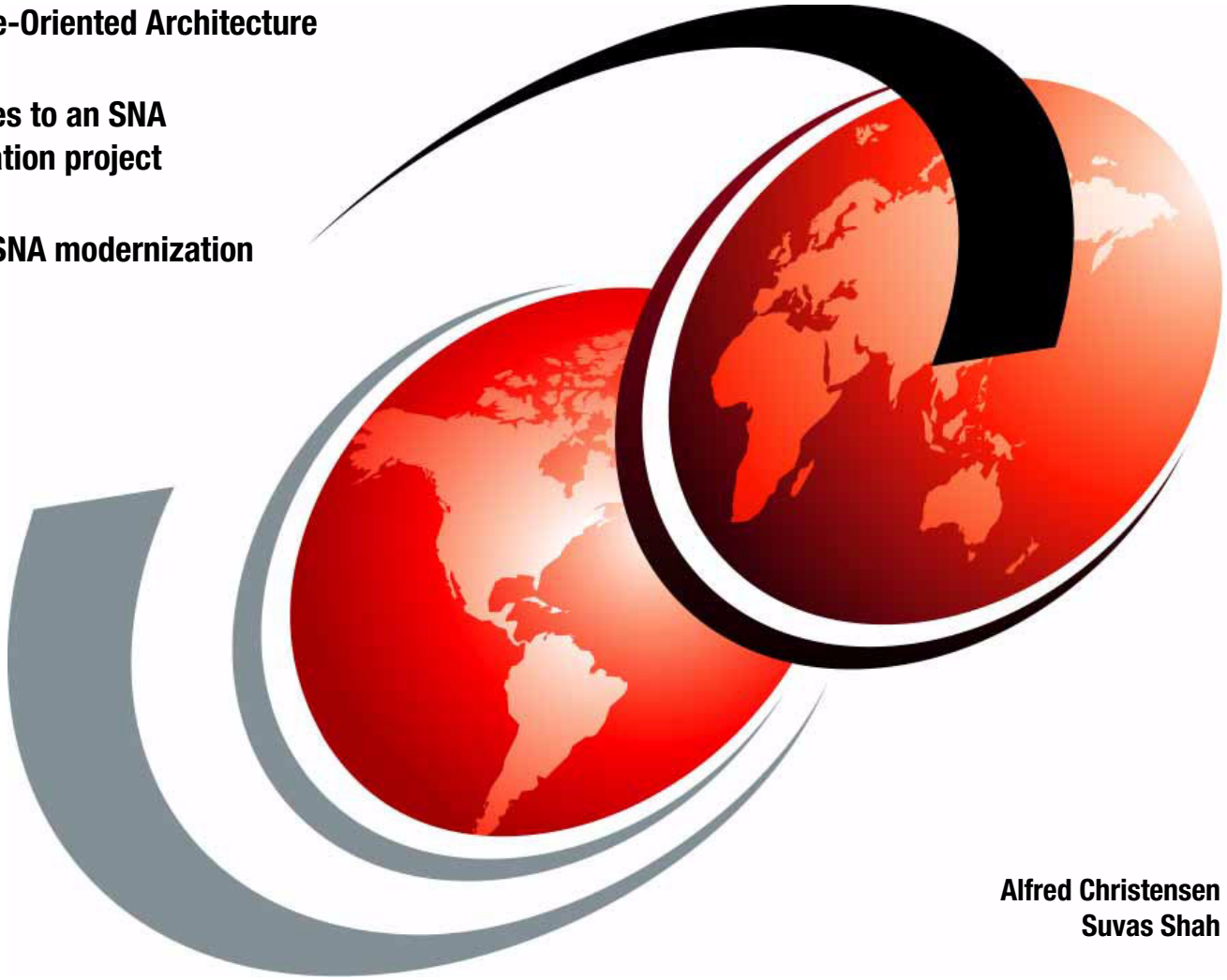
IBM

A Structured Approach to Modernizing the SNA Environment

- Getting your SNA environment ready for Service-Oriented Architecture
- Approaches to an SNA modernization project
- Common SNA modernization scenarios



Alfred Christensen
Suvas Shah

Redbooks

ibm.com/redbooks

International Technical Support Organization

A Structured Approach to Modernizing the SNA Environment

October 2006

SG24-7334-00

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First Edition (October 2006)

This edition applies to IBM z/OS V1R8.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

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

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

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

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include 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.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

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

Advanced Peer-to-Peer Networking®
AnyNet®
AIX®
Candle®
CICS®
DB2®
DRDA®
ESCON®
FICON®
GDDM®
HiperSockets™
IBM®
IMS™
iSeries™
i5/OS®
MVS™
NetView®
Nways®
OpenPower™
OS/2®
OS/400®
pSeries®
Redbooks™
Redbooks (logo)  ™
S/390®
System i5™
System p™
System x™
System z™
System z9™
Tivoli®
VTAM®
WebSphere®
z/OS®
z/VM®
z/VSE™
zSeries®
z9™

The following terms are trademarks of other companies:

EJB, Java, J2EE, Solaris, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Win32, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Preface

The focus of this IBM® Redbook is networking infrastructure aspects of modernizing an SNA network environment. Additionally, modernizing SNA concerns itself with how to modernize access to existing SNA core business applications. In order to provide a reference model for modernizing SNA, this book introduces a selected set of SNA application access modernization technologies, which go beyond the normal networking infrastructure. Some of the technologies discussed in this book aim at improving the user experience when accessing traditional SNA 3270 applications and at exposing existing mainframe SNA applications as Web services. Such technologies will primarily be introduced from a network topology and connectivity perspective.

While a description of the full set of capabilities of such technologies is beyond the scope of this document, readers can refer to other technology-specific and product-specific documentation from IBM and other vendors for that information.

The intended audience for this book are IBM System z™ technical managers, system architects, and network administrators who are responsible for setting the overall strategic directions for an enterprise networking infrastructure. This infrastructure includes Systems Network Architecture (SNA) and Internet Protocol (IP) networking technologies, branch or remote location access networks, data center connectivity, and business partner connectivity.

The book is organized as follows:

- Definition of SNA modernization according to this redbook, how it relates to overall enterprise transformation strategies, and the scope of SNA modernization activities.
- Discussion of the overall objectives of an SNA modernization project and how to prioritize the options to select the proper technologies.
- How to approach an SNA modernization project where the primary objective is to modernize the overall network infrastructure, while preserving the existing SNA node topology more or less unchanged. This section includes a discussion of how to modernize both an SNA subarea and an SNA APPN environment.
- Approaches to an SNA modernization project where the primary objective is to modernize the way you access SNA applications. This section includes discussions of how to address SNA modernization at an application level. These application modernizations can preserve existing user interfaces, transform user interfaces to enhance the user experience, and also allow you to integrate SNA applications into a Service Oriented Architecture (SOA).
- A set of selected general SNA modernization scenarios.

The book assumes a certain familiarity with SNA networking and SNA/IP integration technologies in general. If you are unfamiliar with SNA, or would like to refresh your knowledge, you can find an introduction in Appendix A, “SNA levels” on page 95.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Alfred Christensen is a Programming Consultant in Enterprise Networking and Transformation Solutions in Raleigh, North Carolina, USA. He has 32 years of experience in IBM mainframe software technologies. He has worked at IBM for 21 years, starting as a large account system engineer in Denmark, and relocating to Raleigh, North Carolina in 1995. His areas of expertise include general z/OS® software technologies, with special emphasis on networking on z/OS: SNA and TCP/IP. He has written extensively on mainframe networking technologies and is a frequent speaker at SHARE and other IBM System z technical conferences.

Suvas Shah is a Senior Engineer in Raleigh, North Carolina, and has worked at IBM for 29 years. His area of expertise is communication technology. He was the chief designer for Communications Manager for OS/2®, and worked on software design for Communications Server for Windows® with the Advanced Technology and Architecture group. He has also been involved in Business Development for Communications Server for Windows, Communications Server for Linux® (Intel®, pSeries®, and System z), as well as Communications Server for AIX® and Communication Controller for Linux on System z. Suvas holds a Bachelor's degree in Information Engineering from the University of Illinois at Chicago Circle.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Modernizing Systems Network Architecture - an introduction

One of the key characteristics of the IBM mainframe technology has always been protection of the investments made in application software. Many mainframe installations have considerable investments in current applications that are Systems Network Architecture (SNA) based. Modernizing SNA is about preserving the investments in those SNA applications, while at the same time reducing the cost of maintaining and operating an aging SNA network infrastructure.

Note: Modernizing SNA is not about rewriting or discarding SNA applications. It is about preserving SNA applications in an Internet Protocol (IP)-based network infrastructure. It is also about enabling reuse of SNA applications in new end-user environments in an application-transparent manner.

The ultimate goal for SNA modernization projects is to preserve and enhance SNA applications on the mainframe and in the branch environment, for as long as these SNA applications remain valuable business assets—while at the same time transporting wide-area SNA application-level traffic over an IP wide area network, consolidating SNA network level traffic to the data center or ultimately the System z platform itself. See Figure 1-1 on page 2.

In this IBM Redbook, the term *branch* is used to refer to any remote location where network connectivity to a central data center exists.

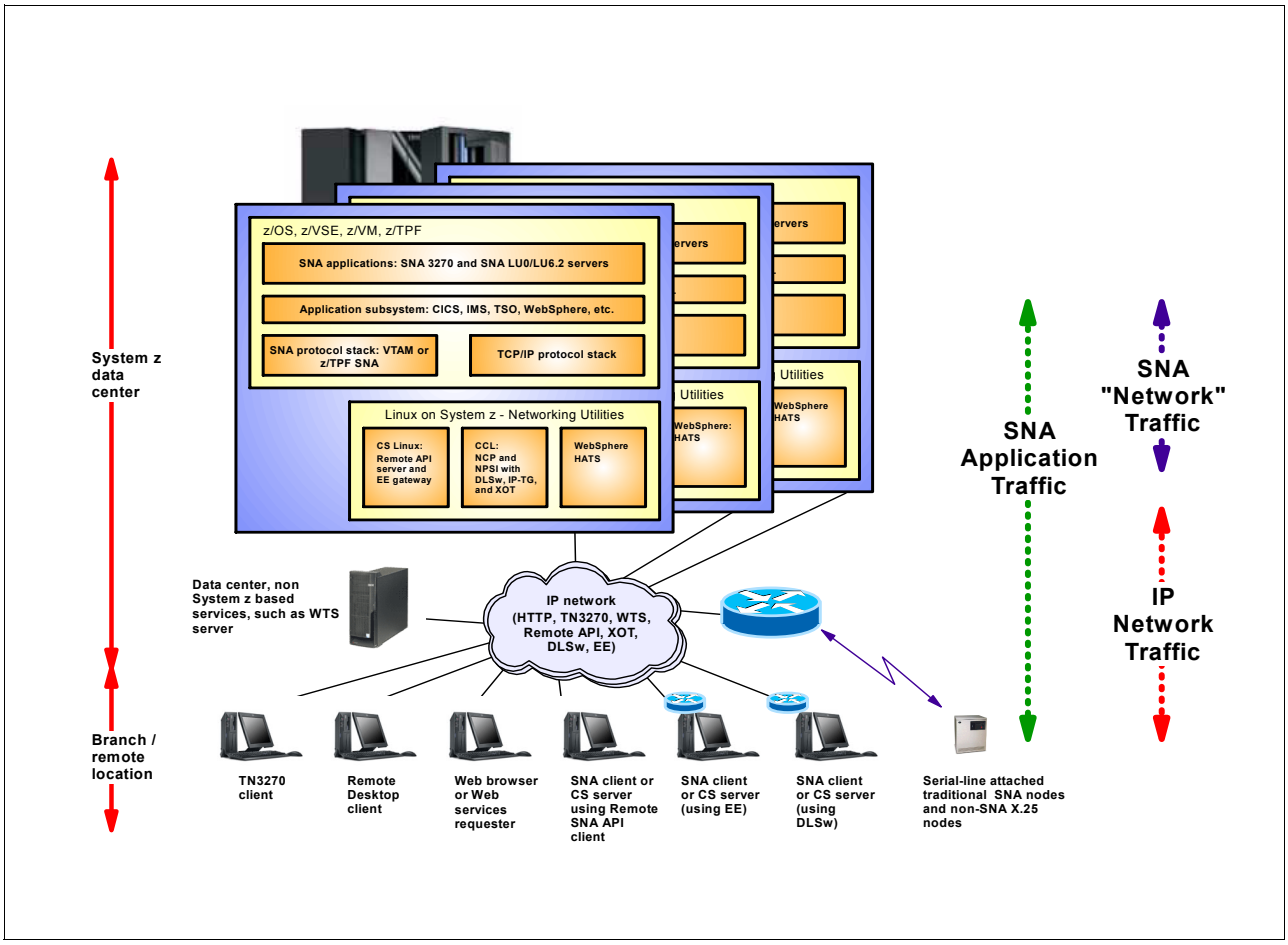


Figure 1-1 SNA networking to the System z data center

It is possible to achieve such an objective using today's technologies, but getting there from your current position may seem like an overwhelming task given the multitude of alternative approaches and overlapping technologies.

To enable you to better approach this task in a structured manner, this redbook clarifies the objectives, describes and organizes the technologies, provides some guidelines for how you can get started and which direction to follow, and finally presents some general SNA modernization scenarios from which you can choose elements that will fit into your environment.

Following is the definition of modernizing SNA that this redbook uses:

- Modernizing the SNA network infrastructure

  This means updating the SNA network infrastructure to remove dependency on outdated SNA-specific hardware technologies and using, instead, a state-of-the-art network technology that is based on a shared high-speed, secure, reliable, and highly available IP-based network topology for transporting both SNA and IP application traffic.

- Modernizing SNA application access

  This means enabling continued use of both SNA client and server applications in their current form over a modernized network infrastructure while, at the same time, allowing you access to integrate SNA server applications into new client environments, such as a Web browser. Also, allowing you access to integrate SNA server applications into modern

application environments including those based on a services-oriented architecture (SOA).

This redbook focuses on how to modernize the SNA networking infrastructure. Additionally, it also focuses on how to modernize and transform the access to SNA applications, primarily within the networking infrastructure itself, without requiring any changes to existing SNA applications or SNA-based subsystems, such as CICS®, IMS™, DB2®, or TSO, that host SNA applications.

There are various subsystem-specific technologies available that you can use to modernize access to SNA applications in those subsystems. Some of the commonly used technologies are the CICS Transaction Gateway (CTG), CICS Service Flow feature (SFF), CICS 3270 bridge, IMS MFS Web enablement, and IMS Connect.

While both the subsystem-specific and the network-based modernization technologies can provide the same end result, the way to reach that end result varies. With the network-based technologies, both mainframe subsystems and applications are unaware of any changes and the changes can be implemented without the need for application development or subsystem administration skills and resources. The modernization project can in most cases be contained within the networking support groups, freeing up development resources to focus on modernizing applications rather than reacting to infrastructure changes.

On the other hand, using subsystem-specific technologies may allow you to replace SNA completely and may offer a higher level of integration between the new client environments and the existing mainframe applications.

There are some applications that are more or less directly exposed to SNA programming interfaces. This document focuses on how to preserve such applications while at the same time modernizing both the network infrastructure and the options for accessing those applications.

There are also applications that use SNA in a more indirect way. Think of an application that uses MQ or SQL programming interfaces where various instances of queue managers and database managers use a network for communication between these instances. Such a communication can be based on either SNA LU6.2 sessions or TCP connections over an IP network. It is a configuration choice that is made by the MQ or DB2 administrator when setting up these environments:

- Distributed Relational Data Access (DRDA®), as used between DB2-Connect and DB2 on System z, can easily be changed from using SNA LU6.2 sessions to use TCP connections over an IP network, removing the need for SNA functions in support of DRDA traffic.
- Message Queuing between MQ clients and MQ servers and between MQ servers themselves can also easily be changed from using SNA LU6.2 sessions to TCP connections over an IP network instead.

Changes to both the MQ and DB2 configurations are relatively simple and will have no impact on the end users or the applications that use the services of DB2 and MQ. To those end users and applications, the transport mechanism used by DRDA and MQ is transparent.

Where such transport changes can be implemented with no impact to end users, applications, or application subsystems, such as CICS or IMS, they should be considered as an initial and simple way of reducing the amount of SNA traffic, with the objective of simplifying the remaining overall SNA modernization project.

1.1 SNA modernization - how it relates to enterprise transformation

The term *Enterprise Transformation* was defined a few years ago with the objective of enabling IBM customers to evolve application systems from traditional application environments to a service-oriented architecture, and to integrate these services into workflows and business processes with other application systems.

At its core, enterprise transformation is about taking existing applications and programmer skills, and cost-effectively and efficiently integrating them into the new world of On Demand business.

The original enterprise transformation model included three stages. Some of the SNA modernization technologies interact with all three stages, but the stages in this original enterprise transformation model do not address all the objectives of SNA modernization.

To provide a more comprehensive model that includes the objectives of transforming the network infrastructure, in addition to transforming the enterprise business processes, this document introduces an enterprise transformation stage 0 that extends the model to four stages.

1.1.1 Network infrastructure modernization

This is for situations where organizations aim at optimizing the investment in IP-based networking technology by supporting both SNA and IP application traffic over a single IP-based network infrastructure, reducing the need for traditional SNA network-specific hardware technologies, simplifying the overall SNA node topology in the enterprise network, and preparing the network infrastructure for a services-oriented application architecture.

1.1.2 Improving user experience through presentation integration

Where application interfaces are difficult to use and user workflows are outdated, organizations are taking action to improve their customers' or end users' online experience. Typically, this means improving the user interface—upgrading from old traditional IBM 3270 screen interfaces, improving site navigation, providing a simple Web-based point-and-click interface, and adding a pre-filled fields function, so that online users need to enter boilerplate information, such as a billing address, only once during a session.

1.1.3 Adapting enhanced relationships through programmatic integration

Where traditional applications cannot easily be integrated into modern workflows, organizations are adapting their applications to participate in today's On Demand workflows, without incurring the risks of replacing an entire platform.

This may involve *wrapping* business functions' existing back-end applications with Web services interfaces, in order to seamlessly incorporate them into a new workflow with other applications in a service-oriented architecture (SOA). For example, a function such as a customer purchase history can be transparently added to a Web site without regard to the application by which it is created.

By integrating legacy applications for richer, differentiated interactions, this *Adapt* stage of enterprise transformation typically helps businesses to develop new and stronger partnerships with customers, partners, and suppliers. On the IT side, it helps reduce

development costs, and also provides a standard-based framework for integration of applications with other Web technologies.

1.1.4 Innovating new capabilities through code modernization

For situations where it is difficult to adapt mission-critical processes to changing business or market conditions, organizations are using their traditional applications to create completely new solutions. By understanding what is in their existing mission-critical applications, they can restructure and *componentize* those applications and integrate parts of them into new, differentiated solutions and a service-oriented architecture.

The *Innovate* stage of enterprise transformation enables an organization to build new, strategic solutions by modifying their current applications and avoiding the expense of developing a new application from scratch. A business can adapt its processes to new marketplace imperatives quickly and at low cost. IT professionals can improve the development process and productivity across the complete application cycle. See Figure 1-2.

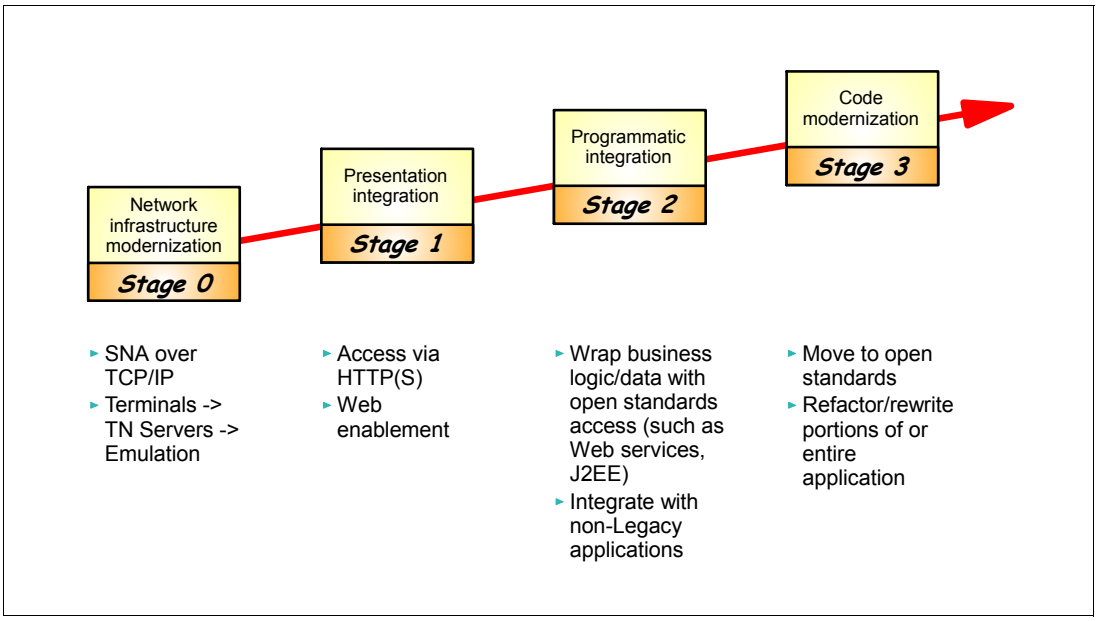


Figure 1-2 Enterprise transformation stages

The staging does not necessarily indicate a sequence. Stage 0 can be addressed in parallel with the activities in the other stages. You can perform some application transformation activities using Stage 1 technologies, while you can perform others using Stage 2 or Stage 3 technologies.

When discussing specific technologies throughout this document, we indicate which stage the technology primarily belongs to. Note that some of the technologies may be useful in more than one stage.

1.2 SNA applications - how they relate to SNA networks

At an overview level, an SNA network consists of the following:

- SNA host nodes that have SNA support through microcode (an IBM 3174 cluster control unit), or through the operating system, or middleware. On z/OS this is VTAM® and on

Linux this is IBM Communications Server for Linux, generally referred to as the SNA protocol stack. In SNA terminology, SNA host nodes are called *physical units* (PUs) as opposed to *logical units* (LUs) that are the actual communication (session) end-points in the SNA network. A PU is of a certain type that defines the overall characteristics of a PU.

SNA host nodes support LUs that are either SNA applications, or fixed-function SNA end-user devices, or both:

- SNA applications on SNA host nodes

  SNA applications use the SNA application programming interfaces (APIs) and protocols that are supported by the SNA protocol stack on the SNA host node where they are deployed. Some of these SNA APIs are standardized and are the same on all SNA host nodes, such as Common Programming Interface for Communications (CPI-C).

- SNA end-user devices connected to SNA host nodes

  This is your typical IBM 3270 display terminal or IBM 328x printer device.

- An SNA network infrastructure that routes SNA traffic over SNA transmission facilities between SNA host nodes and SNA intermediate nodes, that reside within the SNA network infrastructure.

SNA applications use and depend on SNA programming interfaces on the SNA host nodes where they reside, and in some cases also on SNA-related session-level protocols, such as an SNA 3270 data stream.

SNA applications generally fall into two main categories:

- IBM 3270 application

  This is a mainframe SNA application that communicates with an end user that uses a device such as an IBM 3270 terminal. The “application-level” protocol is based on the IBM 3270 data stream formats and protocol standards. This is your typical “green screen” application in the TSO environment, in CICS, in IMS, and so on. Even if the application may not be directly involved in interfacing to the SNA protocol stack through SNA programming interfaces, it often is involved in interpreting, or building parts of, the whole SNA 3270 data stream. See Figure 1-3.

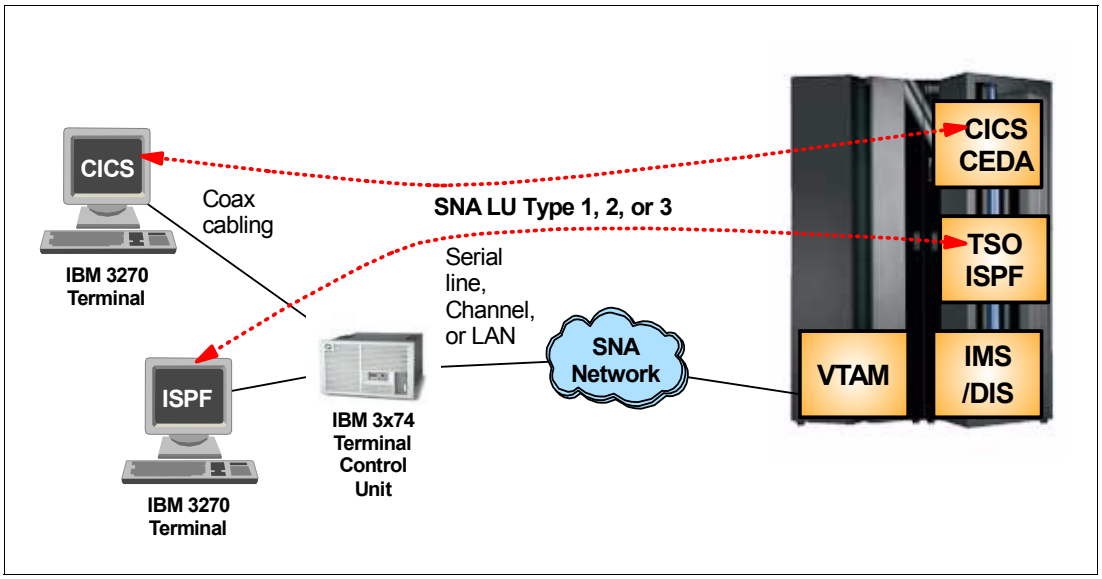


Figure 1-3 SNA 3270 application

The end-user device is typically not a real IBM 3270 terminal, but instead an IBM 3270 emulator software package on a programmable workstation.

The mainframe SNA application does not know the difference between a real IBM 3270 terminal that is coax cable-attached to an IBM 3174 cluster control unit and an emulated IBM 3270 terminal on a programmable workstation running one of many IBM 3270 emulator software packages, such as the IBM Personal Communications product (PCOMM).

In SNA terminology, the IBM 3270 devices (real or emulated) are dependent LU type 1 (SNA Character String (SCS) print), type 2 (3270 display), or type 3 (3270 print) nodes.

- SNA client/server application

  This is a pair of SNA applications that form a traditional client/server relationship. The “application-level” protocol is defined as part of the general client/server application design. Most typically the server SNA application resides on the mainframe on System z or on System i5™, while the client SNA application is deployed remotely on the end-user workstation or on a branch server. See Figure 1-4.

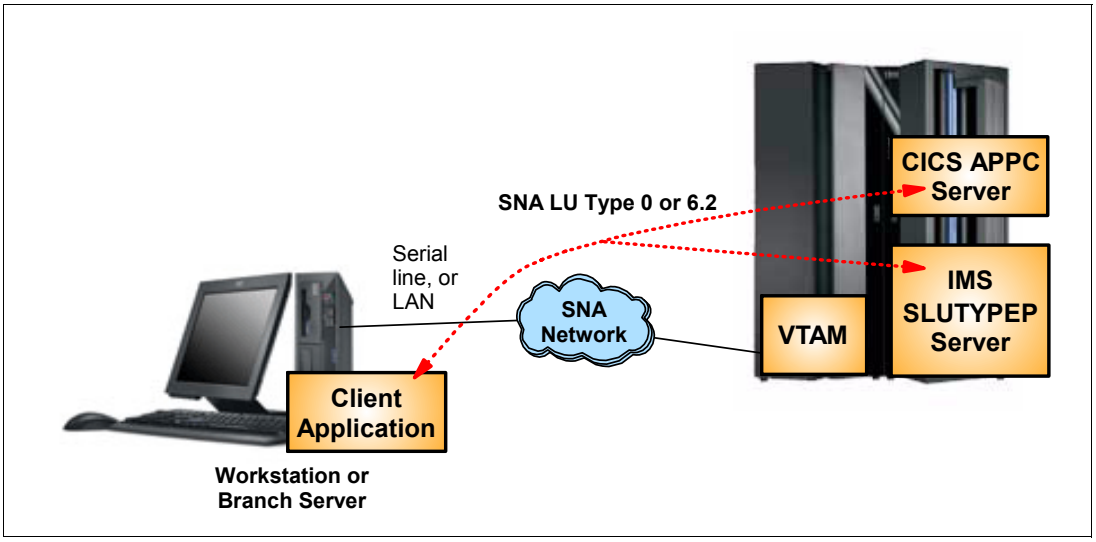


Figure 1-4 SNA client/server application

In SNA terminology, the clients are either dependent LU type 0 or LU type 6.2, which can be either dependent or independent.

Dependent or independent, in this context, refers to the LU's ability to establish an SNA session with or without the assistance of VTAM (an SNA System Services Control Point (SSCP)). A dependent LU must talk with a VTAM (an SSCP) to set up a session, while an independent LU can establish sessions with peer nodes without asking for the assistance of an SSCP. In an SNA subarea network most LUs are dependent, while LUs of LU type 6.2 in an APPN network typically are independent LUs. There is some support for independent LUs in an SNA subarea network through what is known as Low Entry Networking (LEN). We will not discuss LEN in any more detail in this document.

Where you use an IBM 3270 emulator, the first category (the IBM 3270 application category) becomes a special case of the second category (the SNA client/server application). The client becomes a standardized client, the IBM 3270 emulator software that, from an SNA perspective, is a dependent LU type 1, type 2, or type 3.

Note: As long as you do not change the SNA programming interfaces on the SNA host nodes, you can replace the total SNA network infrastructure between the SNA host nodes, where the SNA applications reside, without impacting the application functions.

Modernizing SNA does not mean that SNA applications have to be rewritten. SNA applications can continue to be in use both on the mainframe and in the distributed environment, as long as those applications have business value to a corporation.

In fact, nothing prevents anyone from writing and deploying new SNA applications for SNA programming interfaces that are best suited for the application in question. Some of the SNA programming interfaces have functions that are still superior to what is available in the IP world. The SNA Advanced Program-to-Program Communication (APPC or CPI-C) APIs and associated attach manager functions to schedule server applications, provide authentication and security options that continue to be of value in a corporate network environment.

1.3 Network infrastructure - how it relates to SNA levels

The SNA network infrastructure was originally constructed of dedicated SNA nodes, such as IBM Communication Controllers, and transmission facilities, such as serial SDLC lines. This type of a network infrastructure supported SNA data formats and protocols only, and you could not use it for other network protocols.

When TCP/IP began emerging and entered the corporate network environment in the late 1980s, a separate and parallel IP network infrastructure was most often deployed to support IP application workloads.

Around the early 1990s, corporate networks began moving towards an increasingly complex topology. Here some elements of the corporate network, such as LAN segments, were used for both IP and SNA. Other elements, such as wide area network links, were duplicated and one set of components and links were used for IP and another set of components and links were used for SNA.

Some network hardware technologies, such as a LAN or a frame relay network (FR), support multiple networking protocols on the same medium. It is technically possible to transport both SNA and IP data on the same LAN medium, as long as there are nodes attached to that LAN that support either SNA or IP or both. Most operating systems, such as z/OS, Windows, AIX, Linux, i5/OS®, and so on, that host network applications, support both SNA and IP data formats and protocols. Therefore, they may host both SNA and IP applications that communicate over a shared LAN interface to a single LAN medium.

Other network hardware technologies are SNA-specific, such as serial lines using an SDLC line protocol, or SNA-specific routers (such as IBM 3745/46). You can use these technologies only for SNA data formats and protocols.

Building, maintaining, operating, and managing a duplicate corporate network infrastructure is obviously not as cost efficient as being able to use a single network infrastructure for both SNA and IP application workloads.

Some attempts at developing technologies to encapsulate IP data in SNA data formats and to transport the IP data over an SNA network did occur, for example, the IBM AnyNet® technology known as Sockets over SNA. Such IP over SNA technologies were not successful. Some limited use did occur, but nothing significant. The main reason was that the use of IP increased so rapidly that it very soon became much more relevant to explore the opposite

objective: how to transport SNA application data over an IP network infrastructure, end-to-end.

Note: AnyNet support was removed from VTAM on z/OS in z/OS V1R8 and is no longer shipped with z/OS.

There are currently many technologies available for transporting SNA application data over an IP network. These technologies are typically in use in corporate networks.

Some of these technologies will only work with certain levels of the SNA architecture. Therefore, before continuing with a discussion of the available modernization options, following is a brief revisit of the SNA architecture levels that are in widespread use today:

- SNA subarea

  This is traditional hierarchical SNA, as it was originally defined and deployed around 1974 and enhanced over the following years with features such as Cross-Domain VTAM communication.

  VTAM with the SSCP at the top of the hierarchy connected via mainframe channels to one or more Communication Controllers with NCPs that, in turn, connected to terminal cluster controllers or branch computers over serial lines.

  Between NCPs and VTAM, and between VTAM and other VTAMs, alternate SNA routes can be defined. In SNA terminology, a route is a “path”. The original SNA architecture used a concept of SNA networking subareas between which you could select alternative routes, if they had been defined within the overall SNA subarea topology. If a route became unavailable, all the SNA sessions that were currently assigned to that route would fail but new sessions could be established over an alternate route.

  Within the routable parts of an SNA network, all network elements are uniquely addressed. On a serial line, addressing is local for that serial line only (control unit addresses, and device addresses per control unit). One of the functions of an NCP is, therefore, to map the line-specific addressing scheme to the unique addressing scheme of the SNA network. Performing such address mapping is referred to as performing SNA boundary functions.

  See “First SNA level: SNA subarea networking” on page 96 for more information.

- Advanced Peer-to-Peer Networking® (APPN) with the original APPN routing protocol known as Intermediate Session Routing (ISR).

  The APPN architecture level augments the hierarchical structure of an SNA subarea network. In an APPN network SNA nodes are peers and independent LUs (LUs of type 6.2) can establish sessions between peer nodes without asking VTAM for permission.

  In addition, in an APPN network, most of the network topology in terms of routes, nodes, and location of LUs is learned dynamically and does not have to be predefined.

  Session traffic may be routed through any number of intermediate APPN nodes on the path between the session partners. If one of the routes that a session currently is assigned to fails, the session will, as it did in an SNA subarea network, also fail. A new session can be established over an alternate route.

  See “Second SNA level: APPN” on page 98 for more information.

- Advanced Peer-to-Peer Networking with High Performance Routing (HPR).

  HPR does not really change the SNA topology as it was defined in APPN. It replaces the way SNA data is routed through the APPN network.

  With HPR routing, SNA sessions that currently use an HPR route may survive a route failure. If an underlying network link fails, but an alternate HPR-capable route is available,

the HPR routing infrastructure will nondisruptively switch the SNA sessions from the failed route to the alternate route.

HPR routing may be used over LANs, mainframe channels, and so on. You can also use it over an IP network where the HPR infrastructure sees the IP network as an HPR link. This capability is referred to as HPR over IP, or more commonly as Enterprise Extender (EE).

See “Third SNA level - HPR” on page 101 for more information.

Note: Many SNA nodes are able to support connectivity and routing of SNA data using all three architecture levels at the same time, that is, using subarea routing to some nodes, the original APPN ISR routing to others, while using APPN HPR routing to the rest.

From the beginning, SNA was designed using a traditional layered concept. In this concept, each layer in the architecture delivered a well-defined set of functions using one or more layer-specific protocols for exchanging data and controlling information with the corresponding layer in a partner SNA node. Between the layers were well-defined interfaces that allowed an upper layer to pass information down to a lower layer through a precisely defined set of programming interfaces. See Figure 1-5.

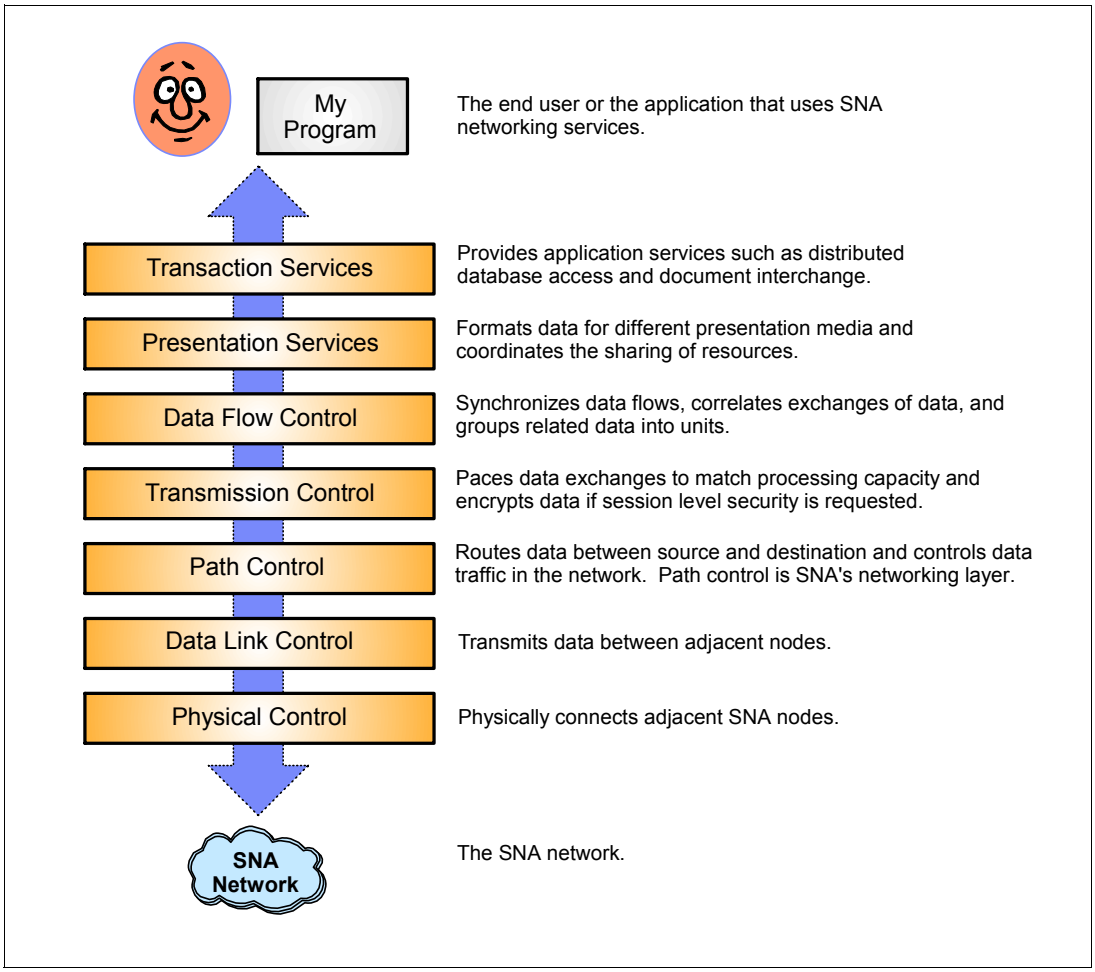


Figure 1-5 SNA protocol layers

The main purpose of such a layered approach was, and still is, to support change.

Note: As long as the interface between two layers does not change, an underlying layer can be totally redesigned and reimplemented using new technologies and protocols for communication with its partner layer in another SNA node, without any impact to upper layers in the architecture, which ultimately includes the SNA applications.

Because SNA was designed for change originally, IBM has been able to continue to enhance the SNA architecture and to invent new SNA networking technologies, such as transporting SNA over an IP network, which is what Enterprise Extender does. The vast majority of SNA CICS or IMS applications that were originally used in 1974 from an SDLC line-attached cluster of IBM 3278 terminals can in 2006 be used from a Web browser over an IP network. In most cases, no changes to the mainframe SNA subsystem or SNA application are required.

It is important to note that from an SNA application perspective, as long as the programming interfaces that you originally used to implement an SNA application do not change, that SNA application will continue to work regardless of how much underlying technology you change in the path between the application and the node it communicates with.

