCID)*. IBM VTAM Version 3.2 and IBM NCP Version 5.2 support the generation and the receipt of fully qualified PCIDs to avoid ambiguous local address translation during session establishment.

PCIDs are essentially random numbers qualified by the name of the control point that has generated them. A PCID always contains an eight-byte long random number. The “full qualification” of PCID is done, using the name of the network in which it is located as a prefix for the control point name. This way of qualifying entities is also used for the names of the other network resources (for example, fully qualified partner LU name is a configuration parameter of OS/2EE Communications Manager, and it is obtained by putting the name of the network in which it is located before the partner LU name).

The FQPCIDs are communicated to the partner LU via the BIND. BIND is a network command issued by an NAU to request a session to another NAU. Not all the NAUs are allowed to request a session using BINDs.

Dependent LUs in peripheral nodes cannot send a BIND. The LU that issues the BIND will be the primary LU and will be responsible for the session. Three different BIND types are defined in the SNA architecture:

- Non-Extended BIND
- Extended BIND
- LU 6.2 BIND

These BINDs can be up to 512 bytes long. They contain information for establishing the session and are sent to the CP of the node in which the partner NAU is located.

The difference among these kinds of BINDs is in the amount and the type of information that they carry. The following is the information carried by an extended BIND:

- A control vector and the control vector included indicator (CVII)
- FQPCID vector
- Adaptive session pacing indicator (ASPI)
- Whole BIUs required indicator (WBRI)
- Network names control vector (contains the name of the partner LU and of the sending LU; they may be fully qualified names)
- Class of service (COS)/transmission priority field (TPF) control vector
- Mode control vector

If the CP is an SSCP, the following processing is performed on the receipt of a BIND from its boundary functions:

- Locate the destination LU in the network (the same would be done by an NNCP).
- Communicate with the CP owner of the destination LU.
- Activate a route through the network for the new session.
- Allocate network addresses for the LUs involved in the session (note that a BIND can be sent to an SSCP only from a type 2.1 node, and type 2.1 node uses LFSIDs).

- Locate an appropriate class of service table entry.
- Extend the BIND if needed; that is, calculate and assign an FQPCID for the session (if not already present).

So far, session characteristics and ways of establishing sessions have been discussed. In order to establish a session, an active link is needed.

Links are activated by CPs. An SSCP is in charge of the activation of the links of its domain. A PNCP is in charge of the activation of the links with its adjacent nodes.

An exchange identification (XID) between the nodes at the ends of a link is required for the establishing of the link. Format 3 XID (that is, a format 3 I-field (Information field) is contained in the SDLC frame) is the one used for both switched and leased links between type 2.1 nodes or between type 5 and type 2.1 nodes. Leased links between type 5 and 2.0 nodes are activated via the Set Normal Response Mode (SNRM) command by NCP. Format 0 XID is required for switched connections between a communication controller and a type 2.0 node.

The following is the information passed with an XID 3:

Node Identification
It is composed of the IDBLOCK (identifying the machine model; for OS/2 Extended Edition this value is X′05D′) and the IDNUM (node identifier) fields. They will be ignored if CPNAME is indicated.

BIND Segmentation Information
This is indicated by the whole BIND PIUs generated indicator bit. This information signals if BIND PIU can be segmented.

ACTPU Suppression Indicator
This is sent by a LEN node to indicate that it does not contain any dependent LUs, and therefore does not need an SSCP-to-PU session.

Link Station Characteristics
It depends on the type of link. It is used to resolve the link station role (primary, secondary or negotiable). It determines whether modulo 8 or modulo 128 is used for the SDLC sequence numbers.

Network Name
The name of the network of which the attaching node is part. If the connection is between a type 2.1 node and an IBM S/370 host, the name of the network must be the same as that of the SSCP, unless IBM VTAM Version 3.3 and IBM NCP Version 5.3 are installed. In that last case only, the network name of an attached type 2.1 node may be different from that of the VTAM network.

Product Set Identification
For hardware, the machine type, machine model, plant of manufacture and machine serial number are sent; for software, the program name, version and release, date of linkedit and loadlib name are sent. OS/2EE Communications Manager sends hardware information.

CP Name Name of the control point in the node that sends the BIND. It can be used as node identification in place of IDBLOCK and IDNUM. For this reason, the SSCPNAME parameter must be specified when configuring IBM VTAM Version 3.2, while before, it was mandatory only if the SSCP was a gateway SSCP.

Maximum PIU size
The maximum PIU length that the node which sends the BIND is able to receive. This is the same parameter that is specified for a type 2.0 node with MAXDATA parameter of the PU statement in VTAM.

If the type 2.1 node is connected to a subarea network, this node must be defined to VTAM and NCP. The VTAM definition for this node is a PU statement as for a type 2.0 node, but the XID= parameter is specified as YES to let VTAM know that it is a type 2.1 node capable of handling the XID protocol. NCP will send or receive the XID; it will reject it if the peripheral node sends an XID requiring it to be a primary link station. VTAM/NCP must always have a primary link station role.

Connectivity for a type 2.1 node can be provided via different kinds of physical media:

- SDLC (Switched and leased over an analog or digital service)
- token-ring local area network
- ETHERAND Network (supports the DIX Version 2.0 and IEEE 802.3 Ethernet LAN specification)
- PC-network local area network (between IBM PCs and IBM PS/2s)
- X.25 network
- S/370 channel connection
- ISDN basic rate connection (available as a special product in some countries)

The support of all these kinds of connections allows the integration of IBM systems to the main networks available today. Moreover, this support enables a global offering for a complete networking structure for today's enterprises.

To complete the discussion on type 2.1 nodes, the internal structure of this kind of node is now introduced.

The internal structure of a type 2.1 node is formed by five basic components:

Logical Units
The interface into the network for application transaction programs.

Control Point
The component which manages the resources of the type 2.1 node. It is composed of the following three subcomponents:

- Configuration Services
- Session Services
- Address Space Manager

Path Control
The message routing mechanism between logical units.

Data Link Control
The component that provides the necessary link protocols to interface with the supported transmission media.

Node Operator Facility
The interface between the node operator and the control point.

All these components are depicted in Figure 9 on page 22.

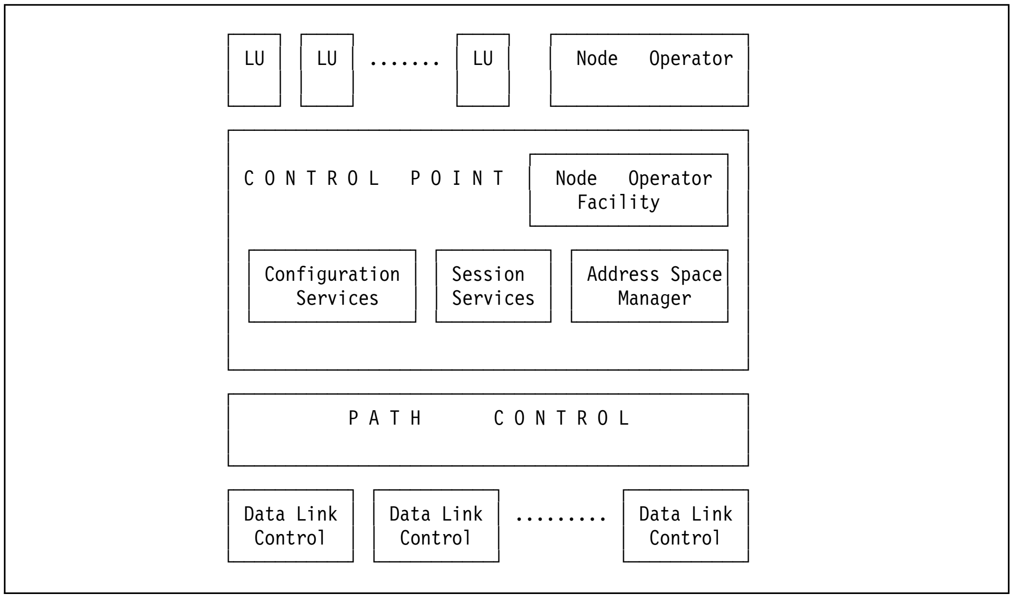


Figure 9. Type 2.1 Node Components

The configuration services subcomponent of the control point provides two main functions:

- First, it maintains a database of all the node resources (for example, link characteristics, local LU definitions, and LU definitions of adjacent link stations).
- Second, it activates and deactivates the links and provides link failure procedures.

The session services subcomponent initiates and terminates sessions. It provides directory service function; that is, it correlates the partner LU name with the node that contains it and the link used to access that node. Moreover, it accesses configuration services to check if a link is active for session initiation, or to provide notification of the possibility to deactivate a link when a session is terminated.

The address space manager assigns and receives LFSID when sessions are, respectively, initiated or terminated. It is interfaced by the session services and by the path control.

1.2.2 Low Entry Networking (LEN)

SNA low entry networking was announced in 1983 as an architecture to standardize a common set of protocols for the communication between unlike systems (IBM PC, IBM S/36, IBM S/88, IBM S/1, IBM RT PC*, etc.).

Low entry networking architecture defines path control network (transport network) connectivity between *adjacent* type 2.1 nodes. LEN *does not* define communication between nonadjacent nodes. This means that routing through intermediate nodes is not expected and, where needed, must be provided by special function in the intermediate nodes.

Subarea networks, under the control of IBM VTAM Version 3.2 and IBM NCP Version 5.2 (or Version 4.3), can provide routing between nonadjacent LEN nodes connected to them. IBM VTAM Version 3.3 provides the same support for LEN nodes that are connected through IBM 4361 or IBM 9370 Integrated Communication Adapters (ICA).

Figure 10 shows some of the possible connections between type 2.1 nodes. The connections drawn with '=' and '-' characters are normal direct end-to-end connections. They are not affected by the presence of the subarea networks. The connections drawn with '#' and '.' characters, instead, use the subarea networks routing mechanism. In the case of the '#', the two type 2.1 nodes are connected to the same subarea network and so the data is handled by only an IBM NCP. In the case of the '.', the connection is established through different subareas.

A connection between two type 2.1 nodes can also be established between the two host nodes. The combination of the type 5 and type 4 nodes, on the right side of Figure 10, communicates with the combination of host and communication controller on the left side of the figure, as a composite type 2.1 node. This connection is similar to the one between two small systems. This is called a *casual* connection between systems. This is another feature available from IBM VTAM Version 3.2 and IBM NCP Version 5.2 (or Version 4.3). This feature is also the one that enables the connection marked with '*'.

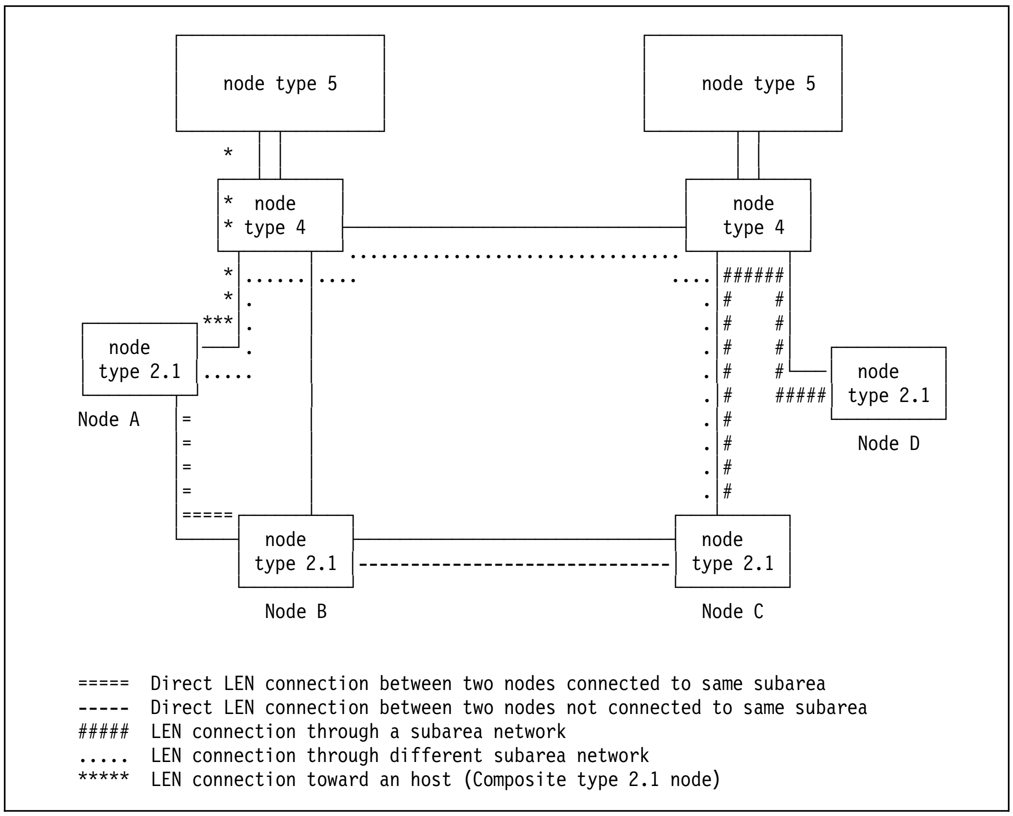


Figure 10. LEN Connections in an SNA Network

LEN was mainly intended to support communication between LUs 6.2 within type 2.1 nodes. LU 6.2 located in LEN nodes are capable of having parallel and multiple sessions. In order to start a session, an LU 6.2 within a LEN node has only to send a BIND to its partner LU, because it is independent.

LEN nodes still have some restrictions in order to fulfill the requirements previously listed. These restrictions can be summarized as:

- A peer link node can be primary or secondary. Although a type 2.1 node can support multiple links and multidropped links, the link connections must be defined either primary or secondary; this can require additional links to achieve an any-to-any connectivity.
- No intermediate network routing is provided by a LEN node.
- Network definitions are static like VTAM subarea definitions.
- Only single links are possible between two LEN machines.

The combination of subarea and peer restrictions led to a set of requirements for APPN development.

1.2.3 Advanced Peer-To-Peer Networking (APPN)

Advanced Peer-To-Peer Networking (APPN) was introduced in the May 1985, *IEEE Journal on Selected Areas in Communications*. This article discussed the possibility for a type 2.1 node to assume a *network node role* and not only a *peripheral node role*.

For more detailed or advanced knowledge of the topic, readers are referred to the following publications:

- *APPN Architecture and Product Implementations Tutorial*, third edition, IBM Corp., 1994, GG24-3669
- *SNA APPN Architecture Reference*, fourth edition, IBM Corp., 1991, SC30-3422.

The APPN architecture does not add new components to type 2.1 nodes structure, but it extends the functions performed by these components. The main extensions are to the control point and to the path control components. This is a brief list of the new services (services normally provided by similar components in a type 2.1 LEN node are shown in parenthesis):

Configuration services
(links activation and deactivation)

- Control point-to-control point session
- Topology database update

Session services
(local and adjacent LUs directory mapping)

- Network directory cache
- Directory search/locate function
- Route selection services

Path control (FID 2 addressing using LFSIDs)

- Intermediate routing
- Flow and congestion control

Some of these new services create the infrastructure to enable APPN functions to be performed. For example, the control point-to-control point session allows the topology and the directory database to be created and updated, as well as the search of a route between two resources to be performed.

Not all these functions must be provided by all nodes in an APPN network. In fact, an APPN network can be formed by:

- Network Node

- End Node
- LEN Node
- Composite Type 2.1 Node (IBM S/370s using IBM VTAM Version 3.2 and IBM NCP Version 5.2 or 4.3)

APPN Network Nodes: APPN network nodes enable the connections between non-adjacent type 2.1 nodes. These nodes are the core of APPN networks and perform intermediate routing functions. These functions include the route selection services. The way in which route selection is done is one of the main differences between IBM AS/400 and IBM System/36 APPN implementation. The APPN architecture, as defined in the IEEE publication, presents a network topology that provides the use of weights on the links, called *transmission groups*, and on the nodes.

These weights are used to choose the *most preferred route* if multiple routes exist between two nodes. Weights are determined by some parameters contained in two tables:

- Link Class of Service Table (TGCOS)
- Node Class of Service Table (NCOS)

They may be affected by further user-defined values in these tables. When activating a node, values in COS tables, for that node and its links, are propagated throughout the network to update the topology database of the other network nodes.

The node characteristics are summarized via the *route addition resistance* and the *congestion*. This last parameter is determined and propagated dynamically. A node notifies a congestion situation when reaching 90% of the maximum intermediate sessions active. It quits the congestion condition when this value goes below 80%.

The link characteristics considered when choosing a route are:

- Link speed
- Security
- Cost per connect time
- Cost per byte
- Propagation delay
- Three user-defined fields in TGCOS table

These values are summarized in a single value that will be the weight.

The route chosen is the one for which the sum of all weights for intermediate nodes and links is minimum.

IBM S/36-based APPN networks do not allow this congestion control mechanism achieved via the use of user-defined parameters. IBM S/36s have fixed values for the user-defined parameters. IBM AS/400s configured as network nodes enable users to define node and link parameters for congestion control. Networks of mixed IBM AS/400s and IBM S/36s as network nodes can be installed, but in this case, a careful configuration of the IBM AS/400s is needed if some congestion control is required.

APPN End Nodes: End nodes do not perform intermediate routing for APPN networks. Their functions are determined depending on the configuration of the CP-CP session. An IBM AS/400 can be configured to utilize a CP-CP session or it can be configured with no CP-CP session. In this second case, IBM AS/400 end nodes and IBM S/36 end nodes are similar. In fact, the use of CP-CP session between end nodes and network nodes is unique to IBM AS/400.

If there is not a CP-CP session, the user must configure all the characteristics of the two nodes and the link in both the nodes. Moreover, all the resources located in end nodes must also be configured in the network node to which they are connected. On the contrary, if a CP-CP session is present, an information exchange at the establishment of the link occurs, and the topology DBs can be updated dynamically.

End nodes with CP-CP sessions allow a more flexible management of the network. Definitions can be changed or added dynamically, and ENs are aware of the route to reach a remote location. They can use network services to find a route via a search request. The result of this search is a *route selection control vector (RSCV)* that is appended to each following BIND from that end node for the node specified in the search request. An RSCV contains the description of the total path from the origin to the destination.

The use of search request for end nodes is extremely interesting if considered together with the possibility of defining more than a single CP name for a node. This means that applications can be written without knowing the network topology, and only when executing will they find out where the correspondent application is located. Moreover, if different applications must be located on a certain node, the user and the developers do not need coordinated definitions because more CP names can be used to identify that node.

LEN nodes: LEN nodes are nodes that have no CP-CP session with other type 2.1 nodes. They have a limited perception of an APPN network. In fact, they consider the whole APPN network to which they may be connected as another LEN node. This means that the CP name of this LEN node must be configured in the NN to which it is connected, as are all the LUs located in this LEN node.

The NN to which it is connected provides the same network functions it provides to EN without CP-CP session capability.

Composite nodes: They are similar to LEN nodes. They can offer subarea services for connection to remote type 2.1 nodes for all the nodes of an APPN network. This view of traditional SNA networks offers an easy seam between APPN and subarea networks.

An example of APPN networks is shown in Figure 11 on page 27.

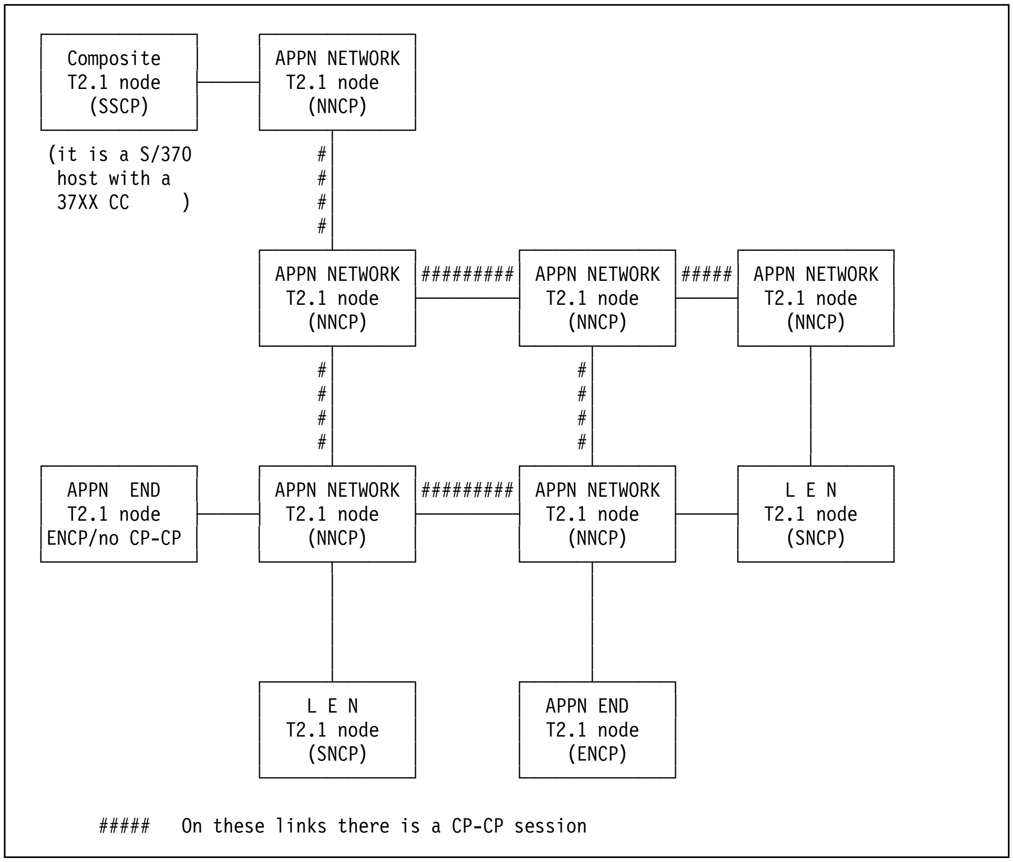


Figure 11. Example of APPN Network

Chapter 2. A Short Introduction to TCP/IP

Since many excellent publications have been written on the topic of TCP/IP and the Internet, we aim to provide only a short chapter for those readers who may not be familiar with the topic or who may desire a quick refresh.

For more detailed or advanced knowledge of the topic, readers are referred to the following publications:

- *Internetworking with TCP/IP, Volume I, Principles, Protocols and Architecture* second edition, Prentice-Hall, Inc., 1991, by D. Comer.

  This popular volume is one of the most complete summaries of TCP/IP in print today.

- *TCP/IP Tutorial and Technical Overview*, fourth edition, IBM Corp., 1993, GG24-3376.

  This similarly well-regarded publication provides a concise introduction to TCP/IP and an overview of the implementations of the IBM TCP/IP products in heterogeneous networks.

- *Recent Internet Books*, RFC 1432, March 1993, by J. Quarterman.

  This article provides a list of books related to using the Internet.

- *The Request For Comments (RFCs)*

  There are over 1000 RFCs today. For those readers who want to keep up-to-date with the latest advances and research activities in TCP/IP, the ever-increasing number of RFCs is the best source of this information. (See 2.2, “Standards and Request for Comments (RFC)” on page 30 for an explanation of RFCs.)

2.1 What are TCP/IP and the Internet?

Transmission Control Protocol/Internet Protocol (TCP/IP) is the common name for a family of data communications protocols used to organize computers and data communications equipments into computer networks.

Today, TCP/IP networks are interconnected together using the same protocol suite, known as the internet protocol suite, or the TCP/IP protocols. These networks are sometimes known as *internets*.

The global *Internet* is the largest collection of national backbone networks, government networks, university campus networks and research institution networks. All of them are ″tied″ together using the TCP/IP protocols, so that users of any of the networks can use the network services provided by TCP/IP to reach users on any of the other networks.

This all began in the early 1970s when a group of researchers in the U.S. came up with the new concept of *internetworking*. For more information on the history of TCP/IP or the Internet, the reader is referred to the aforementioned publications. We will only note that the Internet is growing more rapidly than ever before.

2.2 Standards and Request for Comments (RFC)

We mentioned in the last section that the Internet is a large internetwork. One would immediately come up with some questions. For example, where are the TCP/IP standards? Who sets them? Who assigns the network addresses? Who manages the Internet?

The Internet Activities Board (IAB) is the nonprofit, coordinating committee for Internet design, engineering and management. The IAB members are committed to making the Internet function effectively and evolve to meet a large-scale, high-speed future. The IAB sets the Internet standards and manages the *Request for Comments (RFC)* publication process.

RFC is the mechanism through which the Internet protocol suite has been evolving. For example, an Internet protocol can have one of five *states*: standard, draft standard, proposed standard, experimental, informational and historic. In addition, an Internet protocol has one of five *statuses*: required, recommended, elective, limited use and not recommended. By communicating using the RFC, new protocols are being designed and implemented by researchers from both academic institutions and commercial corporations. At the same time, some old protocols are being superseded by new ones.

The task of coordinating the assignment of values to the parameters of protocols is delegated by the Internet Architecture Board (IAB) to the Internet Assigned Numbers Authority (IANA). These protocol parameters include op-codes, type fields, terminal types, system names, object identifiers, and so on. The ″Assigned Numbers″ RFC, currently RFC 1340, documents these protocol parameters.

2.3 TCP/IP Architecture and Protocols

TCP/IP has four functional layers: applications, transport, internetwork and network interface.

We will summarize them in the following sections:

- 2.3.6, “TCP/IP Application Protocols” on page 45 and 2.3.5, “Ports and Sockets” on page 41
- 2.3.3, “UDP - User Datagram Protocol” on page 39 and 2.3.4, “TCP - Transmission Control Protocol” on page 40
- 2.3.1, “IP - Internet Protocol” on page 31 and 2.3.2, “ICMP - Internet Control Message Protocol” on page 38

A layering model of TCP/IP is shown in Figure 12 on page 31.

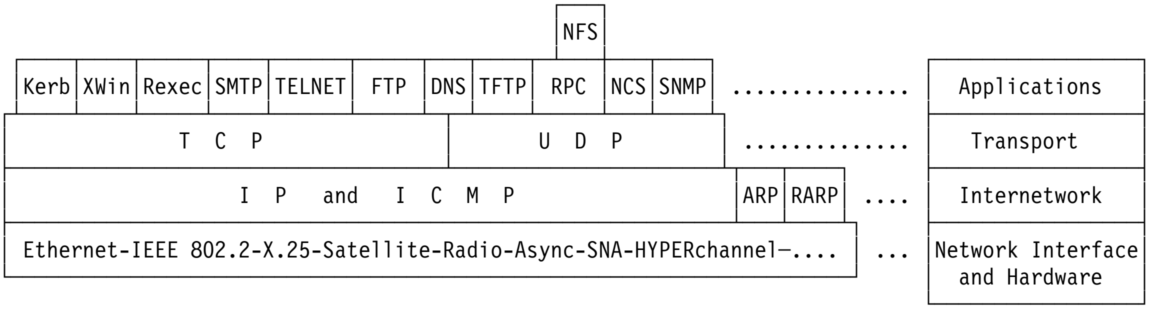


Figure 12. TCP/IP - Architecture Model

2.3.1 IP - Internet Protocol

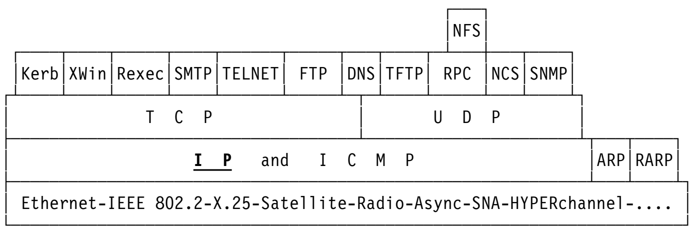


Figure 13. IP - Internet Protocol

IP is a *standard protocol* with a status of *required*.

IP is the layer that hides the underlying physical network from the upper-layer protocols. It is an unreliable, best-effort and connectionless packet delivery protocol. Note that best-effort means that the packets sent by IP may be lost, out of order, or even duplicated, but IP will not handle these situations. It is up to the higher-layer protocols to deal with these situations.

The following sections give a brief description of the following concepts related to IP:

- IP Addressing
- IP Datagram
- IP Routing
- IP Subnets

2.3.1.1 IP Addressing

IP uses *IP addresses* to specify source and target hosts on the internet (For example, we can contrast an IP address in TCP/IP with a fully qualified Netid.LUname in SNA). An IP address consists of 32-bits, usually represented in the form of four decimal numbers, one decimal number for each byte (or octet). For example:

```
00001001 01000011 00100110 00000001        a 32-bit address
    9        67       38        1          decimal notation (9.67.38.1)
```

An IP address consists of two logical parts: a network address and a host address. The network addresses are assigned in the Internet by a central authority, currently named the Network Information Center (NIC). An IP address belongs to one of four classes depending on the value of its first four bits. This is shown in Figure 14 on page 32.

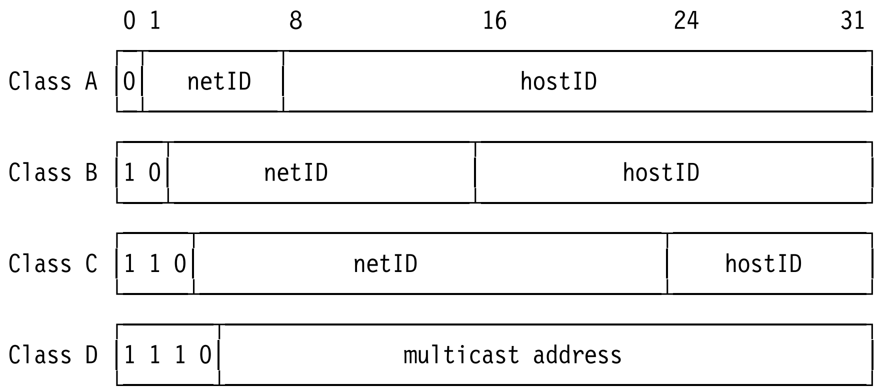


Figure 14. IP - Assigned Classes of IP Addresses

- Class A addresses provide for 128 (in fact 126) networks, each of which can have up to 2^{24} (16777219) hosts.
- Class B addresses allocate 14 bits for the physical networks, and 16 bits for hosts in each of those networks.
- Class C addresses allocate 21 bits for the network ID and each can have up to 256 hosts.
- Class D addresses are reserved for multicasting (a sort of broadcasting, but in a limited area, and only to hosts having this class D address).

Some values for these host IDs and network IDs are pre-assigned:

all bits 0 Stands for ″this″: ″this″ host (IP address with <host address>=0) or ″this″ network (IP address with <network address>=0). When a host wants to communicate over a network but does not yet know the network IP address, it may send packets with <network address>=0. Other hosts on the network will interpret the address as meaning *this network*. Their reply will contain the fully qualified network address, which the sender will record for future use.

all bits 1 Stands for ″all″: ″all″ networks or ″all″ hosts. For example:

```
128.2.255.255
means ″all″ hosts on network 128.2 (class B address).
```

This is called a *directed broadcast address* because it contains both a valid <network address> and a broadcast <host address>.

2.3.1.2 IP Datagram

The unit of transfer of a data packet in TCP/IP is called an IP *datagram*. It is made up of a header containing information for IP and data that is only relevant to the higher level protocols. IP can handle *fragmentation* and *re-assembly* of IP datagrams. The maximum length of an IP datagram is 65,535 bytes (or octets). There is also a requirement for all TCP/IP hosts to support IP datagrams of size up to 576 bytes without fragmentation.

The IP datagram header is a minimum of 20 bytes long:

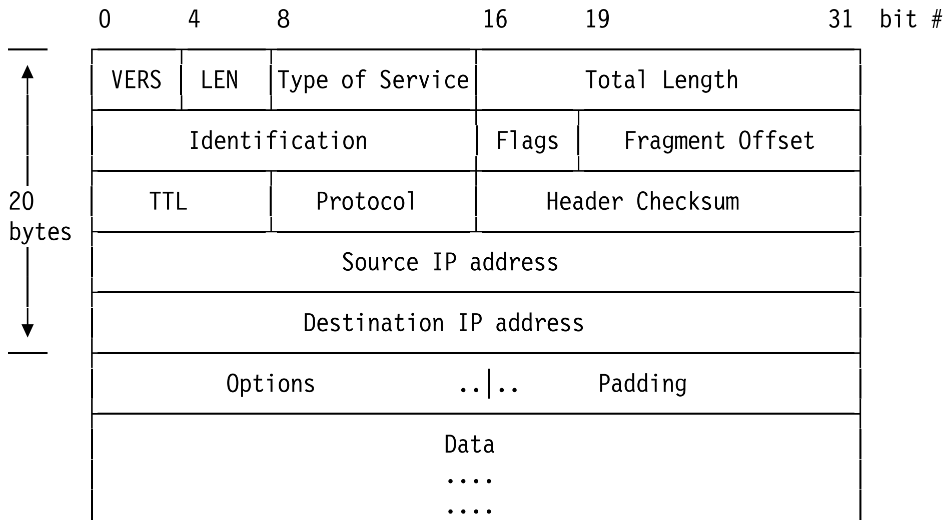


Figure 15. IP Datagram - Format of an IP Datagram Header

We will not elaborate on the format of the IP datagram header. You can find this information in the aforementioned publications.

2.3.1.3 IP Routing

There are two types of IP routing: direct and indirect.

Direct Routing: If the destination host is attached to a physical network to which the source host is also attached, an IP datagram can be sent directly, simply by encapsulating the IP datagram in the physical network frame. This is called *direct* delivery and is referred to as *direct routing*.

Indirect Routing: *Indirect routing* occurs when the destination host is not on a network directly attached to the source host. The only way to reach the destination is via one or more IP gateways (note that in TCP/IP terminology, the term gateway is used to qualify what is defined as a router). The address of the first of these gateways (the *first hop*) is called an *indirect route* in the context of the IP routing algorithm. The address of the first gateway is the only information needed by the source host.

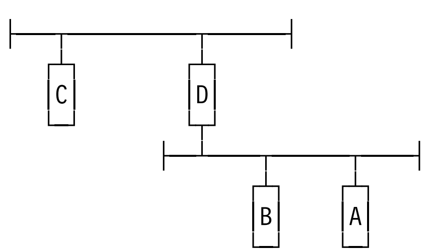


Figure 16. Direct and Indirect IP Routes. (Host A has a direct route to hosts B and D, and an indirect route to host C).

IP Routing Table: Each host keeps the set of mappings between the following:

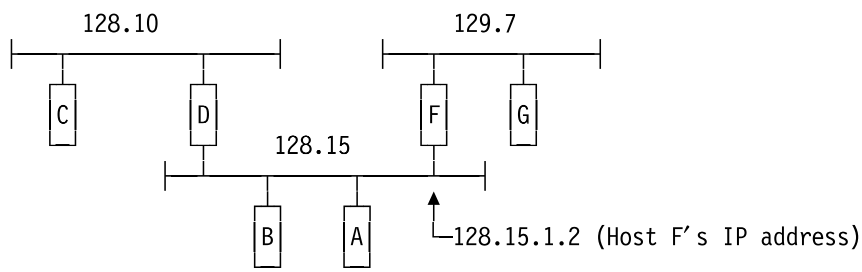
- Destination IP network address
- Route to next gateways

These are kept in a table called the *IP routing table*.

Three types of mappings can be found in this table:

1. The direct routes, for locally attached networks.
2. The indirect routes, for networks reachable via one or more gateways.
3. The default route, which contains the (direct or indirect) route to be used in case the destination IP network is not found in the mappings of type 1 and 2 above.

See the network in Figure 17 for an example configuration.



The routing table of host D will contain the following (symbolical) entries:

destination network address	deliver via route
128.10	direct attachment
128.15	direct attachment
129.7	128.15.1.2
default	128.15.1.2

Figure 17. Example of an IP Routing Table

IP Routing Algorithm: IP uses an algorithm to route an IP datagram. See Figure 18. It is called the *IP routing algorithm*.

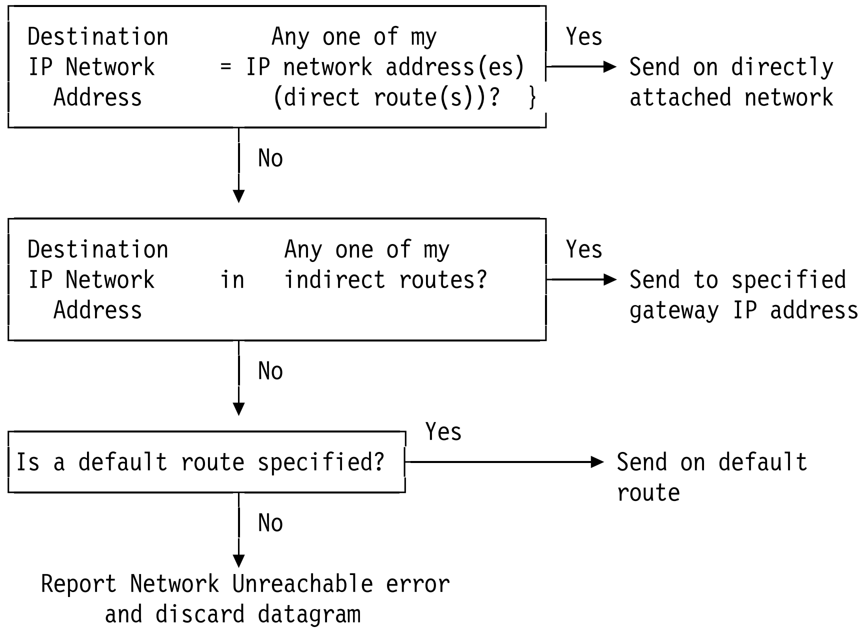


Figure 18. IP Routing Algorithm

Note that this is an iterative process. It is applied by every host handling a datagram, except for the host to which the datagram is finally delivered.

2.3.1.4 IP Subnets

Due to the explosive growth of the Internet, the principle of assigned IP addresses became too inflexible to allow easy changes to local network configurations. Those changes might have occurred when:

- A new type of physical network is installed at a location.
- Growth of the number of hosts requires splitting the local network into two or more separate networks.
- Growing distances require splitting a network into smaller networks, with gateways between them.

To avoid having to request additional IP network addresses in these cases, the concept of *subnets* was introduced.

The assignment of subnets can be done locally, because the whole network appears to be one IP network to the outside world.

Recall that an IP address consists of the pair:

<network address><host address>

```
For example,
     00001001 01000011 00100110 00000001        a 32-bit address
         9       67       38       1            decimal notation (9.67.38.1)

              9.67.38.1         is an IP address  (class A)  having

              9                 as the <network address>,
                67.38.1         as the <host address>.
```

Subnets are an extension of this: a part of the **<host address>** is a *subnetwork address*. IP addresses are then interpreted as:

<network address><subnetwork address><host address>

```
For example,

We may wish to choose the bits (from 8 to 25) of a Class A IP address to
indicate the subnet addresses, and the bits (from 26 to 31) to indicate
the "real" host addresses.
                                          1                   2                   3
         0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
        ┌─┬─────────────┬───────────────────────────────────┬───────────┐
Class A │0│ network addr│      SUBNET NUMBER                │ host addr │
        └─┴─────────────┴───────────────────────────────────┴───────────┘


We normally use a bit mask, known as the subnet mask,
to identify which bits of the original host address field to indicate
the subnet number.

In the above example,
 the subnet mask is 255.255.255.192 in decimal notation
(or 11111111 11111111 11111111 11000000 in bit notation)

Note: by convention, we mask the <network address> as well.
```

The division of the original <host address> part into <subnet> and <host> parts can be chosen freely by the local administrator. Except that the values of all zeroes and all ones in the <subnet> field are reserved for special addresses.

In the above example, we have divided the network into (2**18)-1-2 = 262,141 subnets, each of which has (2**6)-2 = 62 hosts. (Note: the subnet field occupies 18 bits, and we cannot use the bit strings of all zeroes or all ones. Also recall from 2.3.1.1, “IP Addressing” on page 31 that all host addresses with all zeroes and all ones are reserved).

Consider one of these 262,141 subnets; for example: 9.67.38.0 has 62 hosts with IP addresses ranging from 9.67.38.1 to 9.67.38.62.

In order to support subnets, the IP routing algorithm (in Figure 18 on page 35) needs to be updated as follows:

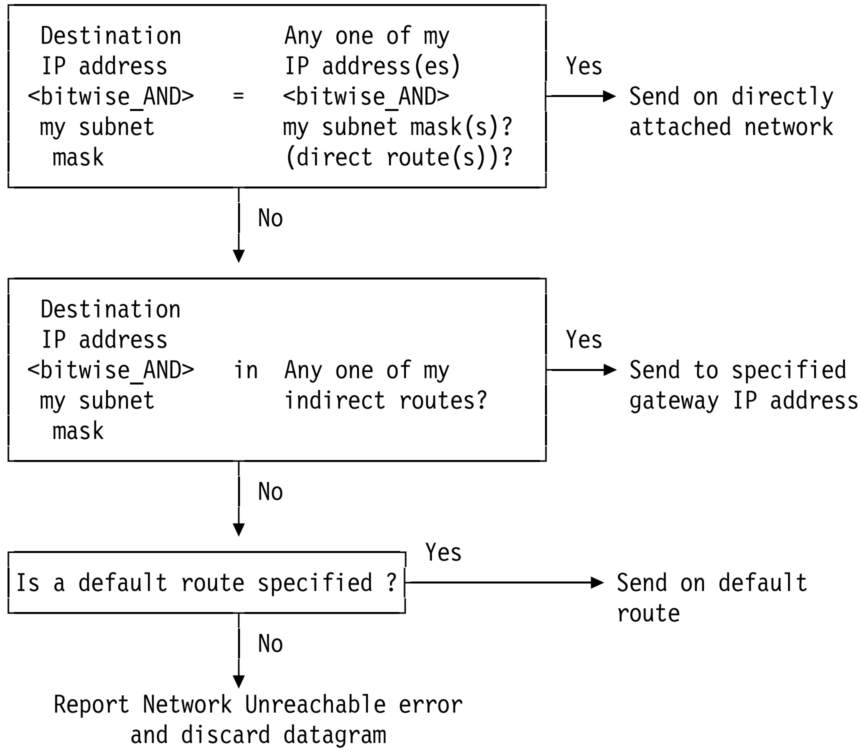


Figure 19. IP Routing Algorithm - with Subnets

2.3.2 ICMP - Internet Control Message Protocol

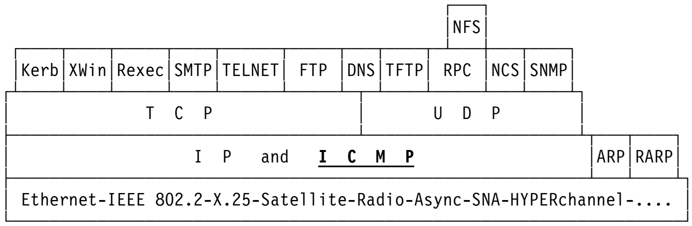


Figure 20. ICMP - Internet Control Message Protocol

ICMP is a *standard protocol*. Its status is *required*.

Although ICMP is shown in the graphic as being in the same protocol layer as IP, it is actually an integral part of IP. ICMP is occasionally used for reporting some errors in datagram delivery.

Perhaps one of the most useful commands available on all TCP/IP implementations is the PING (packet internet groper) application. PING uses ICMP to send an echo datagram to a specified IP address and wait for it to return. This is very useful for debugging purposes and also for knowing if a remote host can be reached from the local host.

2.3.3 UDP - User Datagram Protocol

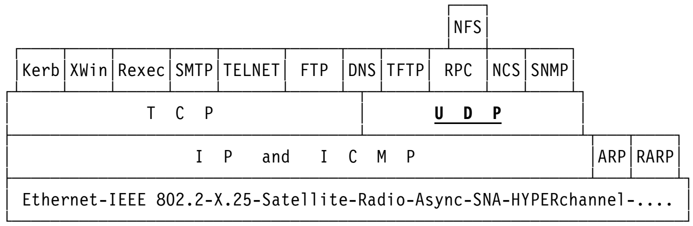


Figure 21. UDP - User Datagram Protocol

UDP is a *standard protocol*. Its status is *recommended*.

UDP is basically an application interface to IP. *It provides no additional reliability, flow-control or error recovery.* It simply serves as a multiplexer/demultiplexer for sending/receiving IP datagrams, using ports to direct the datagrams. (See 2.3.5, "Ports and Sockets" on page 41 for a discussion of ports.)

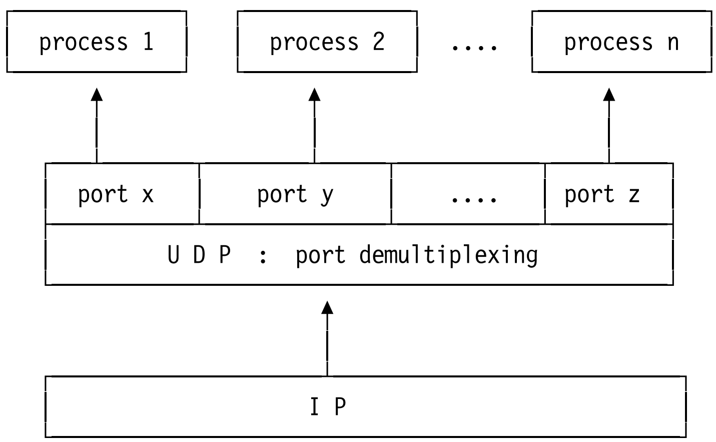


Figure 22. UDP-a Demultiplexer Based on Ports

2.3.4 TCP - Transmission Control Protocol

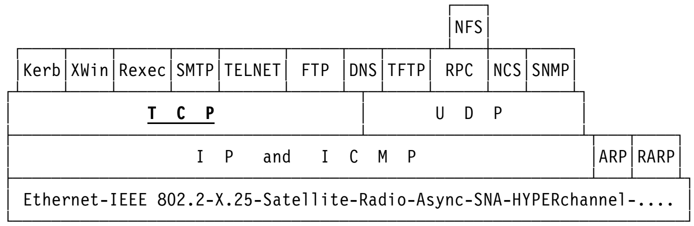


Figure 23. TCP -Transmission Control Protocol

TCP is a *standard protocol*. Its status is *recommended*.

We have previously discussed IP, the unreliable connectionless packet (datagram) system that forms the basis of the TCP/IP protocol suite. TCP is the higher-level protocol that provides reliability, flow control and some error recovery. Most of the TCP/IP application protocols, such as TELNET and FTP, use TCP as the underlying protocol.

TCP is a *connection-oriented*, end-to-end, reliable protocol providing *logical connections* between pairs of processes.

Within TCP, a *connection* is uniquely defined by a pair of sockets (that is, by a pair of processes, on the same or different systems, that are exchanging information). The concept of ports and sockets will be introduced in 2.3.5, “Ports and Sockets” on page 41.

The two processes communicate with each other over the TCP connection (InterProcess Communication - (IPC)). See Figure 24.

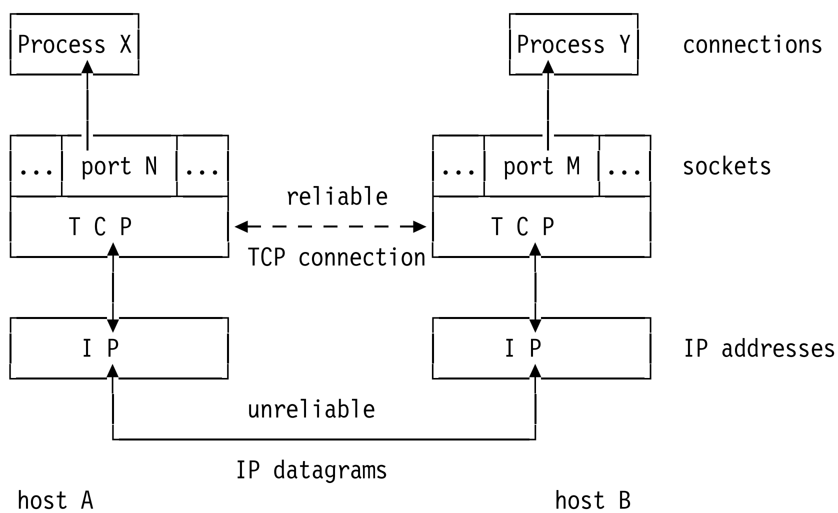


Figure 24. TCP Connection. (Processes X and Y communicate over a TCP connection carried by IP datagrams).

2.3.5 Ports and Sockets

Each *process* that wants to communicate with another process identifies itself to the TCP/IP protocol suite by one or more *ports*. A port is a 16-bit number, used by the host-to-host protocol to identify to which higher-level protocol or application program (process) it must deliver incoming messages.

Because some higher-level programs are themselves protocols, standardized in the TCP/IP protocol suite, such as TELNET and FTP, they use the same port number in all TCP/IP implementations. Those ″assigned″ port numbers are called *well-known ports* and the standard applications are called *well-known services*.

The well-known ports are controlled and assigned by the Internet Assigned Numbers Authority (IANA) and on most systems can only be used by system processes or by programs executed by privileged users. The assigned well-known ports occupy port numbers in the range 0 to 1,023. The ports with numbers in the range 1,024-65,535 are not controlled by the Internet central authority and on most systems can be used by ordinary user-developed programs.

Confusion due to two different applications trying to use the same port numbers on one host is avoided by writing those applications to request an available port from TCP/IP. Because this port number is dynamically assigned, it may differ from one invocation of an application to the next.

UDP, TCP and ISO TP-4 all use the same port principle. (See Figure 22 on page 39 and Figure 23 on page 40.) To the extent possible, the same port numbers are used for the same services on top of UDP, TCP and ISO TP-4.

Let us first consider the following terminologies:

- A *socket* is a special type of *file handle* used by a process to request network services from the operating system.
- A *socket address* is the triple:

  {*protocol, local-address, port number*}

  In the TCP/IP suite, for example:

  {*tcp, 193.44.234.3, 12345*}
- A *conversation* is the communication link between two processes.
- An *association* is the 5-tuple that completely specifies the two processes that constitute a connection:

  {*protocol, local-address, local-process, foreign-address, foreign-process*}

  In the TCP/IP suite, for example:

  {*tcp, 193.44.234.3, 1500, 193.44.234.5, 21*}

  could be a valid association.
- A *half-association* is either:

  {*protocol, local-address, local-process*}

  or

  {*protocol, foreign-address, foreign-process*}

  which specifies each half of a connection.

- The half-association is also called a *socket* or a *transport address.* That is, a socket is an end point for communication that can be named and addressed in a network.

The socket interface is one of several *application programming interfaces (APIs)* for the communication protocols. Designed to be a generic communication programming interface, it was first introduced by the 4.2BSD UNIX system. Although it has not been standardized, it has become a *de facto* industry standard.

4.2BSD allowed two different communication domains: Internet and UNIX. 4.3BSD has added the Xerox Network System (XNS) protocols, and 4.4BSD will add an extended interface to support the ISO OSI protocols.

2.3.5.1 Basic Socket Calls

The following lists some basic socket calls. In the next section we shall see an example scenario using these socket calls.

- Initialize a socket:

  FORMAT: int *sockfd* = **socket** (int *family*, int *type*, int *protocol*)

  where:

  – *family* stands for *addressing family*. It can take on values such as AF_UNIX, AF_INET, AF_NS and AF_IUCV. Its purpose is to specify the method of addressing used by the socket.
  – *type* stands for the type of socket interface to be used. It can take on values such as SOCK_STREAM, SOCK_DGRAM, SOCK_RAW, and SOCK_SEQPACKET.
  – *protocol* can be UDP, TCP or ICMP.
  – *sockfd* is an integer (similar to a file descriptor) returned by the socket call.

- Bind (register) a socket to a port address:

  FORMAT: int **bind** (int *sockfd*, struct sockaddr **localaddr*, int *addrlen*)

  where:

  – *sockfd* is the same integer returned by the socket call.
  – *localaddr* is the local address to be set before issuing the bind call.

  Note that after the bind call, we now have values for the first three parameters inside our 5-tuple association,

  {*protocol, local-address, local-process, foreign-address, foreign-process*}

- Indicate readiness to receive connections:

  FORMAT: int **listen** (int *sockfd*, int *queue-size*)

  where:

  – *sockfd* is the same integer returned by the socket call.
  – *queue-size* indicates the number of connection requests that can be queued by the system while the local process has not yet issued the accept call.

- Accept a connection:

  FORMAT: int **accept** (int *sockfd*, struct sockaddr **foreign-address*, int *addrlen*) where:

– *sockfd* is the same integer returned by the socket call.
– *foreign-address* is the address of the foreign (client) process returned by the accept call.

Note that this accept call is issued by a server process rather than a client process. If there is a connection request waiting on the queue for this socket connection, accept takes the first request on the queue and creates another socket with the same properties as *sockfd*; otherwise, accept will block the caller process until a connection request arrives.

- Request connection to the server:

  FORMAT: int **connect** (int *sockfd*, struct sockaddr **foreign-address*, int *addrlen*)

  where:

  – *sockfd* is the same integer returned by the socket call.
  – *foreign-address* is the address of the foreign (server) that the client wants to connect to.
  – Note that this call is issued by a client process rather than a server process.

- Send and/or Receive data:

  The **read**(), **readv**(), **recv**(), **recvfrom**(), **recvmsg**() with the associated calls **write**(), **writev**(), **send**(), **sendto**(), and **sendmsg**() can be used to receive and send data in an established socket association (or connection).

  Note that these calls are similar to the standard **read** and **write** file I/O system calls.

- Close a socket:

  FORMAT: int **close**(int *sockfd*)

  where:

  – *sockfd* is the same integer returned by the socket call.

2.3.5.2 An Example Scenario

As an example, let us consider the socket system calls for a connection-oriented protocol.

