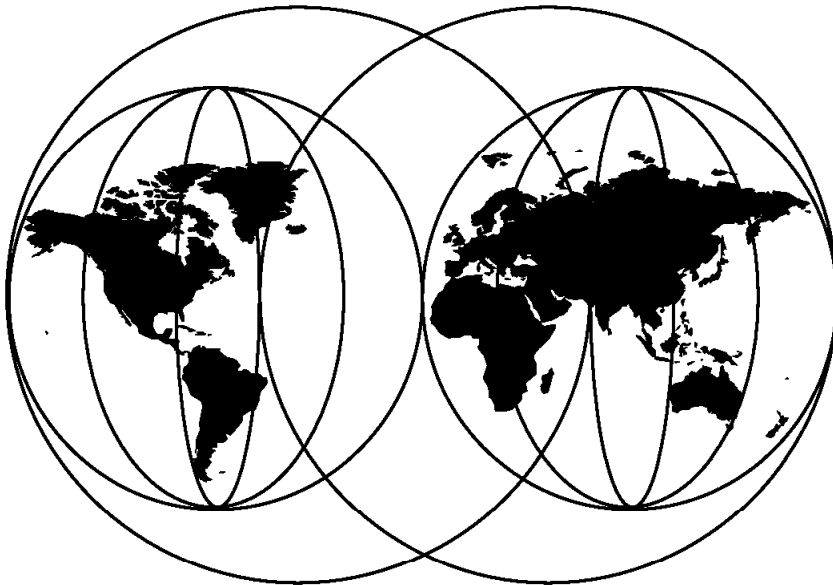




IBM eNetwork Communications Server for AIX: Understanding and Migrating to Version 5: Part 2 - Performance

*Phil Masie
Carla Sadtler*



International Technical Support Organization

<http://www.redbooks.ibm.com>

This book was printed at 240 dpi (dots per inch). The final production redbook with the RED cover will be printed at 1200 dpi and will provide superior graphics resolution. Please see "How to Get ITSO Redbooks" at the back of this book for ordering instructions.

SG24-2136-00



SG24-2136-00

International Technical Support Organization

**IBM eNetwork Communications Server for AIX:
Understanding and Migrating to Version 5:
Part 2 - Performance**

June 1998

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix C, "Special Notices" on page 51.

First Edition (June 1998)

This edition applies to Version 5 of IBM eNetwork Communications Server for AIX, Program Number 5765-D20 for use with the AIX Version 4.1.5 System or later.

Comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1998. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	v
The Team That Wrote This Redbook	v
Comments Welcome	vi
Chapter 1. Planning for Your Network	1
1.1 Overview of Communications Server for AIX	1
1.2 What Is New for This Release	2
1.3 Communications Server for AIX Version 5 Main Functions	4
1.4 Stages of Network Planning	4
1.4.1 Identifying Functional Requirements for the Network	5
1.4.2 Determining How to Configure Communications Server for AIX Products	5
1.4.3 Identifying Resource Requirements for Installation and Operation	6
1.4.4 Memory and Storage Requirements for Operation	10
1.4.5 Estimating Resource Requirements for Operation	12
1.5 Ensuring Compatibility among Cross-Platform Configurations	14
1.6 Naming Conventions	15
Chapter 2. Tuning Communications Server for AIX	17
2.1 Understanding Data Flow	17
2.1.1 TP Data Transformation	18
2.1.2 Potential Bottlenecks	20
2.1.3 U-Shaped Sessions	20
2.2 Allocating and Using Control Blocks and Message Buffers	21
2.3 Using Session-Level Pacing	24
2.3.1 Session-Level Window Size	25
2.3.2 Improving Throughput	26
2.4 Using DLC-Level Pacing	28
2.4.1 Local Area Network (LAN) Connections	28
2.4.2 SDLC Connections	32
2.5 Adapter Queues	33
2.5.1 Receive Queue Overflow	33
2.5.2 Send Queue Overflow	34
2.5.3 Adjusting the Sizes of Adapter Queues	34
2.6 Network Tuning Decisions	34
2.6.1 Modifying SNA and System Parameters	34
2.6.2 TN Server Tuning Considerations	36
2.6.3 Summary of Tuning Recommendations	37
Appendix A. Common Performance and Tuning Problems	39