2

SNA modernization objectives

SNA modernization technologies and activities address the following objectives:

- Allow installations to continue to use existing SNA business applications, both remote and in the data center, unchanged for as long as these SNA applications have business value.
  - This includes preserving existing user interfaces, such as various SNA 3270 technologies, generally referred to as green screen access or traditional SNA 3270 access.
  - It also includes preserving use of distributed SNA client applications and access to SNA server applications, using any of the commonly used SNA client/server LU types and programming interfaces.
- Assist installations in modernizing and simplifying their application portfolio by moving access to SNA applications to a browser-based workstation technology and by adapting the SNA applications to an overall application infrastructure that is based on a service-oriented application architecture.
  - Enhance the user experience by transforming the SNA 3270 data stream to an HTML data stream accessing the SNA 3270 server application from a Web browser.
  - Enable use of the SNA core business application as a Web service by implementing a Web service interface between a service oriented end user environment or other business processes and existing SNA server applications.
- Help remove dependency on an outdated SNA networking infrastructure, concerning both hardware and certain software-based elements:
  - IBM 3705, 3720, 3725, and 3745/46 Communication Controller hardware
  - IBM 2210, 2216, and 2217 Nways® Multi-protocol Routers
  - IBM AnyNet software technology in general
  - OEM ESCON® channel-attached SNA gateways, such as Cisco CIP and Cisco CPA
  - Token-Ring LAN technology in general
- Assist in reducing the need for SNA skills in the overall enterprise.

  Help reduce the need for SNA wide area networking skills by consolidating SNA network infrastructure elements and eventually SNA protocol stacks into the data center itself.
- Help reduce the overall cost of the enterprise networking environment by simplifying the enterprise networking infrastructure so both SNA-based and IP-based application

services share a common high-capacity, scalable, reliable, and secure IP-based transport network that provides both enterprise-wide and inter-enterprise connectivity.

  - One IP network end-to-end
  - One network administration skill set
  - One set of network management tools and procedures

To some installations all of the previous objectives may be equally important, while only some of them are important to others. We recommend some form of prioritization of the objectives. The outcome of such a prioritization may help you decide where to start. Is it most important to get the network infrastructure modernized right now to help reduce network operating cost and reduce risks associated with outdated hardware technologies? Or is there an immediate need for modernizing access to existing SNA applications according to the objectives of the other stages of the enterprise transformation model? Enhance the user experience by transforming the SNA 3270 interface to a Web user interface? Or perhaps to adapt the SNA 3270 server application into new service-oriented business processes?

To address these objectives, there are a number of technologies available, which fall into two main groups:

- Technologies that primarily aim at modernizing the SNA network infrastructure

  The technologies in this group all belong in Stage 0 of the Enterprise Transformation model. They all address network infrastructure modernization objectives.

- Technologies that primarily aim at modernizing the SNA application access

  Some of the technologies in this group also address objectives in Stage 0 of the Enterprise Transformation model, serving the objective of transporting SNA application traffic over an IP network infrastructure. Other technologies in this group belong in Stage 1, presentation integration, and Stage 2, programmatic integration, of the Enterprise Transformation model.

These two groups of technologies will briefly be introduced in the following sections. For a more comprehensive description of all the technologies, refer to relevant product-specific documentation from IBM and other vendors.

2.1 Technologies to modernize the SNA infrastructure

Use technologies in this group to remove dependency on old SNA-specific hardware and merge SNA and IP traffic over a common IP-based network, while at the same time preserving the existing full-function SNA end-user interfaces or functions and SNA node infrastructure in the branch and the data center.

For example, if you have SNA nodes in the branch, such as IBM Communications Server for Windows or IBM PCOMM with a full SNA protocol stack on Windows end-user workstations or branch servers, then this group of technologies offers you a way to retain that SNA topology from a branch perspective, while at the same time addressing your overall SNA infrastructure modernization objectives.

All the technologies in this group belong in what was previously referred to as Stage 0 of enterprise transformation.

The main technologies in this group are:

- Emulated IBM 3745/46: IBM Communication Controller for Linux (CCL)

  CCL is a next-generation IBM Communication Controller that allows an NCP to be deployed in Linux on System z, preserving the traditional SNA subarea network functions without dependence on IBM 3745/46 hardware or earlier levels of IBM Communication Controller hardware. See 3.3.2, “Overview of IBM Communication Controller for Linux (CCL)” on page 31 for further information about CCL.

- Data Link Switching (DLSw)

  This technology encapsulates SNA Logical Link Control (LLC) traffic between two SNA nodes over an intermediate IP network. This is one of the oldest and probably most widely used technologies for transporting SNA subarea traffic over an IP network. See 3.3.3, “Data Link Switching (DLSw)” on page 37 for more details on DLSw.

- X.25 Over TCP/IP (XOT)

  XOT encapsulates X.25 packets (SNA or non-SNA) over an intermediate IP network. A scenario where XOT is used is when X.25 circuits are to be connected to the NCP Packet Switching Interface (NPSI) running in a CCL environment. See 3.3.5, “X.25 packets over TCP/IP (XOT)” on page 40 for more information.

- IP Transmission Group (IP TG)

  IP TG encapsulates SNA INN or SNI traffic over an IP network between two NCPs running in CCL. See 3.3.4, “IP Transmission Group (IP TG)” on page 39 for details on IP TG usage.

- Advanced Peer-to-Peer Networking (APPN) with HPR over IP (also known as Enterprise Extender)

  This technology uses an IP network as a High Performance Routing link. From the SNA APPN network point of view, the IP network appears like any other HPR link in a normal APPN HPR topology. From an IP perspective, HPR over IP looks like any other UDP-based application. This is the most widely used way of transporting SNA data over an IP network in an SNA APPN network environment. See 3.4.3, “APPN with HPR routing over IP (EE)” on page 46.

2.2 Technologies to modernize SNA application access

Use this group of technologies to preserve and enhance existing SNA end-user interfaces and SNA client functions while replacing the SNA transport layer functions with IP-based transport protocols and removing SNA protocol stack functions altogether on nodes outside the data center.

In some cases, the technologies in this group preserve the end-user interface, but replace even the application layer technology with an IP-based technology. The technologies in this group also provide ways for reusing existing SNA server applications from new client environments, such as a Web browser or a Web service requester.

Assuming the scenario this document discusses previously—if you have SNA nodes in the branch, such as IBM Communications Server for Windows or IBM PCOMM with a full SNA protocol stack on Windows end-user workstations or branch servers—then these technologies offer you a way of preserving the SNA end-user interfaces and SNA client functions on these nodes, such as an IBM 3270 emulator or an SNA client application. However, at the same time, these technologies remove the SNA protocol stack functions completely on the end-user workstations or branch servers. In other words, they allow you to collapse SNA protocol stacks into the data center.

- Traditional 3270 interface, TN3270

  This is an application-layer replacement for SNA 3270 emulators. A TN3270 emulator preserves the traditional 3270 interface, the “green screen” user interface, but replaces the communication between the end user workstation and the mainframe with an IP-based application-level connectivity. TN3270 clients come in many varieties: Some are permanently installed and configured on the user workstation, for example IBM PCOMM, while others can be administered centrally and downloaded on demand when required, for example IBM Host On Demand (HOD). TN3270 is a widely used technology and is easy to implement. The base TN3270 technology is typically part of stage 0 in the modified enterprise transformation model, but is a prerequisite for some of the technologies in the other stages as well. See “TN3270 introduction” on page 59 for more information about what TN3270 is and how you can use it.

- Remote SNA API client/server (split stack)

  If the SNA application on the workstation or branch server is not an SNA 3270 emulator, but instead is a homegrown SNA client application program, then TN3270 is not the solution. A remote SNA application programming interface (API) client provides a replacement SNA API layer on the workstation or branch server where the SNA application runs. This replacement SNA API layer presents a traditional SNA API to the SNA client application and ships each SNA API call over an IP network to a remote SNA API server on a node, where a full-function SNA protocol stack is implemented. This technology will most typically be part of stage 0 in the enterprise transformation model. See 4.2.1, “Remote SNA API client/server (split stack)” on page 67 for more details on this technology.

- Remote presentation (split GUI - X-Windows or Microsoft® Remote Desktop and Windows Terminal Services - WTS)

  Provides a technology for preserving the end-user interaction on a workstation, while at the same time consolidating the actual end-user application onto a server node. While this technology is not specific to SNA modernization, it does provide a way to consolidate SNA end-user applications, and possibly SNA protocol stacks, on a server node. The connectivity between the user workstation and the server is over an IP network. If the server is located in the branch, this technology will typically be combined with either TN3270 or remote SNA API to avoid SNA protocol stacks in the branch. This technology

will also typically be part of stage 0 in the enterprise transformation model. See 4.2.2, “Remote desktop (split GUI)” on page 70 for more information.

- SNA 3270 server application access with a Web look

  This type of technology enhances the user experience by transforming the SNA 3270 dialog to HTML, and supporting a Web browser as the client. It belongs in Stage 1 of the enterprise transformation model.

  IBM WebSphere® Host Access Transformation Services (HATS) is an example of a product that implements this technology. HATS transforms the SNA 3270 data stream to HTML “on the fly” using default transformation rules or customized transformation rules that the HATS developer creates. The HATS developer uses the HATS development tools to create a customized transformation through a series of wizards. HATS provides a single, generic, subsystem-independent, HTML transformation solution for both SNA 3270 applications and 5250 applications. HATS is subsystem-independent and can integrate data from multiple SNA 3270 and 5250 applications running in different operating systems and subsystems into a single consolidated Web interface. See “SNA 3270/HTML transformation using HATS” on page 73 for more information about how you can use HATS to transform an SNA 3270 application.

  There are also subsystem-specific technologies available that you can use to provide HTTP(S)-based access to existing SNA 3270 server applications running in CICS or IMS on z/OS. For CICS SNA 3270 transactions, the main components are the CICS Transaction Gateway (CTG), the CICS Web Support (CWS), and the CICS 3270 bridge. For IMS SNA 3270 transactions, the main components are the IMS Message Formatting Services (MFS) Web support in combination with IMS Connect and the IMS Connector for Java™. See “CICS-specific SNA 3270 presentation integration” on page 75 and “IMS-specific SNA 3270 presentation integration” on page 76 for more details about the CICS and IMS-specific support.

- Integrate existing SNA 3270 server applications into reusable business workflow elements.

  These technologies provide development tools and runtime support to create Web services that wrap existing SNA 3270 server applications, and enable their reuse as business workflow elements in a service-oriented application environment. The technologies typically belong in Stage 2, programmatic integration, of the enterprise transformation model.

  There are again both subsystem-specific and subsystem-independent solutions in this area.

  IBM WebSphere Host Access Transformation Services (HATS) can, in addition to providing an SNA 3270 to HTML data stream transformation for SNA 3270 applications, also provide a Web service interface to existing SNA 3270 server applications. In this context, HATS provides a solution that is subsystem-independent and will provide Web service interfaces to SNA 3270 applications in both IMS, CICS, TSO, and so on. See “SNA 3270 applications as Web services - using HATS” on page 79 for more information about how the HATS technology works in this context.

  For SNA 3270 applications running in a CICS environment, CICS Transaction Server 3.1 for z/OS offers the integrated CICS Services Flow Feature (SFF) in combination with the Services Flow Modeler (SFM) that provide tools and runtime support to integrate existing CICS SNA 3270 transactions into a Web services environment. See “CICS SNA applications as Web services - using CICS SFF” on page 81 for more information about using the CICS Service Flow Feature.

For SNA 3270 applications running in IMS, the IMS Connector for Java, IMS MFS Web support, and IMS Connect provide the infrastructure for integrating existing IMS SNA 3270 transactions into a Web services environment. See “IMS SNA applications as Web services - using IMS integration solutions” on page 82 for more information about how IMS Web service support works.

- Integrate existing SNA LU0 or LU6.2 server applications into reusable business workflow elements.

  This type of technology typically belongs in Stage 2, programmatic integration, and in some cases Stage 3, code modernization, of the enterprise transformation model.

  If your mainframe SNA server application is not an IBM 3270 application, but you still require to integrate the services of that mainframe SNA LU0 or LU6.2 server application into a Web services-based business workflow, then you need to develop a piece of transformation logic that can be published as a Web service and interface to your existing SNA LU0 or LU6.2 server application. Based on the technology you choose, this may be done transparently to the SNA LU0/LU6.2 server application and SNA environment in which it runs, by using a technology that performs the transformation in the network and uses SNA communication between the transformation tier and those existing SNA server applications.

  The SNA resource adapters in the IBM Branch Transformation Toolkit (BTT) product provide such a technology. BTT provides SNA LU0 and LU6.2 resource adapters based on the J2EE™ Connector Architecture (J2C). The resource adapters can be used by Java applications, allowing an enterprise to create a Web services wrapper that uses the SNA resource adapters for SNA-based connectivity to existing SNA LU0 and LU6.2 servers and subsystems. See “SNA LU0 or LU6.2 server applications as Web services - using BTT” on page 80 for more information about BTT.

  There are also subsystem-specific technologies in both IMS and CICS that provide for developing Web services wrappers around IMS and CICS transactions. See “CICS SNA applications as Web services - using CICS SFF” on page 81 and “IMS SNA applications as Web services - using IMS integration solutions” on page 82 for more details on the IMS- and CICS-specific technologies in this area.

2.3 Considerations for where to start

You can implement each of these two groups of technologies independently of each other, but they are related to each other.

Note: A “by-product” of modernizing SNA application access is a reduction in the amount of SNA wide area network traffic.

If one implements TN3270 emulation as a replacement for traditional SNA 3270 access, then the SNA 3270 traffic between the end-user workstations and the TN3270 server will be based on IP and not SNA. This reduces the amount of SNA-based network-level traffic that remains in the enterprise network and most likely simplifies the SNA network infrastructure modernization activities.

On the other hand, modernizing SNA application access, at first, may take some time due to the potential need for end-user training, implementation of new software on workstations, or new procedures for how to access existing SNA 3270 applications via a browser interface. All these changes may take considerable time to roll out in a large organization.

Note that modernizing SNA application access using some of the technologies this document introduces can add significant value to your organization and your users, based on the actual capabilities of those technologies. It may help to position your organization better for a future that is based on a service-oriented architecture.

Modernizing the SNA network infrastructure can often be done as a pure technical activity that may not involve end users. It may still require configuration changes on end-user workstations, depending on exactly which technology is chosen, but the engineers in the network maintenance department can perform the majority of the tasks.

A possible approach could be to start with SNA application access modernization. Doing so reduces the need for SNA network-level traffic to a point where an SNA network infrastructure outside the data center either is significantly reduced or is entirely eliminated.

An alternative approach could be to do the opposite. Modernize the network infrastructure to transport SNA data over the wide area network using IP, but leaving the SNA protocol stacks and applications on the nodes where they are currently positioned. Thereby, preserving the SNA nodes and leaving their setup more or less unchanged.

Some of the SNA modernization objectives can be addressed by both groups of modernization technologies. There is no golden rule that says one is better than the other. It depends on local requirements, existing skills, and preferences.

We stated that SNA modernization is not about rewriting applications, but about reusing applications. This is correct, but there may be situations where you need to consider whether a given SNA-based application in its current form meets the business requirements of today and tomorrow. Based on such an assessment, you will encounter situations where, instead of integrating the use of such an existing SNA application, you will rewrite that application to make it fit better within the business processes that it serves.

Table 2-1 and Table 2-2 on page 21 can be used as quick references and aids in determining which technology may have your primary interest. The columns represent technologies, while the rows represent SNA modernization objectives.

The tables identify the primary objective of each of the technologies, not necessarily the only objective.

Table 2-1 summarizes the application access modernization technologies. Table 2-2 on page 21 summarizes the infrastructure modernization technologies.

2.3.1 Modernize application access: quick reference

Table 2-1 Application access modernization technologies

Objectives	Modernization technologies								
	SNA 3270 applications			SNA client/server applications					Rewrite applications to TCP/IP APIs or as Web services
	TN3270 emulation (PCOMM and HOD)	User interface transformation (HATS, IMS, CICS)	Web service and SOA integration (HATS, IMS, CICS)	Remote API	Remote desktop	Web service and SOA integration (BTT, IMS, CICS)	DRDA o. IP	MQ o. IP	
Enterprise Transformation Stage	0	1	2	0	0	1 and 2	0	0	3
Retain traditional 3270 screen on WS	✓[a]				(✓)[b]				
Retain SNA client on WS				✓	(✓)				

Objectives		Modernization technologies								
		SNA 3270 applications			SNA client/server applications					Rewrite applications to TCP/IP APIs or as Web services
		TN3270 emulation (PCOMM and HOD)	User interface transformation (HATS, IMS, CICS)	Web service and SOA integration (HATS, IMS, CICS)	Remote API	Remote desktop	Web service and SOA integration (BTT, IMS, CICS)	DRDA o. IP	MQ o. IP	
Access SNA 3270 applications from a Web browser with a Web look and feel			✓							
Enable as a Web service	SNA 3270 server application			✓						
	SNA LU0 or LU6.2 server application						✓			
Remove dependence on outdated hardware technologies	Token-ring									
	IBM 3745/46									
	Channel-attached SNA gateways									
Share IP network for SNA and IP										
Consolidate SNA stacks into the data center		✓	✓	✓	✓	✓	✓			
Remove need for SNA				(✓)			(✓)	✓	✓	✓
Comments								For DRDA traffic	For MQ traffic	

a. Legend:
(✓) - addresses objective, but not primary objective
b. ✓- primary objective

2.3.2 Modernize network infrastructure: quick reference

Table 2-2 Infrastructure modernization technologies

Objectives		Modernization technologies				
		SNA subarea				APPN
		DLSw	XOT	CCL	IP TG	EE
Enterprise Transformation stage		0	0	0	0	0
Retain traditional 3270 screen on WS				(✓)[a]		(✓)
Retain SNA client on WS				(✓)		(✓)
Access SNA 3270 application from a Web browser with a Web look and feel						
Enable as a Web service	SNA 3270 server application					
	SNA LUO or LU6.2 server application					
Remove dependence on outdated hardware technologies	Token-ring			✓[b]		(✓)
	IBM 3745/46			✓		(✓)
	Channel-attached SNA gateways			✓		(✓)
Share IP network for SNA and IP		✓	✓	(✓)	✓	✓
Consolidate SNA stacks into the data center						
Remove need for SNA						
Comments						

a. Legend:
(✓) - addresses objective, but not primary objective
b. ✓- primary objective

3

Modernizing an SNA network infrastructure

There are two main dimensions to modernizing an SNA network infrastructure:

- SNA architecture ranging from SNA subarea through APPN/ISR, to APPN/HPR including HPR over IP (EE), where such support exists
- Network consolidation and simplification, consisting of two sub-dimensions:
  - Consolidating the transport of SNA data end-to-end, with transport of IP data over a common shared enterprise-wide IP network infrastructure
  - Consolidating SNA infrastructure functions and ultimately SNA protocol stacks into the data center

Important: You can meet most SNA modernization objectives with both an SNA subarea environment and an APPN environment. However, the technologies that you choose—especially for network infrastructure modernization—will vary, based on which SNA architecture level you decide to move forward with.

APPN/HPR is the most function-rich SNA architecture level. It is generally the preferred level for SNA networks today. However, if the existing SNA infrastructure is based on an SNA subarea architecture level and is considered to provide an adequate level of service, a migration to APPN/HPR may not be necessary or may require skills that are not readily available.

Note: You may be left with some SNA subarea node types even if you enable APPN in your SNA environment. Therefore, the choice between SNA subarea and APPN is not always an either/or decision; it is often a combination. For example, consider an environment where you base your intranet SNA setup on APPN with HPR over IP. However, you have a few business partners that only support SNI connectivity. In such a case, you may still require an NCP to act as the gateway NCP to those business partners, while all your intranet SNA traffic flows over IP as APPN/HPR data.

3.1 The SNA architecture dimension

From an SNA perspective, there are obvious advantages to moving an SNA subarea-only infrastructure into an APPN-enabled infrastructure. However, there are also costs associated with doing so, such as acquiring additional SNA APPN skills, designing and implementing administrative changes to network definitions on a multitude of SNA nodes, and updating the network management portfolio and procedures to handle APPN network management.

On the other hand, investing up front in enabling APPN may prove to be more cost-efficient in the long run. If you are able to move forward from an SNA subarea environment into APPN with HPR, you can accomplish the consolidation of both SNA and IP traffic on a single IP-based network infrastructure simply by using HPR over IP.

HPR over IP allows you to implement IP connectivity end-to-end (from the users' workstations right into the data center—or in the case of z/OS—right into z/OS itself) without depending on any traditional SNA infrastructure components. This end-to-end IP connectivity can include both intranet SNA access and, where z/OS is involved, business partner SNA access based on the APPN Extended Border Node (EBN) functions on z/OS.

If you decide to stay either totally or partly with an SNA subarea environment, you must analyze how and to what extent you can meet the general objectives of SNA network infrastructure modernization in such an SNA subarea environment.

Until recently, staying with SNA subarea technologies meant depending on one or more SNA networking hardware technologies that were no longer marketed by IBM or other network equipment providers. Such technologies include, but are not limited to:

- IBM 3745/46 Communication Controller.
- Token-ring technology (IBM 3745/46 only supports token-ring for SNA LAN connectivity to an NCP).
- Various ESCON channel-attached SNA-related gateways (including IBM 2216, Cisco CIP, and Cisco CPA technologies).

In February 2005, IBM announced the next generation Communication Controller to replace the IBM 3745/46 for the most commonly used NCP-based functions: IBM Communication Controller for Linux on System z (CCL). CCL addresses a number of overall infrastructure modernization objectives for an SNA subarea environment by providing the following:

- A next-generation Communication Controller technology on which the NCP and NCP Packet Switching Interface (NPSI) can continue to operate, including both SNA boundary functions and SNA subarea-based business partner connectivity (SNI).
- A token-ring LAN connectivity replacement solution, where the SNA token-ring LAN connectivity can be replaced by Ethernet technology, including the current Gigabit and 10 Gigabit Ethernet technologies.
- A significant throughput improvement for an NCP workload operating in a CCL environment, when compared to the same NCP workload operating in an IBM 3745/46 environment.

In addition to offering a new platform for the NCP, the CCL solution also provides several SNA subarea-based connectivity technologies that address the objective of consolidating SNA network traffic onto an IP-based infrastructure:

- Data Link Switching (DLSw) imbedded into CCL itself, allowing other DLSw nodes to exchange SNA data over TCP connections into Linux on System z, where CCL resides.

- X.25 Over TCP/IP (XOT) that provides TCP-based transport of X.25 packets between NCP Packet Switching Interface (NPSI) in CCL and an XOT router in the network where X.25 circuits with both SNA and non-SNA X.25 devices are attached.

  Note: CCL XOT is not part of the IBM CCL product itself, but is available as an additional offering.

- IP Transmission group (IP TG) for direct TCP-based connectivity over an IP network between two CCL NCPs for INN or SNI traffic.

Important: The most important benefit of the CCL technology, from an SNA modernization perspective, is that you now have a choice: you can stay with SNA subarea, or you can move to APPN. In either case, you can modernize your SNA infrastructure to meet most of the overall SNA modernization objectives.

3.2 The network consolidation and simplification dimension

You can use different alternatives to transport SNA over an IP network:

- At a link layer, using DLSw or IP TG
- At a networking level, using EE
- At an API level, using remote SNA API client/server
- At an application level, using application-layer protocols such as TN3270 or HTTP(S)/HTML

Most often transporting SNA over IP is not the only objective for modernization activities. Other SNA modernization objectives, such as reducing SNA node complexity by consolidating SNA protocol stacks into the data center (see Figure 3-1 on page 26), may influence your choice of technology to transport SNA over IP.

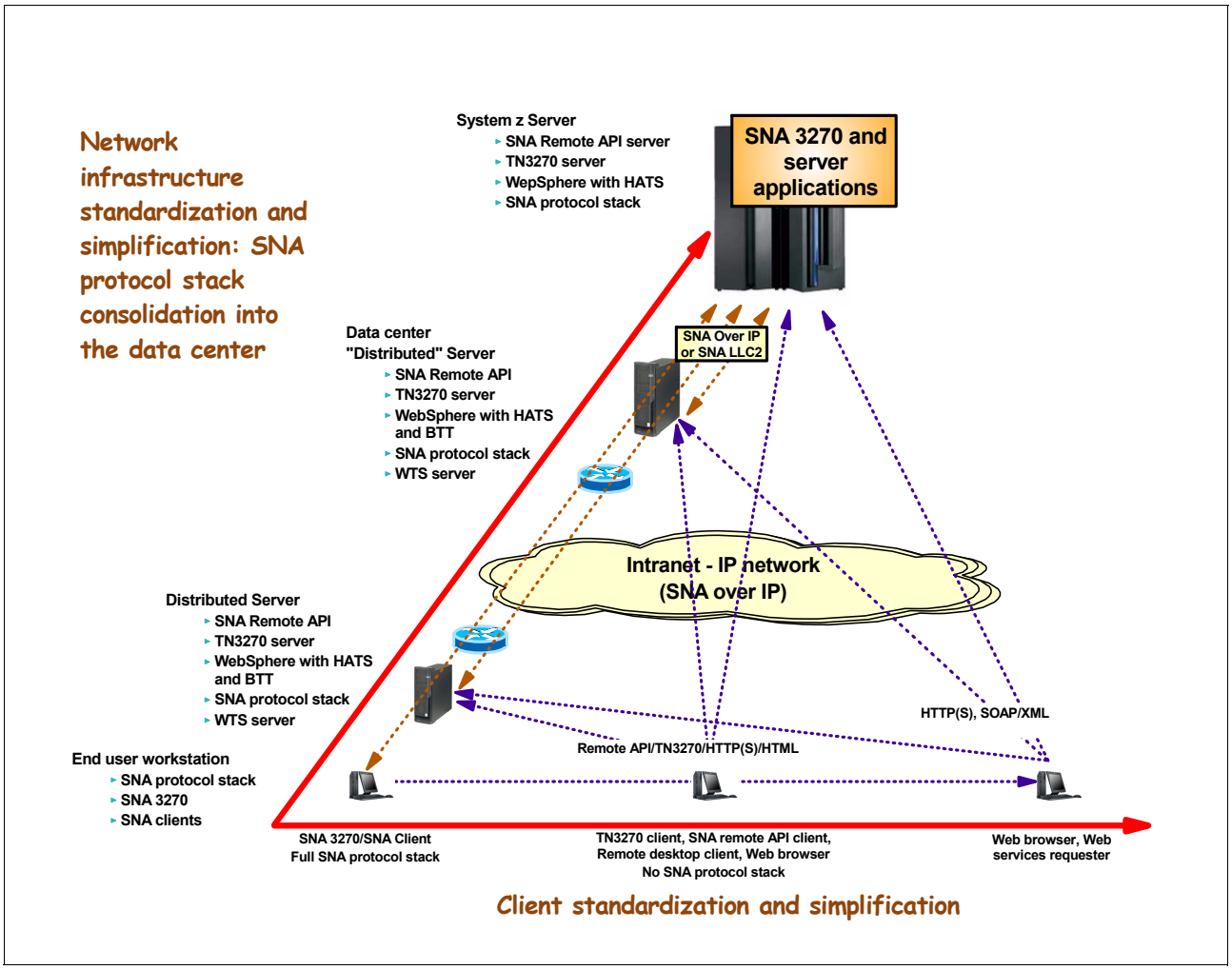


Figure 3-1 SNA protocol stack consolidation and client simplification

Note: The fewer SNA nodes with full SNA protocol stacks in the SNA network, the simpler the SNA topology becomes. The simpler the SNA network topology is, the easier it is to maintain and manage (and the easier it is to modernize).

With the technologies that are available today, it is possible to preserve the remote SNA user interface or client SNA application on a user workstation or branch server, without the need for a full SNA protocol stack on those nodes.

Because the volume of SNA access from remote locations most likely will decline over the coming years as thin client technologies roll out and Web services are deployed, we believe a consolidation of SNA protocol stacks from the end-user workstation—potentially via branch servers or branch SNA routers to the data center and to the mainframe itself—will take place. Such a consolidation will be motivated by reduced software costs due to removal of SNA protocol stack software in the branch environment, and by much simpler administration and management procedures.

The main technologies in support of such a consolidation are the API and application layer SNA access modernization technologies, such as TN3270, remote SNA API, SNA 3270/HTML transformation, Web services integration, and so on. Which of these

technologies you choose depends on your client platform standardization and simplification strategies such as Web browser, .NET clients, and Java application clients.

Note: By consolidating the SNA protocol stacks into the data center, you avoid the issue of choosing between SNA subarea and APPN-based technologies for transporting SNA over IP to some degree. There is hardly any SNA network traffic to transport outside the data center.

SNA subarea or APPN technologies for routing SNA data between nodes in the data center, such as z/OS systems in a z/OS Sysplex, will still be required.

There will be cases where the existing API and application layer technologies may not be able to fully address the needs of some remote SNA applications. In these cases, SNA protocol stacks will remain in remote locations.

- For Windows workstations or servers, you can use Microsoft Host Integration Server, IBM PCOMM with a full SNA protocol stack, or IBM Communications Server for Windows.
- For Linux workstations or servers, you can use IBM Communications Server for Linux.
- For AIX servers, you can use IBM Communications Server for AIX.

When connecting those remote SNA nodes to the data center, you will need to decide between APPN and SNA subarea infrastructure modernization technologies for transporting SNA over an IP network.

3.3 Modernizing the SNA subarea network infrastructure

In an SNA subarea environment, an NCP is a crucial element for both SNA boundary functions and business partner communication based on SNI technologies. Preserving the functions of an NCP becomes one of the main objectives in such an environment.

However, there is also a need to investigate to what extent various SNA application access modernization technologies, such as TN3270, IBM remote client/server APIs, SNA 3270/HTML transformation and so on, can be deployed in an SNA subarea environment without requiring use of APPN-based technologies.

3.3.1 Preserving NCP functions

CCL allows an installation to continue to use an NCP (and, optionally, also NPSI) by moving the software from an IBM 37xx Communication Controller to Linux on System z in an LPAR or in a z/VM® guest. The NCP may perform SNA boundary functions, SNI functions, or both. CCL, as mentioned earlier, supports both traditional NCP functions and NPSI (SNA and non-SNA X.25) functions.

Note: For an existing NCP-based environment, CCL is the modernization technology that will require the least number of changes to the overall SNA subarea topology. It is also the preferred modernization solution where continued use of an NCP is a requirement.

An NCP running in CCL has its normal LAN connectivity functions available. From an NCP perspective, both TIC2 and TIC3 IBM 3745/46 token-ring adapters are available. The CCL infrastructure shields the NCP from the complication that the real LAN to which the OSA adapter is connected may, in fact, be an Ethernet.

If an installation continues to require support for serial lines, a small change in the overall topology is required. Because CCL runs inside a System z server, there are no serial line interface capabilities available for terminating those serial lines directly on the System z server. Serial lines need to be terminated in an aggregation layer router instead.

Such a router is a traditional router, such as a Cisco 7200 or 3800 family router, with serial line interfaces. The router uses internal switching or bridging technologies to switch the serial line SNA data to a LAN as SNA LLC2 traffic, or uses DLSw to transport the SNA data between the router and CCL. The NCP sees the serial line devices as though they were connected over its LAN interfaces.

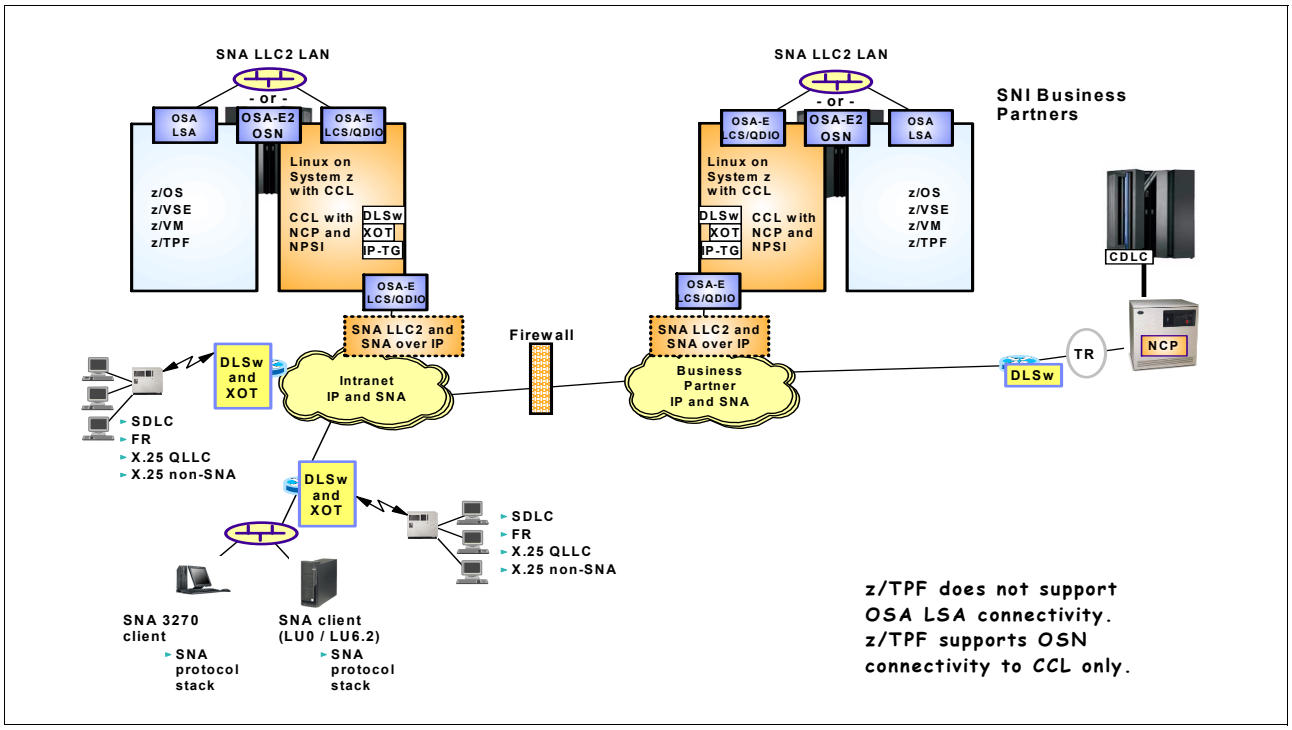


Figure 3-2 Sample CCL NCP topology

SNA access to an NCP running in CCL (Figure 3-3 on page 29) will in some cases be implemented via DLSw technology as follows:

- A branch router that switches SNA LLC2 frames between the branch LAN and an intranet IP network
- A router at a business partner location that switches SNA LLC2 frames between a token-ring LAN to which the partner IBM 37xx is connected, to an inter-enterprise IP network
- The local data center DLSw endpoint where the CCL NCP is deployed may be:
  - Imbedded DLSw in CCL (allows IP-based connectivity all the way into CCL)
  - A data center DLSw router that switches SNA LLC2 frames between the backbone IP network and a data center LAN to which CCL is connected through an OSA port using SNA LLC2 access

DLSw technology is a convenient, well-known, and simple technology that in many cases already is in use for transporting SNA traffic over the company IP backbone network between a branch and the data center LAN environment.

CCL, in combination with DLSw, IP TG, and XOT, will allow wide area network (WAN) communication to occur over IP, consolidating what may be left of LAN SNA LLC2 traffic to be within the data center itself.

Limited SNA subarea-based alternatives to NCP

For INN/SNI connectivity, there are no simple alternatives to the continued use of at least one NCP.

You can establish SNI connectivity to an SNA subarea business partner using only a single NCP connected to the business partner VTAM through an OSA LSA port. The business partner SNA NETID must, in that case, be made part of the gateway SNI NCP (in a single gateway SNI topology), extending the business partner subarea SNA network topology into the gateway NCP; see Figure 3-3.

That may be acceptable to some business partners, but probably not to all. For maximum SNA network topology separation and isolation, the most widely-used SNI topology is based on the use of back-to-back gateway NCPs, in which case both business partners require an NCP.

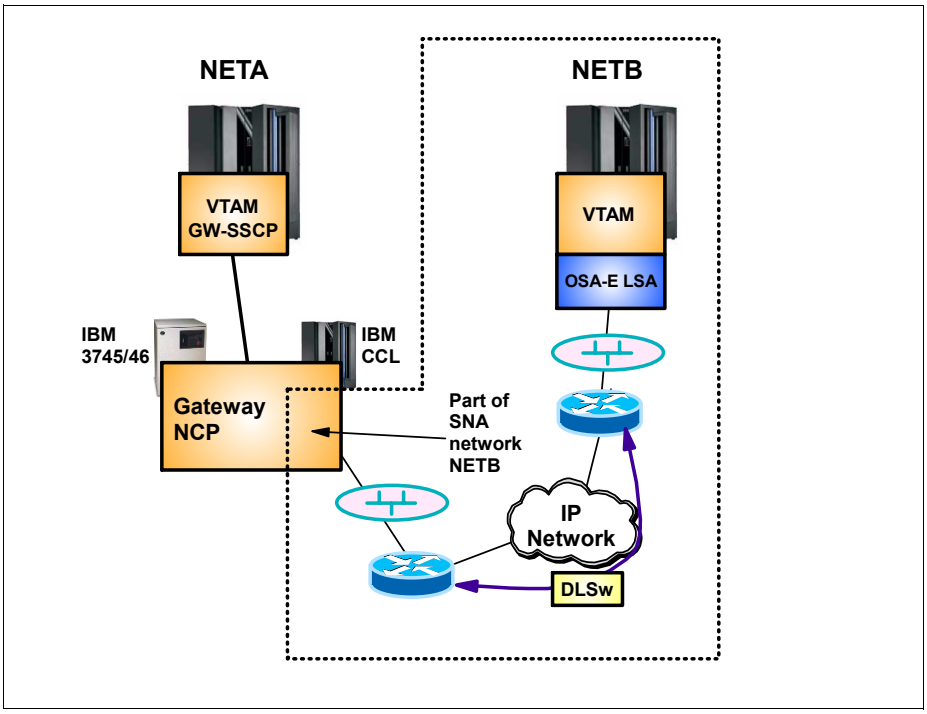


Figure 3-3 SNI using a single NCP

For boundary functions, there are some alternatives in the SNA subarea environment. One alternative is to let VTAM itself perform the boundary functions. VTAM can use an OSA port in LSA mode to attach to a LAN, and use SNA LLC2 communication over that OSA port.

For local SNA nodes in the data center, SNA LLC2 connectivity is generally readily available. For remote SNA nodes, DLSw technologies over an IP WAN infrastructure is typically used to a data center DLSw router that switches the SNA flows back to LAN LLC2 flows, and connects to VTAM over the OSA LSA port.

Following are three important aspects of using VTAM for boundary functions:

- VTAM will use low-order element addresses for the LUs it performs boundary functions for. The low-order element addresses are limited to 64,000 and this limit has, in a number of cases, become an issue for installations with many dependent LUs defined, especially when consolidating the boundary functions of multiple NCPs into a single VTAM. When the NCP performs the boundary functions, VTAM does not use its own low-order element addresses for the NCP-owned resources.
- VTAM will use general mainframe CPU resources to perform the SNA boundary functions and to route SNA session data between the dependent LUs and primary LUs in the SNA network, which will mean an increase in VTAM CPU usage. When the NCP running in CCL on the mainframe performs the boundary functions, the mainframe CPU resources may be Integrated Facility for Linux (IFL) CPU resources.
- If the boundary functions are being moved from an NCP environment, having VTAM perform boundary functions may also result in lower overall availability characteristics since there are no takeover functions available in that case; refer to Figure 3-4.