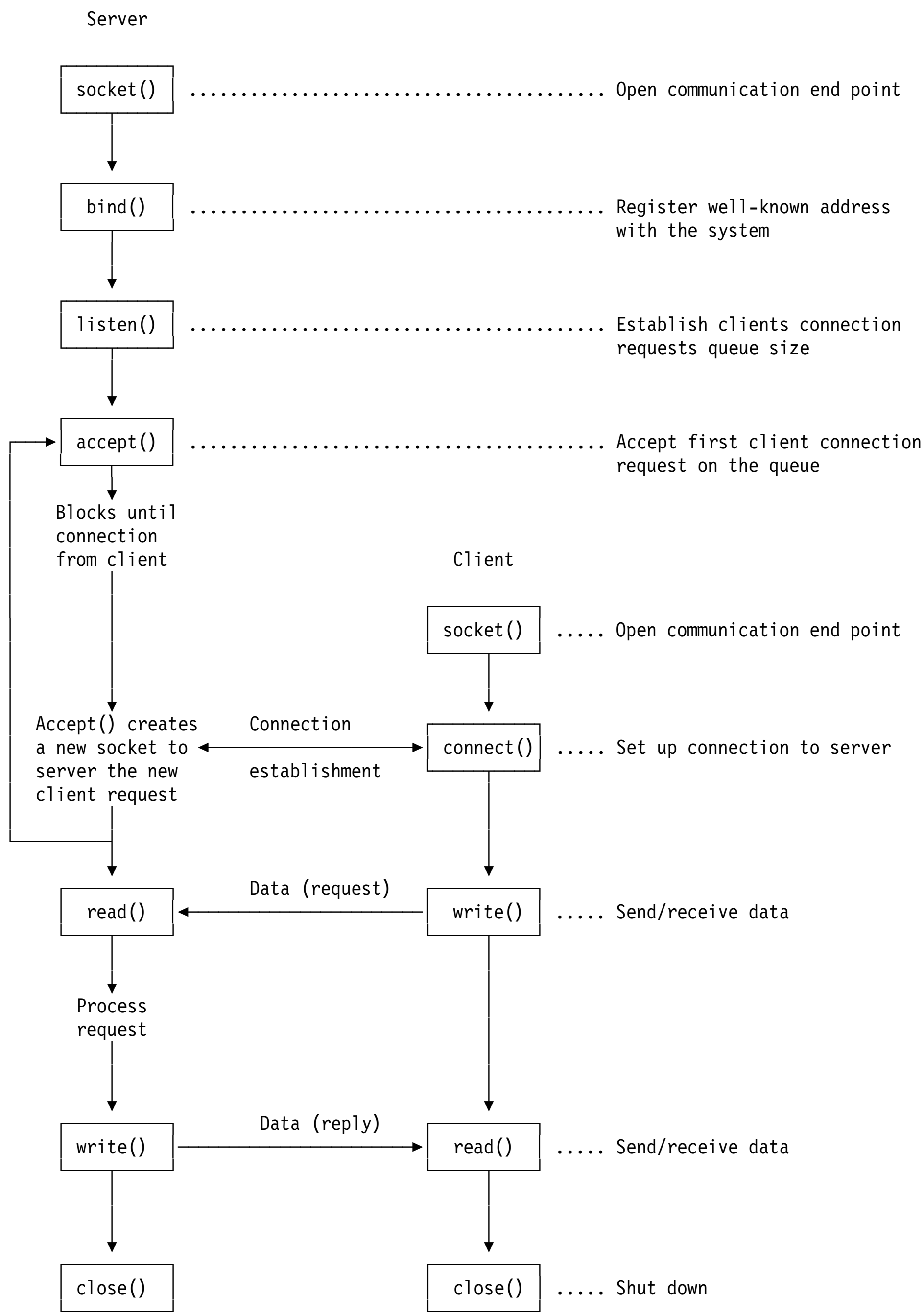


Figure 25. Socket System Calls for Connection-Oriented Protocol

Let us consider the previous socket system calls in terms of specifying the elements of the association:

	Protocol	Local Address	Local Process	Foreign Address	Foreign Process
Connection-oriented server	socket()	bind()		listen() accept()	
Connection-oriented client	socket()	connect()			
Connectionless server	socket()	bind()		recvfrom()	
Connectionless client	socket()	bind()		sendto()	

Figure 26. Socket System Calls and Association

The socket interface is differentiated by the different services that are provided. Stream, datagram, and raw sockets each define a different service available to applications:

- **Stream socket interface** (SOCK_STREAM): It defines a reliable connection-oriented service (over TCP, for example). Data is sent without errors or duplication and is received in the same order as it is sent. Flow control is built-in to avoid data overruns. No boundaries are imposed on the exchanged data, which is considered to be a stream of bytes. An example of an application that uses stream sockets is the file transfer program (FTP).
- **Datagram socket interface** (SOCK_DGRAM): It defines a connectionless service (over UDP, for example). Datagrams are sent as independent packets. The service provides no guarantees; data can be lost or duplicated, and datagrams can arrive out of order. An example of an application that uses datagram sockets is the Network File System (NFS**).
- **Raw socket interface** (SOCK_RAW): It allows direct access to lower-layer protocols such as IP and ICMP. This interface is often used for testing new protocol implementations. An example of an application that uses raw sockets is the PING command.

2.3.6 TCP/IP Application Protocols

One of the reasons for TCP/IP′s popularity is that there are many simple and useful standard applications available. We summarize several common TCP/IP applications in this section.

2.3.6.1 TELNET

```
                                                   ┌───┐
                                                   │NFS│
 ┌────┬────┬─────┬────┬──────┬─────┬───┬────┬─────┼───┼───┬────┐
 │Kerb│XWin│Rexec│SMTP│TELNET│ FTP │DNS│TFTP│ RPC │NCS│SNMP│
┌┴────┴────┴─────┴────┴──────┴─────┼───┴────┴─────┴───┴────┴┐
│             T  C  P              │          U  D  P       │
├──────────────────────────────────┴────────────────────────┴┬───┬────┐
│                   I  P   and   I  C  M  P                  │ARP│RARP│
├────────────────────────────────────────────────────────────┴───┴────┴┐
│ Ethernet-IEEE 802.2-X.25-Satellite-Radio-Async-SNA-HYPERchannel-.... │
└──────────────────────────────────────────────────────────────────────┘
```

Figure 27. TELNET - Virtual Terminal Protocol

TELNET is a *standard protocol*. Its status is *recommended*.

TELNET (teletypewriter network) is the virtual terminal protocol in TCP/IP. It allows users of one host to log into a remote host and interact as normal terminal users of that host.

For readers who are familiar with SNA, we can relate TELNET in TCP/IP to the terminal emulators (3270 or 5250 types) in SNA. In fact, all of the IBM TCP/IP product implementations provide TELNET support of 3270 terminal emulation in addition to the many other terminal emulation protocols.

2.3.6.2 FTP - File Transfer Protocol

```
                                                   ┌───┐
                                                   │NFS│
 ┌────┬────┬─────┬────┬──────┬─────┬───┬────┬─────┼───┼───┬────┐
 │Kerb│XWin│Rexec│SMTP│TELNET│ FTP │DNS│TFTP│ RPC │NCS│SNMP│
┌┴────┴────┴─────┴────┴──────┴─────┼───┴────┴─────┴───┴────┴┐
│             T  C  P              │          U  D  P       │
├──────────────────────────────────┴────────────────────────┴┬───┬────┐
│                   I  P   and   I  C  M  P                  │ARP│RARP│
├────────────────────────────────────────────────────────────┴───┴────┴┐
│ Ethernet-IEEE 802.2-X.25-Satellite-Radio-Async-SNA-HYPERchannel-.... │
└──────────────────────────────────────────────────────────────────────┘
```

Figure 28. FTP - File Transfer Protocol

FTP is a *standard protocol*. Its status is *recommended*.

FTP provides the functions of transferring files between two TCP/IP hosts. Since FTP is built on the services of TCP in the transport layer, it provides a reliable and end-to-end connection during the file transfer operation. Security is provided by the normal userID and password authentication.

2.3.6.3 DNS (Domain Name System)

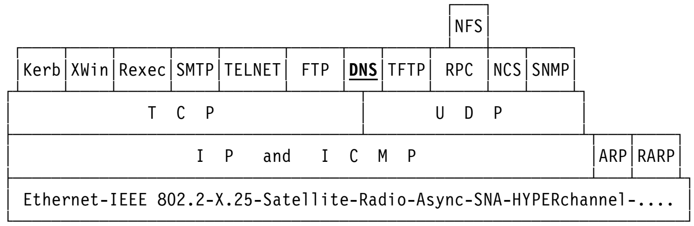


Figure 29. DNS - Domain Name System

The domain name system protocol is a *standard protocol*. Its status is *recommended*.

Recall that TCP/IP hosts are addressed by 32-bit IP addresses, which are represented in decimal notation. For example, to *TELNET* to a remote host with IP address of 9.67.38.1, the users would typically enter *TELNET 9.67.38.1*. This was both very cumbersome and error-prone.

Very quickly, this evolved to the use of symbolic high-level machine names. That is, instead of typing TELNET 9.67.38.1, one would now enter `TELNET small`, where `small` is then translated internally to the IP address 9.67.38.1.

This introduces the problem of maintaining the mappings between IP addresses and high-level machine names in a coordinated and centralized way.

Initially, host names to address mappings were maintained by the Network Information Center (NIC) in a single file (HOSTS.TXT) that was fetched by all hosts using FTP. Most hosts would have a copy of the **hosts** file, which may or may not be current or correct.

Implementation

This is called the *tcpip*.HOSTS.LOCAL file in IBM TCP/IP V2R2.1 for MVS.

Due to the explosive growth in the number of hosts, this mechanism became too cumbersome, time-consuming and error-prone, and was replaced by a new concept: the domain name system.

The domain concept lies in decentralizing the naming mechanism by distributing responsibility (and authority) for mapping between names and addresses. For example, consider the internal structure of a large organization. As the chief executive cannot do everything, the organization will probably be partitioned into divisions, each of them having autonomy within certain limits. Specifically, the executive in charge of a division has authority to make direct decisions, without permission from his chief executive.

Domain names are formed in a similar way, and will often reflect the hierarchical delegation of authority used to assign them. For example, consider the name

small.itsc.raleigh.ibm.com

Here, *itsc.raleigh.ibm.com* is the lowest-level domain name, a subdomain of *raleigh.ibm.com*, which again is a subdomain of *ibm.com*, a subdomain of *com*. We can also represent this naming concept by a hierarchical tree (see Figure 30 on page 48).

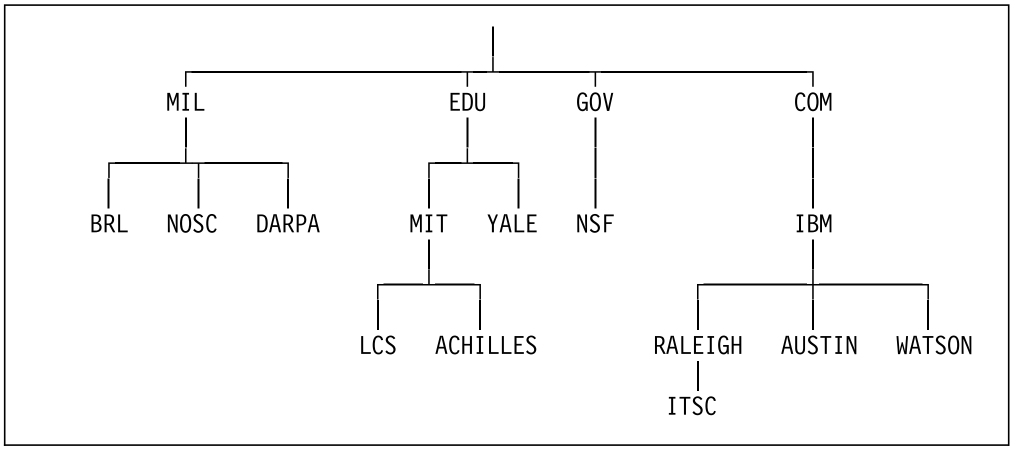


Figure 30. Hierarchical Namespace (Chain of Authority in Assigning Domain Names).

Table 1 shows some of the top-level domains of today's total Internet domain namespace.

Table 1. DNS - Some Top-Level Internet Domains

Domain Name	Meaning
edu	Educational institutions
gov	Government institutions
com	Commercial organizations
mil	Military groups
net	Major network support centers
country code	International standard 2-letter identifier

Mapping Domain Names to IP Addresses: The mapping of names to addresses consists of independent, cooperative systems called *name servers*. A name server is a server program answering requests from the client software, called a *name resolver*.

There are various types of implementations of the resolver functions and the name server functions (For example, full versus stub resolvers, and primary, secondary versus cache name servers). We will not elaborate on them.

Conceptually, all Internet domain servers are arranged in a tree structure that corresponds to the naming hierarchy in Figure 30. Each leaf represents a name server that handles names for a single subdomain. Links in the conceptual tree do not indicate physical connections. Instead, they show which other name server a given server can contact.

Figure 31 on page 49 shows the domain name resolution process:

- A user program issues a request such as the *gethostbyname* sockets call. (This particular call is used to ask for the IP address of a host by passing the host name).
- The resolver formulates a query to the name server.

- The name server checks to see if the answer is in its local authoritative database or cache, and if so, returns it to the client. Otherwise, it will query the other available name server(s).

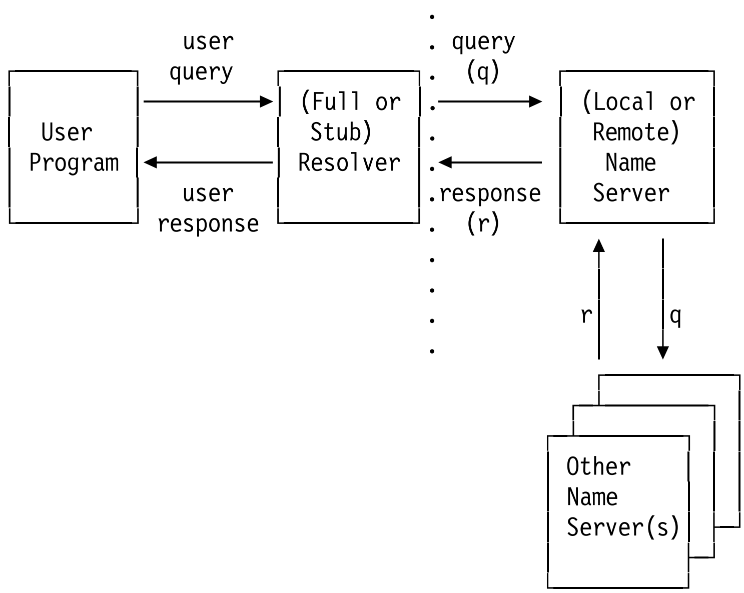


Figure 31. DNS - Resolver and Domain Name Server

Transport: The query/reply messages are transported by either UDP or TCP.

2.3.6.4 RPC - Remote Procedure Call

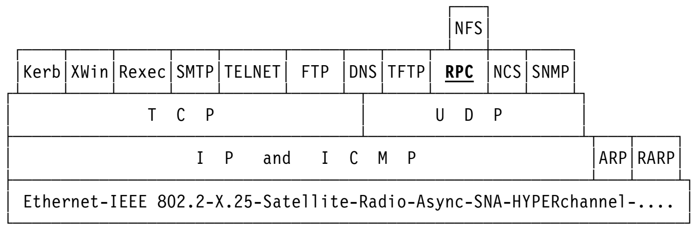


Figure 32. RPC - Remote Procedure Call

Sun-RPC is a *proposed standard protocol*. Its status is *elective*.

Remote Procedure Call is a standard developed by SUN Microsystems** and used by many vendors of UNIX systems.

RPC is an application programming interface (API) available for developing distributed applications. It allows programs to call subroutines that are executed at a remote system. The caller program (called *client*) sends a *call message* to the *server process*, and waits for a *reply message*. The call message includes the procedure's parameters and the reply message contains the procedure's results.

Sun-RPC consists of the following parts:

- *RPCGEN*: a compiler that takes the definition of a remote procedure interface and generates the client stubs and the server stubs.
- *External Data Representation (XDR)*: a standard way of encoding data in a portable fashion between different systems. It imposes a big-endian byte ordering and the minimum size of any field is 32 bits. This means that both the client and the server have to perform some translation.
- A run-time library.

The concept of RPC is very similar to that of an application program issuing a procedure call:

- The caller process sends a call message and waits for the reply.
- On the server side, a process is dormant awaiting the arrival of call messages. When one arrives, the server process extracts the procedure parameters, computes the results and sends them back in a reply message.

See Figure 33 on page 51 for a conceptual model of RPC.

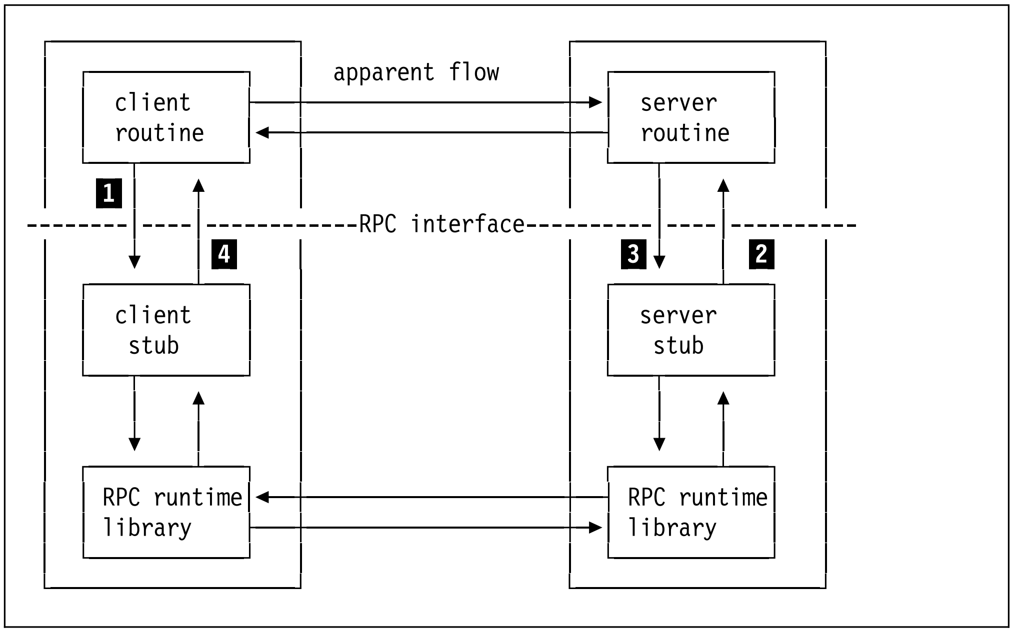


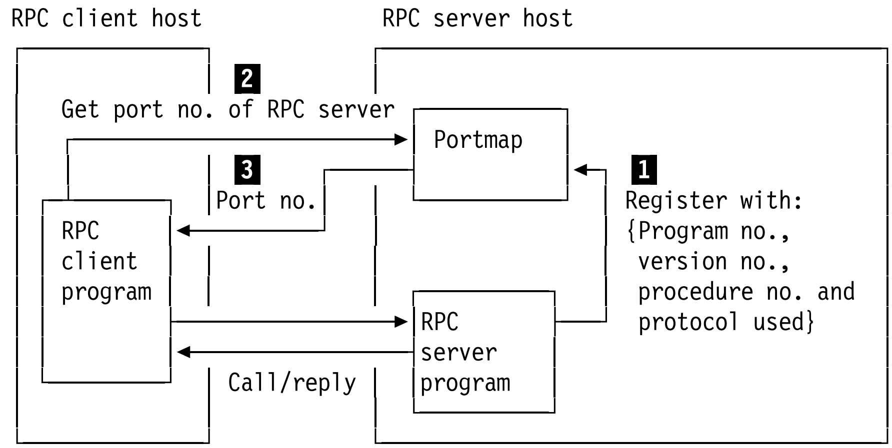
Figure 33. RPC - Remote Procedure Call Model

The RPC call message consists of several fields. For example,

- Remote program number
- Remote program version number
- Remote procedure number

Portmap: Figure 34 on page 52 shows the flow of an RPC call.

The Portmap or Portmapper** is a server application that will map a program number and its version number to the Internet port number used by the program. Portmap is assigned a reserved (well-known service) port number *111*.



Portmap only knows about RPC programs on the host it runs on. In order for Portmap to know about the RPC program, every RPC program should register itself with the local Portmap when it starts up (step 1).

The RPC client (caller) has to ask the Portmap service on the remote host about the port used by the desired server program (steps 2 and 3).

Figure 34. Portmap (Informs the Caller Which Port Number a Program on Its Host Occupies.)

2.3.6.5 NFS - Network File System

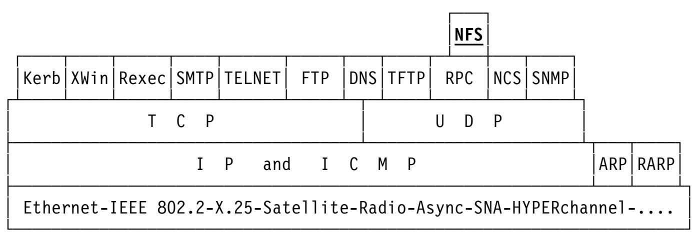


Figure 35. NFS - Network File System

The Network File System (NFS) protocol enables machines to share file systems across a network. It allows authorized users to access files located on remote systems as if they were local. It is designed to be machine-independent, operating system-independent, and transport protocol-independent. This is achieved through implementation on top of Remote Procedure Call (see 2.3.6.4, "RPC - Remote Procedure Call" on page 50).

NFS is a *proposed standard protocol*. Its status is *elective*.

2.3.6.6 X-Window System

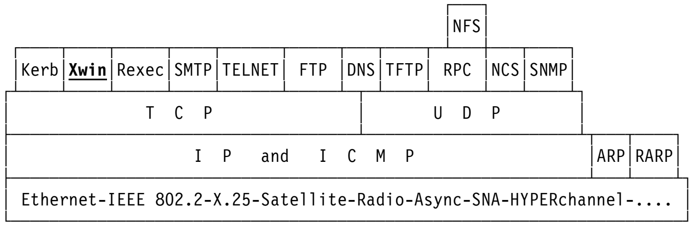


Figure 36. X-Window System

The X-Window System** (hereafter referred to as X) is one of the most widely used *graphical user interface (GUI)* or bitmapped-window display systems.

Current X releases contain two numbers: the *version number*, indicating major protocol or standards revisions, and a *release number*, indicating minor changes. At the time of this book′s publication, the latest version is X11 Release 5, also known as X11R5.

There are two parts which communicate with each other:

- **X-Server**: It is a dedicated program that provides display services on a graphic terminal, on behalf of a user, at the request of the user′s X-client program. It controls the screen and handles the keyboard and the mouse (or other input devices) for one or more X-clients. Equally, it is responsible for output to the display, the mapping of colors, the loading of fonts and the keyboard mapping. Typically X-server programs run on high performance graphics PCs and workstations, as well as X terminals, which are designed to run only the X-server program.
- **X-Client**: It is the actual *application* and is designed to employ a graphical user interface to display its output. Typically, many X-clients compete for the service of one X-server per display per user. *Xterm* and *Xclock* are two examples of X-clients.

Note: The X Window System uses *sockets* to communicate over a TCP/IP network.

Implementation

IBM TCP/IP V2R2.1 for MVS provides X-Window System client support.

IBM TCP/IP V1.2.1 for OS/2 provides X-Window System server support.

2.3.6.7 SNMP - Simple Network Management Protocol

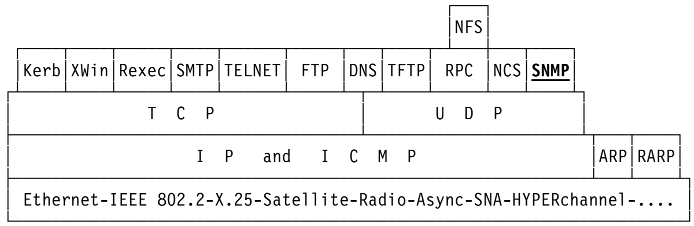
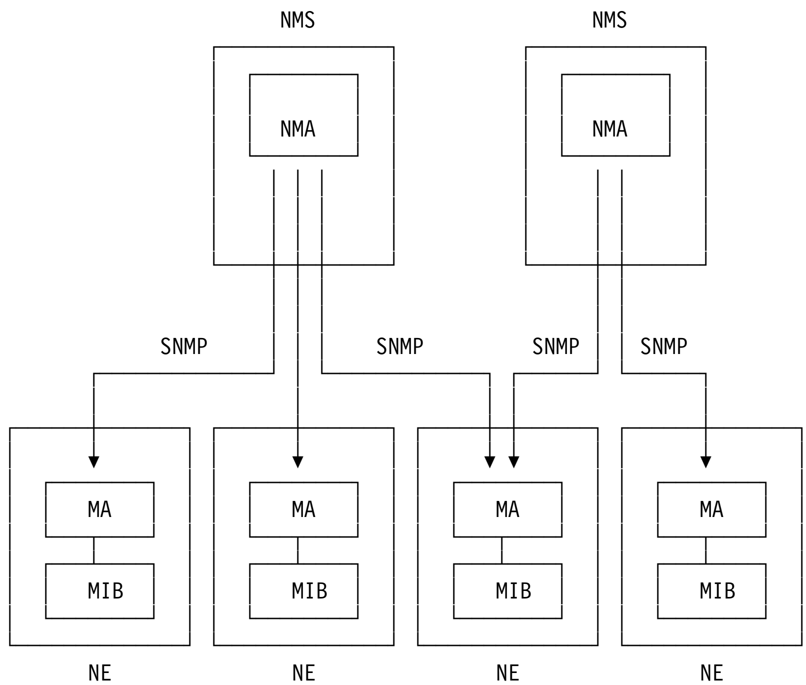


Figure 37. SNMP - Network Management

SNMP (*Simple Network Management Protocol*) is an Internet *standard protocol*. Its status is *recommended*.

With the growth in size and complexity of the TCP/IP-based internets, the need for network management became very important. The current network management framework for TCP/IP-based internets consists of the following:

1. Structure and Identification of Management Information (SMI) describes how managed objects contained in the MIB (Management Information Base) are defined.
2. Management Information Base, second version (MIB-II) describes the managed objects.
3. SNMP defines the protocol used to manage these objects.



NMS - Network Management Station
NMA - Network Management Application
NE - Network Element
MA - Management Agent
MIB - Management Information Base

Figure 38. SNMP - Components of the Simple Network Management Protocol

The network management station (NMS) executes network management applications (NMA) that monitor and control network elements (NE) such as hosts, gateways and terminal servers. These network elements use a management agent (MA) to perform the network management functions requested by the network management stations. The Simple Network Management Protocol (SNMP) is used to communicate management information between the network management stations and the agents in the network elements (See Figure 38).

Chapter 3. Multiprotocol Transport Networking (MPTN) Architecture

Today a communicating application tends to be written in the context of a particular transport service provider, called the native protocol.

For example, most UNIX or AIX (Advanced Interactive Executive) programs were originally written for Transmission Control Protocol/Internet Protocol (TCP/IP) transport using the sockets API. Similarly most SAA (Systems Application Architecture) programs were written for SNA (Systems Network Architecture) transport using CPI-C (Common Programming Interface for Communications) or other APPC (Advanced Program-to-Program Communication) interfaces.

In March 1993, IBM announced the new Multiprotocol Transport Networking (MPTN) architecture. MPTN is a description of the logical structure, formats, protocols, and set of operating principles that allows matching applications to use other networks in addition to the one which the application was written to use. MPTN does this with no changes to the existing applications.

Therefore, MPTN is a mechanism that allows the forced binding between the upper layer application support and the transport service provider to be eliminated. See Figure 39 for some MPTN terminology: Transport User and Transport Provider.

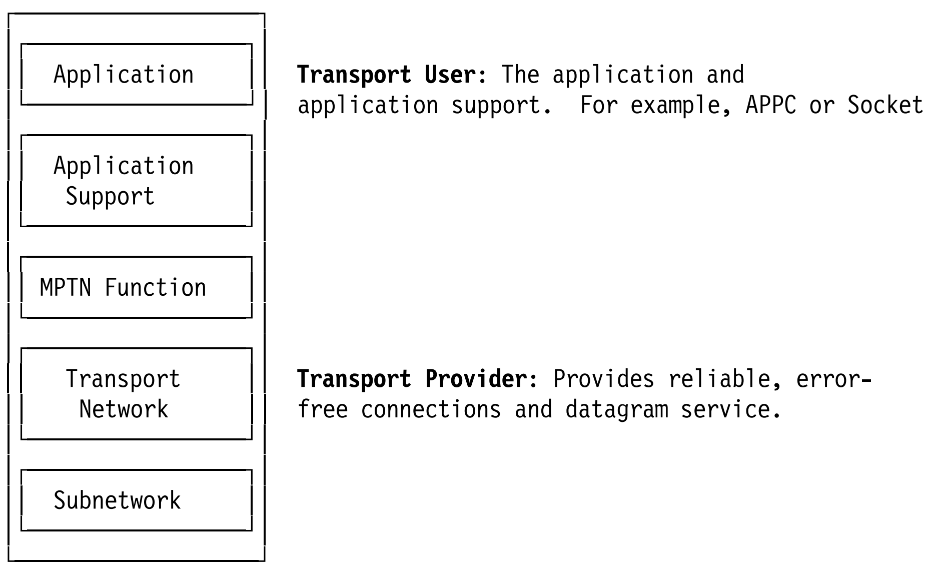


Figure 39. MPTN Terminology -Transport User and Transport Provider

For example, traditional support would allow two socket programs to communicate over a TCP/IP network. MPTN permits them to communicate also over a SNA network. See Figure 40 for a graphic description and also some MPTN terminology: native versus nonnative transport users. Similarly, MPTN also permits two APPC programs to communicate over a TCP/IP network in addition to SNA, their native transport service provider.

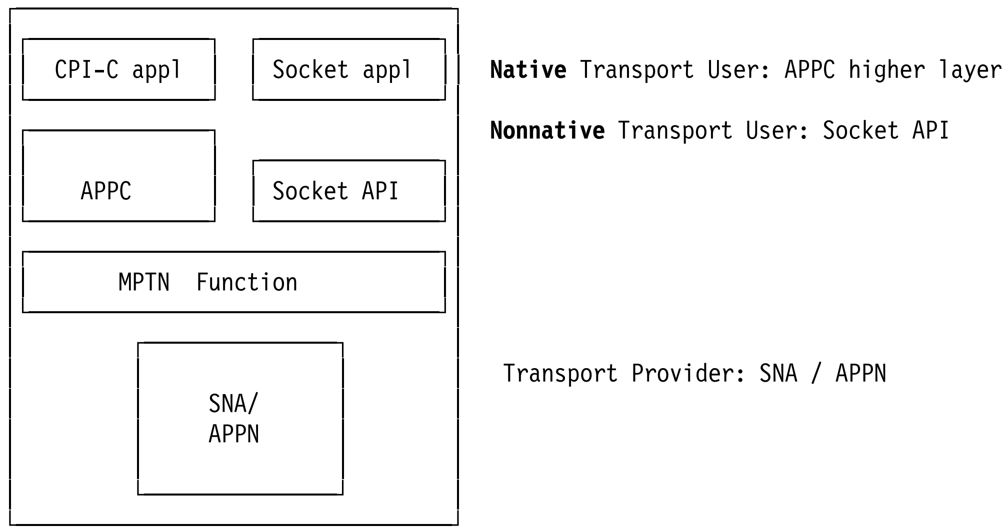


Figure 40. MPTN Terminology - Example of Two Transport Users and a Transport Provider in One MPTN Access Node

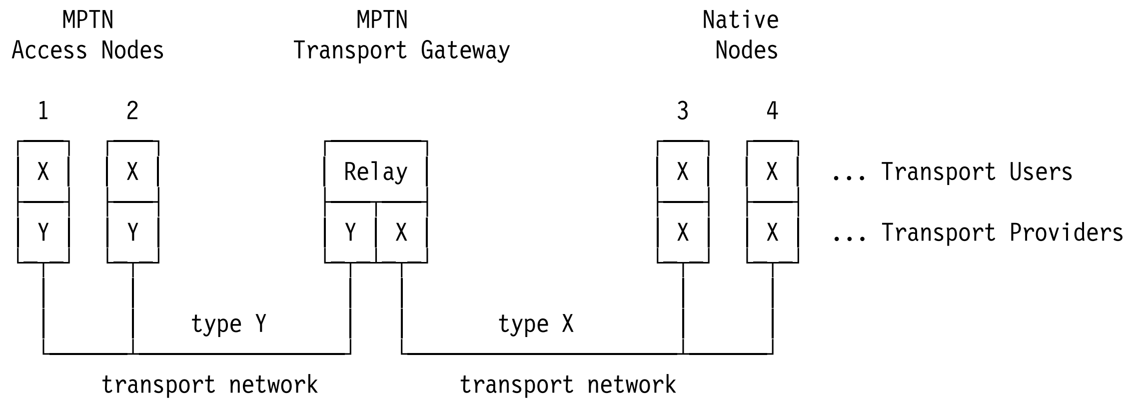
3.1 Objectives of MPTN

There are two general objectives of the MPTN architecture:

1. To break the binding between the upper-layer application support and the underlying transport service provider.

2. To provide for an end-to-end connection (or datagram service) that crosses more than one transport service provider. This is achieved by converting a transport connection (or datagram service) of one network to the corresponding transport connection (or datagram) of another network.

   See Figure 41 on page 59 for some MPTN terminology: MPTN *access node* and MPTN *transport gateway*.



The right-hand portion shows two nodes that have applications using their *native* network.

The left-hand portion shows two nodes that support the MPTN function called *MPTN access node*.

The *MPTN transport gateway* provides conversion between networking protocols X and Y, so that the X application in access nodes 1 and 2 can communicate with the matching X applications in the native nodes 3 and 4.

Figure 41. MPTN Terminology - Native Nodes, MPTN Access Nodes and MPTN Transport Gateway

3.2 Benefits of MPTN

MPTN decouples an application from the underlying transport network, thus providing additional degrees of freedom to both network users and network administrators:

- **Network users** can add a new application program based solely on how well it meets their business needs without requiring their network administrator to install a specific transport network. Off-the-shelf application programs can now be used in new environments.
- **Network planners** can choose the transport network best suited to their business needs without constraining their users′ choice of applications programs. Nonnative application programs can be added to existing networks.

  Furthermore, they have the flexibility to configure their networks so that different transport networks can be deployed as needed, while allowing users and matching application programs on different transport networks to communicate.
- **Network managers** will be better able to organize multiple transport networks into their desired transport network(s), while maintaining the investment in existing application programs.
- **Application designers** are free to develop new application programs on an application programming interface (API) based solely on the function of that API, and are not limited to an API that accompanies the installed transport network.

- **Network managers** can manage their overall network using the management tools native to the various transport networks.

3.3 Relationship of MPTN to the IBM Networking Blueprint

In March 1992, IBM presented its networking blueprint (see Figure 42).

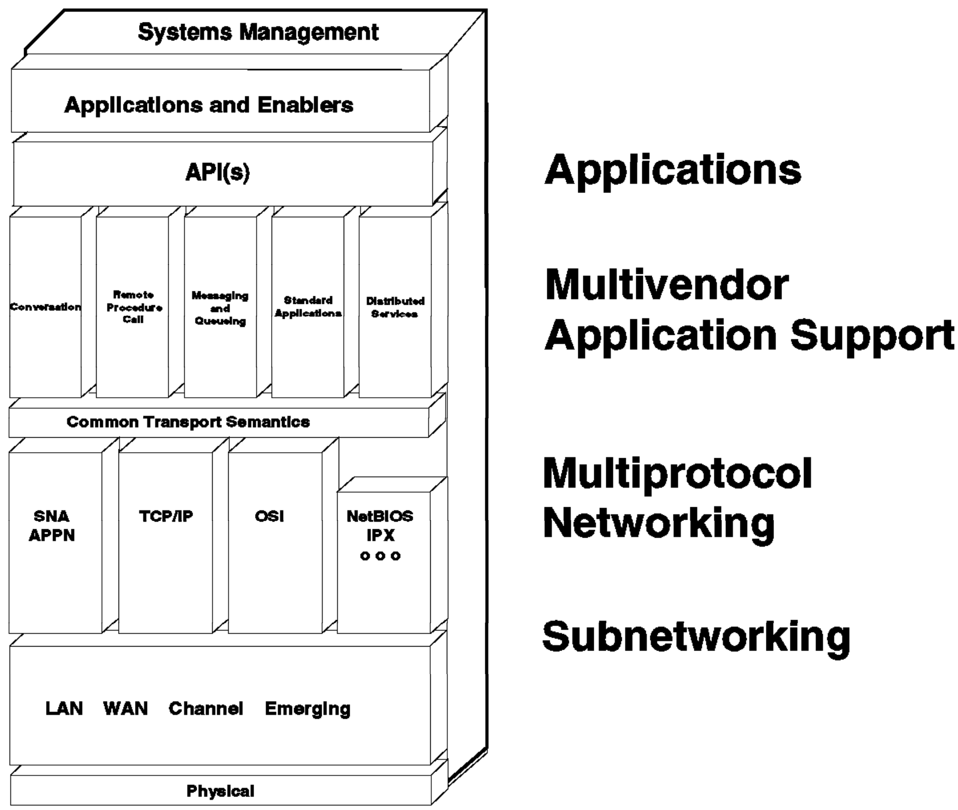


Figure 42. IBM Networking Blueprint

Common Transport Semantics (CTS) in the networking blueprint divides the protocol stacks at layer 4, the transport layer. The applications, APIs, and applications support layers are above the CTS, while the transport network is below CTS (See Figure 43 on page 61).

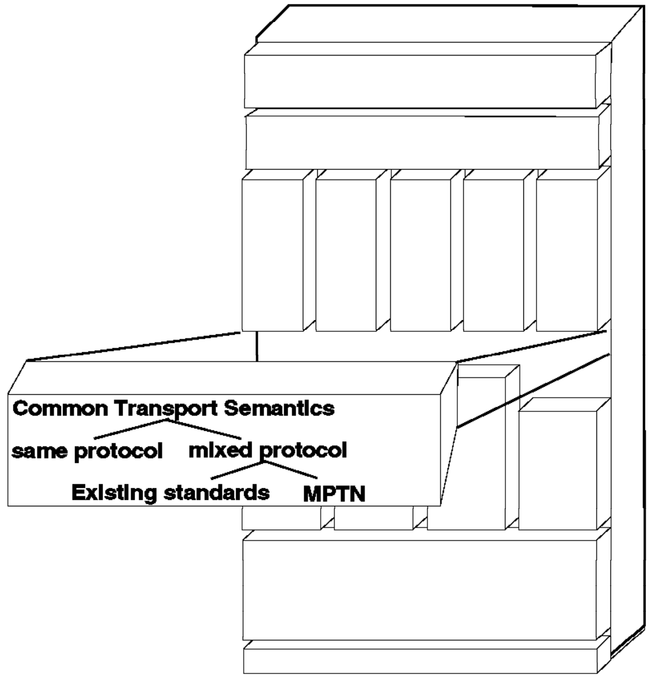


Figure 43. IBM Networking Blueprint - Common Transport Semantics (CTS) and MPTN

CTS includes all of the functions in the underlying transport providers. It delivers the functions in the following ways depending on the situation:

1. CTS encompasses the normal case where the native protocols of the application match those of the transport network.
2. CTS function can be delivered in specific situations by using accepted industry standards, such as RFC 1006 (for OSI over TCP/IP) or RFCs 1001 and 1002 (for NetBIOS over TCP/IP).
3. The formats and protocols of MPTN deliver CTS function in situations where the native protocol of the application does not match the transport network and standard solution does not exist. Compensation for missing functions is provided in those situations by MPTN.

Figure 44 on page 62 illustrates the three situations:

- The top of this graphic shows three applications representing the application layer of the blueprint.
- The bottom of this graphic shows two transport providers: SNA and TCP/IP, respectively.
- The middle box shows the CTS layer of the blueprint:
  1. CTS includes the traditional cases: APPC applications over SNA, and sockets applications over TCP/IP. In these two cases, no changes are required to the protocol flows.
  2. CTS delivers its function by using accepted industry standard. In this example, RFCs 1001 and 1002 for NetBIOS over TCP/IP are used.
  3. MPTN delivers CTS function in the two examples: APPC application over TCP/IP and sockets applications over SNA, respectively.

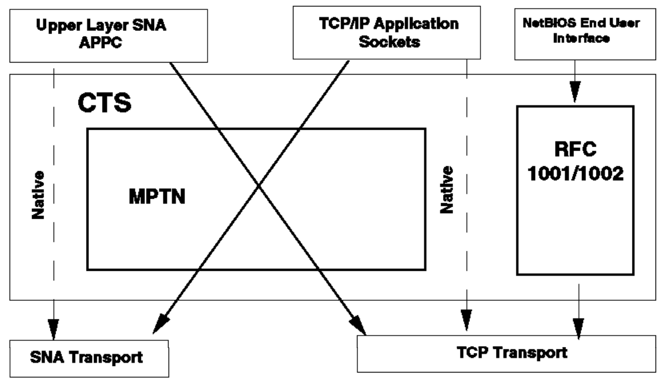


Figure 44. IBM Networking Blueprint - Common Transport Semantics (CTS) Examples

For more details on the IBM networking blueprint, please refer to *Networking Blueprint Executive Overview*.

3.4 Address Mapping in MPTN

Address mapping is required when the transport user (application) and the transport provider (network protocol) have different addressing schemes. For example, in Figure 45 on page 63, there are two MPTN access nodes that contain matching transport users (applications). The applications want to communicate over the matching nonnative transport protocol.

In order for the applications to communicate with each other using their existing transport address format, the MPTN components in both access nodes would need to provide the address mapping between the user (applications) and the provider.

For example, APPC applications use SNA fully qualified LU names to communicate with each other. Similarly, CPI-C applications use symbolic destination names that are equivalent to LU names. In both cases, if the transport provider is a TCP/IP transport network, MPTN needs to perform some address mapping.

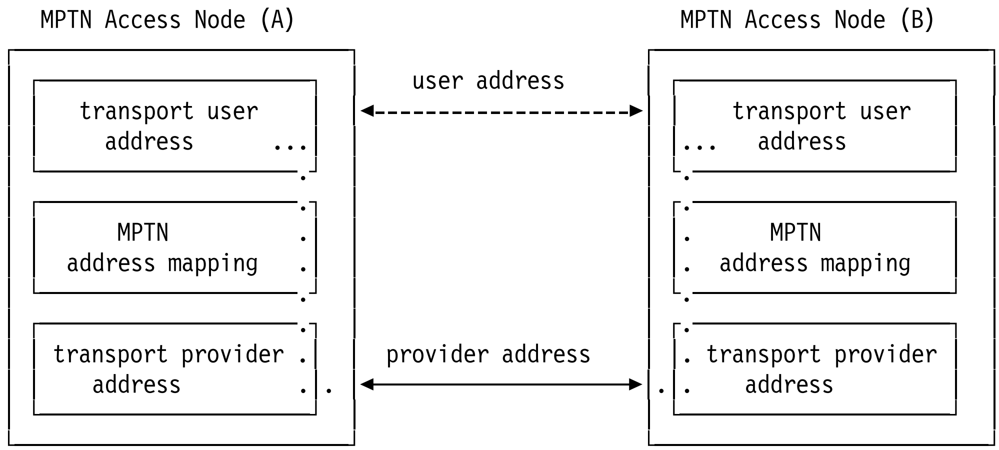


Figure 45. MPTN Address Mapping

In MPTN, there are three architected approaches to address mapping:

1. **Algorithmic**

   MPTN uses an algorithm to generate a transport provider address based on the transport user address. This approach is appropriate when the user's address space is smaller than the provider's address space and the transport provider allows more than one local address.

   Implementation of Sockets over SNA

   For example, this is the case for mapping from internet addresses to SNA LU names. We will see in Part 2 of this publication how this is implemented in Sockets over SNA.

2. **Extended protocol-specific directory**

   This involves extending a protocol-specific directory to handle transport addresses of various other formats. That is, all transport user addresses are registered with an "extended" protocol-specific directory. This approach is appropriate when the transport provider's directory supports the registration of different address types.

   Implementation of SNA over TCP/IP

   For example, we can register an SNA LU name in a TCP/IP domain name server. Note that the algorithmic approach is not feasible for mapping SNA LU names to TCP/IP internet addresses since the SNA address space (LU names) is larger than the TCP/IP address space (internet addresses) and one node in TCP/IP network can only have one IP address.

3. **Address mapper**

   This is basically a database that holds the transport user to transport provider mappings. This approach is the most general, but also the most costly. It is appropriate when the first two are not feasible.

3.5 Function Compensation in MPTN

Every transport provider lacks some functions supported by other transport providers. For example, SNA, NetBIOS and OSI all support a record model that is lacking in TCP/IP, while TCP/IP supports a stream model that is lacking in SNA, NetBIOS and OSI. In order to support multiple transport users over a common transport provider, MPTN provides function compensation when a transport user requests services that are not provided by the transport provider.

3.6 MPTN Network Management

Multiprotocol networks represent a set of heterogeneous networks for network administrators to manage. MPTN uses *existing* network management protocols in their native environments. For example:

- SNA Management Services (MS) for Alerts in SNA
- SNMP (Simple Network Management Protocol) in TCP/IP
- CMIP (Common Management Information Protocol) in OSI

In the future, MPTN network management will include providing a single user interface to manage not only the native environment, but also:

- The association between the transport users and the transport provider in the MPTN access node.
- The association between concatenated transport connections at each MPTN gateway.

3.7 Implementations of MPTN

The current implementations of the MPTN architecture include:

1. MVS/ESA and OS/2: SNA applications over TCP/IP and SNA over TCP/IP gateway
2. MVS/ESA and OS/2: Sockets applications over SNA
3. OS/2: NetBios applications over SNA
4. AIX: APPC applications over TCP/IP
5. AIX: Sockets applications over SNA
6. Windows: APPC applications over TCP/IP
7. Others

In Figure 46 on page 65, the two arrows depict APPC applications over TCP/IP and socket applications over SNA, respectively. The MPTN architecture delivers CTS function in both of these situations.

Both of these implementations became available as optional features of VTAM Version 3 Release 4.2 in April of 1993. The rest of this book explores these implementations by demonstrating some working scenarios, showing the systems definitions, configurations and test results, and finally, presenting some useful hints and tips.

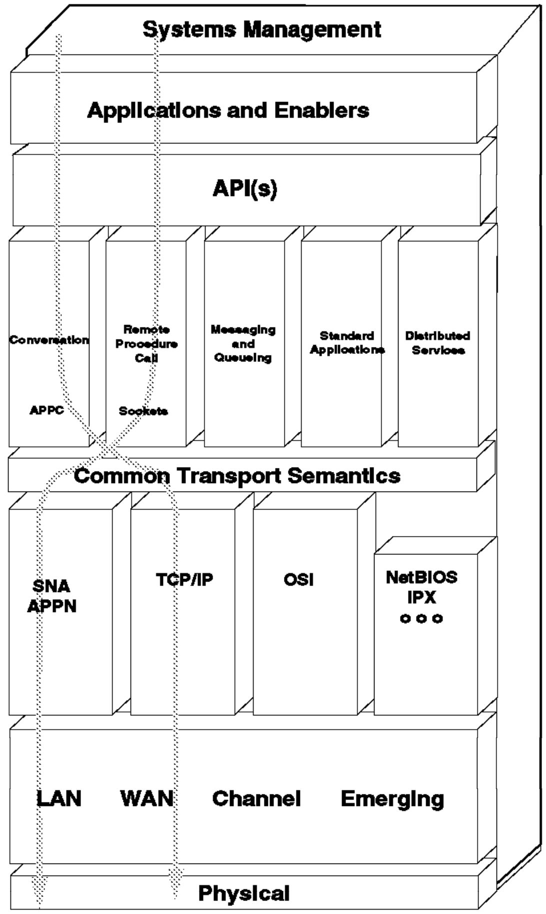


Figure 46. Implementations of MPTN - APPC over TCP/IP, and Sockets over SNA

For more detail on the MPTN architecture, please refer to the following publications:

- *Multiprotocol Transport Networking (MPTN) Architecture, Pocket Guide*
- *Multiprotocol Transport Networking: Technical Overview*

Chapter 4. AnyNet Product Family

The AnyNet product family enables application programs to communicate without change over different transport networks and across interconnected networks. Using the AnyNet functions, it is possible to reduce the number of transport networks, thus reducing operational complexity. These benefits are gained without modification to existing application programs or hardware.

The AnyNet product family was introduced as the VTAM* Multiprotocol Transport Feature (MPTF) in VTAM V3R4.2 for MVS/ESA. The VTAM MPTF has been renamed to the VTAM AnyNet Feature for V4R2 and includes the following functions:

- AnyNet/MVS and AnyNet/2 SNA over TCP/IP
- AnyNet/MVS and AnyNet/2 Sockets over SNA
- AnyNet/2 Sockets over SNA Gateway

The following AnyNet products are available on diskette for workstation use and can be ordered in the following combinations:

- AnyNet/2 Version 2.0

  SNA over TCP/IP

  Sockets over SNA

- AnyNet/2 Sockets over SNA Gateway Version 1.1
- AnyNet/2 NetBIOS over SNA Version 1.0

The following sections provide an overview of the products that are available as part of the AnyNet product family.

AnyNet/2 indicates that the function is designed for use in OS/2 environments. AnyNet/MVS indicates that the function is designed for use in MVS environments. Call your IBM* representative or the IBM branch office serving your locality for ordering information.

AnyNet/MVS and AnyNet/2 SNA over TCP/IP: These functions enable SNA application programs to communicate over a TCP/IP network. SNA over TCP/IP supports all LU types, not just LU 6.2, and supports concurrent sessions over the TCP/IP network to LUs in different SNA networks. SNA over TCP/IP provides support for dependent logical unit communications, such as printers and emulators, if the host is defined as a dependent LU server and the VTAM-provided OS/2 dependent LU requester support is enabled at the workstation. Finally, AnyNet/MVS provides an SNA over TCP/IP gateways that allows SNA applications located in an SNA network to communicate through the gateway to SNA application programs located in a TCP/IP network. For more information about AnyNet/MVS and AnyNet/2 SNA over TCP/IP, refer to *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP*. For information about just AnyNet/2 SNA over TCP/IP, refer to *IBM AnyNet/2 Version 2.0: Guide to SNA over TCP/IP*.

AnyNet/MVS and AnyNet/2 Sockets over SNA: These functions enable application programs that use the C socket API to communicate over SNA networks with other application programs that also use the C socket interface. Sockets over SNA now supports LU 6.2 full-duplex conversations and, for AnyNet/MVS, VTAM′s fully-qualified network name capability. For more information about AnyNet/MVS and AnyNet/2 Sockets over SNA, refer to *VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA*. For information about just AnyNet/2 Sockets over SNA, refer to *IBM AnyNet/2 Version 2.0: Guide to Sockets over SNA*.

AnyNet/2 Sockets over SNA Gateway: This OS/2-based function connects TCP/IP and SNA networks to enable communication between socket applications. Socket applications on existing TCP/IP networks can communicate, without change, with socket applications on SNA networks that have AnyNet/MVS or AnyNet/2 Sockets over SNA installed. In addition, AnyNet/2 Sockets Gateways can be used to connect TCP/IP networks across an SNA network, thereby allowing communication between socket applications running on the separate TCP/IP networks. For more information about Sockets Gateway, refer to *VTAM AnyNet Feature for V4R2: Guide to Sockets over SNA Gateway for OS/2* or *IBM AnyNet/2: Guide to Sockets over SNA Gateway Version 1.1*.

AnyNet/2 NetBIOS over SNA: This OS/2-based function enables NetBIOS applications to communicate over an SNA network with other NetBIOS applications. NetBEUI over SNA supports full-duplex conversations, non-blocking APPC operation, data compression, LU-LU session-level security and modem support. For more information about NetBEUI over SNA, refer to the *IBM AnyNet/2: NetBEUI over SNA Version 1.0 User's Guide* or the *IBM AnyNet/2: NetBEUI over SNA Version 1.0 Administrator's Guide*.

AnyNet/6000 APPC over TCP/IP and **AnyNet/6000 Sockets over SNA** These AIX-based functions are features of the **AIX SNA Server/6000 V2.1.1** product. AnyNet/6000 APPC over TCP/IP provides existing TCP/IP networks with the capability to add APPC or CPI-C applications without adding a separate SNA network. AnyNet/6000 Sockets over SNA allows existing SNA networks to add BSD Socket application support without having to add a separate TCP/IP network.

AnyNet/400 APPC over TCP/IP and **AnyNet/400 Sockets over SNA**: These AS/400-based functions are integrated parts of the OS/400 Version 3 Release 1 base operating system. AnyNet/400 APPC over TCP/IP allows any OS/400 APPC application written to ICF or CPI-C to communicate between AS/400 systems across a TCP/IP network. AnyNet/400 APPC over TCP/IP provides this with little or no change to application programs. AnyNet/400 APPC over TCP/IP is compatible with AnyNet/2 and AnyNet/MVS and thus provides connectivity to workstation and host environments. AnyNet/400 Sockets over SNA allows applications written to the sockets interface to communicate between AS/400 systems in an SNA environment. AnyNet/400 Sockets over SNA provides this with little or no change to application programs. AnyNet/400 Sockets over SNA is compatible with AnyNet/2 and AnyNet/MVS and thus provides connectivity to workstation and host environments.

AnyNet SNA over TCP/IP Gateway for OS/2: This OS/2-based product allows SNA applications such as CICS, DB2, emulators and printers to communicate over combined SNA and TCP/IP networks. Using paired AnyNet gateways to link SNA networks over existing TCP/IP networks provides significant savings over adding communication lines to link SNA networks.

AnyNet IPX over SNA Gateway for OS/2: This OS/2-based product allows NetWare**-based IPX ** applications to take advantage of the cost-effective bandwidth utilization, traffic prioritization and predictable response-time of SNA networks. AnyNet IPX over SNA Gateway enables IPX applications such as

NetWare Server, NetWare Requester or NetWare Management Services to run across an SNA network that connects IPX LANs.

After the time of the residency writing this publication the the following additional members of the AnyNet product family were available:

AnyNet APPC over TCP/IP for Windows

AnyNet/2 SNA over TCP/IP Gateway

Part 2. Platform Implementations

Chapter 5. SNA over TCP/IP Implementation

This chapter is an introduction to SNA over TCP/IP implementation for both platforms MVS/ESA and OS/2.

We will discuss both implementations in the following chapters

- *Chapter 6, "Running on MVS/ESA" on page 81*
- *Chapter 7, "AnyNet/2 SNA over TCP/IP" on page 93*

The AS/400 platform does not support SNA over TCP/IP but it does support APPC over TCP/IP. We will discuss OS/400 APPC over TCP/IP in following chapter

- *Chapter 8, "AnyNet/400 APPC over TCP/IP" on page 113*

Refer to Chapter 2, "A Short Introduction to TCP/IP" on page 29 for any reference to TCP/IP terms or functions and Chapter 3, "Multiprotocol Transport Networking (MPTN) Architecture" on page 57 for any reference to MPTN terms or functions.