Appendix B. Performance Tests	41
B.1 Functions Tested	41
B.1.1 APPC	41
B.1.2 APPN	42
B.1.3 SNA Gateway	43
B.1.4 AnyNet	44
B.1.5 SNA Channel: Block Multiplexer and ESCON	45
B.1.6 TN Server	48
Appendix C. Special Notices	51
Appendix D. Related Publications	53
D.1 International Technical Support Organization Publications	53
D.2 Redbooks on CD-ROMs	53
D.3 Other Publications	53
How to Get ITSO Redbooks	55
How IBM Employees Can Get ITSO Redbooks	55
How Customers Can Get ITSO Redbooks	56
IBM Redbook Order Form	57
List of Abbreviations	59
Index	61
ITSO Redbook Evaluation	63

Preface

This redbook will help you plan, and understand performance and tuning information for the IBM Communications Server for AIX Version 5 Release 0. The redbook shows recent performance tests, and documents the results on the following Communications Server for AIX functions:

- APPC/CPI-C
- APPN
- SNA Gateway
- TN Server
- AnyNet access node (APPC/IP and Sockets/SNA)

This book identifies the factors to consider when planning workstations and networks, and provides tuning guidelines as well as experimental data to help you optimize overall performance. Covering every possible scenario that can arise in a customer environment is nearly impossible. Instead, this book tries to shed light on the factors driving the interactions of various code components, algorithms, and parameters.

This book is based on a previous product publication, whose title was *Communications Server for AIX Version 4 Release 2 Planning and Performance Guide*, SC31-8220-01.

The Team That Wrote This Redbook

Phil Masie is the Communications Server for AIX System Test Manager. He has had this responsibility for two years. Prior to that, he was one of the technical leads on the Communications Server for AIX System Test team. All in all, he has been associated with the CS/AIX product since 1992. Prior to 1992, Phil led an IBM Corporate-wide effort to establish an e-mail gateway to the Internet for IBMers worldwide to access from their desktops.

Carla Sadtler is a Senior ITSO Specialist for Communications Server for AIX and network integration projects at the Systems Management and Networking ITSO Center, Raleigh. She has been in the ITSO since 1985 and has been working in the networking field for her entire career.

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

Ronnie Jones
IBM AIX Networking Test

Philip Pearl
Data Connection Ltd.

Comments Welcome

Your comments are important to us!

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

- Fax the evaluation form found in "ITSO Redbook Evaluation" on page 63 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web sites:
For Internet users <http://www.redbooks.ibm.com/>
For IBM Intranet users <http://w3.itso.ibm.com/>
- Send us a note at the following address:
 redbook@us.ibm.com

Chapter 1. Planning for Your Network

This chapter provides an overview of Communications Server for AIX and the stages of planning a network. It also summarizes the functions that you can configure for an AIX workstation and presents guidelines for estimating the resources required to support the functions.

1.1 Overview of Communications Server for AIX

Communications Server for AIX is a stand-alone program that connects applications across an SNA network. It equips an AIX workstation with SNA resources and protocols, enabling it to support the following types of networks, link-level protocols, logical units (LUs), application programming interfaces (APIs), software features, and products:

- SNA host-dependent networks including SNA Gateway function
- TN Server to connect TN3270 IP clients to SNA Hosts
- Advanced Peer-to-Peer Networks (APPN)
- Local area networks (LANs) using token-ring, standard Ethernet, and 802.3 Ethernet protocols, Fiber Distributed Data Interface (FDDI), and Asynchronous Transfer Mode Local Area Network Emulator (ATM LANE)
- Wide area networks (WANs) using any of the following data link controls (DLCs):
 - EIA232D
 - Smart modem
 - X.21
 - EIA422A
 - V.25bis
 - V.35
 - X.25
 - Frame relay
- LU types 0, 1, 2, 3, and 6.2 (dependent and independent)
- The following new APIs:
 - MS API for providing functions for network management and management services applications. It supports network messaging.
 - LUA API, enabling application programmers to write applications that communicate with host applications at the RU level, and to send and receive data on both the SSCP-LU session and the PLU-SLU session. This API can be used to support LU 0, 1, 2, or 3 communications with the host.

- The NOF API can be used to write applications that administer CS/AIX configuration and management resources.
- The new CPI-C (Common Programming Interface for Communications) provides CPI-C 2.0+ support and is backwards compatible with existing CPI-C applications written for CS/AIX.
- The new APPC (advanced program-to-program communication) API supports LU 6.2 communication using either independent sessions for peer-to-peer communication or dependent sessions for host communication.
- The CSV (Common Service Verb) API provides utility verbs that enable an application program to perform functions such as character set conversion and trace file control.
- Back-level support for the following APIs:
 - Generic SNA
 - LU 0
 - Operating System Subroutines (LU types 1, 2, 3, and 6.2)
 - Library Subroutines for TP Conversions (LU types 1, 2, 3, and 6.2)
 - Library Subroutines for Network Management, used for exchanging network management vector transports (NMVTs) on SSCP-PU sessions
 - Management Services (MS)

1.2 What Is New for This Release

Communications Server for AIX Version 5 Release 0 is the follow-on version to Communications Server for AIX Version 4 Release 2. The following functions have been added since Communications Server for AIX Version 4 Release 2.