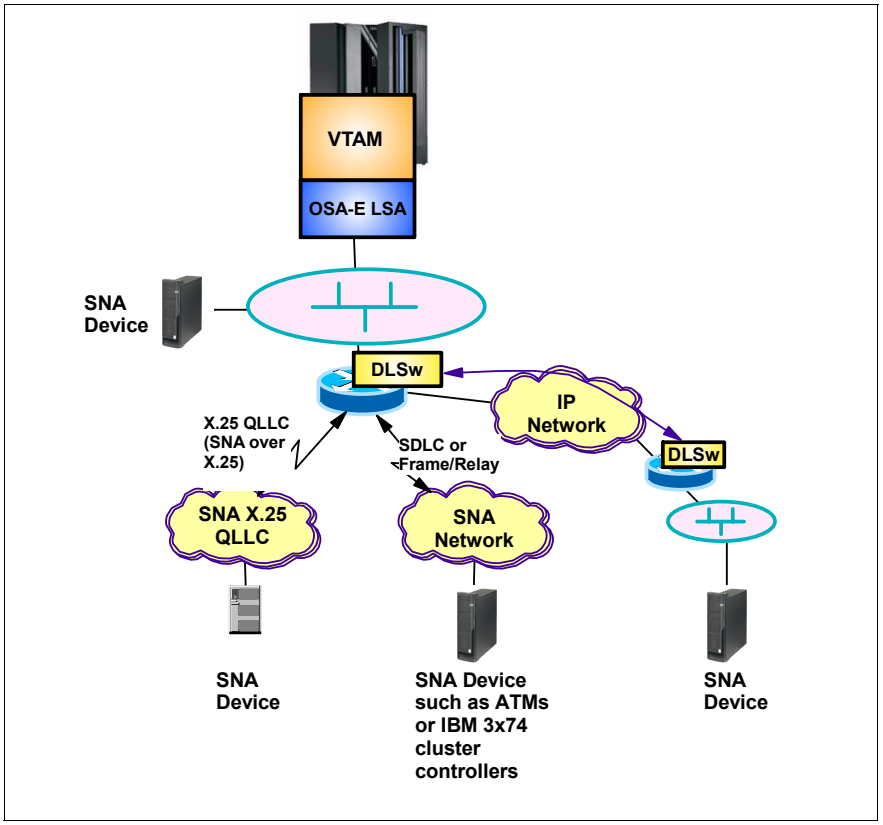


Figure 3-4 VTAM performing boundary functions

For a low number of dependent LUs, using VTAM to perform SNA boundary functions is a realistic alternative to an NCP. For large numbers of LUs, however, be mindful of running out of low-order element addresses in VTAM and CPU resources that VTAM will use to perform the boundary functions.

This topology does not support non-SNA X.25 connectivity, which requires NCP Packet Switching Interface (NPSI). SNA connectivity over X.25 (QLLC) is supported through an aggregation layer router.

If any of the downstream SNA devices are IBM 3745/46 Communication Controllers with NCPs, those NCPs cannot be loaded or owned by VTAM over an LSA interface. In other words, VTAMs with an OSA LSA interface can communicate with the IBM 3745/46 controllers only if a local VTAM that loads and owns the NCP resources is channel-connected.

3.3.2 Overview of IBM Communication Controller for Linux (CCL)

CCL is a software product that emulates the IBM 3745/46 hardware environment in support of a Network Control Program (NCP) and NCP Packet Switching Interface (NPSI). As mentioned, CCL is the next generation IBM Communication Controller technology, replacing the IBM 3745/46 for a range of selected, commonly used NCP-based functions.

CCL runs in Linux for System z in a System z LPAR or in a z/VM guest. CCL does not run on any other hardware platform than the System z platform. If your System z server has Integrated Facility for Linux (IFL) engines installed, then you can direct the CCL and NCP workload to those IFLs.

Not all functions that are supported in an IBM 3745/46 are supported in a CCL environment; however, the commonly-used NCP-based and NPSI-based functions are supported. Refer to Appendix E, “IBM Communication Controller for Linux - functional overview” on page 109, for details about what is supported in a CCL environment.

In 2002, IBM announced that the IBM 3745/46 hardware was withdrawn from marketing, meaning that it could no longer be ordered from IBM. When that announcement was made, there was no replacement technology that would allow continued use of an NCP.

For many of the NCP-based functions, there were various alternatives. Most of these alternatives were based on a migration from an SNA subarea network environment to an SNA APPN network environment. For some installations, however, this was an issue. Migrating from the SNA subarea-based technology for business partner connectivity (SNI) to the APPN-based technology (EBN) requires coordinated migration activities between business partners, which is not always possible.

CCL: SNI connectivity

The NCP software has not been withdrawn from marketing. Service and support of the IBM 3745/46 hardware has not, at this point in time, been withdrawn either.

One of the primary objectives of CCL is to preserve the capability of SNI connectivity between business partners, where such connectivity was based on SNI and there was a need to continue to base such connectivity on SNI; refer to Figure 3-5 on page 32.

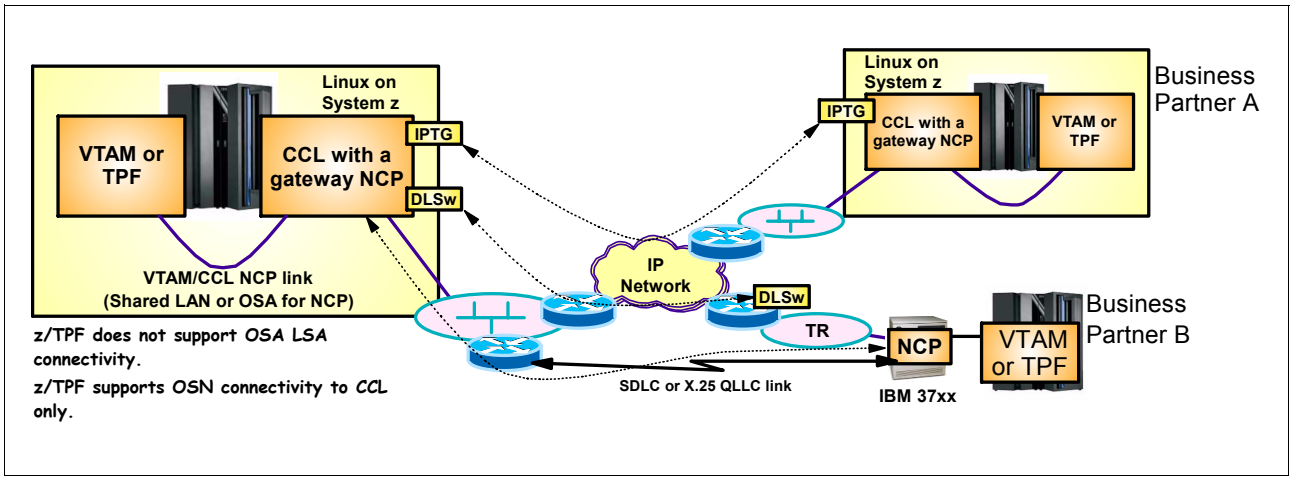


Figure 3-5 CCL preserving SNI connectivity

CCL does not impose any restrictions on which SNI gateway model is being used in the NCP definitions. It can be a single gateway or a back-to-back gateway, as required by the partners:

- If partner NCPs also run in CCL, then NCP-to-NCP connectivity (INN or SNI) can be accomplished using the CCL built-in IP Transmission Group technology that uses IP from CCL to partner CCL, thus totally avoiding any SNA network traffic. IP TG uses a single TCP connection per partner NCP with a pre-configured TCP port number, which makes firewall administration in between the NCPs an easy task.
- If partner NCPs run in IBM 37xx hardware, then connectivity from CCL can either be via DLSw or via SNA LLC2 to a data center DLSw router. Note that if the partner NCP is connected to your data center using line connectivity, such as an SDLC line, then that line can be terminated in a router with serial line interfaces in your data center. The router will then use internal switching technology to switch the SNA data between the SDLC line and a LAN to which the CCL NCP is also connected over an OSA port.

  In this case, the network level traffic over the OSA port is SNA LLC2 LAN frames and not IP packets (as would be the case for IP TG or DLSw, when DLSw is terminated inside CCL). From an NCP perspective, it looks as though the partner NCP is connected over a LAN interface instead of a serial line interface; however, the SNI topology and SNI-related capabilities do not change.

CCL: boundary functions

A secondary objective of CCL is to cover SNA boundary functions (as shown in Figure 3-6 on page 33) for as broad a range of physical connectivity options as possible. Because CCL runs on System z hardware, and System z hardware does not support direct serial line attachments, CCL relies on routers in the data center to terminate serial lines and switch the SNA traffic to a LAN environment to which System z very efficiently connects through OSA ports.

From an NCP topology point of view, it looks as though all peripheral nodes are connected through a LAN interface to the NCP. An NCP only supports token-ring LANs, which obviously is an issue because one of the SNA modernization objectives is to provide a solution for token-ring replacement. CCL addresses this by providing a built-in conversion between Ethernet LAN and what the NCP believes is a token-ring LAN.

The OSA port may be connected to an Ethernet LAN; however, CCL will hide that fact from the NCP by internally converting frame formats between Ethernet formats and token-ring formats. From an NCP generation perspective, you still define Token-ring Interface Coupler

(TIC) line addresses (TIC2 or TIC3 addresses). The actual OSA LAN interface, however, may be an Ethernet interface.

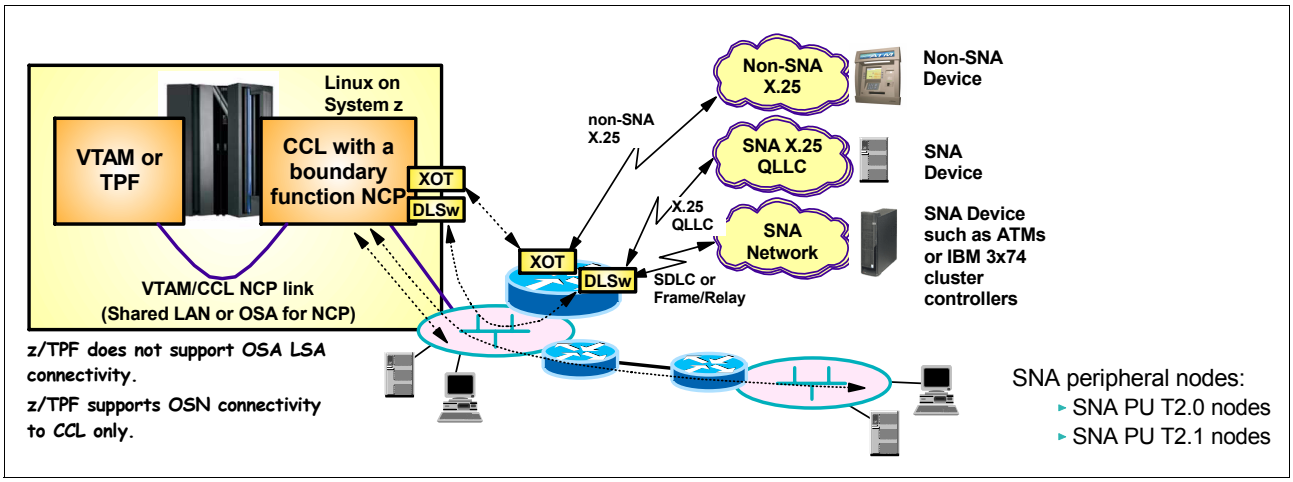


Figure 3-6 CCL preserving selected boundary functions

Use DLSw technology for peripheral node connectivity over IP between the location of the peripheral node and CCL. For local peripheral nodes, CCL supports traditional SNA LLC2 connectivity over Ethernet and token-ring.

Peripheral node connectivity over SNA-specific wide area link technologies (such as SDLC, FR, ISDN, or X.25 QLLC) is supported by terminating the links in an aggregation layer router node in the data center that implements link layer switching between the serial line interfaces and SNA LLC2 on a LAN, or connects to CCL over IP via DLSw.

CCL: connectivity to VTAM and z/TPF

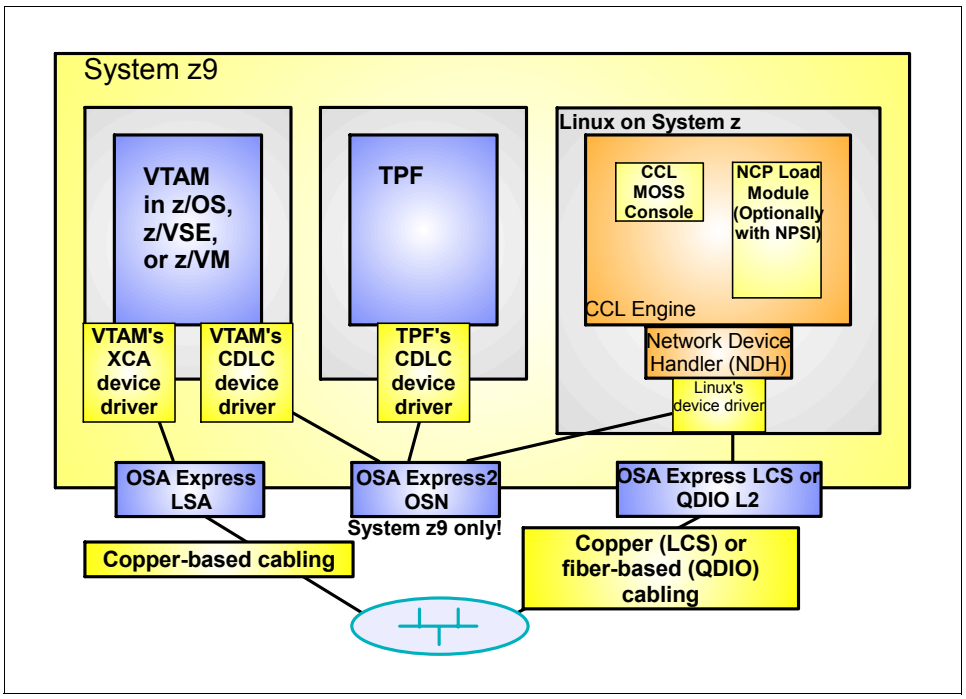


Figure 3-7 CCL connectivity to VTAM and z/TPF

As shown in Figure 3-7 on page 33, CCL may connect to other System z operating systems using one of the following technologies:

- SNA LLC2 LAN connectivity, where CCL connects to the LAN via an OSA port operating in LAN Channel Station (LCS) mode (CHPID type OSE) or in QDIO layer-2 mode (CHPID type OSD), and VTAM connects via an OSA port operating in Link Station Architecture (LSA) mode (CHPID type OSE).

  LCS and LSA modes are limited to using copper-based ports: token-ring or Ethernet up to and including 1000BASE-T Ethernet. QDIO can use copper-based or fiber-based ports, including both Gigabit and 10 Gigabit Ethernet ports.

  The SNA LLC2 LAN connectivity can be used for both downstream communications to peripheral and subarea nodes, and upstream (to VTAM). QDIO layer-2 mode supports virtual Medium Access Control (MAC) addresses, which is an important feature for an SNA LAN-based workload where SNA LAN end points are identified via their MAC addresses.

  Virtual MAC addressing allows for multiplexing many SNA LAN end points onto a single physical OSA port, and simplifies migration from an IBM 3745/46 environment with many TIC adapters. VTAM in z/OS, z/VM, and z/VSE™ can connect to a CCL NCP over a LAN, but z/TPF cannot.

- CDLC channel connectivity via a shared OSA Express2 port configured in OSA for NCP (CHPID type OSN) mode.

  Such an OSA port is supported on System z9™ servers only. When CCL and VTAM or z/TPF reside on the same physical System z9 server, an OSA Express2 port can be shared between them.

  When configured in OSN mode, the adapter allows a connectivity that, from a VTAM, z/TPF and NCP perspective, appears to be an ESCON channel over which the traditional CDLC channel protocol can be used. This preserves existing NCP load/dump/management operator procedures. VTAM in z/OS, z/VSE, z/VM and the SNA support in z/TPF all support OSN connectivity to a CCL NCP.

CCL: downstream connectivity

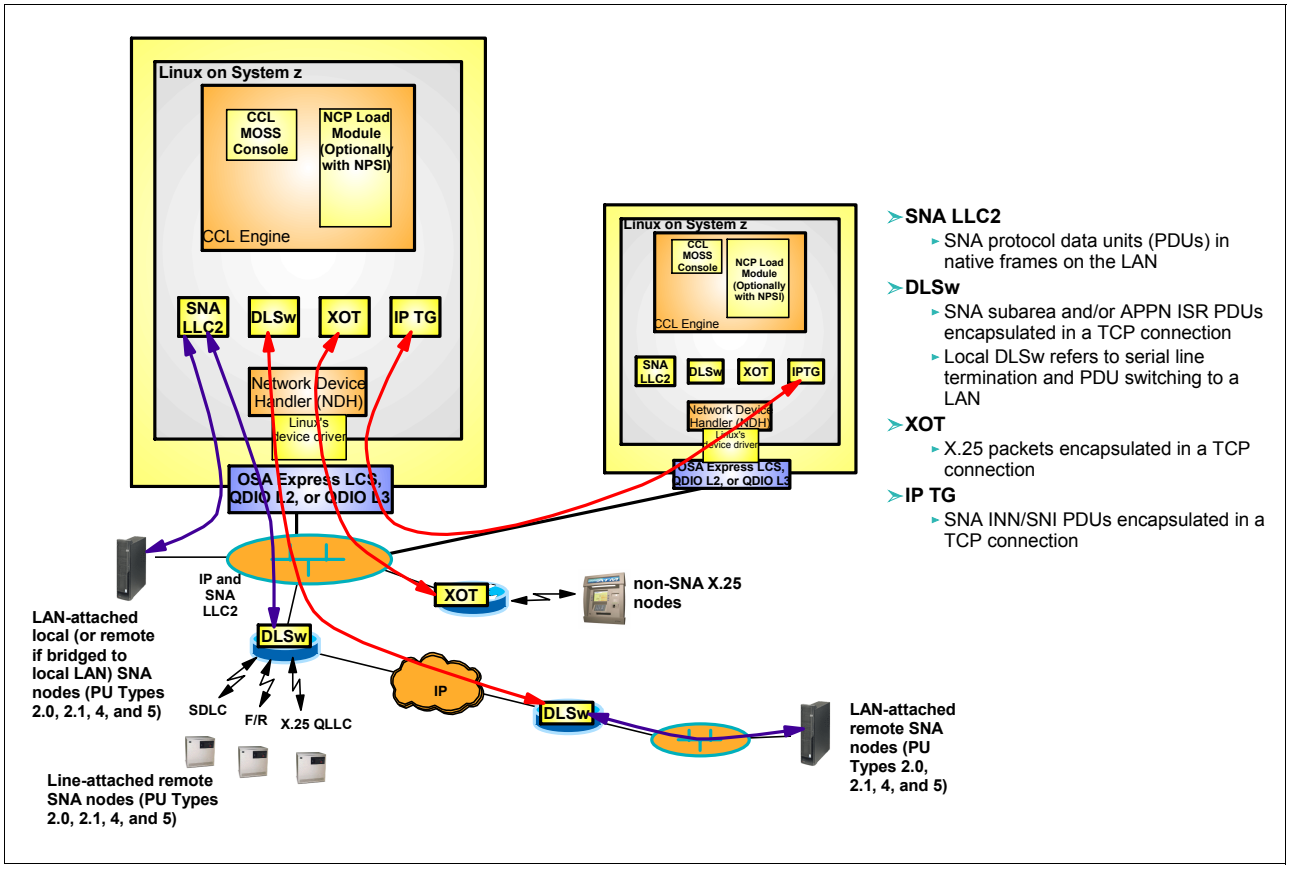


Figure 3-8 CCL downstream connectivity overview

If non-SNA X.25 connectivity through NCP Packet Switching Interface (NPSI) is required, CCL can connect to X.25 circuits via the X.25 over TCP/IP protocol (XOT) to a router; refer to Figure 3-8 for an overview of CCL downstream connectivity options. This allows an installation to preserve existing mainframe NPSI-based applications unchanged.

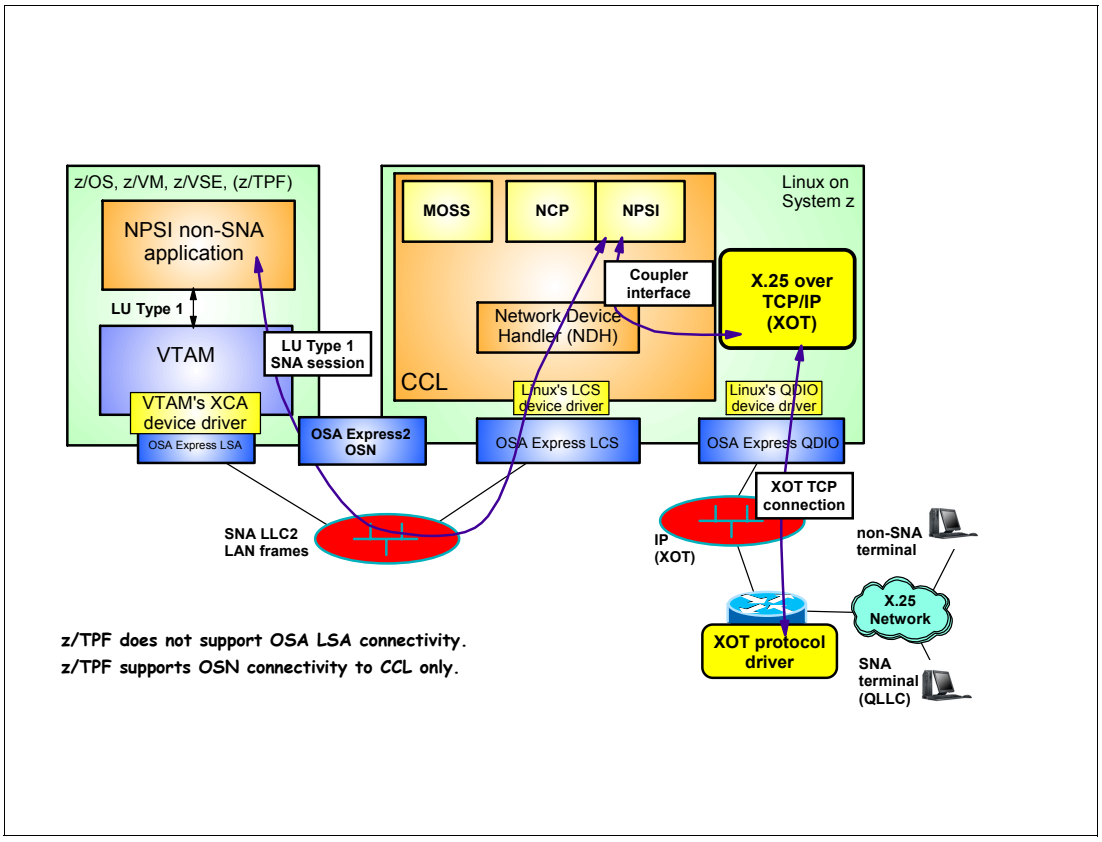


Figure 3-9 CCL with NPSI for non-SNA X.25 connectivity

The XOT protocol is a standard protocol that is supported by various router vendors. XOT is defined in RFC 1613 “Cisco Systems X.25 over TCP (XOT)”.

NPSI can in general be used for both non-SNA and SNA over X.25 (QLLC). QLLC links can be supported by an NCP in a CCL environment without NPSI, simply by terminating the QLLC links in a router the same way as other SNA links are terminated in such a router. If an installation today uses NPSI for QLLC links, then it may be the simplest to continue to do so, but it can potentially be done without NPSI.

CCL: support restrictions

A few IBM 3745/46-based functions which a CCL environment does *not* support are:

- SNA Binary Synchronous Communication (BSC) 3270 connectivity - 3270 cluster controllers attached via a BSC line to an NCP
- Air Lines Control (ALC)
- TCAM start/stop lines
- X.25 Interconnect (XI) and Network Supervisory Function (NSF) transport of X.25 packets
- Network Terminal Option (NTO)
- Non-SNA Interconnect (NSI), and MERVA (SWIFT) connectivity
- Emulation Program (EP) and Partitioned Emulation Program (PEP) are not supported

If you are using any of these functions in your IBM 37xx environment, then migrating to CCL will not provide a complete solution for you. Instead, you will need to investigate alternative

solutions for each of the functions, which may mean replacing applications or rewriting applications that depend on these technologies. Alternatively, you may benefit from migrating most of your NCP workload to CCL and maintaining one IBM 37xx controller for the functions that are not supported in a CCL environment. Refer to *IBM Communication Controller Migration Guide*, SG24-6298, for suggestions about how to migrate those functions. Refer to Appendix E, “IBM Communication Controller for Linux - functional overview” on page 109 for more details about CCL capabilities.

3.3.3 Data Link Switching (DLSw)

Data Link Switching (DLSw) is an open standard networking function that resides at the Logical Link Layer (LLC) in a networking protocol stack and is used to encapsulate SNA link-level data frames over an IP network, as shown in Figure 3-10.

Data Link Switching is defined in “DLSw Standard Version 1.0”, RFC1795, and “DLSw v2.0 Enhancements”, RFC2166.

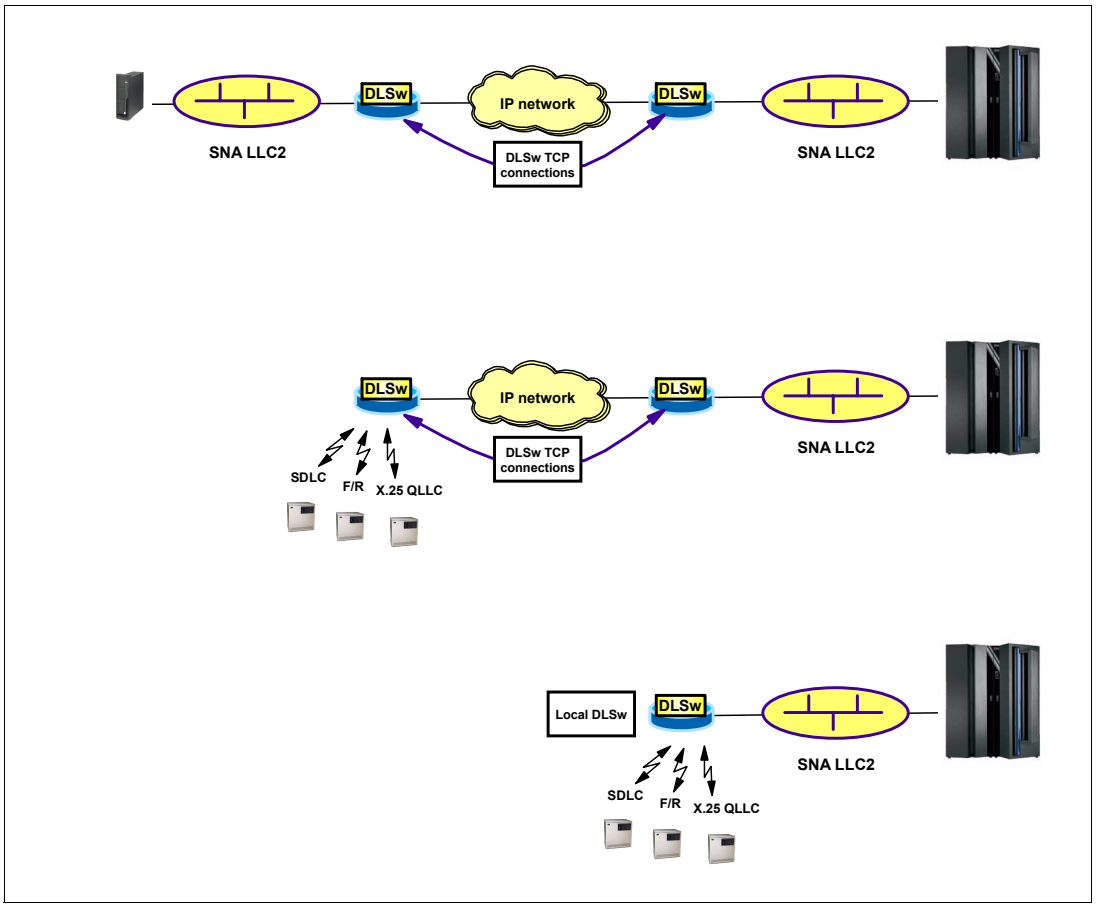


Figure 3-10 DLSw basic topologies

DLSw is typically used in three basic configuration topologies:

- LAN SNA LLC2 - TCP/IP - LAN SNA LLC2
- Serial line (SDLC, X.25 QLLC) - TCP/IP - LAN SNA LLC2 (remote aggregation)
- Serial line (SDLC, X.25 QLLC) - LAN SNA LLC2 (local aggregation)

Each DLSw end point terminates the SNA LLC2 connections, to avoid WAN latency impact on local LLC2 timers and also avoid wasting WAN network bandwidth on LLC2 control flows such as polling.

DLSw uses TCP connections between the DLSw nodes when connecting over an IP network. Because all SNA frames are sent over the same TCP connection, DLSw (unlike EE) cannot preserve the SNA Class of Service (COS) priorities over the IP network.

You can use the open standards version of DLSw to encapsulate SNA subarea flows and APPN/ISR, but not APPN/HPR routing. Vendor-specific variations of DLSw can also, in some cases, be used to transport APPN/HPR traffic, but you would normally use EE for that purpose.

DLSw is incompatible with Multi Link Transmission Groups (MLTG), and it does not support MLTG topologies between NCPs. This incompatibility can be an issue if you are migrating an MLTG connection between two IBM 3745/46s from serial lines to DLSw. In general, MLTG is used to provide load-balancing over multiple serial lines and availability in case one of the lines has a failure. Both of these objectives can be addressed with IP-based technologies in a DLSw setup by proper DLSw router and IP network design.

In an SNA subarea network environment, DLSw provides a simple and inexpensive mechanism to transport SNA data over an IP wide area network with minimal or no impact to the SNA topology and general SNA subarea functions.

CCL has recently shipped support for terminating DLSw connections in the Linux operating system environment where CCL is deployed, as shown in Figure 3-11.

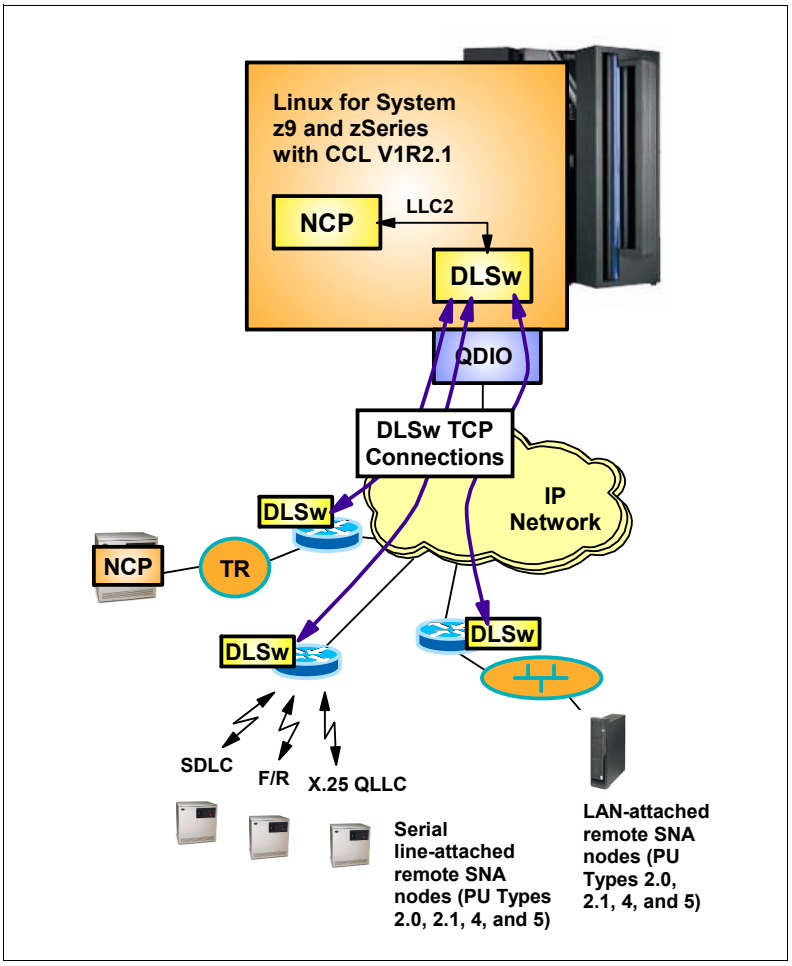


Figure 3-11 CCL DLSw support overview

CCL DLSw support is based on the open standards version of DLSw, as defined in RFC 2166. Some network equipment vendors have expanded this open standard with proprietary extensions. Such implementations will dynamically determine that the other DLSw end point is an open standards version and adapt to that level of the DLSw protocol. Therefore, interoperability between such extended implementations and the CCL open standards DLSw implementation is, in general, expected to work without problems.

3.3.4 IP Transmission Group (IP TG)

IP Transmission Group (IP TG) is a CCL-specific technology that also addresses the objective of transporting SNA over an IP network.

IP TG can be used between two CCL NCPs (as shown in Figure 3-12) that exchange data with each other over a subarea link. Such a link can be an INN link within the same SNA NETID, or an SNI link between enterprises.

When both NCPs are deployed in CCL, CCL can transport that SNA subarea communication over a single direct TCP connection between the two CCL nodes, thereby avoiding SNA WAN flows. IP TG uses a configurable TCP port number, which simplifies firewall administration for SNI connectivity between business partners.

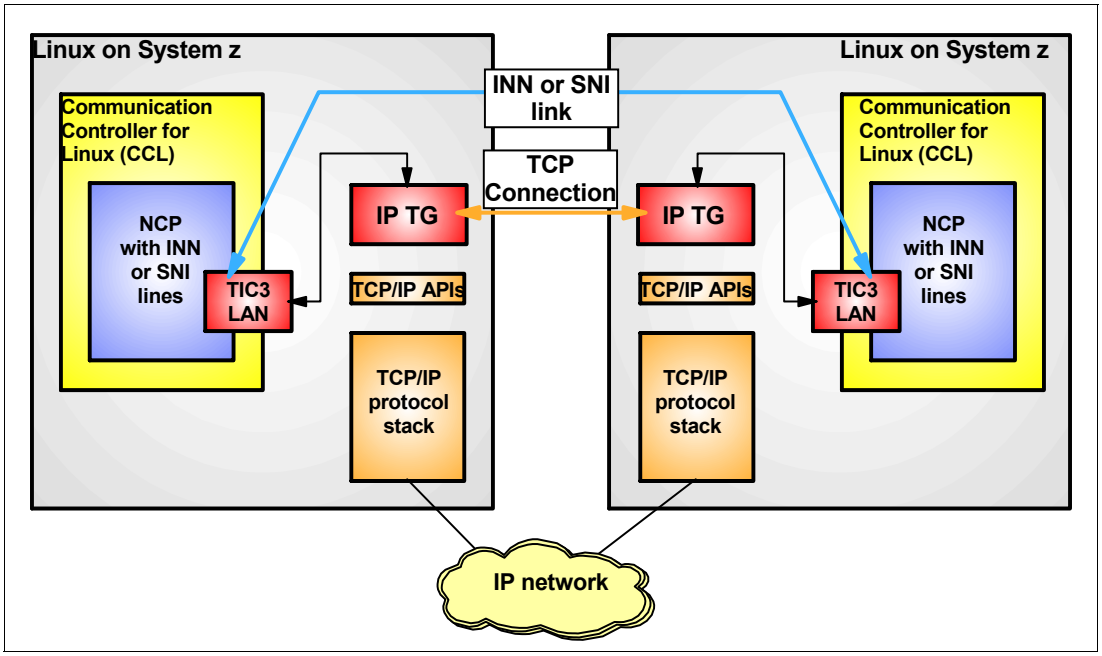


Figure 3-12 IP TG between CCL NCPs for INN/SNI traffic

There is no real LLC2 processing when using IP TG, so IP TG performs very well for INN/SNI traffic between two CCL NCPs. For SNI workloads with a selected set of characteristics, two CCL NCPs connected via IP TG, running on System z9, in combination with the OSN (CDLC) connectivity to VTAM, supports five to six times the number of transactions per second as two similarly connected IBM 3745/46 configurations.

3.3.5 X.25 packets over TCP/IP (XOT)

X.25 Over TCP/IP is an encapsulation protocol that allows transport of X.25 packets over a TCP connection between two XOT routers that are IP network-connected to each other.

From an SNA modernization perspective, XOT is of interest for those that continue to depend on NCP Packet Switching Interface (NPSI). CCL supports running NPSI in combination with NCP. However, since System z does not support X.25 network interfaces, we require an encapsulation protocol that passes X.25 packets in or out of the CCL/NPSI environment as data over a TCP connection to a router where the X.25 circuits are terminated. This is what XOT does.

The Linux for System z XOT component that is required by CCL in support of NPSI connectivity to the X.25 circuits is available as a separate offering. It is not included in CCL. However, you can order and install it along with CCL if you require NPSI X.25 connectivity.

3.3.6 Preserving multiple TIC support in an Ethernet or CCL environment

An IBM 3745/46 supports SNA LAN traffic over token-ring only. Token-ring technology supports multiple token-ring LAN segments that are bridged together using Source Route Bridging (SRB). Each bridged token-ring segment may have a token-ring station connected using the same Medium Access Control (MAC) address. This feature is used extensively in an SNA LAN environment to provide load-balancing of LLC2 connections and redundancy between two NCPs, each typically running in their own IBM 3745/46.

Ethernet technology does not support multiple LAN segments with duplicate MAC addresses. When migrating from token-ring to Ethernet LAN technology for SNA traffic, it is not possible to create a similarly simple implementation. However, you can still achieve the same overall capabilities for both load-balancing of LLC2 connections and redundancy for remote SNA nodes by combining Ethernet technology with the use of DLSw technology; see Figure 3-13.

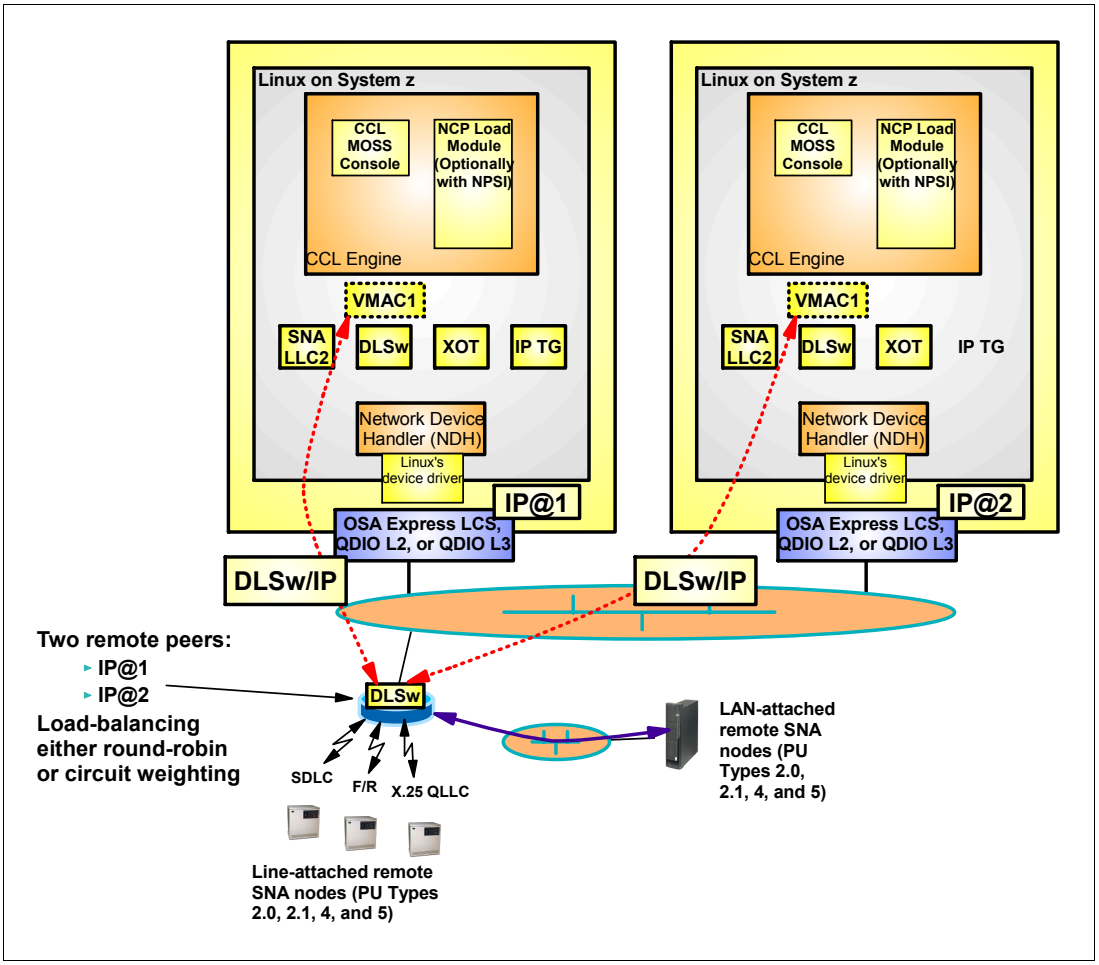


Figure 3-13 Duplicate MAC addressing with Ethernet and DLSw

In this example, the NCP MAC address (VMAC1) is defined on physical token-ring lines in both NCPs. The remote DLSw node has two DLSw peer nodes defined (IP@1 and IP@2) and a load-balancing algorithm to use when both of the two peer nodes are available. The load-balancing algorithm is typically a simple round-robin or some form of a weighted round-robin algorithm. The end result is the same as what was possible for a token-ring LAN environment: load-balancing and redundancy.