5.1 What is SNA over TCP/IP?

SNA over TCP/IP is one of the two functions that make up the AnyNet/MVS feature and the AnyNet/2 program. Using SNA over TCP/IP, an application program designed to use the SNA application program interface (API) can now communicate with another SNA application program over a TCP/IP transport network by mapping an LU-LU *session* to a TCP *connection*.

Using MPTN terminology, a *Native* SNA *Transport User* is able to use the *Non-Native* TCP/IP *Transport Provider*.

5.2 System Software Required to Use SNA over TCP/IP (MVS/ESA)

The following are the minimum system software levels required to use SNA over TCP/IP on MVS/ESA.

- IBM MVS/ESA System Product (SP) Version 3 Release 1.3
- IBM MVS/ESA System Modification Program/Extended (SMPE) Release 5
- IBM TCP/IP Version 2 Release 2.1 for MVS
- IBM C for System/370 Version 2 at PUT level 9107
- VTAM Version 4 Release 2 base
- VTAM Version 4 Release 2 AnyNet host feature.

SNA nodes *not* configured with SNA over TCP/IP that communicate through intermediate nodes that have SNA over TCP/IP must have the VTAM Version 3 Release 2 program or higher, or a level of SNA software capable of communicating with VTAM Version 3 Release 2 or higher (refer to *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP*)

5.3 System Software Required to Use SNA over TCP/IP (OS/2)

The following are the minimum system software levels required to use SNA over TCP/IP on OS/2.

- IBM OS/2 Version 2.0 with Service Pak XR06055
- One of the following IBM Communications Manager/2 versions:
  - Version 1.11 Service Pak WR06150, or the country equivalent for national language versions, for dependent LU support
  - Version 1.0 for APPC support
- IBM TCP/IP Version 2 for OS/2
- VTAM Version 4 Release 2 AnyNet IWS feature or the AnyNet/2 Version 2.0 workstation product
- VTAM Version 4 Release 2 base (if the VTAM AnyNet IWS feature is used).

If you are using an earlier version of Communications Manager/2 than version 1.11, only APPC over TCP/IP can be installed and used. This means that the dependent LU support for SNA over TCP/IP is not available.

5.4 Attachments Supported by SNA over TCP/IP

Both AnyNet/MVS SNA over TCP/IP and AnyNet/2 SNA over TCP/IP use the installed TCP/IP application as base. Thus SNA over TCP/IP supports all attachments supported by the underlying TCP/IP program.

Refer to your IBM TCP/IP documentation to determine which attachments are supported (*IBM TCP/IP Version 2 for OS/2: Installation and Administration*).

5.5 Application Interfaces Supported by SNA over TCP/IP

SNA over TCP/IP supports application programs and subsystems that use the following application program interfaces (APIs)

- For AnyNet/MVS SNA over TCP/IP
  - VTAM APPCCMD
  - VTAM record application program interface (RAPI)
  - MVS/APPC including CPI-C (for example, IMS uses MVS/APPC and CPI-C). These APIs use LU 6.2 for networking support.
- For AnyNet/2 SNA over TCP/IP
  - Communications Manager APPC
  - Communications Manager CPI-C
  - Communications Manager 3270 emulator and LUA

5.6 TCP/IP Support Services Used by SNA over TCP/IP (MVS/ESA)

SNA over TCP/IP uses the following support services provided by IBM TCP/IP Version 2 Release 2.1 for MVS and IBM TCP/IP Version 2 for OS/2

- IUCV socket interface (IBM TCP/IP Version 2 Release 2.1 for MVS only)
- Socket API (for C language programs)
- TCP/IP protocol support
- Domain name system (DNS) resolver function or HOSTS file.

The domain name system (DNS) resolver is a function of the domain name system which sends request to the domain name server. The domain name server is a server program that supplies name-to-address translation by mapping the domain names to an IP address.

5.7 LU Session Types Supported by SNA over TCP/IP (MVS/ESA)

In addition to the dependent LU support provided by DLUS/R that is now available in AnyNet SNA over TCP/IP for MVS and OS/2, non LU 6.2 sessions can also be supported using AnyNet MVS with the following restrictions:

- Both communicating logical units must use AnyNet/MVS, either as an MPTN access node or as an MPTN gateway node, to go over the internet protocol (IP) network. LUs do not necessarily have to reside in the node where SNA over TCP/IP is installed.
- The primary LU (PLU) side must initiate the session in one of the following ways:
  - PLU requests to access the SLU (for example SIMLOGON)
  - VTAM initiates the session (for example LOGAPPL operand is coded)
  - An operator issues a VARY LOGON command.

Caution!

You cannot initiate the session by issuing a LOGON command or INITSELF from the secondary logical unit.

The restrictions listed above do not apply when communicating with logical units using SNA over TCP/IP for OS/2 and the DLUR/DLUS functions.

5.8 Dependent LU Requestor/Server

The dependent LU requestor and server functions facilitate conversion from subarea to APPN networks. These functions are also used by AnyNet SNA over TCP/IP to support dependent LU sessions across a TCP/IP network.

The dependent LU server (DLUS) function is a feature of an interchange node or a T5 network node node (VTAM).

The dependent LU requestor (DLUR) function the client side of the dependent LU server function. This may reside in the same node as the secondary LU it supports or in a node adjacent to and upstream from it.

Between the dependent LU server (DLUS) and the dependent LU requestor (DLUR), the SSCP-PU and SSCP-LU sessions that are required to support dependent LUs are encapsulated in an LU 6.2 session known as the CP_SVR pipe.

DLUS is used by AnyNet/MVS and DLUR is used by AnyNet/2 to support SNA dependent LU sessions over TCP/IP.

5.9 How SNA over TCP/IP Works (MVS/ESA)

Figure 47 shows that SNA over TCP/IP uses protocols that bypass the lower transport layers of SNA. Instead of encapsulating an entire SNA PIU within a transmission frame, SNA over TCP/IP builds a basic information unit (BIU) that TCP/IP can accept to send or receive data. The process of building the unique transmission frame is transparent to the SNA application program. This means that application programs do not need to be modified.

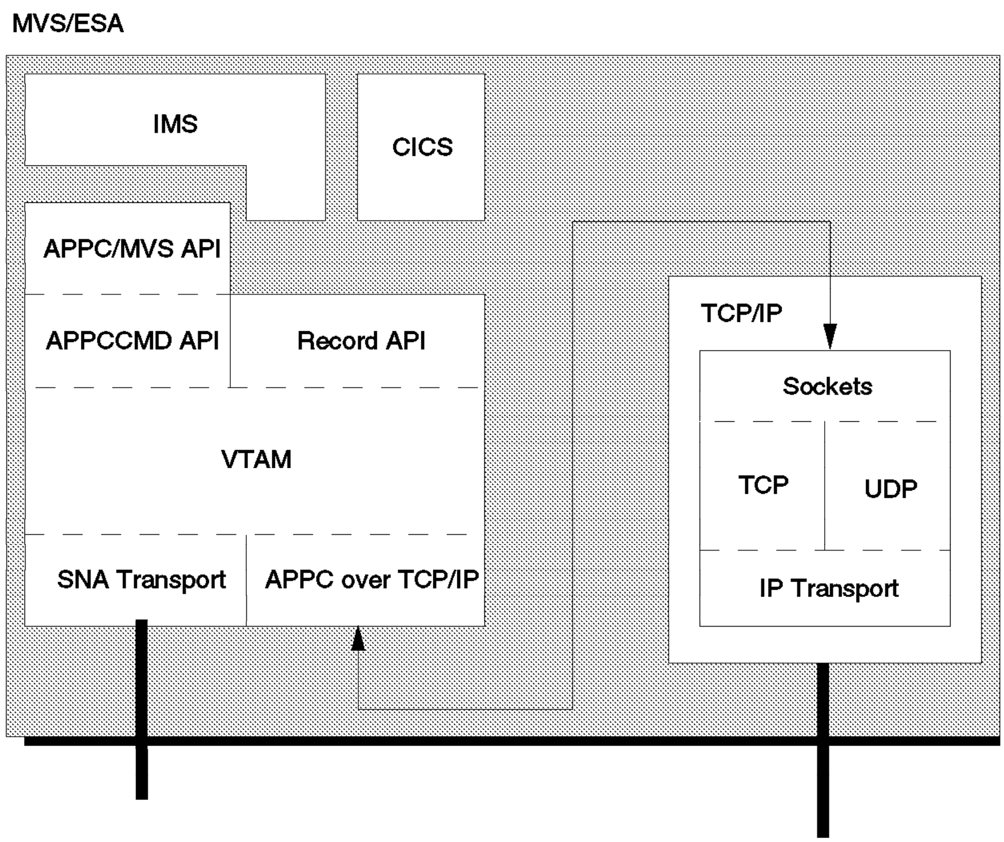


Figure 47. SNA over TCP/IP System Structure on MVS/ESA

When using SNA over TCP/IP, VTAM acts as a TCP/IP application. VTAM uses the IUCV socket interface to perform the following operations:

- Open a socket:
  - TCP (stream) protocol
  - UDP (user data) protocol

- Create a TCP connection to a remote system
- Send data over the socket (TCP or UDP)
- Receive data over the socket (TCP or UDP)
- Close the TCP connection
- Close the socket

VTAM sees the destination LU as an independent LU accessed through an adjacent link station PU. We have to define this ALS PU in a new VBUILD major node type=TCP. This major node is also called TCP/IP major node. The LINE and PU statements defined in this major node do not represent real communication lines or remote devices. Since VTAM only supports the use of one TCP/IP product instance at a time, only one LINE and PU can be active at a time. The configuration information you define about the destination LU enables SNA over TCP/IP to determine whether to route the data using SNA transport or to route using IP transport. For example, including the ALS PU name in an ALSLIST operand on the definition of the destination LU, allows VTAM to gain session connectivity over a TCP/IP network based on the order of the ALSLIST. When the TCP/IP ALS PU is activated, communication with TCP/IP is started. A TCP and UDP protocol port is designated for the exclusive use of SNA over TCP/IP.

5.10 How SNA over TCP/IP Works (OS/2)

Figure 48 on page 78 shows that SNA over TCP/IP uses protocols that bypass the lower transport layers of SNA architecture. The process of building the unique transmission frame is transparent to an SNA application program.

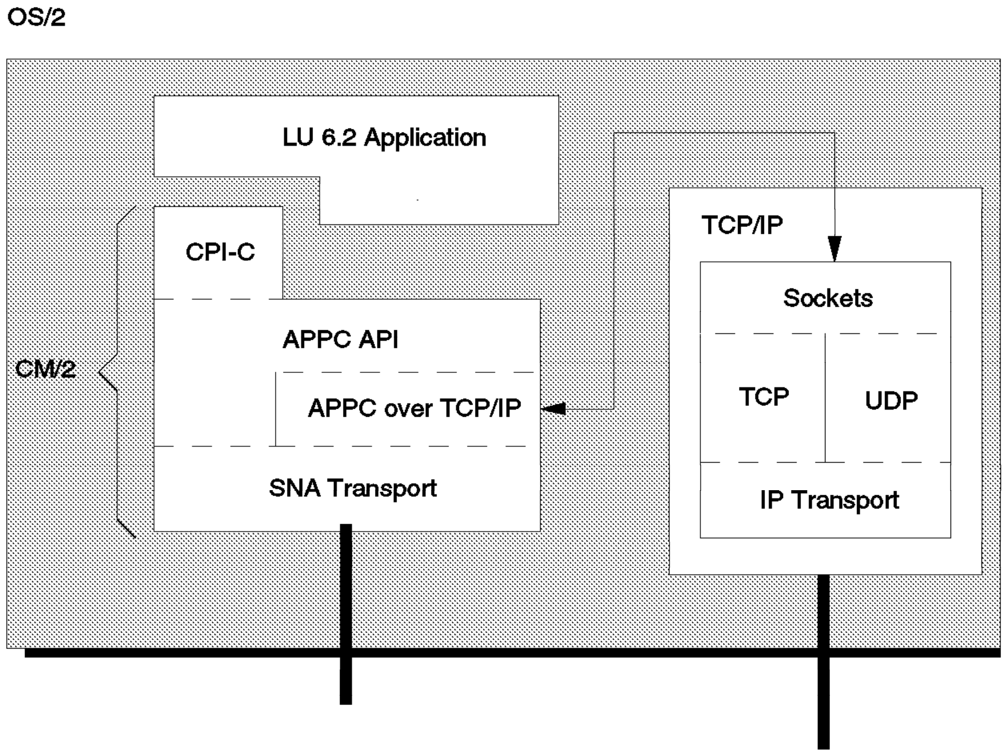


Figure 48. SNA over TCP/IP on OS/2. System Structure on OS/2

During the initialization of Communications Manager/2, SNA over TCP/IP determines whether it is enabled to route SNA frames over the IP network. If TCP/IP for OS/2 is not started, SNA over TCP/IP will not initialize. Communications Manager will route all SNA frames over the SNA network. The configuration information you define enables SNA over TCP/IP to determine which transport takes precedence, TCP/IP or SNA. Routing precedence is set by the user in the routing preference table. If the default transport is not available, SNA over TCP/IP uses the alternative transport.

SNA over TCP/IP on OS/2 supports all kinds of sessions (LU0, LU1, LU2, LU3, LU6.2).

5.11 Test Configuration for SNA over TCP/IP

Figure 49 on page 79 groups SNA and TCP/IP network configurations that enable communication among SNA programs using the IP network. In this example, nodes A and B can be configured with any operating system that supports SNA, for example, VSE/ESA, VM/ESA, MVS/ESA, OS/400 or OS/2. SNA over TCP/IP (MVS/ESA) and TCP/IP for MVS must be installed on nodes C and D. The LU2 **1** shown in this figure is a dependent LU in an EN node. This was used in the dependent LU scenario.

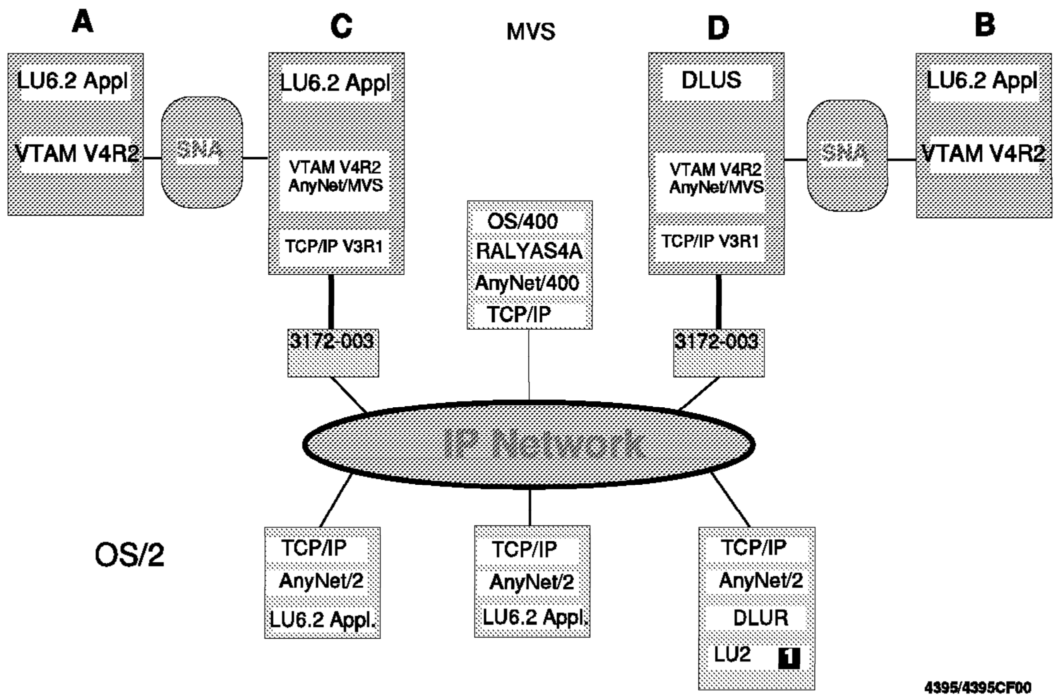


Figure 49. SNA over TCP/IP Implementation. MPTN network used in our test scenarios.

5.12 Network Management

SNA over TCP/IP operates as a TCP/IP socket application, which allows the use of NetView Family products to manage SNA over TCP/IP network activity: NetView to manage the SNA part and NetView/6000 to manage the underlying TCP/IP network. When you interconnect SNA and IP networks using SNA over TCP/IP on MVS/ESA and OS/2, NetView, together with NetView/6000 and NetView Service Point/6000, will allow you to do the following:

- Manage the SNA network using NetView
- Manage the TCP/IP network using NetView/6000
- Filter SNMP traps
- Convert SNMP traps to NetView alerts
- Customize alerts at NetView/6000
- Use the RUNCMD from NetView to NetView/6000 for any SNMP and TCP/IP command
- Use SNMP commands through an MVS/ESA TCP/IP server when IBM TCP/IP is installed
- Use MVS/ESA TCP/IP commands through TSO

Network Management Correlation

To get the most complete network picture you have to correlate the data from:

- Each VTAM acting as MPTN Access Node or MPTN Gateway Node using NetView
- Each OS/2 acting as MPTN Access Node
- The TCP/IP network using NetView/6000

Chapter 6. Running on MVS/ESA

This chapter presents the process of defining and running SNA over TCP/IP for MVS/ESA. It also summarizes how SNA over TCP/IP works on MVS/ESA. The information will be presented in the following order:

1. MVS SMP/E installation
2. TCP/IP definitions
   a. TCP/IP control blocks related to LU-LU sessions
   b. Environment related TCP/IP configuration options
   c. TCP/IP control blocks related to VTAM subtasks
   d. TCP/IP data sets used by SNA over TCP/IP
3. VTAM definitions
   a. TCP/IP major node
   b. Defining independent LUs (CDRSC major node)
   c. Defining dependent LUs (switched major node)
4. Mapping LU names to IP addresses
   a. Domain name server (DNS)
   b. HOSTLOCAL file
5. Initializing SNA over TCP/IP (MVS/ESA)
6. Operating SNA over TCP/IP (MVS/ESA)
7. Tuning SNA over TCP/IP (MVS/ESA)
8. Problem determination

6.1 MVS SMP/E Installation

Before starting to configure SNA over TCP/IP, ensure you finish the installation tasks described in *Program Directory ACF/VTAM AnyNet Feature for V4R2*. The installation instructions chapter in the program directory contains a post-SMP apply job named *ISTJEPST*. The purpose of this job is to resolve some external references listed in the same chapter.

Note!

The post-SMP apply job ISTJEPST must be run in the following two cases:

- After you complete the TCP/IP installation.
- Each time you run the TCP/IP job *EZAPPRFX*
  (See *Changing Data Set Prefixes* in the Program Directory for TCP/IP).

6.2 TCP/IP Definitions

This section lists the TCP/IP resources used by SNA over TCP/IP. Refer to *IBM TCP/IP Version 3 Release 1 for MVS: Customization and Administration Guide,* for more details on how to customize the TCP/IP Configuration Data Set.

Because AnyNet/MVS SNA over TCP/IP presents itself to TCP/IP as a socket application program, TCP/IP has to maintain a set of control blocks to serve the requests of this application. We can distinguish three kinds of control blocks:

- Those related to the LU-LU sessions
- Those related environment
- Those related to the VTAM tasks

6.2.1 Use of TCP/IP Control Blocks Related to LU-LU Sessions

For each LU-LU session established over an IP network, AnyNet/MVS SNA over TCP/IP creates one TCP connection. The following TCP/IP control blocks are used

- ACB — Activity Control Block.
See ACBPOOLSIZE Configuration Statement.
2 to 3 ACBs for Each LU-LU session.
- Data Buffers — Regular Data Buffer.
See DATABUFFERPOOLSIZE Configuration Statement.
1 to 2 Regular Data Buffers for Each LU-LU session.
- Envelopes — See ENVELOPEPOOLSIZE Configuration Statement.
1 to 4 Envelopes for Each LU-LU session.
- SCB — Socket Control Block.
See SCBPOOLSIZE Configuration Statement.
1 SCB for Each LU-LU session.
- SKCB — Socket Interface Control Block.
See SKCBPOOLSIZE Configuration Statement.
1 SKCB for Each LU-LU session.
- TCB — Transmission Control Block.
See TCBPOOLSIZE Configuration Statement.
1 TCB for Each LU-LU session.

Note: The TCP/IP TCB is not the same as the MVS TCB defined in the VBUILD type=TCP. Also the TCP/IP ACB is not the same as the VTAM ACB.

Make sure the values for the configuration statements you defined in the TCP/IP profile data set also account for the number of LU-LU sessions that can be activated across the IP network.

The TCP/IP profile data set we used for our test scenarios is shown in Appendix A, "System Definitions - MVS" on page 181.

6.2.2 Environment Related TCP/IP Configuration Options

Other environment-related TCP/IP variables are:

KEEPALIVE — The amount of time that TCP waits after receiving a packet for a connection before it sends a keep-alive packet to see if the other side is still alive. If there is no response, then TCP brings the connection down.

	The KEEPALIVEOPTIONS statement contains the interval in minutes to wait. The default interval is 120 minutes.
DATAGRAM QUEUE LIMIT	The default number of incoming datagrams queued to a UDP port is 21. By specifying NOUDPQUEUELIMIT on the ASSORTEDPARMS statement there is no limit to the number of queued incoming datagrams.

6.2.3 Use of TCP/IP Control Blocks Related to VTAM Subtasks

TCP/IP uses its client control blocks to interface with VTAM and serve the SNA over TCP/IP requests. The CCBPOOLSIZE configuration statement defines the number of available CCBs and consequently the number of subtasks VTAM will be allowed to start.

VTAM starts:

- Four subtasks when it activates the TCP/IP major node. See 6.3, "VTAM Definitions" on page 84 for a description of the TCP/IP major node.
- Additional subtasks as sessions are activated across the IP network.

The number of additional subtasks is limited to the value specified for the TCB operand in the TCP/IP major node. For example, if you use the default value of 10, VTAM starts 14 subtasks to service the TCP/IP major node.

Note

Each subtask uses one CCB. Consider that you have to balance the use of CCBs among SNA over TCP/IP and other TCP/IP applications.

6.2.4 Use of TCP/IP Data Sets by SNA over TCP/IP

SNA over TCP/IP is a TCP/IP socket application and uses a number of TCP/IP data sets including the TCPIP.DATA, HOSTS.SITEINFO and HOSTS.ADDRINFO. The use of these data sets and the search sequence for them is described in *IBM TCP/IP Version 3 Release 1 for MVS: Customization and Administration Guide*.

Note

If you are running TCP/IP Version 3 Release 1, TCPIP.DATA can be explicitly allocated by specifying //SYSTCPD DD statement. But during our testing it was found that VTAM was NOT using the search sequence as described in *IBM TCP/IP Version 3 Release 1 for MVS: Customization and Administration Guide*, even though a //SYSTCPD DD statement was included in the VTAM start procedure and pointed to SYS1.TCPPARMS(TCPDATA). VTAM always used the default high level qualifier to dynamically allocate TCPIP.DATA, HOSTS.SITEINFO, HOSTS.ADDRINFO, and other TCP/IP data sets.

6.3 VTAM Definitions

This section describes the three main steps to define SNA over TCP/IP to VTAM MVS/ESA in the following order:

1. TCP/IP major node
2. Defining independent LUs (CDRSC major node)
3. Defining dependent LUs (switched major node)

Before you start to configure SNA over TCP/IP, make sure you finish the installation tasks described in program directory ACF/VTAM AnyNet feature for V4R2.

6.3.1 TCP/IP Major Node

The IP network is represented to VTAM in a TCP/IP major node. To do so AnyNet/MVS provides a new VBUILD TYPE=TCP. You must define a PU which represents the adjacent link station for remote LUs.

In the following example you see our definition statements on SA11 used to define the TCP/IP major node. Only the more important parameters are explained. For a complete description please refer to *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP*.

```
**********************************************************************
*                                                                    *
*          VTAM 4.2   ANYNET   -      SNA OVER TCP/IP               *
*                                                                    *
*                     SA 11  DEFINITIONS                             *
*                                                                    *
*                 -------------------------------                    *
*                                                                    *
**********************************************************************
RABBSNIP VBUILD  TYPE=TCP,                                             X
               CONTIMER=30,            WAIT FOR MPTN TO COME UP        X
               DGTIMER=30,             INTERVAL BETWEEN RETRIES        X
               DNSUFX=IBM.COM,         DOMAIN NAME SUFFIX              X
               EXTIMER=3,              BETW. SEND SNA EXPEDITED DATA   X
               IATIMER=120,            TIME BEFORE MPTN KEEPALIVE      X
               PORT=397,               WELLKNOWN PORT FOR ANYNET       X
               TCB=50,                 NUMBER MVS SUBTASKS             X
               TCPIPJOB=T11ATCP        TCP/IP JOBNAME
RABGSNIP GROUP ISTATUS=ACTIVE          GROUPNAME
RABLSNIP LINE  ISTATUS=ACTIVE          LINENAME
RABPSNIP PU    ISTATUS=ACTIVE          PUNAME
```

Figure 50. SNA over TCP/IP Example TCP/IP Major Node definition

- The VBUILD TYPE=TCP is required to define a TCP/IP major node.
- DNSUFX stands for domain name suffix. This suffix is used by VTAM to generate a valid IP domain name in the form *luname.netid.DNSUFX*. Refer to 6.3.4, "How to Map the LU Names to the IP Addresses" on page 87 for detailed information.
- The PORT number specifies the TCP and UDP protocol port that VTAM uses to support SNA sessions over an IP network. It is recommended to use the

default value (397), which is the architected well known port for SNA over TCP/IP communication.

- The TCB operand is the number of MVS task control blocks (TCB) used by VTAM to access TCP/IP. See also how to code this operand in 6.2.3, "Use of TCP/IP Control Blocks Related to VTAM Subtasks" on page 83 and some performance hints in 6.5.1, "SNA over TCP/IP Performance Hints" on page 90.
- The TCPIPJOB operand must specify the TCP/IP job name as defined in the TCPIPJOBNAME statement coded in the TCP/IP Data File. You will find an example of this Data File in Figure 129 on page 194.
- If you plan to define more than one LINE definition statement in the GROUP you should code ISTATUS=INACTIVE and only one LINE with the operand ISTATUS=ACTIVE.
- The PU name represents the adjacent link station used to access the remote ILU. It is the same name as specified in:
  - The ALSLIST operand on the CDRSC definition statement for this ILU
  - The ALS selection function of the Session Management Exit routine (SME) if it is used

  If you plan to define more than one PU definition statement in this LINE you should code ISTATUS=INACTIVE and only one PU with the operand ISTATUS=ACTIVE. Only one PU must be active at one time. See the note and recommendations hereafter.

Note:

1. More than one TCP/IP major node can be active at one time, but:
   - Only one LINE among all TCP/IP major nodes can be active at one time.
   - Only one PU on this line can be active at one time.
2. You can define more than one GROUP, LINE and PU in a single TCP/IP major node, but:
   - Only one LINE among all TCP/IP major nodes can be active at one time.
   - Only one PU on this line can be active at one time.
3. You may want to take advantage of the sift-down effect by coding ISTATUS=ACTIVE in the GROUP definition statement. In this case, only the first PU of the first LINE will be activated. The other LINE and PU activations will fail.

6.3.2 Defining Independent LUs

Remote independent LUs are represented to AnyNet/MVS SNA over TCP/IP in a cross-domain resource major node.

You must define a CDRSC for the destination ILUs using one of the following:

- A CDRSC definition statement
- The ALS selection function of the session management exit routine (SME)

If a remote independent logical unit (ILU) in an IP Network initiates a LU-LU session, one possibility is to let VTAM define the LU dynamically. In this case the DYNLU=YES start option must be coded in the destination VTAM host.

You can also code a cross-domain resource major node containing the LU.

We defined our independent LUs in a major node. In the following example, you see our definitions:

```
**********************************************************************
         VBUILD  TYPE=CDRSC
**********************************************************************
WTR05221 CDRSC ALSLIST=RABPSNIP,ALSREQ=YES
RAIAC    CDRSC ALSLIST=RABPSNIP,ALSREQ=YES
RAPAC    CDRSC ALSLIST=RABPSNIP,ALSREQ=YES
**********************************************************************
```

Figure 51. SNA over TCP/IP Example CDRSC Major Node

The name of the cross-domain resource must be the independent LU name.

The ALSLIST definition must contain the name of the PU statement defined in the TCP/IP major node.

6.3.3 Defining Dependent LUs

Remote dependent LUs are represented to AnyNet/MVS SNA over TCP/IP in a switched major node.

```
RABSSNIP VBUILD MAXDLUR=4,                                          X
               TYPE=SWNET
ISNIPJ01 PU    ADDR=01,                                             X
               IDBLK=05D,                                           X
               IDNUM=05221,                                         X
               DISCNT=NO,                                           X
               ISTATUS=ACTIVE,                                      X
               MAXDATA=1033,                                        X
               MAXPATH=4,                                           X
               PACING=0,                                            X
               PUTYPE=2,                                            X
               DLOGMOD=D4C32XX3,                                    X
               MODETAB=ISTINCLM,                                    X
               USSTAB=US327X,                                       X
               VPACING=0
*
*      PATH  PID=1,                      required for VTAM        X
*            DLURNAME=WRT05221,          to initiate activation   X
*            DLCADDR=(1,C,INTPU),        of a PU                  X
*            DLCADDR=(2,X,05D05221)
*
*
ISNIPJL1 LU    LOCADDR=2
ISNIPJL2 LU    LOCADDR=3
ISNIPJL3 LU    LOCADDR=4
ISNIPJL4 LU    LOCADDR=5
```

For more information please refer to VTAM Network Implementation Guide (SC31-6494) on Dependent LUs, and Guide to SNA over TCP/IP Version 4 Release 2 for MVS/ESA (SC31-6527) on Defining Independent and Dependent Logical Units for SNA over TCP/IP.

6.3.4 How to Map the LU Names to the IP Addresses

This section describes how to ensure that an SNA LU name and SNA network ID are mapped to a unique IP address.

For an OUTBOUND flow (from SNA network to IP network), the mapping process starts with SNA network-qualified LU names which are reformatted into TCP/IP fully qualified domain names and then mapped to an IP addresses.

For an INBOUND flow (from IP network to SNA network), the SNA LU name is passed in the MPTN connect command.

1. SNA Network-Qualified LU Names
   - The SNA network knows the two communicating LUs by their network ID and LU names: *netid.luname*
   - The SNA network-qualified LU names are formatted in decreasing scope (netid comes first).
2. TCP/IP Fully Qualified Domain Names
   - SNA over TCP/IP reformats the LU names into IP domain names: *luname.netid.DNSUFX*
   - TCP/IP domain names are formatted in increasing scope (luname comes first).
   - The domain name suffix (DNSUFX parameter in the TCP/IP major node) ensures that any SNA LU name and SNA network ID specified in the form *luname.netid* are distinct from any existing IP domain name.
   - The maximum length for a TCP/IP fully qualified name is 255 characters (refer to RFC1034). So you have to ensure that domain name suffix, luname and netid altogether (including the periods) don′t exceed this lengths.

   Note!

   TCP/IP for MVS has a limitation of 24 characters for a fully qualified name if using the HOSTS.LOCAL file.

3. IP address mapping
   - Converting the TCP/IP fully qualified domain name to an IP address can be done either in the Domain Name Server or by using the HOSTS.LOCAL file.
   - You can use the HOSTS file to define domain names and IP addresses in your local network. However, if your IP network interconnects to other IP networks, it is recommended that you use the domain name server instead, to ensure that the local names and addresses are consistent with the names and addresses established for the other networks.

6.4 Initializing AnyNet/MVS SNA over TCP/IP

To start AnyNet/MVS SNA over TCP/IP, you have to perform the following steps:

- Start VTAM.
- Start TCP/IP.
- Activate your TCP/IP major node and the PU defined in this major node.

- Activate the required cross-domain resource major node for independent LUs or the switched major node for dependent LUs.
- The VTAM DISPLAY command may be used to verify the status of resources.

To ensure that AnyNet/MVS SNA over TCP/IP is properly installed, execute the following display:

```
D NET,STATS,TYPE=VTAM,NUM=*
IST097I  DISPLAY  ACCEPTED
IST350I  DISPLAY TYPE = STATS,TYPE=VTAM
IST1349I  COMPONENT ID IS 5695-11701-201
IST1345I     ID      VALUE        DESCRIPTION
IST1227I    151          0 = DEPENDENT LU TOTAL FOR ISTPUS11

...

IST1227I    130        YES = ANYNET/MVS SNA OVER TCP/IP INSTALLED
IST1227I    127          1 = TCP/IP MAJOR NODES
IST1227I    128         50 = MAXIMUM TCB VALUE FOR TCP/IP MAJOR NODES
IST1227I    129          2 = TCP/IP LU-LU SESSIONS

...
```

The above display is only an extract of the total display.

The message YES = ANYNET/MVS SNA OVER TCP/IP INSTALLED indicates that your AnyNet/MVS SNA over TCP/IP is properly installed.

Important!

If you do not code *num=** you will only see the first ten rows of the display (this is the default value). Probably you will not find the AnyNet information in the first ten rows.

6.5 Operating AnyNet/MVS SNA over TCP/IP

In the following examples we list some console display samples for a running AnyNet/MVS SNA over TCP/IP environment. We highlighted both the new and important information given by the display commands.

1. Display of the TCP/IP major node in SA11:

```
DISPLAY NET,ID=RABBSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABBSNIP, TYPE = TCP/IP MAJOR NODE
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST1342I  DNSUFX = IBM.COM
IST1344I  TCPIPJOB = T11ATCP   TCB = 50  TCP PORT =   397
IST1400I DGTIMER  =    30  EXTIMER =     3
IST1406I CONTIMER =    30  IATIMER =   120
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST170I LINES:
IST232I RABLSNIP, ACTIV
IST314I END
```

The IST075I message group shows you this new type of major node and information messages which show you the parameters defined in the TCP/IP major node.

2. Display of the TCP PU defined in the TCP major node:

```
DISPLAY NET,ID=RABPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABPSNIP, TYPE = PU_T2.1
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RABLSNIP, LINE GROUP = RABGSNIP, MAJNOD = RABBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I WTR05221 ACT/S      ISNIPJL2 ACT/S
IST314I END
```

The above display shows you that the TCP PU is active and is in session with two LUs.

You should always make sure that the PU is active. Otherwise your connections via AnyNet/MVS SNA over TCP/IP do not work.

3. Display of a cross-domain resource major node:

```
DISPLAY NET,ID=RABRSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABRSNIP, TYPE = CDRSC SEGMENT 432
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST478I CDRSCS:
IST483I WTR05221 ACT/S     , CDRM = ***NA***, NETID = USIBMRA
IST483I WTR05222 ACTIV     , CDRM = ***NA***, NETID = ***NA***
IST483I WTR05115 ACTIV     , CDRM = ***NA***, NETID = ***NA***
IST483I RAPAC    ACTIV     , CDRM = ***NA***, NETID = ***NA***
IST483I RA3AC    ACTIV     , CDRM = RA3     , NETID = ***NA***
IST483I RALYAS4B ACTIV     , CDRM = ***NA***, NETID = ***NA***
IST314I END
```

4. Display of a the remote independent LU WTR05221:

```
DISPLAY NET,ID=WTR05221,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.WTR05221, TYPE = CDRSC
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RABRSNIP
IST1184I CPNAME = USIBMRA.RAB - NETSRVR = ***NA***
IST1044I ALSLIST = RABPSNIP
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000002, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RABPSNIP
IST634I NAME     STATUS         SID              SEND RECV VR TP NETID
IST635I RAB      ACTIV/DL-S D96FD635E0901E31 0010 0000       USIBMRA
IST635I RAB      ACTIV/DL-P F7EFD164B8FDB804 0000 0012       USIBMRA
IST1355I PHYSICAL UNITS SUPPORTED BY DLUR USIBMRA.WTR05221
IST089I ISNIPJ01 TYPE = PHYSICAL UNIT    , ACTIV
IST314I END
```

In the ALSLIST statement you see the name of the TCP PU defined in the TCP major node.

For more information on the session, you can display the session identification (sid).

5. Display of the session D96FD635E0901E31:

```
D NET,SESSIONS,SID=D96FD635E0901E31
IST097I DISPLAY ACCEPTED
IST350I DISPLAY TYPE = SESSIONS 548
IST879I PLU/OLU REAL = USIBMRA.WTR05221  ALIAS = ***NA***
IST879I SLU/DLU REAL = USIBMRA.RAB       ALIAS = ***NA***
IST880I SETUP STATUS = ACTIV
IST875I ALSNAME TOWARDS PLU = RABPSNIP
IST933I LOGMODE=CPSVRMGR, COS=*BLANK*
IST1165I  LOCAL  TCP/IP   ADDRESS = 9.67.38.11..397
IST1165I  REMOTE TCP/IP   ADDRESS = 9.67.38.36..1352
IST314I END
```

In this display we see the two new messages IST1165I giving us the TCP/IP network addresses of both Local and Remote LUs.

6. For more information on the session you can also use the NetView Session Monitor.

6.5.1 SNA over TCP/IP Performance Hints

The SNA over TCP/IP function supports a maximum of 11,800 sessions per SNA over TCP/IP node, whether VTAM is acting as an MPTN access node, or VTAM is acting as an MPTN gateway node.

Consider the following:

- Each TCP connection is associated with an SNA session.
- The maximum TCB is 99 (value specified in the TCP/IP major node).
- There is one TCP/IP CCB per MVS TCB.
- Transmission Control Protocol/Internet Protocol handles synchronously the read and write calls through one CCB.

That means a maximum of 99 sessions/connections can be processed concurrently. Too many MVS/ESA TCBs coded in the VBUILD increases the overhead for MVS/ESA system task dispatching, but a large number of TCBs coded permits more parallel synchronous processing among the MVS/ESA tasks.

6.6 Diagnosing SNA over TCP/IP (MVS/ESA)

In this chapter we will introduce the new messages for AnyNet/MVS and the possibilities you have to trace AnyNet related data.

6.6.1 VTAM Diagnostic Messages for SNA over TCP/IP

- IST1387I TCP PU *puname* IS UNABLE TO ACCEPT CONNECTION REQUESTS
- IST1388I SOCKET *callname* CALL FAILED, TCP ERROR NUMBER = *errno*
- IST1389I NO TCB IS AVAILABLE FOR SOCKET
- IST1390I NO SOCKET DESCRIPTOR IS AVAILABLE

These new messages warn the user that VTAM is unable to accept inbound connections from a TCP/IP network.

6.6.2 Traces

In VTAM three types of trace are available to diagnose SNA over TCP/IP problems.

- VTAM I/O trace
- VTAM Buffer trace
- VTAM Internal trace

In the following subchapters we will give you a short description of these three traces. For more information please refer to *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP* and *VTAM Diagnosis Version 4 Release 2 for MVS/ESA* .

6.6.2.1 VTAM I/O Trace

The I/O trace provided by SNA over TCP/IP traces up to 256 bytes of the MPTN format. You may want to trace:

- All the messages handled by SNA over TCP/IP for one destination LU, then enter:

  F NET,TRACE,TYPE=IO,ID=***CDRSC_name***

- All the messages handled by SNA over TCP/IP for all destination LUs, then enter:

  F NET,TRACE,TYPE=IO,ID=***TCP_PU_name***,SCOPE=ALL

6.6.2.2 VTAM Buffer Trace

If you use the BUFFER trace, you will not get any MPTN format output. You will get only the FID4 PIUs flowing to SNA over TCP/IP:

F NET,TRACE,TYPE=BUF,ID=***CDRSC_name***

6.6.2.3 VTAM Internal Trace

AnyNet/MVS provides some new VIT trace entries. There is a new VIT option TCP, added to the VTAM internal trace (VIT). This is a sample of the command to start this trace:

```
F NET,TRACE,TYPE=VTAM,OPTION=TCP
```

Chapter 7. AnyNet/2 SNA over TCP/IP

This chapter presents the process of defining and running AnyNet/2 SNA over TCP/IP. This information will be presented in the following logical order:

- Installing and configuring SNA over TCP/IP
- Setting up TCP/IP for OS/2 for SNA over TCP/IP
- Setting up Communications Manager for SNA over TCP/IP
- Mapping LU names to IP addresses
- Defining and modifying the routing preference table
- Running applications on SNA over TCP/IP
- Debugging problems related to SNA over TCP/IP

We start by showing, in the following diagram, the relationship of the various components and the new terminology we will discuss in this chapter:

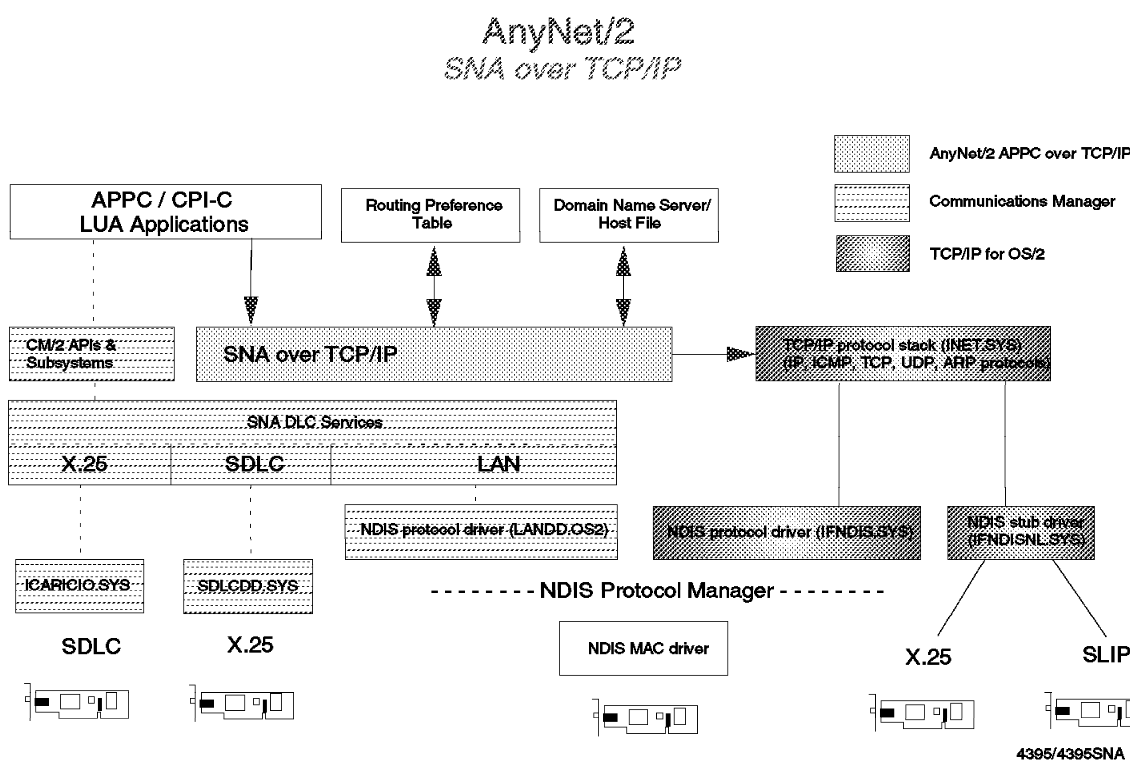


Figure 52. SNA over TCP/IP (OS/2) - System Structure

The AnyNet/2 product actually provides two ways for running SNA applications over a TCP/IP transport network:

1. *APPC over TCP/IP*

   APPC over TCP/IP only supports LU6.2 applications

2. *SNA over TCP/IP*

   SNA over TCP/IP will support LU6.2 as well as LU0, LU1, LU2, and LU3 applications. SNA over TCP/IP supports both, dependent and independent LUs.

Throughout this chapter and the adjacent scenarios, we shall test and describe SNA over TCP/IP only since it includes all functions provided by APPC over TCP/IP.

7.1 Installing AnyNet/2 SNA over TCP/IP

Before you install AnyNet/2 SNA over TCP/IP, make sure all the required programs are properly installed and operational. The basic requirements are:

- OS/2
- Communications Manager
- TCP/IP for OS/2

To allow SNA applications to make use of SNA over TCP/IP, you have to install the AnyNet/2 support feature of Communications Manager. You must use IBM Communications Manager/2 Version 1.11 or later in order to support SNA over TCP/IP. Older versions of Communications Manager only support APPC over TCP/IP.

To install the AnyNet/2 Support feature of IBM Communications Manager/2 Version 1.11, select it to be installed from the list of options on the Install Additional Functions menu. You get to this menu after finishing with your first IBM Communications Manager/2 Version 1.11 configuration. You can later access this menu by clicking on the **Options** pull-down menu of the Communications Manager Setup program, and by then selecting **Install additional functions**.

The IBM Communications Manager/2 Version 1.11 AnyNet Support feature will install the following files into the specified sub-directories:

- CMLIB
  - LUTPM.EXE
  - LUTPM.HLP
  - PLUVERB.H
  - CMMH.MSG
  - CMM.MSG
  - LULIST.EXE
- CMLIBDLL
  - LUTAB.DLL
  - ACSMPTN.DLL

Make sure these files are present prior to installing AnyNet/2 SNA over TCP/IP. Figure 53 on page 95 shows the Install Additional Functions menu of IBM Communications Manager/2 Version 1.11.

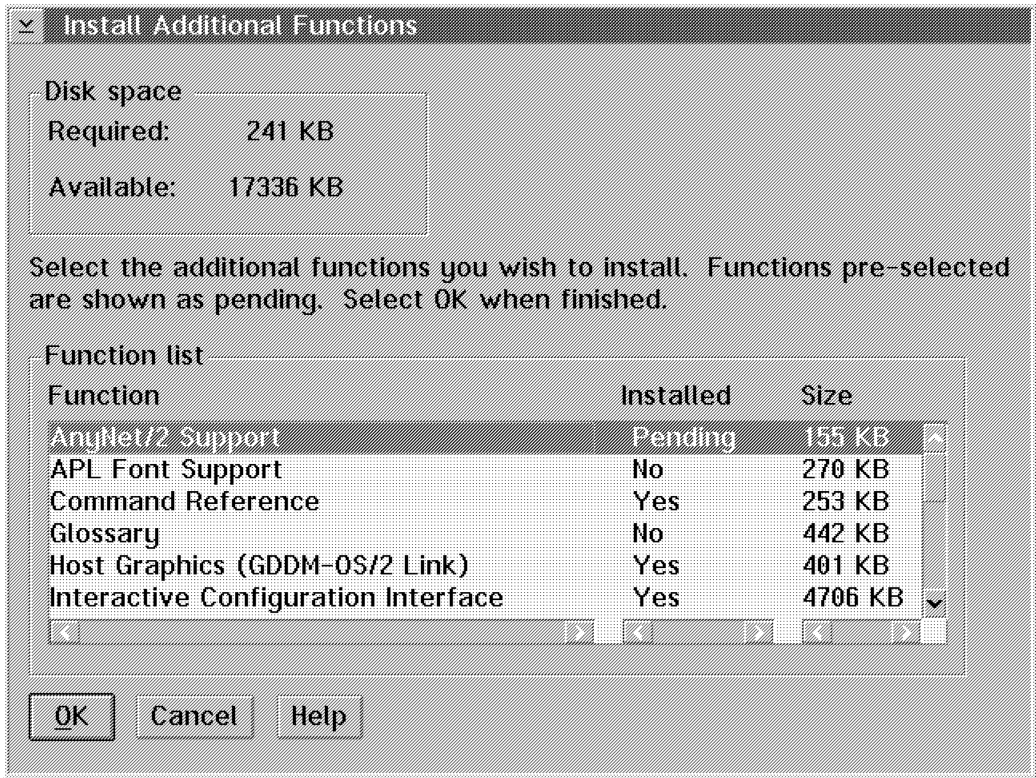


Figure 53. Installing AnyNet/2 Support of IBM Communications Manager/2 Version 1.11

Please refer to *IBM Communications Manager/2 Version 1.1 Workstation Installation and Configuration Guide* for more details on installing and configuring IBM Communications Manager/2.

There are several ways to install SNA over TCP/IP in the OS/2 environment. Basically, the code can be obtained either by downloading it from the MVS host where the VTAM AnyNet IWS feature of VTAM V4R2 is installed, or as part of the AnyNet/2 product. In both cases, SNA over TCP/IP for OS/2 can be installed:

- From the VTAM downloaded files
- From product diskettes
- From a code server

For our scenarios we used the AnyNet/2 product and chose the diskette method of installation which will be described further in this section.

For a more detailed discussion on:

- Downloading the SNA over TCP/IP for OS/2 code from MVS
- Setting up a code server
- Installing from a code server

please refer to the following documentations: *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP* or *IBM AnyNet/2 Version 2.0: Guide to SNA over TCP/IP*.

To install the AnyNet/2 SNA over TCP/IP code from the product diskettes, insert diskette 1 of AnyNet/2 into, for instance, drive A: and enter

```
a:install
```

from the OS/2 command prompt.

From the Installation and Maintenance menu which is shown in Figure 54 choose **SNA over TCP/IP**, then select **Install** from the Action pull-down menu.

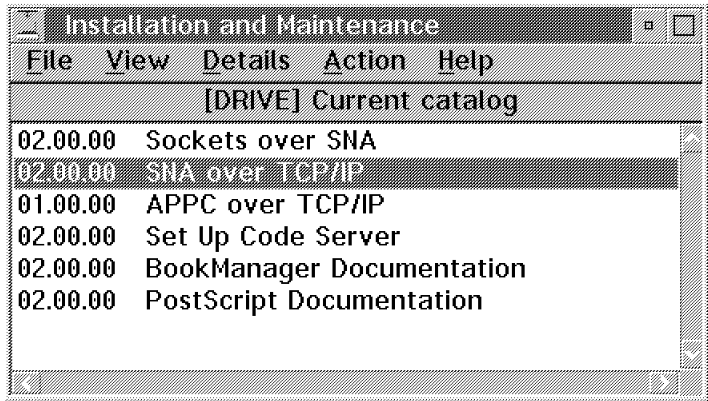


Figure 54. AnyNet/2 Installation and Maintenance

This menu serves several purposes:

- Installs product code
- Sets up a code server
- Installs product documentation
- Checks available disk space
- Checks for already installed versions of AnyNet/2

Leave the following panel unchanged. If you are installing AnyNet/2 for the first time, make sure that the check box to update the CONFIG.SYS file is checked.

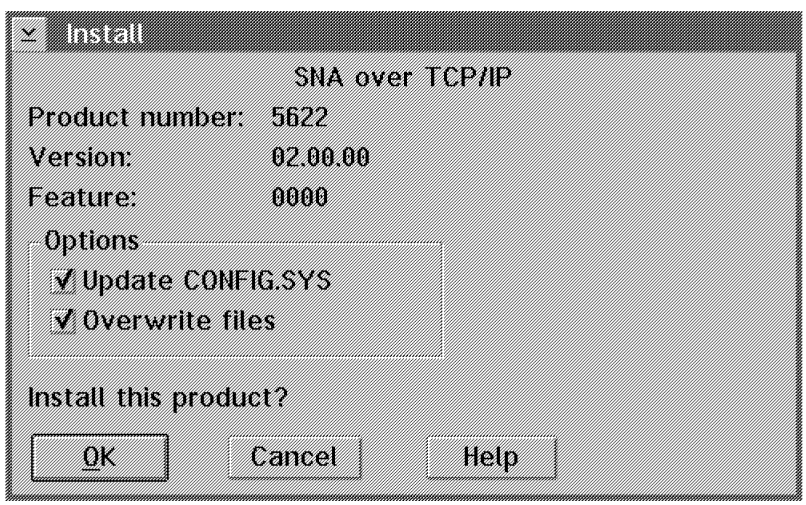


Figure 55. AnyNet/2 SNA over TCP/IP Installation

We also suggest that you use the default path for the AnyNet/2 files. For SNA over TCP/IP that would be ANYNET on the boot drive of your OS/2 system. You also have to specify the path where IBM Communications Manager/2 is installed. That will, especially in the latter case, ensure that all necessary files for AnyNet/2 can be found in the system paths.

As part of the installation process you will be asked to do a primary configuration for SNA over TCP/IP. See 7.3, "Configuring AnyNet/2 SNA over TCP/IP" on page 98 for more information about this task.

A progress indicator will inform you how far the installation has proceeded and which files are currently being unpacked or installed.

After the installation procedure has finished, you will return to the Installation and Maintenance menu. If you want to install product documentation, either in PostScript or in BookManager Read/2 format, you may do so now or at a later time. Otherwise, select **Exit** from the AnyNet/2 Installation window.

Do not forget to reboot your workstation before running AnyNet/2 SNA over TCP/IP.

7.1.1 Changes to CONFIG.SYS

The following changes will be made to the CONFIG.SYS file of your OS/2 system if you selected it to be updated during installation:

```
SET SNASUFFIX=IBM.COM
SET CRITICAL_WS=NO
SET CONNWAIT_SECS=30
SET CONN_RETRY_SECS=300
SET MPTN_WELL_KNOWN_PORT=397
SET TCPWAIT_MINS=15
SET UNACKED_DG_RETRY_SECS=30
SET UNSENT_DG_RETRY_SECS=3
SET INACTIVITY_TIMER_SECS=90
SET ANYNETPATH=C:\ANYNET
```

The entries shown above are a result of your AnyNet/2 SNA over TCP/IP configuration and may look different according to your environment. They will be discussed in 7.3, "Configuring AnyNet/2 SNA over TCP/IP" on page 98.

SNA over TCP/IP will also add the AnyNet/2 program directory to the system path statements in the CONFIG.SYS file which may look as follows:

```
LIBPATH=C:ANYNETDLL; ...
SET PATH=C:\ANYNET\BIN; ...
SET DPATH=C:\ANYNET\MISC; ...
```

7.1.2 New Files and Desktop Elements

AnyNet/2 SNA over TCP/IP will install the following files into the specified sub-directories:

- ANYNET
  - ANYCFG.ICO
  - ANYFLDR.ICO
  - ANYNET.ICO
  - ANYNETCF.EXE
  - ANYNETCF.HLP
  - CHECKCM.EXE
  - ISTSKKEY.PKG
  - SNIP2.PKG
  - SYSLEVEL.ANY
- ANYNETBIN

– ABINFO.EXE
– ANMB.EXE
– GETIPINT.EXE
– PWDCHECK.EXE

• ANYNETDLL

– ANUTIL.DLL
– MPTNCMM.DLL
– TCPPMM.DLL

• ANYNETMISC

– ANY.MSG
– ANYH.MSG
– ANYNET.CFN

AnyNet/2 will also create new desktop elements for SNA over TCP/IP as shown in the following figure.

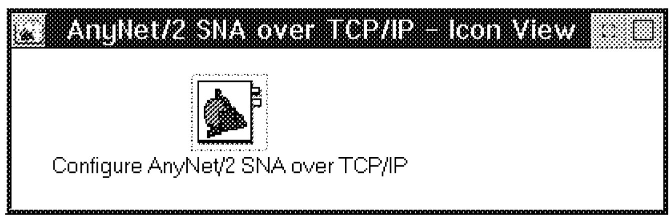


Figure 56. AnyNet/2 SNA over TCP/IP Folder

7.2 Removing AnyNet/2 SNA over TCP/IP

To remove AnyNet/2 SNA over TCP/IP from your system, do the following:

1. Stop SNA over TCP/IP by stopping CM/2.
2. Run the INSTALL procedure from diskette 1.
3. Select **SNA over TCP/IP** on the Installation Menu.
4. Select **Delete** from the Action pull-down menu.
5. Restore your original CONFIG.SYS file by removing or uncommenting all changes made by SNA over TCP/IP.
6. Reboot your system.

7.3 Configuring AnyNet/2 SNA over TCP/IP

Before actually performing the task of configuration, you should already be familiar with SNA over TCP/IP LU-IP address mapping and routing preference tables, and you should already have installed configured TCP/IP for OS/2 on your system. For a more detailed discussion of these topics, please refer to Chapter 5, “SNA over TCP/IP Implementation” on page 73.

Configuration for AnyNet/2 SNA over TCP/IP is performed from within the AnyNet/2 Configuration Notebook. This will be displayed during installation and can later be loaded either from the AnyNet/2 folder or with the following OS/2 command:

ANYNETCF

The primary purpose of this tool is to set up the routing preference table, OS/2 environment variables, and TCP/IP timer values to be used for SNA over TCP/IP.

The first page of the configuration notebook specifies the domain name suffix for SNA over TCP/IP, as shown in the following figure.

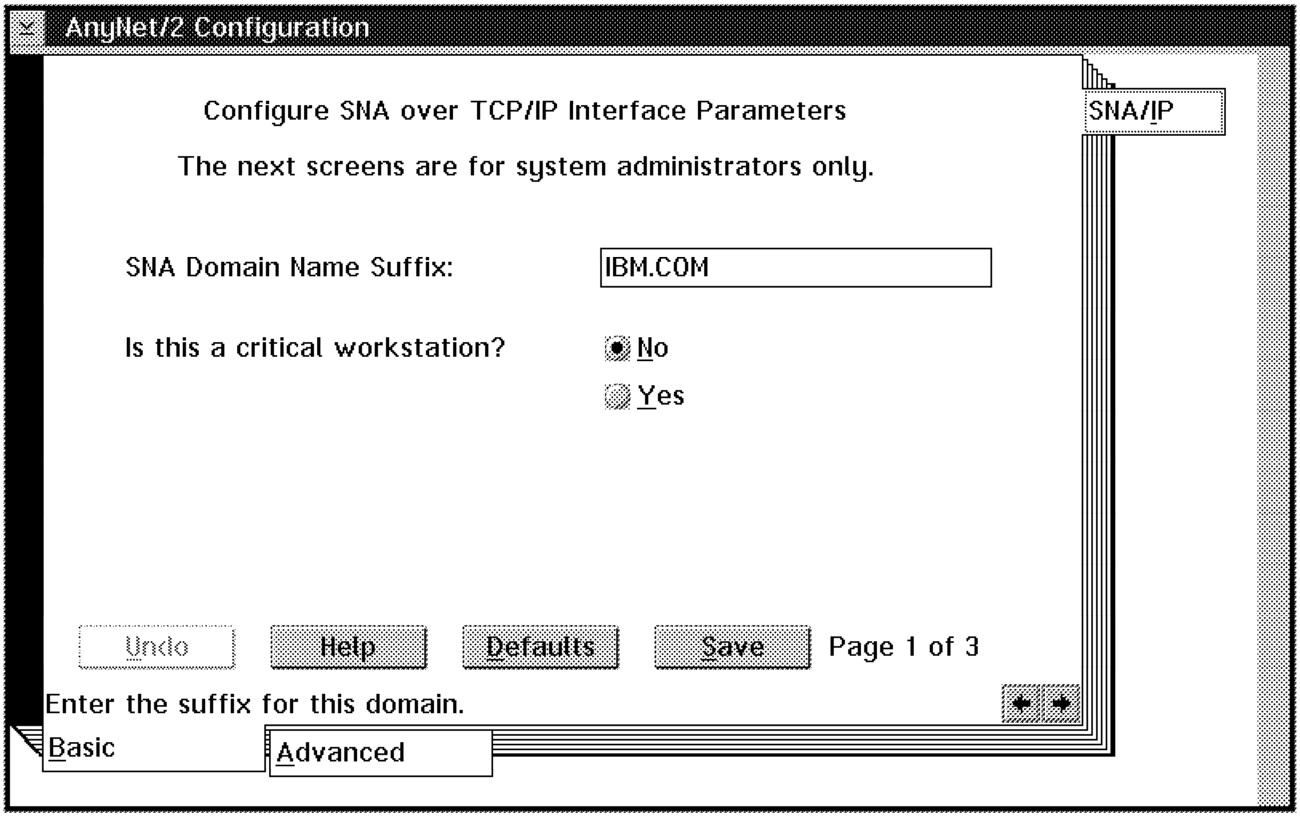


Figure 57. AnyNet/2 SNA over TCP/IP Configuration, Page 1 - Interface Parameters I

The following table explains what to configure in the above panel.

Table 2. AnyNet/2 SNA over TCP/IP Configuration Page 1 - Interface Parameters I

Setting	**Meaning**
SNA Domain Name Suffix	This field defines the SNA domain name suffix to be used when SNA over TCP/IP creates an IP domain name from an SNA LU name, SNA network ID, and this suffix. The resulting domain name will be mapped to an IP address using a domain name server, a local hosts file, or both. This setting relates to the following CONFIG.SYS entry: `SET SNASUFFIX=IBM.COM`
Critical Workstation	If you define a workstation as critical, it will send both conditional and unconditional alerts. Please see *VTAM AnyNet Feature for V4R2: Guide to SNA over TCP/IP* or *IBM AnyNet/2 Version 2.0: Guide to SNA over TCP/IP* for more information on these alerts (″MPTN Messages and Error Codes). This setting relates to the following CONFIG.SYS entry: `SET CRITICAL_WS=NO`

The second page specifies the TCP and UDP port, and some timers to be used by SNA over TCP/IP.

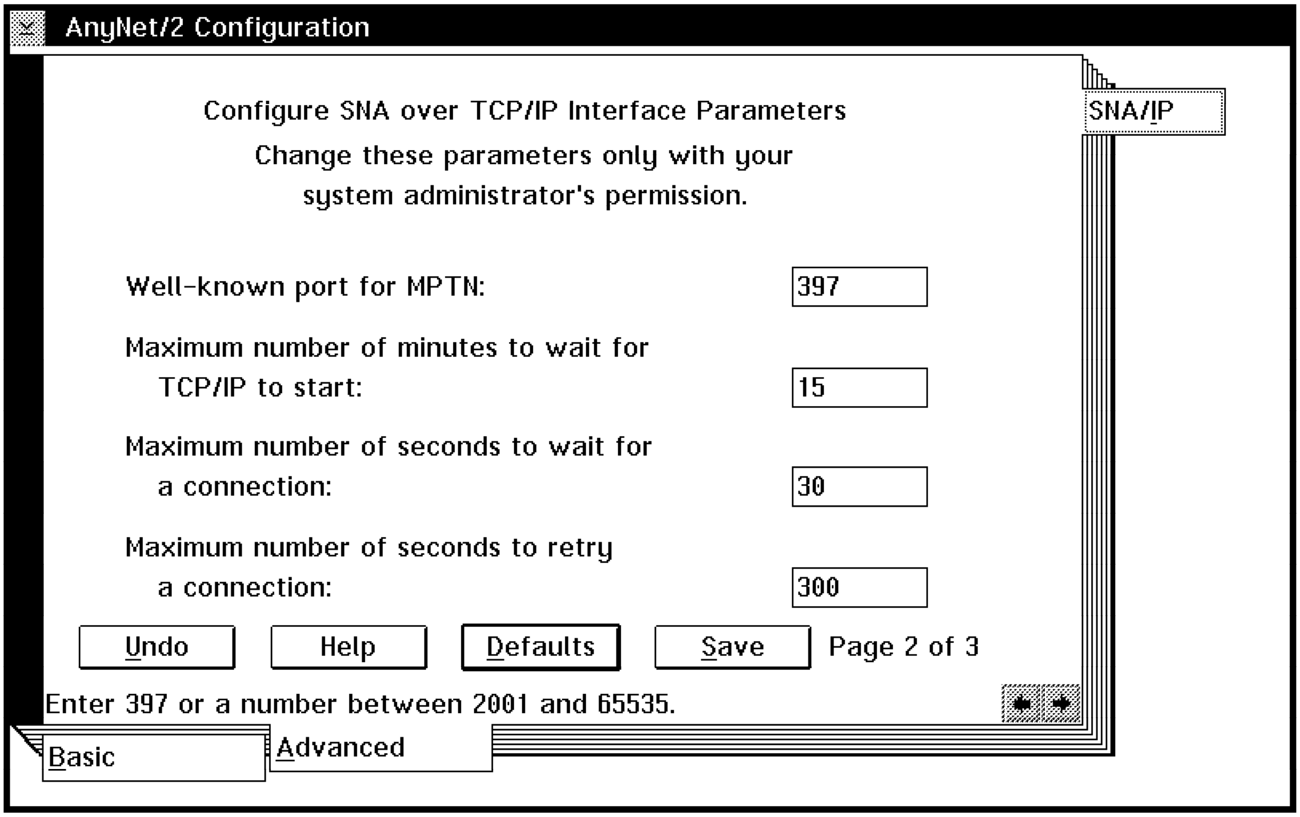


Figure 58. AnyNet/2 SNA over TCP/IP Configuration, Page 2 - Interface Parameters II

The following table explains what to configure in the above panel.

Table 3 (Page 1 of 2). AnyNet/2 SNA over TCP/IP Configuration Page 2 - Interface Parameters II

Setting	**Meaning**
Well-known port for MPTN	This field allows you to define an alternative to the well-known port used by SNA over TCP/IP, which is port number 397 by default. MPTN nodes running SNA over TCP/IP must use the same well-known port to be able to communicate. This setting relates to the following CONFIG.SYS entry: `SET MPTN_WELL_KNOWN_PORT=397`
Max. # of minutes to wait for TCP/IP	This field allows you to define how long SNA over TCP/IP will wait for TCP/IP to become active if you have started Communications Manager before OS/2 TCP/IP. This setting relates to the following CONFIG.SYS entry: `SET TCPWAIT_MINS=15`
Max. # of seconds to wait for connection	Specify how long SNA over TCP/IP will wait for an MPTN connection after the native TCP/IP connection has been established. This setting relates to the following CONFIG.SYS entry: `SET CONNWAIT_SECS=30`

Table 3 (Page 2 of 2). AnyNet/2 SNA over TCP/IP Configuration Page 2 - Interface Parameters II

Setting	Meaning
Max. # of seconds to retry connection:	Specify how long SNA over TCP/IP will try to establish a connection. This setting relates to the following CONFIG.SYS entry: `SET CONN_RETRY_SECS=300`

The third page specifies additional timers to be used by SNA over TCP/IP.

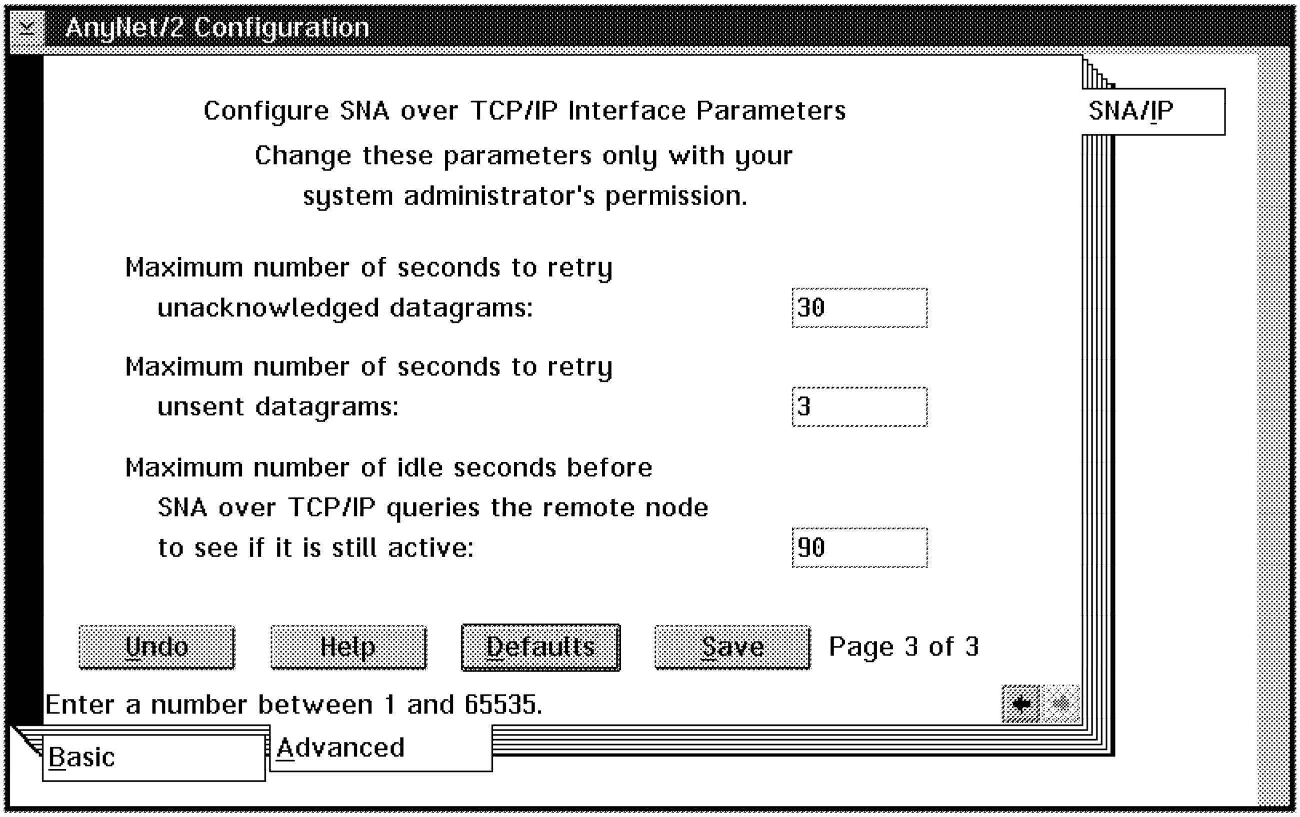


Figure 59. AnyNet/2 SNA over TCP/IP Configuration Page 3 - Interface Parameters III

The following table explains what you configure in the above panel.

Table 4 (Page 1 of 2). AnyNet/2 SNA over TCP/IP Configuration, Page 3 - Interface Parameters III

Setting	Meaning
Max. # of seconds to retry unacknowledged datagrams:	Specify the amount of time after which SNA over TCP/IP will resend unacknowledged out-of-band (OOB) and MPTN KEEPALIVE datagrams. SNA over TCP/IP may use OOB datagrams for expedited data. Such data contains important control messages that are allowed to travel ahead of normal data and are not subject to congestion control. This setting relates to the following CONFIG.SYS entry: `SET UNACKED_DG_RETRY_SECS=30`

Table 4 (Page 2 of 2). AnyNet/2 SNA over TCP/IP Configuration, Page 3 - Interface Parameters III

Setting	Meaning
Max. # of seconds to retry unsent datagrams:	Specify the amount of time for SNA over TCP/IP to wait for an acknowledgement for expedited data on the TCP connection before it will send expedited data as an OOB datagram. This setting relates to the following CONFIG.SYS entry: `SET UNSENT_DG_RETRY_SECS=3`
Max. # of idle seconds before querying a remote node:	Specify the amount of idle time after which SNA over TCP/IP will try to determine whether a remote IP address is still alive. This setting relates to the following CONFIG.SYS entry: `SET INACTIVITY_TIMER_SECS=90`

To finish configuration, press the Save button and then select **Save and Exit**.

After the configuration has finished, you can set up the routing preference table for SNA over TCP/IP. This is done using the LUTPM utility which is automatically invoked by the installation program. You can use it any time later to modify the routing preference table.

Note!

A fix to APAR IC08187 is available to correct two minor problems with the LU preference table implementation.

Figure 60 shows the LUTPM tool.

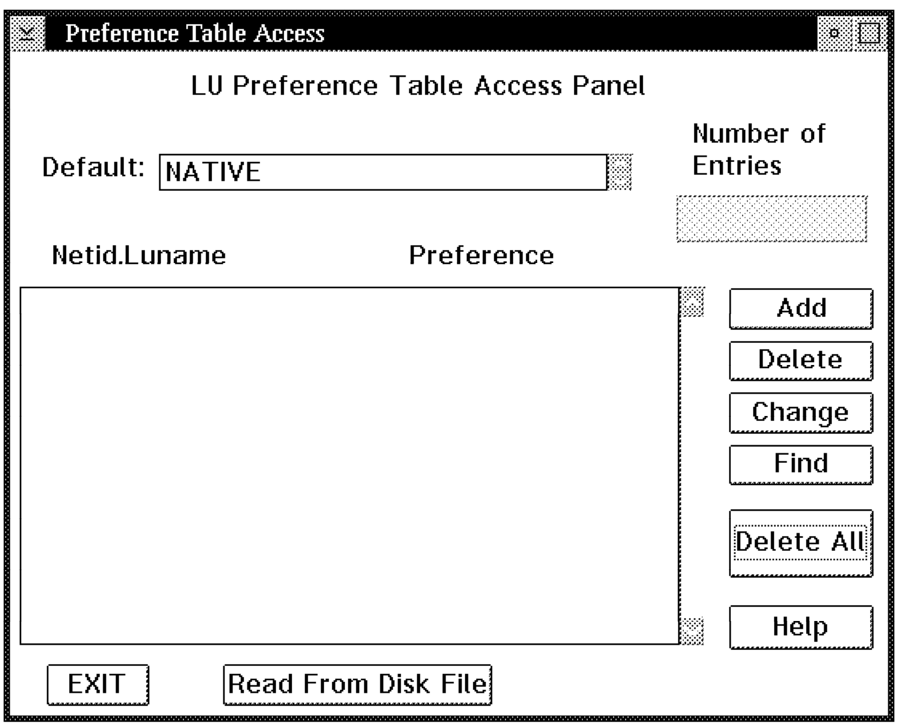


Figure 60. Routing Preference Table for AnyNet/2 SNA over TCP/IP

The following table explains what to configure in the above panel.

Table 5. Routing Preference Table for AnyNet/2 SNA over TCP/IP

Setting	**Meaning**
Default	Specify how you want SNA over TCP/IP to establish a connection by default: 1. NATIVE Try SNA first, then TCP/IP 2. NON-NATIVE Try TCP/IP first, then SNA 3. NATIVE ONLY Try SNA only 4. NON-NATIVE ONLY Try TCP/IP only
LuName Preference	Specify how you want SNA over TCP/IP to establish a connection for a particular partner LU. The same possibilities as above are applicable. Figure 61 on page 103 shows the LUTPM menu to add LU names to the routing preference table.

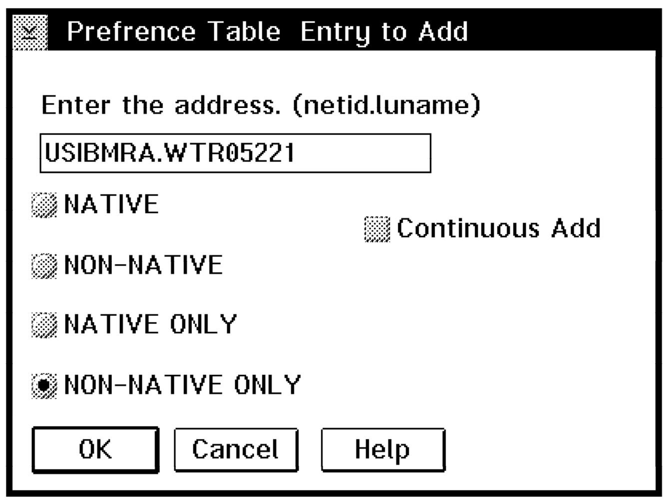


Figure 61. Adding a Routing Preference to the Table

Select **Exit** to end LUTPM.

7.4 Starting and Stopping AnyNet/2 SNA over TCP/IP

AnyNet/2 SNA over TCP/IP will be automatically loaded into memory when Communications Manager is started. It will wait a specified amount of time to wait for TCP/IP to become active if it has not already been started. We suggest that you start TCP/IP for OS/2 first, and then IBM Communications Manager/2 which will also start SNA over TCP/IP.

To confirm that SNA over TCP/IP is running properly, use NETSTAT command with the -s option to display the status of currently running sockets applications. NETSTAT is provided with the TCP/IP for OS/2 program.

Since SNA over TCP/IP is a socket application, you should see a result similar to the following, provided that no other socket applications are running:

```
[C:]netstat -s
SOCK      TYPE         FOREIGN          LOCAL         FOREIGN      STATE
                          PORT           PORT            HOST
==== ==========      ==========      ==========      ========== ==========
25       DGRAM               0      mptn..397         0.0.0.0   UDP
24       DGRAM               0              0         0.0.0.0   UDP
23      STREAM            1031           1030       127.0.0.1   ESTABLISHED
22      STREAM            1030           1031       127.0.0.1   ESTABLISHED
20      STREAM            1029           1028       127.0.0.1   ESTABLISHED
19      STREAM            1028           1029       127.0.0.1   ESTABLISHED
17      STREAM               0      mptn..397         0.0.0.0   LISTEN
```

NETSTAT displays a state of LISTEN for port 397 which is the default port assignment for SNA over TCP/IP. This port has also been opened for UDP. Two MPTN connections are established to the node itself for internal purposes.

You can determine whether a routing preference table for SNA over TCP/IP has been established successfully by running the LULIST utility together with the p option to print the contents of the table:

```
[C:]lulist p
Default is  NATIVE
Luname              Flag
--------------------------------
USIBMRA.WTR05221        NON-NATIVE_ONLY
```

Default is NATIVE means that IBM Communications Manager/2 will route all LU session requests to the SNA network which is its native network.

7.5 Preparing TCP/IP for OS/2 for SNA over TCP/IP Applications

In order to be able to run SNA applications over an IP network using AnyNet/2 SNA over TCP/IP, you must have TCP/IP for OS/2 installed and running on your system. We also suggest that you start TCP/IP for OS/2 before you start Communications Manager.

If your system is attached to the IP network via LAN or serial line, all that is required is the Base Kit of the OS/2 TCP/IP product.

If your system is attached to the IP network via X.25, the Extended Networking Kit for the OS/2 TCP/IP product is also required.

Note: An attachment via SNALink would also require the Extended Networking Kit (Version 2.0 only), but such a connection would not make much sense in this configuration.

You need at least one IP interface configured for TCP/IP for OS/2 to support the native IP part of the SNA over TCP/IP MPTN access node. You also need to configure system parameters for your TCP/IP host as determined by your TCP/IP environment, such as:

- Host name
- Domain name
- Name server address

- Default router address
- SNMP community

and whatever else you require.

Please refer to the scenarios later in this documentation for further details on specific TCP/IP for OS/2 definitions.

For more details on setting up and configuring TCP/IP for OS/2, if you are using IBM TCP/IP Version 2 for OS/2, please refer to the following publications:

- *IBM TCP/IP Version 2 for OS/2: User's Guide*
- *IBM TCP/IP Version 2 for OS/2: Extended Networking Guide*
- *TCP/IP V2.0 for OS/2: Installation and Interoperability*

7.6 Preparing OS/2 Communications Manager for SNA over TCP/IP Applications

To support AnyNet/2 SNA over TCP/IP, you have to install the AnyNet/2 Support feature of IBM Communications Manager/2 Version 1.11. See 7.1, "Installing AnyNet/2 SNA over TCP/IP" on page 94 for details on how to install that feature.

7.6.1 Preparing for Independent LU Support

To support applications using independent LU6.2, you need to make the following definitions to Communications Manager, according to the particular application requirements:

- Local LU(s)
- Mode name(s)
- Transaction program(s)
- Conversation security

You do not need to define the following if you want to establish a connection via SNA over TCP/IP:

- Link(s)
- Partner LU(s)

The partner LUs to be used with SNA over TCP/IP are defined in the routing preference table. The link for those partner LUs will be over TCP/IP.

You only need to have links defined for systems you can reach either via native SNA only, or via native SNA as well as via TCP/IP.

7.6.2 Preparing for Dependent LU Support

In order to support dependent LU communications for SNA over TCP/IP, an OS/2 Communications Manager node must be defined as a dependent LU requester (DLUR), and the VTAM it is attached to must be defined as a dependent LU server (DLUS). Refer to 5.8, "Dependent LU Requestor/Server" on page 75 for an overview of the DLUS and DLUR functions, and to *APPN Architecture and Product Implementations Tutorial* (GG24-3669) for more detailed information. Also refer to *VTAM Resource Definition Reference Version 4 Release 2 for MVS/ESA* (SC31-6498) and *VTAM Network Implementation Guide Version 4 Release 2 for MVS/ESA* (SC31-6494) for more information on how to define DLUR and DLUS.

There is no difference to the application whether it runs on native SNA or on SNA over TCP/IP.

7.7 Defining SNA over TCP/IP Environment Variables (OS/2)

The operation of AnyNet/2 SNA over TCP/IP is controlled by OS/2 environment variables which are defined in the CONFIG.SYS file. After installation, those variables will be set to default values. You can change them either by using the AnyNet/2 Configuration Notebook, or by manually editing the CONFIG.SYS file.

The following variables relate to settings in the Configuration Notebook discussed in more detail in 7.3, "Configuring AnyNet/2 SNA over TCP/IP" on page 98:

```
SET SNASUFFIX=IBM.COM
SET CRITICAL_WS=NO
SET CONNWAIT_SECS=30
SET CONN_RETRY_SECS=300
SET MPTN_WELL_KNOWN_PORT=397
SET TCPWAIT_MINS=15
SET UNACKED_DG_RETRY_SECS=30
SET UNSENT_DG_RETRY_SECS=3
SET INACTIVITY_TIMER_SECS=90
```

The following environment variable will additionally be defined by AnyNet/2 SNA over TCP/IP:

```
SET ANYNETPATH=C:ANYNET
```

7.8 How to Map LU Names to IP Addresses

This section describes how to ensure that an SNA LU name and SNA network ID are mapped to a unique IP address.

For an outbound flow (from SNA network to IP network), the mapping process is done in the following logical order:

1. Take an SNA fully-qualified LU name
2. Add the SNA domain name suffix to form a TCP/IP fully qualified domain name
3. Find the IP addresses associated with that name

SNA Fully Qualified LU Names

The SNA network knows the two communicating LUs by their network ID and LU names in the form:

NETID.LUNAME

SNA fully qualified LU names are formatted in decreasing scope (NETID comes first).

TCP/IP Fully Qualified Domain Names

SNA LU names are reformatted by SNA over TCP/IP into IP domain names in the form:

LUNAME.NETID.SNASUFFIX

TCP/IP domain names are formatted in increasing scope (LU name comes first).

The SNASUFFIX is the domain name suffix. It ensures that any SNA LU name and SNA network ID specified in the form LUNAME.NETID is distinct from any existing IP domain name.

SNASUFFIX is one of the environment variables you have to define in your CONFIG.SYS file. It is equivalent to the DNSUFX operand of the VTAM TCP/IP major node.

Finding Associated IP Addresses
The IP network knows these LUs by their IP addresses. SNA over TCP/IP processes the mapping from the TCP/IP fully-qualified domain name to an IP address by using one of the following:

- A local HOSTS file
- A Domain Name Server

To define the domain names, you can use a ETC\HOSTS file, a domain name server or both.

Using a Local HOSTS File
You can use the HOSTS file to define domain names and IP addresses for your local network. However, if your IP network interconnects to other IP networks, we recommend that you use a domain name server instead, to ensure the following:

- Local names and addresses are consistent with the names and addresses established for other networks.
- You do not have to maintain large numbers of addresses in every HOSTS file.

Using a Domain Name Server
You do not need to define the domain name associated with the LU in a particular DNS. However, you must define the LU name in a DNS that can be reached from the node originating the LU-LU session over the IP network.

Please see 2.3.6.3, "DNS (Domain Name System)" on page 47 for more information about the Domain Name System.

7.9 Using the Routing Preference Table

When you first install and configure SNA over TCP/IP, it creates the routing preference table, LUTAB.LST, and defines one entry (Default is NATIVE) which is the network to be used for all unspecified LU names. Initially the default is set to the native routing which means SNA applications will be directed to SNA transport. You must add LU names to the table before you attempt to set up connections with partner LUs.

When a session is initiated, SNA over TCP/IP first determines which transport network to use by querying the routing preference table. It then tries to establish the requested session in the following order:

1. Through the network specified by the routing preference table
2. Through the alternative network unless it is native only or nonnative only

Note: The LUTAB.LST file will be placed in the directory where Communications Manager has been installed.

There are several ways to add, change or remove routing preference table entries:

- Using the LULIST command
- Using the LUTPM tool
- Editing the LUTAB.LST file
- Write a procedure using APPC verbs

7.9.1.1 Using the LULIST.EXE

If you enter the command `lulist` or `lulist -h`, you will get the following output showing you the LULIST command syntax:

```
[C:]lulist
usage: lulist {a|r|l|p|f|c|d|u|h} argument(s).
    Arguments by function:.
    a netid.luname flag ( ADD LUNAME    ).
    r netid.luname      ( REMOVE LUNAME ).
    l netid.luname      ( LOOKUP LUNAME ).
    p                   ( PRINT TABLE   ).
    f                   ( FLUSH TABLE   ).
    c netid.luname flag ( CHANGE LUNAME ).
    d                   ( PRINT DEFAULT ).
    d flag              ( SET DEFAULT   ).
    u                   ( UPDATE TABLE  ).
    h                   ( HELP          ).
flag: 0=Native, 1=Non-Native, 2=Native Only, 3=Non-Native Only.
```

Note: The flush option erases all entries in both the active version in memory and the LUTAB.LST file; it also resets the default network to NATIVE (SNA).

For example, adding an entry to the LUTAB.LST in order to establish sessions from your workstation and RAIAC in net USIBMRA, using the IP network you will obtain:

```
[C:]lulist a usibmra.raiac 1
Luname usibmra.raiac added to table.
```

To check that this remote ILU has been properly set to NON-NATIVE network (argument = 1), you use the print option, which gives you:

```
[C:]lulist p
Default is NATIVE

Luname               Flag
--------------------------------
USIBMRA.RAIAC    NON-NATIVE
```

The LULIST output shows which network will be used to access the listed LUs, and we correctly get an entry showing that to access RAIAC we will use the NON-NATIVE (TCP/IP) network.

7.9.1.2 Using the LUTPM.EXE

LUTPM is an OS/2 Presentation Manager application that allows you to modify the routing preference table for SNA over TCP/IP LUTPM offers you the same functions as LULIST.EXE but provides a user-friendly interface. Figure 60 on page 102 shows an example of LUTPM.

7.9.1.3 Editing the LUTAB.LST

This method may be preferable if you want to manage a lot of entries at one time.

Note: When you modify the table using an editor, you have to run the `LULIST U` (update) command in order for your changes to take effect.

7.9.1.4 Using APPC Verbs

You can invoke an APPC verb coding procedure to add, remove or look up one or more LU names. The *APPC over TCP/IP User's Guide* describes the C structure used to enable an APPC application to manage entries in the routing preference table.

7.10 Using SNA Applications with SNA over TCP/IP

This section describes how to monitor SNA over TCP/IP and employ it to run SNA applications.

7.10.1 Configuring and Starting SNA Applications

To configure and start applications for use with SNA over TCP/IP, you have to perform the same procedures as you would have when using those applications over native SNA transport. The only difference is that link statement for Communications Manager is not required for applications that use TCP/IP transport only. This definition is not necessary for SNA over TCP/IP.

Please see the test scenarios later in this document for sample applications that have been tested with SNA over TCP/IP.

Please see Appendix B, "System Definitions - OS/2" on page 217 for system definitions that have been made for any of the tested applications.

7.10.2 Monitoring SNA over TCP/IP

You may obtain information related to your SNA over TCP/IP environment by using one of the following:

- Communications Manager Subsystem Management
- The ABINFO mapping utility

To display active LU sessions and active LU6.2 transaction programs, you can use the Communications Manager Subsystem Management.

ABINFO.EXE is a utility provided by SNA over TCP/IP. It enables you to determine:

- The remote IP addresses and LU names associated with an LU session

- The local LU names of associated active sessions for a local IP address

If you enter the command abinfo or abinfo h, you will get the ABINFO command syntax:

```
[C:]abinfo
Usage: abinfo {a2b|b2a|help} argument(s).
     Arguments by function:
     a[2b]   [Protocol_Name Attribute(s)]
     b[2a]   [Protocol_Name Attribute(s)]
     h[help]
```

In the a2b or b2a operand, the a stands for transport user names, in our case SNA names, and b stands for transport provider addresses, in our case IP network addresses.

The possible commands are:

```
abinfo a2b SNA pcid

abinfo b2a INET ip_address
```

Mapping For a Given APPC Session

To obtain the remote Internet addresses and LU names associated with a given session, use the following command:

```
[C:]abinfo a2b SNA d96fd635f5b8fb6f
************** LOCAL A NAME   ************
NSAP = USIBMRA.WTR05221
************** PARTNER A NAME ************
NSAP = USIBMRA.RAB
************** LOCAL B NAME   ************
NSAP = 9.67.38.36
TSEL = 1152
************** PARTNER B NAME ************
NSAP = 9.67.38.11
TSEL = 397
```

Using the command a2b, which should be read ″mapping A LU names to B IP addresses″, you can only specify SNA and the pcid as arguments. You can get the pcid by using the PMDSPLAY command. The command output shows that a local LU, named USIBMRA.WTR05221 with an IP address of 9.67.38.36, uses port number 1152 for an IP connection with a remote LU, named USIBMRA.RAB with an IP address of 9.67.38.11, which uses port number 397.

Mapping For a Local Internet Address

To obtain the local LU names of associate active sessions for a local Internet address, use the following command:

```
[D:]abinfo b2a INET 9.67.38.36
NSAP = USIBMRA.WTR05221
```

Using this command b2a, which should be read ″mapping B IP addresses to A LU names″, you can only specify INET and the ip_address as arguments.

7.11 Debugging Problems Related to SNA over TCP/IP SNA over TCP/IP

Diagnostic information is recorded in the Communications Manager/2 log and error log. Some SNA over TCP/IP error messages generate either conditional or unconditional alerts. See CRITICAL_WS environment variable in 7.7, "Defining SNA over TCP/IP Environment Variables (OS/2)" on page 106 for alerts handling.

To start tracing SNA over TCP/IP data, use the Communications Manager/2 Problem Determination Aids Trace Services. From the trace panel, specify:

- The MPTNTCP_DATA option to capture inbound and outbound data
- The MPTNTCP_INT option to trace internal calls and socket calls to TCP

Traces may also be started by the following method.

At the command prompt of an OS/2 window, type the following command to start the trace:

```
[C:]cmtrace start /dlc mptn /event 32
```

To stop tracing SNA over TCP/IP enter the following command:

```
[C:]cmtrace stop
```

To copy the CMTRACE output to a file named for example MPTNINT.TRC, enter the following command:

```
[C:]cmtrace copy c:mptnint.trc
```

If you use the FMTTRACE to format the CMTRACE output, only the socket calls (MPTNTCP_DATA option) are formatted. To view all the MPTN_DATA trace look at the raw data file produced by CMTRACE. FMTTRACE provides a summary output named *file_name.SUM*, and a detailed output named *file_name.DET*.

Important

When you are using IBM Communications Manager/2 Version 1.11, you need a fix for APAR JR08099 that contains the MPTN trace functions for that version of IBM Communications Manager/2.

Please see *IBM Communications Manager/2 Version 1.1 Network Administration and Subsystem Management Guide* for more information about IBM Communications Manager/2 traces.

Chapter 8. AnyNet/400 APPC over TCP/IP

This chapter is an introduction to APPC over TCP/IP for the AS/400 platform.

With AnyNet/400 APPC over TCP/IP, TCP/IP users can gain access to APPC or CPI-C applications without adding a separate SNA network. Windows and OS/2 Client Access/400 also support AnyNet allowing these applications to be used across TCP/IP networks. The required portion of AnyNet APPC over TCP/IP is shipped as part of the Client Access/400 product and downloaded to the workstation as part of the installation of Client Access/400.

8.1 System Software Required to use APPC over TCP/IP (OS/400)

The following minimum software level is required to use APPC over TCP/IP on the AS/400 platform:

- OS/400 Version 3 Release 1.

  AnyNet/400 APPC over TCP/IP and Sockets over SNA are shipped with the base OS/400 V3R1 operating system. No additional software is required.

8.2 Configuring AnyNet/400 APPC over TCP/IP

In order to run APPC over TCP/IP on your AS/400, the following OS/400 configuration steps are required:

1. Establish a TCP/IP connection between the systems.
2. Change the Network Attribute ALWANYNET to *YES.
3. Create an APPC controller with LINKTYPE(*ANYNW).
4. Add entries to the APPN remote location list.
5. Map APPC LU name to an internet address.

The User ID, under which the APPC over TCP/IP configuration is created, must have sufficient authority to access the relevant commands. Some of the commands require the user ID to have the IOSYSCFG authority. The examples shown here were created using a profile with QSECOFR authority.

1. **Establish a TCP/IP connection between the systems**

   This step will show the basic steps to establishing a TCP/IP connection between two systems. If your system already has a TCP/IP connection to the remote system with which you want to communicate via APPC over TCP/IP, then you can skip this step and proceed to step 2 in this section.

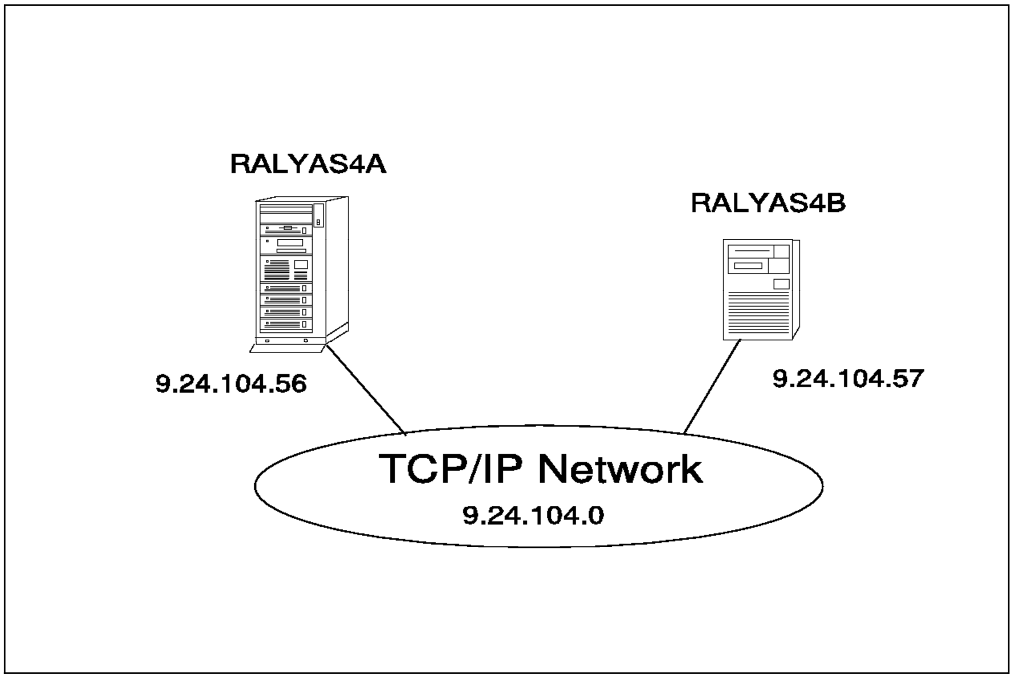


Figure 62. Two Systems Connected Using TCP/IP

In the following panels we will create the TCP/IP configuration for RALYAS4A in the diagram above. The configuration steps for RALYAS4B would be the same using the different adapter (LAN) address, internet address and host name.

The following panels show the configuration screens for a token-ring connection. If you require help in establishing a TCP/IP connection over another type of interface, refer to the manual *AS/400 TCP/IP Configuration and Reference* SC41-3420.

The AS/400 line description defines the physical interface to the network. If an appropriate line description does not already exist (they can be shared), you need to create one. Here we use the CRTLINTRN command to create a token-ring line description.

```
                   Create Line Desc (Token-Ring) (CRTLINTRN)

Type choices, press Enter.

Line description . . . . . . . . > TRNLINE       Name
Resource name  . . . . . . . . . > LIN021        Name, *NWID, *NWSD
Online at IPL  . . . . . . . . .   *YES          *YES, *NO
Vary on wait . . . . . . . . . .   *NOWAIT       *NOWAIT, 15-180 (1 second)
Maximum controllers  . . . . . .   40            1-256
Line speed . . . . . . . . . . .   4M            4M, 16M, *NWI
Maximum frame size . . . . . . .   1994          265-16393, 265, 521, 1033...
Local adapter address  . . . . . > 400000000044  400000000000-7FFFFFFFFFFF...
Exchange identifier  . . . . . .   *SYSGEN       05600000-056FFFFF, *SYSGEN
SSAP list:
  Source service access point  .   *SYSGEN       02-FE, *SYSGEN
  SSAP maximum frame . . . . . .                 *MAXFRAME, 265-16393
  SSAP type  . . . . . . . . . .                 *CALC, *NONSNA, *SNA, *HPR
               + for more values
Text 'description' . . . . . . . > 'Token-ring for Anynet/400'

                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 63. AS/400 Create Token-Ring Line Description - System RALYAS4A

For a TCP/IP configuration, there is no need to create controller and device descriptions, they are automatically created when TCP/IP first uses the token-ring line.

TCP/IP Interface

Enter the CFGTCP command to access the Configure TCP/IP panel and take option 1 to work with TCP/IP interfaces.

```
CFGTCP                         Configure TCP/IP
                                                          System:   RALYAS4A
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change local domain and host names
    13. Change remote name server

    20. Configure TCP/IP applications
    21. Configure related tables


Selection or command
===> 1

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 64. AS/400 TCP/IP Configuration Menu

```
                         Work with TCP/IP Interfaces
                                                          System:   RALYAS4A
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   9=Start   10=End

     Internet         Subnet             Line      Line
Opt  Address          Mask               Description  Type

__   _______________
__   9.24.104.56      255.255.255.0      L41TR        *TRLAN
__   127.0.0.1        255.0.0.0          *LOOPBACK    *NONE










                                                                      Bottom
F3=Exit   F5=Refresh   F6=Print list   F10=Work with IP over SNA interfaces
F11=Display interface status           F12=Cancel   F17=Top   F18=Bottom
```

Figure 65. AS/400 TCP/IP Interface Definition - System RALYAS4A

The TCP/IP interface defines this AS/400 on the TCP/IP network. If one does not already exist, add an entry using the internet address allocated to this system and the mask of the subnet in which the system resides.

Besides allowing you to add, change and remove TCP/IP interfaces, this screen also allows you to start and end these interfaces.

TCP/IP Route

If the route to the remote host is via a gateway or resides in a different network or subnetwork, it will be necessary to use option 2 from the Configure TCP/IP screen to configure the route. This is not the case in this simple scenario.

TCP/IP Host table

The local host table on the AS/400 contains a list of the internet addresses and associated host names for this network. To access the AS/400 host table enter the CFGTCP command and take option 10 (Work with TCP/IP Host Table Entries).

```
                     Work with TCP/IP Host Table Entries
                                                            System:   RALYAS4A
 Type options, press Enter.
   1=Add   2=Change   4=Remove   5=Display   7=Rename

      Internet         Host
 Opt  Address          Name
  _   _______________
  _   9.24.104.56      RALYAS4A
                       RALYAS4A.ITSO.RAL.IBM.COM
  _   9.24.104.57      RALYAS4B
                       RALYAS4B.ITSO.RAL.IBM.COM







 F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 66. AS/400 TCP/IP Host Table Entries - System RALYAS4A

Unless you are planning to use a name server, add an entry for the local system and any remote system(s) to which TCP/IP is to be used. In the above example, both the short and long names have been entered.

2. **Change the Network Attribute ALWANYNET to *YES**

Now we start the AnyNet specific configuration steps. Changing this attribute will allow both APPC over TCP/IP and Sockets over SNA support to run on your system. The default for this value, when V3R1 is initially installed, is *NO. Use the DSPNETA command see what your system is set to. If it is set to *NO, use the command:

CHGNETA ALWANYNET (*YES)

After changing this attribute, you can verify the change by entering the DSPNETA command. The resulting displays are shown in the following figure.

```
                         Display Network Attributes
                                                      System:   RALYAS4A
Current system name . . . . . . . . . . . . . . :   RALYAS4A
  Pending system name . . . . . . . . . . . . . :
Local network ID . . . . . . . . . . . . . . . . :   USIBMRA
Local control point name . . . . . . . . . . . . :   RALYAS4A
Default local location . . . . . . . . . . . . . :   RALYAS4A
Default mode . . . . . . . . . . . . . . . . . . :   BLANK
APPN node type . . . . . . . . . . . . . . . . . :   *NETNODE
Data compression . . . . . . . . . . . . . . . . :   *NONE
Intermediate data compression . . . . . . . . . :   *NONE
Maximum number of intermediate sessions . . . . :   200
Route addition resistance . . . . . . . . . . . :   128
Server network ID/control point name . . . . . . :   *LCLNETID   *ANY


                                                                  More...
```

```
                         Display Network Attributes
                                                      System:   RALYAS4A
Alert status . . . . . . . . . . . . . . . . . . :   *ON
Alert logging status . . . . . . . . . . . . . . :   *ALL
Alert primary focal point . . . . . . . . . . . :   *YES
Alert default focal point . . . . . . . . . . . :   *NO
Alert backup focal point . . . . . . . . . . . . :
  Network ID . . . . . . . . . . . . . . . . . . :     *NONE
Alert focal point to request . . . . . . . . . . :   RAK
  Network ID . . . . . . . . . . . . . . . . . . :     USIBMRA
Alert controller description . . . . . . . . . . :   *NONE
Alert hold count . . . . . . . . . . . . . . . . :   0
Alert filter . . . . . . . . . . . . . . . . . . :   AS400NET
  Library . . . . . . . . . . . . . . . . . . . :     QALSNDA
Message queue . . . . . . . . . . . . . . . . . :   QSYSOPR
  Library . . . . . . . . . . . . . . . . . . . :     QSYS
Output queue . . . . . . . . . . . . . . . . . . :   QPRINT
  Library . . . . . . . . . . . . . . . . . . . :     QGPL
Job action . . . . . . . . . . . . . . . . . . . :   *FILE

                                                                  More...
```

```
                         Display Network Attributes
                                                      System:   RALYAS4A
Maximum hop count . . . . . . . . . . . . . . . :   16
DDM request access . . . . . . . . . . . . . . . :   *OBJAUT
Client request access . . . . . . . . . . . . . :   *OBJAUT
Default ISDN network type . . . . . . . . . . . :
Default ISDN connection list . . . . . . . . . . :   QDCCNNLANY
Allow ANYNET support . . . . . . . . . . . . . . :   *YES
Network Server Domain . . . . . . . . . . . . . :   RALYAS4A


                                                                 Bottom
Press Enter to continue.

F3=Exit   F12=Cancel
```

*Figure 67. AS/400 Display of Network Attributes with ALWANYNET(*YES)*

Changing the ALWANYNET network attribute to *YES will result in the APPC over TCP/IP job (QAPPCTCP) being started in the QSYSWRK subsystem.

3. **Create an APPC controller with LINKTYPE(*ANYNW)**

   The AS/400 controller description defines the remote system. A new LINKTYPE has been added to the APPC controller description for APPC over TCP/IP. With APPC over TCP/IP, the APPC controller is no longer directly attached to a line description. Use the CRTCTLAPPC (Create APPC Controller Description) command to create an APPC controller with LINKTYPE(*ANYNW).

```
                      Create Ctl Desc (APPC) (CRTCTLAPPC)

 Type choices, press Enter.

 Controller description . . . . . > ANYNWAS4B     Name
 Link type  . . . . . . . . . . . > *ANYNW        *ANYNW, *FAX, *FR, *IDLC...
 Online at IPL  . . . . . . . . .   *YES          *YES, *NO
 Remote network identifier  . . .   *NETATR       Name, *NETATR, *NONE, *ANY
 Remote control point . . . . . . > AS4BANYT      Name, *ANY
 User-defined 1 . . . . . . . . .   *LIND         0-255, *LIND
 User-defined 2 . . . . . . . . .   *LIND         0-255, *LIND
 User-defined 3 . . . . . . . . .   *LIND         0-255, *LIND
 Text 'description' . . . . . . . > 'RALYAS4B via AnyNet/400'



                                                                       Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys
```

*Figure 68. AS/400 Create Controller Description with LINKTYPE(*ANYNW)*

   The remote network identifier should match the local network identifier on the remote system, *NETATR indicates that the value in the network attributes should be used, that the local system and remote system have the same network ID. The remote control point name, however, is not used external to the system.

   APPC Device Description and Mode Description

   The APPC device description will be automatically created when the above controller is activated.

   APPC over TCP/IP uses mode descriptions in the same way as APPC over SNA does.

   Note: It is *not* possible to map an APPC mode to an IP type of service.

4. **Add entries to the APPN remote location list**

   To communicate using APPC over TCP/IP support, the system requires a configuration list entry for each remote location. APPC over TCP/IP communications needs the information in the APPN remote location list to determine which controller description to use when it activates the session. Furthermore, the entry allows the AS/400 system to automatically configure the APPC device description. To update the APPN remote configuration list, use the following command:

```
CHGCFGL *APPNRMT
```

```
                         Change Configuration List                  RALYAS4A
                                                           11/10/94  10:47:23
Configuration list  . . :   QAPPNRMT
Configuration list type :   *APPNRMT
Text  . . . . . . . . . :

Type changes, press Enter.

--------------------------APPN Remote Locations---------------------------
          Remote                Remote     Control
Remote    Network    Local      Control    Point      Location          Secure
Location  ID         Location   Point      Net ID     Password           Loc
RALYAS4B  *NETATR    *NETATR    AS4BANYT   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
________  *NETATR    *NETATR    ________   *NETATR    _______________    *NO
                                                                       More...
F3=Exit   F11=Display session information   F12=Cancel   F17=Top   F18=Bottom
```

Figure 69. AS/400 APPN Remote Location List Panel

APPN requires that all remote location names be unique. Thus, it can not have the same remote location name in both its SNA network and its TCP/IP network.

The remote control point name in the configuration list entry of the APPN remote location list must match the remote control point name of an APPC controller with a LINKTYPE(*ANYNW).

5. **Map APPC LU name to an internet address**

   The TCP/IP host table provides the mapping between host name and internet address. Here it is providing the mapping between the SNA remote location name/remote network ID and the remote internet address. Enter the CFGTCP command to access the Configure TCP/IP panel, and take option 10 to work with the TCP/IP host table.

```
                    Work with TCP/IP Host Table Entries
                                                          System:   RALYAS4A
 Type options, press Enter.
   1=Add   2=Change   4=Remove   5=Display   7=Rename

      Internet         Host
 Opt  Address          Name

  _   _______________
  _   9.24.104.56      RALYAS4A
                       RALYAS4A.ITSO.RAL.IBM.COM
                       RALYAS4A.USIBMRA.SNA.IBM.COM
  _   9.24.104.57      RALYAS4B
                       RALYAS4B.ITSO.RAL.IBM.COM
                       RALYAS4B.USIBMRA.SNA.IBM.COM




 F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 70. AS/400 TCP/IP Host Table Entries

For APPC over TCP/IP, the host name entries added are in the following three parts:

- **RALYAS4B** - remote SNA location (LU) name
- **USIBMRA** - remote SNA network ID
- **SNA.IBM.COM** - SNA Domain Name Suffix

Add an entry using the remote system internet address. The remote SNA location name and SNA network ID should be as specified in the APPN remote location list.

You will need to add entries for both of the following:

- The local system
- Any remote systems you require to communicate with using APPC over TCP/IP

Note:

A PTF is now available to allow the AS/400 to use an SNA suffix of other than SNA.IBM.COM. The PTF is shipped in two parts: MF08352 and SF21042. The PTF was not used during our residency.

When communicating between systems using APPC over TCP/IP, both systems must use the same SNA Domain Name Suffix.

This host table will be used by native TCP/IP and APPC over TCP/IP. The entries *without extension* SNA.IBM.COM are for native TCP/IP.

With all of the configuration steps completed, you are now ready to use the APPC over TCP/IP support of AnyNet/400.

Part 3. Working Scenarios

Chapter 9. SNA over TCP/IP Working Scenarios

This chapter presents the seven test scenarios of SNA over TCP/IP conducted at the International Technical Support Organization. This information will be presented in the following order:

- Testing Environment
- Scenario 1: Host to Host
- Scenario 2: Gateway Support
- Scenario 3: Chained Gateways
- Scenario 4: From Host to OS/2
- Scenario 5: Host Gateway to OS/2
- Scenario 6: OS/2 to OS/2
- Scenario 7: Dependent LU Support
- Scenario 8: AnyNet/400 APPC over TCP/IP

We will use the following shortened designations:

Native Node: VTAM subarea host or OS/2 system acting as a native SNA node

MVS Access Node: VTAM 4.2 acting as an MPTN access node

OS/2 Access Node: OS/2 acting as an MPTN access node

Gateway Node: VTAM 4.2 acting as an MPTN gateway node

LU2: Dependent LU Type 2

(SNA): SNA network

(IP): IP network

9.1 Testing Environment

We defined and configured two SNA over TCP/IP (MVS/ESA) systems following the logical steps illustrated in 6.3, “VTAM Definitions” on page 84. Similarly, we defined and configured two SNA over TCP/IP (OS/2) systems as illustrated 7.7, “Defining SNA over TCP/IP Environment Variables (OS/2)” on page 106. Figure 71 on page 126 shows all the resources used to run our tests.