3.3.7 SNA application access modernization considerations

This section discusses issues that are specific to SNA access modernization technologies when deployed in an SNA subarea-based infrastructure. (For detailed information about SNA application access modernization, refer to Chapter 4, “How to the modernize SNA application access” on page 57 and to product-specific documentation.)

If the mainframe operating system environment is IP-enabled, then it would be an obvious choice to locate SNA application access modernization solutions in the mainframe operating system itself. Most operating systems on the mainframe support TN3270 server technologies

along with other SNA application access modernization solutions. Housing a TN3270 server on z/OS, z/VSE, or z/VM can be done without affecting VTAM's subarea status.

Generally, SNA application access modernization solutions are implemented in the data center to support IP connectivity from remote locations and to consolidate SNA connectivity to inside the data center itself. If VTAM is subarea-only and IP or SNA application access modernization technologies are not enabled or deployed on the mainframe operating system—but are still required in the installation—then a front-end SNA node that is subarea-attached to VTAM can be deployed in the data center.

The distributed communications servers from IBM all support a range of SNA application access modernization technologies, as well as the capability to appear to VTAM like a traditional peripheral SNA node (PU Type 2.0 or 2.1 LEN node) over an OSA LSA interface (if VTAM performs the boundary functions), or through a CCL NCP (if the NCP performs the boundary functions).

From the perspective of consolidating SNA protocol stacks into the data center, the IBM remote SNA API technology (remote SNA API), X-Windows, or the Microsoft Windows Terminal Services (WTS) technology (remote GUI) are all of interest. These technologies provide a way for a user at a remote workstation to work with what appears to be a local SNA application, when there is no SNA protocol stack on the workstation. SNA protocol stacks only reside on servers in the data center (or, optionally, in the branch).

The distributed communications servers from IBM, in combination with IBM WebSphere Application Server technology, also offer Web-based integration in the SNA subarea environment through IBM WebSphere Host Access Transformation Services (HATS) for SNA 3270 access and through the IBM Business Transformation Toolkit (BTT) for SNA client/server access. Refer to 4.3, “SNA application access transformation” on page 71 for more details about these technologies.

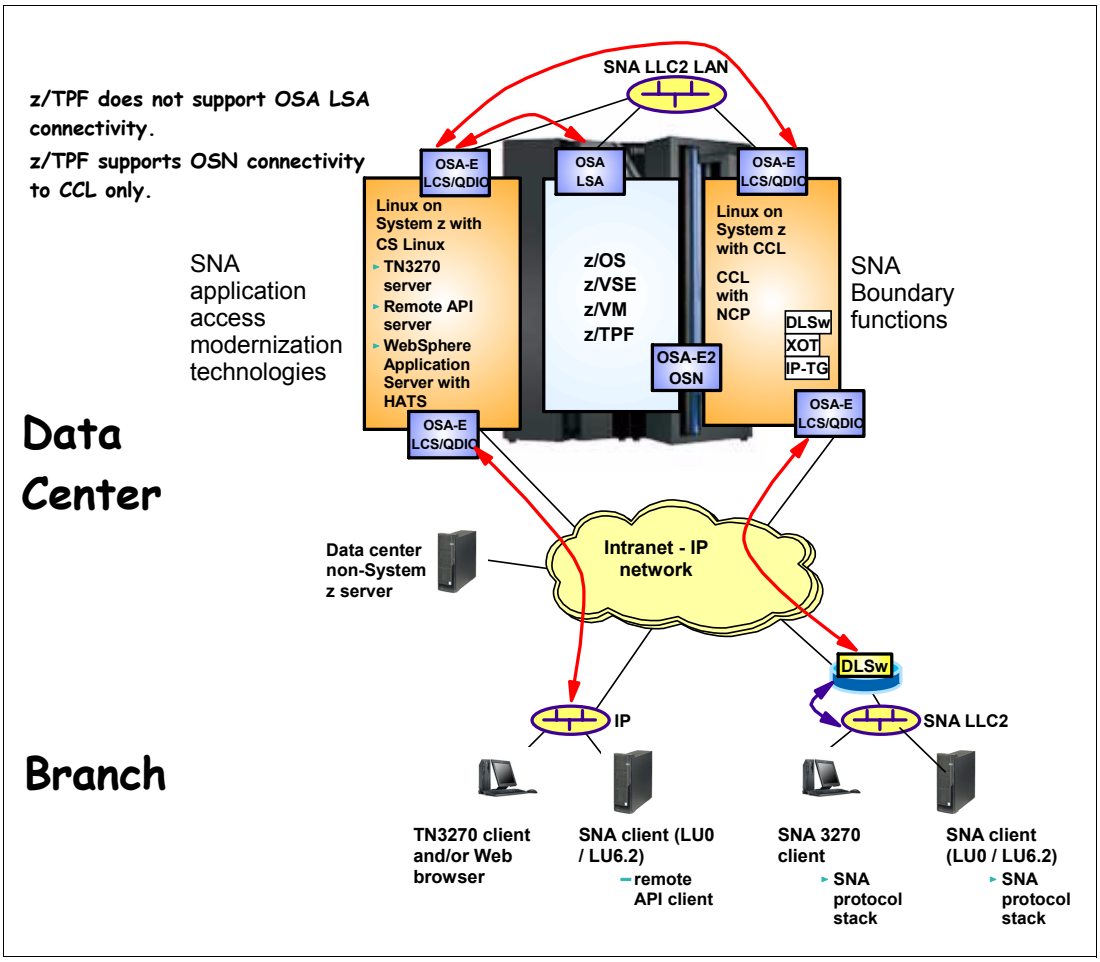


Figure 3-14 SNA application access modernization on System z in an SNA subarea environment

The distributed communications server in the data center may be implemented on:

- An X-series server running either Windows (CS/Windows) or Linux for Intel (CS/Linux for Intel)
- A Power-based server running Linux for Power (CS/Linux for Power) or AIX (CS/AIX)
- A System z server, in an LPAR or as a guest under z/VM, running Linux for System z (CS Linux for System z)

In an SNA subarea environment, the SNA connectivity between the distributed communications server and VTAM or an NCP would be based on a shared LAN to which both the distributed communications server and VTAM or NCP are connected.

z/TPF does not support the OSA LSA technology. A distributed communications server must connect through a boundary function NCP running in a CCL environment that is OSN-attached (CDLC channel protocols over a shared OSA Express2 port in OSN mode on a System z9 processor) to z/TPF. z/OS, z/VSE, and z/VM VTAM all support OSA LSA and can optionally perform the boundary functions for the distributed communications server without requiring an NCP in between.

3.4 Modernizing an SNA APPN network infrastructure

In an APPN environment, an NCP is not as crucial an element as it is in an SNA subarea environment. However, an NCP may still be required in an APPN setup to provide SNA subarea connectivity functions (such as SNI connectivity), or together with VTAM to provide a Composite Network Node (CNN) function for the APPN environment.

In an APPN network, you will typically have both dependent and independent LUs that need to establish sessions.

Independent LUs can establish sessions between each other on peer nodes without having to traverse the network hierarchy to request assistance of a System Services Control Point (SSCP) in VTAM. Independent LUs can make decisions about the location of their session partners using the services of the local APPN nodes' Control Point (CP). They can also use the services of the APPN infrastructure, such as the directory services to locate partner LUs, and topology and routing services to calculate an optimal session route to the session partner.

3.4.1 Dependent LUs in APPN network

Dependent LUs on APPN nodes still require an SNA boundary function and the assistance of a controlling SSCP to locate session partners and establish a session. Therefore, the questions that arise for dependent LUs in an APPN network are:

- Where is the boundary function located?
- How to reach the controlling SSCP?

In an SNA subarea environment, SNA boundary functions must be located in subarea nodes that are connected through subarea links to the VTAM Systems Services Control Point (SSCP) that controls session setup between dependent LUs and session partners, typically mainframe SNA 3270 or server applications.

Note: In an APPN environment, SNA boundary functions may be located on any APPN node that supports the Dependent LU Requester (DLUR) functions.

The issue in APPN is that the APPN node that acts as the boundary function node typically is not attached through SNA subarea links to VTAM. Instead, it is attached through a single EE-link or through a series of other APPN nodes that are connected to each other over APPN links using either Intermediate Session Routing (ISR) or High Performance Routing (HPR) SNA routing protocols.

The dependent LU-to-SSCP communication required for setting up sessions between dependent LUs and session partners in APPN networks is sent over a pair of special-purpose (independent LU) sessions between the APPN node that acts as boundary function and the controlling SSCP. These sessions' end points are known as the Dependent LU Requester (DLUR) and the Dependent LU Server (DLUS).

DLUR is on the APPN node that acts as the boundary function node. DLUS is on the SSCP controlling node. DLUR functions may be implemented on APPN network nodes (NN), end nodes (EN), or a special type of NN called branch extender (BX).

Only VTAM in z/OS, z/VSE, and z/VM can act as a DLUS.

All the IBM distributed Communications Server solutions can act as DLUR nodes. Various other vendors' SNA technologies can also act as DLUR nodes in an APPN network, including Cisco SNA Switch routers.

The DLUR-DLUS session (and hence the controlling SSCP) is only used during dependent LU session setup. During session setup, another route through the APPN network may be chosen for the partner LU-to-dependent LU session. The dependent LU-to-session partner traffic may flow through a completely different route in the APPN network from the route that is used for the DLUR-DLUS session between the boundary function APPN node and the controlling SSCP. Refer to Figure 3-15.

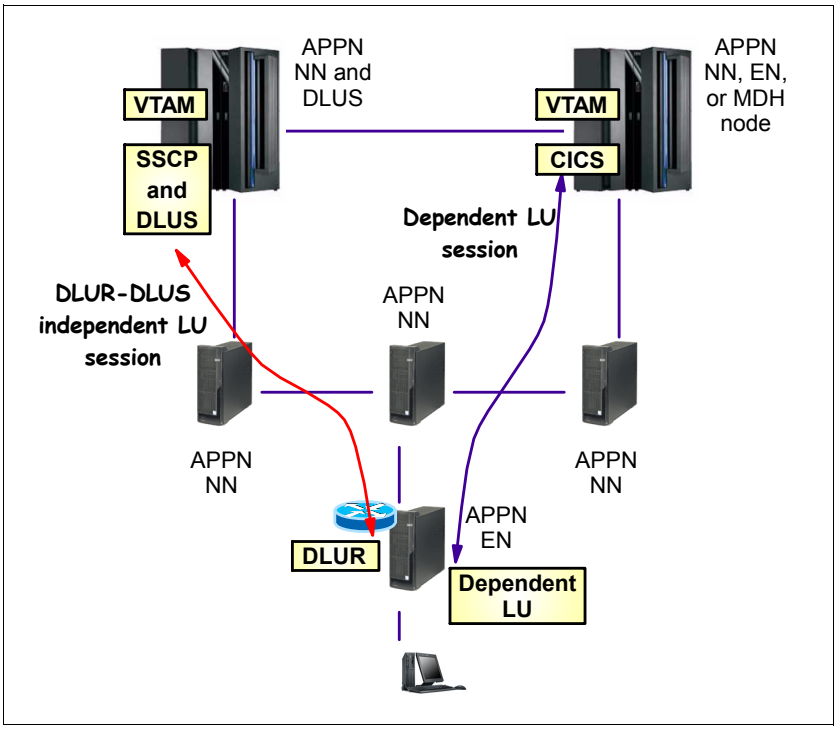


Figure 3-15 Dependent LU session setup in an APPN network

3.4.2 APPN with HPR routing

The links between the APPN nodes may either be Intermediate Session Routing (ISR) links or they may be High Performance Routing (HPR) links. As always for SNA sessions, when a session is set up, a route through the SNA network is chosen and the session is mapped to that route.

For both SNA subarea and for APPN/ISR links, the SNA sessions that are currently mapped to a given link will break if that link becomes unavailable. For HPR links, the HPR route end points will non-disruptively switch the affected sessions to another HPR route if such backup routes are available between the HPR-capable nodes.

Note: If an APPN node supports HPR routing, then use HPR between the APPN nodes. In other words, if you have enabled APPN on your SNA node and the APPN technology on that node supports HPR routing, then you can enable HPR routing for the links that HPR can be used on.

HPR may be used over various network technologies; the most common are:

- A local area network using SNA LLC2 protocols on the LAN

  For VTAM on z/OS, this means an OSA port operating in LSA mode to connect VTAM to the LAN for HPR routing (and subarea and ISR routing).

- A mainframe channel (either ESCON or FICON®) using the Multi Path Channel “Plus” (MPC+) channel protocols

  Note that there is also a version of MPC without the “plus”. MPC without the plus supports APPN/ISR routing, but not APPN/HPR routing. When CS Linux on System z connects to VTAM over a channel (real or emulated under z/VM), it uses MPC channel protocols, and hence supports APPN/ISR routing over that channel, but not APPN/HPR.

- A z/OS Sysplex XCF messaging group (XCF links between z/OS systems that are part of a single z/OS Sysplex)

- An IP network using UDP/IP protocols

  This is also known as HPR over IP, or Enterprise Extender (EE). Since EE uses IP networking protocols, EE is supported over all networking technologies over which IP is supported. This includes HiperSockets™ on System z.

You can even use HPR on a special type of combined subarea/APPN connections known as Virtual Route Transmission Group (VRTG), providing nondisruptive path switch for virtual route failures.

There are three levels of routing protocols for APPN nodes. (Note that not all existing APPN implementations support all of the following routing protocols):

- APPN/ISR routing

  All APPN nodes are supposed to support APPN/ISR routing over one or more physical link types.

- APPN/HPR routing
  - z/OS VTAM, z/TPF, CS Linux (System z, Intel, and Power), CS Windows, CS AIX, i5/OS, OS/400®, Microsoft Host Integration Server (HIS) 2004, PCOMM, and Cisco SNA Switch support APPN/HPR routing.
  - z/VM VTAM and z/VSE VTAM do *not* support HPR routing.
- APPN/HPR over IP routing (EE)
  - z/OS, CS Linux (System z, Intel, and Power), CS Windows, CS/AIX, i5/OS, Microsoft HIS 2004, PCOMM, and Cisco SNA Switch all support EE.
  - z/VM VTAM, z/VSE VTAM and z/TPF do *not* support EE.

Even though OS/2 and its Communications Server/2 component are both withdrawn from marketing, OS/2-based SNA solutions are still in use in certain geographies. CS/2 *does* support APPN/HPR over IP routing (EE).

There are other operating systems that support one or more of these APPN implementations. The ones listed in this section seem to be the most commonly used. Refer to “Appendix B: SNA Node Capability Summary” on page 100 for details on SNA capabilities of selected SNA nodes.

3.4.3 APPN with HPR routing over IP (EE)

Enterprise Extender or High Performance Routing (HPR) over IP is an APPN/HPR technology that uses APPN/HPR routing over an IP network.

Since EE is an HPR link type, it inherits all the characteristics of HPR including all the APPN base characteristics and the nondisruptive path switch mechanism of HPR. If an EE route fails and another HPR route exists between this HPR node and a partner node, any SNA sessions that were currently assigned to the EE route will be path-switched to the non-EE HPR route. It is quite common to use EE between z/OS systems in a sysplex, and to have backup HPR routes defined via XCF or MPC+ connectivity. If the EE connectivity fails, SNA sessions will continue uninterrupted over the backup HPR routes.

From an APPN topology perspective, EE looks upon the entire IP network as a single-hop HPR link. If an installation has enabled APPN with HPR routing, then transporting the SNA HPR data over an IP network is a simple matter of defining an HPR EE link and configuring the local TCP/IP environment to support the five UDP port numbers that are used by EE (12000 to 12004) which include one UDP port for each SNA Class of Service (COS).

By using separate port numbers per SNA class of service, it is possible to maintain the SNA prioritization in the IP network by assigning Differentiated Services (DS) settings (or Type of Service or TOS, as it was earlier known) per UDP port that match the relative priority of the SNA class of service definitions, thus providing the IP network routers with IP-packet priority information based on the original SNA network priorities.

EE uses the UDP transport layer in TCP/IP. From a TCP/IP perspective, EE is just another UDP application. UDP is a connectionless, non-reliable, best-effort transport protocol. It appears an odd match for the reliability we normally associate with SNA networks, however EE is an extension to HPR and HPR implements reliability itself by means of the Rapid Transport Protocol (RTP) layer in the SNA protocol stack.

If EE had used TCP, there would have been two layers on top of each other that, from a transport protocol perspective, would monitor packet loss or delays and initiate retransmissions and recovery functions. Because of different algorithms and protocol details, however, this might have resulted in conflicts and would have been a waste of CPU resources. By having EE use UDP, RTP is the single entity that is responsible for flow control, retransmission, and recovery at the HPR transport protocol layer.

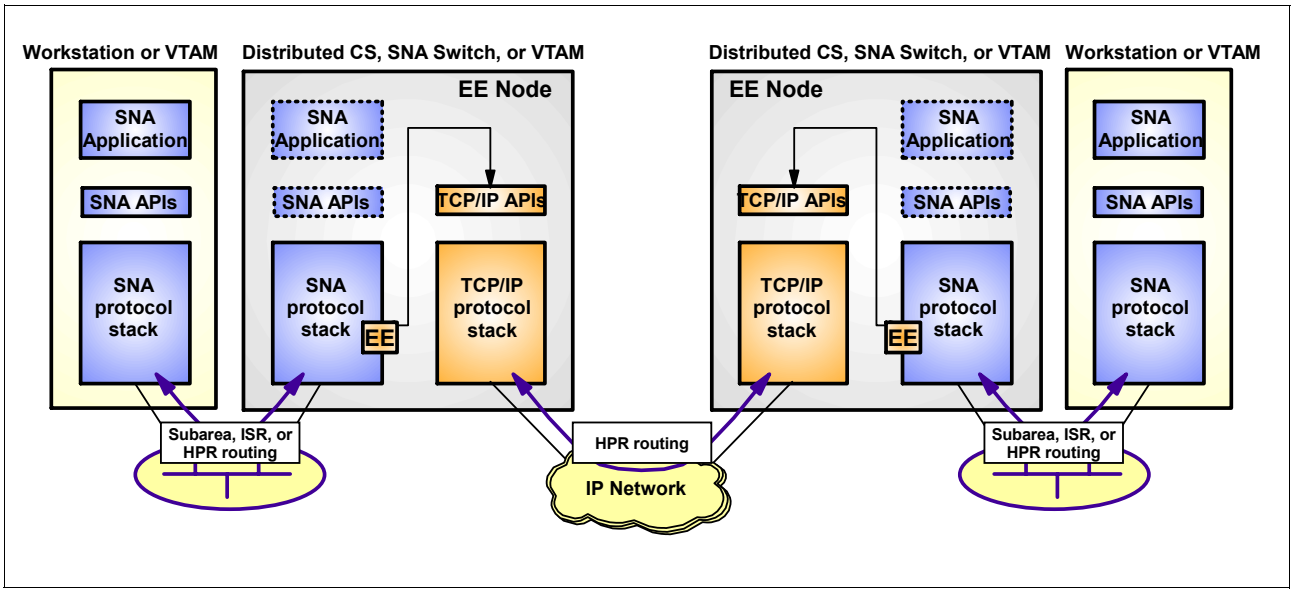


Figure 3-16 Enterprise Extender sample topology

While use of UDP by EE is the most efficient solution, it has also resulted in IP network infrastructure issues wherein it is not uncommon that IP firewalls implement restrictions on

use of UDP-based communication through the firewalls. This has especially been an issue when EE is used between business partners, and each business partner maintains firewalls between its respective intranets and some form of a public or shared network domain.

The best way of addressing the concerns of the firewall administrators is to use IP Security (IPSec), also known as Virtual Private Network (VPN) technology, preferably between the two EE end points; otherwise, between the firewalls. Each firewall will, in such a setup, be configured to only permit UDP packets to or from one of the five UDP port numbers and only over a secured channel. Whether the secure channel only authenticates the data (using the IPSec Authentication Header (AH) protocol), or also encrypts it (using the IPSec Encapsulated Security Payload (ESP) protocol) is up to the network and security administrators to determine. Use of IPSec between the EE end points or the firewalls is transparent to EE and to the SNA APPN network.

Enabling APPN modernizes the overall SNA environment itself by virtue of the dynamic nature of APPN and the reduction in manual definitions that are needed in an APPN network as compared to an SNA subarea network.

However, just enabling APPN does not in itself address the objectives of sharing an IP network infrastructure for both SNA and IP applications. To address this objective, HPR over IP (EE) must be used for APPN routing.

Note: Enabling EE offers the potential for true end-to-end SNA over IP transport between the branch and the data center, and between the data center and business partners.

There are many possible topologies in terms of where the EE connectivity ends and normal HPR or ISR routing (or—in the case of an Interchange Node (ICN)—subarea routing) takes over. An Interchange node routes between SNA subarea and SNA APPN networks. Refer to “Second SNA level: APPN” on page 98 for more information about Interchange nodes. The general approach, from a network infrastructure simplification perspective, is to push the EE end points as far out as possible towards the user workstations, and as far into the System z platform as the mainframe software allows.

There are potentially two issues with this overall approach:

- Not all mainframe operating system environments support EE all the way into the mainframe operating system (z/VSE, z/VM, and z/TPF are clear examples of this).
- By pushing the EE end point all the way out to the workstation, you may end up with a complex and large APPN network topology that may impact overall APPN network performance and management.

We briefly discuss how to address both these issues in the following sections.

3.4.4 Data center EE gateway to z/VSE and z/VM

For mainframe operating system environments that do not support EE directly, an EE gateway solution can be implemented in the data center. Such an EE gateway may act as an APPN intermediate node that uses HPR over IP routing downstream into the network, and APPN/ISR routing upstream to the mainframe operating system.

The EE gateway can be located on System z servers, in either z/OS or in CS Linux on System z. The EE gateway can also be located in a non-z server in the data center, such as an Intel or Power server where the IBM distributed Communications Server is deployed. The non-z server needs to be LAN-attached to the System z server for APPN/ISR routing to the System z operating systems, which must all be APPN network nodes.

If z/OS acts as the EE gateway, then z/VSE and z/VM may be in a different SNA NETID from z/OS itself. z/OS can act as an APPN EBN and manage the APPN network-to-network communication between the APPN NETIDs. If the EE gateway is one of the distributed IBM Communications Servers, including CS Linux on System z, then the EE gateway must be in the same APPN NETID as z/VSE and z/VM (CS Linux cannot act as an APPN EBN).

Also, the z/VSE and z/VM systems need to be configured as APPN Network Nodes if the gateway is provided by a non-z/OS system. We recommend that you not have a z/VSE or z/VM APPN End Node receive network node services (NNS) from a non-VTAM system; refer to Figure 3-17.

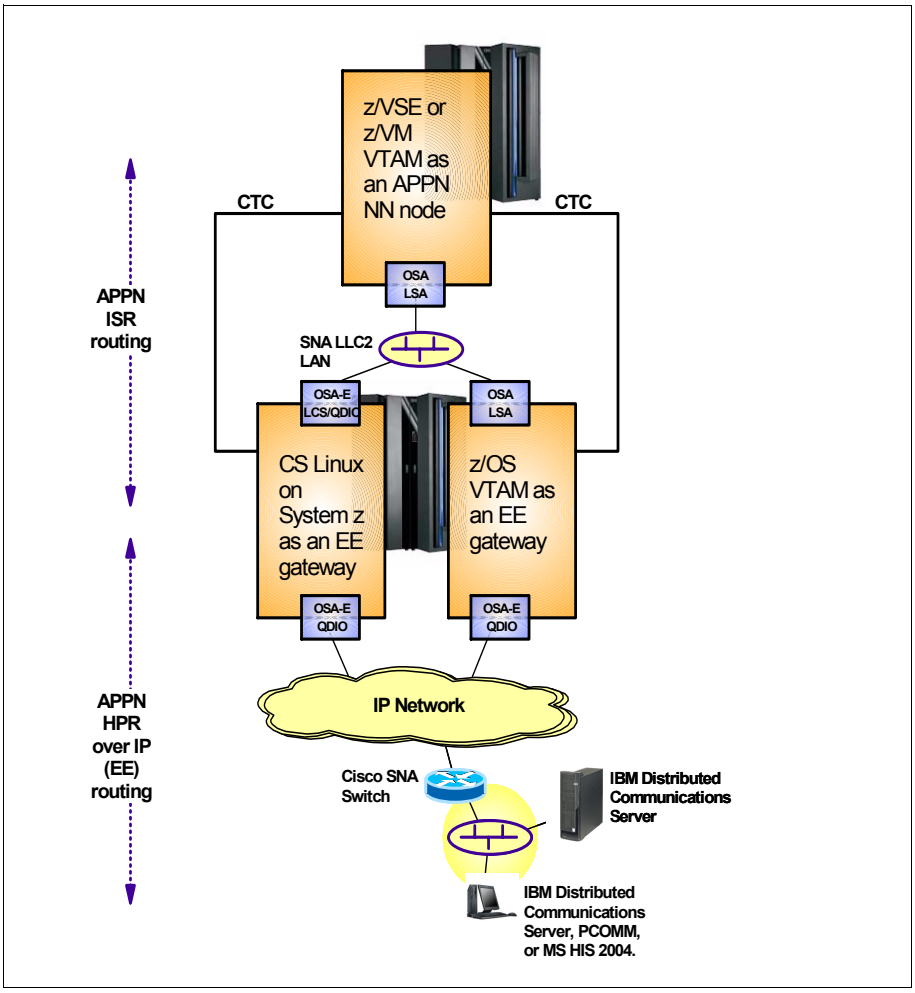


Figure 3-17 EE gateway in the data center to z/VSE and z/VM VTAM

There may even be scenarios where you might want to use a distributed Communications Server as an EE gateway to z/OS for the purpose of offloading some of the processing when a high number of APPN partner nodes are connected through an EE network.

3.4.5 Branch extender node topology

To address the potential for rather large APPN topologies when pushing the EE end points all the way out onto the workstations, the special APPN node type that is known as a Branch Extender (BX or BrNN) node can help address this issue; refer to Figure 3-18.

The more Network Nodes there are in an APPN network, then the greater potential there is for increased APPN CP-to-CP exchange of control information such as partner LU search requests, APPN topology updates, and so on, which can consume considerable amounts of network bandwidth in very large and complex networks.

An APPN node that acts as a BX node presents an APPN Network Node view to downstream APPN End Nodes so that these downstream APPN End Nodes can register their LUs with their APPN Network Node Server (NNS). At the same time, the BX node presents a view upstream of an APPN End Node that registers its LUs (including those that were registered by downstream APPN End Nodes) with an upstream APPN Network Node.

If the connectivity between the BX node and the data center is based on EE, then the APPN Network Nodes can be collapsed into the data center. This eliminates APPN Network Nodes in remote locations. By doing so, it reduces the network overhead (especially WAN overhead and delays) for exchange of APPN control information between APPN Network Nodes.

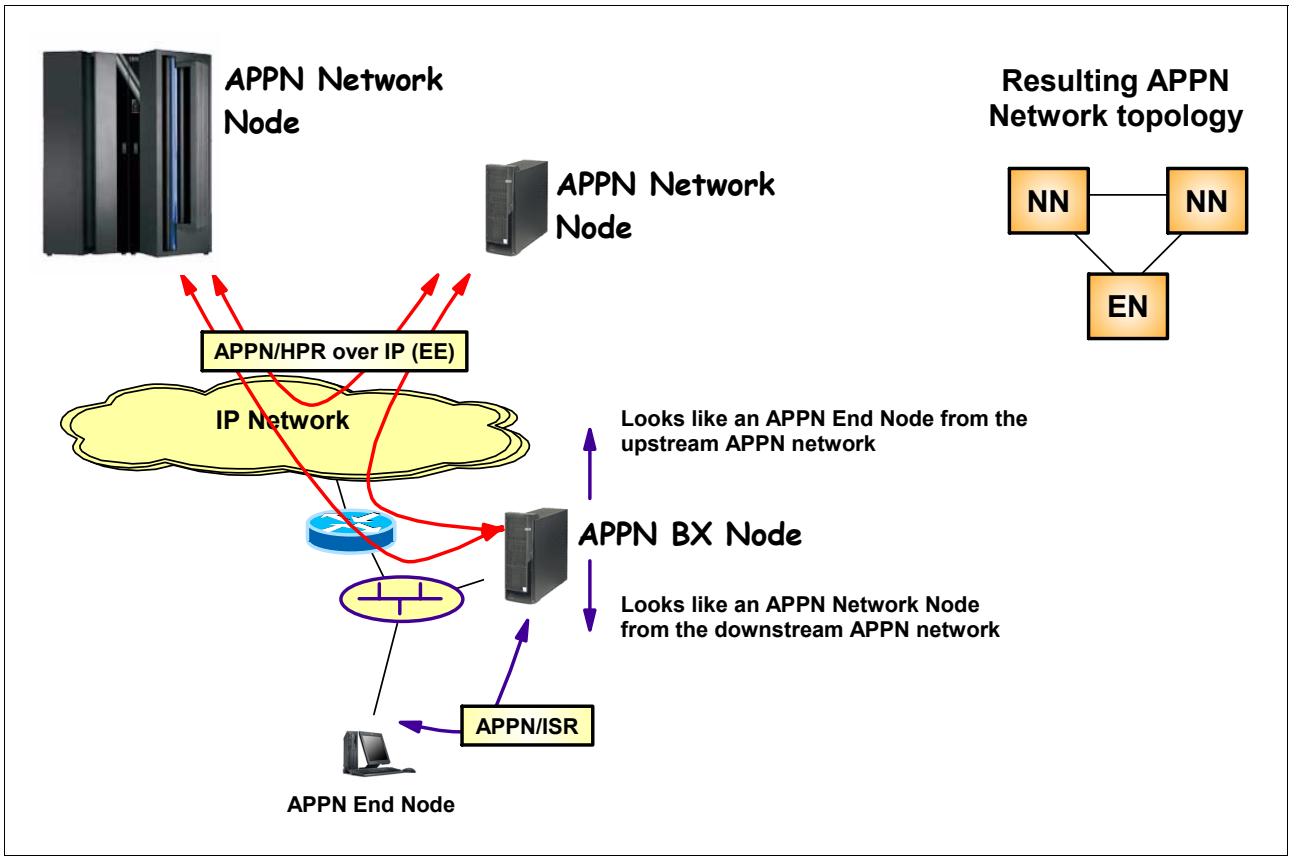


Figure 3-18 Simplifying APPN topology with Branch Extender node

IBM's distributed Communications Servers can all be configured as BX nodes.

Note: A DLUR node cannot be located downstream from a BX node.

3.4.6 Connection networks and EE

One additional technology simplifies the APPN network infrastructure and reduces the amount of manual definitions needed in that infrastructure. This technology is known as

connection network, implemented through the concept of a node type known as a Virtual Routing Node (VRN).

A connection network reduces the need for predefining APPN links between nodes that are connected to a shared access transport facility (SATF), of which a LAN is the most typical example.

Consider the following example: you have five APPN nodes that are connected to a LAN. To communicate directly between these five nodes, each of them would normally have to define links to the other four (for a total of 20 link definitions). Now consider a LAN with 100 APPN nodes that you want to be able to communicate directly between over that LAN.

In a connection network, you define a single virtual routing node that in a sense represents the LAN, and you then predefine a link from each of the nodes that are attached to the LAN to that virtual routing node. The APPN network infrastructure then dynamically determines how to communicate directly with each of the other APPN nodes on that shared access transport facility (the LAN) without you having to define links for each partner node.

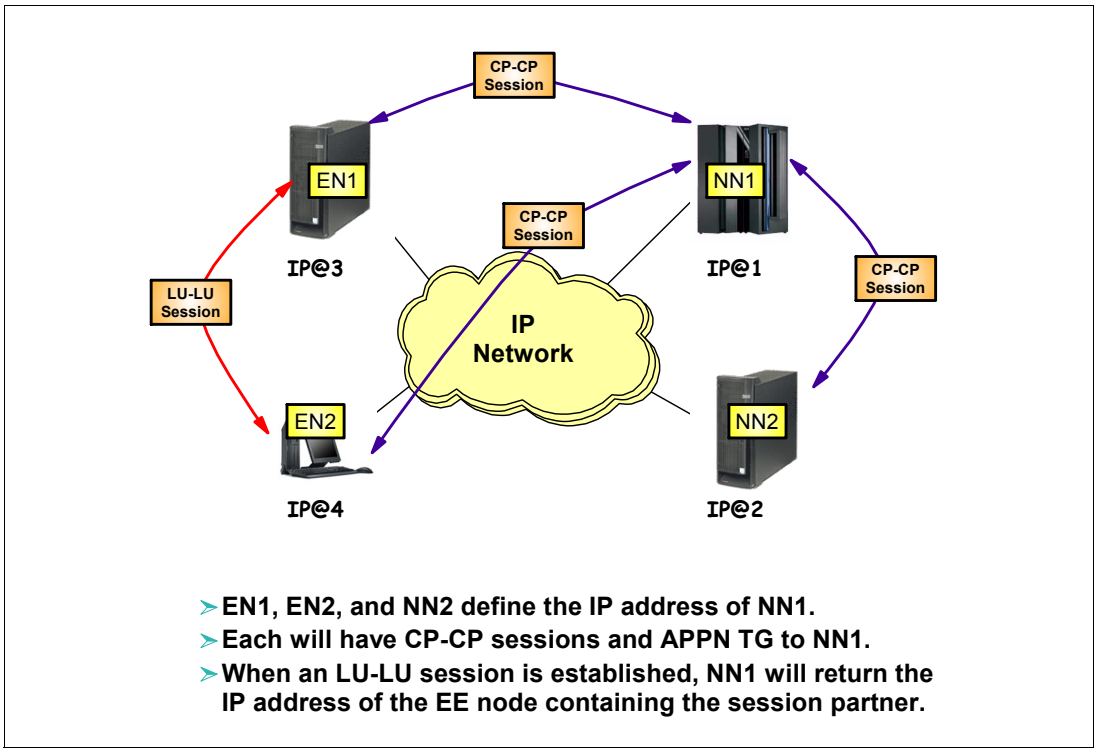


Figure 3-19 Connection network using EE

Consider EE in this context and consider the IP network—no matter how large and complex in terms of IP routers—as a single shared access transport facility. Every IP node that is connected to the IP network can communicate with every other IP node that is attached to that IP network.

From an EE perspective, the IP network can be seen as a connection network that is represented by a single virtual routing node. This enables all EE end points to communicate directly through IP end-to-end over that IP network without predefining links to all potential EE endpoints on the IP network.

Therefore, from a definition point of view, each EE node defines an APPN link to the virtual routing node that represents the IP network, and the EE node then dynamically learns how to reach all other EE endpoints that have links defined to that same virtual routing node.

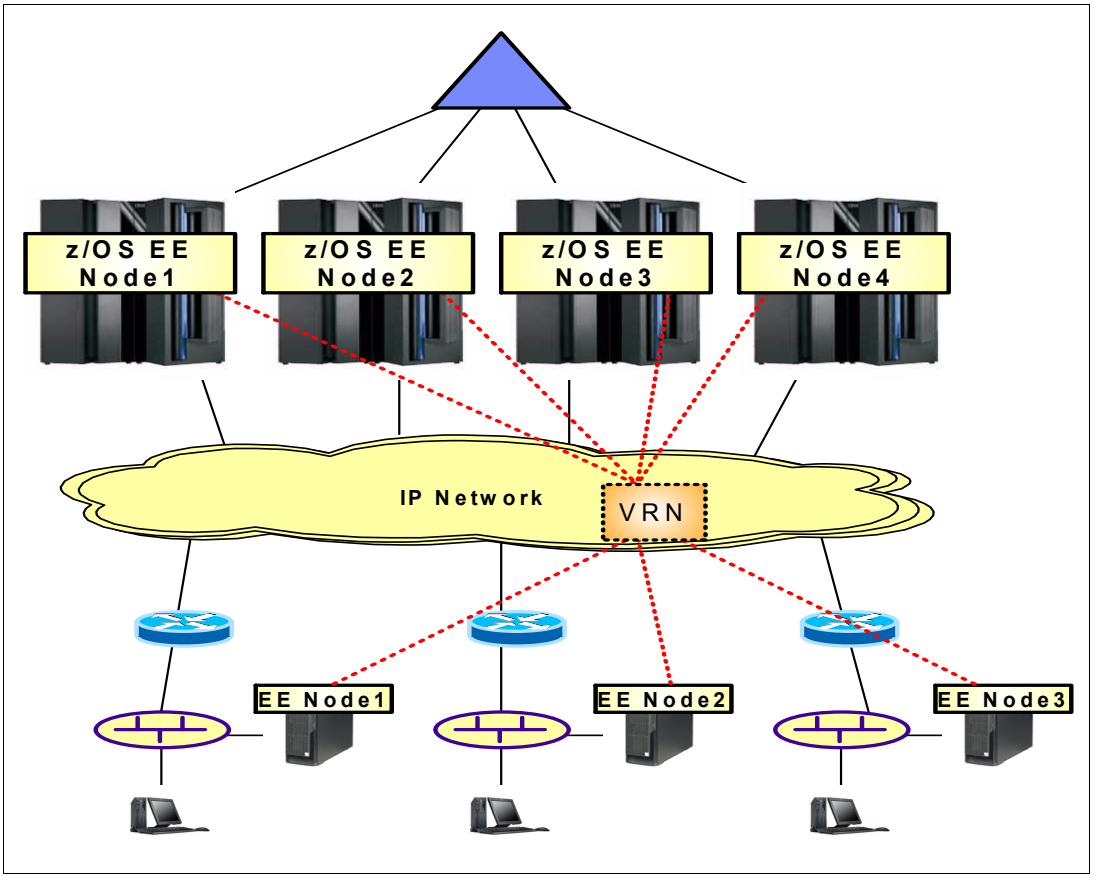


Figure 3-20 EE and connection networks

In the previous illustration, all EE nodes can send EE (UDP/IP) packets directly to each other without defining links to all the other nodes. Note that in a connection network topology, each node still needs some predefined links for selected CP-CP sessions.

Generally, the combination of EE with connection network technology is beneficial in reducing the amount of link definitions that are required and to allow EE end point-to-end point communication to flow directly between the associated IP end points.

However, there are scenarios which require more control over which links exist and are allowed to be used for HPR routing. In the previous illustration, EE Node1 can route HPR packets directly to any of the four z/OS EE nodes. However, what if management functions on z/OS EE Node1 need to have knowledge about the traffic that flows between EE Node1 and any of the z/OS EE nodes? If EE Node1 communicates directly with z/OS EE Node4, then z/OS EE Node1 will have no knowledge of that communication.

By not using connection networks and instead predefining links between the remote EE nodes and the data center EE nodes, traditional CMC-based management functions can be preserved and used with EE traffic flowing in and out of the z/OS sysplex, as long as one accepts the overhead of routing HPR through such CMC z/OS nodes. If z/OS EE Node1 is the CMC host, then EE Node1, when communicating with z/OS EE Node4, sends HPR data through z/OS EE Node1 and vice versa.

For availability purposes, each remote EE node should be preconfigured with at least two links, one to each of the z/OS CMC hosts that act as backup for each other.

3.4.7 SNA application access modernization considerations

EE is the obvious technology in an APPN environment to modernize the transport of SNA data over an IP-based network infrastructure. EE allows you to continue using SNA protocol stacks in remote locations on workstations, on servers in the branch, or both. However, it allows you to remove the SNA-based WAN technologies at the same time.

EE in itself does not address the objective of consolidating SNA protocol stacks into the data center. An EE node is a full SNA protocol stack node.

To address the SNA protocol stack consolidation objective, consider technologies similar to the ones discussed in 3.3.7, “SNA application access modernization considerations” on page 41 for the APPN case.

The only addition to what we discussed earlier would be the SNA-based connectivity between the node that implements the various SNA application access modernization solutions and the operating systems where the SNA applications are located.

For the distributed IBM Communications Servers, such connectivity includes use of EE to z/OS. If CS Linux for System z is located on the same System z server as z/OS, then that EE connectivity can be via IP over HiperSockets.

If the mainframe operating system is z/VSE or z/VM, APPN/ISR routing over a shared LAN or over an MPC channel (real or virtual under z/VM) can be used to connect CS Linux on System z to VTAM on z/VSE or z/VM.

3.5 Modernizing SNA business partner communication

Business partner SNA-based communication can be done using either SNA subarea technologies or SNA APPN technologies.

Keep in mind that business partners that are directly adjacent must agree on what technology to base the communication on. If business partners are not directly adjacent, but rather communicate through a third party, they may use different SNA business partner connectivity technologies.

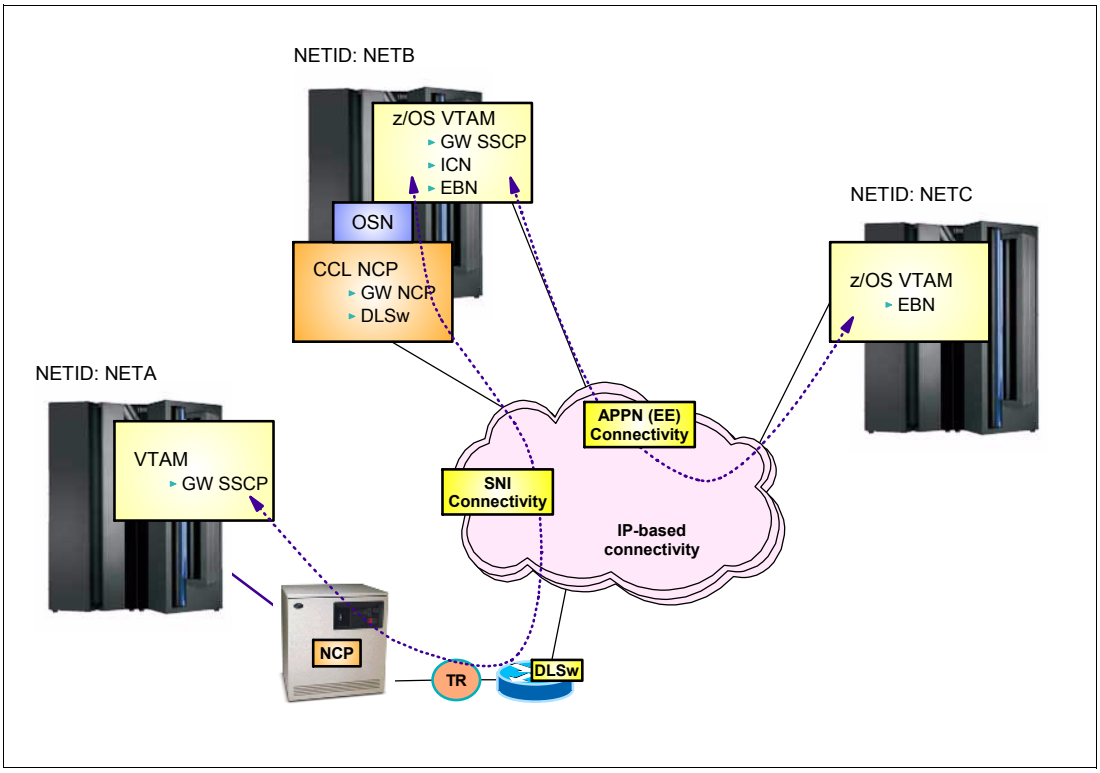


Figure 3-21 SNI and APPN EBN connectivity

In the topology shown in Figure 3-21, NETA can communicate with NETB using SNI functions. NETB can communicate with NETC using APPN/EBN functions over EE. NETA cannot communicate directly with NETC since they use different technologies, but NETA can communicate with NETC by going through NETB where VTAM in NETB performs both GW SSCP functions, Inter Change Node (ICN) functions, and Extended Border Node (EBN) functions. The connectivity between NETB and NETC in this example is based on EE, but it could be any APPN link technology.

If both business partners are APPN-enabled, support EE, and are able to establish IP-based connectivity between the two sites, then APPN/EBN functions over an IP network using EE can be used to interconnect the two business partners.

Note: APPN/EBN over EE is the preferred business partner connectivity technology. It is based on the latest SNA architecture level, and it offers IP-based connectivity from z/OS VTAM to z/OS VTAM.

If directly adjacent business partners require SNI connectivity, then you must continue to support SNI in some form. You can still use APPN/EBN connectivity over EE to other business partners, as shown in Figure 3-21. VTAM can easily be both a gateway SSCP for SNI business partners and an APPN/EBN Node for APPN network-to-network connectivity to other business partners.

To address our SNA modernization objectives for a continued SNI-based workload, the preferred solution is to deploy a gateway NCP in CCL and use one of CCL's SNA over IP technologies for connectivity to the business partner site.

If the business partner continues to use an NCP in an IBM 3745/46, then you can use DLSw between the CCL NCP and a DLSw router in the business partner site. If the business partner

also implements CCL, then the connectivity can be a direct TCP connection over an IP network using the CCL IP Transmission Group technology.

CCL does not impose any limitations on the SNI topology in terms of single gateway, back-to-back gateway, and so on.

Although other platforms support both APPN and EE, only z/OS VTAM supports being both an Extended Border Node and use EE.

To control the APPN network-to-network connectivity, a z/OS VTAM in each network is, in most topologies, configured to perform the EBN functions between the two APPN networks.

If both APPN networks have deployed EE, and EE in both networks use a common global virtual routing node (a global connection network), then an EE node in one network can establish direct EE connectivity to most EE nodes in the other network. All session setup requests will flow through the EBN nodes.

However, the actual session may use a more direct EE connection between the two session partner nodes. This is highly desirable from a performance perspective, but may not always be desirable from a management perspective.

If a global virtual routing node is not used, then links between the networks will only exist between the EBN nodes and must be manually defined. All cross-network traffic can in that case be forced through the EBN nodes to ensure that management functions can be applied to all traffic.

Also, from a management objective, controls are often needed on the EBN nodes over which session partners are allowed to establish sessions and to collect session-related accounting data. An EBN node may not have session awareness if it acts as an HPR Automatic Network Routing (ANR) node.

An ANR node is an intermediate node between the two nodes that implement the Rapid Transport Protocol (RTP) functions, the HPR pipe end points. To ensure session awareness on EBN nodes, you can force HPR pipes to terminate on the EBN nodes through use of the RTPONLY option on the adjacent CP definitions. You can have one pipe come into the EBN node, and then continue the HPR routing onto another pipe going out of the EBN Node.

Use of this option also effectively blocks EE nodes in two different networks from connecting directly through a global virtual routing node. All HPR traffic, both session setup and session data, is forced through the EBN nodes. Note that a consequence of such a configuration choice is that the EBN which terminates these HPR pipes becomes a single point of failure.

4

How to the modernize SNA application access

This section discusses how to approach an SNA modernization project if the primary purpose is to modernize the SNA application access.

4.1 How to modernize SNA 3270 application access

SNA 3270 applications is the largest portion of SNA applications. It is, also, the type of SNA applications that is the easiest to address from an overall modernization perspective.

For SNA 3270 access, you need to distinguish between two main groups of SNA 3270 clients:

- An IBM 3270 emulator that a software package on a programmable workstation implements.
- The real IBM 3270 device that is coax-attached to a cluster control unit that is a channel, LAN, or serial line attached into the SNA networking infrastructure.

4.1.1 SNA 3270 emulator as the IBM 3270 client

Most of the currently available and widely used SNA 3270 emulator software products support two main flavors of operation:

- SNA - The original SNA-based functions, where the emulator uses the original SNA-dependent LU programming interfaces. The actual network connectivity from the workstation may use one of three technologies:
  - SNA LLC - The workstation implements a full SNA protocol stack that sends and receives SNA link-level traffic, which on a LAN generally is referred to as SNA LLC2 (Logical Link Control type 2) flows as opposed to other network protocols, such as IP flows. The workstation in SNA terms is an SNA PU with a dependent LU per emulator session. The PU may be a peripheral node or an APPN node, depending on the upstream SNA infrastructure.
  - EE - The workstation implements a full SNA protocol stack that is EE enabled, allowing it to send and receive SNA data over an HPR pipe that uses IP packets in or out of the workstation. Also, in this case, the workstation is an SNA PU with a dependent LU per emulator session. Because EE is APPN, the workstation is an APPN Node that uses DLUR protocols to a DLUS server to establish SNA sessions for its dependent LUs.
  - Remote API - The workstation does not implement a full SNA protocol stack, but only the IBM remote SNA client API layer that intercepts the SNA API calls and ships them over a TCP connection to an IBM SNA API server using IP packets in or out of the workstation.
- TN3270 - An IP-based network connectivity where the emulator software uses TCP/IP programming interfaces to communicate, using the TN3270 protocol over a TCP connection that is over an IP network to a TN3270 server.

The actual SNA 3270 emulator software product you use may impose limitations on the previously mentioned technologies that you are able to use, but the end-user interface, the “green screen”, is unchanged. There may be small variations in what is displayed in the 3270 Operator Information Area (OIA) down at the bottom of the emulated 3270 screen, but the user dialog as such is unchanged.

If one of the SNA-based connectivity technologies is used, some SNA 3270 emulator products may depend on other workstation products to implement an SNA protocol stack, while others, such as IBM PCOMM, come with an SNA protocol stack built in. If SNA 3270 emulation was the only reason for such SNA protocol stack software on the workstation, a switch to TN3270 connectivity may remove the requirement for that additional workstation software product.

Use of the TN3270 technology in the SNA 3270 emulator requires a TCP/IP protocol stack and not an SNA protocol stack. TCP/IP is readily available in all the environments of today where you might need to use a TN3270 emulator.

TN3270 introduction

Telnet 3270 (TN3270) is a standard application protocol defined by the Internet Engineering Task Force (IETF) in Request for Comment (RFC) 2355, plus various additions and draft RFCs.

The TN3270 protocol allows an IBM 3270 software emulator to connect over an IP network using a TCP connection to a TN3270 server. The TN3270 server emulates secondary SNA logical units (LU type 1, 2, or 3) on behalf of the IBM 3270 emulator and establishes SNA sessions to the requested SNA server applications (SNA primary LUs), such as TSO, IMS, CICS, and so on. See Figure 4-1.

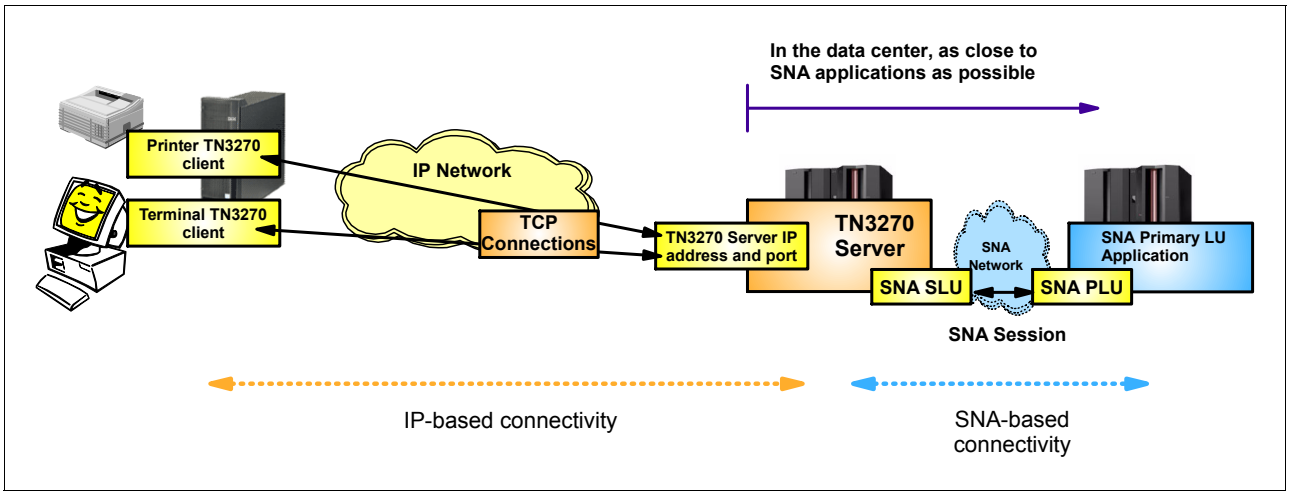


Figure 4-1 TN3270 component overview

Between the TN3270 emulator and the TN3270 server is a TCP connection over an IP network. Between the TN3270 server and the SNA application is an SNA session over an SNA network. There is no SNA protocol stack on the node where the TN3270 emulator runs. This node only needs an IP protocol stack.

The TN3270 protocol supports both traditional IBM 3270 display terminals, such as IBM 3278 and IBM 3279 (LU Type 2), and printer devices, such as IBM 3286 or IBM 3287 (LU Type 1 (SCS data streams) or LU Type 3 (3270 data streams)). The protocol does not impose limitations on the 3270 data stream. Both the basic IBM 3270 protocol functions and advanced functions, such as Graphical Data Display Manager (GDDM®) graphics are fully supported by the TN3270 protocol. But the advanced IBM 3270 data stream functions may not be supported by all TN3270 emulators. The SNA 3270 data stream is largely transparent to the TN3270 protocol.

Between the TN3270 server and the SNA applications is an SNA network. This SNA network may be a traditional SNA network infrastructure with an NCP and VTAM, or it may be an APPN infrastructure including APPN with HPR over IP. In the second case, the physical network between the TN3270 server and the SNA applications may be another IP network.

A general objective when deciding where to place the TN3270 server is to make the “distance” between the TN3270 server and the SNA applications as short as possible. This means that a location in the data center is desirable, allowing the connectivity between the remote locations where the users are located and the data center to be based entirely on an

IP network infrastructure and, from the perspective of SNA 3270 access, collapse the SNA network into the data center.

Some of the potential platforms to implement the TN3270 server in the data center are:

- IBM Communications Server for z/OS
- IBM TCP/IP for z/VM
- IBM TCP/IP for z/VSE
- IBM Communications Server for Linux on Intel, Power, or IBM System z
- IBM Communications Server for Windows on Intel
- IBM Communications Server for AIX

Other vendors provide various products that also implement TN3270 server technology on a range of hardware and operating system platforms

Placing the TN3270 server on the mainframe itself has advantages from an SNA network simplification point of view. You can reduce SNA connectivity to consist of only connectivity within mainframes and between mainframes in the data center.

Placing the TN3270 server on z/OS offers additional advantages:

- In terms of reliability based on the high availability capabilities of the z/OS Sysplex TCP/IP infrastructure
- In terms of scalability both within single LPARs and across LPARs
- Management in terms of built-in response time monitoring and accounting
- In terms of overall performance

IBM has successfully tested z/OS TN3270 server configurations with up to 200,000 concurrent TN3270 connections to a single TN3270 server instance.

Most of the current TN3270 client and server software supports the process of securing the individual TN3270 connections using Secure Sockets Layer (SSL) protocols. This is of importance for SNA 3270 dialogs where mainframe passwords are exchanged whenever an end user logs on to TSO, CICS, IMS, and so on. IP-based “sniffer” technologies are more frequently used than what used to be referred to as Network Protocol Analyzers for SNA traffic.

Where SNA applications or subsystems, such as CICS or IMS, require some form of terminal-related security functions based on the LU name of the SNA 3270 device (the TN3270 client in this context), both the TN3270 protocol itself and TN3270 servers offer a variety of mechanisms for assigning predictable SNA LU names to individual TN3270 connections. This allows installations to continue to use their existing terminal-based security functions. Such LU name assignment may be based on the TN3270 client IP address or host name. On z/OS, it may be additionally based on the user ID of the end user using the TN3270 client. To base LU name assignment on a user ID, you require secure connections with client authentication.

Note: In most cases, it is a simple transition to move IBM 3270 emulation from native SNA connectivity to TN3270 using IP connectivity. The end-user interface does not change, the emulator continues to present a traditional IBM 3270 screen interface to the end user. The mainframe SNA applications continue to see an SNA LU type 1, 2, or 3 connecting over an SNA session. The TN3270 infrastructure, including both the IP network and the TN3270 server, is transparent to both the end user and the SNA applications.

However, if workstation or branch server applications use SNA 3270 emulator APIs, such as the EHLLAPI programming interface, to “screen scrape” data off a 3270 dialog, then the

switch to TN3270 may not always be transparent. Most TN3270 emulators do support the EHLLAPI interface, but the status information in the traditional SNA 3270 status line (the Operator Information Area, OIA) varies slightly between an SNA 3270 and a TN3270 dialog.

Some EHLLAPI applications are known to have had problems when switching to TN3270. For those, one may have to continue to use an SNA 3270 emulator but combine that emulator with other SNA/IP integration or network infrastructure modernization technologies, such as IBM remote SNA API client/server, HPR over IP, or DLSw.

IBM PCOMM and HOD emulators both support a more modern programming interface known as Host Access Class Library (HACL), which allows object-oriented programming environments, such as Java applications or applets, to interface through programs with the 3270 data stream as it appears over the emulator session. Compared to EHLLAPI, HACL provides a more high-level and much easier-to-use programming interface.

Note: Given that the majority of SNA network traffic is IBM 3270 traffic, replacing SNA-based 3270 access with an IP-based TN3270 access, and placing the TN3270 server in the data center, can greatly reduce the amount of SNA-based wide area network traffic that still exists in an enterprise.

TN3270-based access to SNA 3270 applications

There are two general models for how to provide TN3270-based access to SNA 3270 applications. See Figure 4-2.

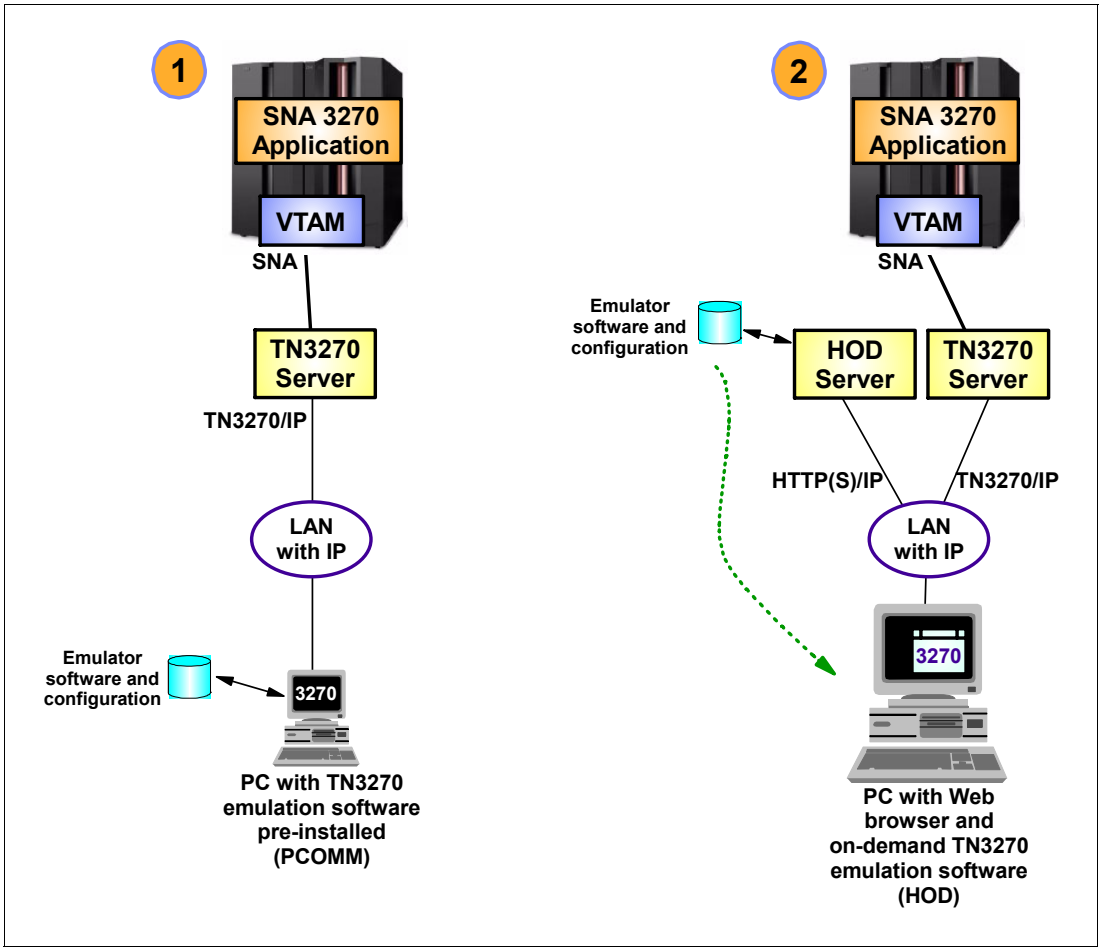


Figure 4-2 SNA 3270 access over IP - two general models

In this context, we primarily focus on providing traditional 3270 screen access to SNA 3270 applications. The following sections discuss how to transform the user experience through further integration with Web browser technology.

- Permanently installed TN3270 client on user workstations (fat client)

  A TN3270 client software package is permanently installed on the user workstation and the TN3270 client configuration is local on the user's workstation. TN3270 connections are established from the TN3270 client to a TN3270 server. The IBM PCOMM product provides such an option, as do numerous other TN3270 client packages from many vendors.

- On Demand, downloadable TN3270 client integrated into browser environment

  A TN3270 client can be downloaded dynamically On Demand by a browser when needed. The IBM Host On Demand (HOD) product provides such an option. An administrator can maintain the TN3270 emulator software itself and the configuration for individual users on a central HOD server. HOD is a Java application that you can download as a Java applet when required or cached locally by the Web browser and used to establish TN3270 connections between the workstation and the TN3270 server.

You typically will see an evolution occur within an enterprise. Because many enterprises come from an environment where users already use SNA 3270 emulators, a switch to use TN3270 in the already installed emulator software is a relatively small change. For PCOMM and most other SNA 3270 emulator products, an emulator switch between SNA 3270 and TN3270 is a simple setup option per emulator session.

In the long run, you may want to explore the advantages of not having to install, maintain, and manage the 3270 emulation software on each user's workstation, but instead centralize the software and the administration using an HOD model.

If your business requirement is to enhance the user's experience when working with traditional SNA 3270 applications, an SNA 3270/HTML transformation technology offers transformation of the SNA 3270 dialog to make it more consistent with the user interface that is used by more recently developed Web-based applications. In addition to enhancing the user experience, such technologies also offer relief where firewall traversal for TN3270 traffic between the end-user workstation and the server node is an issue, and it often is if public networks are in the path between the end user and the TN3270 server. See 4.3.1, “Presentation integration - SNA 3270/HTML transformation” on page 73 for more details.

TN3270 client options - installed client

TN3270 client software can be purchased from a range of vendors, installed, and configured locally on each user's workstation.

The IBM PCOMM product provides such an option. PCOMM supports the TN3270E protocol levels (Enhanced TN3270), plus various additional functions that have been added to the TN3270E protocols over the last few years, such as contention resolution.

Attention: There are freeware versions of TN3270 emulators available and while some of these in some environments may provide sufficient functions, you should be careful. Some of the freeware technologies have been known to use somewhat “loose interpretations” of the TN3270 protocol and should only be used after you have conducted a thorough test to prove the solutions work according to your requirements in your environment.

Installed TN3270 clients provide a rich set of TN3270 functions and generally provide good performance. But the cost of maintaining copies of the TN3270 client software on multiple workstations or LAN servers combined with the administration overhead associated with

maintaining the local emulator session configuration files on each user's workstation tends to become prohibitive in large organizations.

TN3270 client options - presentation using a Web browser

To address some of the issues with the installed TN3270 clients, you can integrate the SNA 3270 emulator presentation into a Web browser interface where the emulator software can be downloaded dynamically and emulator session configuration information for each user can be maintained centrally.

HOD supports the TN3270E protocol levels (Enhanced TN3270), plus various additional functions that have been added to the TN3270E protocols over the last few years, such as contention resolution.

With IBM Host On-Demand (HOD), a TN3270 emulator client can be dynamically downloaded to the workstation as a Java applet within a browser session. HOD will present a traditional SNA 3270 “green screen” interface, looking very much like a traditional TN3270 client, such as PCOMM. Administration of HOD usage is centralized on a HOD server. Host On-Demand is a full-function TN3270 emulator that supports display sessions, printer sessions, security, macros, keyboard remap facility, host file transfer, FTP, and so on. HOD, in addition to supporting the TN3270 protocol, also supports the TN5250 protocol.

HOD can be downloaded On Demand or cached in your browser. When using a cached copy, the configuration is still accessed from the central HOD server and code updates are supported under administrator control. By storing a copy of the HOD Java code locally, one avoids the network overhead of downloading On Demand, which potentially can be an issue on locations that are connected using slower-speed transmission facilities.

When using HOD, the TN3270 emulator runs on the workstation and establishes a TN3270 TCP connection over the IP network from the workstation to the TN3270 server location. If HOD is to be used in a topology with firewalls between the user workstations and the TN3270 server location, one needs to be aware that the firewalls have to permit TN3270 connectivity from the user workstations to the TN3270 server location in terms of TN3270 server IP address and port. HTTP(S) is only used for contacting the HOD server and downloading the HOD Java applet and the configuration, not for the actual SNA 3270 session.

The main benefits of a solution such as HOD are:

- It avoids the issues with software maintenance of permanently installed TN3270 emulators.
- It provides a centralized configuration repository and administrative interface.
- It integrates with the user's Web browser.

In general, because these types of technologies implement a TN3270 emulator on the workstation, the same SNA 3270 functions as were supported by a permanently installed TN3270 client are supported. There may be minor deviations in support based on the actual product that one uses. From an IBM perspective, IBM's PCOMM and HOD provide comparable TN3270 emulator functions.

Considerations for placing the TN3270 server

If you have not already set up a TN3270 solution, you should consider doing so for the following reasons:

- Switching from SNA to TN3270 will, in most cases, remove a considerable portion of your SNA wide area network traffic and replace it with TCP/IP-based network traffic from the end-user workstation and to the TN3270 server. It will also remove the need for

maintaining and modernizing an SNA network infrastructure between the end-user locations and the TN3270 server location.

- TN3270 is a well-proven and widespread technology that today is in use by a large number of enterprises and users.
- Switching from SNA to TN3270 connectivity preserves the end-user interface. It is transparent to the end user and it is transparent to the 3270-based mainframe subsystems and applications, for example TSO, CICS, IMS, and so on.
- A TN3270 server infrastructure is often the base for further SNA 3270 application access modernization steps: SNA 3270/HTML transformation and Web services integration.

There are a significant number of options available as to which hardware or software platform to use and where in your overall network topology you place your TN3270 servers. It is not possible to give a general and simple recommendation, but here are some important points to consider when deciding where to place your TN3270 servers and how to design the environment around your TN3270 service:

- Continuous availability

  The TN3270 service is likely to be a critical service in your network infrastructure. Your design should ensure that there are no single points of failure around your TN3270 service. This includes TN3270 server instances, operating system instances, hardware upon which the TN3270 service runs, switches, load balancers, and the back-end SNA connectivity itself between the TN3270 service and the SNA applications.

- Scalability

  Your TN3270 service must be able to scale to the maximum number of concurrent TN3270 server connections you need in your environment. If that requires more than a single TN3270 server, it means determining which load balancing technology to use.

  Note that use of some TN3270 functions, such as reconnect, printer association, and some forms of specific LU name assignment, may require load balancing with timed affinity to the server instance that was initially chosen for a given client IP address.

- Manageability
  - How to maintain the TN3270 server and associated VTAM configurations without having to manually enter every single possible LU name that you can use. Support of clone definitions, generic LU names and IP address ranges to minimize configuration definitions.
  - Real-time response time monitoring capabilities.
  - Data for charge-back or accounting purposes.
- Performance: predictable and consistent
  - Predictable end-user response times for predictable workloads—connection setup and steady-state.
  - Predictable resource usage on the TN3270 server nodes: CPU and memory.
- Security
  - Is any access control to SNA applications required, primary LUs, based on client IP address, range or specific?
  - Are there special requirements for early user authentication?
  - SSL or TLS client authentication, session monitor log on, IBM Session Manager, IBM Candle®/SuperSession, CA-TPX, and so on, network solicitor.
  - Are secure (encrypted) connections required for all or a subset of connections?
  - What are the requirements for encryption algorithms and key length?

  - How do you manage the associated Public Key Infrastructure (PKI), server/client certificates, Certificate Revocation List (CRL) processing, and so on?

- Will all TN3270 access be from intranet users, or are there external TN3270 users?
  - Business partners
  - Customers
  - Private accounts
  - Business accounts
  - Employee remote access (over the Internet)
  - Support personnel (IBM, etc.)
- LU name assignment, also known as LU nailing
  - Is there a requirement for predictable LU names (specific LU naming)?
  - This is a large area with many detailed requirements that need to be identified and mapped to the various TN3270 server offerings you are looking at. They may not all provide the type of mapping you need.
- Do you need to support 328x printer technology through your TN3270 server?

  If you require printer support, do you require to ensure a certain match between display LU names and printer LU names, known as printer association?

Analyze the previously noted areas and map your design requirements for each, in comparison to the capabilities of the potential TN3270 service platforms you are looking at, and choose the one that best meets your requirements.

Note: In general, the higher the number of TN3270 connections you need to support, the more you will prefer to implement the TN3270 service on the mainframe itself.

For anything over 20,000 concurrent TN3270 connections, your first choice should be z/OS itself. You can still implement TN3270 services that support the same number of connections on other platforms. However, the higher the number, the more complex the overall design tends to become with non-z/OS platforms due to scalability issues.

4.1.2 Real IBM 3270 devices

Real IBM 3270 cluster controllers and terminals still exist in surprising numbers.

Such devices are still in use either connected through channels or LANs to the mainframe or via serial lines or token-ring LANs to an IBM 3745/46.

Because a real IBM 3270 terminal is a rather “unintelligent” device, one cannot just switch to TN3270 from such devices. If it is possible to replace the terminals with workstations, you can use TN3270 for SNA 3270 application access, but if such replacement was possible, it has probably already been done.

Some of the existing channel-attached cluster controllers are currently used for operating system console access. The OSA adapter technology supports a feature that is known as Integrated Console Controller (ICC), which is a small built-in TN3270 server that resides in the OSA adapter itself and presents an interface to the mainframe operating system as though it were a channel-attached IBM 3174. This technology assumes that the actual IBM 3270 console terminals are replaced with programmable workstations upon which a TN3270 client can be deployed.

For the wide area network connected cluster controllers and terminals, the question to answer is one of physical connectivity. Because one of the objectives of SNA modernization is to

remove dependency on IBM 3745/46 hardware, you need to find another technology to support the SNA boundary functions for such physical devices.

There are two technologies that you can use:

- In an SNA subarea environment where a boundary function NCP is deployed in CCL, you can terminate the serial lines in an aggregation layer router in the data center and switch to LAN connectivity into the boundary function NCP. The remote cluster controller and terminals will appear to the host operating system as they do currently, without any further changes required.

  As an alternative in an SNA subarea environment, VTAM itself, by connecting to the LAN over an OSA LSA port, can perform the boundary functions without the assistance of an NCP. Note that the SNA element addresses in this type of configuration come out of VTAM's low-order address pool; however, for a small number of devices this is probably not an issue.
- In an APPN environment, you can again terminate the serial lines in a router or a distributed server that also acts as an APPN node supporting SNA boundary functions in terms of DLUR. The router or the server needs to have serial line interface adapters installed, and will, via its DLUR-DLUS session, set up sessions for the physical IBM 3270 terminals. The remote cluster controller and terminals will appear to the host operating system and SNA environment as they do presently. Cisco routers can perform such functions, and also Intel servers with the IBM Communications Server for Windows or the IBM Communications Server for Linux.

So for real IBM 3270 terminal access, you cannot modernize the application access, but you are required to address its continued use through an SNA infrastructure modernization, preserving the end-user environment completely as it is presently.

4.2 How to modernize SNA client/server application access

You can modernize SNA client/server application access using SNA network infrastructure modernization technologies, or select SNA application access modernization technologies, where the primary objective is one of preserving the SNA client application on whatever platform it may be running on.

To preserve the SNA application, or at least the GUI dialog associated with the remote SNA application, on the remote workstation or branch server, one of three technologies can be used:

- Preserve the SNA client application on the workstation or branch server, but collapse the SNA protocol stack into the data center using the IBM remote SNA API technology.
  - If the SNA client is located on an operating system platform that is covered by the IBM Remote SNA API client technologies, the IBM remote SNA API client can be installed on the workstation and a TCP connection over an IP network established to an IBM remote SNA API server in the data center.
  - The IBM Remote SNA API server may be either a peripheral node in an SNA subarea network environment or a peer node in an APPN network environment.
- Preserve the SNA protocol stack and SNA client application on the workstation, and transport SNA traffic over an IP network using EE or DLSw.
  - If an APPN network environment has been established and the node where the SNA client application resides supports HPR over IP, then EE can be used as the SNA network infrastructure modernization technology between the SNA client and the

mainframe. EE will work for both dependent (LU Type 0, 1, 2, and 3 and LU Type 6.2) and independent LU types (LU Type 6.2), using DLUR or DLUS for dependent LUs.

  - In an SNA subarea environment one will normally use SNA LLC2 flows over a LAN to a local router that either implements a DLSw connection over a wide-area IP network up to the data center, where an NCP or VTAM itself will perform SNA boundary functions, or the router itself will perform boundary functions and use DLUR over EE up to the data center.

- Preserve the GUI interface on the workstation, but collapse the SNA application and the SNA protocol stack into the data center using X-Windows or Microsoft Windows Terminal Services technology. These technologies are also known as *split GUI*.
  - If the SNA client is deployed on Windows, the X-Windows or MS Windows Terminal Services technology can be used to preserve the GUI dialog on the end-user workstation, but move the SNA application itself to a server in the data center.
  - The communication between the end-user workstation and the server is based on a TCP connection over an IP network.

All three technologies will meet the overall objective of removing SNA wide area network traffic. For Windows-based client workstations, both the IBM Remote SNA API technology and the split GUI technology will further address the objective of collapsing SNA protocol stacks into the data center.

4.2.1 Remote SNA API client/server (split stack)

The TN3270 protocol is a special-purpose protocol that supports IBM 3270 data streams and variations thereof only (dependent SNA LU types 1, 2, and 3).

If other LU types are deployed in the branch environment, then you can use the Remote SNA API client/server technology to achieve similar network topology advantages as TN3270 does for LU Types 1, 2, and 3.

Note: The remote SNA API allows the SNA application to remain on the end user workstation or branch server node, where it runs presently, by replacing the SNA protocol stack with a remote SNA API client function.

The remote API client function provides an SNA API layer to the local SNA application, which is equivalent to the SNA API that was provided by the full SNA protocol stack. The remote API client function sends SNA API calls over a TCP connection via an IP network to a remote SNA API server node that is typically deployed in the data center. This again converts the wide area network traffic to IP and consolidates the SNA nodes and network traffic to the data center or the mainframe itself. See Figure 4-3 on page 68.

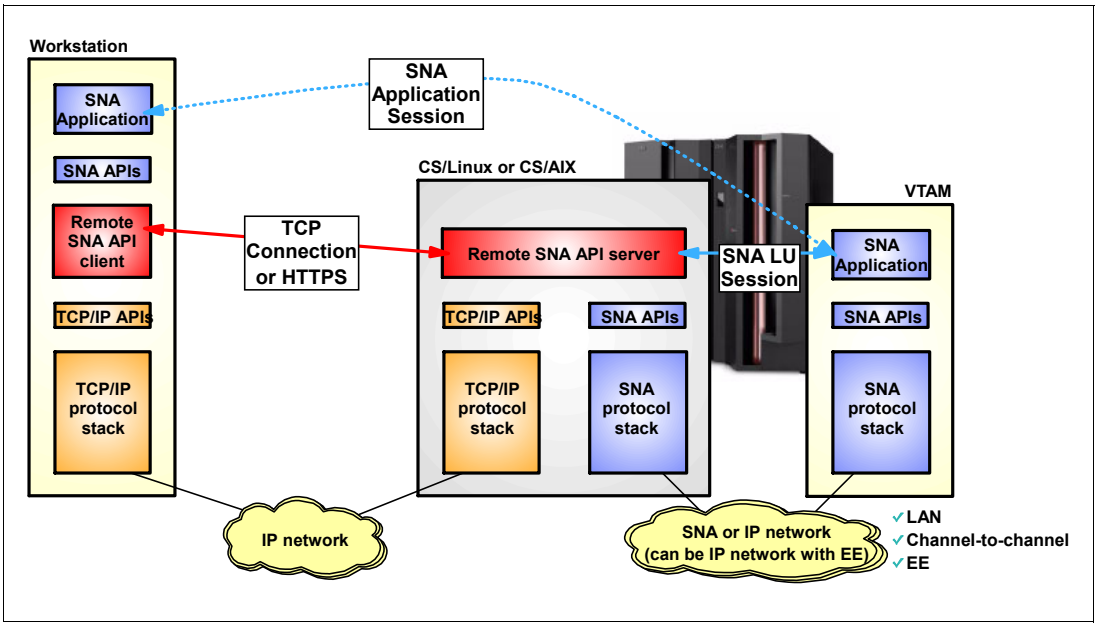


Figure 4-3 Remote SNA API client/server

The topology becomes one of having the SNA application on the workstation, connected via an IP network to an SNA node, where the LU representing that SNA application resides. From here the topology continues, as before, with a traditional SNA LU session to the partner SNA application. The node where the SNA remote API server is deployed may be part of an SNA subarea network or an APPN network, and establish sessions on behalf of the remote SNA application to both SNA subarea nodes and peer nodes.

The remote SNA API client/server technology supports both dependent and independent LUs and is transparent to both the local and remote SNA application.

The main objective of this technology is to address LU0 and LU6.2 branch applications that must remain in the branch. 3270-based communication will most likely be addressed via the TN3270 protocol, but it is possible to use a local SNA-based 3270 emulator with the remote API client to support SNA 3270 traffic over this technology also.

Consider the case described previously: An EHLLAPI program that proves to be sensitive to a switch from an SNA 3270 emulator to a TN3270-based emulation. One way to try and address such an issue, without rewriting the EHLAPPI program, and still avoid a full SNA protocol stack on the workstation, is to combine an SNA 3270-based emulator, such as the IBM PCOMM emulator, with the remote SNA API client/server technology.