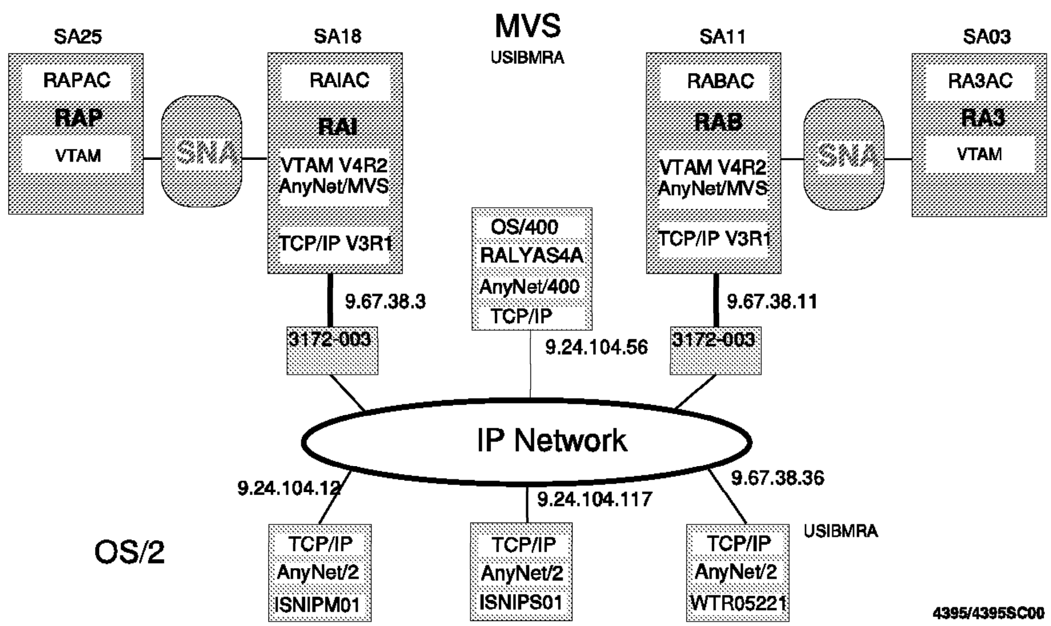


Figure 71. SNA over TCP/IP Test Configuration

- RA3, RAB, RAI and RAP are the SSCP names for SA03, SA11, SA18 and SA25 respectively.
- RA3AC, RABAC, RAIAC and RAPAC are CICS applications in SA03, SA11, SA18 and SA25, respectively.
- RABAC, RAPAC, are APING applications in SA11, and SA25, respectively (not shown here).
- RABAT is a TSO applications in SA11 (not shown here).
- ISNIPM01, ISNIPS01 and WTR05221 are the CP names on the OS/2 systems.
- USIBMRA is the netID of all test systems. .

The configuration data and definitions used are provided in Appendix A, “System Definitions - MVS” on page 181 and Appendix B, “System Definitions - OS/2” on page 217.

The following naming conventions were used:

```
Resource name:
     R A x y cccc
         │ │ └┬┘
         │ │  └─────────── variable (SNIP - SNA over TCP/IP)
         │ │
         │ └────────────── NODE TYPE: B for TCP major nodes
         │                            R for CDRSC major nodes
         │                            G for GROUP definition statements
         │                            L for LINE definition statements
         │                            P for PU definition statements
         │                            S for SWNET major nodes
         │
         └──────────────── Alphabetical representation of the Subarea Number
                                      3 for subarea 03
                                      B for subarea 11
                                      I for subarea 18
                                      P for subarea 25
```

Figure 72. SNA over TCP/IP - VTAM Naming Conventions Used in the Tested Scenarios

The CDRSCs minor nodes must have the name of the remote LU, so in this case the naming convention shown in Figure 72 was not followed.

Note!

In Scenario 6 (OS/2 to OS/2) a Domain Name Suffix of ANYNET.IBM.COM was used and a Domain Name Server was used to resolve IP addresses.

In all other scenarios IBM.COM was used as the Domain Name Suffix and a HOSTS.LOCAL file was used to resolve IP addresses.

Chapter 10. Scenario 1 - MVS Access Node to MVS Access Node

As shown in Figure 73, this scenario is between two AnyNet/MVS SNA over TCP/IP systems, SA11 and SA18, via the IP network.

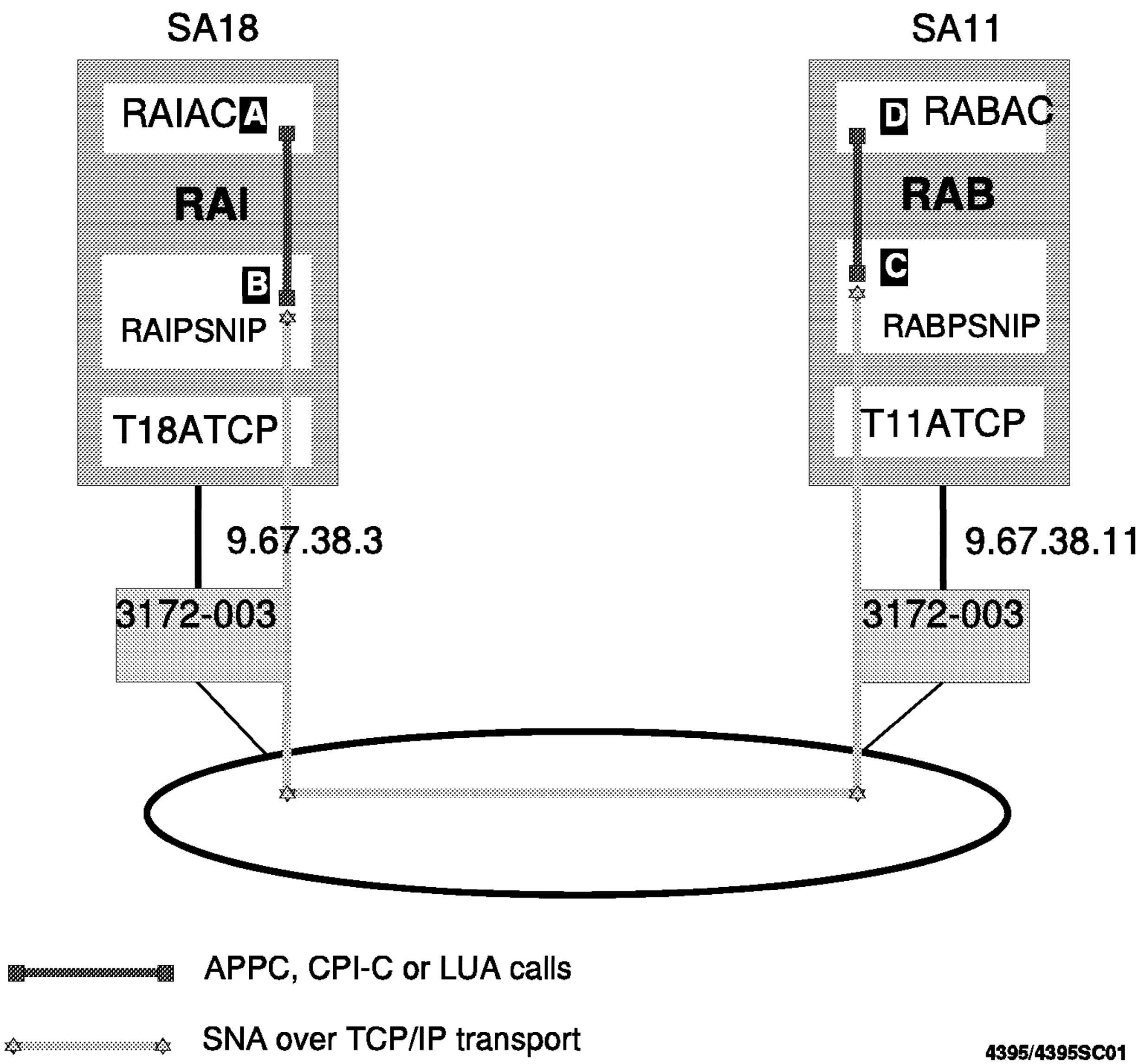


Figure 73. SNA over TCP/IP Test Scenario 1 - MVS-to-MVS

10.1 Scenario 1 - Test Application

In this scenario a CICS - CICS LU6.2 connection was tested. The LU6.2 sessions between RABAC and RAIAC were started by acquiring RABAC from RAIAC.

10.2 Scenario 1 - Procedures

The following procedure was used to test this scenario.

Subarea 11:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RABBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RABRSNIP

Subarea 18:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RAIBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RAIRSNIP

Testing:

- Logon to RAIAC (Subarea 18)
- Enter the following CICS transaction:

  CEMT I CONN and acquire RABAC.

10.3 Scenario 1 - Test Results

The following displays were captured to show the LU 6.2 sessions with these commands.

Subarea 11:

- Display CDRSC of the remote LU:

  D NET,ID=RAIAC,SCOPE=ALL

- Display the TCP PU:

  D NET,ID=RABPSNIP,SCOPE=ALL

```
D NET,ID=RAIAC,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RAIAC, [A] TYPE = CDRSC
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RABRSNIP
IST1184I CPNAME = USIBMRA.RAB - NETSRVR = ***NA***
IST1044I ALSLIST = RABPSNIP [C]
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RABPSNIP
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RABAC    ACTIV-S    F86FE164BF5D0777 0004 0003       USIBMRA [D]
IST635I RABAC    ACTIV-P    F7EFD164B8D7DB2D 0000 0001       USIBMRA
IST635I RABAC    ACTIV-P    F7EFD164B8D7DB2C 0000 0001       USIBMRA
IST635I RABAC    ACTIV-P    F7EFD164B8D7DB2B 0002 0003       USIBMRA
IST314I END
```

Figure 74. SNA over TCP/IP Scenario 1 - Display on SA 11 of the Remote LU RAIAC

From a local LU (RAIAC) **A** we establish LU 6.2 sessions to a remote LU (RABAC) **D** through an ALS PU (RABPSNIP) **C**, displayed in Figure 75.

```
D NET,ID=RABPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABPSNIP, [C] TYPE = PU_T2.1
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RABLSNIP, LINE GROUP = RABGSNIP, MAJNOD = RABBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I RAIAC    ACT/S
IST314I END
```

Figure 75. SNA over TCP/IP Scenario 1 - Display on SA 11 of the Adjacent Link Station PU RABPSNIP

Chapter 11. Scenario 2 - MPTN Gateway and MVS Access Node

As shown in Figure 76, this scenario is between an AnyNet/MVS SNA over TCP/IP system, SA18, and a native SNA MVS system, SA03, running CICS over the IP network and across an MPTN SNA over TCP/IP gateway, SA11.

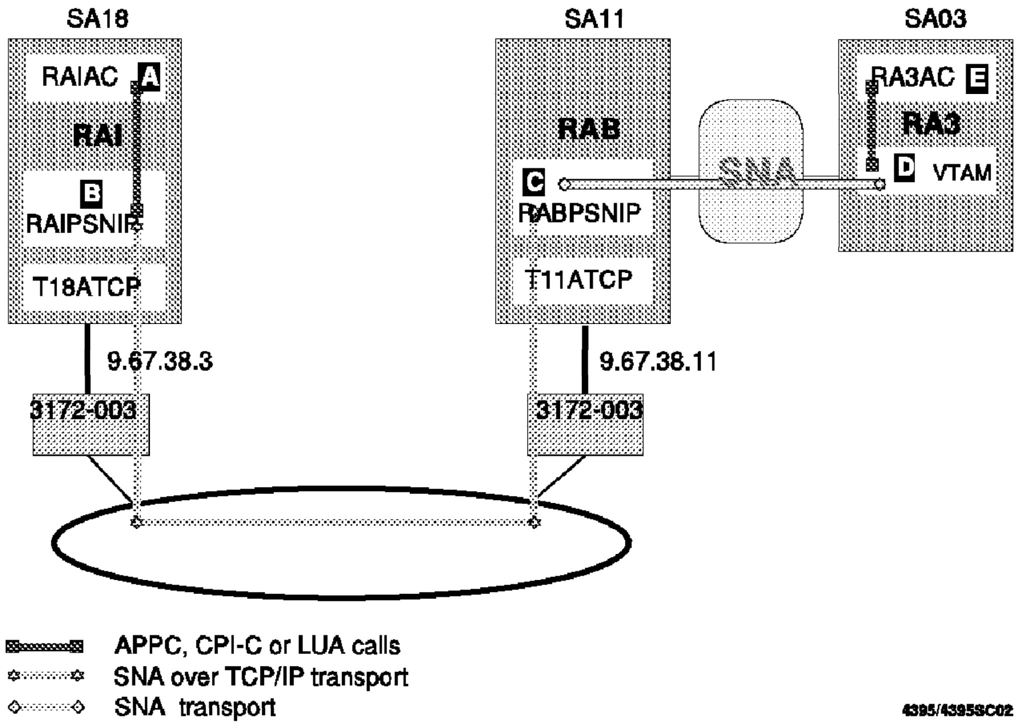


Figure 76. SNA over TCP/IP Test Scenario 2 - MVS-to-MVS via Gateway

11.1 Scenario 2 - Test Application

In this scenario a CICS - CICS LU6.2 connection was tested. The LU6.2 sessions between RA3AC and RAIAC were started by acquiring RA3AC from from RAIAC.

11.2 Scenario 2 - Procedures

The following procedure was used to test this scenario.

Subarea 11:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RABBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RABRSNIP

Subarea 18:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RAIBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RAIRSNIP

Subarea 03:

- Activate the CDRSC major node:

  V NET,ACT,ID=RA3RSNIP

Testing:

- Log on to RAIAC (Subarea 18)
- Enter the following CICS transaction:

  CEMT I CONN and acquire RA3AC.

11.3 Scenario 2 - Test Results

The following displays were captured to show the LU 6.2 sessions with these commands.

Subarea 18:

- Display CDRSC of the remote LU:

  D NET,ID=RA3AC,SCOPE=ALL

- Display the TCP PU:

  D NET,ID=RAIPSNIP,SCOPE=ALL

Subarea 11:

- Display the TCP PU:

  D NET,ID=RAIPSNIP,SCOPE=ALL

Display from Subarea 18

```
D NET,ID=RA3AC,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RA3AC,  E  TYPE = CDRSC
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RAIRSNIP
IST1184I CPNAME = USIBMRA.RAI - NETSRVR = ***NA***
IST1044I ALSLIST = RAIPSNIP  B
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RAIPSNIP
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RAIAC    ACTIV-S    F6FF41648E938A7B 0001 0000       USIBMRA  A
IST635I RAIAC    ACTIV-S    F6FF41648E938A7A 0001 0000       USIBMRA
IST635I RAIAC    ACTIV-S    F6FF41648E938A79 0002 0001       USIBMRA
IST635I RAIAC    ACTIV-P    F86FE164BF5D07EB 0003 0004       USIBMRA
IST314I END
```

Figure 77. SNA over TCP/IP Scenario 2 - Display on SA 18 of the Remote LU RA3AC

From a local LU (RAIAC) **A** LU 6.2 sessions are established to a remote LU (RA3AC) **E** through an ALS PU (RAIPSNIP) **B**, displayed in Figure 78.

```
D NET,ID=RAIPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RAIPSNIP, TYPE = PU_T2.1
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RAILSNIP, LINE GROUP = RAIGSNIP, MAJNOD = RAIBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I RA3AC    ACT/S
IST314I END
```

Figure 78. SNA over TCP/IP Scenario 2 - Display on SA 18 of the Adjacent Link Station RAIPSNIP

Display from Subarea 11

```
D NET,ID=RABPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABPSNIP, C PU_T2.1
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RABLSNIP, LINE GROUP = RABGSNIP, MAJNOD = RABBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I RAIAC A  ACT/S
IST314I END
```

Figure 79. SNA over TCP/IP Scenario 2 - Display on SA 11 of the Adjacent Link Station RABPSNIP

On the MPTN gateway node SA11 (see Figure 79), the sessions coming from the remote LU (RAIAC) **E** go via the ALS PU (RABPSNIP) **C** to reach RA3AC.

Display from Subarea 03

```
D NET,ID=RAIAC,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RAIAC, E TYPE = CDRSC
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RA3RSNIP
IST479I CDRM NAME = RAB, VERIFY OWNER = NO
IST1184I CPNAME = USIBMRA.RAB - NETSRVR = ***NA***
IST082I DEVTYPE = CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RA3AC    ACTIV-S    F86FE164BF5D07EB 0004 0003  0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938A7B 0000 0001  0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938A7A 0000 0001  0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938A79 0001 0002  0  0 USIBMRA
IST314I END
```

Figure 80. SNA over TCP/IP Scenario 2 - Display on SA 03 of the Remote LU RA3AC

Chapter 12. Scenario 3 - Chained MPTN Gateways

As shown in Figure 81, this scenario is between two native SNA MVS systems, SA03 and SA25, running CICS across two MPTN SNA over TCP/IP gateways, SA11 and SA18.

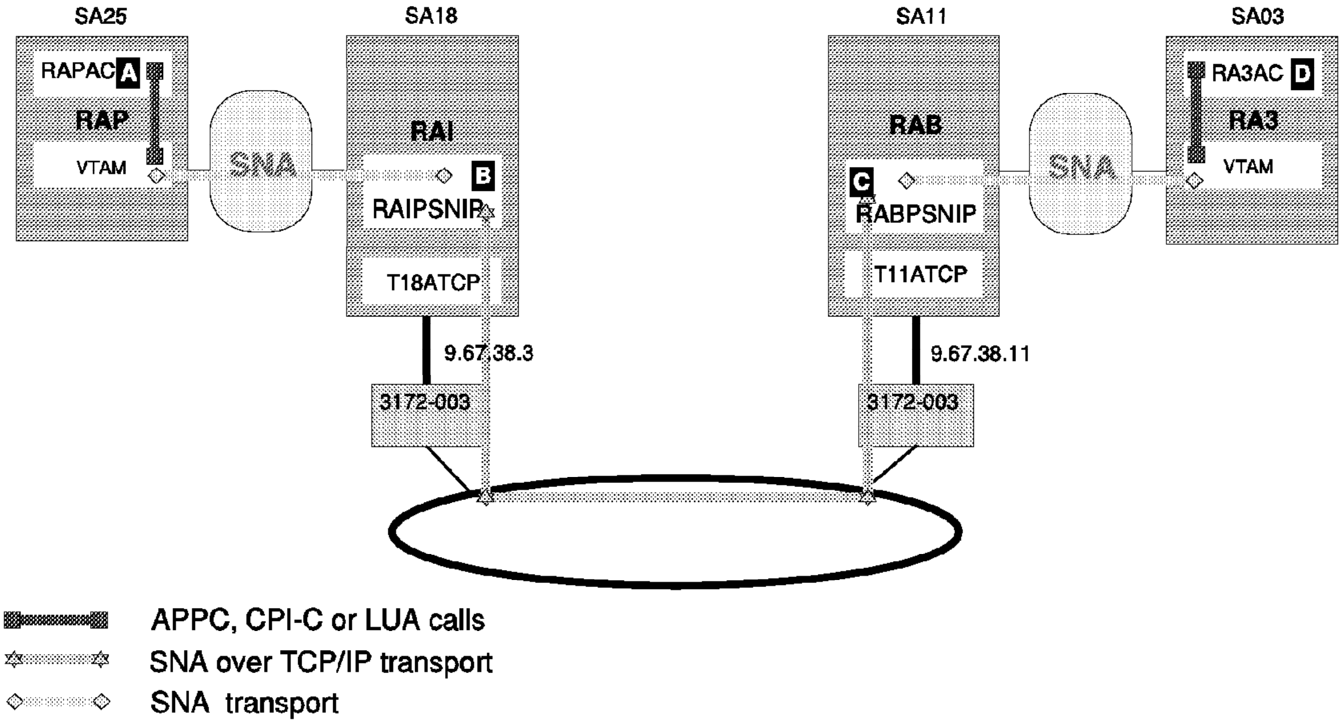


Figure 81. SNA over TCP/IP Test Scenario 3 - MVS-to-MVS via Chained Gateways

12.1 Scenario 3 - Test Application

In this scenario a CICS - CICS LU6.2 connection was tested. The LU6.2 sessions between RAPAC and RA3AC were started by acquirung RA3AC from RAPAC.

12.2 Scenario 3 - Procedures

The following procedure was used to test this scenario.

Subarea 03:

- Activate the CDRSC major node:

  V NET,ACT,ID=RA3RSNIP

Subarea 11:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RABBSNIP

- Activate the CDRSC major node:

V NET,ACT,ID=RABRSNIP

Subarea 18:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RAIBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RAIRSNIP

Subarea 25:

- Activate the CDRSC major node:

  V NET,ACT,ID=RAPRSNIP

Testing:

- Log on to RAPAC (Subarea 25)
- Enter the following CICS transaction:

  CEMT I CONN and acquire RA3AC.

12.3 Scenario 3 - Test Results

The following displays were captured to show the LU 6.2 sessions with these commands.

Subarea 25:

- Display CDRSC of the remote LU:

  D NET,ID=RA3AC,SCOPE=ALL

Subarea 18:

- Display CDRSC of the remote LU:

  D NET,ID=RA3AC,SCOPE=ALL

- Display the TCP PU:

  D NET,ID=RAIPSNIP,SCOPE=ALL

Subarea 11:

- Display the TCP PU:

  D NET,ID=RABPSNIP,SCOPE=ALL

- Display CDRSC of the remote LU:

  D NET,ID=RAPAC,SCOPE=ALL

Subarea 03:

- Display CDRSC of the remote LU:

  D NET,ID=RAPAC,SCOPE=ALL

Display from Subarea 25

```
D NET,ID=RA3AC,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RA3AC, TYPE = CDRSC 062
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RAPRSNIP
IST479I CDRM NAME = RAI, VERIFY OWNER = NO
IST1184I CPNAME = USIBMRA.RAI - NETSRVR = ***NA***
IST1131I DEVICE = ILU/CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RAPAC    ACTIV-S    F6FF41648E938949 0001 0000  0  0 USIBMRA  A
IST635I RAPAC    ACTIV-S    F6FF41648E938948 0001 0000  0  0 USIBMRA
IST635I RAPAC    ACTIV-S    F6FF41648E938947 0003 0002  0  0 USIBMRA
IST635I RAPAC    ACTIV-P    F88F016488B1014A 0002 0003  0  0 USIBMRA
IST924I -------------------------------------------------------------
IST075I NAME = USIBMRA.RA3AC, TYPE = DIRECTORY ENTRY
IST1186I DIRECTORY ENTRY = DYNAMIC LU
IST1184I CPNAME = USIBMRA.RAI - NETSRVR = ***NA***
IST314I END
```

Figure 82. SNA over TCP/IP Scenario 3 - Display on SA 25 of the Remote LU RA3AC

Display from Subarea 18

```
D NET,ID=RA3AC,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RA3AC, TYPE = CDRSC 377
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RAIRSNIP
IST1184I CPNAME = USIBMRA.RAI - NETSRVR = ***NA***
IST1044I ALSLIST = RAIPSNIP
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RAIPSNIP
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RAPAC    ACTIV-S    F6FF41648E938949            0  0 USIBMRA  A
IST635I RAPAC    ACTIV-S    F6FF41648E938948            0  0 USIBMRA
IST635I RAPAC    ACTIV-S    F6FF41648E938947            0  0 USIBMRA
IST635I RAPAC    ACTIV-P    F88F016488B1014A            0  0 USIBMRA
IST314I END
```

Figure 83. SNA over TCP/IP Scenario 3 - Display on SA 18 of the Remote LU RA3AC

```
D NET,ID=RAIPSNIP,SCOPE=ALL    B
IST097I DISPLAY ACCEPTED
IST075I NAME = RAIPSNIP, TYPE = PU_T2.1 394
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RAILSNIP, LINE GROUP = RAIGSNIP, MAJNOD = RAIBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I RA3AC    ACT/S         D
IST314I END
```

Figure 84. SNA over TCP/IP Scenario 3 - Display on SA 18 of the Adjacent Link Station RAIPSNIP

Display from Subarea 11

```
D NET,ID=RABPSNIP,SCOPE=ALL     C
IST097I DISPLAY ACCEPTED
IST075I NAME = RABPSNIP, TYPE = PU_T2.1 144
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RABLSNIP, LINE GROUP = RABGSNIP, MAJNOD = RABBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I RAPAC    ACT/S          A
IST314I END
```

Figure 85. SNA over TCP/IP Scenario 3 - Display on SA 11 of the Adjacent Link Station RABPSNIP

```
D NET,ID=RAPAC,SCOPE=ALL        A
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RAPAC, TYPE = CDRSC 162
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RABRSNIP
IST1184I CPNAME = USIBMRA.RAP - NETSRVR = ***NA***
IST1044I ALSLIST = RABPSNIP     C
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RABPSNIP
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RA3AC    ACTIV-S    F88F016488B1014A            0  0 USIBMRA  D
IST635I RA3AC    ACTIV-P    F6FF41648E938949            0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938948            0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938947            0  0 USIBMRA
IST314I END
```

Figure 86. SNA over TCP/IP Scenario 3 - Display on SA 11 of the Remote LU RAPAC

Display from Subarea 03

```
D NET,ID=RAPAC,SCOPE=ALL       A
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RAPAC, TYPE = CDRSC 696
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RA3RSNIP
IST479I CDRM NAME = RAB, VERIFY OWNER = NO
IST1184I CPNAME = USIBMRA.RAB - NETSRVR = ***NA***
IST1044I ALSLIST = ISTAPNPU
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RA3AC    ACTIV-S    F88F016488B1014A 0003 0002  0  0 USIBMRA  D
IST635I RA3AC    ACTIV-P    F6FF41648E938949 0000 0001  0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938948 0000 0001  0  0 USIBMRA
IST635I RA3AC    ACTIV-P    F6FF41648E938947 0002 0003  0  0 USIBMRA
IST924I -------------------------------------------------------------
IST075I NAME = USIBMRA.RAPAC, TYPE = DIRECTORY ENTRY
IST1186I DIRECTORY ENTRY = DYNAMIC LU
IST1184I CPNAME = USIBMRA.RAP - NETSRVR = ***NA***
IST314I END
```

Figure 87. SNA over TCP/IP Scenario 3 - Display on SA 03 of the Remote LU RAPAC

Chapter 13. Scenario 4 - MVS Access Node to OS/2 Access Node

As shown in Figure 88, this scenario is between an AnyNet/MVS SNA over TCP/IP system, SA11, and an AnyNet/2 SNA over TCP/IP system, WTR05221 via the IP network.

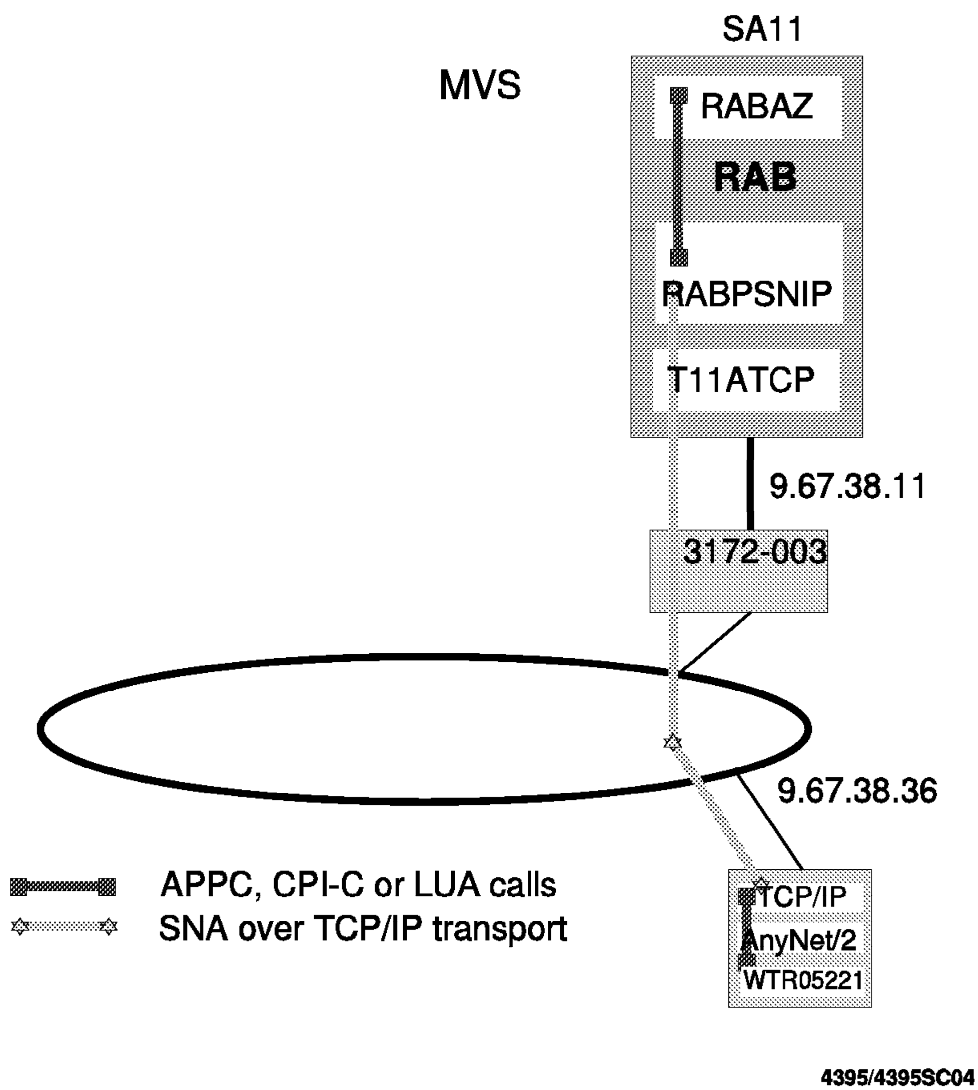


Figure 88. SNA over TCP/IP Test Scenario 4 - MVS-to-OS/2

13.1 Scenario 4 - Test Application

In this scenario the APING application for MVS and OS/2 was used.

APING is part of the updated APPC Application Suite that comes as additional sample application with IBM Communications Manager/2 Version 1.11.

In this scenario LU6.2 sessions between RABAZ and WTR05221 were started by issuing the APING command from the MVS system, RAB.

13.2 Scenario 4 - Procedures

The following procedure was used to test this scenario.

Subarea 11:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RABBSNIP

Testing:

- Log on to RABAT - TSO (Subarea 11)
- Start APING to USIBMRA.WTR05221

13.3 Scenario 4 - Test Results

The following displays were captured to show the LU 6.2 sessions with these commands.

Subarea 11:

- Display the APING ACB:

  D NET,ID=RABAZ,SCOPE=ALL

Display from Subarea 11

```
D NET,ID=RABAZ,SCOPE=ALL
IST097I  DISPLAY  ACCEPTED
IST075I  NAME = USIBMRA.RABAZ    , TYPE = APPL
IST486I  STATUS= ACT/S     , DESIRED STATE= ACTIV
IST977I  MDLTAB=***NA*** ASLTAB=***NA***
IST861I  MODETAB=MTAPPC   USSTAB=***NA*** LOGTAB=***NA***
IST934I  DLOGMOD=LU62APPC USS LANGTAB=***NA***
IST597I  CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I  APPL     MAJOR NODE = RABAZZ
IST654I  I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I  JOBNAME = APPC11  , STEPNAME = APPC11  , DSPNAME = IST22F8B
IST1050I  MAXIMUM COMPRESSION LEVEL - INPUT = 0   , OUTPUT = 0
IST171I  ACTIVE SESSIONS = 0000000002, SESSION REQUESTS = 0000000002
IST206I  SESSIONS:
IST634I  NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I  WTR05221 PSEST/B-S  F7EFD164B97BD70D 0000 0000       USIBMRA
IST635I  WTR05221 PSEST/B-S  F7EFD164B97BD70C 0000 0000       USIBMRA
IST635I  WTR05221 ACTIV-S    F7EFD164B97BD70B 0001 0000       USIBMRA
IST635I  WTR05221 ACTIV-S    F7EFD164B97BD70A 0001 0001       USIBMRA
IST314I  END
```

Figure 89. SNA over TCP/IP Scenario 4 - Display on SA 11 of the APING LU - RABAZ

```
APING version 2.44  APPC echo test with timings.
  by Peter J. Schwaller (pjs ralvm6.vnet.ibm.com)

Allocate duration:                         4000 ms

Connected to a partner running on: OS/2

Program startup and Confirm duration:      2000 ms

        Duration        Data Sent        Data Rate        Data Rate
        (msec)          (bytes)          (KB/s)           (Mb/s)
        --------        ---------        ---------        ---------
            1000              200              0.2            0.002
               0              200
Totals:     1000              400              0.4            0.003

Duration statistics:   Min = 0   Ave = 500   Max = 1000
```

Figure 90. SNA over TCP/IP Scenario 4 - TSO APING Messages

Chapter 14. Scenario 5 - VTAM MPTN Gateway and OS/2 Access Node

As shown in Figure 91, this scenario is between an AnyNet/2 SNA over TCP/IP system, WTR052211, and a native SNA MVS system, SA18, running APING over the IP network and across an MPTN SNA over TCP/IP gateway, SA03.

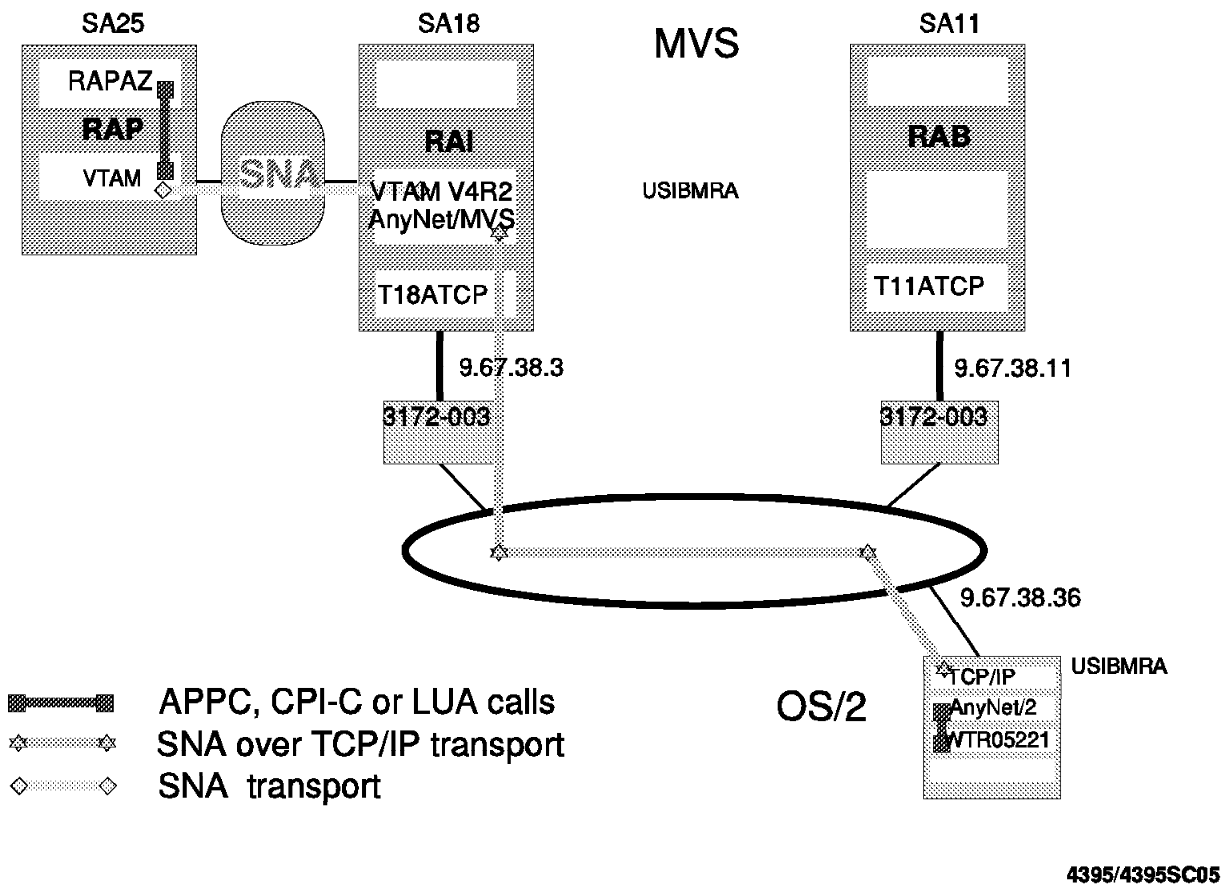


Figure 91. SNA over TCP/IP Test Scenario 5 - MVS-to-OS/2 via Gateway

14.1 Scenario 5 - Application Tested - APING

In this scenario an APING connection was tested. The LU6.2 sessions between RAPAC and WTR05221 were started by issuing the APING command from the MVS system, RAP.

14.2 Scenario 5 - Procedures

The following procedure was used to test this scenario.

Subarea 25:

- Activate the CDRSC major node:

  V NET,ACT,ID=RAPRSNIP

Subarea 18:

- Activate the TCP/IP major node:

  V NET,ACT,ID=RAIBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RAIRSNIP

Testing:

- Log on to RAPAT - TSO (Subarea 25)
- Enter the following command:

  `Start APING to USIBMRA.WTR05221`

14.3 Scenario 5 - Test Results

The following displays were captured to show the LU 6.2 sessions with these commands.

Subarea 25:

- Display the APING ACB:

  D NET,ID=RAPAZ,SCOPE=ALL

Display from Subarea 25

```
D NET,ID=RAPAZ,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.RAPAZ, TYPE = APPL 074
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=MTAPPC USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=LU62APPC USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I APPL MAJOR NODE = RAPAZZ
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I JOBNAME = APPC25, STEPNAME = APPC25, DSPNAME = IST220DE
IST1050I MAXIMUM COMPRESSION LEVEL - INPUT = 0, OUTPUT = 0
IST171I ACTIVE SESSIONS = 0000000004, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I WTR05221 ACTIV-S    F88F01648F45B38E 0001 0000  0  0 USIBMRA
IST635I WTR05221 ACTIV-S    F88F01648F45B38D 0005 0003  0  0 USIBMRA
IST635I WTR05221 ACTIV-S    F88F01648F45B38C 0004 0003  0  0 USIBMRA
IST635I WTR05221 ACTIV-S    F88F01648F45B38B 0002 0002  0  0 USIBMRA
IST314I END
```

Figure 92. SNA over TCP/IP Scenario 5 - Display on SA 25 of the APING LU - RAPAZ

```
APING VERSION 2.44  APPC ECHO TEST WITH TIMINGS.
  BY PETER J. SCHWALLER (PJS@RALVM6.VNET.IBM.COM)

ALLOCATE DURATION:                         2000 MS

CONNECTED TO A PARTNER RUNNING ON: OS/2

PROGRAM STARTUP AND CONFIRM DURATION:      6000 MS

        DURATION        DATA SENT        DATA RATE        DATA RATE
        (MSEC)          (BYTES)          (KB/S)           (MB/S)
        --------        ---------        ---------        ---------
               0              200
               0              200
TOTALS:        0              400

DURATION STATISTICS:   MIN = 0   AVE = 0   MAX = 0
```

Figure 93. SNA over TCP/IP Scenario 5 - TSO APING Messages

Displays from Subarea 18

```
D NET,ID=RAIPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RAIPSNIP, TYPE = PU_T2.1 207
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RAILSNIP, LINE GROUP = RAIGSNIP, MAJNOD = RAIBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I WTR05221 ACT/S      WTR05115 ACT/S
IST314I END
```

Figure 94. SNA over TCP/IP Scenario 5 - Display on SA 18 of the TCP PU RAIPSNIP

Chapter 15. Scenario 6 - OS/2 Access Node to OS/2 Access Node

In this scenario we established LU6.2 sessions between two OS/2 systems, MARTIN and SUSI, acting as MPTN access nodes.

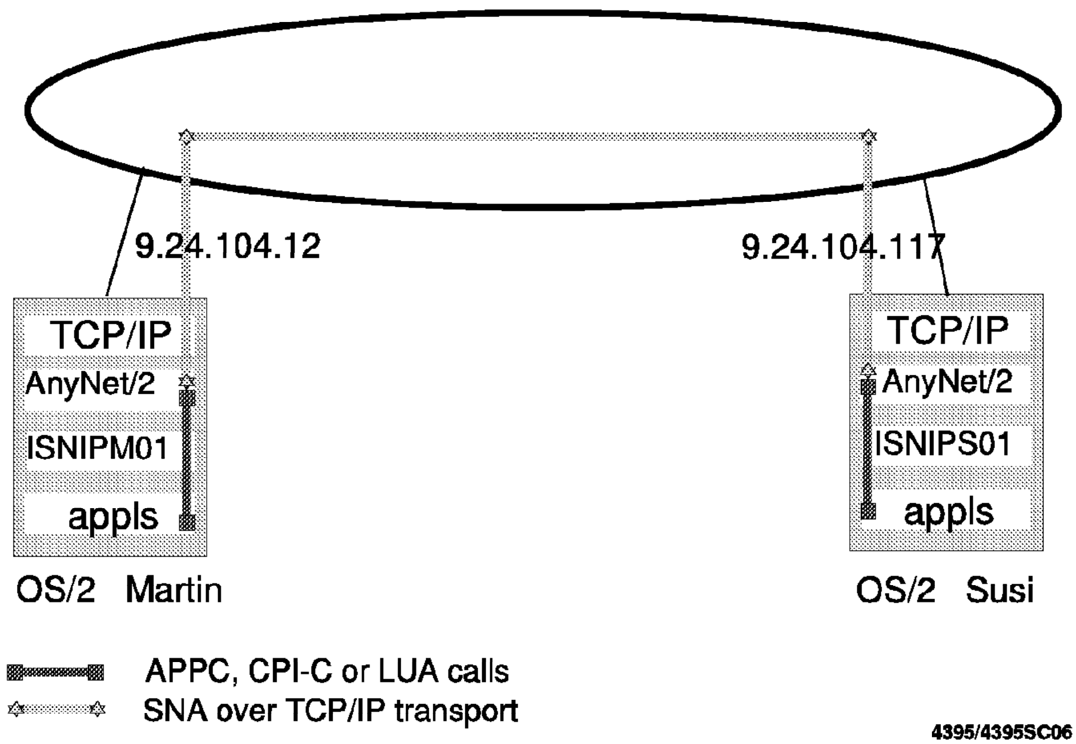


Figure 95. SNA over TCP/IP Scenario 6 - Network Picture

15.1 Scenario 6 - Applications Tested

The following applications have been tested in this scenario:

- APING
- Database 2 OS/2 (DB2/2)

APING is part of the updated APPC Application Suite that comes as additional sample application with IBM Communications Manager/2 Version 1.11.

DB2/2 is set up as a database server on system SUSI, and as a database client on system MARTIN.

15.2 Scenario 6 - Procedures

Following are the basic steps to set up a SNA over TCP/IP connection between two OS/2 systems. We assume that TCP/IP for OS/2 and Communications Manager are already installed.

- Start TCP/IP for OS/2 on both systems and verify that they can communicate with each other using the PING command:

```
[D:]ping 9.24.104.117
PING 9.24.104.117: 56 data bytes
64 bytes from 9.24.104.117: icmp_seq=0. time=32. ms
64 bytes from 9.24.104.117: icmp_seq=1. time=0. ms
64 bytes from 9.24.104.117: icmp_seq=2. time=0. ms
64 bytes from 9.24.104.117: icmp_seq=3. time=0. ms

----9.24.104.117 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/8/32
```

- Start Communications Manager on both systems.
- Verify that SNA over TCP/IP is initialized using the NETSTAT command with the -s option:

```
[D:]netstat -s
SOCK      TYPE        FOREIGN          LOCAL        FOREIGN      STATE
                         PORT           PORT          HOST
==== ==========     ==========     ==========     ========== ==========
14       DGRAM             0       mptn..397        0.0.0.0   UDP
13       DGRAM             0            1025        0.0.0.0   UDP
12      STREAM          1027            1026      127.0.0.1   ESTABLISHED
11      STREAM          1026            1027      127.0.0.1   ESTABLISHED
9       STREAM          1025            1024      127.0.0.1   ESTABLISHED
8       STREAM          1024            1025      127.0.0.1   ESTABLISHED
6       STREAM             0       mptn..397        0.0.0.0   LISTEN
```

  NETSTAT -s lists the active sockets on an OS/2 TCP/IP host. Since SNA over TCP/IP is a socket application, it will register itself to both, TCP and UDP, using the well-known port number 397. Additionally, connections have been established to the host itself, using either a local interface or the local loopback interface, if the latter has been configured (it has been in our case - 127.0.0.1). These connections are used for internal purposes.

- In an APPN environment, you can use the CP name of your workstation as an LU name for applications. You may, however, need to define specific local LU names for whatever application you want to use, so we defined a local LU name, as shown in the following example from the .NDF file of system MARTIN:

```
DEFINE_LOCAL_LU  LU_NAME(ISNIPM01)
                 LU_ALIAS(ISNIPM01)
                 NAU_ADDRESS(INDEPENDENT_LU);
```

- Define the necessary mode names for your LU6.2 applications, if any.
- Define the necessary transaction programs for your LU6.2 applications, if any.
- Set up the routing preference table using either the LULIST or the LUTPM program. In there, you define all fully-qualified partner LU names you want to communicate with via SNA over TCP/IP. For a simple test, select

NON-NATIVE_ONLY. This will leave TCP/IP as the only choice to be used as transport network for SNA over TCP/IP.

If you later want to have a backup connection via native SNA, change the routing preference to **NON-NATIVE** or to **NATIVE.**, whichever you want SNA over TCP/IP to try first. If you want to use just SNA transport, use a routing preference of **NATIVE_ONLY**. In all those cases, you have to add definitions for links and partner LUs to your Communications Manager profile.

The following screen shows the routing preference table for system SUSI using LULIST:

```
[D:temp]lulist p
Default is  NATIVE
Luname              Flag
--------------------------------
USIBMRA.RAIDB2          NON-NATIVE
USIBMRA.ISNIPMO1        NON-NATIVE
```

- Start your SNA application and establish a connection to the other system.
- Verify that the connection is properly set up and using TCP/IP as a transport using the Communications Manager Subsystem Management, as shown in the following example for system SUSI being a Database 2 for OS/2 server:

```
2>Session ID                                  X'1C6C37978627A680'
  Conversation ID                             X'97272E0C'
  LU alias                                    ISNIPS01
  Partner LU alias                            ISNIPM01
  Mode name
  Send maximum RU size                        8192
  Receive maximum RU size                     8192
  Send pacing window                          1
  Receive pacing window                       3
  Link name                                   overTCP
  Outbound destination address (DAF)          X'02'
  Outbound origin address (OAF)               X'04'
  OAF-DAF assignor indicator (ODAI)           B'0'
  Session type                                LU-LU session
  Connection type                             Peer
  Procedure correlator ID (PCID)              X'F0FB3EDF6B82A0D2'
  PCID generator CP name                      USIBMRA.WTR32271
  Conversation group ID                       X'8927A680'
  LU name                                     USIBMRA.ISNIPS01
  Partner LU name                             USIBMRA.ISNIPM01
  Pacing type                                 Adaptive
  Primary LU indicator                        Partner LU
  FMD PIUs sent by primary LU                 43
  FMD PIUs sent by secondary LU               42
  Non-FMD PIUs sent by primary LU             1
  Non-FMD PIUs sent by secondary LU           1
  Bytes sent by primary LU                    25670
  Bytes sent by secondary LU                  2835
  PLU to SLU compression level                None
  PLU to SLU compression percent              0
  SLU to PLU compression level                None
  SLU to PLU compression percent              0
```

Note: No mode name definitions are required for DB2/2 using LU6.2 connections. A mode name of BLANK is implied if none is specified.

- Verify that the proper IP addresses are associated with the connection using the ABINFO utility. Use the procedure correlator ID of the session above, as shown for system SUSI:

```
[C:anynetbin]abinfo a2b SNA f0fb3edf6b82a0d2
************** LOCAL A NAME   ************
NSAP = USIBMRA.ISNIPS01
************** PARTNER A NAME ************
NSAP = USIBMRA.ISNIPM01
************** LOCAL B NAME   ************
NSAP = 9.24.104.117
TSEL = 397
************** PARTNER B NAME ************
NSAP = 9.24.104.12
TSEL = 1031
```

- Verify that the connection is properly set up across SNA over TCP/IP using the NETSTAT command. It should list additional sockets active on port 397, as shown below for system SUSI:

```
[D:temp]netstat -s
SOCK      TYPE        FOREIGN          LOCAL        FOREIGN      STATE
                         PORT           PORT           HOST
==== ==========      ==========      ==========      ========== ==========
75      STREAM           1031       mptn..397     9.24.104.12   ESTABLISHED
74      STREAM           1030       mptn..397     9.24.104.12   ESTABLISHED
73      STREAM           1029       mptn..397    9.24.104.118   ESTABLISHED
72       DGRAM              0            1031         0.0.0.0   UDP
71      STREAM           1028       mptn..397    9.24.104.118   ESTABLISHED
24       DGRAM              0       mptn..397         0.0.0.0   UDP
23       DGRAM              0            1025         0.0.0.0   UDP
22      STREAM           1031            1030       127.0.0.1   ESTABLISHED
21      STREAM           1030            1031       127.0.0.1   ESTABLISHED
19      STREAM           1029            1028       127.0.0.1   ESTABLISHED
18      STREAM           1028            1029       127.0.0.1   ESTABLISHED
16      STREAM              0       mptn..397         0.0.0.0   LISTEN
```

- Test your SNA application(s).

15.3 Scenario 6 - Test Results

- The tested applications were run successfully on SNA over TCP/IP.
- In addition, all tests were also conducted using native SNA connections over a Communications Manager link. We observed a longer session establishment when using SNA over TCP/IP, but the application performance appeared to be as good as over native SNA connections.

Chapter 16. Scenario 7 - Dependent LU Support

As shown in Figure 96, this scenario is between an AnyNet/2 SNA over TCP/IP system, WTR05221, and an AnyNet/MVS SNA over TCP/IP system, SA11, running IBM Communications Manager/2 3270 Emulator (dependent LU2) via the IP network.

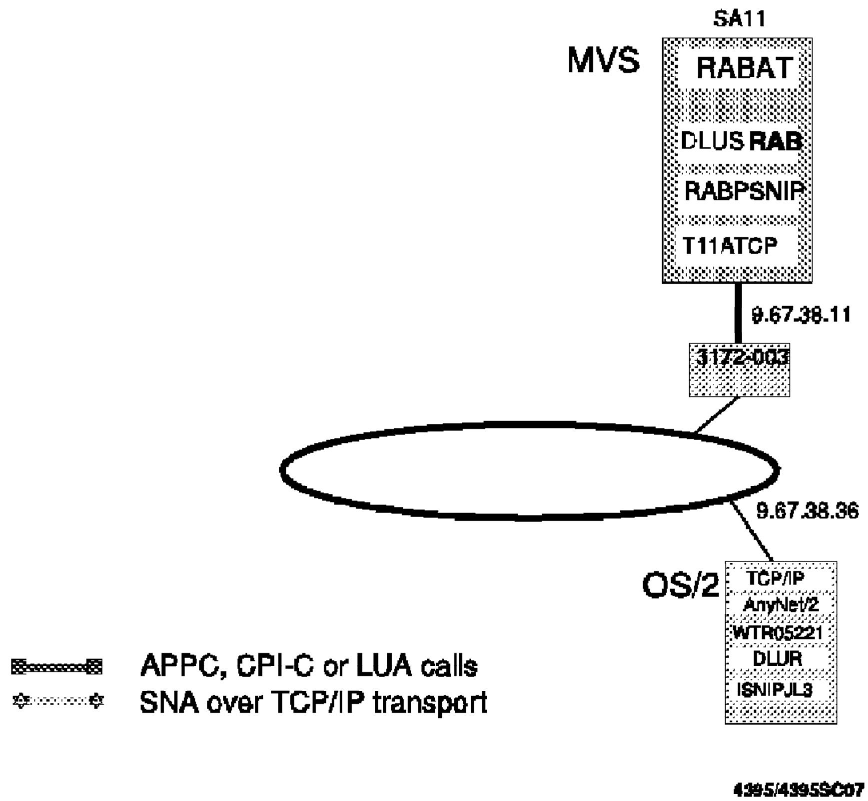


Figure 96. SNA over TCP/IP Test Scenario 7 - MVS-to-OS/2 (Dependent LU Support)

16.1 Scenario 7 - Test Application

In this scenario a 3270 emulation application running under CM/2 logs on to TSO (RABAT) running on the MVS system - RAB.

16.2 Scenario 7 - Procedures

The following procedure was used to test this scenario.

Subarea 11:

- Activate the TCP/IP major node:

V NET,ACT,ID=RABBSNIP

- Activate the CDRSC major node:

  V NET,ACT,ID=RABRSNIP

- Activate the Switched Major Node (Dependent LUs):

  V NET,ACT,ID=RABSSNIP

Testing:

- Log on to RABAT (TSO Subarea 11) from a 3270 emulation LU (ISNIPJL3) on the OS/2 system (WTR05221).

16.3 Scenario 7 - Test Results

The following displays were captured to show the dependent LU sessions with these commands.

Subarea 11:

- Display the Dependent LU Requestor/Server (DLURS)

  D NET,DLURS

- Display the Dependent LU Requestor CP

  D NET,WTR05221,SCOPE=ALL

- Display the TCP PU:

  D NET,ID=RABPSNIP,SCOPE=ALL

- Display the PU:

  D NET,ID=ISNIPJ01,SCOPE=ALL

- Display the dependent LU:

  D NET,ID=ISNIPJL3,SCOPE=ALL

Displays from Subarea 11

```
D NET,DLURS
IST097I DISPLAY ACCEPTED
IST350I DISPLAY TYPE = DLURS
IST1352I DLUR NAME          DLUS CONWINNER STATE  DLUS CONLOSER STATE
IST1353I USIBMRA.WTR05221 A   ACTIVE                ACTIVE
IST314I END
```

Figure 97. SNA over TCP/IP Scenario 7 - Display of DLURs on SA 11

```
D NET,ID=WTR05221,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.WTR05221, TYPE = CDRSC 373
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST1333I ADJLIST = ***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = RABRSNIP
IST1184I CPNAME = USIBMRA.RAB - NETSRVR = ***NA***
IST1044I ALSLIST = RABPSNIP
IST082I DEVTYPE = INDEPENDENT LU / CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000002, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST1081I ADJACENT LINK STATION = RABPSNIP B
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RAB      ACTIV/DL-S D96FD635E0901E31 000B 0000       USIBMRA C
IST635I RAB      ACTIV/DL-P F7EFD164B8FDB804 0000 000E       USIBMRA
IST1355I PHYSICAL UNITS SUPPORTED BY DLUR USIBMRA.WTR05221
IST089I ISNIPJ01 TYPE = PHYSICAL UNIT    , ACTIV
IST314I END
```

Figure 98. SNA over TCP/IP Scenario 7 - Display of the Dependent LU Requestor (WTR05221)

```
D NET,ID=RABPSNIP,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = RABPSNIP, TYPE = PU_T2.1
IST486I STATUS= ACTIV--L--, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST081I LINE NAME = RABLSNIP, LINE GROUP = RABGSNIP, MAJNOD = RABBSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I WTR05221 ACT/S      ISNIPJL3 ACT/S
IST314I END
```

Figure 99. SNA over TCP/IP Scenario 7 - Display of TCP/IP PU on SA 11

```
D NET,ID=ISNIPJ01,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = ISNIPJ01, TYPE = PU_T2 284
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST1043I CP NAME = ***NA***, CP NETID = USIBMRA, DYNAMIC LU = YES
IST1354I DLUR NAME = WTR05221           MAJNODE = RABSSNIP
IST136I SWITCHED SNA MAJOR NODE = RABSSNIP
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST355I LOGICAL UNITS:
IST080I ISNIPJL1 ACTIV      ISNIPJL2 ACTIV      ISNIPJL3 ACT/S
IST080I ISNIPJL4 ACTIV
IST314I END
```

Figure 100. SNA over TCP/IP Scenario 7 - Display of ISNIPJ01 - PU defined in VTAM

```
D NET,ID=ISNIPJL3,SCOPE=ALL
IST097I DISPLAY ACCEPTED
IST075I NAME = USIBMRA.ISNIPJL3, TYPE = LOGICAL UNIT 250
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=ISTINCLM USSTAB=US327X LOGTAB=***NA***
IST934I DLOGMOD=D4C32XX3 USS LANGTAB=***NA***
IST597I CAPABILITY-PLU INHIBITED,SLU ENABLED  ,SESSION LIMIT 00000001
IST136I SWITCHED SNA MAJOR NODE = RABSSNIP
IST135I PHYSICAL UNIT = ISNIPJ01
IST082I DEVTYPE = LU
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000001, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I RABAT03  ACTIV-P    F7EFD164B97BD6C2 00CF 0109       USIBMRA
IST314I END
```

Figure 101. SNA over TCP/IP Scenario 7 - Display of the Dependent LU

Chapter 17. Scenario 8 - AnyNet/400 APPC over TCP/IP

In this chapter we show two AnyNet/400 APPC over TCP/IP working scenarios:

- AnyNet/400 APPC over TCP/IP Scenario A: AS/400 to PS/2
- AnyNet/400 APPC over TCP/IP Scenario B: AS/400 to MVS

A section on AnyNet/400 APPC over TCP/IP verification is also included.

17.1.1 AnyNet/400 APPC over TCP/IP Scenario A: AS/400 to PS/2

Shown in the following figure are the two systems used in this scenario and their respective IP addresses. A TCP/IP connection is already in place between the systems using the internet addresses shown.

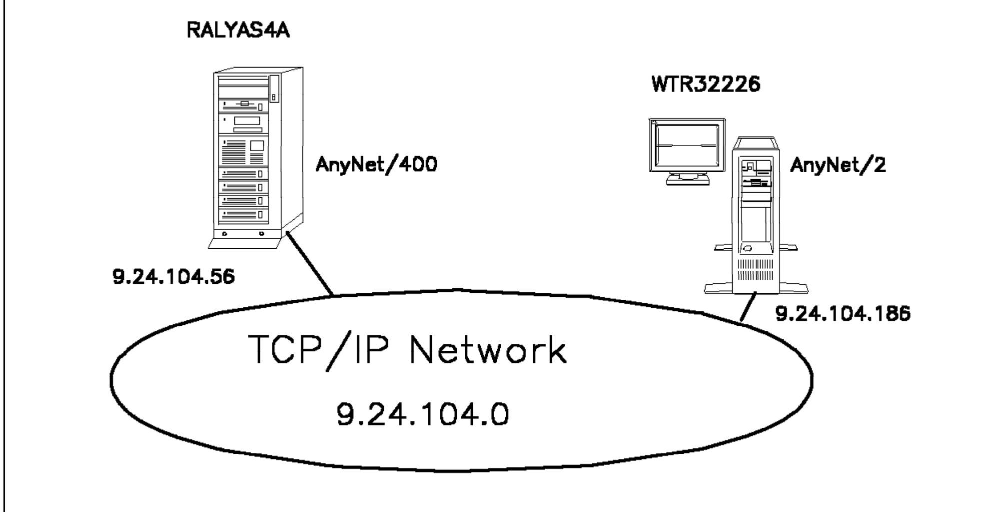


Figure 102. Systems and Addresses Used for AnyNet/400 APPC over TCP/IP Scenario A

The following series of panels show the AS/400 and PS/2 configuration screens taken from the RALYAS4A and WTR32226 systems. They illustrate the configuration steps required for this APPC over TCP/IP scenario.

Please note that only the key AnyNet/2 configuration displays are shown in this section.

PS/2 Software Installed

The following software was installed on WTR32226:

- OS/2 Version 2.1
- CM/2 Version 1.11 with AnyNet/2 support installed (additional functions)
- TCP/IP Version 2.0 for OS/2 Base kit plus CSD UN64092
- AnyNet/2 Version 2.0

The software was installed in the above order.

RALYAS4A Configuration

First we must check that Allow ANYNET Support is set to *YES in the network attributes of RALYAS4A. Use DSPNETA to determine this, and if necessary, use CHGNETA ALWANYNET (*YES) to change.

Next we create a controller description on RALYAS4A with LINKTYPE *ANYNW.

```
                     Create Ctl Desc (APPC) (CRTCTLAPPC)

 Type choices, press Enter.

 Controller description . . . . . > ANYNWPS2A     Name
 Link type  . . . . . . . . . . . > *ANYNW        *ANYNW, *FAX, *FR, *IDLC...
 Online at IPL  . . . . . . . . .   *YES          *YES, *NO
 Remote network identifier  . . .   *NETATR       Name, *NETATR, *NONE, *ANY
 Remote control point . . . . . . > WTR32226      Name, *ANY
 User-defined 1 . . . . . . . . .   *LIND         0-255, *LIND
 User-defined 2 . . . . . . . . .   *LIND         0-255, *LIND
 User-defined 3 . . . . . . . . .   *LIND         0-255, *LIND
 Text 'description' . . . . . . . > 'To PC Workstation via AnyNet/400'




                                                                         Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys
```

Figure 103. AnyNet/400 Scenario A: Controller Description - RALYAS4A

Before we can add an APPN remote location list entry, we need to determine the network ID of the remote system. The .ndf file from WTR32226 is show below.

```
DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR32226 )
                 CP_ALIAS(WTR32226)
                 NAU_ADDRESS(INDEPENDENT_LU)
                 NODE_TYPE(EN)
                 NODE_ID(X'05D32226')
                 NW_FP_SUPPORT(NONE)
                 HOST_FP_SUPPORT(YES)
                 HOST_FP_LINK_NAME(HOST$1  )
                 MAX_COMP_LEVEL(NONE)
                 MAX_COMP_TOKENS(0);

DEFINE_LOGICAL_LINK  LINK_NAME(LINK0001)
                     FQ_ADJACENT_CP_NAME(USIBMRA.RALYAS4A )
                     ADJACENT_NODE_TYPE(LEARN)
                     DLC_NAME(IBMTRNET)
                     ADAPTER_NUMBER(0)
                     DESTINATION_ADDRESS(X'400010020001O4')
                     ETHERNET_FORMAT(NO)
                     CP_CP_SESSION_SUPPORT(NO)
                     SOLICIT_SSCP_SESSION(NO)
                     ACTIVATE_AT_STARTUP(YES)
                     USE_PUNAME_AS_CPNAME(NO)
                     LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
```

```
                     LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                     MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                     EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                     COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                     COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                     SECURITY(USE_ADAPTER_DEFINITION)
                     PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_3(USE_ADAPTER_DEFINITION);

DEFINE_PARTNER_LU  FQ_PARTNER_LU_NAME(USIBMRA.RALYAS4A )
                   PARTNER_LU_ALIAS(RALYAS4A)
                   PARTNER_LU_UNINTERPRETED_NAME(RALYAS4A)
                   MAX_MC_LL_SEND_SIZE(32767)
                   CONV_SECURITY_VERIFICATION(NO)
                   PARALLEL_SESSION_SUPPORT(YES);

DEFINE_PARTNER_LU_LOCATION  FQ_PARTNER_LU_NAME(USIBMRA.RALYAS4A )
                            WILDCARD_ENTRY(NO)
                            FQ_OWNING_CP_NAME(USIBMRA.RALYAS4A )
                            LOCAL_NODE_NN_SERVER(NO);

DEFINE_MODE  MODE_NAME(QPCSUPP )
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(1024)
             RECEIVE_PACING_WINDOW(7)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(64)
             MIN_CONWINNERS_SOURCE(32)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                 DEFAULT_MODE_NAME(BLANK)
                 MAX_MC_LL_SEND_SIZE(32767)
                 DIRECTORY_FOR_INBOUND_ATTACHES(*)
                 DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                 DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                 DEFAULT_TP_CONV_SECURITY_RQD(NO)
                 MAX_HELD_ALERTS(10);

START_ATTACH_MANAGER;
```

The network attributes from RALYAS4A are as follows:

```
                          Display Network Attributes
                                                          System:   RALYAS4A
 Current system name  . . . . . . . . . . . . . . :   RALYAS4A
   Pending system name  . . . . . . . . . . . . . :
 Local network ID . . . . . . . . . . . . . . . . :   USIBMRA
 Local control point name . . . . . . . . . . . . :   RALYAS4A
 Default local location . . . . . . . . . . . . . :   RALYAS4A
 Default mode . . . . . . . . . . . . . . . . . . :   BLANK
 APPN node type . . . . . . . . . . . . . . . . . :   *NETNODE
 Data compression . . . . . . . . . . . . . . . . :   *NONE
 Intermediate data compression  . . . . . . . . . :   *NONE
 Maximum number of intermediate sessions  . . . . :   200
 Route addition resistance  . . . . . . . . . . . :   128
 Server network ID/control point name . . . . . . :   *LCLNETID   *ANY


                                                                     More...
```

Figure 104. AnyNet/400 Scenario A: Network attributes - RALYAS4A

Here we add the APPC over TCP/IP entry to the APPN remote location list at RALYAS4A.

```
                          Change Configuration List                 RALYAS4A
                                                          11/10/94  10:47:23
 Configuration list  . . :   QAPPNRMT
 Configuration list type :   *APPNRMT
 Text  . . . . . . . . . :

 Type changes, press Enter.

 --------------------------APPN Remote Locations---------------------------
           Remote              Remote    Control
 Remote    Network   Local     Control   Point     Location         Secure
 Location  ID        Location  Point     Net ID    Password          Loc
 WTR32226  *NETATR   *NETATR   WTR32226  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
 ________  *NETATR   *NETATR   ________  *NETATR   _______________   *NO
                                                                     More...
 F3=Exit   F11=Display session information   F12=Cancel   F17=Top   F18=Bottom
```

Figure 105. AnyNet/400 Scenario A: APPN Remote Locations List - RALYAS4A

From the PS/2 .ndf file and AS/400 network attributes above, we can see that the network IDs on both systems are the same (USIBMRA) hence we have specified *NETATR for Remote Network ID and Control Point Net ID in the APPN remote location list entry above.

The host table at RALYAS4A in the following figure has had the APPC over TCP/IP entries added.

```
                    Work with TCP/IP Host Table Entries
                                                          System:   RALYAS4A
 Type options, press Enter.
   1=Add   2=Change   4=Remove   5=Display   7=Rename

      Internet          Host
 Opt  Address           Name

  _   _______________
  _   9.24.104.56       RALYAS4A
                        RALYAS4A.ITSO.RAL.IBM.COM
                        RALYAS4A.USIBMRA.SNA.IBM.COM
  _   9.24.104.186      AS4PS2A
                        AS4PS2A.ITSO.RAL.IBM
                        WTR32226.USIBMRA.SNA.IBM.COM




 F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 106. AnyNet/400 Scenario A: TCP/IP Host Table Entries

Note that in this scenario the PS/2′s SNA name (USIBMRA.WTR32226) is different from its TCP/IP host name (AS4PS2A).

WTR32226 Configuration

To configure AnyNet/2 SNA over TCP/IP we define the following:

- SNA Domain Name Suffix
- Routing Preference

An SNA Domain Name Suffix is required to establish an IP domain name for SNA over TCP/IP, its format is *luname.netid.snasuffix*, and defined as follows:

- luname is the SNA LU name.
- netid is the SNA network ID (NETID).
- snasuffix is the domain name suffix defined in CONFIG.SYS.

To define the SNA Domain Name Suffix, we use the AnyNet/2 SNA over TCP/IP configuration tool. To access the AnyNet/2 SNA over TCP/IP configuration tool, select the **Configure AnyNet/2 SNA over TCP/IP** icon from the AnyNet/2 folder. The folder icon should be displayed on the OS/2 desktop, if the AnyNet/2 code has been installed correctly.

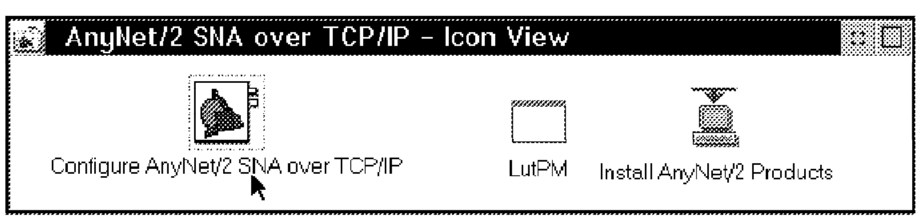


Figure 107. The AnyNet/2 SNA over TCP/IP Folder

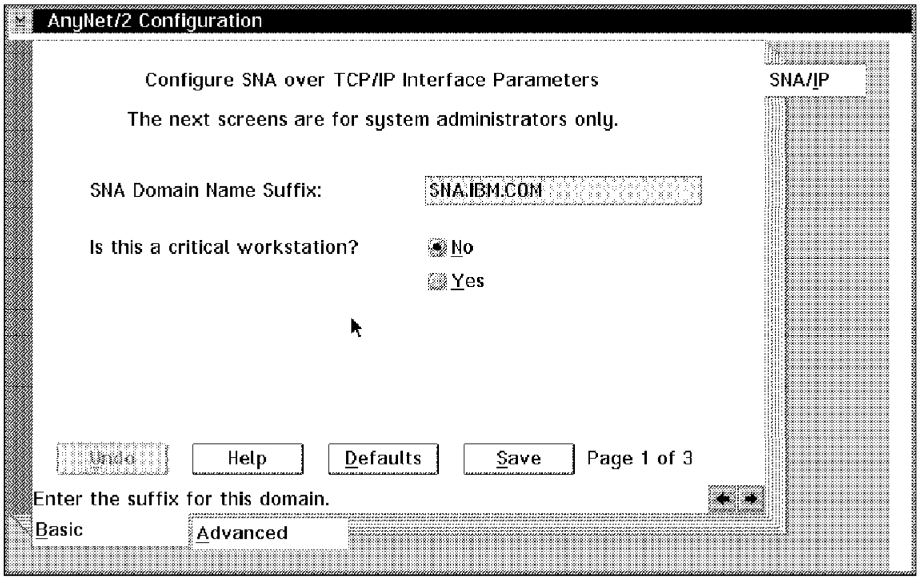


Figure 108. AnyNet/400 Scenario A: SNA Domain Name Suffix - System WTR32226

When initiating a session, AnyNet/2 SNA over TCP/IP uses a preference table to determine whether native SNA or SNA over TCP/IP (nonnative) will be used for that session. If no routing preference table is configured, the default is to first try to establish the session over native SNA. If this session setup fails, SNA over TCP/IP will be used.

To customize the routing preference table, we can use the lulist OS/2 command. When entered, the command prompts with the following information:

```
usage: lulist {a|r|l|p|f|c|d|u|h} argument(s).
    Arguments by function:.
    a netid.luname flag ( ADD LUNAME    ).
    r netid.luname      ( REMOVE LUNAME ).
    l netid.luname      ( LOOKUP LUNAME ).
    p                   ( PRINT TABLE   ).
    f                   ( FLUSH TABLE   ).
    c netid.luname flag ( CHANGE LUNAME ).
    d                   ( PRINT DEFAULT ).
    d flag              ( SET DEFAULT   ).
    u                   ( UPDATE TABLE  ).
    h                   ( HELP          ).
flag: 0=Native, 1=Non-Native, 2=Native Only, 3=Non-Native Only.
```

The options available for the table default and table entries are:

Native: SNA will be tried first. If the session request fails, SNA over TCP/IP will be used.

Non-native: SNA over TCP/IP will be tried first. If the session fails, SNA will be used.

Native only: Only SNA will be used.

Non-native only: Only SNA over TCP/IP will be used.

As for AnyNet/400, AnyNet/2 SNA over TCP/IP uses the native TCP/IP host table to map SNA LU names to internet addresses. The OS/2 TCP/IP host table is changed either via the TCP/IP Configuration icon (page 3 of the Services configuration section), or by editing the HOSTS file (\tcpip\etc\hosts).

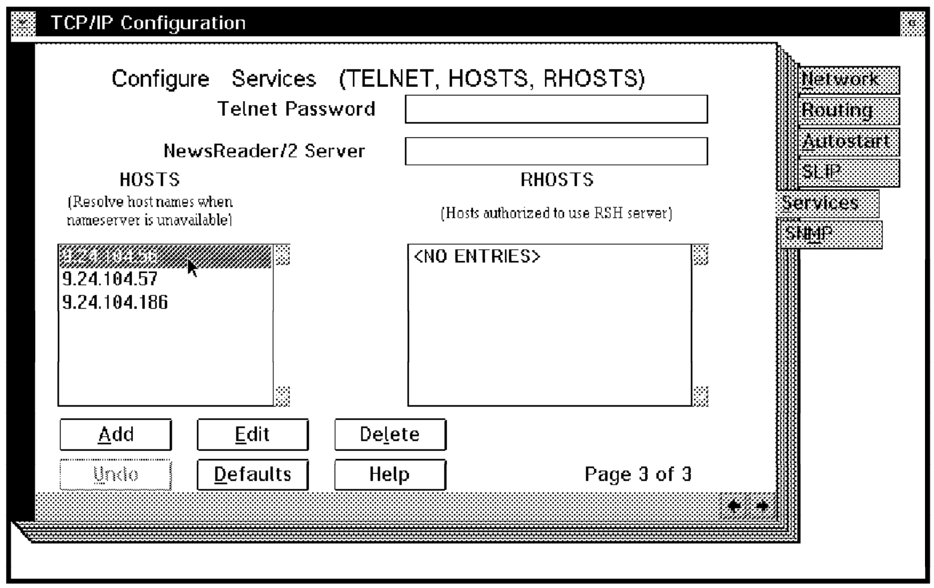


Figure 109. OS/2 TCP/IP Host Table Menu

Update the table with the required mapping, as shown in Figure 110.

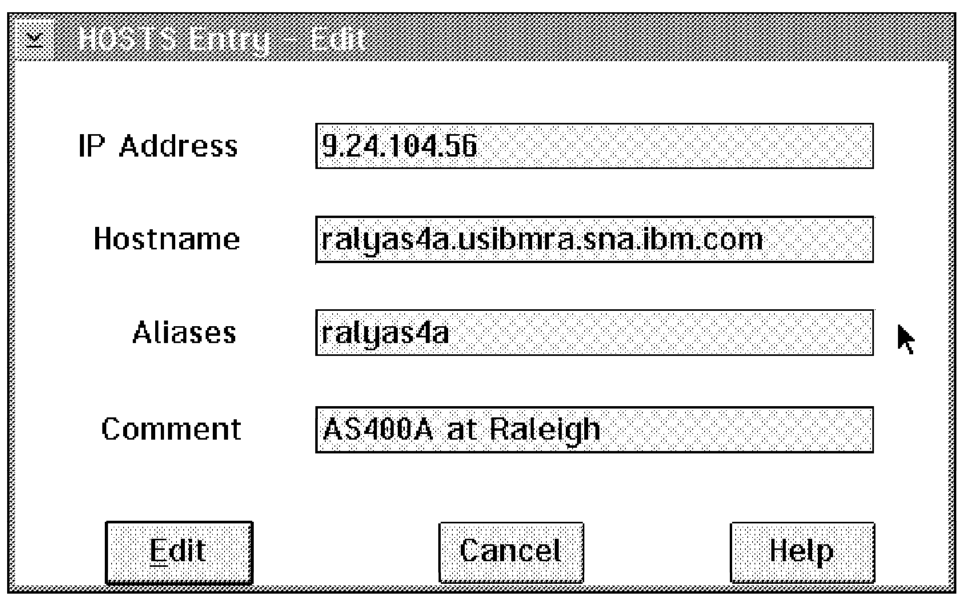


Figure 110. OS/2 TCP/IP Host Table Entry

The *Aliases* field above can contain multiple host names. This would have allowed us to enter the long TCP/IP host name for RALYAS4A in addition to the short one shown.

Shown next are the matching parameters between RALYAS4A and WTR32226.

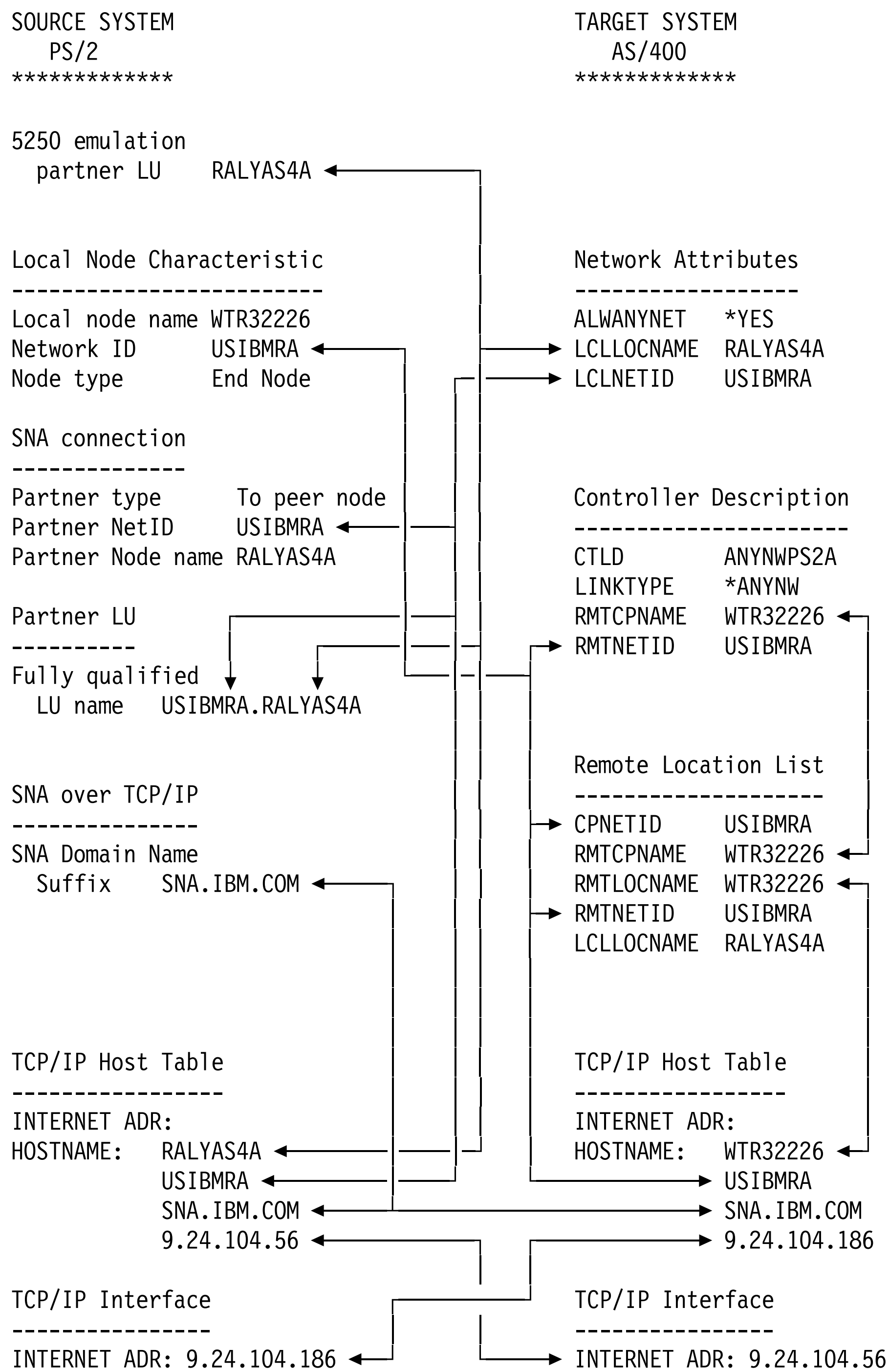


Figure 111. AnyNet/400 APPC over TCP/IP Scenario A: Matching Parameters Table

17.1.2 AnyNet/400 APPC over TCP/IP Scenario B: AS/400 to MVS

Shown in the following figure are the two systems used in this scenario and their respective IP addresses. A TCP/IP connection is already in place between the systems using the internet addresses shown.

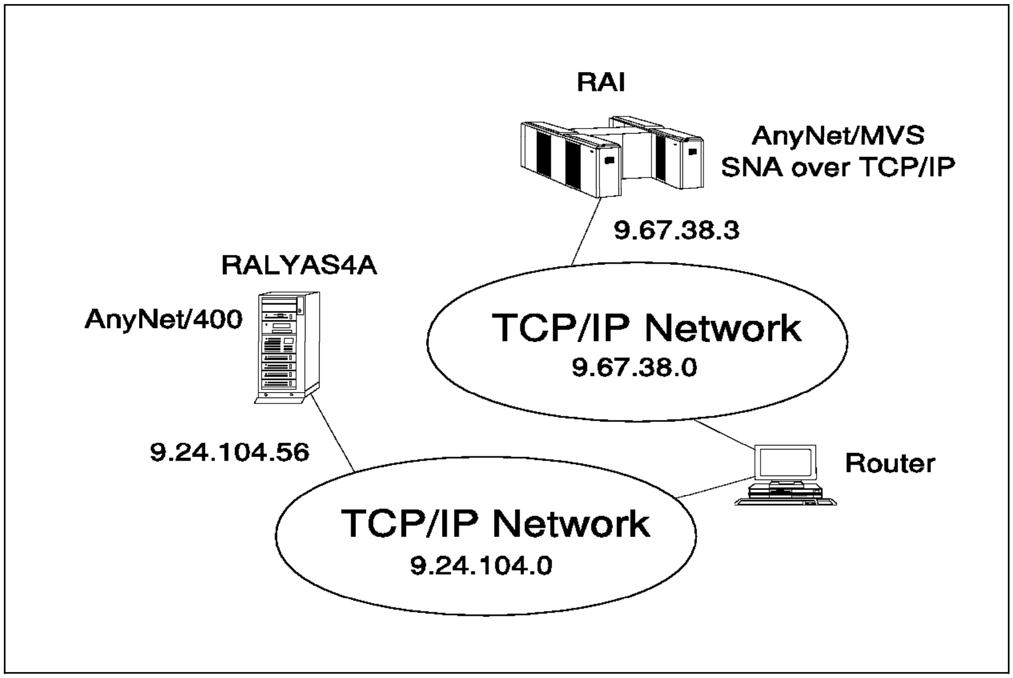


Figure 112. Systems and Addresses Used for AnyNet/400 APPC over TCP/IP Scenario B

The following series of panels show the AS/400 configuration screens taken from the RALYAS4A system. Also shown is a subset of the AnyNet/MVS configuration information from RAI. They illustrate the configuration steps required for this APPC over TCP/IP Gateway scenario.

Please note that only the key AnyNet/MVS configuration displays are shown in this section.

Host Software Installed

The following software was installed on RAI:

- IBM MVS/ESA System Product (SP) Version 3 Release 1.3
- IBM MVS/ESA System Modification Program/Extended (SMPE) Release 5
- IBM TCP/IP Version 2 Release 2.1 for MVS
- IBM C for System/370 Version 2 at PUT level 9107
- VTAM Version 4 Release 2 base
- VTAM Version 4 Release 2 AnyNet host feature

RALYAS4A Configuration

First we must check that Allow ANYNET Support is set to *YES in the network attributes of RALYAS4A. Use DSPNETA to determine this and if necessary use CHGNETA ALWANYNET (*YES) to change.

Next we create a controller description on RALYAS4A with LINKTYPE *ANYNW.

```
                  Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

Controller description . . . . . > ANYNWMVSI     Name
Link type  . . . . . . . . . . . > *ANYNW        *ANYNW, *FAX, *FR, *IDLC...
Online at IPL  . . . . . . . . .   *YES          *YES, *NO
Remote network identifier  . . .   *NETATR       Name, *NETATR, *NONE, *ANY
Remote control point . . . . . . > RAIANYNT      Name, *ANY
User-defined 1 . . . . . . . . .   *LIND         0-255, *LIND
User-defined 2 . . . . . . . . .   *LIND         0-255, *LIND
User-defined 3 . . . . . . . . .   *LIND         0-255, *LIND
Text 'description' . . . . . . . > 'To AnyNet/MVS'




                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 113. Anynet/400 Scenario B: AS/400 Controller Description

The network attributes from RALYAS4A are as follows.

```
                        Display Network Attributes
                                                      System:   RALYAS4A
Current system name  . . . . . . . . . . . . . . :   RALYAS4A
  Pending system name  . . . . . . . . . . . . . :
Local network ID . . . . . . . . . . . . . . . . :   USIBMRA
Local control point name . . . . . . . . . . . . :   RALYAS4A
Default local location . . . . . . . . . . . . . :   RALYAS4A
Default mode . . . . . . . . . . . . . . . . . . :   BLANK
APPN node type . . . . . . . . . . . . . . . . . :   *NETNODE
Data compression . . . . . . . . . . . . . . . . :   *NONE
Intermediate data compression  . . . . . . . . . :   *NONE
Maximum number of intermediate sessions  . . . . :   200
Route addition resistance  . . . . . . . . . . . :   128
Server network ID/control point name . . . . . . :   *LCLNETID   *ANY


                                                                     More...
```

Figure 114. AnyNet/400 Scenario B: AS/400 Network Attributes

In the following panel we add the APPC over TCP/IP entries to the APPN remote location list at RALYAS4A.

```
                          Change Configuration List                    RALYAS4A
                                                             03/08/95  14:06:54
Configuration list  . . :   QAPPNRMT
Configuration list type :   *APPNRMT
Text  . . . . . . . . . :

Type changes, press Enter.

--------------------------APPN Remote Locations----------------------------
          Remote              Remote    Control
Remote    Network   Local     Control   Point     Location          Secure
Location  ID        Location  Point     Net ID    Password           Loc
RAI_____  USIBMRA   RALYAS4A  RAIANYNT  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
________  *NETATR   *NETATR   ________  *NETATR   _______________    *NO
                                                                       More...
F3=Exit   F11=Display session information   F12=Cancel   F17=Top   F18=Bottom
```

Figure 115. AnyNet/400 Scenario B: AS/400 APPN Remote Locations

The host table at RALYAS4A shown following has had the APPC over TCP/IP entry added.

```
                     Work with TCP/IP Host Table Entries
                                                             System:   RALYAS4A
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   7=Rename

     Internet         Host
Opt  Address          Name

 _   _______________
 _   9.24.104.56      RALYAS4A
                      RALYAS4A.ITSO.RAL.IBM.COM
                      RALYAS4A.USIBMRA.SNA.IBM.COM
 _   9.67.38.3        RAI.USIBMRA.SNA.IBM.COM




F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 116. AnyNet/400 Scenario B: AS/400 TCP/IP Host Table

AnyNet/MVS Configuration

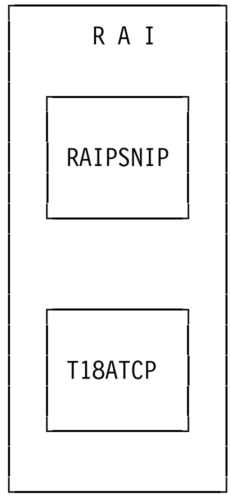


Figure 117. Anynet/MVS Configuration

The IP network is represented to VTAM as a TCP/IP major node, using a VBUILD TYPE=TCP as shown in Figure 118.

```
EDIT       RISC.VTAMLST(RAIBSNIP) - 01.03                  Columns 00001 00072
****** ***************************** Top of Data ******************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001 *********************************************************************
000002 *                                                                   *
000003 *           VTAM 42   ANYNET   -      SNA OVER TCP/IP               *
000004 *                                                                   *
000005 *                      SA 18  DEFINITIONS                           *
000006 *                                                                   *
000007 *********************************************************************
000008 RAIBSNIP VBUILD  TYPE=TCP,                                             X
000009                CONTIMER=30,           WAIT FOR MPTN TO COME UP       X
000010                DGTIMER=30,            INTERVAL BETWEEN RETRIES       X
000011                DNSUFX=IBM.COM,        DOMAIN NAME SUFFIX             X
000012                EXTIMER=3,             BETW. SEND SNA EXPEDITED DATA  X
000013                IATIMER=120,           TIME BEFORE MPTN KEEPALIVE     X
000014                PORT=397,              WELLKNOWN PORT FOR ANYNET      X
000015                TCB=10,                NUMBER MVS SUBTASKS            X
000016                TCPIPJOB=T18ATCP       TCP/IP JOBNAME
000017 RAIGSNIP GROUP ISTATUS=ACTIVE         GROUPNAME
000018 RAILSNIP LINE  ISTATUS=ACTIVE         LINENAME
000019 RAIPSNIP PU    ISTATUS=ACTIVE         PUNAME
****** **************************** Bottom of Data ****************************
```

Figure 118. IP Network Representation to VTAM

When using APPC over TCP/IP, VTAM sees any remote LUs as independent LUs, which are defined as CDRSCs.

```
EDIT       RISC.VTAMLST(RAIRSNIP) - 01.07                  Columns 00001 00072
****** ***************************** Top of Data ******************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001 *********************************************************************
000002 *                          UPDATE LOG                               *
000003 *                                                                   *
000004 * 03/07/95 MCLI MODIFY COMMENTS                                     *
000003 *                                                                   *
000003 *                                                                   *
000008 * ---------------------------------------------------------------- *
000009 *        VTAM  42  ANYNET SNA OVER TCP/IP                           *
000010 *                                                                   *
000011 *                      SA 18  DEFINITIONS                           *
000012 *                                                                   *
000013 * NAME   CDRSC ALSLIST=.....NAME OF THE PU STATEMENT DEFINED        *
000014 *                           WITHIN THE VBUILD TYPE=TCP              *
000015 *                                                                   *
000016 *     - THE NAME LABEL OF THE CDRSC DEFINITION STATEMENT MUST BE    *
000017 *       THE REMOTE ILU NAME.                                        *
000018 *                                                                   *
000019 *     - WE MUST CODE ALSREQ=YES TO USE THE PREDEFINED LIST.         *
000020 *                                                                   *
000021 *     - WE USE SOME CDRSC WITH THE NETID NOT CODED IN ORDER TO      *
000022 *       THE CDRMNAME AS AN ADJSSCP.                                 *
000023 *                                                                   *
000024 ************************************************************************
000025          VBUILD  TYPE=CDRSC
000026 *
000027          NETWORK NETID=USIBMRA
000028 *
000034 RALYAS4A CDRSC ALSLIST=RAIPSNIP,ALSREQ=YES     AS400
****** **************************** Bottom of Data ****************************
```

Figure 119. LU Representation to VTAM

The TCP/IP Host Table used by AnyNet/MVS SNA over TCP/IP is the normal host table.

```
EDIT       TCPIP.ITSC.HOSTS.LOCAL                          Columns 00001 00072
****** ***************************** Top of Data ******************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001 ; ----------------------------------------------------------------------
000002 ;                          Update log
000003 ; 01/31/95 mcli  Change 9.67.38.3 to 9.67.38.20
000004 ;
000005 ; ----------------------------------------------------------------------
000006 ;                          WATSON IP ADDRESSES
000007 ; NOTES:
000008 ;  1. To request additions, changes, or deletions from this file please
000009 ;     use the WATIP REQUEST online form which can be found on the
000010 ;     CMSSYS 19f disk (also known as the U disk). Follow further
000011 ;     instructions within WATIP REQUEST.
000012 ;  2. This file should NOT contain any blank lines.
000013 ; ----------------------------------------------------------------------
000014 ;
000015 ;  Ring 9.2.1.0 - Netmask 255.255.255.128 - Hawthorne I 16Mb
000016 ;  Begin 9.2.1.0
000017 HOST : 14.0.0.0  : YKTVMV , CIAMPA, GARY , GTC, ME , TEST ::::
000018 HOST:9.67.38.36:WTR05221.USIBMRA.IBM.COM,ISNIPJL1.USIBMRA.IBM.COM ::::
000019 HOST:9.67.38.36:ISNIPJL2.USIBMRA.IBM.COM,ISNIPJL3.USIBMRA.IBM.COM ::::
000020 HOST:9.67.38.36:ISNIPJL4.USIBMRA.IBM.COM ::::
000021 HOST:9.67.38.37:WTR05115.USIBMRA.IBM.COM,ISNIPML1.USIBMRA.IBM.COM ::::
000022 HOST:9.67.38.37:ISNIPML2.USIBMRA.IBM.COM,ISNIPML3.USIBMRA.IBM.COM ::::
000023 HOST:9.67.38.37:ISNIPML4.USIBMRA.IBM.COM ::::
000024 HOST : 9.67.38.35 : WTR05222.USIBMSC.IBM.COM ::::
000025 HOST : 9.67.38.20 : RAIAC.USIBMRA.IBM.COM ::::
000026 HOST : 9.67.38.11 : RABAT.USIBMRA.IBM.COM ::::
000027 HOST : 9.67.38.20 : RAPAC.USIBMRA.IBM.COM ::::
000028 HOST : 9.67.38.11 : RA3AC.USIBMRA.IBM.COM ::::
000029 HOST : 9.67.38.11 : RABAC.USIBMRA.IBM.COM ::::
000030 HOST : 9.24.104.56: RALYAS4A.USIBMRA.IBM.COM ::::
000032 ;
****** **************************** Bottom of Data ****************************
```

Figure 120. VTAM TCP/IP Host Table

No matching parameter table was created for this scenario.

17.1.3 AnyNet/400 APPC over TCP/IP Verification

First we should check that the APPC over TCP/IP job is running. The command WRKACTJOB SBS(QSYSWRK) will display the active jobs in the QSYSWRK subsystem. The APPC over TCP/IP job QAPPCTCP should be active as shown in the following figure.

```
                              Work with Active Jobs                       RALYAS4A
                                                               03/08/95  17:24:02
CPU %:     .0     Elapsed time:   00:00:00     Active jobs:   63

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
__   QSYSWRK        QSYS        SBS      .0                   DEQW
_5     QAPPCTCP     QSYS        BCH      .0  PGM-QZPAIJOB     TIMW
__     QECS         QSVSM       BCH      .0  PGM-QNSECSJB     DEQW
__     QMSF         QMSF        BCH      .0                   DEQW
__     QNSCRMON     QSVSM       BCH      .0  PGM-QNSCRMON     DEQW
__     QTCPIP       QTCP        BCH      .0                   DEQW
__     QTFTP00619   QTCP        BCH      .0                   DEQW
__     QTFTP00734   QTCP        BCH      .0                   DEQW
__     QTFTP02472   QTCP        BCH      .0                   TIMW
                                                                         More...
Parameters or command
===> _______________________________________________________________________
F3=Exit      F5=Refresh   F10=Restart statistics   F11=Display elapsed data
F12=Cancel   F23=More options   F24=More keys
```

Figure 121. AS/400 Work with Active Jobs Panel

If we look at the job log associated with QAPPCTCP, we see the following:

```
                               Display Job Log
                                                          System:   RALYAS4A
Job . . :   QAPPCTCP      User . . :   QSYS         Number . . . :   011338

 >> CALL QSYS/QZPAIJOB
    APPC over TCP/IP job started.









                                                                      Bottom
Press Enter to continue.

F3=Exit   F5=Refresh   F10=Display detailed messages   F12=Cancel
F16=Job menu           F24=More keys
```

Figure 122. AS/400 Display Job Log (QAPPCTCP) Panel

Note

The APPC over TCP/IP job (QAPPCTCP) is initially started when the Allow AnyNet support (ALWANYNET) network attribute is changed to *YES. If the job fails for any reason, it is necessary to stop TCP/IP (ENDTCP), and start TCP/IP (STRTCP) again to re-start the job.

Before we can use the AS/400 APPC over TCP/IP configuration, we must Vary on the APPC controller description we created for the APPC over TCP/IP connection. The Work with Configuration Status command can be used to show the status of the controller. For example, the following command resulted in the display shown in Figure 123 on page 174.

```
WRKCFGSTS *CTL ANYNWPS2A
```

```
                         Work with Configuration Status                RALYAS4A
                                                              03/08/95  16:30:11
Position to  . . . . .   __________   Starting characters

Type options, press Enter.
  1=Vary on   2=Vary off   5=Work with job   8=Work with description
  9=Display mode status ...

Opt  Description       Status                -------------Job--------------
__   ANYNWPS2A         VARIED OFF





                                                                          Bottom
Parameters or command
===> _________________________________________________________________________
F3=Exit   F4=Prompt   F12=Cancel   F23=More options   F24=More keys
```

Figure 123. AS/400 Work with Configuration Status for Controller at RALYAS4A

To make the configuration available, use option 1 (Vary on). The configuration should then go to a VARIED ON status.

When the fist controller with link type *ANYNW is varied on, two TCP/IP jobs will be started to allow the system to accept incoming APPC over TCP/IP sessions. NETSTAT option 3 can be used to display all TCP/IP sessions (native TCP/IP and APPC over TCP/IP). Figure 124 on page 175 shows NETSTAT option 3 prior to any APPC over TCP/IP sessions being established.

```
                     Work with TCP/IP Connection Status
                                                          System:   RALYAS4A
Local internet address . . . . . . . . . . . :   *ALL

Type options, press Enter.
  4=End   5=Display details

     Remote           Remote     Local
Opt  Address          Port       Port       Idle Time  State
     *                *          ftp-con >  026:45:25  Listen
     *                *          telnet     025:04:38  Listen
     *                *          APPCove >  000:09:55  Listen
     *                *          APPCove >  000:09:55  *UDP
     *                *          lpd        026:44:24  Listen





                                                                      Bottom
F5=Refresh   F11=Display byte counts   F13=Sort by column
F14=Display port numbers   F22=Display entire field   F24=More keys
```

Figure 124. AS/400 NETSTAT Option 3 - TCP/IP Connection Status

The two APPC over TCP/IP (APPCove) jobs are for serving incoming TCP and UDP, APPC over TCP/IP session requests. If the jobs fail for any reason, it is necessary to stop TCP/IP (ENDTCP) and start TCP/IP (STRTCP) again to re-start the jobs.

An AS/400 command that can be used to verify an APPC connection to *any* remote APPC system is STRMOD. For example:

```
STRMOD RMTLOCNAME(WTR32226)
Command STRMOD completed successfully for mode BLANK device WTR32226.
The STRMOD command completed successfully for all modes.
```

With a session active, WRKCFGSTS shows the active session in the normal way, Figure 125 on page 176 shows the active session and autocreated device for WTR32226.

```
                     Work with Configuration Status              RALYAS4A
                                                        03/08/95  16:40:03
Position to  . . . . .   __________   Starting characters

Type options, press Enter.
  1=Vary on   2=Vary off   5=Work with job   8=Work with description
  9=Display mode status ...

Opt  Description       Status                -------------Job--------------
__   ANYNWPS2A         ACTIVE
__     WTR32226        ACTIVE









                                                                      Bottom
Parameters or command
===> _________________________________________________________________________
F3=Exit   F4=Prompt   F12=Cancel   F23=More options   F24=More keys
```

Figure 125. AS/400 Work with Configuration Status for Controller at RALYAS4A

The NETSTAT option 3 display in Figure 126 shows the active session from a TCP/IP perspective.

```
                      Work with TCP/IP Connection Status
                                                       System:   RALYAS4A
Local internet address  . . . . . . . . . . . :   *ALL

Type options, press Enter.
  4=End   5=Display details

     Remote           Remote     Local
Opt  Address          Port       Port       Idle Time  State
     *                *          ftp-con >  027:06:16  Listen
     *                *          telnet     025:25:30  Listen
     *                *          APPCove >  000:30:47  Listen
     *                *          APPCove >  000:30:46  *UDP
     *                *          lpd        027:05:16  Listen
     9.24.104.186     APPCove >  1036       000:01:19  Established




                                                                      Bottom
F5=Refresh   F11=Display byte counts   F13=Sort by column
F14=Display port numbers   F22=Display entire field   F24=More keys
```

Figure 126. AS/400 NETSTAT Option 3 - TCP/IP Connection Status

The ability to establish APPC over TCP/IP sessions can be verified in many ways. Above we showed the use of STRMOD which results in a CNOS (Change Number of Sessions) LU6.2 command flowing to the remote system.

Another means of testing the connection is to use APING, this test tool is available for all IBM platforms and many non-IBM platforms. It functions, in an APPC environment, in a very similar way to PING in a TCP/IP environment. In

Figure 127 on page 177 we have used APING from a PS/2 system to an AS/400 via an APPC over TCP/IP connection.