There are at least three different remote SNA API client/server protocols (Figure 4-4 on page 69) available from Microsoft and IBM:

- Microsoft's remote SNA API client/server
- IBM Communications Server for Windows remote SNA API client/server
- IBM Distributed Communications Server remote SNA API client/server
  - IBM Communications Server for Linux
  - IBM Communications Server for Linux on System z
  - IBM Communications Server for AIX

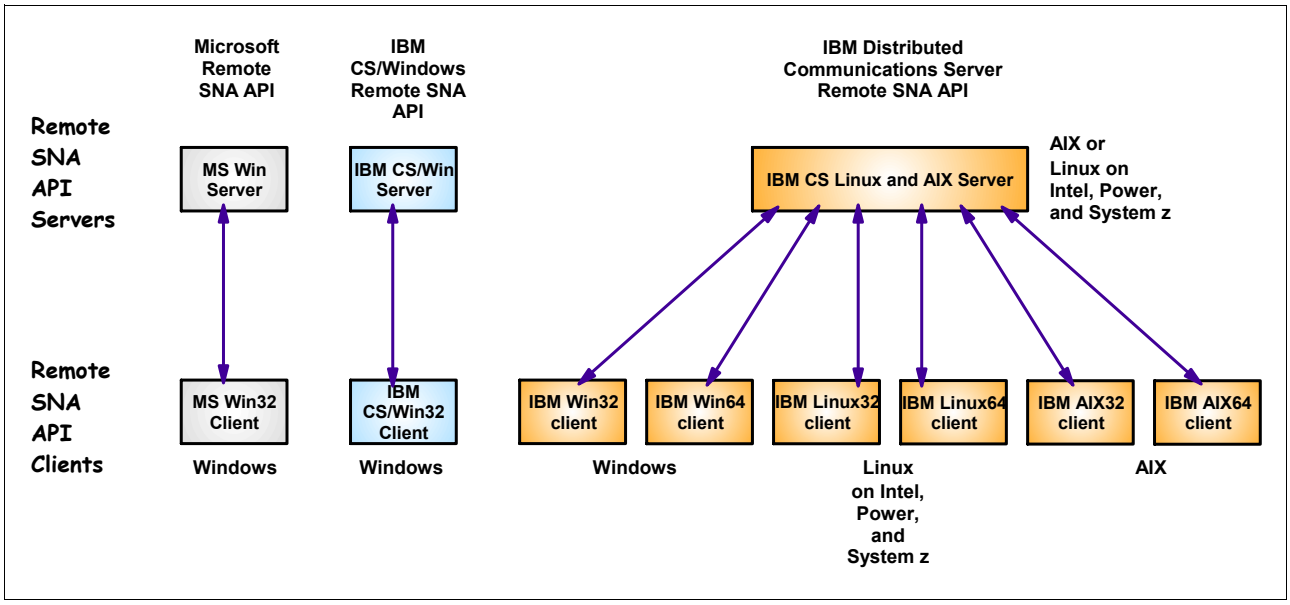


Figure 4-4 Remote SNA client/server protocol compatibility

From an SNA API perspective, both the MS Win32®, the IBM CS/Win32, the IBM Win32, and the IBM Win64 client support the same programming interfaces: LU0 (Request Unit Interface (RUI) and Session Level Interface (SLI)), and LU6.2 (Common Programming Interface for Communications (CPI-C)).

IBM Distributed Communications Server Remote SNA API client is supported on Windows i686, Windows x86_64, AIX, Linux i686, Linux x86_64, Linux OpenPower™, Linux Power5, and Linux on System z.

IBM Distributed Communications Server Remote SNA API server is supported on Communications Server for AIX, Communications Server for Linux (i686, x86_64, OpenPower, Power5), and Communications Server for Linux on System z.

IBM Distributed Communications Server Remote SNA API Client communicates over a TCP connection to an IBM Distributed Communications Server Remote SNA API Server. In addition, IBM Distributed Communications Server Remote SNA API Client configuration is extended to allow a Web server to be specified as part of the server and use HTTP(S) transport to communicate to IBM Distributed Communications Server Remote SNA API Server. The HTTP transport implements a reliable—and optionally secure, in the case of HTTP—link between the client and the Web server.

As an example, specifying a server name of snawebserver.ibm.com:cslinuxserver.ibm.com instructs the client to use the Web server snawebserver.ibm.com as an intermediary to connect to the IBM Distributed Communications Server Remote SNA API Server cslinuxserver.ibm.com. This allows you to access different IBM Distributed Communications Server Remote SNA API Servers directly or via different Web servers.

Note: The remote SNA API client/server technology is a simple, efficient, and transparent technology that allows continued use of SNA clients on workstations and distributed servers without the full impact of an SNA protocol software on those client nodes.

With remote SNA API server functions being supported on Linux on System z, SNA over IP flows can be implemented all the way into System z.

4.2.2 Remote desktop (split GUI)

X-Windows or Microsoft Remote Desktop and Windows Terminal Services (WTS) are solutions that often are used in combination with one or more of the general SNA or IP integration technologies.

X-Windows or Microsoft Remote Desktop with WTS provide a remote GUI interface. The SNA application can be consolidated onto a server in the branch or in the data center, while the GUI remains on the user workstation that communicates with the server over an IP network.

If the application in question is an SNA application, such as an SNA 3270 emulator, the “green screen” appears on the user workstation, but the SNA emulator and, optionally, the SNA protocol stack reside on the server.

By combining split GUI with SNA or IP integration technologies, the SNA network coverage can be further reduced and the number of SNA protocol stacks can be smaller than the number of split GUI server nodes, which has to be sized based on the number of remote workstations to support. The 3270 emulator on the split GUI server may be a TN3270 emulator using a TCP connection between the split GUI server and the TN3270 server that may be located on System z.

The closer the split GUI servers can be pulled into the data center, the larger the IP network coverage will be and the smaller the SNA network coverage will be. But nothing prevents a split GUI server from being located in the branch and used in combination with EE, DLSw, or IBM remote SNA API technology to transport the SNA application data from the server over the IP wide area network to the SNA partner application in the data center. See Figure 4-5.

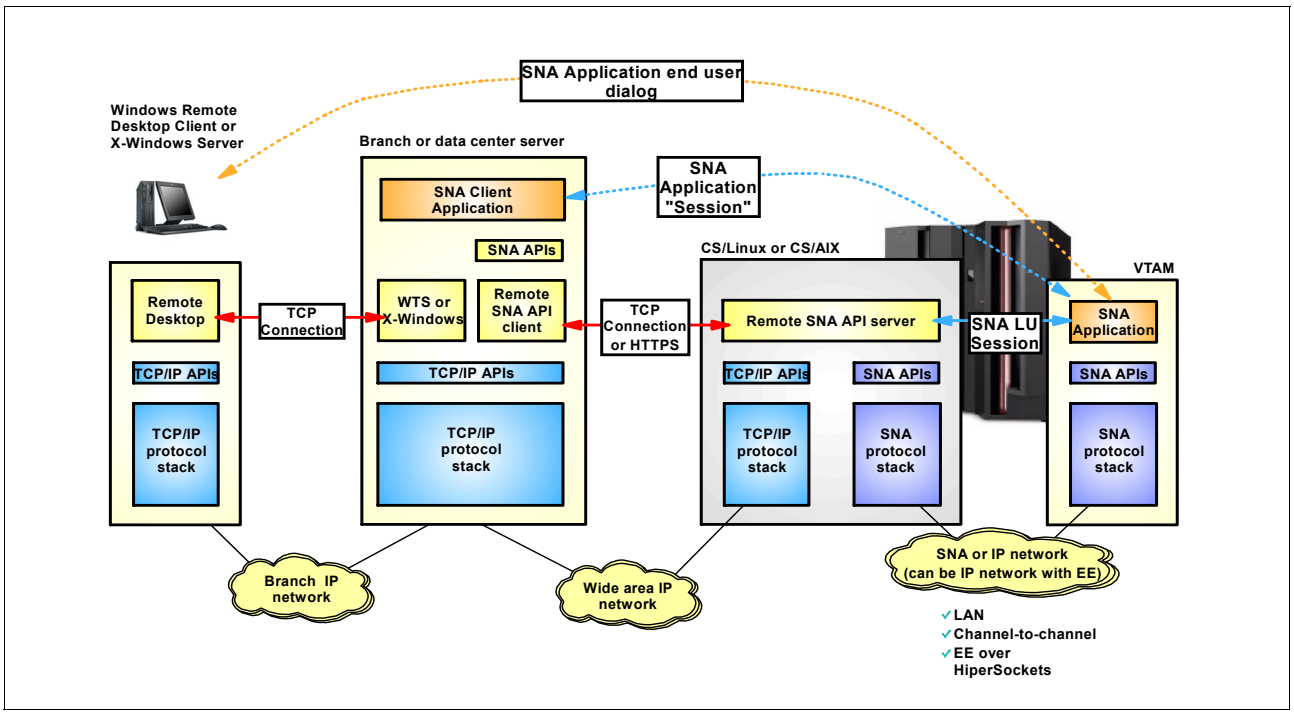


Figure 4-5 Split GUI in combination with IBM Remote SNA API client/server

For further information about the Microsoft Remote Desktop and Windows Terminal Services solutions, refer to Microsoft-specific product documentation at:

http://www.microsoft.com

4.3 SNA application access transformation

This section looks at transforming access to existing SNA core business applications from the perspective of the stages in the Enterprise Transformation model:

- Stage 1: Presentation integration (enhancing the user experience)

  Enabling a Web browser to act as the client for existing SNA applications by transforming the SNA data stream to an HTML data stream over an HTTP(S) connection from a Web browser.

  For SNA 3270 applications, such transformation can often be done without writing new presentation logic. The transformation can in most cases be rules-driven where the transformation rules are generated by a development toolkit based on the SNA 3270 data stream or, when the transformation is done in CICS or IMS, the unformatted application data structures (ADS). ADS is the data format that is used by CICS or IMS transactions before Basic Mapping Support (BMS) or Message Formatting Services (MFS) adds or strips off the 3270-specific data stream elements.

  For SNA server applications that are accessed from LU0 or LU6.2 client programs, the transformation normally requires development of some new presentation logic to transform whatever presentation the existing client program did, to the new user interface.

- Stage 2: Programmatic integration (adapt for enhanced relationships)

  Enabling SNA applications as Web services by providing a Web service development tooling and runtime technology that can be used to expose SNA application as Web services.

  For SNA 3270 applications, exposing the SNA 3270 core business application as a Web service is an additional element of the transformation that was discussed previously. The transformation is extended to also expose the SNA 3270 application as a Web service, and transform the SNA 3270 data stream to or from SOAP/XML instead of HTML.

  For SNA server applications, the transformation again normally requires development of a Web service wrapper component that is aware of the SNA client/server application protocol and knows how to transform the application data stream to or from SOAP/XML.

For both levels of transformation, there are both subsystem-independent solutions and subsystem-specific solutions; see Figure 4-6 on page 72:

- At a subsystem-independent level, where the transformation is applied to the actual SNA data stream as it flows through the network infrastructure.

  The subsystem-independent solutions implement a transformation tier between the new client environments (Web browsers and Web service requesters) and the mainframe SNA applications in such a way that the application subsystems, such as IMS and CICS, continue to be accessed through an SNA session.

  From the perspective of IMS, CICS, TSO, and so on, there is no difference between an end user who uses this core SNA business application from an SNA 3270 terminal, a TN3270 client, a Web browser, or a Web service client. It is the same SNA 3270 application environment and same application that is being invoked.

  Because it is unlikely that all users will change the client environment at the same time, the use of this type of technology offers an easy-to-manage transition environment where only a single version of the core SNA business application needs to be maintained.

  Refer to topology A in Figure 4-6 on page 72.

- At a subsystem-specific level where the transformation is applied by subsystem-specific components that have awareness of the application input/output data areas with or without the 3270-specific data stream elements.

  The subsystem-specific solutions do not use a single common topology, but they do have some common characteristics, such as the ability to totally remove the requirement for SNA connectivity into the subsystems. The Web browser or the Web service requester will either connect directly into the subsystem or to an intermediate tier that will use non-SNA connectivity technologies for connectivity into the subsystem.

  CICS supports two main topologies: where clients connect directly into a CICS region, and where clients connect via a WebSphere Application Server tier that then uses CICS-specific J2EE Connector Architecture (J2C)-based resource adapters via CICS Transaction Gateway (CTG). CTG uses cross-memory services into CICS (if it is CTG for z/OS) or either TCP/IP or SNA into CICS (if it is Multiplatform CTG). Refer to topology B in Figure 4-6.

  IMS uses either a WebSphere Application Server tier or an IMS-specific SOAP gateway tier that uses TCP/IP connectivity to a z/OS address space known as IMS Connect. IMS Connect then uses IMS Open Transaction Manager Access (OTMA) for connectivity into the IMS transaction manager. Refer to topology C in Figure 4-6.

  In the discussion of the subsystem-specific solutions, this document will limit itself to the capabilities that are relevant from an SNA data stream-to-HTML transformation and an SNA Web service interface perspective. The subsystem-specific solutions offer many capabilities for other variations of Web enablement, but we will refer you to product-specific documentation for more details on those additional capabilities.

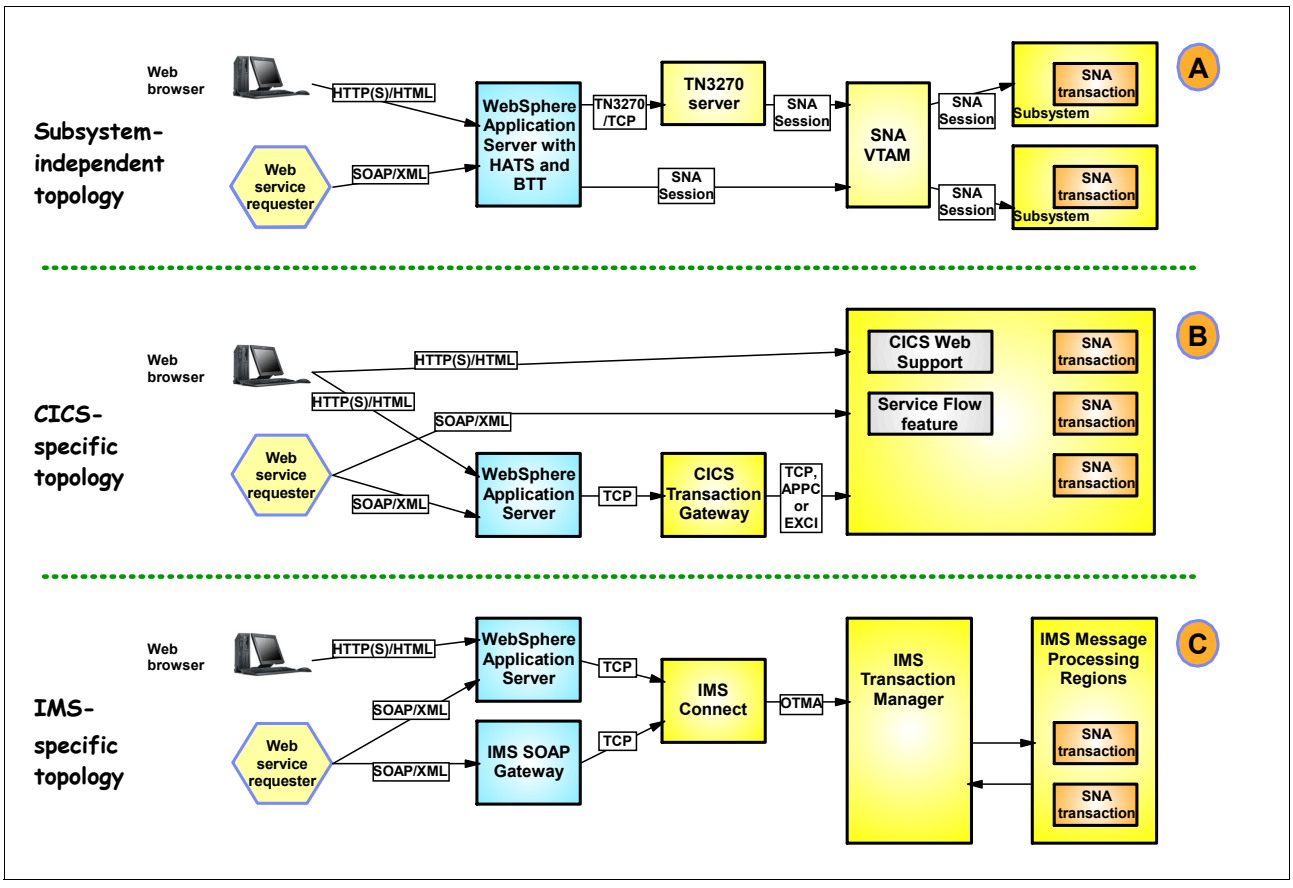


Figure 4-6 Web-enabling solutions: topology overview

Note: When deciding if you are going to use a subsystem-independent solution or one or more subsystem-dependent solutions to Web-enable your SNA core business applications, you need to analyze much more than just the topology aspects of the two types of solutions. Each offers different levels of transformation capabilities. A detailed analysis of what is required, compared to what is offered by these solutions, in combination with the network topology aspects, should determine which you will use.

See Table 4-1 for a list of the main solutions you can use to Web-enable SNA core business applications, and which type of SNA applications they will work with.

Table 4-1 Web-enabling SNA core business applications - main solutions and applications

	SNA 3270 application			**SNA LU0 and SNA LU6.2 application**		
	Subsystem independent	**CICS specific**	**IMS specific**	**Subsystem independent**	**CICS specific**	**IMS specific**
Presentation integration SNA data stream/HTML transformation	Host Access Transformation Services (HATS)	CICS Web Support and 3270 Web bridge, or CTG	IMS MFS Web Enablement	Branch Transformation Toolkit (BTT)	CICS Web support or CTG	IMS MFS Web support
Programmatic integration - present SNA core business applications as Web services	Host Access Transformation Services (HATS)	CICS Service Flow feature and Link3270	IMS MFS Web services	Branch Transformation Toolkit (BTT)	CICS Service flow feature	IMS MFS Web services

4.3.1 Presentation integration - SNA 3270/HTML transformation

This section looks at the solutions that are available for transforming SNA 3270 data streams to HTML, and the following section looks at how to extend that level of transformation into a programmatic transformation.

SNA 3270/HTML transformation using HATS

IBM Host Access Transformation Services (HATS) provides a subsystem-independent SNA 3270/HTML transformation solution.

HATS consists of two components:

- The development toolkit

  The HATS development toolkit is integrated with the Eclipse-based IBM Software Development Platform (IBM SDP).

- The runtime support

  HATS runs on an IBM WebSphere Application Server on AIX, Solaris™, Windows, i5/OS, z/OS, HP UNIX®, Linux for Intel, Linux for iSeries™, Linux for pSeries, or Linux for zSeries®.

  Deploying HATS projects that were created in the HATS development toolkit as portlets requires IBM WebSphere Portal.

The HATS development toolkit provides a set of wizards and editors that guide the HATS developer through creating and modifying HATS applications.

HATS applications are packaged in standard J2EE formats, and can be deployed directly to IBM WebSphere Application Server or IBM WebSphere Portal Server.

From a topology perspective, the end user accesses the HATS application on the application server using a browser via standard HTTP(S) protocols (Figure 4-7). HATS connects to a TN3270 server. The TN3270 server can be located on the same server node as where WebSphere Application Server with HATS is running, or the TN3270 server can be located on another node in the IP network, in which case, there will be a TN3270 TCP connection over an IP network between the WebSphere Application Server running HATS and the TN3270 server.

The WebSphere Application Server running HATS may be deployed on a variety of operating system platforms, including z/OS itself. In this case, all three logical tiers (Transformation tier, TN3270 server tier, and application tier) are collapsed into a single operating system tier, which is z/OS. Each tier may alternatively be deployed on separate operating system tiers and even hardware platform tiers, but from an overall network simplification perspective, limiting the number of operating system and hardware tiers to manage is preferable.

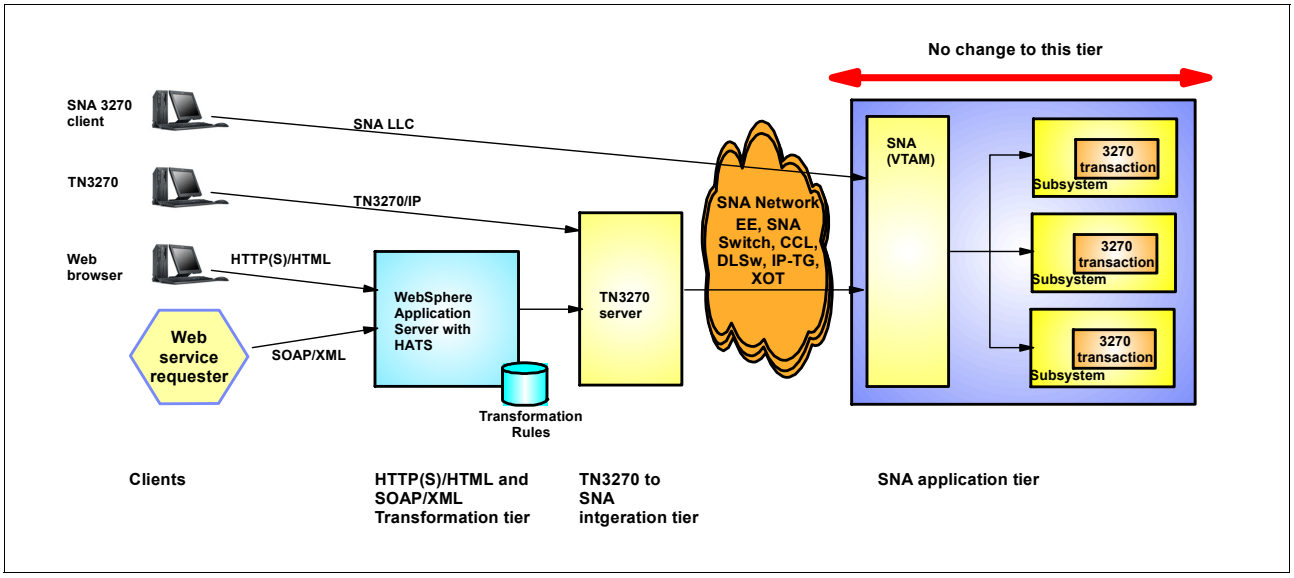


Figure 4-7 HATS connectivity topology

As usual with TN3270, there needs to be SNA connectivity between the TN3270 server and the mainframe SNA applications. From an overall network infrastructure simplification perspective, placing WebSphere Application Server running HATS in the data center is preferred since it allows for use of HTTP(S) over the IP network infrastructure as opposed to a TN3270 protocol.

The size of the HTML data that results from the HATS transformation requires more network bandwidth than the compact TN3270 protocol. On average, each screen that is sent over the TN3270 protocol is roughly 2 KB. Using HATS, an average screen, transmitted as HTML and images over HTTP is about 25 KB, and can range higher, depending upon the rendering options chosen and the number of images contained in the application. Technologies such as caching (browser and Web server), Web server compression, and HTTP compression can be used to reduce the size of the data to roughly 3 to 4 KB. However, preparation for this change in network bandwidth requirements is necessary.

Because the connectivity between the workstation and HATS is based on HTTP(S), firewalls in between the workstations and the HATS server only need to permit traditional HTTP(S) protocols. There is no TN3270 connectivity between the workstation and the HATS server.

HATS offers a wide range of options for customizing the SNA 3270/HTML transformation, but from the perspective of just providing an “out of the box” transformation, HATS also offers what is known as default transformation rules where HATS can present a somewhat familiar SNA 3270 look-and-feel on the browser interface. The default transformation rules of HATS can be used with no or minimal customization of the HATS environment.

With relatively simple customization efforts, it is possible to transform the end-user experience. For example:

- Application navigation can be streamlined with the use of skip screen macros and global variables to enter saved information on behalf of the end user.
- Data from multiple 3270 screens or hosts can be combined onto a single HTML page.
- 3270 menus and function keys can be transformed to links, buttons, radio buttons, or drop downs, that will be easier to use.
- The presentation of the application can be modernized to match your corporate Web site look and feel.
- Tables of numeric data on the host screen can be transformed into bar or line graphs.

This list does not include all the capabilities of HATS. Refer to the detailed HATS documentation for more information.

Note: HATS is the most comprehensive SNA 3270/HTML transformation technology from IBM for stage 1 of Enterprise Transformation, enhancing the user experience.

The main advantage of HATS is that HATS provides a 3270/HTML transformation technology that is independent of SNA application subsystem. HATS may be used to transform access to SNA 3270 applications in any subsystem including, but not limited to, CICS, IMS, and TSO. HATS supports 5250 transformation also and can be used for 5250/HTML transformation in addition to 3270/HTML transformation. This includes the capability to integrate data from both SNA 3270 and 5250 sessions into a single HATS application. Refer to the following Web site for more details about HATS capabilities:

http://www.ibm.com/software/webservers/hats/

CICS-specific SNA 3270 presentation integration

CICS provides built-in solutions for Web-enabling CICS SNA 3270 transactions through its integrated 3270 bridge technology.

The 3270 bridge solution comes in three different implementations:

- The CICS Web Bridge
- The Link3270 Bridge
- The WebSphere MQ Bridge

The CICS Web Bridge implementation is the technology of main interest when transforming CICS SNA 3270 applications interfaces to HTML. It is used in conjunction with a built-in HTTP(S) server in CICS, the CICS Web Support (CWS) component; see Figure 4-8 on page 76.

A Web browser connects to CWS using HTTP(S) and sends an HTTP request that gets translated into a request through the CICS 3270 bridge components to the SNA 3270 transactions. Output is again transformed, using customizable transformation rules, and returned to the Web browser as an HTML document. The CICS Web Bridge supports SNA 3270 transactions that use Basic Mapping Support (BMS) and also transactions that do not

use BMS. For transformation of BMS transactions, the transformation is based on the Application Data Structure (ADS), and not the final 3270 data stream.

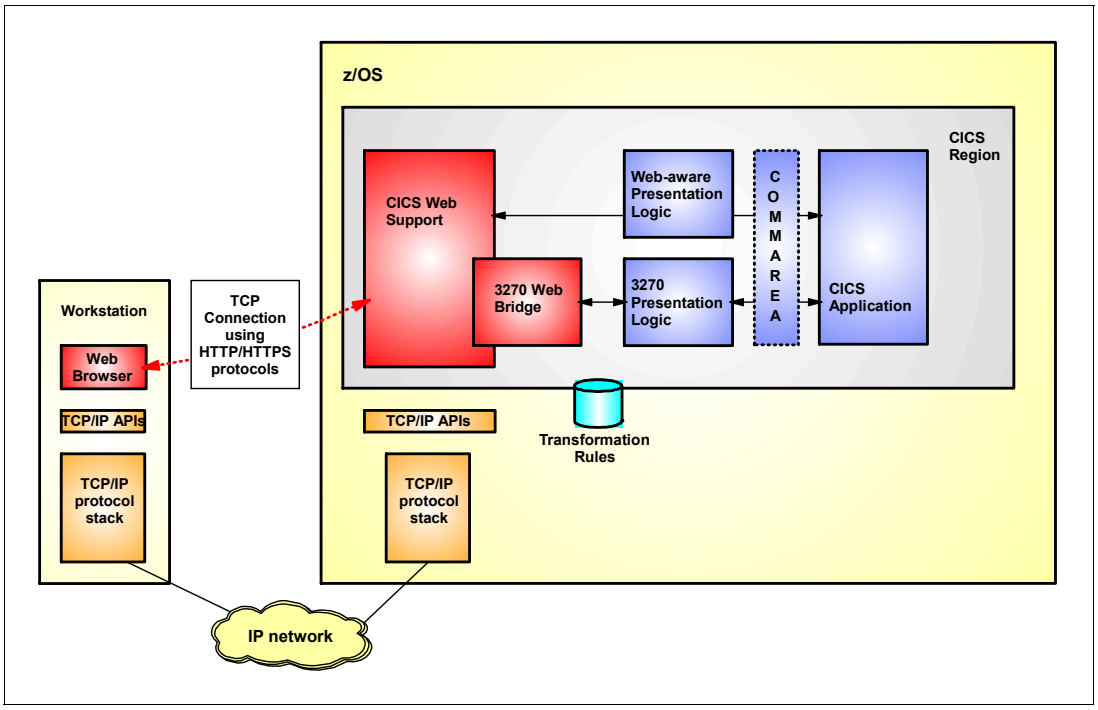


Figure 4-8 CICS Web Support with the CICS Web Bridge topology

The advantage from a network topology perspective of using the CICS Web Bridge solution is that SNA connectivity is totally removed. Connectivity into the CICS Web Bridge is IP-based. There are no SNA LUs involved in this topology.

For a relatively simple SNA 3270/HTML transformation of CICS SNA 3270 transactions, the CICS Web bridge technology may be sufficient. If complex transformations are required, or if data from multiple SNA 3270 applications, of which only some may be CICS transactions, is required, then HATS provides a more comprehensive SNA 3270/HTML transformation solution.

By using one of the other 3270 bridge implementations, other connectivity options into CICS, such as CICS Transaction Gateway or WebSphere MQ messaging, are offered. Each with their own set of Web-enablement capabilities. Some amount of application development to implement the actual SNA 3270/HTML transformation may be required in those cases. Refer to the following Web site for further details on CICS Web-enablement support:

`http://www.ibm.com/software/htp/cics/tserver/v31/`

IMS-specific SNA 3270 presentation integration

IMS provides HTML transformation for applications based on IMS Message Formatting Services (MFS) through components of its IMS Integration Solutions Suite. MFS is the IMS equivalent to CICS Basic Mapping Support (BMS) and serves the same purpose, which is processing 3270 data streams on display devices to relieve the transaction programs of having to deal with device-dependent display information.

The components of interest from an SNA 3270/HTML transformation perspective are:

- The IMS Connect - A gateway function on z/OS that provides a TCP/IP based interface to the IMS Open Transaction Manager Access (OTMA) interface.

- The IMS Connector for Java - A J2EE Connector Architecture (J2C) resource adapter that can be used by J2EE applications to access IMS transactions via IMS Connect.
- The IMS Message Formatting Services (MFS) Web Enablement - A servlet-based data transformer that allows reuse of existing MFS-based IMS business logic and preserves 3270 screen flow through dynamically generated MFS-based Web pages; see Figure 4-9.

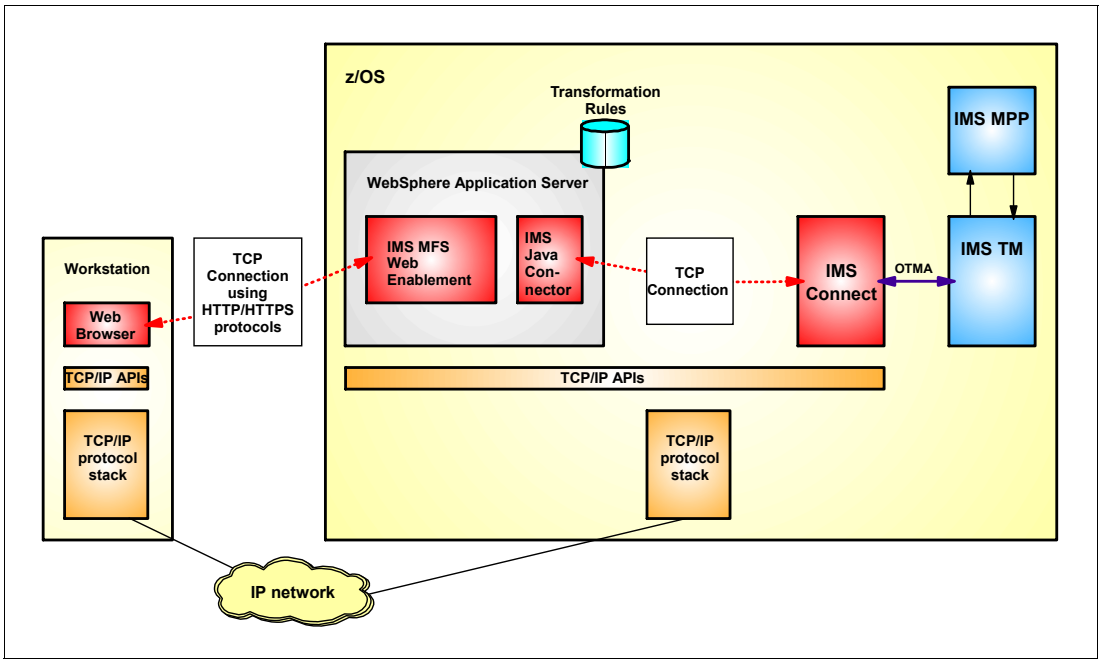


Figure 4-9 IMS MFS Web Enablement topology

The advantage of using the IMS Web enablement solution, from a network topology perspective, is that SNA connectivity is totally removed. Connectivity into IMS Connect is TCP/IP-based, and connectivity between IMS Connect and IMS transaction manager is XCF-based. There are no SNA LUs involved in this topology at all.

IMS MFS Web enablement consists of two components:

- The development tool

  The MFS XML utility is a command-line tool that runs on Windows. It provides step-by-step instructions to guide you through parsing of existing MFS source files and creation of the MFS Web enablement application.

- The runtime support

  An MFS Web enablement application runs on IBM WebSphere Application Server on AIX, Windows, or z/OS. The end user accesses MFS Web enablement applications on the application server using a browser via standard HTTP(S) protocols. MFS Web enablement runtime support transforms the request from device data structure, in IMS known as Device Input Format (DIF) and Device Output Format (DOF), to the application data structure, in IMS known as Message Input Descriptor (MID) and Message Output Descriptor (MOD). The transformed application data structure is then passed via IMS Connector for Java, IMS Connect, and OTMA, to invoke an IMS transaction. On return, an MFS Web enablement application dynamically renders MFS-based Web pages using cascading style sheet. The application developer can customize the style sheet to change the look and feel of generated pages. Refer to the following Web site for more details on IMS MFS Web Enablement support:

  http://www.ibm.com/ims

4.3.2 Programmatic integration - SNA applications as Web services

This section provides a brief introduction to what a Web service is and how to integrate and reuse existing SNA core business applications as Web services.

See the following Web site for more details on Web services and the Services Oriented Architecture:

http://www.ibm.com/developerworks/webservices

A Web service is a software component that is described via an industry-specific description language known as Web Services Description Language (WSDL) and can be accessed via standard network protocols such as, but not limited to, Simple Object Access protocol (SOAP) over HTTP(S). On many platforms, IBM WebSphere Message Queuing (WMQ) is another often-used transport mechanism for SOAP messages between Web services.

Software applications written in various programming languages and running on various platforms can use Web services to exchange data over computer networks in a manner similar to interprocess communication on a single computer. This interoperability (programming languages, operating systems, application server platforms, and so on) is due to the use of open standards.

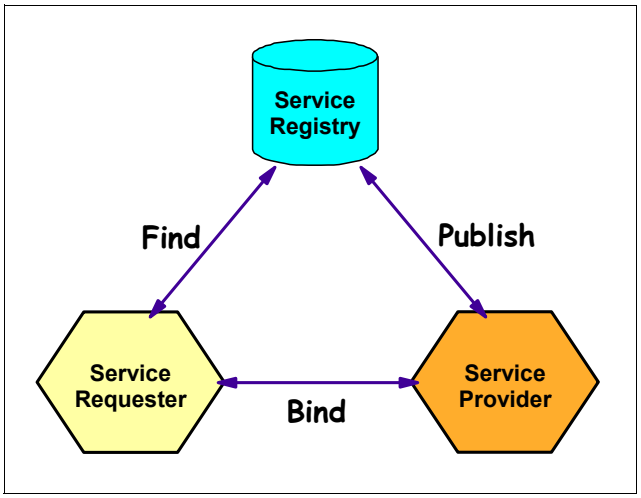


Figure 4-10 Web service architecture

The Web services architecture (see Figure 4-10) is based upon interactions between three components: a service provider, a service requester, and an optional service registry.

The service provider is the platform that hosts access to the service.

The service requester is the application that is looking for and invoking or initiating an interaction with a service. The requester role can be played by a browser driven by a person or a program without a user interface, such as another Web service.

The service registry is a place where service providers publish their service descriptions, and where service requesters can find them. The registry is an optional component of the Web services architecture and provides for dynamic binding. For static bound service requesters, a service provider can send the description directly to the service requester. Likewise, service requesters can obtain a service description from other sources besides a service registry, such as a local file or an FTP site.

The interactions between the components involve the following operations:

1. Publish

   In order to be accessible, a service needs to publish its description such that the requester can subsequently find it. Where it is published can vary depending upon the requirements of the application.

2. Find

   The service requester uses a find operation to retrieve the service description from the registry. The find operation may be involved in two different life cycle phases for the service requester: at design time in order to retrieve the services interface description for program development, and at runtime in order to retrieve the services binding and location description for invocation.

3. Bind

   The service requester uses the service description to bind with the service provider and interact with the Web service implementation.

You can view Web services as the application building blocks that you can use individually or orchestrate into business processes using business process models that describe the flow of the overall processes, data input and output, policies, and so on, that make up a specific business process.

Note: By providing a Web service interface (a Web service wrapper) to existing SNA core business applications, those SNA applications can be exposed as Web services. They can then be included as reusable Web service elements in new business processes. These are implemented using today's business process modeling and business process runtime technologies from both IBM and other vendors that support the Web services-related standards.

SNA 3270 applications as Web services - using HATS

We have discussed how to transform the presentation between the 3270 session and an HTML browser-based dialog with HATS, but HATS offers integration capabilities beyond that.

In conjunction with IBM Software Development Platform, HATS also helps simplify the creation of standard Web service interfaces to provide access to host SNA 3270 applications. Web services protocols and standards, such as SOAP and WSDL, are an efficient and reusable means to provide standardized access to your host systems, helping you lower the cost to maintain and deploy connectors to these systems.

Use these Web services to integrate your host-based business tasks with other internal or external applications to augment your service-oriented architecture (SOA) solutions.

Note: HATS provides the capability to create a Web service interface to existing SNA 3270 applications, independent of which subsystems these applications are deployed in.

The overall network topology for a HATS environment in this scenario is more or less the same as for an SNA 3270/HTML transformation. The only addition is that the client does not have to be a Web browser, but it can be any Web service requester, including a Microsoft .NET client.

SNA LU0 or LU6.2 server applications as Web services - using BTT

While HATS provides capabilities to create Web service interfaces for existing SNA 3270 applications, it does not provide the same capability for non-SNA 3270 applications, such as SNA LU0 or LU6.2 servers.

To create a Web service interface to existing SNA LU0 or LU6.2 server applications, IBM Branch Transformation Toolkit (BTT) for WebSphere Studio can be used to create a “wrapper” application in the form of a Web service that interacts with the existing SNA LU0 or LU6.2 server application.

BTT is a component-based toolkit for developing enterprise e-business applications (Figure 4-11 on page 80). The Branch Transformation Toolkit enables the development of interfaces to the services of existing information systems so that they become ubiquitous through all delivery channels, such as the traditional branch, call center, kiosk, Internet, and mobile access. This minimizes the requirement for developing new code and reduces the time required to make new services available to all delivery channels.

BTT provides a set of adapters that allow Web services running in a WebSphere Application Server environment to interface with existing enterprise applications and data. Two of the adapters provided by BTT are of interest from an SNA application access modernization perspective:

- SNA J2C LU0 resource adapter
- SNA J2C LU62 resource adapter

Both adapters are based on the J2EE Connector Architecture (J2C).

The J2C LU0 resource adapter supports a Request Unit programming Interface (RUI). The LU62 resource adapter supports the Common Programming Interface for Communications (CPI-C) programming interface.