```
APING for Destination: USIBMRA.RALYAS4A -- RALYAS4A
Allocate duration:                          3390 ms
Connected to a partner running on: OS/400
Program startup and Confirm duration:      3437 ms
      Duration       Data Sent       Data Rate       Data Rate
      (msec)         (bytes)         (KB/s)          (Mb/s)
      --------       ---------       ---------       ---------
           125             200             1.6           0.013
            62             200             3.2           0.025
Totals:    187             400             2.1           0.017
Duration statistics:   Min = 62   Ave = 93   Max = 125
```

Figure 127. APING sample Output from PS/2 to AS/400

Chapter 18. Conclusions

This chapter presents the conclusions that are drawn from the planning, installing, configuring and testing the scenarios of SNA over TCP/IP at the International Technical Support Organization in Raleigh.

This information is grouped into the following sections:

1. Installation and configuration
2. Quality of the code
3. Ease of use of the SNA over TCP/IP technology
4. Functionality
5. Performance

18.1 Installation and Configuration

The installation and configuration process on MVS/ESA should be fairly easy in a production environment that is well documented and looked after.

The installation and configuration of AnyNet/2 is very straightforward, but you have to be careful to meet the product requirements and installation sequence. For more information please see the following:

- 5.10, "How SNA over TCP/IP Works (OS/2)" on page 77.
- 5.9, "How SNA over TCP/IP Works (MVS/ESA)" on page 76.
- 6.3, "VTAM Definitions" on page 84.
- 7.8, "How to Map LU Names to IP Addresses" on page 106.
- 7.9, "Using the Routing Preference Table" on page 107.

18.2 Quality of the Code

The overall quality of the code was good, and the functions that were implemented did meet the announced objectives as well as our expectations.

18.3 Ease of Use of the SNA over TCP/IP Technology

As a second release, the implementation of SNA over TCP/IP on OS/2 offers excellent ease of use.

18.4 Functionality

As a second release, the implementation of SNA over TCP/IP on OS/2 provides excellent functionality.

18.5 Performance

The authors did not have a dedicated testing environment for running performance tests, nor did we have a chance to tune any part of the network. All parameters and definitions were taken by default values or were set by common values used by past experience. Therefore, the following ″performance related″ test results obtained in the test scenarios should be viewed with caution:

1. The LU6.2 session establishment between two SNA over TCP/IP (OS/2) systems takes significantly longer than using native SNA with IBM Communications Manager/2. Once the sessions are established, application performance appears to be almost as good as over native SNA.

   This shows that the performance characteristics of SNA over TCP/IP (OS/2) are at least partly driven by the performance characteristics of TCP/IP for OS/2, especially the inherent delay involved for looking up a domain name server.

   Note

   This indicates, however, that implementing MPTN in terms of using SNA applications as the *transport user* and TCP/IP as the *transport provider* does not necessarily imply a performance penalty.

   Because of the relative easiness of handling TCP/IP definitions compared to SNA, especially in the workstation environment, there should be no considerable disadvantage for an installation to choose TCP/IP as its native transport network by implementing MPTN.

Appendix A. System Definitions - MVS

A.1 TCP/IP Definitions for SA18

```
; -------------------------------------------------------------------*
; -------------------------------------------------------------------*
;     ITSC-Raleigh:  TCP/IP MVS V3 System  *RAIANJE*                 *
;     ---------------------------------------------                  *
;   - MVS/ESA 4.3 - MVS18 / SA18                                     *
;   - Name of TCPIP Configuration File =                             *
;     TCPIP.V3R1.RAIANJE.A.PROFILE.TCPIP / SYS1.TCPPARMS(T18APROF)   *
; ------------------------------------------------------ 94/09/14 ----*
; ****************************************************************
; * DATA BUFFER DEFINITIONS:                                         *
; ****************************************************************
;                                       | size | default | max     | max    |
;                                       |(byte)| nbr     | nbr     | size   |
;                                       |======|=========|=========|========|
; TABUFFERPOOLSIZE         100 8192   ; | 8192 |   160   |   VS    | 262144|
DATABUFFERPOOLSIZE         100 28672  ; | 8192 |   160   |   VS    | 262144|
SMALLDATABUFFERPOOLSIZE    1200       ; | 2048 |     0   |   VS    |        |
TINYDATABUFFERPOOLSIZE     500        ; |  256 |     0   |   VS    |        |
;
;
; Note: VS=Limited by the amount of available Virtual Storage
;
; ****************************************************************
; * FREE BUFFER POOL DEFINITIONS:                                *
; ****************************************************************
;                                       | size | default | max     | max    |
;                                       |(byte)| nbr     | nbr     | size   |
;                                       |======|=========|=========|========|
ACBPOOLSIZE                1000       ; |   96 |  1000   |         |        |
CCBPOOLSIZE                300        ; | 1184 |   150   |   VS    |        |
RCBPOOLSIZE                50         ; |  228 |    50   |   VS    |        |
SKCBPOOLSIZE               256        ; |  781 |   256   |   VS    |        |
IPROUTEPOOLSIZE            300        ; |  128 |   600   |   VS    |        |
ADDRESSTRANSLATIONPOOLSIZE 1500       ; |  153 |  1500   |   VS    |        |
ENVELOPEPOOLSIZE           750        ; | 2048 |   750   |   VS    |        |
LARGEENVELOPEPOOLSIZE      100 32768  ; | 8192 |    50   |   VS    |        |
SCBPOOLSIZE                256        ; |  228 |   256   |   VS    |        |
TCBPOOLSIZE                256        ; | 1056 |   256   |   VS    |
UCBPOOLSIZE                100        ; |  245 |   100   |   VS    |
;
;
; Note: VS=Limited by the amount of available Virtual Storage
;
```

Figure 128 (Part 1 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; ****************************************************************
; * ASSORTED Parameters, which have scope of entire TCP/IP engine*
; ****************************************************************
ASSORTEDPARMS       ;
; ALWAYSWTO         ; Issue WTO messages for the TCP/IP server messages
; NOFWD             ; All gateway functions of TCP/IP are disabled
; IGNOREREDIRECT    ; Ignore ICMP redirect packets (like ?)
; NOUDPQUEUELIMIT   ; Relaxes incoming UDP datagram queuesize (def=21)
; NOACBCUSHION      ; Disables dynamic allocation of ACBs.
  MESSAGECASE MIXED ; TCP/IP Messages will be displayed in mixed case.
  TCPIPSTATISTICS   ; Print TCP/IP counters.
ENDASSORTEDPARMS    ;
;
; ****************************************************************
; * KEEP alive options, defining the contents and timing of      *
; * KEEP alive messages sent over a TCP connection               *
; ****************************************************************
; KEEPALIVEOPTIONS    ;
;  INTERVAL 120       ; send message every 120 minutes
;  SENDGARBAGE TRUE   ; send random data and invalid sequence number
; ENDKEEPALIVEOPTIONS ;
;
; ****************************************************************
; * TRACE  DEFINITIONS: (NEEDS TO BE CUSTOMIZED)                 *
; ****************************************************************
;
; TRACE RESOLVER SCREEN
; TRACE ARP
; MORETRACE ARP
; MORETRACE CLAW
; FILE 'JOAO.TRACE.DATA'
; FILE 'TCPIP.V3R1.RAIANJE.A.TRACE'
; TRACE SOCKET NOSCREEN
; MORETRACE SOCKET NOSCREEN
; MORETRACE TELNET
; TRACE TELNET MORETRACE TELNET NOSCREEN
; MORETRACE ALL NOSCREEN
; MORETRACE PCCA NOSCREEN
; TIMESTAMP PREFIX            ; TIMESTAMP PREFIXES EVERY MESSAGE
; PKTTRACE                    ; START IP PACKET TRACING, GTF must be
;                             ; started and TRACE of process too.
;  LINKNAME=*                 ; ALL LINKS
;  PROT=TCP                   ; PROTOCOL is TCP
;  IP=9.24.104.17             ; TRACE IP in and out to this address only
;  SUBNET=255.255.255.0       ; USED subnetmask
;  SRCPORT=*                  ; UDP or TCP source port to be traced
;  DESTPORT=*                 ; UDP or TCP destination port to be traced
;  FULL                       ; Trace entire packet
; PKTTRACE CLEAR              ; STOP IP PACKET TRACING
;  LINKNAME=*                 ; ALL LINKS
; ****************************************************************
; * SYSTEM ADMINISTRATION:                                       *
; ****************************************************************
;
;   ****************************************************************
;   * INFORM FOLLOWING USERS ABOUT SERIOUS RUN-TIME CONDITIONS     *
;   ****************************************************************
INFORM
  FONVILL
ENDINFORM
;
```

Figure 128 (Part 2 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
;   ****************************************************************
;   * ALLOW USE OF OBEYFILE COMMAND FOR FOLLOWING USERS:           *
;   ****************************************************************
;
OBEY
  STCTCP1                    ; Real RACF User Id's of SNMPQE SNMPD
  T18AROUT                   ; ROUTED
  T18ASNMQ                   ; SNMPQE
  T18ASNMD                   ; SNMPD
  SNMPQE                     ; Dummy RACF ID for SNMPQE
  SNMPD18                    ; Dummy RACF ID for SNMPD
  T18ANCPR                   ; NCP route server
  STUSER                     ; User on which NCPROUTE is running
  MCLI                       ; Michael LI
  ALFREDC                    ; Alfred
  PURRING                    ; JOHN PURRINGTON
ENDOBEY
;
;   ****************************************************************
;   * RESTRICT USE OF TCPIP SERVICES FOR FOLLOWING USERS:          *
;   ****************************************************************
;
RESTRICT
; HACKER1                    ; RESTRICT User Id's from TCPIP Services
; HACKER2                    ; Which services????
ENDRESTRICT                  ;
;
;   ****************************************************************
;   * SMF LOGGING OF TCPIP APPLICATIONS                            *
;   ****************************************************************
;
  SMFPARMS 121 122 123
;  121           ; sub type field in API initialization
;  122           ; sub type field in API termination
;  123           ; sub type field for FTP or TELNET client
;  LAODEXIT      ; Call TCPCNSMF
;
;
;   ****************************************************************
;   * PROBLEMS? CALL YOUR SYSTEM CONTACT: SNMP OBJECT SYSCONTACT   *
;   ****************************************************************
;
SYSCONTACT
  MICHAEL LI     301-2323
ENDSYSCONTACT
;
;
;   ****************************************************************
;   * KNOW WHERE YOUR SYSTEM IS RUNNING: SNMP OBJECT SYSLOCATION   *
;   ****************************************************************
;
SYSLOCATION
  MVS18 ITSO RALEIGH
ENDSYSLOCATION
;
;
```

Figure 128 (Part 3 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
;   ****************************************************************
;   * ARP TABLE ENTRIES FLUSHED AFTER THIS TIME LIMIT EXPIRES:     *
;   ****************************************************************
;
; ARPAGE N      ; AFTER N MINUTES, THE CREATED OR REVALIDATED ARP
;               ; TABLE ENTRY IS DELETED.
;
; ****************************************************************
; * DATASET prefix for search order of server and client         *
; * configuration files                                          *
; ****************************************************************
;
DATASETPREFIX TCPIP.V3R1.RAIANJE.A
; ****************************************************************
; * HARDWARE DEFINITIONS:                                        *
; ****************************************************************
;
;
;
;  *************************************************************
;  *    3172 Model 3 attachments: OFFLOAD or (MPE) dual boot box
;  *************************************************************
;          ================================
;          |                              |
;          |       TCPIP18                |
;          | IP addr: 9.24.104.126= OFFTR1|
;          | IP addr: 9.67.32.22  = OFFEN1|
;          | IP addr: 9.67.46.85  = OFFWAC|
;          --------------------------------
;      subch addr || 320-33F
;     ESCON ChPID || 18
;          +----+----+----+
;     9032 !    ! E1 !    !
;          !    +----+    !
;    ESCON !              !
; Director !    +----+    !
;          !    ! E4 !    !
;          +----+----+----+
;                 ||
;                 ||
;                 ||
;        ------------------------+
; ESCON  |       320              |
; slot   |       1-2              |
;        |    3172 Model 3        |
;        |     OFFLOAD/MPE        |
; Dev to |OFFEN1 OFFTR1 OFFWAC    |
; Rel adp|  0     0      0        |
; number |                        |
; Slot   |  6     7      4        |
;        ------------------------+
; MACs:     |     |      |
;           |     |      4000 012A 3172
;           |     4000 014A 3172
;           4000 012A 3172
;
```

Figure 128 (Part 4 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; ****************************************************************
; * Device and link definitions for 3172 C   - MPE (OFFLOAD)     *
; ****************************************************************
;
  DEVICE MPEDEV    CLAW  320  TCPIP  OS2TCP  NONE
  LINK   MPEIPLINK OFFLOADLINK1  1  MPEDEV
  LINK   OFFEN1  OFFLOADAPIBROAD 9.67.32.22     MPEDEV  MPEIPLINK
  LINK   OFFTR1  OFFLOADAPIBROAD 9.67.38.3      MPEDEV  MPEIPLINK
; LINK   OFFWAC  OFFLOADAPIBROAD 9.67.46.85     MPEDEV  MPEIPLINK
;
;
; *********************************************************************
; * TO START THE PROCEDURES (HIGHER LEVEL SERVICES) USE AUTOLOG.
; *********************************************************************
;   These procedures can be found in TCPIP.PROCLIB
;
;
AUTOLOG
;  T18AMISC                      ; Miscelaneous Server
;  T18ASNO                       ; SNALINK TO MVS03
;  T18ASNOX                      ; SNALINK TO TCPIP(B)
;  T18ASNOA                      ; SNAlink dummy AB
;  T18ASNOB                      ; SNAlink dummy BA
;  T18AFTPP                      ; FTP server - Pascal
;  T18AFTPC                      ; FTP server - C
;  T18AROUT                      ; ROUTED server
;  T18ASNMQ                      ; SNMP Query Engine
;  T18ASNMD                      ; SNMP Agent
;  T18APORT                      ; PORT MAPPER
;  T18ALK09                      ; SNALINK TO NCP IP ROUTER IN SA=09
;  T18ANCPR                      ; NCP ROUTE SERVER
;  T18ALU62                      ; SNAlink LU 6.2 TO MVS03
;  T18AOS62                      ; SNAlink LU 6.2 TO OS2
;  T18ASMTP                      ; SMTP server
;  T18AX25                       ; X.25 NPSI support
;  T18ADNS                       ; NAME SERVER
;  T18ANFS                       ; NFS Server
;  T18ALPD                       ; LPD Server
;  T18ANDBP                      ; NDB Portmap manager
;  T18ANDB1                      ; NDB Server 1
;  T18ALLBD                      ; NCS Local location broker
;  T18AGLBD                      ; NCS Global location broker
;  T18ANDB2                      ; NDB Server 2
;  T18ARXEC                      ; Remote Execution Server
ENDAUTOLOG
;
```

Figure 128 (Part 5 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; *********************************************************************
; * THE PORT NUMBER assignment for the procedures (applications)
; * the PROTOCOL used on the port.
; *********************************************************************
;   These procedures can be found in TCPIP.PROCLIB
;   Port values from RFC 1010, "Assigned numbers"
;
PORT
    7 UDP T18AMISC              ; MISCSERVer
    7 TCP T18AMISC              ; MISCSERVer
    9 UDP T18AMISC              ; MISCSERVer
    9 TCP T18AMISC              ; MISCSERVer
   19 UDP T18AMISC              ; MISCSERVer
   19 TCP T18AMISC              ; MISCSERVer
   20 TCP T18AFTPC              ; FTP C server data port
          NOAUTOLOG             ; Do not restart if stopped previously
;         DELAYACKS             ; Delay transmission acknowledgements
   21 TCP T18AFTPC              ; FTP C server control port
   23 TCP INTCLIEN              ; Telnet server,same port number as
                                ; on INTERNALCLIENTPARMS statement
   25 TCP T18ASMTP              ; SMTP server
   53 TCP T18ADNS               ; Name Server
   53 UDP T18ADNS               ; Name Server
   111 UDP T18APORT             ; Portmapper Server
   111 TCP T18APORT             ; Portmapper Server
   125 UDP T18ALLBD             ; NCS Local Location Broker
   161 UDP T18ASNMD             ; Agent Port for SNMP Messages
   162 UDP T18ASNMQ             ; SNMP Client port for Trap's
;  xxx TCP xxxx NOAUTOLOG       ; CSFI INTERFACE
   512 TCP T18ARXEC             ; Remote Execution
   514 TCP T18ARXEC             ; Remote Execution
   515 TCP T18ALPD              ; Remote Printer Server
   520 UDP T18AROUT             ; RouteD Server
   580 UDP T18ANCPR             ; NCPROUTE for RIP support on NCPs
  2049 UDP T18ANFS              ; NFS Server
  3000 TCP CICSTCP              ; CICS Sockets
  3001 TCP CICSTCP              ; CICS Sockets
; 3002 TCP T18AIMSL             ; IMS SOCKETS DAVE HERR SAMPLE
; 3010 TCP IMS18TCP             ; IMS SOCKETS
; 3011 TCP IMS18TCP             ; IMS SOCKETS
; 2221 TCP OFFLFTP              ; reserved port for Offload FTP
; 2223 TCP OFFLTN               ; reserved port for Offload telnet
; 4000 TCP MPEBOX               ; reserved port for MPE box telnet
  7020 TCP T18AFTPP NOAUTOLOG   ; FTP Pascal server data port
  7021 TCP T18AFTPP             ; FTP Pascal server control port
;
```

Figure 128 (Part 6 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; *********************************************************************
; * HOME IP addresses for the LINKS of TCPIP in MVS18
; *********************************************************************
; The first IP address in the home link is used as source address
; in IP datatgrams sent to non-adjacent nodes, unless the statement
; PRIMARYINTERFACE is specified, in which case the IP address of this
; primary interface is used.
;
HOME  ; The Local Host's
; Local host's Internet addresses
;   9.24.104.74    TR1          ; MAC=4000 5001 3172  IP into MPN
;   9.12.11.51     TR1          ; MAC=4000 5001 3172  IP INTO MPN
;   9.67.38.3      OFFTR1       ; OFFload TR1 interface
;   9.67.32.1      EN1          ; MAC=4000 6001 3172 ; JF
;   9.67.42.1    CLAWLINK       ; RS60007 ESCON
;   9.67.41.1      SNALMV03     ; snalink to TCPIP SA03 9.67.41.2
;   9.67.41.1      L62L1803     ; SNALINK to TCPIP SA03 9.67.41.2
;   9.67.81.1      L62L1811     ; SNALINK to TCPIP SA11 9.67.81.2
;   9.67.51.1      L62L1825     ; SNALINK to TCPIP SA25 9.67.51.2
; 9.67.43.1      X25LMV18
; 9.67.47.1     CTCLMVTS        ; CTC TO TCPIP SA03 9.67.47.2
; 9.67.130.1      L62LOS2       ; SNALINK LU6.2 TO OS/2 9.67.130.2
; 9.67.44.1      TONCP9L      ; SNALINK TO NCP IN SA09 9.67.44.2
; 9.67.55.1      SNALMV20     ; SNALINK to TCP/IP in MVS 20
; 9.67.56.18     IUCLM18B     ; IUCV CONNECTION TO MVS 18B
; *********************************************************************
; * PRIMARYINTERFACE STATEMENT for TCPIP in MVS18
; *********************************************************************
; The first IP address in the home link is used as source address
; in IP datatgrams sent to non-adjacent nodes, unless the statement
; PRIMARYINTERFACE is specified, in which case the IP address of this
; primary interface is used.
;
PRIMARYINTERFACE   OFFTR1
;
;
; *********************************************************************
; * TRANSLATE STATEMENT FOR IP ADDRESS to LINK ADDRESS TRANSLATION
; *********************************************************************
; TRANSLATE
;   9.67.56.1  HCH
;   ttxxxxxxhhcc
;   HCHLINK1
;   9.67.32.4  ETHERNET
;   201020304050
;   EN4
;   9.67.38.4  IBMTR
;   401020304050
;   TR4
;   9.67.40.4  FDDI
;   400000111111
;   FD4
;
;
; ********************************************************************
; * GATEWAY statements for static routing
; ********************************************************************
;
GATEWAY
  9.0.0.0      =        OFFTR1 DEFAULTSIZE  0.255.255.0  0.67.38.0
```

Figure 128 (Part 7 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; *******************************************************************
; * 3172 Offload OFFTR1 LINK connection to 9.24.104.0, use 9.67.38.03
; *******************************************************************
; 9.0.0.0      =       OFFTR1 DEFAULTSIZE  0.255.255.0    0.67.38.0
; 9.0.0.0      =       OFFEN1 DEFAULTSIZE  0.255.255.0    0.67.32.0
; 9.0.0.0      =          EN1 DEFAULTSIZE  0.255.255.0    0.67.32.0
; 9.0.0.0      =       OFFWAC DEFAULTSIZE  0.255.255.192  0.67.46.64
; 9.0.0.0  9.67.46.82   OFFWAC DEFAULTSIZE  0.255.255.192  0.67.46.128
; *******************************************************************
; * X.25 link to RS60007 over 9.67.43.0 network.
; *******************************************************************
; 9.0.0.0      =     X25LMV18 1024         0.255.255.0    0.67.43.0
; MAXIMUMRETRANSMITTIME 6.00  ; wait time for ACK before resending packet
; MINIMUMRETRANSMITTIME 0.75  ; The formula is:
; ROUNDTRIPGAIN         0.12  ; ?
; VARIANCEGAIN          0.25  ; ?
; VARIANCEMULTIPLIER    2.00  ; ?
; DELAYACKS                   ; Delay of ACKs for TCP connections.
; *******************************************************************
; * Anything not routed above, give to router 9.24.104.1, via TR1
; *******************************************************************
  DEFAULTNET 9.67.38.10  OFFTR1   DEFAULTSIZE  0
;
; ***************************************************************
; MAXIMUMRETRANSMITTIME 60.00 ; wait time for ACK before resending packet
; MINIMUMRETRANSMITTIME 00.75 ; The formula is:
; ROUNDTRIPGAIN         00.12 ; ?
; VARIANCEGAIN          00.25 ; ?
; VARIANCEMULTIPLIER    20.00 ; ?
; DELAYACKS             20.00 ; Delay of ACKs for TCP connections.
;
; ********************************************************************
; * RouteD Routing statements for dynamic routing
; * Static definitions are in SYS1.TCPPARMS(GATEWAYS)
; *
; ********************************************************************
; BSDROUTINGPARMS false    ; default max mtu size of 576 bytes is used
;                          for packets to non-adjacent networks.
;    LINK     maxmtu   METRIC   SUBNET MASK     DESTINATION ADDRESS
;                                               (point to point links)
;    CTCLMVTS  2000      0     255.255.255.0   9.67.47.2
;    CTCLMVTS  2000      0     255.255.255.0   9.67.89.135  ?????????
;    L62LMV03  2000      0     255.255.255.0   9.67.41.2
;    L62LMV03  2000      0     255.255.255.0   9.67.41.26   ?????????
;    SNALMV03  2000      0     255.255.255.0   9.67.41.2
;    EN1       2000      0     255.255.255.0   0
;    OFFEN1    2000      0     255.255.255.0   0
;    OFFTR1    2000      0     255.255.255.0   0
;    L62LOS2   2000      0     255.255.255.0   9.24.62.80
;    TR1       2000      0     255.255.255.0   0
;    X25LMV18  2000      0     255.255.255.0   0
;    IUCLM18B  2000      0     255.255.255.0   9.67.56.81
;    TONCP9L   2000      0     255.255.255.0   9.67.44.2
; ENDBSDROUTINGPARMS
;
```

Figure 128 (Part 8 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; ****************************************************************
; * TELNET DEFINITIONS:                                          *
; ****************************************************************
;
;
INTERNALCLIENTPARMS
  TIMEMARK 600        ; SET TELNET TIMEOUT TO 10 MINUTES
  SMFINIT  171        ; SMF RECORD TYPE 171 IS USED FOR LOGN RECORDS
  DBCSTRANSFORM       ; Load the JTELNET 3270 tranform module
  SMFTERM  172        ; SMF RECORD TYPE 172 IS USED FOR LOGF RECORDS
; PORT XXX            ; ACCEPTS INCOMING REQUESTS ON PORT XXX (DEF=23)
  INACTIVE 600        ; CLOSE TELNET CONNECTION AFTER SEC SECONDS
;                     ;  (DEFAULT  IS UNLIMITED)
  BINARYLINEMODE      ; Allow binary linemode telnet clients
; SMFEXIT             ; CALL EXIT TCPTNSMF BEFORE WRITING SMF RECORD
ENDINTERNALCLIENTPARMS
;
; *********************************************************************
; * VTAM definitions for TELNET SERVER
; *********************************************************************
;
; Define the VTAM parameters required for the TELNET server
BEGINVTAM
```

Figure 128 (Part 9 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; **********************************************************************
; * Define logon mode tables to be the defaults shipped with the lates
; * level of VTAM
; * The TELNET Device type is specified as part of the TN3270 protocol
; * The device type is mapped in the next table on a VTAM LOGMODE
; * entry as specified in the ISTINCLM file of VTAM.
; **********************************************************************
  LINEMODE INTERACT ; LINEMODETERMINAL, no logmode used.
  3277     D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3278-2-E NSX32702 ; 24x80 primary screen, no alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-2   D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3278-3-E NSX32703 ; 24x80 primary screen, 32x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-3   D4B32783 ; 24x80 primary screen 32x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M3 screen
  3278-4-E NSX32704 ; 24x80 primary screen, 43x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-4   D4B32784 ; 24x80 primary screen, 43x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M4 screen
  3278-5-E NSX32705 ; 24x80 primary screen, 27x132 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-5   D4B32785 ; 24x80 primary screen, 27x132 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M5 screen
  3279-2-E NSX32702 ; 24x80 primary screen, no alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-2   D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3279-3-E NSX32703 ; 24x80 primary screen, 32x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-3   D4B32783 ; 24x80 primary screen 32x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M3 screen
  3279-4-E NSX32704 ; 24x80 primary screen, 43x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-4   D4B32784 ; 24x80 primary screen, 43x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M4 screen
  3279-5-E NSX32705 ; 24x80 primary screen, 27x132 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-5   D4B32785 ; 24x80 primary screen, 27x132 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M5 screen
```

Figure 128 (Part 10 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; **********************************************************************
; * - Default LUs is a generic pool of LUs used to represent remote
; *   telnet clients when the following conditions are met:
; * -a- The IP address of the telnet client does not appear in IPGROUP
; *     statement
; * -b- The IP address of the telnet client does not appear in the LUMAP
; *     statement
; * - The telnet client is assigned to the first free LU in the list
; * - The LUs in this list are defined in the VTAM APPL Maj Node:
; *   RISC.VTAMLST(RAIBTCP)
; * - The LUs in this list and the LUGROUP list define ALL the
; *   available LUs for the telnet clients.
; * - The LUs specified on the ALLOWAPPL or RESTRICTAPPL
; *   statement must appear in this list
; * - The LUs specified on the LUMAP statement
; *   must appear in this list or in the LUGROUP list
; * - Define a range of LUs as follows:
; *   LUA..LUD= LUA, LUB, LUC, LUD
; *   LU0..LU5= LU0, LU1, LU2, LU3, LU4, LU5
; **********************************************************************
DEFAULTLUS
      RAIAE001 RAIAE002 RAIAE003 RAIAE004 RAIAE005
      RAIAE006 RAIAE007 RAIAE008 RAIAE009 RAIAE010
      RAIAE011 RAIAE012 RAIAE013 RAIAE014 RAIAE015
      RAIAE016 RAIAE017 RAIAE018 RAIAE019 RAIAE020
      RAIAE021 RAIAE022 RAIAE023 RAIAE024 RAIAE025
      RAIAE026 RAIAE027 RAIAE028 RAIAE029 RAIAE030
ENDDEFAULTLUS
;
; **********************************************************************
; * - Define LUGROUPs to which you can refer on the LUMAP statement
; * - Define a range of LUs as follows:
; *   LUA..LUD= LUA, LUB, LUC, LUD
; *   LU0..LU5= LU0, LU1, LU2, LU3, LU4, LU5
; * - LU RAIAE002..RAIAE006 belong to LUGROUP RISCLUG
; * - LU RAIAE007..RAIAE016 belong to LUGROUP TR1LUG
; **********************************************************************
;
; LUGROUP RISCLUG
;        RAIAE002 RAIAE003 RAIAE004 RAIAE005 RAIAE006
; ENDLUGROUP
;
; LUGROUP TR1LUG
;        RAIAE007..RAIAE016
; ENDLUGROUP
;
; **********************************************************************
; * - Define IPGROUPs to which you can refer on the LUMAP
; *   DEFAULTAPPL and LINEMODEAPPL statements
; * - IPGROUP consists of entire IP networks and or individual IP
; *   host addresses. Each IP address can appear only once within all
; *   IPGROUPS.
; * - IPGROUP TR1IPG defines all the hosts in the 9.24.104 network
; **********************************************************************
;
; IPGROUP TR1IPG
;        255.255.255.0:9.24.104.0
; ENDIPGROUP
```

Figure 128 (Part 11 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; *********************************************************************
; * - Use the LUMAP statement to map IP addresses to LU names
; * - You can use LUGROUPs and IPGROUPS to make mapping easier
; * - IP address 9.67.46.2 is mapped to LU RAIAE001
; * - IP address 9.24.104.28 is mapped to LUGROUP RISCLUG
; * - IP addresses in net 9.24.104.0 are mapped to LUGROUP TR1LUG
; *********************************************************************
; LUMAP RAIAE001 9.67.46.2
;
; LUMAP RISCLUG 9.24.104.28
;
; LUMAP TR1LUG TR1IPG
; LUMAP TR1LUG 9.24.104.17
;
; *********************************************************************
; * - Use the SSTAB to sen USS Message 10 to telnet client that
; *   uses 3270 device (eg TN3270)
; * - Define USSTAB per IP address, IPGROUP or link name.
; * - USSTCP is defined in RISC.VTAMLIB??? and presented to 9.24.104.28
; *********************************************************************
;  !!!!!! USSTAB is changed to USSTCP statement !!!!!!!!!!!
; USSTAB USSTCP  9.24.104.28  ; CNM20 and RAKAA for RS60002
  USSTCP USSABC  9.24.104.17  ; TSO  /    RAIAT for pc
  USSTCP USSABC  9.67.41.2    ; TSO  /    RAIAT for host sa3
; DEFAULTAPPL RAIAT             ; RAIAT for All nets,TSO18     -3270
; DEFAULTAPPL RAKAA TR1IPG    ; RAKAA for Token Ring Network -3270
; LINEMODEAPPL RAIAT EN1        ; RAIAT for Ethernet Network   -LineMode
; LINEMODEAPPL RAIAT            ; RAIAT for all nets           -LineMode
;
; *********************************************************************
; * ALLOW access to following list of applications:
; *********************************************************************
;
; ALLOWAPPL RA%%%%%% DISCONNECTABLE ; All applications starting with RA
;                                  ; followed by any 6 characters are
;                                  ; allowed and are disconnected when
;                                  ; session is dropped.
ALLOWAPPL *                        ; All other applications are allowed
;                                  ; and user is logged of when the
ALLOWAPPL RAIAT                    ; All other applications are allowed
;                                  ; session is dropped.
; *********************************************************************
; * Restrict access to following list of applications to users with PW
; *********************************************************************
;
; RESTRICTAPPL RA%%%%%% DISCONNECTABLE ; Applications starting with RA
; USER FONVILL                         ; are only accessible by USER,
; USER SADTLER                         ; provided that USER gives valid
ENDVTAM
```

Figure 128 (Part 12 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
; ----------------------------------------------------
; START CH1EN1              ;  3172-1 ETHERNET(MOD 1)
; START CH1TR1              ;  3172-1 T/R     (MOD 1)
; ----------------------------------------------------
; START X25DMV18            ;  X25
; START IUCDM18B            ;  IUCV TO SA18 TCPIP(B)
; START L62D1803            ;  LU 6.2 TO MVS03
; START L62D1811            ;  LU 6.2 TO MVS11
; START L62D1825            ;  LU 6.2 TO MVS25
; START L62DOS2             ;  LU 6.2 TO OS2 9.24.62.80
; START SNADMV03            ;  SNALINK LU0 device SA 03
; START TONCP9D             ;  SNALINK TO NCP IN SA09
; START CTCDMVTS            ;  CTC TO 03
  START MPEDEV              ;  3172-3 T/R     (OFF model 3)
; START CLAWDEV             ;  RS60007 ESCON
; end of profile
```

Figure 128 (Part 13 of 13). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.PROFILE.TCPIP

```
;*********************************************************************
;                                                                    *
;        TCP/IP MVS V3 at ITSO-Raleigh                               *
;        -----------------------------                               *
;                                                                    *
;   - This file is used on TCP/IP MVS V3 systems of:                 *
;                                                                    *
;     + MVS/ESA 4.1    (MVS18 / SA18)      ** RAIANJE  **            *
;                                                                    *
;                                                                    *
;                                                                    *
;************************************************** 94/12/02 *********
;                                                                    *
;   Name of File:     TCPIP.V3R1.RAIANJE.A.TCPPARMS(TCPDATA) /        *
;                     SYS1.TCPPARMS(TCPDATA)                         *
;                                                                    *
;                                                                    *
;   This data set, TCPIP.DATA, is used to specify configuration      *
;   information required by TCP/IP client programs.                  *
;                                                                    *
;   Syntax Rules for the TCPIP.DATA configuration data set:          *
;                                                                    *
;   (a) All characters to the right of and including a ';' will be   *
;       treated as a comment.                                        *
;                                                                    *
;   (b) Blanks and <end-of-line> are used to delimit tokens.          *
;                                                                    *
;   (c) The format for each configuration statement is:              *
;                                                                    *
;       <SystemName||':'>   keyword  value                           *
;                                                                    *
;       where <SystemName||':'> is an optional label which may be    *
;       specified before a keyword; if present, then the keyword-    *
;       value pair will only be recognized if the SystemName matches *
;       the node name of the system, as defined in the IEFSSNxx      *
;       PARMLIB member.  This optional label permits configuration   *
;       information for multiple systems to be specified in a single *
;       TCPIP.DATA data set.                                         *
;                                                                    *
;*********************************************************************
;
DATASETPREFIX TCPIP.V3R1.RAIANJE.A
;
; TCPIPuserid specifies the Name of the TCPIP address space and *not*
; the name of the USERID of the TCPIP address space!!!!
; TCPIP is used as the default name of the started procedure name.
;
TCPIPJOBNAME T18ATCP       ; ON MVS/ESA 4.2  (MVS18 / SA18)
;
MESSAGECASE MIXED
;
```

Figure 129 (Part 1 of 3). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.TCPPARMS(TCPDATA)

```
; HostName specifies the TCP host name of this system.  If not
; specified, the default HostName will be the node name specified
; in the IEFSSNxx PARMLIB member.
;
; For example, if this TCPIP.DATA data set is shared between two
; systems, MVSXA and MVSESA, then the following two lines will define
; the HostName correctly on each system.
;
HostName mvs18
```

Figure 129 (Part 2 of 3). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.TCPPARMS(TCPDATA)

```
;
; DomainOrigin specifies the domain origin that will be appended
; to host names passed to the resolver.  If a host name contains
; any dots, then the DomainOrigin will not be appended to the
; host name.
;
DomainOrigin itso.ral.ibm.com
;
; NSinterAddr specifies the internet address of the Name Server.
; Multiple Name Server addresses may be specified.  The Name Servers
; will be tried in the given order.
; If a Name Server will not be used, then do not code an NSinterAddr
; statement.  This will cause all names to be resolved via host table
; lookup.
;
; NSinterAddr  14.0.0.0
; NSinterAddr  9.24.104.126
; NSinterAddr  129.34.128.61
; NSinterAddr  9.24.104.74
; NSinterAddr  9.19.141.242
; NSinterAddr  9.24.1.9
; NSinterAddr  9.67.38.93
;
; NSportAddr specifies the Name Server port.
; 53 is the default value.
;
NSportAddr 53
;
; ResolveVia specifies how the Resolver is to communicate
; with the Name Server. TCP indicates use of TCP virtual circuits.
; UDP indicates use of UDP Datagrams.
; The default is UDP.
;
ResolveVia UDP
;
; ResolverTimeout specifies the time in seconds that the Resolver
; will wait while trying to open a TCP connection to the name server,
; or how long it will wait for a response when using UDP.
; Default is 20 seconds.
;
ResolverTimeout 20
;
; ResolverUdpRetries specifies the number of times the resolver should
; retry a query to the Nameserver when using UDP datagrams.
; Default is 1.
;
ResolverUdpRetries 1
;
Trace Resolver
;
; End of file.
```

Figure 129 (Part 3 of 3). Definitions for TCP/IP on SA18 - Dataset TCPIP.V3R1.RAIANJE.TCPPARMS(TCPDATA)

A.2 VTAM Definitions for SA18

```
*******************************************************************
*                                                                 *
*          VTAM 42   ANYNET   -      SNA OVER TCP/IP              *
*                                                                 *
*                     SA 18  DEFINITIONS                          *
*                                                                 *
*******************************************************************
RAIBSNIP VBUILD  TYPE=TCP,                                          X
               CONTIMER=30,            WAIT FOR MPTN TO COME UP     X
               DGTIMER=30,             INTERVAL BETWEEN RETRIES     X
               DNSUFX=IBM.COM,         DOMAIN NAME SUFFIX           X
               EXTIMER=3,              BETW. SEND SNA EXPEDITED DATA X
               IATIMER=120,            TIME BEFORE MPTN KEEPALIVE   X
               PORT=397,               WELLKNOWN PORT FOR ANYNET    X
               TCB=10,                 NUMBER MVS SUBTASKS          X
               TCPIPJOB=T18ATCP        TCP/IP JOBNAME
RAIGSNIP GROUP ISTATUS=ACTIVE          GROUPNAME
RAILSNIP LINE  ISTATUS=ACTIVE          LINENAME
RAIPSNIP PU    ISTATUS=ACTIVE          PUNAME
```

Figure 130. AnyNet/MVS TCP/IP Major Node Definitions on SA18 - Member RAIBSNIP in RISC.VTAMLST

```
*******************************************************************
*                                                                 *
*       VTAM  42  ANYNET SNA OVER TCP/IP                          *
*                                                                 *
*                     SA 18  DEFINITIONS                          *
*                                                                 *
*NAME    CDRSC ALSLIST=.....NAME OF THE PU STATEMENT DEFINED      *
*                            WITHIN THE VBUILD TYPE=TCP           *
*                                                                 *
*     - THE NAME LABEL OF THE CDRSC DEFINITION STATEMENT MUST BE  *
*       THE REMOTE ILU NAME.                                      *
*                                                                 *
*     - WE MUST CODE ALSREQ=YES TO USE THE PREDEFINED LIST.       *
*                                                                 *
*     - WE USE SOME CDRSC WITH THE NETID NOT CODED IN ORDER TO    *
*       THE CDRMNAME AS AN ADJSSCP.                               *
*                                                                 *
*******************************************************************
         VBUILD  TYPE=CDRSC
*
         NETWORK NETID=USIBMRA
*
RABAC    CDRSC ALSLIST=RAIPSNIP,ALSREQ=YES  CICS SA11
RA3AC    CDRSC ALSLIST=RAIPSNIP,ALSREQ=YES  CICS SA03
RALYBTCP CDRSC ALSLIST=RAIPSNIP,ALSREQ=YES  AS400
RAPAC    CDRSC CDRM=RAP                     CICS SA25
```

Figure 131. CDRSCS Major Node Definitions for Remote ILUs on SA18 - Member RAIRSNIP in RISC.VTAMLST

A.3 TCP/IP Definitions for SA11

```
; -------------------------------------------------------------------*
; -------------------------------------------------------------------*
;     ITSC-Raleigh:  TCP/IP MVS V3 System  *RABANJE*                 *
;     ---------------------------------------------                  *
;   - MVS/ESA 4.3 - MVS11 / SA11                                     *
;   - Name of TCPIP Configuration File =                             *
;     TCPIP.V3R1.RABANJE.A.PROFILE.TCPIP / SYS1.TCPPARMS(T11APROF)   *
; ------------------------------------------------------- 94/09/14 ----*
; ****************************************************************
; * DATA BUFFER DEFINITIONS:                                         *
; ****************************************************************
;                                         | size | default | max    | max    |
;                                         |(byte)| nbr     | nbr    | size   |
;                                         |======|=========|========|=======|
; TABUFFERPOOLSIZE          100 8192    ; | 8192 |   160   |   VS   | 262144|
DATABUFFERPOOLSIZE          100 28672   ; | 8192 |   160   |   VS   | 262144|
SMALLDATABUFFERPOOLSIZE     1200        ; | 2048 |     0   |   VS   |       |
TINYDATABUFFERPOOLSIZE      500         ; |  256 |     0   |   VS   |       |
;
;
; Note: VS=Limited by the amount of available Virtual Storage
;
; ****************************************************************
; * FREE BUFFER POOL DEFINITIONS:                                 *
; ****************************************************************
;                                         | size | default | max    | max    |
;                                         |(byte)| nbr     | nbr    | size   |
;                                         |======|=========|========|=======|
ACBPOOLSIZE                 1000        ; |   96 |  1000   |        |       |
CCBPOOLSIZE                 300         ; | 1184 |   150   |   VS   |       |
RCBPOOLSIZE                 50          ; |  228 |    50   |   VS   |       |
SKCBPOOLSIZE                256         ; |  781 |   256   |   VS   |       |
IPROUTEPOOLSIZE             300         ; |  128 |   600   |   VS   |       |
ADDRESSTRANSLATIONPOOLSIZE  1500        ; |  153 |  1500   |   VS   |       |
ENVELOPEPOOLSIZE            750         ; | 2048 |   750   |   VS   |       |
LARGEENVELOPEPOOLSIZE       100 32768   ; | 8192 |    50   |   VS   |       |
SCBPOOLSIZE                 256         ; |  228 |   256   |   VS   |       |
TCBPOOLSIZE                 256         ; | 1056 |   256   |   VS   |
UCBPOOLSIZE                 100         ; |  245 |   100   |   VS   |
;
;
; Note: VS=Limited by the amount of available Virtual Storage
;
```

Figure 132 (Part 1 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ****************************************************************
; * ASSORTED Parameters, which have scope of entire TCP/IP engine*
; ****************************************************************
ASSORTEDPARMS       ;
; ALWAYSWTO         ; Issue WTO messages for the TCP/IP server messages
; NOFWD             ; All gateway functions of TCP/IP are disabled
; IGNOREREDIRECT    ; Ignore ICMP redirect packets (like ?)
; NOUDPQUEUELIMIT   ; Relaxes incoming UDP datagram queuesize (def=21)
; NOACBCUSHION      ; Disables dynamic allocation of ACBs.
  MESSAGECASE MIXED ; TCP/IP Messages will be displayed in mixed case.
  TCPIPSTATISTICS   ; Print TCP/IP counters.
ENDASSORTEDPARMS    ;
;
; ****************************************************************
; * KEEP alive options, defining the contents and timing of      *
; * KEEP alive messages sent over a TCP connection               *
; ****************************************************************
; KEEPALIVEOPTIONS    ;
;  INTERVAL 120       ; send message every 120 minutes
;  SENDGARBAGE TRUE   ; send random data and invalid sequence number
; ENDKEEPALIVEOPTIONS ;
;
; ****************************************************************
; * TRACE  DEFINITIONS: (NEEDS TO BE CUSTOMIZED)                 *
; ****************************************************************
;
; TRACE RESOLVER SCREEN
; TRACE ARP
; MORETRACE ARP
; MORETRACE CLAW
; FILE 'JOAO.TRACE.DATA'
; FILE 'TCPIP.V3R1.RABANJE.A.TRACE'
; TRACE SOCKET NOSCREEN
; MORETRACE SOCKET NOSCREEN
; MORETRACE TELNET
; TRACE TELNET MORETRACE TELNET NOSCREEN
; MORETRACE ALL NOSCREEN
; MORETRACE PCCA NOSCREEN
; TIMESTAMP PREFIX            ; TIMESTAMP PREFIXES EVERY MESSAGE
; PKTTRACE                    ; START IP PACKET TRACING, GTF must be
;                             ; started and TRACE of process too.
;  LINKNAME=*                 ; ALL LINKS
;  PROT=TCP                   ; PROTOCOL is TCP
;  IP=9.24.104.17             ; TRACE IP in and out to this address only
;  SUBNET=255.255.255.0       ; USED subnetmask
;  SRCPORT=*                  ; UDP or TCP source port to be traced
;  DESTPORT=*                 ; UDP or TCP destination port to be traced
;  FULL                       ; Trace entire packet
; PKTTRACE CLEAR              ; STOP IP PACKET TRACING
;  LINKNAME=*                 ; ALL LINKS
; ****************************************************************
; * SYSTEM ADMINISTRATION:                                       *
; ****************************************************************
;
;   ****************************************************************
;   * INFORM FOLLOWING USERS ABOUT SERIOUS RUN-TIME CONDITIONS     *
;   ****************************************************************
INFORM
  MICHAEL LI
ENDINFORM
;
```

Figure 132 (Part 2 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
;   ****************************************************************
;   * ALLOW USE OF OBEYFILE COMMAND FOR FOLLOWING USERS:           *
;   ****************************************************************
;
OBEY
  STCTCP1                    ; Real RACF User Id's of SNMPQE SNMPD
  T11AROUT                   ; ROUTED
  T11ASNMQ                   ; SNMPQE
  T11ASNMD                   ; SNMPD
  SNMPQE                     ; Dummy RACF ID for SNMPQE
  SNMPD18                    ; Dummy RACF ID for SNMPD
  T11ANCPR                   ; NCP route server
  STUSER                     ; User on which NCPROUTE is running
  ALFREDC                    ; Alfred
  MCLI                       ; Michael Li
  Purring                    ; John Purring
ENDOBEY
;
;   ****************************************************************
;   * RESTRICT USE OF TCPIP SERVICES FOR FOLLOWING USERS:          *
;   ****************************************************************
;
RESTRICT
; HACKER1                    ; RESTRICT User Id's from TCPIP Services
; HACKER2                    ; Which services????
ENDRESTRICT                  ;
;
;   ****************************************************************
;   * SMF LOGGING OF TCPIP APPLICATIONS                            *
;   ****************************************************************
;
  SMFPARMS 121 122 123
;  121           ; sub type field in API initialization
;  122           ; sub type field in API termination
;  123           ; sub type field for FTP or TELNET client
;  LAODEXIT      ; Call TCPCNSMF
;
;
```

Figure 132 (Part 3 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
;   ****************************************************************
;   * PROBLEMS? CALL YOUR SYSTEM CONTACT: SNMP OBJECT SYSCONTACT   *
;   ****************************************************************
;
SYSCONTACT
  Michael Li 301-2323
ENDSYSCONTACT
;
;
;   ****************************************************************
;   * KNOW WHERE YOUR SYSTEM IS RUNNING: SNMP OBJECT SYSLOCATION   *
;   ****************************************************************
;
SYSLOCATION
  MVS11 ITSO RALEIGH
ENDSYSLOCATION
;
;
```

Figure 132 (Part 4 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
;   ****************************************************************
;   * ARP TABLE ENTRIES FLUSHED AFTER THIS TIME LIMIT EXPIRES:     *
;   ****************************************************************
;
; ARPAGE N      ; AFTER N MINUTES, THE CREATED OR REVALIDATED ARP
;               ; TABLE ENTRY IS DELETED.
;
; ****************************************************************
; * DATASET prefix for search order of server and client         *
; * configuration files                                          *
; ****************************************************************
;
DATASETPREFIX TCPIP.V3R1.RABANJE.A
; ****************************************************************
; * HARDWARE DEFINITIONS:                                        *
; ****************************************************************
;
;
;
;  **************************************************************
;  *    3172 Model 3 attachments: SNACOMM and Offload
;  **************************************************************
;          ================================
;          |                              |
;          |       TCPIP11                |
;          | IP addr: 9.67.38.11  = OFFTR1|
;          | IP addr: 9.67.32.21  = OFFEN1|
;          | IP addr: 9.67.46.89  = OFFWAC|
;          --------------------------------
;      subch addr || 340-35F
;     ESCON ChPID || 18
;          +----+----+----+
;     9032 !    ! E1 !    !
;          !    +----+    !
;    ESCON !              !
; Director !    +----+    !
;          !    ! E4 !    !
;          +----+----+----+
;                 ||
;                 ||
;                 ||
;        ------------------------+
; ESCON  |       340              |
; slot   |       1-2              |
;        |    3172 Model 3        |
;        |   SNACOMM and Offload  |
; Dev to |OFFEN1 OFFTR1 OFFWAC    |
; Rel adp|  0     0      0        |
; number |                        |
; Slot   |  6     7      4        |
;        ------------------------+
; MACs:     |     |      |
;           |     |      4000 0023 172A
;           |     4000 0023 172C
;           4000 0023 172B
;
```

Figure 132 (Part 5 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ****************************************************************
; * Device and link definitions for 3172 C   - SNACOMM and Offload
; ****************************************************************
;
  DEVICE OFFDEV    CLAW  340  TCPIP  OS2TCP  NONE
  LINK   OFFIPLINK OFFLOADLINK1  1  OFFDEV
; LINK   OFFWAC  OFFLOADAPIBROAD 9.67.46.89     OFFDEV  OFFIPLINK
  LINK   OFFEN1  OFFLOADAPIBROAD 9.67.32.21     OFFDEV  OFFIPLINK
  LINK   OFFTR1  OFFLOADAPIBROAD 9.67.38.11     OFFDEV  OFFIPLINK
; *********************************************************************
; * TO START THE PROCEDURES (HIGHER LEVEL SERVICES) USE AUTOLOG.
; *********************************************************************
;   These procedures can be found in TCPIP.PROCLIB
;
;
AUTOLOG
;  T11AMISC                         ; Miscelaneous Server
;  T11ASNO                          ; SNALINK TO MVS03
;  T11ASNOX                         ; SNALINK TO TCPIP(B)
;  T11ASNOA                         ; SNAlink dummy AB
;  T11ASNOB                         ; SNAlink dummy BA
;  T11AFTPP                         ; FTP server - Pascal
;  T11AFTPC                         ; FTP server - C
;  T11AROUT                         ; ROUTED server
;  T11ASNMQ                         ; SNMP Query Engine
;  T11ASNMD                         ; SNMP Agent
;  T11APORT                         ; Port Mapper
;  T11ALK09                         ; SNAlink to NCP IP router in SA=09
;  T11ANCPR                         ; NCP Route server
;  T11ALU62                         ; SNAlink LU 6.2 TO MVS03
;  T11AOS62                         ; SNAlink LU 6.2 TO OS2
;  T11ASMTP                         ; SMTP server
;  T11AX25                          ; X.25 NPSI support
;  T11ADNS                          ; Name Server
;  T11ANFS                          ; NFS Server
;  T11ALPD                          ; LPD Server
;  T11ANDBP                         ; NDB Portmap manager
;  T11ANDB1                         ; NDB Server 1
;  T11ALLBD                         ; NCS Local location broker
;  T11AGLBD                         ; NCS Global location broker
;  T11ANDB2                         ; NDB Server 2
;  T11ARXEC                         ; Remote Execution Server
ENDAUTOLOG
;
```

Figure 132 (Part 6 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ********************************************************************
; * THE PORT NUMBER assignment for the procedures (applications)
; * the PROTOCOL used on the port.
; ********************************************************************
;   These procedures can be found in TCPIP.PROCLIB
;   Port values from RFC 1010, "Assigned numbers"
;
PORT
    7 UDP T11AMISC              ; MISCSERVer
    7 TCP T11AMISC              ; MISCSERVer
    9 UDP T11AMISC              ; MISCSERVer
    9 TCP T11AMISC              ; MISCSERVer
   19 UDP T11AMISC              ; MISCSERVer
   19 TCP T11AMISC              ; MISCSERVer
   20 TCP T11AFTPC              ; FTP C server data port
          NOAUTOLOG             ; Do not restart if stopped previously
;         DELAYACKS             ; Delay transmission acknowledgements
   21 TCP T11AFTPC              ; FTP C server control port
   23 TCP INTCLIEN              ; Telnet server,same port number as
                                ; on INTERNALCLIENTPARMS statement
   25 TCP T11ASMTP              ; SMTP server
   53 TCP T11ADNS               ; Name Server
   53 UDP T11ADNS               ; Name Server
   111 UDP T11APORT             ; Portmapper Server
   111 TCP T11APORT             ; Portmapper Server
   125 UDP T11ALLBD             ; NCS Local Location Broker
   161 UDP T11ASNMD             ; Agent Port for SNMP Messages
   162 UDP T11ASNMQ             ; SNMP Client port for Trap's
;  xxx TCP xxxx NOAUTOLOG       ; CSFI INTERFACE
   512 TCP T11ARXEC             ; Remote Execution
   514 TCP T11ARXEC             ; Remote Execution
   515 TCP T11ALPD              ; Remote Printer Server
   520 UDP T11AROUT             ; RouteD Server
   580 UDP T11ANCPR             ; NCPROUTE for RIP support on NCPs
  2049 UDP T11ANFS              ; NFS Server
  3000 TCP CICSTCP              ; CICS Sockets
  3001 TCP CICSTCP              ; CICS Sockets
  3002 TCP T11AIMSL             ; IMS Sockets Dave Herr sample
  3010 TCP IMS18TCP             ; IMS Sockets
  3011 TCP IMS18TCP             ; IMS Sockets
; 2221 TCP OFFLFTP              ; reserved port for Offload FTP
; 2223 TCP OFFLTN               ; reserved port for Offload telnet
; 4000 TCP MPEBOX               ; reserved port for MPE box telnet
  7020 TCP T11AFTPP NOAUTOLOG   ; FTP Pascal server data port
  7021 TCP T11AFTPP             ; FTP Pascal server control port
; ********************************************************************
; * HOME IP addresses for the LINKS of TCPIP in MVS11
; ********************************************************************
; The first IP address in the home link is used as source address
; in IP datatgrams sent to non-adjacent nodes, unless the statement
; PRIMARYINTERFACE is specified, in whcih case the IP address of this
; primary interface is used.
;
HOME  ; The Local Host's
; Local host's Internet addresses
;   9.67.38.11     OFFTR1      ; OFFload TR1 interface
```

Figure 132 (Part 7 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ********************************************************************
; * PRIMARYINTERFACE STATEMENT for TCPIP in MVS11
; ********************************************************************
; The first IP address in the home link is used as source address
; in IP datatgrams sent to non-adjacent nodes, unless the statement
; PRIMARYINTERFACE is specified, in which case the IP address of this
; primary interface is used.
;
PRIMARYINTERFACE   OFFTR1
;
;
; ********************************************************************
; * TRANSLATE STATEMENT FOR IP ADDRESS to LINK ADDRESS TRANSLATION
; ********************************************************************
; TRANSLATE
;   9.67.56.1  HCH
;   ttxxxxxxhhcc
;   HCHLINK1
;   9.67.32.4  ETHERNET
;   201020304050
;   EN4
;   9.67.38.4  IBMTR
;   401020304050
;   TR4
;   9.67.40.4  FDDI
;   400000111111
;   FD4
;
;
; *******************************************************************
; * GATEWAY statements for static routing
; *******************************************************************
;
GATEWAY
  9.0.0.0      =        OFFTR1 DEFAULTSIZE  0.255.255.0  0.67.38.0
; 9.0.0.0      =           TR1 DEFAULTSIZE  0.255.255.0  0.12.11.0
; 9.0.0.0      =           TR1 DEFAULTSIZE  0.255.255.0  0.12.12.0
; *******************************************************************
; * Anything not routed above, give to router 9.24.104.1, via TR1
; *******************************************************************
  DEFAULTNET 9.67.38.10  OFFTR1   DEFAULTSIZE  0
; DEFAULTNET 9.12.11.4      TR1   DEFAULTSIZE  0.255.255.0      0
;
; ***************************************************************
; MAXIMUMRETRANSMITTIME 60.00 ; wait time for ACK before resending packet
; MINIMUMRETRANSMITTIME 00.75 ; The formula is:
; ROUNDTRIPGAIN         00.12 ; ?
; VARIANCEGAIN          00.25 ; ?
; VARIANCEMULTIPLIER    20.00 ; ?
; DELAYACKS             20.00 ; Delay of ACKs for TCP connections.
;
```

Figure 132 (Part 8 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ********************************************************************
; * RouteD Routing statements for dynamic routing
; * Static definitions are in SYS1.TCPPARMS(GATEWAYS)
; *
; ********************************************************************
; BSDROUTINGPARMS false    ; default max mtu size of 576 bytes is used
;                          for packets to non-adjacent networks.
;    LINK     maxmtu   METRIC   SUBNET MASK     DESTINATION ADDRESS
;                                               (point to point links)
;    CTCLMVTS  2000      0     255.255.255.0   9.67.47.2
;    CTCLMVTS  2000      0     255.255.255.0   9.67.89.135  ?????????
;    L62LMV03  2000      0     255.255.255.0   9.67.41.2
;    L62LMV03  2000      0     255.255.255.0   9.67.41.26   ?????????
;    SNALMV03  2000      0     255.255.255.0   9.67.41.2
;    EN1       2000      0     255.255.255.0   0
;    OFFEN1    2000      0     255.255.255.0   0
;    OFFTR1    2000      0     255.255.255.0   0
;    L62LOS2   2000      0     255.255.255.0   9.24.62.80
;    TR1       2000      0     255.255.255.0   0
;    X25LMV18  2000      0     255.255.255.0   0
;    IUCLM18B  2000      0     255.255.255.0   9.67.56.81
;    TONCP9L   2000      0     255.255.255.0   9.67.44.2
; ENDBSDROUTINGPARMS
;
; ****************************************************************
; * TELNET DEFINITIONS:                                          *
; ****************************************************************
;
;
INTERNALCLIENTPARMS
  TIMEMARK 600        ; SET TELNET TIMEOUT TO 10 MINUTES
  SMFINIT  171        ; SMF RECORD TYPE 171 IS USED FOR LOGN RECORDS
  DBCSTRANSFORM       ; Load the JTELNET 3270 tranform module
  SMFTERM  172        ; SMF RECORD TYPE 172 IS USED FOR LOGF RECORDS
; PORT XXX            ; ACCEPTS INCOMING REQUESTS ON PORT XXX (DEF=23)
  INACTIVE 600        ; CLOSE TELNET CONNECTION AFTER SEC SECONDS
;                     ;  (DEFAULT  IS UNLIMITED)
  BINARYLINEMODE      ; Allow binary linemode telnet clients
; SMFEXIT             ; CALL EXIT TCPTNSMF BEFORE WRITING SMF RECORD
ENDINTERNALCLIENTPARMS
;
```

Figure 132 (Part 9 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; *********************************************************************
; * VTAM definitions for TELNET SERVER
; *********************************************************************
;
; Define the VTAM parameters required for the TELNET server
BEGINVTAM
; *********************************************************************
; * Define logon mode tables to be the defaults shipped with the lates
; * level of VTAM
; * The TELNET Device type is specified as part of the TN3270 protocol
; * The device type is mapped in the next table on a VTAM LOGMODE
; * entry as specified in the ISTINCLM file of VTAM.
; *********************************************************************
  LINEMODE INTERACT ; LINEMODETERMINAL, no logmode used.
  3277     D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3278-2-E NSX32702 ; 24x80 primary screen, no alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-2   D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3278-3-E NSX32703 ; 24x80 primary screen, 32x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-3   D4B32783 ; 24x80 primary screen 32x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M3 screen
  3278-4-E NSX32704 ; 24x80 primary screen, 43x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-4   D4B32784 ; 24x80 primary screen, 43x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M4 screen
  3278-5-E NSX32705 ; 24x80 primary screen, 27x132 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3278-5   D4B32785 ; 24x80 primary screen, 27x132 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M5 screen
  3279-2-E NSX32702 ; 24x80 primary screen, no alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-2   D4B32782 ; 24x80 primary screen no alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M2 screen
  3279-3-E NSX32703 ; 24x80 primary screen, 32x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-3   D4B32783 ; 24x80 primary screen 32x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M3 screen
  3279-4-E NSX32704 ; 24x80 primary screen, 43x80 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-4   D4B32784 ; 24x80 primary screen, 43x80 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M4 screen
  3279-5-E NSX32705 ; 24x80 primary screen, 27x132 alternate screen
                    ; Non-SNA 3270 extended datastreaming
  3279-5   D4B32785 ; 24x80 primary screen, 27x132 Alternate screen
                    ; Local Non-SNA 3274 model 1B/1D or 3274/6 BSC, M5 screen
;
```