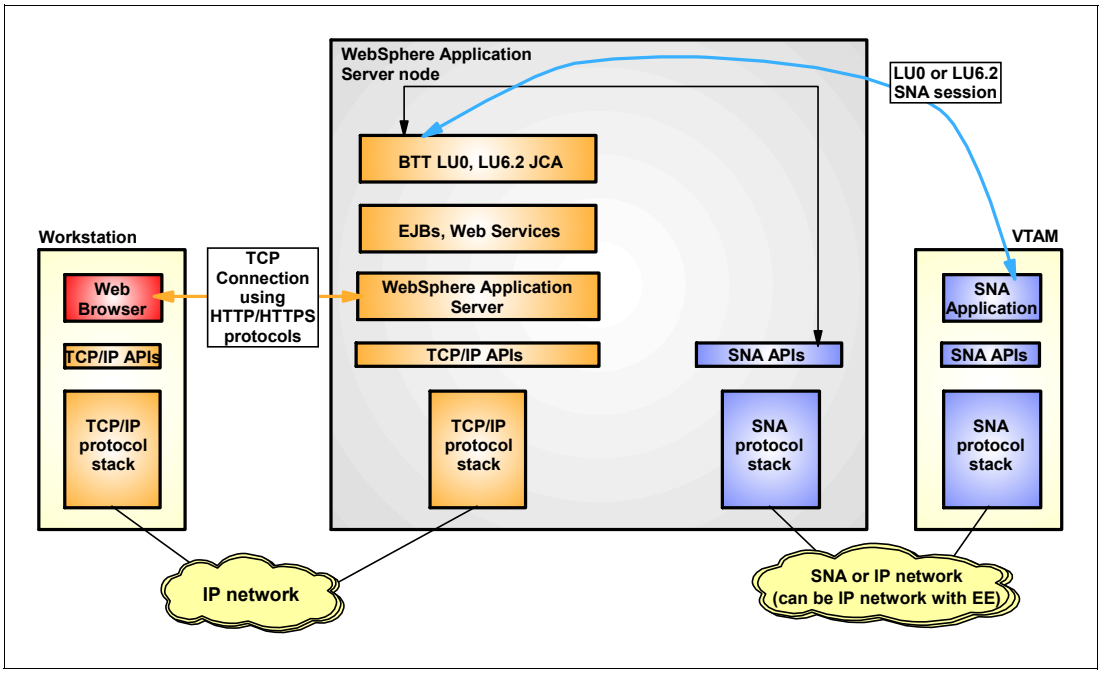


Figure 4-11 Branch Transformation Toolkit topology

Using BTT to integrate an existing SNA LU0 or LU6.2 server application as a Web service requires that a developer using the development tools creates a small piece of business logic that exploits the SNA J2C adapters and presents a Web service interface to Web service clients.

When using the SNA J2C resource adapters, BTT relies on an SNA protocol stack to provide the common SNA network services and programming interface support. If BTT is deployed on a Linux platform, this support is provided by IBM Communications Server for Linux.

The BTT runtime may be used in conjunction with IBM WebSphere Application Server, IBM WebSphere Application Server Network Deployment, or with IBM WebSphere Business Integration Server Foundation. The SNA J2C resource adapters of BTT are only supported by BTT when BTT is running on Windows, AIX, and Linux on Intel. Note that these SNA adapters are not supported by any System z operating system environment.

Note: You can use BTT to provide a Web service interface to SNA LU0 and LU6.2 server applications, independent of where these applications are running, for example in CICS, in IMS, in APPC/MVS™, and so on.

Refer to the following Web site for more details on the BTT technology:

http://www.ibm.com/software/awdtools/studiobranchtransformation/

CICS SNA applications as Web services - using CICS SFF

CICS Transaction Server for z/OS V3.1 provides a set of integrated functions that allow access to a variety of existing CICS applications, including CICS SNA 3270 applications, as Web services.

The CICS Transaction Server 3.1 Service Flow Feature (SFF) is a business service integration adapter for all CICS applications. It offers both tooling and run-time components. These components enable developers to create CICS business services for integration in service-oriented architectures (SOA), business process collaborations, or enterprise solutions that exploit a loose coupling approach.

The CICS Service Flow Feature delivers the capability to implement CICS business services by composing a sequence of CICS application interactions. The CICS Service Flow Feature delivers:

- A graphical modeling integrated development environment that enables the creation of CICS business services by composing a flow of CICS application interactions. This graphical development tooling is provided by IBM Service Flow Modeler (SFM), which is integrated into WebSphere Developer for zSeries (WDz).
- A generation capability that transforms the composed flow of CICS application interactions to form a runtime application, highly optimized for the CICS environment that retains the inherent qualities of service that the existing CICS application implementation provides.
- A runtime component, known as the Service Flow Runtime, that extends the CICS Transaction Server environment. It offers adapters that exploit CICS interfaces to invoke the CICS terminal-oriented transactions and communication area (COMMAREA) programs, as required by the service flow.

CICS business services built with and hosted in CICS Service Flow Feature, expose business function interfaces that can readily be published as Web services in a service-oriented architecture by exploiting the Web services capabilities of CICS Transaction Server V3.1.

Additionally, these business services can be integrated as process steps in a business process, orchestrated by a business process engine such as WebSphere Process Server.

CICS application interfaces, including 3270 transaction screens and COMMAREAs, can be imported as components into the composition workspace. Imported CICS application components are composed to produce a CICS business service adapter that extends the CICS application components with a business service interface.

CICS SFF supports SOAP over both HTTP(S) and WebSphere MQ.

See US Announcement letter 205-303 for more details on the CICS Service Flow Feature. This announcement letter can be accessed at:

http://www.ibm.com/isource/cgi-bin/goto?it=usa_annred&on=202-303

IMS SNA applications as Web services - using IMS integration solutions

Similar to “IMS-specific SNA 3270 presentation integration” on page 76, IMS provides MFS Web Services tooling and runtime support:

- The development tool

  MFS Web Services tooling is a GUI wizard integrated into WebSphere Studio Application Developer Integration Edition (WSADIE). It guides the user through parsing of existing MFS source files and creation of the MFS Web Service application, with data fields from MFS MID as the input types and data fields from MFS MOD as the output types.

- The runtime adapter

  The MFS Web Service application runs on IBM WebSphere Application Server on AIX, Windows, or z/OS. The MFS runtime adapter is packaged as part of IMS Connector for Java, transforming input data into application data structures as it is passed to the MFS-based IMS transaction. On return, the MFS runtime adapter dynamically determines the correct output type to populate the transformed data, based on the MAPNAME returned from the IMS application.

For non-MFS-based IMS transactions, Web service applications can be created using WSADIE, RAD, and WebSphere Integration Developer (WID).

IMS also supports a Web service technology that is referred to as the IMS SOAP gateway. The IMS SOAP gateway provides a “lightweight” Web service support for IMS that does not require the use of WebSphere Application Server. We do not discuss it here in the context of SNA modernization. If you have an IMS installation, you need to be aware that the IMS SOAP gateway technology exists.

5

Selected SNA modernization scenarios

The following network infrastructure scenarios were chosen as representative of many real SNA environments.

The scenarios may look somewhat complex. In creating these scenarios, our aim was to cover as many aspects with as few scenarios as possible. Note that despite this objective, the scenarios do not cover every possible aspect in terms of platforms, connectivity, and placement of functions.

The numbered notes following each scenario correspond to the numbers in the associated diagrams.

5.1 Branch access to z/OS, z/VM, or z/VSE: SNA subarea environment

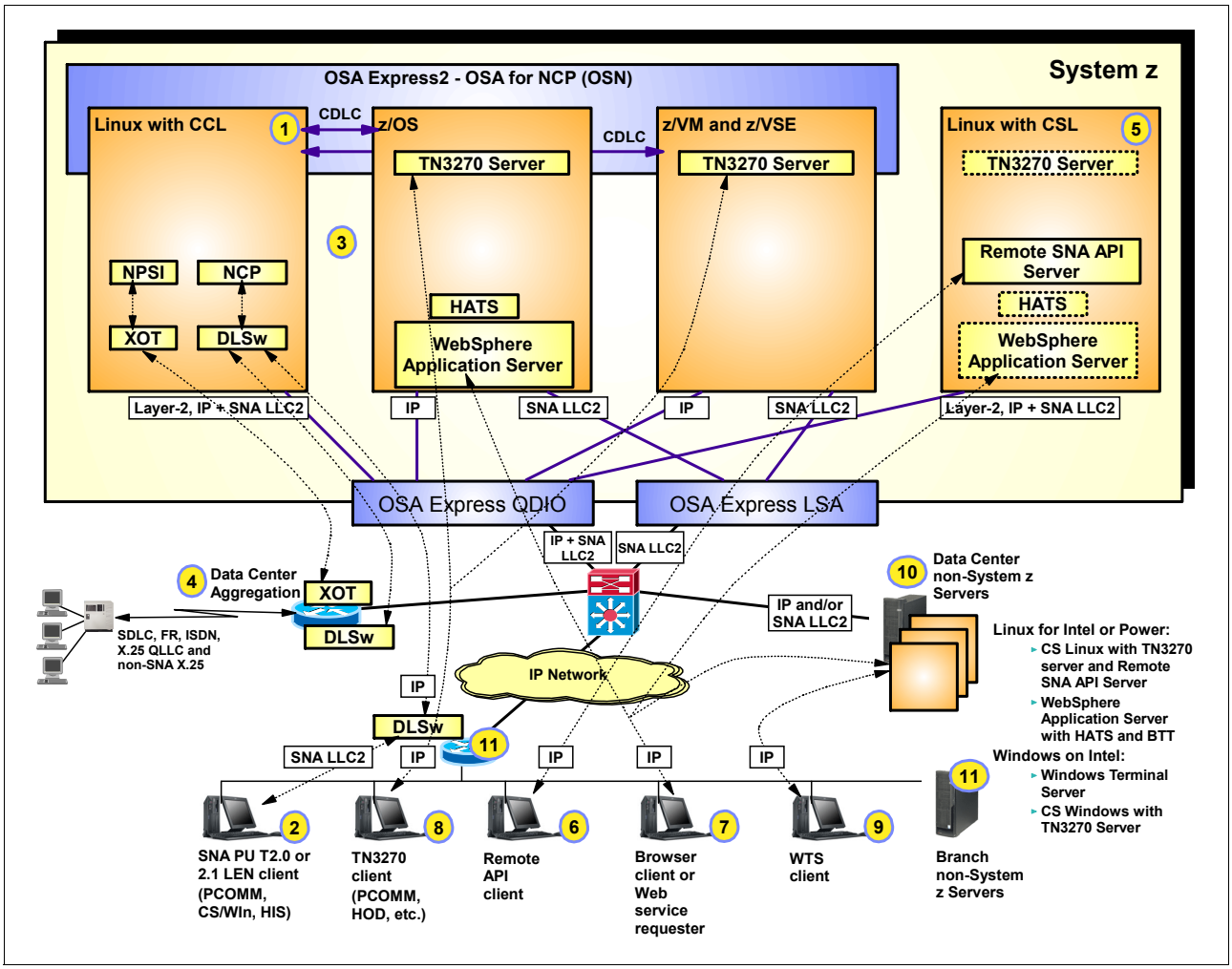


Figure 5-1 Branch access to z/OS, z/VM, and z/VSE data center using SNA subarea technologies only

This scenario depicts a setup where only SNA subarea technologies are used. VTAM on z/OS, z/VSE, and z/VM are all subarea nodes only, having no APPN capabilities enabled. It is not necessarily a recommended environment, however, we do include it here for informational purposes and to show that the overall SNA modernization objectives can be achieved in an SNA subarea environment.

Notes

1. An NCP is deployed in Linux on System z for SNA boundary functions, potentially along with NPSI for non-SNA X.25 access.
2. An SNA PU Type 2.0 or 2.1 LEN peripheral node in the branch connects to the NCP via a DLSw branch router and CCLs imbedded DLSw component.

3. The Linux CCL system is connected to VTAM in z/OS, z/VSE, or z/VM using one of two technologies:
   - A shared OSA Express2 port configured in OSN mode. If VTAM and CCL are on the same System z9 server, this is the preferred connectivity option.
   - A shared LAN to which Linux is connected via an OSA port in LCS or QDIO Layer-2 mode and VTAM via an OSA port in LSA mode.
4. If any remote locations remain connected to the data center through serial lines, they are terminated in a data center aggregation layer router that is connected to the data center LAN environment as follows:
   - SNA traffic between serial lines, such as SDLC, frame relay, or X.25 QLLC and CCL is exchanged between the aggregation layer router and CCL as either SNA LLC2 over a shared LAN or as DLSw traffic over an IP network.
   - X.25 traffic is sent over an intermediate IP network using XOT in the aggregation layer router and the Linux image in which CCL runs with both NPSI and an NCP.
5. CS Linux is deployed in a Linux on System z image (LPAR or z/VM guest) for optional SNA/IP integration functions in the SNA subarea environment. CS Linux can be defined to VTAM (or the NCP) as a peripheral node PU Type 2.0 or 2.1 LEN, if connected over a shared LAN (a LAN using SNA LLC2 frames). Linux may be connected to the LAN via an OSA port in LCS mode or in QDIO Layer-2 mode.

   Note that the CCL NCP can perform SNA boundary functions for the peripheral CS Linux SNA node if CCL is also connected to the shared SNA LAN using either an OSA port in LCS mode or QDIO Layer-2 mode. Whether you want VTAM or the NCP to perform the boundary functions for CS Linux in this setup is up to you and should be governed by the same considerations as mentioned earlier when using VTAM for boundary functions: element address scalability, general availability requirements, and VTAM CPU usage.

   Also note that when CS Linux is defined as a peripheral node to the NCP or VTAM, you need one subarea link station per PU and each PU supports up to 255 LUs. If you need more than 255 LUs, then define more PUs and link stations. Multiple CS Linux link stations can share a single local OSA MAC address as long as you configure individual local Service Access Point (SAP) numbers for each link station.
6. An SNA client program on a user workstation or branch server connects via remote SNA API to the remote SNA API server in CS Linux.
7. A browser or a Web service requester can use the WebSphere Application Server with HATS and BTT. Note that WebSphere Application Server with HATS could also have been deployed on z/OS directly and that WebSphere Application Server with BTT cannot be deployed on System z, but must be deployed on a non-System z server node in the branch or in the data center.
8. TN3270 clients connect directly to the mainframe operating system TN3270 servers. Note that it is possible to deploy the TN3270 server in CS Linux and use the peripheral node access to VTAM or the NCP for the 3270 SNA sessions. Do keep scalability in mind in such a setup - 255 LUs per PU (per link station) in CS Linux.
9. A Windows Remote Desktop client connects to a WTS server in the data center. From the WTS server in the data center, TN3270, Remote API, or HTTP protocols over IP are used to System z.

   A WTS server could alternatively be deployed in the branch. Such a server would then again use one or more of the other SNA application access modernization technologies for access into the data center over IP.
10. As an alternative to deploying the SNA application access servers in Linux on System z, they could also be deployed on a non-System z server in the data center. Clients in the

branch would connect over IP, and the server would be connected to System z over a data center LAN using either SNA LLC2 or IP traffic.

If you use such a non-System z server, be aware of the scalability, performance, and availability characteristics of such a non-System z solution.

A non-System z data center server could be:

- A System x™ server with Linux on Intel and IBM Communications Server for Linux on Intel
- A System x server with Windows and either Microsoft Host Integration Server or IBM Communications Server for Windows
- A System p™ server with Linux for Power and IBM Communications server for Linux on Power
- A System p server with AIX and IBM Communications Server for AIX

11. When a server is deployed in the branch, clients will connect to a local branch server for the services. Any of the client options can be used to connect to the branch server. The branch server will use an SNA gateway in support of SNA PU Type 2.0 or 2.1 LEN clients. The TN3270 Server will support TN3270 clients. APPN Network Node or Branch Extender Node will support APPN clients. Remote SNA API Server technology will allow connection for Remote SNA API Clients and so on. The branch server may communicate upstream using Remote SNA API client/server technology, Enterprise Extender, or SNA LLC2 connecting through a DLSw router.

5.2 Branch access to z/OS - APPN/HPR environment

The scenario illustrated in Figure 5-2 is based on use of APPN/HPR technologies with direct Enterprise Extender access to a z/OS environment. A follow-on scenario will extend such a setup to a z/VM and z/VSE environment (only VTAM in z/OS supports direct EE connectivity).

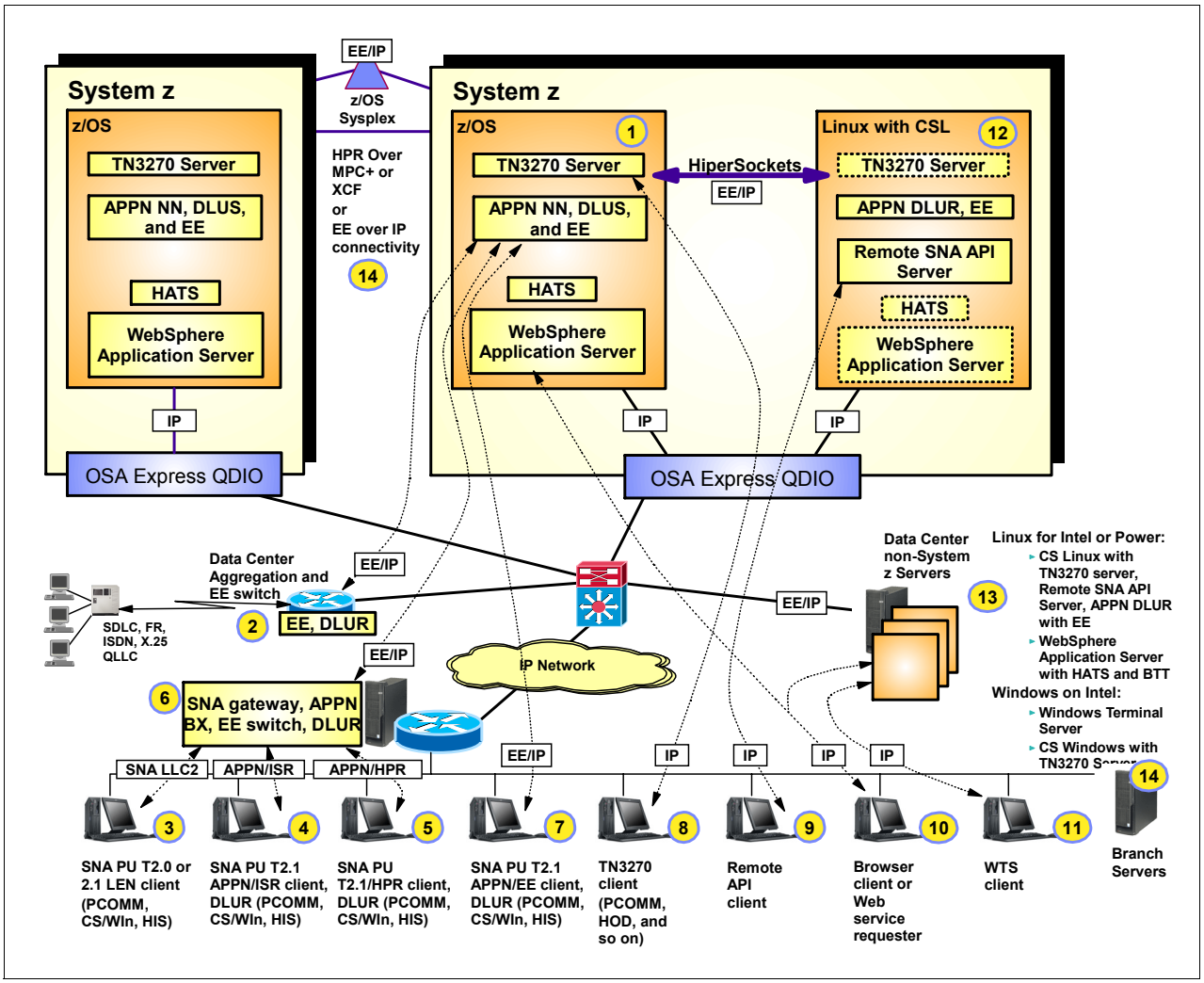


Figure 5-2 Branch access to z/OS based on APPN/HPR and EE technologies

Notes

1. VTAM on z/OS is enabled for APPN/HPR and Enterprise Extender.
2. If any remote locations remain connected to the data center via serial lines, these are terminated in a data center or branch aggregation layer router that uses DLUR and Enterprise Extender upstream and is connected to VTAM over an IP network.
   - SNA traffic between serial lines, such as SDLC, frame relay, or X.25 QLLC and CS z/OS is exchanged between the aggregation layer router and z/OS as Enterprise Extender traffic over an IP network.
   - An example of an aggregation layer router in this context is a Cisco router with the Cisco SNA Switch function enabled.

3. An SNA PU Type 2.0 or 2.1 LEN peripheral node in the branch connects to VTAM via an Enterprise Extender branch server or a branch router that performs SNA boundary functions and Dependent LU Requester (DLUR).
4. An SNA PU Type 2.1 APPN end node in the branch routes via APPN/ISR to a branch server or a branch router that uses APPN/HPR over IP (Enterprise Extender) to z/OS VTAM.
5. An SNA PU Type 2.1 HPR client in the branch routes via APPN/HPR to a branch server or router over the LAN. The branch server or router acts as an HPR ANR router and uses Enterprise Extender to connect to VTAM directly via an IP network.
6. An SNA PU Type 2.1 branch server or a branch router is configured as an SNA Gateway and Branch Extender node and connects to VTAM via an IP network using HPR over IP.
7. An SNA PU Type 2.1 HPR over IP (EE) client in the branch connects to VTAM directly using Enterprise Extender to connect to VTAM via an IP network.
8. TN3270 clients connect directly to the mainframe operating system TN3270 servers. Note that it would be possible to deploy the TN3270 server in CS Linux.
9. An SNA client program on a user workstation connects to CS Linux on System z via remote SNA API to the remote SNA API server in CS Linux on System z.
10. A browser can access the WebSphere Application Server with HATS on Linux. Note that WebSphere Application Server with HATS could also have been deployed on z/OS directly. It does not necessarily have to run on Linux on the System z platform.
11. A Windows Remote Desktop client connects to a WTS server in the data center. From the WTS server in the data center, TN3270, Remote API, or HTTP protocols over IP are used to System z.

    A WTS server could alternatively be deployed in the branch. Such a server would then again use one or more of the other SNA application access modernization technologies for access into the data center over IP.
12. CS Linux is deployed in a Linux on System z image (LPAR or z/VM guest) for SNA/IP integration functions in the SNA APPN/HPR environment. CS Linux can be defined to VTAM as an APPN Node including an APPN node that supports Enterprise Extender. Connectivity to z/OS VTAM may be based on Enterprise Extender traffic over HiperSockets or a shared LAN.
13. As an alternative to deploying the SNA application access servers in Linux on System z, they could also be deployed on a non-System z server in the data center. Clients in the branch would connect over IP and the server would be connected to System z over a data center LAN using either SNA LLC2 or IP traffic.

    If you use such a non-System z server, be aware of the scalability, performance, and availability characteristics of such a non-System z solution.

    A non-System z data center server could be:

    – A System x server with Linux on Intel and IBM Communications Server for Linux on Intel
    – A System x server with Windows and either Microsoft's Host Integration Server or IBM Communications Server for Windows
    – A System p server with Linux for Power and IBM Communications server for Linux on Power
    – A System p server with AIX and IBM Communications Server for AIX
14. When a server is deployed in the branch, clients will connect to a local branch server for the services. You can use any of the client options to connect to the branch server. The TN3270 Server supports TN3270 clients. APPN Network Node or Branch Extender Node

supports APPN clients. Remote SNA API Server technology allows connection for Remote SNA API clients etc. The branch server may communicate upstream using Remote SNA API client/server technology or Enterprise Extender.

5.3 Branch access to z/VSE or z/VM - APPN/HPR environment

The scenario in Figure 5-3 depicts a setup where CS Linux provides the same SNA NETID EE Gateway for z/VM and z/VSE to allow customers to modernize their networking infrastructure.

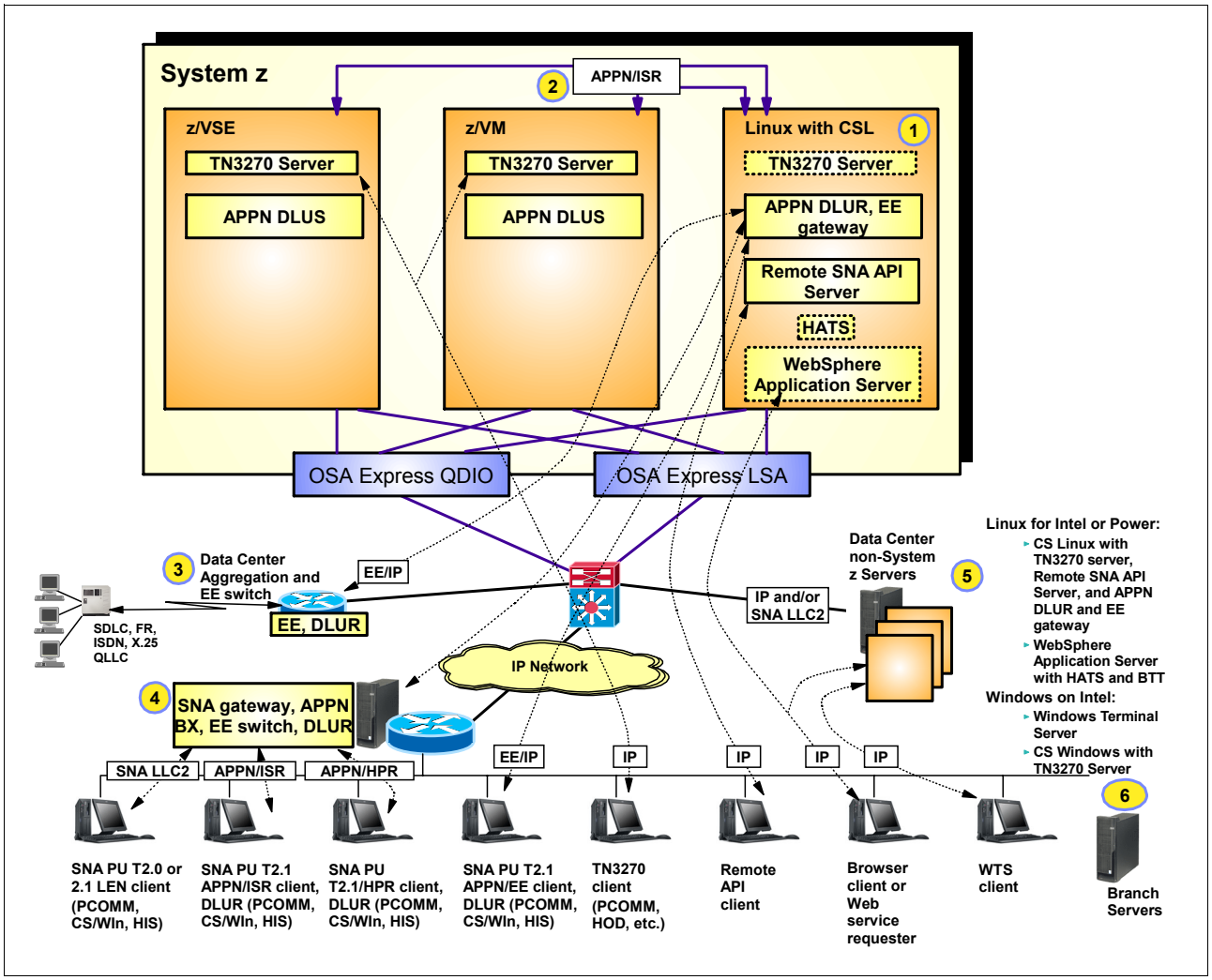


Figure 5-3 Branch access to z/VSE and z/VM based on APPN/HPR and EE technologies

Notes

1. CS Linux is deployed in a Linux on System z image (LPAR or z/VM guest) for SNA/IP integration functions in the SNA APPN environment. CS Linux is defined to z/VM or z/VSE as a Network Node using ISR routing over a shared LAN or a channel operating in Multi Path Channel (MPC) mode.

   You can configure CS Linux as a remote SNA API server to support remote SNA API clients. You can also set it up as a DLUR APPN node to connect to z/VM or z/VSE.

Additionally, you can configure it as an APPN HPR Branch Extender or Network Node to provide EE Gateway server functions for EE clients.

2. z/VM or z/VSE is configured as an APPN Network Node and as a DLUS to provide support for DLUR nodes. z/VM and z/VSE TN3270E servers optionally provide support for dependent LU-based SNA 3270 sessions.
3. If any remote locations remain connected to the data center via serial lines, these terminate in a data center aggregation layer router that uses Enterprise Extender upstream to connect to CS Linux as an EE Gateway.

   SNA traffic between serial lines, such as SDLC, frame relay, or X.25 QLLC and CS z/OS is exchanged between the aggregation layer router as Enterprise Extender traffic over an IP network to CS Linux as an EE Gateway.
4. An SNA PU Type 2.1 branch server or a branch router is configured as an EE node and connects to CS Linux as an EE Gateway.
   a. SNA traffic between serial lines, such as SDLC, frame relay, or X.25 QLLC and CCL is exchanged between the aggregation layer router or a branch server as an Enterprise Extender concentrator and uses the IP network to connect to CS Linux EE Gateway.
   b. PU Type 2.1 LEN devices go through a branch server or through a DLUR-based PU concentrator on a branch router.
   c. PU Type 2.1 APPN end node devices connect to a branch server or a branch router, which uses intermediate session routing (ISR) as a Branch Extender node and Enterprise Extender to connect to CS Linux as an EE Gateway.
5. A Windows Remote Desktop client connects to a WTS server in the data center. From the WTS server in the data center, TN3270, Remote API, or HTTP protocols over IP are used to System z.

   A WTS or X-Windows server could alternatively be deployed in the branch. Such a server would then again use one or more of the other SNA application access modernization technologies for access into the data center over IP.

   As an alternative to deploying the SNA application access servers in Linux on System z, they could also be deployed on a non-System z server in the data center. The branch server connects to z/VM or z/VSE via LLC2 or to CS Linux on System z as EE Gateway.
6. When a server is deployed in the branch, clients will connect to a local branch server for the services. Any of the client options can be used to connect to the branch server. The TN3270 Server supports TN3270 clients. APPN Network Node or Branch Extender Node supports APPN clients. Remote SNA API Server technology allows connection for Remote SNA API clients and so on. The branch server may communicate upstream using Remote SNA API client/server technology or Enterprise Extender.

5.4 z/TPF SNA connectivity

The scenario in Figure 5-4 depicts a setup for z/TPF where CCL provides SNA and X.25 connectivity.

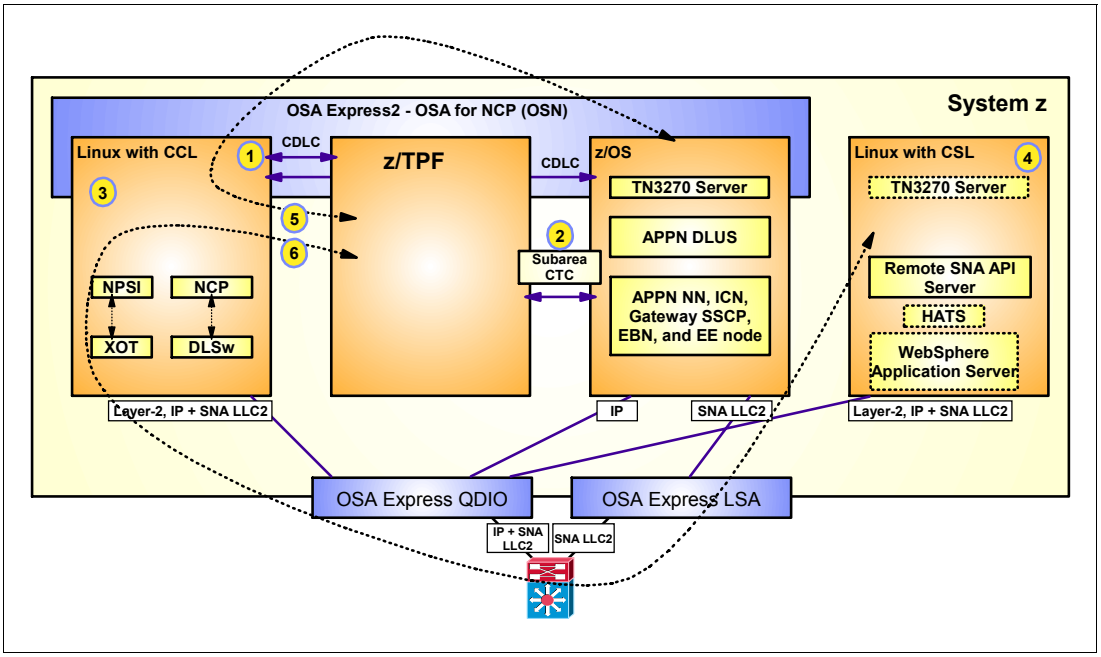


Figure 5-4 z/TPF SNA connectivity

Notes

1. z/TPF communicates with the NCP in CCL via a shared OSA-Express2 port configured in OSA for NCP (OSN) mode using CDLC channel protocols.
2. z/TPF and z/OS VTAM may optionally be connected via a subarea CTC for limited local SNA connectivity.
3. An NCP is deployed in Linux on System z for SNA boundary functions, potentially along with NPSI for non-SNA X.25 access. CCL as a DLSw server allows DLSw routers to connect to z/TPF. X.25 over TCP (XOT) provides a concentrator for SNA X.25 and non-SNA X.25 devices.
4. CS Linux may be deployed in a Linux on System z image for optional SNA/IP integration functions in support of the z/TPF environment, such as TN3270 server, remote SNA API server, and so on.
5. z/OS and z/TPF will use the shared NCP for SNA routing between them.
6. CS Linux, in this scenario, relies on the CCL NCP for SNA boundary functions and routes via a shared LAN to CCL and then to z/TPF.

5.5 SNA business partner access

SNA business partner connectivity may either be based on the SNA network interconnect (SNI) subarea functions or the Extended Border Node (EBN) functions of an APPN network; refer to Figure 5-5.

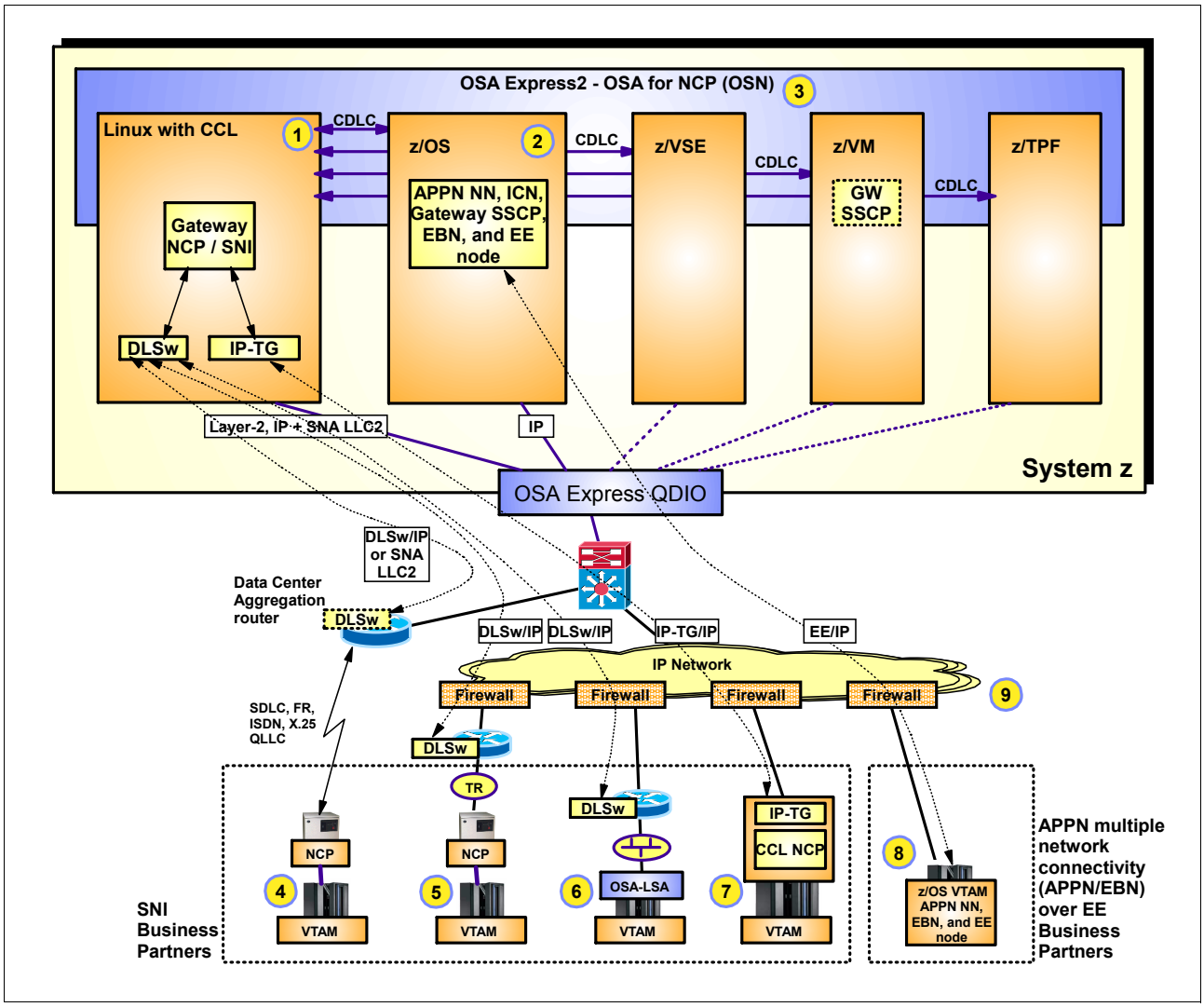


Figure 5-5 Business partner SNA communication

Notes

1. CCL on System z is used to deploy an SNI gateway NCP for SNI-based connectivity to SNA business partners. VTAM in z/OS and in z/VM can act as the matching gateway SSCP node for SNI-based connectivity.
2. Communications Server on z/OS supports EBN functions and can use Enterprise Extender for SNA business partner connectivity over an IP network.
3. A shared OSA Express2 port configured in OSN mode allows connectivity between VTAM, z/TPF, and NCP. It appears to be an ESCON channel over which the traditional CDLC channel protocol is used. VTAM in z/OS, z/VSE, z/VSE and the SNA support in z/TPF all support OSN connectivity to a CCL NCP.

4. Business partners who continue to use a serial line-attached IBM 3745/46 controller can be supported by a router in our data center with a serial line interface. The router may either switch the SNA traffic to a LAN as SNA LLC2, or it may use DLSw over an IP network into Linux on System z where our SNI NCP is deployed.
5. Other business partners who also continue to use an IBM 3745/46 controller, but replace the serial line interface with a token-ring interface, may use a DLSw router at the business partner location and exchange SNI traffic over an IP network into Linux on System z, where our SNI NCP is deployed.
6. An SNI business partner who does not have an SNI gateway NCP may still, under certain topology restrictions, be SNI-connected to our gateway NCP. Such a business partner will use an OSA-LSA adapter to connect to our gateway NCP via a DLSw router in the business partner data center to DLSw in Linux on System z, where our SNI NCP is deployed.
7. Two business partners who both have CCL deployed, can communicate directly between the two SNI gateway-NCPs using IP TG over an IP network.
8. Two business partners who both have z/OS enabled for Enterprise Extender, can communicate directly over an IP network.
9. Business partner communication normally takes place over shared or public network segments. Firewalls are often used. IP TG and DLSw both use TCP connections and are relatively simple to enable through firewall access control lists. EE uses UDP-based communication over five UDP port numbers between specific EE virtual IP addresses (VIPA) and is in some cases a little more complicated to enable through firewall access control lists.

A

SNA levels

This appendix provides a brief introduction to the three main SNA levels. We describe the main architectural differences between the SNA levels, and explain how these differences influence the choice of SNA modernization technologies.

Note that the information presented in this appendix is for introductory purposes only and is not intended to provide an in-depth understanding of SNA.

First SNA level: SNA subarea networking

When SNA was first introduced in 1974, the primary objective of a network was to allow fixed-function terminals (IBM 3270 terminal types) and programmable branch computers (IBM 3600 or IBM S/3x) online access to applications and data on the IBM mainframe.

A network, at that point in time, consisted of a few, relatively simple, building blocks:

- A mainframe computer hosted the applications and controlled the network. In SNA terminology, this is known as a System Services Control Point (SSCP). An SSCP is part of Virtual Telecommunications Access Method (VTAM).
- A channel-attached communication controller (IBM 3705) to which serial lines were attached. A communication controller was a specialized processor under the control of an SNA Network Control Program (NCP).
- Leased or dial-up serial lines between the data center and the remote locations, which typically consisted of Synchronous Data Link Control (SDLC) lines terminated by modems at both ends.
- In the remote locations, the serial lines were attached to terminal control units (IBM 3x74) to which terminals (IBM 327x) and printers (IBM 328x) were attached using coax cables. Otherwise, the serial lines were connected to branch computers, such as the IBM 3601 computer or IBM S/36, to which terminals and printers were connected using various cabling technologies.

The objectives of a networking architecture in the past were focused on providing network functions for such an environment. Because all meaningful use of such a network was tied to the mainframe, the mainframe became the central hub upon which all the remote nodes depended for communication with applications. In SNA terminology, such nodes are currently known as dependent logical units (LUs), because they depend on the networking software in the mainframe to establish communication sessions with applications.

The structure of an original SNA network was very hierarchical: VTAM with the SSCP at the top; then one or more NCPs; followed by terminal controllers or branch computers; and then the terminals or remote SNA client programs on the branch computers.

Between NCPs and VTAM, and between one VTAM and other VTAMs, alternate SNA routes could be configured, and the original SNA architecture used a concept of SNA networking subareas between which alternative routes could be selected. In SNA terminology, SNA routes are referred to as *paths*.

Between the NCP and the peripheral nodes, no alternate routes are possible. The NCP acts as the boundary between the remote locations and the “routable” SNA network, and it is therefore referred to as “performing SNA boundary functions” for such remote nodes.

Currently, this original SNA network topology is known as an SNA subarea network.

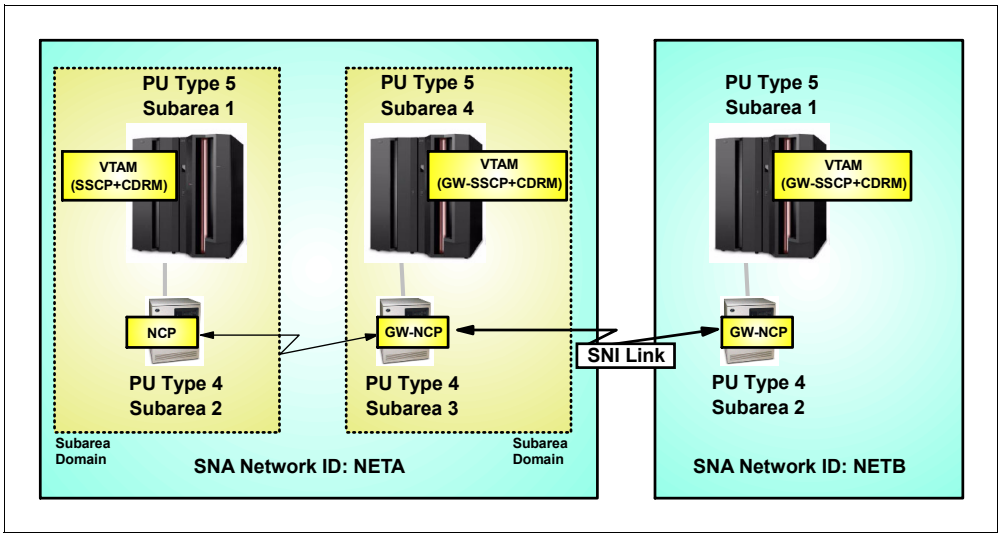


Figure A-1 SNA subarea topology

As shown in Figure A-1, an SNA subarea network consists of the following:

- Subarea nodes: VTAMs (PU Type 5) and NCPs (PU Type 4).
- Peripheral nodes: The terminal control units and branch computers are referred to as peripheral nodes (PU Type 2).
- Subarea domains: This refers to the combination of one or more NCPs and the VTAM that "owns" the NCPs. "Owning" in this context refers to activation of the NCP subarea node.
- SNA network: This refers to the combination of one or more VTAMs and one or more NCPs that constitute an administrative unit known as an SNA network that an SNA NETID identifies.
- Interconnected SNA networks: These are based on an SNA subarea-specific technology that is known as SNA Network Interconnect (SNI). SNI allows two or more SNA NETIDs to communicate between them. SNI is a combination of functions in VTAM, known as Gateway SSCP functions, and functions in an NCP, known as Gateway NCP functions.

You must predefine all resources, such as nodes, links, and paths, in an SNA subarea network (a NETID) on each subarea node in order for it to be able to establish sessions through the SNA subarea network.

You must predefine all possible session paths on all subarea nodes, known as *path tables*. Maintaining path tables in the subarea nodes in a large SNA subarea network is probably one of the most complicated tasks of administrating an SNA subarea network.

A characteristic of an SNA subarea network is that if a remote node is in session with an SNA application over a given path in the SNA network and one of the links or network interfaces that this path is constructed of fails, the SNA session fails, even if there are alternative paths available. The session can be reestablished and the SNA subarea nodes will then select a path that is available at the time of the session setup, but the path switch is not automatic and is not transparent to the user or the application.

Second SNA level: APPN

As networking technologies evolved from simple serial lines and fixed function devices into local area networks (LANs) and increasingly powerful distributed computers, a need for communication between applications on remote peer nodes emerged. For such communication, the idea that the mainframe is the “center of the universe” and that its involvement is necessary to establish communication between the SNA nodes was disproved, and a requirement for peer nodes to be more self-controlled became apparent.

The original SNA subarea topology did not meet such demands and the SNA architecture was, therefore, extended into what is currently referred to as Advanced Peer to Peer Networking (APPN).

Important: Be aware that an SNA node can be an SNA subarea node and an APPN node at the same time, having a role in both types of SNA networks.

The original APPN architecture uses a routing technology between APPN nodes that is known as Intermediate Session Routing (ISR). A characteristic of ISR is that each APPN node, through which traffic for a given SNA session flows, has awareness of that session and maintains state information about that session and how traffic for it uses the node's network interfaces.

Some of the main characteristics of an APPN network are:

- LUs do not have to predefine partner nodes. They can locate nodes dynamically, anywhere in the APPN network, using APPN directory services.
- Selection of a route, or a path in SNA terminology, can be done dynamically and will include any dynamic changes to the APPN network topology. APPN topology and routing services (TRS) select routes and conform to the requirements that the APPN Class of Service (COS) specifies.
- Traffic between two APPN nodes does not have to follow a specific hierarchy, but can traverse through one or mode intermediate APPN nodes.
- There is potential for much less system definition than in an SNA subarea network. LUs need to be defined on the owning node only, and routes are determined dynamically at session setup time.

There are two main SNA node types in an APPN network:

- An end node (EN)

  An EN will register its resources with its network node. It will not act as an intermediate router. It runs applications.

- A network node (NN)

  An NN provides all the general APPN functions, such as directory services, routing services, intermediate session routing, and so on. NNs exchange APPN control information, such as network topology updates and search requests for LUs. As with an EN, an NN may also run applications.

In an APPN network, all nodes are the same PU type. This is unlike SNA subarea, where the PU type varies, depending on the role of the SNA node. In an APPN topology they are all PU Type 2.1 nodes, and they all include what is generally referred to as their APPN Control Point (CP).

When VTAM on the mainframe is part of both a subarea network and an APPN network, VTAM actually has two PU types: the subarea SSCP is PU Type 5, while the APPN Control Point (CP) is PU Type 2.1.

LUs on APPN nodes can establish sessions with other LUs on APPN nodes without contacting VTAM, as long as they are independent LUs of type 6.2.

What if an APPN node has dependent LUs, such as an IBM 3270 terminal or an emulator that needs to establish a session with TSO on z/OS? Only independent LUs can set up sessions through the APPN network. A dependent LU needs an SSCP that is a subarea connected to the SNA node that acts as the boundary function node (in this case, the owning APPN node). For that reason, there is Dependent LU Requester (DLUR) support on the remote APPN node and a Dependent LU Server (DLUS) on VTAM.

The traditional SNA subarea boundary functions for dependent LUs may be located on any APPN node that supports the DLUR functions to establish independent LU6.2 sessions through the APPN network to a DLUS. The dependent LU-to-SSCP dialog is then carried over those DLUR to DLUS LU6.2 sessions, and the dependent LU can establish a session with whatever mainframe SNA application it needs to communicate with.

You can interconnect two SNA subarea NET IDs using the SNI technologies described previously. What if the two SNA NET IDs are APPN networks? In that case, you can use an APPN-equivalent function that is known as APPN Multiple Network Connectivity.

A Network Node in each APPN network is assigned the role of being an APPN network Border Node, generally referred to as an Extended Border Node (EBN). EBNs can implement APPN links between them, which interconnect the two APPN networks (similar to the process that SNI uses, based on NCP-to-NCP connectivity).

There are other specialized node types in an APPN network topology, but the end node and the network node are the basic building blocks upon which the other specialized node types depend, as explained here:

- Composite Network Node (CNN)

  This is a VTAM node (PU Type 5) and one or more NCP nodes (PU Type 4) that the VTAM node owns, which work together to present the appearance of a single collapsed APPN network node (PU Type 2.1).

- Extended Border Node (EBN)

  This is an APPN network node (PU Type 2.1) that can interconnect multiple APPN networks of different SNA NET IDs. Only VTAM on z/OS, z/VSE, and z/VM can act as an EBN node.

- Interchange Node (ICN)

  This is a VTAM node (PU Type 5) that is also an APPN network node (PU Type 2.1) that provides interchange functions for sessions that cross the APPN-to-subarea network boundary. In other words, this node makes it possible for an LU that resides on an APPN node to establish sessions with an LU that resides on a subarea node.

- Central Directory Server (CDS)

  A VTAM (PU Type 2.1) that is an APPN network node may be configured as a Central Directory Server. All APPN nodes maintain a local directory database with information about local LUs. End nodes also register their LUs with their network node. When an APPN node needs to locate a given LU and a CDS server is present in the APPN network, locate requests are sent to the CDS. The CDS performs additional searches and maintains a central cache where it tracks the location of the LUs that are in the APPN

network. This reduces the amount of network broadcast traffic that is required to locate LUs in the APPN network.

- Migration Data Host (MDH)

  This is a VTAM node (PU Type 5) that is also an APPN End Node (PU Type 2.1).

- Branch Extender (BX)

  This is an APPN node that provides the appearance of an APPN network node to “downstream” APPN nodes, while providing the appearance of an APPN end node to the “upstream” APPN nodes. In a large APPN network with many network nodes and with WAN links between the network nodes, the network resources used for APPN control traffic to maintain overall topology information and to locate LUs in the APPN network can be significant.

  Therefore, a BX node in a remote location helps to reduce that overhead by collapsing the APPN network node topology to the data center, while at the same time providing the APPN network node services to APPN end nodes in the remote location.

- Virtual Routing Node (VRN)

  This is not a real node. As the name suggests, it is a virtual node that represents a virtual network node. VRNs are used by a technology that is known as Connection Networks to reduce the amount of predefined SNA links that normally are required in order for two APPN nodes to communicate with each other.

  Without connection networks, if an APPN node on a multi-access shared network technology, such as a LAN (or an IP network, when Enterprise Extender is used) needs to communicate with 10 other APPN nodes on that same network, then that APPN node has to define 10 SNA links (one link for each partner node). These definitions also need to be reversed on those 10 other nodes.

  In a connection network topology, each of the 10 APPN nodes on the LAN define only two links: one to the virtual node (the VRN), and the other to an APPN Network Node or CP-CP partner. Each of the APPN nodes connected to the VRN dynamically learns about links to the other APPN nodes that are connected to the same VRN.

The APPN network technology is of special importance in the distributed environment, where SNA applications on workstations and servers in remote locations can establish sessions without involving VTAM in the data center.

APPN also provides much more freedom in setting up the actual network topology, by removing the requirement for SNA peripheral nodes to be adjacent to a subarea node (an NCP or a VTAM). APPN allows SNA session traffic between LUs to traverse a series of APPN nodes on the path between the two SNA application end points.

APPN, in combination with the DLUR/DLUS technology, allows the traditional SNA subarea LU types (dependent LUs) to gain the advantages of the more flexible APPN network topology while at the same time preserving the application functions these dependent LUs provided in the past.

A characteristic of an APPN network that uses ISR routing is similar to that of the SNA subarea network. If a remote node is in session with an SNA application over a given path in the APPN network and one of the links or network interfaces that path is made up of fails, then the SNA session fails, even if there are alternative paths available. The session can be reestablished and the APPN topology and routing services then selects a path that is available at the time of the session setup, but the path switch is not automatic and not transparent to the user or the application.

Third SNA level - HPR

High Performance Routing (HPR) replaces the way SNA data is routed through an APPN network, but without changing the overall APPN topology. In an APPN network that uses HPR routing, you have the same node types and APPN capabilities as you had in an APPN network that used ISR routing, accompanied by a much more reliable and efficient routing of the SNA data through the SNA network.

HPR implements a reliable transport layer in the SNA protocol stack. Neither of the two other SNA architecture levels had that. In previous versions of SNA, reliability was left to the data link layer and if a link failed, the sessions over that link would fail too (as was explained for both SNA subarea and APPN with ISR routing).

With HPR, that is no longer the case. If the link of an SNA session that is established fails, then when HPR is used over that link, the HPR technology will perform a nondisruptive path switch to another path if such an alternate path is available at the time of the failure of the original path.

HPR can be used over much of today's relevant SNA link types, such as over a LAN, over Multi Path Channel (MPC+) channels (both ESCON and FICON channels), or over XCF signaling inside a z/OS sysplex. In addition, HPR defines a new SNA link type as being an IP network. From an APPN/HPR topology perspective, an IP network becomes an HPR link, which is a single hop in the SNA topology. This capability is referred to as HPR over IP (HPR/IP), or more commonly known as Enterprise Extender (EE).

Not all SNA APPN nodes support HPR. Of those that do support HPR, not all support HPR over IP. The most commonly used SNA APPN nodes that support HPR over IP are:

- z/OS VTAM
- Communications Server for Linux (Intel, Power, and System z)
- Communications Server for Windows
- Communications Server for AIX
- PCOMM
- I5OS (on the i-Series servers)
- Cisco SNA Switch
- Microsoft Host Integration Server 2004

On System z, neither z/VSE nor z/VM support HPR, and therefore none of them support HPR over IP. z/TPF does support HPR, but it does *not* support HPR over IP.

Also note that if APPN multiple network connectivity is used to interconnect two APPN NET IDs, the EBN nodes may use HPR over IP between them if they both support both EBN functions and HPR over IP connectivity. z/OS VTAM *does* support both.

HPR over IP uses UDP datagrams over the IP network. HPR implements its own reliable transport layer, known as Rapid Transport Protocol (RTP). Therefore, HPR does not need the reliability of the TCP transport protocol in TCP/IP, but is perfectly fine using the less reliable services of UDP. If an IP packet is lost, RTP will detect that and retransmit the SNA data.

HPR over IP uses five standard UDP port numbers, one per SNA class of service (COS): port 12000 through port 12004. If you use HPR over IP between two different APPN NET IDs, then it is important that any firewalls between the two HPR over IP nodes allow UDP traffic to or from those five port numbers to flow in each direction.

B

SNA node capability summary

Table B-1 SNA node capability summary

Functional area		z/OS VTAM	z/VSE VTAM	z/VM VTAM	z/TPF	NCP	CS Linux	CS AIX	PCOMM	i5/OS	CS Win	Cisco SNA Switch
SNA Subarea node types	2.0						✓	✓	✓	✓	✓	
	2.1 LEN	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓
	4					✓						
	5	✓	✓	✓	✓							
SNI functions	SNI GW SSCP	✓		✓								
	SNI GW NCP					✓						
SNA LAN LLC2	OSA LSA	✓	✓	✓								
	OSA LCS					✓ (CCL)	✓ (on z)					
	OSA QDIO layer-2					✓ (CCL)	✓ (on z)					
	Other						✓	✓	✓	✓	✓	✓
Mainframe channel	Subarea channel	✓	✓	✓	✓							
	MPC host PUT4	✓	✓	✓								
	MPC LEN							✓			✓	
	MPC APPN (ISR)	✓	✓	✓			✓ (on z)	✓			✓	
	MPC+ (HPR)	✓						✓			✓	
	CDLC	✓	✓	✓	✓	✓						

Functional area		z/OS VTAM	z/VSE VTAM	z/VM VTAM	z/TPF	NCP	CS Linux	CS AIX	PCOMM	i5/OS	CS Win	Cisco SNA Switch
APPN node types	NN	✓	✓	✓			✓	✓		✓	✓	
	EN	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	BX						✓	✓		✓	✓	✓
Additional APPN functions	MDH	✓	✓	✓	(✓)							
	ICN	✓	✓	✓								
	EBN	✓	✓	✓								
	DLUR						✓	✓	✓	✓	✓	✓
	DLUS	✓	✓	✓								
	CDS	✓	✓	✓								
APPN HPR	RTP endpoint	✓			✓		✓	✓	✓	✓	✓	✓
	ANR router	✓				✓	✓	✓		✓	✓	✓
	HPR over IP (EE)	✓					✓	✓	✓	✓	✓	✓
SNA/IP integration technologies	TN3270 server	✓	✓	✓			✓	✓		✓	✓	
	SNA gateway						✓	✓			✓	
	EE to APPN/ISR gateway	✓					✓	✓		✓	✓	✓
	DLSw					✓ (CCL)						✓ (IOS)
Remote API Server	HTTP(S)						✓	✓				
	Native TCP						✓	✓			✓	
Remote API Clients	Win32						✓	✓			✓	
	Win64						✓	✓				
	Linux32						✓	✓				
	Linux64						✓	✓				
	AIX						✓	✓				

C

TN3270 server capability summary

Table C-1 TN3270 server capability summary

Functional area of interest	CS Linux on System z	CS z/OS
Multiple TN3270 server ports per OS image	Yes, multiple ports can be defined	Yes, multiple ports (255) per server instance (8 instances per z/OS LPAR)
LU assignment rules per port	Shared among all ports	Can be shared inside a server instance or separate per port
LU name assignment based on client IP address	Yes	Yes
LU name assignment based on client host name	Yes	Yes
LU name assignment based on server IP address	No	Yes
LU name assignment based on server link name over which connection was received	No	Yes
LU name assignment based user ID (if SSL/TLS with client authentication is used)	No	Yes
Secure connections	Yes (SSL connections)	Yes (SSL or TLS connections)
Secure connections with client authentication	Yes (signature verification of certificate signer)	Yes (signature verification with optional SAF authentication and port protection)
Support for ELF (Express Logon Feature)	Yes (via z/OS DCAS server)	Yes
TN3270E support	Yes	Yes
TN3270E contention support	Yes	Yes

Functional area of interest	CS Linux on System z	CS z/OS
Printer association	Yes	Yes
Specific LU name request - both specific LU and specific LU group	Yes	Yes
Support for user exit routine to assign LU names	No	Yes
ANS=CONT support	Yes	No
Capacity per server instance	Tests done with up to 20,000 connections	Tests done with up to 128,000 connections
Built-in response time monitoring	No	Yes - SNA, IP, and full round-trip response time
SNA session re-connect support	No	Yes - for both generic and specific LU name assignments
Telnet re-director support	Yes - including SSL offload to redirector with non-SSL redirection	No
How is initial SNA application chosen?	N/A - done via traditional VTAM definitions (LOGAPPL)	Can be assigned based on client IP address, host name, server IP address, link name, or user ID
Can TN3270 server perform access authorization to SNA application?	No	Yes - via assignment rules based on certificate-derived user ID, or via user ID derived through use of built-in network solicitor function
USS table support	N/A - uses standard VTAM SSCP USS table processing	Controlled by TN3270 server - VTAM USS table can be used as is, or TN3270 server-specific versions can be used
Connection load-balancing support	Traditional load balancing	Sysplex Distributor or traditional load balancing with SASP support
Server identity takeover	Manual or automated operations to move IP address to another Linux OS image	Sysplex dynamic VIPA policies to move IP address to another z/OS image in the sysplex

D

IBM Distributed Communications Server - functional overview

Table D-1 Distributed Communications Server - functional overview

Functional area	Functions	CS Windows	CS Linux on Intel	CS Linux On Power	CS Linux on System z	CS AIX
Security features	SSL data encryption for TN3270 server	✓	✓	✓	✓	✓
	Server authentication	✓	✓	✓	✓	✓
	Client authentication	✓	✓	✓	✓	✓
	SNA session level encryption	✓				✓
	Remote API client/server data encryption	✓	✓	✓	✓	✓
Application programming interfaces	LUA (RUI)	✓	✓	✓	✓	✓
	LUA (SLI)	✓	✓	✓	✓	✓
	CPI-C for Java	✓	✓	✓	✓	✓
	APPC	✓	✓	✓	✓	✓
Administration	Web-based administration	✓	✓	✓	✓	✓
	Tivoli® enabled and Tivoli Ready	✓				
	Load balancing for TN3270 server	✓	✓	✓	✓	✓
	High Availability/Hot standby	✓	✓	✓		✓
	Express logon	✓	✓	✓	✓	✓

Functional area	Functions	CS Windows	CS Linux on Intel	CS Linux On Power	CS Linux on System z	CS AIX
SNA support	SNA gateway for dependent LUs (LU0,1,2,3, and dependent LU6.2)	✓	✓	✓	✓	✓
	APPN end node (EN)	✓	✓	✓	✓	✓
	APPN branch extender node (BX)	✓	✓	✓	✓	✓
	APPN network node (NN)	✓	✓	✓	✓	✓
	APPN Connection Network	✓	✓	✓	✓	✓
	HPR intermediate node	✓	✓	✓	✓	✓
	HPR connection endpoint	✓	✓	✓	✓	✓
	Dependent LU over APPN (DLUR)	✓	✓	✓	✓	✓
	Data compression	✓	✓	✓	✓	✓
	LU 6.2 sync point	✓	✓	✓	✓	✓
Multi-protocol support	TN3270E server	✓	✓	✓	✓	✓
	TN5250 server	✓				
	Enterprise Extender (HPR only)	✓	✓	✓	✓	✓
	Wide area network connectivity - SDLC	✓	✓			✓
	Wide area network connectivity - X.25	✓	✓			✓
	LLC2 LAN (Ethernet and Token-ring)	✓	✓	✓	✓	✓
	S/390® channel connectivity	✓			✓	✓
	Multi-channel support (MPC)	✓				✓
	SNA over TCP/IP access node	✓				
Remote SNA API clients for desktops or branch servers - supported by the remote SNA API services	Remote API client (SNA over TCP)	✓	✓	✓	✓	✓
	Remote API client (SNA over HTTP(S))		✓	✓	✓	✓
	Windows remote API client	✓				
	Linux remote API client		✓	✓	✓	✓
	AIX remote API client		✓	✓	✓	✓

E

IBM Communication Controller for Linux - functional overview

The tables in this appendix summarize which functions of the IBM Communication Controllers (IBM 3705, 3720, 3725, 3745, and 3746) and associated software components are supported in a Communication Controller for Linux on System z (CCL) Version 1 Release 2.1 environment.

You can also refer to the following documentation for further information about this topic:

- *Communication Controller for Linux on System z Implementation and User's Guide*, SC31-6872, offers more information about CCL Release 2.1 and the functions it supports.
- *IBM Communication Controller for Linux on System z V1.2.1 Implementation Guide*, SG24-7223, discusses how to plan for and implement CCL Release 2.1.
- *IBM Communication Controller Migration Guide*, SG24-6298, contains details about IBM 3745/46 migration in general, and identifies alternative technologies for many of the functions that are not supported in a CCL environment.

Note that since CCL executes on System z hardware, it cannot directly support attachment of any serial lines. Indirect attachment of selected types of serial lines is supported via aggregation layer routers.

An *aggregation layer router* is a traditional router with serial line interfaces. The aggregation layer router uses router-specific technologies to switch or bridge the SNA traffic between the supported serial lines and SNA LLC2 traffic on a LAN. The CCL NCP is attached to this network through an OSA adapter operating in either LCS mode or in QDIO layer-2 mode. An example of an aggregation layer router is a Cisco 3700, 1800, 2800, 3800, or 7200 family router. Other router vendors support similar technologies.

CCL R2.1 also supports non-SNA X.25 traffic via an aggregation layer router, where X.25 circuits are terminated in the router and the router exchanges the X.25 packets with Linux on System z using the X.25 over TCP (XOT) protocol. XOT support on Linux for System z is not part of CCL, but is offered as a separate product offering.

CCL R2.1 supports DLSw connectivity into Linux on System z, allowing a downstream DLSw router to use IP connectivity all the way into Linux on System z, where the DLSw TCP connections are terminated and the SNA traffic is switched to an internal SNA LLC2 connectivity to CCL NCP.

In addition, CCL R2.1 supports an optimized direct TCP/IP connectivity between two CCL NCPs for INN or SNI traffic. Such connectivity is referred to as IP Transmission Group (IP TG).

The tables list functions which are supported by CCL and those which are not:

- Table E-1 on page 111 provides a high level functional overview of supported and non-supported software components and connectivity options.
- Table E-2 on page 111 provides a detailed functional breakdown of supported functions and options.
- Table E-3 on page 116 provides an overview of compatible software releases in a CCL environment.

Table E-1 High level functional overview

CCL R2.1 supports	CCL R2.1 support of serial lines via an aggregation layer router	CCL R2.1 does *not* support
Software: ► NCP (V7R5 or later) and compatible levels of NRF ► NPSI (V3R8 or later) ► SSP, NTuneMON, NetView®, and NPM continue to work as they have in the past		Software: ► Other IBM 3745 software products: XI/NSF, EP, NTO, NSI, MERVA, or TPNS ► Functions provided by the IBM 3746 NNP and MAE ► NCP-based IP routing
Physical network interfaces: ► SNA LLC2 over 4/16/100 Mb token-ring and 10/100/1000 Mb Ethernet LAN attached through OSA Express ports operating in LCS mode (copper-based cabling) ► SNA LLC2 over 1000BASE-T, Gigabit, and 10 Gigabit Ethernet OSA Express ports on z890, z990, and system z9 - operating in QDIO layer-2 mode (copper or fiber-based cabling) ► SNA CDLC channel connectivity on System z9 to same-CEC SNA operating systems (z/OS, z/VM, z/VSE, and z/TPF) through a 1000BASE-T or Gigabit Ethernet feature on OSA-Express2 - operating in OSA for NCP (OSN) mode ► IP-based connectivity for IP Transmission Group INN and SNI communication between CCL NCPs ► IP-based connectivity for Data Link Switching (DLSw) to a downstream DLSw node, such as a remote router that is connected over an IP WAN to the data center ► IP-based connectivity for NPSI non-SNA X.25 traffic via X.25 over TCP (XOT) to an aggregation layer router where X.25 circuits are terminated.	Physical network interfaces: ► SDLC, Frame Relay, X.25 QLLC, and ISDN serial line interfaces are not supported directly by CCL, but are supported via an aggregation layer router ► X.25 circuits for non-SNA X.25 traffic to/from NPSI are not supported directly by CCL, but are via an aggregation layer router that uses the XOT protocol to transport the X.25 packets to/from an XOT protocol component on Linux interfacing to NPSI running in CCL	Physical network interfaces: ► BSC, ALC, Start/Stop

Table E-2 Detailed functional support information

Communication Controller physical network interfaces	Directly supported by CCL R2.1	Supported via an Aggregation Layer router	*Not* supported by CCL R2.1	Comments
Communication lines		Some	Some	System z hardware does not support direct communication line attachment to CCL. Some serial lines can be terminated in an aggregation layer router and SNA data switched to a LAN to which CCL is attached using an OSA adapter.
Token-Ring LAN copper cabling	✓			4/16/100 Mb token-ring

Ethernet LAN copper cabling	✓			10/100/1000 Mb Ethernet (1000BASE-T)
Ethernet LAN Fiber optic cabling	✓			Gigabit and 10 Gigabit Ethernet
3745 TIC2 adapter support	✓			TIC2 LINE addresses and mode of operation
3746 TIC3 adapter support	✓			TIC3 LINE addresses and mode of operation
FDDI			x	FDDI is supported by the MAE, not the NCP
ATM			x	ATM is supported by the MAE, not the NCP
Channel attachment	✓			OSA for NCP (OSN) on System z9
NCP link-level protocol functions	**Directly Supported by CCL R2.1**	**Supported via an Aggregation Layer router**	**Not supported by CCL**	**Comments**
Air Lines Control (ALC)			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
SNA BSC lines for access to SNA 3270 applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
Frame Relay		✓		Works via an aggregation layer router Has been tested for both peripheral and subarea links.
Integrated Services Digital Network (ISDN)		(✓)		Is expected to work via an aggregation layer router, but has not yet been tested.
X.21		(✓)		Is expected to work via an aggregation layer router, but has not yet been tested.
Token-Ring LAN and Ethernet LAN - both TIC2 and TIC3 NCP LINE addresses	✓			Supported by CCL using an OSA port operating in LCS mode (copper cabling only), or QDIO layer-2 mode (copper or fiber cabling).
Start/Stop lines connected to TCAM			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
Synchronous Data Link Control (SDLC)		✓		Works with an aggregation layer router. Has been tested for both peripheral and subarea links.
X.25 SNA QLLC (with licensed support feature on IBM 3746)		✓		Is supported without NPSI and works with an aggregation layer router. Has been tested for both peripheral and subarea links.

CCL-unique NCP connectivity functions	**Directly Supported by CCL R2.1**			
IP-based connectivity via integrated DLSw support in CCL to local/remote DLSw node	✓			DLSw connectivity is supported for both subarea links and boundary function links.
IP-based connectivity via IP Transmission Group to partner CCL NCP for INN or SNI traffic	✓			IP TG connectivity is supported for subarea links to NCPs that also run in CCL R2 or higher.
NCP advanced functions	**Directly supported by CCL R2.1**	**Supported via an aggregation layer router**	***Not* supported by CCL R2.1**	**Comments**
SNA Class of Service (COS)	✓			
Multi-Link Transmission Group (MLTG)	✓			MLTG over multiple LAN adapters is supported. MLTG is not supported by DLSw technology.
SNA subarea addressing, routing, and boundary functions (BF)	✓			
SNA Network Interconnect	✓			
APPN (and LEN) Composite Network Node (CNN)	✓			CNN is an APPN node that is composed of the combined functions of an NCP and VTAM.
IP Routing			x	These functions are better handled by a traditional IP router.
EXtended Recovery Facility (XRF)	✓			
NTuneMON	✓			
Network Performance Analyzer PU/LU	✓			Network management products based on the NPA LU will work for most functions as they do today; some data items are not reported by the NPA LU when operating in a CCL environment.

NCP Packet Switching Interface (NPSI), X.25 Interconnect (XI), and Network Supervisory Function (NSF)	Directly supported by CCL R2.1	Supported via an aggregation layer router	Not supported by CCL R2.1	Comments
NPSI program product support	✓			
NPSI SNA (PSH and QLLC) communication over X.25 connectivity		✓		QLLC is supported with or without NPSI by terminating the X.25 QLLC lines in an aggregation layer router.
NPSI non-SNA (PCNE, GATE, DATE, and PAD) communication over X.25 connectivity		✓		Non-SNA X.25 with NPSI is supported by terminating the X.25 circuits in an aggregation layer router that communicates using an XOT protocol to an XOT protocol component on Linux that interfaces to NPSI running in CCL
XI and NSF transport of X.25 traffic			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
NSF-based charge-back for X.25 transport services			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
Emulation Program (EP), Partitioned Emulation Program (PEP), and Network Terminal Option (NTO)	**Directly supported by CCL R2.1**	**Supported via an Aggregation Layer router**	***Not* supported by CCL R2.1**	**Comments**
EP/PEP BSC 3270 terminal connection to non-SNA applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
EP/PEP BSC RJE connection to non-SNA applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
EP/PEP Start/Stop terminal connection to non-SNA applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.

NTO Start/Stop terminal connection to SNA applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
NTO BSC RJE connection to SNA applications			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
NTO peer-to-peer connection of non-SNA devices			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
Network Routing Facility (NRF) and non-SNA Interconnect (NSI), MERVA, and TeleProcessing Network Simulator (TPNS)	**Directly supported by CCL R2.1**	**Supported via an Aggregation Layer router**	***Not* supported by CCL R2.1**	**Comments**
NRF peer-to-peer connection of SNA devices (before SNA PU Type 2.1)	✓			
NRF peer-to-peer connections involving non-SNA devices			x	This function requires NTO, and NTO is not supported by the CCL Release 1 environment.
NSI non-SNA NJE to NJE connections between hosts			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
NSI transport of BSC traffic			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
MERVA connection to the S.W.I.F.T network			x	Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
TPNS NCP			x	TPNS traffic through a CCL NCP is supported, but the TPNS NCP itself is not supported in a CCL environment.
Network Node Processor (NNP) and Multi-Access Enclosure (MAE) functions	**Directly supported by CCL R2.1**	**Supported via an Aggregation Layer router**	***Not* supported by CCL R2.1**	**Comments**
NNP or MAE APPN Network Node including HPR and DLUR support			x	These functions can optionally be migrated to Communications Server for Linux on zSeries. Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.

NNP or MAE IP routing			x	These functions are better handled by a traditional IP router.
MAE TN3270 server			x	These functions can optionally be migrated to Communications Server for Linux on zSeries or directly to z/OS, z/VM, or z/VSE. Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.
MAE Network Dispatcher			x	These functions can optionally be migrated to an external load balancer or to z/OS itself using the Sysplex Distributor functions. Refer to IBM Redbook *IBM Communication Controller Migration Guide*, SG24-6298, for alternative technologies.

Table E-3 Software release compatibility overview

Software product	**Compatible releases for CCL Release 2.1**				
NCP V7	R5	R6	R7	R8	R8.1 [a]
NPSI V3	R8	R8	R9	R9	R9 [a]
SSP V4	R5 [b]	R6 [b]	R7 [b]	R8 [b]	R8.1 [a]
NTuneMON V3	R2	R2	R2	R2	R2 [a]
NRF V1	R9	R9	R9	R9	R9 [a]

a. Releases that can be ordered as of May 2006

b. Or later releases

Abbreviations and acronyms

ADS	Application Data Structure
AHHC	APPN Host-to-Host Channel
ANNC	APPN Node-to-Node Communication
ANR	Automatic Network Routing
APPC	Advanced Program-to-Program Communication
APPN	Advanced Peer-to-Peer Networking
BMS	Basic Mapping Support (CICS component)
BN	Border Node
BTT	Branch Transformation Toolkit
CCL	Communication Controller for Linux (IBM 3745 software emulator)
CDLC	Channel Data Link Control (mainframe channel protocol)
CDS	Central Directory Server
CICS	Customer Information Control System (a transaction manager)
CMC	Communications Management Configuration
CNN	Composite Network Node (an APPN NN composed of a VTAM and one or more NCPs)
COS	Class of Service
CP	Control Point
CPI-C	Common Programming Interface for Communications (LU6.2 programming interface)
CSL	Communications Server for Linux
CTG	CICS Transaction Gateway
CWS	CICS Web Support
DLC	Data Link Control
DLSw	Data Link Switching (SNA subarea and APPN/ISR encapsulation over TCP connections)
DRDA	Distributed Relational Data Access
EBN	Extended Border Node (gateway between two APPN networks - APPN equivalent of SNI)
EE	Enterprise Extender (also known as HPR over IP)
EJB™	Enterprise Java Bean
EN	End Node
GR	Generic Resources
HATS	Host Access Transformation Services (SNA 3270/HTML transformation)
HPR	High Performance Routing
HTML	HyperText Markup Language (tag language for Web pages)
HTTP	Hyper Text Transfer Protocol (application protocol between Web browser and Web server)
ICN	Interchange Node (SNA node that routes between APPN and SNA subarea)
ILU	Independent Logical Unit
IMS	Information Management System (z/OS transaction and database manager)
INN	Intermediate Network Node
IP	Internet Protocol
IP TG	Internet Protocol Transmission Group (NCP INN/SNI encapsulation over an IP network between two CCL NCPs)
ISR	Intermediate Session Routing
J2C	J2EE Connector Architecture
LLC2	Logical Link Control type 2 (SNA traffic on a LAN)
MAE	Multi-Access Enclosure
MDH	Migration Data Host (a combined SNA subarea node and APPN EN)
MFS	Message Formatting Services (IMS component)
MID	Message Input Descriptor (IMS MFS input ADS)
MNPS	Multi-Node-Persistence Sessions
MOD	Message Output Descriptor (IMS MFS output ADS)
MPC	Multipath Channel (mainframe channel protocol)
MPC+	HPDT Multipath Channel (mainframe channel protocol)
MPP	Message Processing Program (Transaction program in IMS)
MPR	Message Processing Region
NCP	Network Control Program

NN	Network Node
NNP	(950) Network Node Processor
NNS	Network Node Server
NPSI	NCP Packet Switching Interface (X,25 connectivity support for an NCP)
OTMA	Open Transaction Manager Access
RTP	Rapid Transport Protocol (the transport protocol layer in HPR)
RUI	Request Unit Interface (an SNA programming interface)
SDP	Software Development Platform
SFF	Service Flow Feature (CICS Web services infrastructure)
SNA	Systems Network Architecture
SNI	SNA Network Interconnect (SNA subarea-based business partner communication)
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SSCP	System Services Control Point (PU Type 5 in an SNA subarea network)
TG	Transmission Group
VTAM	Virtual Telecommunications Access Method
WAS	WebSphere Application Server
WMQ	WebSphere Message Queuing
WSADIE	WebSphere Application Developer Integration Edition
WTS	Windows Terminal Server (split GUI technology)
XML	Extensible Markup Language
XOT	X.25 Over TCP/IP (encapsulate X.25 packets over a TCP connection)

Related publications

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

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 120. Note that some of the documents referenced here may be available in softcopy only.

- *IBM Communication Controller for Linux on System z V1.2.1 Implementation Guide*, SG24-7223
- *IBM Communication Controller Migration Guide*, SG24-6298
- *Using IBM WebSphere Host Access Transformation Services V5*, SG24-6099
- *IBM Branch Transformation Toolkit 5.1 Migration and Usage Guidelines*, SG24-7160
- *CICS Transaction Gateway for z/OS Version 6.1*, SG24-7161
- *Revealed! Architecting e-business Access to CICS*, SG24-5466
- *IMS Connectivity in an On Demand Environment: A Practical Guide to IMS Connectivity*, SG24-6794
- *Communications Server for z/OS V1R7 TCP/IP Implementation, Volume 1: Base Functions, Connectivity, and Routing*, SG24-7169
- *Communications Server for z/OS V1R7 TCP/IP Implementation, Volume 2 - Standard Applications*, SG24-7170
- *Communications Server for z/OS V1R7 TCP/IP, Implementation Volume 3 - High Availability, Scalability, and Performance*, SG24-7171
- *Communications Server for z/OS V1R7 TCP/IP Implementation, Volume 4: Policy-Based Network Security*, SG24-7172

Other ublications

- *Communication Controller for Linux on System z Implementation and User's Guide*, SC31-6872

Online resources

These Web sites and URLs are also relevant as further information sources:

- HATS Version 6 Information Center

  http://publib.boulder.ibm.com/infocenter/hatshelp/v60/index.jsp

- HATS Demo site

  http://websphere.dfw.ibm.com/whidemo/atdemo_hats.html

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

T

V

W

Z

IBM Redbooks

A Structured Approach to Modernizing the SNA Environment

(0.2”spine)
0.17”<->0.473”
90<->249 pages

A Structured Approach to Modernizing the SNA Environment

Getting your SNA environment ready for Service-Oriented Architecture

Approaches to an SNA modernization project

Common SNA modernization scenarios

The focus of this IBM Redbook is networking infrastructure aspects of modernizing an SNA network environment. Additionally, modernizing SNA concerns itself with how to modernize access to existing SNA core business applications. In order to provide a reference model for modernizing SNA, this book introduces a selected set of SNA application access modernization technologies, which go beyond the normal networking infrastructure. Some of the technologies discussed in this book aim at improving the user experience when accessing traditional SNA 3270 applications and at exposing existing mainframe SNA applications as Web services. Such technologies will primarily be introduced from a network topology and connectivity perspective.

While a description of the full set of capabilities of such technologies is beyond the scope of this document, readers can refer to other technology-specific and product-specific documentation from IBM and other vendors for that information.

The intended audience for this book are IBM System z technical managers, system architects, and network administrators who are responsible for setting the overall strategic directions for an enterprise networking infrastructure. This infrastructure includes Systems Network Architecture (SNA) and Internet Protocol (IP) networking technologies, branch or remote location access networks, data center connectivity, and business partner connectivity.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7334-00 ISBN 0738496847