Figure 132 (Part 10 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; *********************************************************************
; * - Default LUs is a generic pool of LUs used to represent remote
; *   telnet clients when the following conditions are met:
; * -a- The IP address of the telnet client does not appear in IPGROUP
; *     statement
; * -b- The IP address of the telnet client does not appear in the LUMAP
; *     statement
; * - The telnet client is assigned to the first free LU in the list
; * - The LUs in this list are defined in the VTAM APPL Maj Node:
; *   RISC.VTAMLST(RABBTCP)
; * - The LUs in this list and the LUGROUP list define ALL the
; *   available LUs for the telnet clients.
; * - The LUs specified on the ALLOWAPPL or RESTRICTAPPL
; *   statement must appear in this list
; * - The LUs specified on the LUMAP statement
; *   must appear in this list or in the LUGROUP list
; * - Define a range of LUs as follows:
; *   LUA..LUD= LUA, LUB, LUC, LUD
; *   LU0..LU5= LU0, LU1, LU2, LU3, LU4, LU5
; *********************************************************************
DEFAULTLUS
      RABAE001 RABAE002 RABAE003 RABAE004 RABAE005
      RABAE006 RABAE007 RABAE008 RABAE009 RABAE010
      RABAE011 RABAE012 RABAE013 RABAE014 RABAE015
      RABAE016 RABAE017 RABAE018 RABAE019 RABAE020
      RABAE021 RABAE022 RABAE023 RABAE024 RABAE025
      RABAE026 RABAE027 RABAE028 RABAE029 RABAE030
ENDDEFAULTLUS
;
; *********************************************************************
; * - Define LUGROUPs to which you can refer on the LUMAP statement
; * - Define a range of LUs as follows:
; *   LUA..LUD= LUA, LUB, LUC, LUD
; *   LU0..LU5= LU0, LU1, LU2, LU3, LU4, LU5
; * - LU RABAE002..RABAE006 belong to LUGROUP RISCLUG
; * - LU RABAE007..RABAE016 belong to LUGROUP TR1LUG
; *********************************************************************
;
; LUGROUP RISCLUG
;        RABAE002 RABAE003 RABAE004 RABAE005 RABAE006
; ENDLUGROUP
;
; LUGROUP TR1LUG
;        RABAE007..RABAE016
; ENDLUGROUP
;
; *********************************************************************
; * - Define IPGROUPs to which you can refer on the LUMAP
; *   DEFAULTAPPL and LINEMODEAPPL statements
; * - IPGROUP consists of entire IP networks and or individual IP
; *   host addresses. Each IP address can appear only once within all
; *   IPGROUPS.
; * - IPGROUP TR1IPG defines all the hosts in the 9.24.104 network
; *********************************************************************
;
; IPGROUP TR1IPG
;        255.255.255.0:9.24.104.0
; ENDIPGROUP
```

Figure 132 (Part 11 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; *********************************************************************
; * - Use the LUMAP statement to map IP addresses to LU names
; * - You can use LUGROUPs and IPGROUPS to make mapping easier
; * - IP address 9.67.46.2 is mapped to LU RABAE001
; * - IP address 9.24.104.28 is mapped to LUGROUP RISCLUG
; * - IP addresses in net 9.24.104.0 are mapped to LUGROUP TR1LUG
; *********************************************************************
; LUMAP RABAE001 9.67.46.2
;
; LUMAP RISCLUG 9.24.104.28
;
; LUMAP TR1LUG TR1IPG
; LUMAP TR1LUG 9.24.104.17
;
; *********************************************************************
; * - Use the SSTAB to sen USS Message 10 to telnet client that
; *   uses 3270 device (eg TN3270)
; * - Define USSTAB per IP address, IPGROUP or link name.
; * - USSTCP is defined in RISC.VTAMLIB??? and presented to 9.24.104.28
; *********************************************************************
;  !!!!!! USSTAB is changed to USSTCP statement !!!!!!!!!!!
; USSTAB USSTCP  9.24.104.28  ; CNM20 and RAKAA for RS60002
  USSTCP USSABC  9.24.104.17  ; TSO  /    RABAT for pc
  USSTCP USSABC  9.67.41.2    ; TSO  /    RABAT for host sa3
; DEFAULTAPPL RABAT             ; RABAT for All nets,TSO18     -3270
; DEFAULTAPPL RAKAA TR1IPG    ; RAKAA for Token Ring Network -3270
; LINEMODEAPPL RABAT EN1        ; RABAT for Ethernet Network   -LineMode
; LINEMODEAPPL RABAT            ; RABAT for all nets           -LineMode
;
```

Figure 132 (Part 12 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
; ********************************************************************
; * ALLOW access to following list of applications:
; ********************************************************************
;
; ALLOWAPPL RA%%%%%% DISCONNECTABLE ; All applications starting with RA
;                                   ; followed by any 6 characters are
;                                   ; allowed and are disconnected when
;                                   ; session is dropped.
ALLOWAPPL *                         ; All other applications are allowed
;                                   ; and user is logged of when the
ALLOWAPPL RABAT                     ; All other applications are allowed
;                                   ; session is dropped.
; ********************************************************************
; * Restrict access to following list of applications to users with PW
; ********************************************************************
;
; RESTRICTAPPL RA%%%%%% DISCONNECTABLE ; Applications starting with RA
; USER MCLI                            ; are only accessible by USER,
; USER SADTLER                         ; provided that USER gives valid
ENDVTAM
; ----------------------------------------------------
; START CH1EN1              ;  3172-1 Ethernet(MOD 1)
; START CH1TR1              ;  3172-1 T/R     (MOD 1)
; ----------------------------------------------------
; START X25DMV18            ;  X25
; START IUCDM18B            ;  IUCV TO SA11 TCPIP(B)
; START L62D1803            ;  LU 6.2 TO MVS03
; START L62D1811            ;  LU 6.2 TO MVS11
; START L62D1825            ;  LU 6.2 TO MVS25
; START L62DOS2             ;  LU 6.2 TO OS2 9.24.62.80
; START SNADMV03            ;  SNALINK LU0 device SA 03
; START TONCP9D             ;  SNALINK to NCP in SA09
; START CTCDMVTS            ;  CTC TO 03
  START OFFDEV              ;  3172-3 T/R     (OFF MODEL 3)
; START CLAWDEV             ;  RS60007 ESCON
; end of profile
```

Figure 132 (Part 13 of 13). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.PROFILE.TCPIP

```
;*********************************************************************
;                                                                    *
;        TCP/IP MVS V3 at ITSO-Raleigh                               *
;        -----------------------------                               *
;                                                                    *
;   - This file is used on TCP/IP MVS V3 systems of:                 *
;                                                                    *
;     + MVS/ESA 4.1    (MVS11 / SA11)      ** RABANJE  **            *
;                                                                    *
;                                                                    *
;************************************************** 94/11/18 *********
;                                                                    *
;   Name of File:     TCPIP.V3R1.RABANJE.A.TCPPARMS(TCPDATA) /       *
;                     SYS1.TCPPARMS(TCPDATA)                         *
;                                                                    *
;                                                                    *
;   This data set, TCPIP.DATA, is used to specify configuration      *
;   information required by TCP/IP client programs.                  *
;                                                                    *
;   Syntax Rules for the TCPIP.DATA configuration data set:          *
;                                                                    *
;   (a) All characters to the right of and including a ';' will be   *
;       treated as a comment.                                        *
;                                                                    *
;   (b) Blanks and <end-of-line> are used to delimit tokens.         *
;                                                                    *
;   (c) The format for each configuration statement is:              *
;                                                                    *
;       <SystemName||':'>   keyword  value                           *
;                                                                    *
;       where <SystemName||':'> is an optional label which may be    *
;       specified before a keyword; if present, then the keyword-    *
;       value pair will only be recognized if the SystemName matches *
;       the node name of the system, as defined in the IEFSSNxx      *
;       PARMLIB member.  This optional label permits configuration   *
;       information for multiple systems to be specified in a single *
;       TCPIP.DATA data set.                                         *
;                                                                    *
;*********************************************************************
;
;
DATASETPREFIX TCPIP.V3R1.RABANJE.A
;
;
; TCPIPuserid specifies the Name of the TCPIP address space and *not*
; the name of the USERID of the TCPIP address space!!!!
; TCPIP is used as the default name of the started procedure name.
;
TCPIPJOBNAME T11ATCP       ; ON MVS/ESA 4.2  (MVS11 / SA11)
;
MESSAGECASE MIXED
;
; HostName specifies the TCP host name of this system.  If not
; specified, the default HostName will be the node name specified
; in the IEFSSNxx PARMLIB member.
;
; For example, if this TCPIP.DATA data set is shared between two
; systems, MVSXA and MVSESA, then the following two lines will define
; the HostName correctly on each system.
;
```

Figure 133 (Part 1 of 2). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.TCPPARMS(TCPDATA)

```
HostName MVS11
;
;
; DomainOrigin specifies the domain origin that will be appended
; to host names passed to the resolver.  If a host name contains
; any dots, then the DomainOrigin will not be appended to the
; host name.
;
DomainOrigin itso.ral.ibm.com
;
; NSinterAddr specifies the internet address of the Name Server.
; Multiple Name Server addresses may be specified.  The Name Servers
; will be tried in the given order.
; If a Name Server will not be used, then do not code an NSinterAddr
; statement.  This will cause all names to be resolved via host table
; lookup.
;
; NSinterAddr  14.0.0.0
; NSinterAddr  9.67.38.11
; NSinterAddr  129.34.128.61
; NSinterAddr  9.24.104.74
; NSinterAddr  9.19.141.242
; NSinterAddr  9.24.1.9
; NSinterAddr  9.67.38.93
;
; NSportAddr specifies the Name Server port.
; 53 is the default value.
;
;
NSportAddr 53
;
; ResolveVia specifies how the Resolver is to communicate
; with the Name Server. TCP indicates use of TCP virtual circuits.
; UDP indicates use of UDP Datagrams.
; The default is UDP.
;
ResolveVia UDP
;
; ResolverTimeout specifies the time in seconds that the Resolver
; will wait while trying to open a TCP connection to the name server,
; or how long it will wait for a response when using UDP.
; Default is 20 seconds.
;
ResolverTimeout 20
;
; ResolverUdpRetries specifies the number of times the resolver should
; retry a query to the Nameserver when using UDP datagrams.
; Default is 1.
;
ResolverUdpRetries 1
;
; Trace Resolver
;
;
;
; End of file.
```

Figure 133 (Part 2 of 2). Definitions for TCP/IP on SA11 - Dataset TCPIP.V3R1.RABANJE.TCPPARMS(TCPDATA)

A.4 VTAM Definitions for SA11

```
*********************************************************************
*                                                                   *
*           VTAM 42   ANYNET   -      SNA OVER TCP/IP               *
*                                                                   *
*                      SA 11  DEFINITIONS                           *
*                                                                   *
*********************************************************************
RABBSNIP VBUILD  TYPE=TCP,                                             X
               CONTIMER=30,            WAIT FOR MPTN TO COME UP        X
               DGTIMER=30,             INTERVAL BETWEEN RETRIES        X
               DNSUFX=IBM.COM,         DOMAIN NAME SUFFIX              X
               EXTIMER=3,              BETW. SEND SNA EXPEDITED DATA   X
               IATIMER=120,            TIME BEFORE MPTN KEEPALIVE      X
               PORT=397,               WELLKNOWN PORT FOR ANYNET       X
               TCB=50,                 NUMBER MVS SUBTASKS             X
               TCPIPJOB=T11ATCP        TCP/IP JOBNAME
RABGSNIP GROUP ISTATUS=ACTIVE          GROUPNAME
RABLSNIP LINE  ISTATUS=ACTIVE          LINENAME
RABPSNIP PU    ISTATUS=ACTIVE
```

Figure 134. AnyNet/MVS TCP/IP major node definitions on SA11 - Member RABBSNIP in RISC.VTAMLST

```
*********************************************************************
*                                                                   *
*                       VTAM 42  CDRSCS                             *
*                                                                   *
*           SA 11  DEFINITIONS (SNA OVER TCP/IP)                    *
*                                                                   *
* NAME   CDRSC ALSLIST=...  PU DEFINED WITHIN THE TCP/IP            *
*                           MAJOR NODE                              *
*                                                                   *
*     - THE NAME LABEL OF THE CDRSC DEFINITION STATEMENT MUST BE    *
*       THE REMOTE ILU NAME.                                        *
*                                                                   *
*     - WE MUST CODE ALSREQ=YES TO USE THE PREDEFINED LIST.         *
*                                                                   *
*********************************************************************
         VBUILD  TYPE=CDRSC
*********************************************************************
WTR05221 CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     OS/2
WTR05222 CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     OS/2
WTR05115 CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     OS/2
RAIAC    CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     CICS SA18
RAPAC    CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     CICS SA25
RA3AC    CDRSC CDRM=RA3                        CICS SA03
RALYAS4A CDRSC ALSLIST=RABPSNIP,ALSREQ=YES     AS/400
*********************************************************************
         NETWORK NETID=USIBMRA
*********************************************************************
```

Figure 135. CDRSCS major node definitions for remote ILUs on SA11 - Member RABRSNIP in RISC.VTAMLST

```
*******************************************************************
*                                                                 *
*         VTAM SWITCHED MAJOR NODE FOR DLUR (ANYNET)              *
*                                                                 *
*******************************************************************
RABSSNIP VBUILD MAXDLUR=4,                                          X
               TYPE=SWNET
ISNIPJ01 PU    ADDR=01,                                             X
               IDBLK=05D,                                           X
               IDNUM=05221,                                         X
               DISCNT=NO,                                           X
               ISTATUS=ACTIVE,                                      X
               MAXDATA=1033,                                        X
               MAXPATH=4,                                           X
               PACING=0,                                            X
               PUTYPE=2,                                            X
               DLOGMOD=D4C32XX3,                                    X
               MODETAB=ISTINCLM,                                    X
               USSTAB=US327X,                                       X
               VPACING=0
*
*      PATH  PID=1,                                               X
*            DLURNAME=WRT05221,                                   X
*            DLCADDR=(1,C,INTPU),                                 X
*            DLCADDR=(2,X,05D05221)
*
ISNIPJL1 LU    LOCADDR=2
ISNIPJL2 LU    LOCADDR=3
ISNIPJL3 LU    LOCADDR=4
ISNIPJL4 LU    LOCADDR=5
*
ISNIPM01 PU    ADDR=01,                                             X
               IDBLK=05D,                                           X
               IDNUM=05115,                                         X
               DISCNT=NO,                                           X
               ISTATUS=ACTIVE,                                      X
               MAXDATA=1033,                                        X
               MAXPATH=4,                                           X
               PACING=0,                                            X
               PUTYPE=2,                                            X
               DLOGMOD=D4C32XX3,                                    X
               MODETAB=ISTINCLM,                                    X
               USSTAB=US327X,                                       X
               VPACING=0
*
*      PATH  PID=1,                                               X
*            DLURNAME=WRT05115,                                   X
*            DLCADDR=(1,C,INTPU),                                 X
*            DLCADDR=(2,X,05D05115)
*
ISNIPML1 LU    LOCADDR=2
ISNIPML2 LU    LOCADDR=3
ISNIPML3 LU    LOCADDR=4
ISNIPML4 LU    LOCADDR=5
```

Figure 136. Switched major node for DLUR for remote DLUs on SA11 - Member RABSSNIP in RISC.VTAMLST

A.5 VTAM Definitions for SA03

```
********************************************************************
*                                                                  *
*           VTAM 42   MPTF     -      SNA OVER TCP/IP              *
*                                                                  *
*                      SA 03  DEFINITIONS                          *
*                                                                  *
*        NETWORK NETID=USIBMRA - NOT CODED IN ORDER TO USE THE     *
*                                CDRMNAME AS ADJSSCP               *
*                                                                  *
********************************************************************
         VBUILD  TYPE=CDRSC
         NETWORK NETID=USIBMRA
RABAC    CDRSC CDRM=RAB        CICS SA11
RAIAC    CDRSC CDRM=RAB        CICS SA18
RAPAC    CDRSC CDRM=RAB        CICS SA25
RALYAS4A CDRSC CDRM=RAB        AS400
```

Figure 137. CDRSCS Major Node Definitions for Remote ILUs on SA03 - Member RA3RSNIP in RISC.VTAMLST

A.6 VTAM Definitions for SA25

```
**********************************************************************
*                                                                    *
*          VTAM  42  MPTF     -      SNA OVER TCP/IP                 *
*                                                                    *
*                     SA 25  DEFINITIONS                             *
*                                                                    *
*       NETWORK NETID=USIBMRA - NOT CODED IN ORDER TO USE THE        *
*                               CDRMNAME AS ADJSSCP                  *
*                                                                    *
**********************************************************************
         VBUILD  TYPE=CDRSC
RALYAS4A CDRSC CDRM=RAI        AS400
RA3AC    CDRSC CDRM=RAI        CICS SA03
RABAC    CDRSC CDRM=RAI        CICS SA11
RAIAC    CDRSC CDRM=RAI        CICS SA18
RAKAC001 CDRSC CDRM=RAI        CICS SA20
BREL     CDRSC CDRM=RAI        OS2
BRELIP   CDRSC CDRM=RAI        OS2  APPC OVER TCPIP
HUGO     CDRSC CDRM=RAI        OS2
HUGOIP   CDRSC CDRM=RAI        OS2  APPC OVER TCPIP
```

Figure 138. CDRSCS Major Node Definitions for Remote ILUs on SA25 - Member RAPRSNIP in RISC.VTAMLST

A.7 VTAM CDRM Definitions Shared by SA03 and SA25

```
         VBUILD TYPE=CDRM
         NETWORK NETID=USIBMRA
RA3      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=03
RAB      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=11
RAK      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=20
RAP      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=25
RAH      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=17
RAI      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=18
RAS      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=28
RAX      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=33
RAG      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=16
RA1      CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=1
RA39     CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=39
RA36     CDRM    CDRDYN=YES,CDRSC=OPT,SUBAREA=36
```

Figure 139. CDRM Major Node Definitions for SA03 and SA25 - Member RA$WRA in RISC.VTAMLST

Appendix B. System Definitions - OS/2

This appendix lists the configuration files of the OS/2 systems that were used for the scenarios described throughout this document.

B.1 Scenario 6: Configuration Files for System SUSI

The following configuration was used for this system.

Table 6. System Unit and Installed Software Products for System SUSI

Hardware	Software	Functions
• IBM PS/2 80-A31 • 16MB Memory • Token-Ring Adapter 16/4/A	• OS/2 2.11 • IBM TCP/IP Version 2 for OS/2 • IBM Communications Manager/2 Version 1.11 • AnyNet/2 SNA over TCP/IP • Database 2 OS/2 (DB2/2) • Distributed Database Connection Services/2 (DDCS/2) • APING	• MPTN access node • Native TCP/IP host • Domain Name Server • Database server • DDCS/2 gateway • APING client/server

C:CONFIG.SYS File

```
IFS=C:OS2HPFS.IFS  /CACHE:2048 /CRECL:32 /AUTOCHECK:CD
SET SQLNETB=16
PROTSHELL=C:\OS2\PMSHELL.EXE
SET USER_INI=C:\OS2\OS2.INI
SET SYSTEM_INI=C:\OS2\OS2SYS.INI
SET OS2_SHELL=C:\OS2\CMD.EXE
SET AUTOSTART=PROGRAMS,TASKLIST,FOLDERS
set restartobjects=startupfoldersonly
SET RUNWORKPLACE=C:\OS2\PMSHELL.EXE
SET COMSPEC=C:\OS2\CMD.EXE
LIBPATH=C:\ANYNET\DLL;C:\SQLLIB\DCSLIB;C:\SQLLIB\DLL;C:\IBMCOM\DLL;.;
        C:\USERDLLS;C:\OS2\DLL;C:\MUGLIB\DLL;C:\OS2\APPS\DLL;C:\CMLIB\DLL;
        C:\OS2\MDOS;C:\;C:\TCPIP\DLL;
SET PATH=C:\ANYNET\BIN;C:\SQLLIB;C:\CMDS;C:\OS2UTILS;C:\OS2;C:\MUGLIB;
         C:\CMLIB;C:\OS2\SYSTEM;C:\OS2\APPS;C:\OS2\MDOS\WINOS2;C:\OS2\INSTALL;
         C:\OS2\MDOS;C:\;C:\TCPIP\BIN;
SET DPATH=C:\ANYNET\MISC;C:\SQLLIB;C:\TCPIP\BIN;C:\IBMCOM;C:\OS2UTILS;
          C:\OS2;C:\MUGLIB;C:\CMLIB;C:\OS2\SYSTEM;C:\OS2\APPS;C:\OS2\MDOS\WINOS2;
          C:\OS2\INSTALL;C:\OS2\BITMAP;C:\OS2\MDOS;C:\;
SET HELP=C:\SQLLIB;C:\OS2\HELP;C:\HELPLIB;C:\OS2\HELP\TUTORIAL;C:\CMLIB;
         C:\TCPIP\HELP;C:\TCPIP\BIN;C:\ANYNET;
SET BOOKSHELF=C:\SQLLIB\BOOK;C:\OS2\BOOK;C:\INFLIB;C:\CMLIB\BOOK;C:\TCPIP\DOC;
SET EPMPATH=C:\OS2\APPS;
SET GLOSSARY=C:\OS2\HELP\GLOSS;
SET CMPATH=C:\CMLIB
SET IPF_KEYS=SBCS
SET KEYS=ON
PRIORITY_DISK_IO=NO
```

```
FILES=20
BUFFERS=30
IOPL=YES
MAXWAIT=3
MEMMAN=SWAP,PROTECT
SWAPPATH=D:\ 2048 12288
BREAK=OFF
THREADS=255
PROTECTONLY=NO
SHELL=C:\OS2\MDOS\COMMAND.COM C:\OS2\MDOS
FCBS=16,8
RMSIZE=640
DOS=LOW,NOUMB
PRINTMONBUFSIZE=134,134,134
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
BASEDEV=PRINT02.SYS
BASEDEV=IBM2FLPY.ADD
BASEDEV=IBM2SCSI.ADD /LED
BASEDEV=OS2DASD.DMD
BASEDEV=XGA.SYS
DEVICE=C:\IBMCOM\PROTOCOL\LANPDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANVDD.OS2
DEVICE=C:\OS2\TESTCFG.SYS
DEVICE=C:\OS2\DOS.SYS
DEVICE=C:\OS2\PMDD.SYS
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSE.SYS
DEVICE=C:\OS2\COM.SYS
DEVICE=C:\OS2\LOG.SYS
DEVICE=C:\OS2\EPWDD.SYS
DEVICE=C:\OS2\XGARING0.SYS
DEVICE=C:\OS2\MDOS\VEMM.SYS
DEVICE=C:\OS2\MDOS\VXMS.SYS /UMB
DEVICE=C:\OS2\MDOS\VDPMI.SYS
DEVICE=C:\OS2\MDOS\VDPX.SYS
DEVICE=C:\OS2\MDOS\VCDROM.SYS
DEVICE=C:\OS2\MDOS\VWIN.SYS
DEVICE=C:\OS2\MDOS\VMOUSE.SYS
DEVICE=C:\OS2\MDOS\VCOM.SYS
DEVICE=C:\OS2\MDOS\VVGA.SYS
DEVICE=C:\OS2\MDOS\VXGA.SYS
CODEPAGE=437,850
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
DEVINFO=SCR,VGA,C:\OS2\VIOTBL.DCP
SET VIDEO_DEVICES=VIO_XGA
SET VIO_XGA=DEVICE(BVHVGA,BVHXGA)
DEVICE=C:\IBMCOM\LANMSGDD.OS2 /I:C:\IBMCOM
DEVICE=C:\IBMCOM\PROTMAN.OS2 /I:C:\IBMCOM
DEVICE=C:\IBMCOM\PROTOCOL\LANDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANDLLDD.OS2
DEVICE=C:\IBMCOM\MACS\IBMTOK.OS2
DEVICE=C:\CMLIB\ACSLANDD.SYS
DEVICE=C:\CMLIB\CMKFMDE.SYS
RUN=C:\IBMCOM\PROTOCOL\NETBIND.EXE
RUN=C:\IBMCOM\LANMSGEX.EXE
RUN=C:\IBMCOM\PROTOCOL\LANDLL.EXE
RUN=C:\OS2\SYSTEM\LOGDAEM.EXE
RUN=C:\OS2\EPWDDR3.EXE
RUN=C:\OS2\EPWROUT.EXE 1
```

```
RUN=C:\OS2\EPW.EXE
SET ETC=C:\TCPIP\ETC
SET TMP=C:\TCPIP\TMP
SET READIBM=C:\TCPIP\DOC
SET HOSTNAME=anygw1sna
RUN=C:\TCPIP\BIN\CNTRL.EXE
IFS=C:\TCPIP\BIN\NFS200.IFS
DEVICE=C:\IBMCOM\PROTOCOL\INET.SYS
DEVICE=C:\IBMCOM\PROTOCOL\IFNDIS.SYS
SET TZ=est5edt
SET NFS.PERMISSION.BITS=700
SET NFS.PERMISSION.DBITS=700
SET USER=thilo
SET PASSWD=thilo
SET TELNET.PASSWORD.ID=thilo
SET SYSCONTACT=Martin W. Murhammer
SET SYSLOCATION=CC-103
SET INCLUDE=C:\SQLLIB;
SET LIB=C:\SQLLIB;
SET SNASUFFIX=ANYNET.IBM.COM
SET CRITICAL_WS=NO
SET CONNWAIT_SECS=30
SET CONN_RETRY_SECS=300
SET MPTN_WELL_KNOWN_PORT=397
SET TCPWAIT_MINS=15
SET UNACKED_DG_RETRY_SECS=30
SET UNSENT_DG_RETRY_SECS=3
SET INACTIVITY_TIMER_SECS=90
SET ANYNETPATH=C:\ANYNET
```

C:IBMCOMPROTOCOL.INI File

```
[PROT_MAN]

   DRIVERNAME=PROTMAN$

[IBMLXCFG]

   landd_nif=landd.nif
   tcpip_nif=tcpip.nif
   IBMTOK_nif=IBMTOK.nif

[landd_nif]

   DriverName=LANDD$
   Bindings=IBMTOK_nif
   ETHERAND_TYPE="I"
   SYSTEM_KEY=0x0
   OPEN_OPTIONS=0x2000
   TRACE=0x0
   LINKS=8
   MAX_SAPS=3
   MAX_G_SAPS=0
   USERS=3
   TI_TICK_G1=255
   T1_TICK_G1=15
   T2_TICK_G1=3
   TI_TICK_G2=255
   T1_TICK_G2=25
   T2_TICK_G2=10
```

```
   IPACKETS=250
   UIPACKETS=100
   MAXTRANSMITS=6
   MINTRANSMITS=2
   TCBS=64
   GDTS=30
   ELEMENTS=800

[tcpip_nif]

   DriverName=TCPIP$
   Bindings=IBMTOK_nif

[IBMTOK_nif]

   DriverName=IBMTOK$
   NETADDRESS="400000033317"
   ADAPTER="PRIMARY"
   MAXTRANSMITS=12
   RECVBUFS=10
   RECVBUFSIZE=256
   XMITBUFS=2
   XMITBUFSIZE=2024
```

C:STARTUP.CMD File

```
@Echo Off
call c:\tcpip\bin\setup.cmd
start c:\tcpip\bin\named.exe
start  c:\cmlib\cmstart.exe
startdbm
exit
```

C:CMLIBLUTAB.LST File

```
-DEFAULT␣␣␣␣␣␣␣␣␣␣NATIVE␣␣␣␣␣␣␣␣␣
USIBMRA.ISNIPM01␣␣NON-NATIVE␣␣␣␣␣
USIBMRA.RAIDB2␣␣␣␣NON-NATIVE_ONLY
USIBMRA.ISNIPT01␣␣NON-NATIVE␣␣␣␣␣
```

C:CMLIBWTRMODEL.NDF File

```
  DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR33317 )
                   CP_ALIAS(WTR33317)
                   NAU_ADDRESS(INDEPENDENT_LU)
                   NODE_TYPE(EN)
                   NODE_ID(X'05D33317')
                   NW_FP_SUPPORT(NONE)
                   HOST_FP_SUPPORT(YES)
                   HOST_FP_LINK_NAME(HOST$1  )
                   MAX_COMP_LEVEL(NONE)
                   MAX_COMP_TOKENS(0);

  DEFINE_LOGICAL_LINK  LINK_NAME(HOST$1  )
                       ADJACENT_NODE_TYPE(LEARN)
                       DLC_NAME(IBMTRNET)
                       ADAPTER_NUMBER(0)
                       DESTINATION_ADDRESS(X'400030000001')
                       ETHERNET_FORMAT(NO)
                       CP_CP_SESSION_SUPPORT(NO)
                       SOLICIT_SSCP_SESSION(YES)
```

```
                    PU_NAME(WTR33317)
                    NODE_ID(X'05D33317')
                    ACTIVATE_AT_STARTUP(NO)
                    USE_PUNAME_AS_CPNAME(NO)
                    LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
                    LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                    MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                    EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                    COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                    COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                    SECURITY(USE_ADAPTER_DEFINITION)
                    PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_3(USE_ADAPTER_DEFINITION);

DEFINE_LOCAL_LU  LU_NAME(ISNIPS01)
                 LU_ALIAS(ISNIPS01)
                 NAU_ADDRESS(INDEPENDENT_LU);

DEFINE_MODE  MODE_NAME(IBMRDB  )
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(1024)
             RECEIVE_PACING_WINDOW(4)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(8)
             MIN_CONWINNERS_SOURCE(0)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                 DEFAULT_MODE_NAME(BLANK)
                 MAX_MC_LL_SEND_SIZE(32767)
                 DIRECTORY_FOR_INBOUND_ATTACHES(*)
                 DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                 DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                 DEFAULT_TP_CONV_SECURITY_RQD(NO)
                 MAX_HELD_ALERTS(10);

DEFINE_TP  SNA_SERVICE_TP_NAME(X'07',6DB)
           PIP_ALLOWED(NO)
           FILESPEC(c:\sqllib\sqlciaa.exe)
           CONVERSATION_TYPE(EITHER)
           CONV_SECURITY_RQD(YES)
           SYNC_LEVEL(EITHER)
           TP_OPERATION(NONQUEUED_AM_STARTED)
           PROGRAM_TYPE(BACKGROUND)
           RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

DEFINE_TP  SNA_SERVICE_TP_NAME(X'07',6SN)
           PIP_ALLOWED(NO)
           FILESPEC(c:\sqllib\sqlcnsm.exe)
           CONVERSATION_TYPE(EITHER)
           CONV_SECURITY_RQD(YES)
           SYNC_LEVEL(EITHER)
           TP_OPERATION(NONQUEUED_AM_STARTED)
           PROGRAM_TYPE(BACKGROUND)
```

```
            RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

  DEFINE_TP  TP_NAME(APINGD)
             PIP_ALLOWED(NO)
             FILESPEC(c:\os2utils\apingd.exe)
             CONVERSATION_TYPE(ANY_TYPE)
             CONV_SECURITY_RQD(NO)
             SYNC_LEVEL(EITHER)
             TP_OPERATION(NONQUEUED_AM_STARTED)
             PROGRAM_TYPE(VIO_WINDOWABLE)
             RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

  START_ATTACH_MANAGER;
```

C:TCPIPBINSETUP.CMD File

```
route -fh
arp -f
ifconfig lo 127.0.0.1
ifconfig lan0 9.24.104.117 netmask 255.255.255.0 mtu 2000
route add default 9.24.104.1 1
```

C:TCPIPETCRESOLV File

```
domain USIBMRA.ANYNET.IBM.COM
nameserver 9.24.104.117
```

C:TCPIPETCNAMEDBNAMED.BT File

```
;
; NAMED.BT file for name server configuration.
;
; type      domain                      source file or host
;
domain   USIBMRA.ANYNET.IBM.COM
;
;cache    .                        c:\\tcpip\\etc\\namedb\\named.ca
;
primary  USIBMRA.ANYNET.IBM.COM      c:\\tcpip\\etc\\namedb\\named.dom
;
primary  104.24.9.in-addr.arpa       c:\\tcpip\\etc\\namedb\\named.rev
;
;secondary  test.raleigh.ibm.com  9.67.30.143  c:\\tcpip\\etc\\namedb\\named.sac
;
```

C:TCPIPETCNAMEDBNAMED.DOM File

```
;
;*******************************
;*  Start of Authority Records  *
;*******************************
;
@   IN  SOA  anygw1sna.USIBMRA.ANYNET.IBM.COM.   anygw1sna.USIBMRA.ANYNET.IBM.COM (
        94080802 ; Serial number for this data (yymmdd##)
        86400    ; Refresh value for secondary name servers
        300      ; Retry value for secondary name servers
        864000   ; Expire value for secondary name servers
        3600 )   ; Minimum TTL value
;
@   IN  NS   anygw1sna.USIBMRA.ANYNET.IBM.COM.
;
```

```
;*******************************
;*  Domain Address Information  *
;*******************************
anygw1sna                 86400  IN  A      9.24.104.117
                                 IN  HINFO  IBM-PS/2 OS2 2.1
;
ISNIPM01                  86400  IN  A      9.24.104.12
                                 IN  HINFO  IBM-PS/2 OS/2 2.1
;
ISNIPS01                  86400  IN  A      9.24.104.117
                                 IN  HINFO  IBM-PS/2 OS/2 2.1
;
RABAC                     86400  IN  A      9.24.104.75
                                 IN  HINFO  CICS SA 11/25
;
RA3AC                     86400  IN  A      9.24.104.74
                                 IN  HINFO  CICS SA 03/18
;
RAIDB2                    86400  IN  A      9.24.104.74
                                 IN  HINFO  DB2 SA 18
;
MVS03                     86400  IN  A      9.24.104.74
                                 IN  HINFO  MVS SA 03
;
```

C:TCPIPETCNAMEDBNAMED.REV File

```
;
;*******************************
;*  Start of Authority Records  *
;*******************************
;
104.24.9.in-addr.arpa.   IN  SOA  anygw1sna.USIBMRA.ANYNET.IBM.COM.
                                  anygw1sna.USIBMRA.ANYNET.IBM.COM. (
        94080801 ; Serial number for this data (yymmdd##)
        86400    ; Refresh value for secondary name servers
        300      ; Retry value for secondary name servers
        864000   ; Expire value for secondary name servers
        3600 )   ; Minimum TTL value

104.24.9.in-addr.arpa.  IN  NS   anygw1sna.USIBMRA.ANYNET.IBM.COM.
;
12          IN  PTR    ISNIPM01.USIBMRA.ANYNET.IBM.COM.
74          IN  PTR    RA3AC.USIBMRA.ANYNET.IBM.COM.
74          IN  PTR    RAIDB2.USIBMRA.ANYNET.IBM.COM.
75          IN  PTR    RABAC.USIBMRA.ANYNET.IBM.COM.
117         IN  PTR    anygw1sna.USIBMRA.ANYNET.IBM.COM.
117         IN  PTR    ISNIPS01.USIBMRA.ANYNET.IBM.COM.
118         IN  PTR    ISNIPT01.USIBMRA.ANYNET.IBM.COM.
;
```

DB2/2 Node Directory

```
 Node Directory

 Number of entries in the directory = 2

Node 1 entry:

 Workstation alias           = MVS18
```

```
 Comment                     = MVS Subarea 18
 Comment code page           = 437
 Protocol                    = APPN
 Network ID                  = USIBMRA
 Local logical unit alias    = ISNIPS01
 Partner logical unit        = RAIDB2
 Transmission service mode   = IBMRDB

Node 2 entry:

 Workstation alias           = WTR32271
 Comment                     = Martin
 Comment code page           = 437
 Protocol                    = APPN
 Network ID                  = USIBMRA
 Local logical unit alias    = ISNIPS01
 Partner logical unit        = ISNIPM01
 Transmission service mode   =
```

DB2/2 Database Connection Services (DCS) Directory

```
 Database Connection Services (DCS) Directory

 Number of entries in the directory = 1

DCS 1 entry:

 Local database name           = DB2MVS
 Target database name          = DB218
 Application requestor name    =
 DCS parameters                =
 Comment                       = DB2 on MVS18
 Comment code page             = 437
 DCS directory release level   = 0x0100
```

DB2/2 Database Directory

```
 System Database Directory

 Number of entries in the directory = 2

Database 1 entry:

 Database alias         = DB2MVS
 Database name          = DB2MVS
 Database drive         =
 Database directory     =
 Workstation alias      = MVS18
 Database type          = DB2/2  VER. 4.00
 Comment                = MVS Database local Alias
 Comment code page      = 437
 Directory entry type   = Remote

Database 2 entry:

 Database alias         = SAMPLE
 Database name          = SAMPLE
 Database drive         = C:
 Database directory     =
 Workstation alias      =
```

```
Database type          = DB2/2  VER. 4.00
Comment                = Sample OS/2 Database
Comment code page      = 437
Directory entry type   = Indirect
```

B.2 Scenario 6: Configuration Files for System MARTIN

The following configuration was used for this system.

Table 7. System Unit and Installed Software Products for System MARTIN

Hardware	Software	Functions
• IBM PS/2 80-321 • 16MB Memory • Token-Ring Adapter 16/4/A	• OS/2 2.11 • IBM TCP/IP Version 2 for OS/2 • IBM Communications Manager/2 Version 1.11 • AnyNet/2 SNA over TCP/IP • Database 2 OS/2 (DB2/2) • APING	• MPTN access node • Native TCP/IP host • Database client • APING client/server

C:CONFIG.SYS File

```
IFS=C:OS2HPFS.IFS  /CACHE:2048 /CRECL:32 /AUTOCHECK:CD
SET SQLNETB=16
PROTSHELL=C:\OS2\PMSHELL.EXE
SET USER_INI=C:\OS2\OS2.INI
SET SYSTEM_INI=C:\OS2\OS2SYS.INI
SET OS2_SHELL=C:\OS2\CMD.EXE
SET AUTOSTART=PROGRAMS,TASKLIST,FOLDERS
set restartobjects=startupfoldersonly
SET RUNWORKPLACE=C:\OS2\PMSHELL.EXE
SET COMSPEC=C:\OS2\CMD.EXE
LIBPATH=C:\MUGLIB\DLL;C:\ANYNET\DLL;C:\SQLLIB\DLL;C:\IBMCOM\DLL;.;
        C:\USERDLLS;C:\OS2\DLL;C:\OS2\APPS\DLL;C:\CMLIB\DLL;C:\OS2\MDOS;
        C:\;C:\TCPIP\DLL;
SET PATH=C:\ANYNET\BIN;C:\SQLLIB;C:\CMDS;C:\OS2UTILS;C:\OS2;C:\MUGLIB;
         C:\CMLIB;C:\OS2\SYSTEM;C:\OS2\APPS;C:\OS2\MDOS\WINOS2;
         C:\OS2\INSTALL;C:\OS2\MDOS;C:\;C:\TCPIP\BIN;
SET DPATH=C:\ANYNET\MISC;C:\SQLLIB;C:\TCPIP\BIN;C:\IBMCOM;C:\OS2UTILS;
          C:\OS2;C:\MUGLIB;C:\CMLIB;C:\OS2\SYSTEM;C:\OS2\APPS;
          C:\OS2\MDOS\WINOS2;C:\OS2\INSTALL;C:\OS2\BITMAP;C:\OS2\MDOS;C:\;
SET HELP=C:\SQLLIB;C:\OS2\HELP;C:\HELPLIB;C:\OS2\HELP\TUTORIAL;C:\CMLIB;
         C:\TCPIP\HELP;C:\TCPIP\BIN;C:\ANYNET;
SET GLOSSARY=C:\OS2\HELP\GLOSS;
SET IPF_KEYS=SBCS
PRIORITY_DISK_IO=YES
FILES=20
DEVICE=C:\IBMCOM\PROTOCOL\LANPDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANVDD.OS2
DEVICE=C:\IBMCOM\LANMSGDD.OS2 /I:C:\IBMCOM
DEVICE=C:\IBMCOM\PROTMAN.OS2 /I:C:\IBMCOM
DEVICE=C:\OS2\TESTCFG.SYS
DEVICE=C:\OS2\DOS.SYS
DEVICE=C:\OS2\PMDD.SYS
BUFFERS=30
```

```
IOPL=YES
rem DISKCACHE=64,LW
MAXWAIT=3
MEMMAN=SWAP,PROTECT
SWAPPATH=D:\ 2048 12288
BREAK=OFF
THREADS=255
PRINTMONBUFSIZE=134,134,134
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
SET KEYS=ON
REM SET DELDIR=C:\DELETE,512;D:\DELETE,512;
BASEDEV=PRINT02.SYS
BASEDEV=IBM2FLPY.ADD
BASEDEV=IBM2SCSI.ADD /LED
BASEDEV=OS2DASD.DMD
SET BOOKSHELF=C:\SQLLIB\BOOK;C:\OS2\BOOK;C:\INFLIB;C:\CMLIB\BOOK;C:\TCPIP\DOC;
SET EPMPATH=C:\OS2\APPS;
PROTECTONLY=NO
SHELL=C:\OS2\MDOS\COMMAND.COM C:\OS2\MDOS
FCBS=16,8
RMSIZE=640
DEVICE=C:\OS2\MDOS\VEMM.SYS
DOS=LOW,NOUMB
DEVICE=C:\OS2\MDOS\VXMS.SYS /UMB
DEVICE=C:\OS2\MDOS\VDPMI.SYS
DEVICE=C:\OS2\MDOS\VDPX.SYS
DEVICE=C:\OS2\MDOS\VCDROM.SYS
DEVICE=C:\OS2\MDOS\VWIN.SYS
DEVICE=C:\OS2\MDOS\VMOUSE.SYS
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSE.SYS
DEVICE=C:\OS2\COM.SYS
DEVICE=C:\OS2\MDOS\VCOM.SYS
CODEPAGE=437,850
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
BASEDEV=XGA.SYS
DEVICE=C:\OS2\XGARING0.SYS
DEVINFO=SCR,VGA,C:\OS2\VIOTBL.DCP
SET VIDEO_DEVICES=VIO_XGA
SET VIO_XGA=DEVICE(BVHVGA,BVHXGA)
DEVICE=C:\OS2\MDOS\VVGA.SYS
DEVICE=C:\OS2\MDOS\VXGA.SYS
RUN=C:\IBMCOM\PROTOCOL\NETBIND.EXE
RUN=C:\IBMCOM\LANMSGEX.EXE
DEVICE=C:\IBMCOM\PROTOCOL\LANDD.OS2
DEVICE=C:\IBMCOM\PROTOCOL\LANDLLDD.OS2
DEVICE=C:\IBMCOM\MACS\IBMTOK.OS2
RUN=C:\IBMCOM\PROTOCOL\LANDLL.EXE
DEVICE=C:\CMLIB\ACSLANDD.SYS
DEVICE=C:\CMLIB\CMKFMDE.SYS
SET CMPATH=C:\CMLIB
DEVICE=C:\OS2\EPWDD.SYS
RUN=C:\OS2\EPWDDR3.EXE
RUN=C:\OS2\EPW.EXE
RUN=C:\OS2\SYSTEM\LOGDAEM.EXE
RUN=C:\OS2\EPWROUT.EXE 1
DEVICE=C:\OS2\LOG.SYS
SET ETC=C:\TCPIP\ETC
SET TMP=C:\TCPIP\TMP
```

```
SET READIBM=C:\TCPIP\DOC
SET HOSTNAME=martin
RUN=C:\TCPIP\BIN\CNTRL.EXE
IFS=C:\TCPIP\BIN\NFS200.IFS
DEVICE=C:\IBMCOM\PROTOCOL\INET.SYS
DEVICE=C:\IBMCOM\PROTOCOL\IFNDIS.SYS
SET TZ=est5edt
SET NFS.PERMISSION.BITS=700
SET NFS.PERMISSION.DBITS=700
SET TELNET.PASSWORD.ID=thilo
SET SYSCONTACT=Martin W. Murhammer
SET SYSLOCATION=CC-103
SET INCLUDE=C:\SQLLIB;
SET LIB=C:\SQLLIB;
SET QRWDR=C
SET QRWINST=C:\SQLLIB
SET SNASUFFIX=ANYNET.IBM.COM
SET CRITICAL_WS=NO
SET CONNWAIT_SECS=30
SET CONN_RETRY_SECS=300
SET MPTN_WELL_KNOWN_PORT=397
SET TCPWAIT_MINS=15
SET UNACKED_DG_RETRY_SECS=30
SET UNSENT_DG_RETRY_SECS=3
SET INACTIVITY_TIMER_SECS=90
SET ANYNETPATH=C:\ANYNET
```

C:IBMCOMPROTOCOL.INI File

```
[PROT_MAN]

   DRIVERNAME=PROTMAN$

[IBMLXCFG]

   landd_nif=landd.nif
   tcpip_nif=tcpip.nif
   IBMTOK_nif=IBMTOK.nif

[landd_nif]

   DriverName=LANDD$
   Bindings=IBMTOK_nif
   ETHERAND_TYPE="D"
   SYSTEM_KEY=0x0
   OPEN_OPTIONS=0x2000
   TRACE=0x0
   LINKS=8
   MAX_SAPS=3
   MAX_G_SAPS=0
   USERS=3
   TI_TICK_G1=255
   T1_TICK_G1=15
   T2_TICK_G1=3
   TI_TICK_G2=255
   T1_TICK_G2=25
   T2_TICK_G2=10
   IPACKETS=250
   UIPACKETS=100
   MAXTRANSMITS=6
```

```
   MINTRANSMITS=2
   TCBS=64
   GDTS=30
   ELEMENTS=800

[tcpip_nif]

   DriverName=TCPIP$
   Bindings=IBMTOK_nif

[IBMTOK_nif]

   DriverName=IBMTOK$
   ADAPTER="PRIMARY"
   NETADDRESS="400000032271"
   MAXTRANSMITS=12
   RECVBUFS=10
   RECVBUFSIZE=256
   XMITBUFS=2
   XMITBUFSIZE=2024
```

C:STARTUP.CMD File

```
@Echo Off
call c:\tcpip\bin\setup.cmd
start  c:\cmlib\cmstart.exe sa25
startdbm
exit
```

C:CMLIBLUTAB.LST File

```
-DEFAULT␣␣␣␣␣␣␣␣␣␣NATIVE␣␣␣␣␣␣␣␣␣
USIBMRA.ISNIPS01␣␣NON-NATIVE_ONLY
```

C:CMLIBWTRMODEL.NDF File

```
  DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR32271 )
                   CP_ALIAS(WTR32271)
                   NAU_ADDRESS(INDEPENDENT_LU)
                   NODE_TYPE(EN)
                   NODE_ID(X'05D32271')
                   NW_FP_SUPPORT(NONE)
                   HOST_FP_SUPPORT(YES)
                   HOST_FP_LINK_NAME(HOST$1   )
                   MAX_COMP_LEVEL(NONE)
                   MAX_COMP_TOKENS(0);

  DEFINE_LOGICAL_LINK  LINK_NAME(HOST$1   )
                       ADJACENT_NODE_TYPE(LEARN)
                       DLC_NAME(IBMTRNET)
                       ADAPTER_NUMBER(0)
                       DESTINATION_ADDRESS(X'400030000001')
                       ETHERNET_FORMAT(NO)
                       CP_CP_SESSION_SUPPORT(NO)
                       SOLICIT_SSCP_SESSION(YES)
                       PU_NAME(WTR32271)
                       NODE_ID(X'05D32271')
                       ACTIVATE_AT_STARTUP(NO)
                       USE_PUNAME_AS_CPNAME(NO)
                       LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
                       LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
```

```
                    MAX_ACTIVATION_ATTEMPTS(USE_ADAPTER_DEFINITION)
                    EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                    COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                    COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                    SECURITY(USE_ADAPTER_DEFINITION)
                    PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                    USER_DEFINED_3(USE_ADAPTER_DEFINITION);

  DEFINE_LOCAL_LU  LU_NAME(ISNIPM01)
                   LU_ALIAS(ISNIPM01)
                   NAU_ADDRESS(INDEPENDENT_LU);

  DEFINE_TP  TP_NAME(APINGD)
             PIP_ALLOWED(NO)
             FILESPEC(c:\os2utils\apingd.exe)
             CONVERSATION_TYPE(ANY_TYPE)
             CONV_SECURITY_RQD(NO)
             SYNC_LEVEL(EITHER)
             TP_OPERATION(NONQUEUED_AM_STARTED)
             PROGRAM_TYPE(VIO_WINDOWABLE)
             RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

  DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                   DEFAULT_MODE_NAME(BLANK)
                   MAX_MC_LL_SEND_SIZE(32767)
                   DIRECTORY_FOR_INBOUND_ATTACHES(*)
                   DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                   DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                   DEFAULT_TP_CONV_SECURITY_RQD(NO)
                   MAX_HELD_ALERTS(10);

  START_ATTACH_MANAGER;
```

C:TCPIPBINSETUP.CMD File

```
route -fh
arp -f
ifconfig lo 127.0.0.1
ifconfig lan0 9.24.104.117 netmask 255.255.255.0 mtu 2000
route add default 9.24.104.1 1
```

C:TCPIPETCRESOLV File

```
domain USIBMRA.ANYNET.IBM.COM
nameserver 9.24.104.117
```

DB2/2 Node Directory

```
 Node Directory

 Number of entries in the directory = 2

Node 1 entry:

 Workstation alias      = TEST1
 Comment                = netbios
 Comment code page      = 437
 Protocol               = NETBIOS
 Adapter number         = 0
```

```
 Server NNAME          = WTR33317

Node 2 entry:

 Workstation alias           = TEST2
 Comment                     = appc
 Comment code page           = 437
 Protocol                    = APPN
 Network ID                  = USIBMRA
 Local logical unit alias    = ISNIPM01
 Partner logical unit        = ISNIPS01
 Transmission service mode   =
```

DB2/2 Database Directory

```
 System Database Directory

 Number of entries in the directory = 3

Database 1 entry:

 Database alias          = TEST1
 Database name           = SAMPLE
 Database drive          =
 Database directory      =
 Workstation alias       = TEST1
 Database type           = DB2/2  VER. 4.00
 Comment                 =
 Comment code page       = 0
 Directory entry type    = Remote

Database 2 entry:

 Database alias          = TEST2
 Database name           = SAMPLE
 Database drive          =
 Database directory      =
 Workstation alias       = TEST2
 Database type           = DB2/2  VER. 4.00
 Comment                 =
 Comment code page       = 0
 Directory entry type    = Remote

Database 3 entry:

 Database alias          = TEST3
 Database name           = DB2MVS
 Database drive          =
 Database directory      =
 Workstation alias       = TEST2
 Database type           = DB2/2  VER. 4.00
 Comment                 =
 Comment code page       = 0
 Directory entry type    = Remote
```

B.3 Scenario 7: Configuration Files for System WTR05221

The following configuration was used for the dependent LU Requestor.

Table 8. System Unit and Installed Software Products for System WTR05221

Hardware	Software	Functions
• IBM PS/2 80-A31 • 16MB Memory • Token-Ring Adapter 16/4/A	• OS/2 2.11 • IBM TCP/IP Version 2 for OS/2 • IBM Communications Manager/2 Version 1.11 • AnyNet/2 SNA over TCP/IP • APING	• MPTN access node • Native TCP/IP host • APING client/server

C:CMLIBWTRDLURL.NDF File

```
DEFINE_LOCAL_CP  FQ_CP_NAME(USIBMRA.WTR05221 )
                 CP_ALIAS(WTR05221)
                 NAU_ADDRESS(INDEPENDENT_LU)
                 NODE_TYPE(EN)
                 NODE_ID(X'05D05221')
                 NW_FP_SUPPORT(NONE)
                 HOST_FP_SUPPORT(NO)
                 MAX_COMP_LEVEL(NONE)
                 MAX_COMP_TOKENS(0);

DEFINE_DEPENDENT_LU_SERVER  LINK_NAME(HOST$1  )
                            FQ_ADJACENT_CP_NAME(USIBMRA.RAB      )
                            PU_NAME(WTR05221)
                            ACTIVATE_AT_STARTUP(NO);


DEFINE_PARTNER_LU_LOCATION FQ_PARTNER_LU_NAME(USIBMSC.WTR05115)
                   WILDCARD_ENTRY(NO)
                   FQ_OWNING_CP_NAME(USIBMRA.RAB)
                   LOCAL_NODE_NN_SERVER(NO);

DEFINE_MODE  MODE_NAME(#INTER  )
             COS_NAME(#INTER  )
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(1024)
             RECEIVE_PACING_WINDOW(7)
             MAX_NEGOTIABLE_SESSION_LIMIT(8)
             PLU_MODE_SESSION_LIMIT(8)
             MIN_CONWINNERS_SOURCE(4)
             COMPRESSION_NEED(PROHIBITED)
             PLU_SLU_COMPRESSION(NONE)
             SLU_PLU_COMPRESSION(NONE);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                 DEFAULT_MODE_NAME(BLANK)
                 MAX_MC_LL_SEND_SIZE(32767)
                 DIRECTORY_FOR_INBOUND_ATTACHES(*)
                 DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                 DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                 DEFAULT_TP_CONV_SECURITY_RQD(NO)
```

```
                    MAX_HELD_ALERTS(10);

  DEFINE_TP  TP_NAME(APINGD)
             PIP_ALLOWED(NO)
             FILESPEC(c:Öos2utilsÖapingd.exe)
             CONVERSATION_TYPE(ANY_TYPE)
             CONV_SECURITY_RQD(NO)
             SYNC_LEVEL(EITHER)
             TP_OPERATION(NONQUEUED_AM_STARTED)
             PROGRAM_TYPE(VIO_WINDOWABLE)
             RECEIVE_ALLOCATE_TIMEOUT(INFINITE);

  START_ATTACH_MANAGER;
```

C:CMLIBLUTAB.LST File

```
-DEFAULTNATIVE
USIBMRA.RABNON-NATIVE_ONLY
USIBMRA.RAINON-NATIVE_ONLY
USIBMSC.WTR05115NON-NATIVE_ONLY
USIBMSC.WTR05222NON-NATIVE_ONLY
USIBMRA.RALYAS4BNON-NATIVE_ONLY
USIBMRA.RALYAS4ANON-NATIVE_ONLY
USIBMRA.RAIAZNON-NATIVE_ONLY
USIBMRA.RABAZNON-NATIVE_ONLY
USIBMRA.RALABTCPNON-NATIVE_ONLY
```

Index

T

U

V

W

X

ITSO Technical Bulletin Evaluation RED000

AnyNet: SNA over TCP/IP
Installation and Interoperability

Publication No. GG24-4395-00

Your feedback is very important to help us maintain the quality of ITSO Bulletins. **Please fill out this questionnaire and return it using one of the following methods:**

- Mail it to the address on the back (postage paid in U.S. only)
- Give it to an IBM marketing representative for mailing
- Fax it to: Your International Access Code + 1 914 432 8246
- Send a note to REDBOOK@VNET.IBM.COM

Please rate on a scale of 1 to 5 the subjects below.
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction ____

Organization of the book	____	Grammar/punctuation/spelling	____
Accuracy of the information	____	Ease of reading and understanding	____
Relevance of the information	____	Ease of finding information	____
Completeness of the information	____	Level of technical detail	____
Value of illustrations	____	Print quality	____

Please answer the following questions:

a) If you are an employee of IBM or its subsidiaries:

Do you provide billable services for 20% or more of your time? Yes____ No____

Are you in a Services Organization? Yes____ No____

b) Are you working in the USA? Yes____ No____

c) Was the Bulletin published in time for your needs? Yes____ No____

d) Did this Bulletin meet your needs? Yes____ No____

If no, please explain:

____

____

What other topics would you like to see in this Bulletin?

____

____

What other Technical Bulletins would you like to see published?

____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

Name	Address
Company or Organization	
Phone No.	

ITSO Technical Bulletin Evaluation **RED000**
GG24-4395-00

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 International Technical Support Organization
Department 545, Building 657
P.O. BOX 12195
RESEARCH TRIANGLE PARK NC
USA 27709-2195

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

GG24-4395-00

Cut or Fold
Along Line

Printed in U.S.A.

GG24-4395-00