- A Motif Administration Graphical User Interface (GUI) to configure and maintain your network. This is a significant 'ease of use' enhancement.
- Frame relay support can be provided over an emulated token-ring interface with the TPS/SoftFRAD product.
- MultiPath channel (MPC) support is provided for the ESCON adapter.
- ATM (Asynchronous Transfer Mode) support can be provided using an emulated LAN (token-ring or Ethernet) interface.
- TN Server supports both TN3270E and TN3270 protocols, providing access to 3270 sessions for clients using the telnet protocol over TCP/IP.

- Configuration changes made using the Motif administration program, the command-line administration program, and the NOF (Node Operator Facility) API are applied immediately to the node configuration file.
- Storage of configuration information is in text files, not in an ODM database.
- The High-Performance Routing (HPR) feature of APPN now includes Rapid Transport Protocol (RTP). This feature provides end-to-end support for nondisruptive rerouting of session traffic around route failures or congestion. This function is supported over all link types.
- Dependent LU requester (DLUR) enables sessions between dependent LUs and hosts to be routed over APPN networks. DLUR support was previously available for LU types 1-3 and dependent LU 6.2, and is now extended to sessions that use Generic SNA, secondary LU 0, and the new LUA API.
- DDDL (Dynamic Definition of Dependent LUs) enables CS/AIX to dynamically define LUs to VTAM systems, reducing the need for host configuration. This function is sometimes known as SDDL (Self-Defining Dependent LUs).
- CS/AIX provides enhanced APIs that are more compatible with the APIs provided by products in the Communications Server family running on other operating systems. Older API support is continued. CS/AIX now includes the following new or changed application programming interfaces (APIs):
 - The new LUA API enables application programmers to write applications that communicate with host applications at the RU level, and to send and receive data on both the SSCP-LU session and the PLU-SLU session. This API can be used to support LU 0, 1, 2, or 3 communications with the host.
 - The NOF API can be used to write applications that administer CS/AIX configuration and management resources.
 - The new CPI-C (Common Programming Interface for Communications) provides CPI-C 2.0+ support and is backwards compatible with existing CPI-C applications written for CS/AIX.
 - The new APPC (advanced program-to-program communication) API supports LU 6.2 communication using either independent sessions for peer-to-peer communication or dependent sessions for host communication.
 - The CSV (Common Service Verb) API provides utility verbs that enable an application program to perform functions such as character set conversion and trace file control.

- The new MS (Management Services) API supports network messaging functions.
- Communications Server for AIX SNA gateway is now part of the base product.
- AnyNet APPC over TCP/IP and AnyNet Sockets over SNA are now part of the base product.
- APPC Application Suite is now part of the base product.

1.3 Communications Server for AIX Version 5 Main Functions

SNA Gateway

SNA Gateway enables an AIX workstation to function as a gateway that links multiple downstream workstations in an SNA network to one or more host computers.

APPC over TCP/IP

APPC over TCP/IP is a Multiprotocol Transport Networking (MPTN) software feature. It implements mixed protocol networking to enable APPC and Common Programming Interface (CPI) Communications applications to communicate over TCP/IP networks.

Sockets over SNA

Sockets over SNA is an MPTN software feature that supports mixed protocol networking to enable sockets applications to communicate with each other over SNA networks.

TN Server

TN Server allows TN3270 IP clients to connect to SNA hosts.

1.4 Stages of Network Planning

This section presents some general guidelines for planning, configuring, and managing networks through the use of various Communications Server for AIX and AIX utilities.

Planning a network involves balancing function, performance, resources, and cost. Although there is no single best plan for a network, some general guidelines and techniques can help ensure that your plan meets your needs.

To plan a network, perform the following tasks:

- Determine the functions the network should provide (such as file transfer or 3270 emulation) and your performance requirements.
- Determine how Communications Server for AIX products can be configured to provide the functions you need.

- Identify the resources required for installing Communications Server for AIX products.
- Estimate the resources needed to support your performance and capacity requirements as well as Communications Server for AIX functions, and identify the associated costs.

1.4.1 Identifying Functional Requirements for the Network

To determine the functions a network should provide, you need to consider the following questions:

- Do I need to merge SNA and TCP/IP networks?
- Do I need to run socket applications over an SNA network?
- Do I need to run APPC applications over a TCP/IP network?
- Is the network to be APPN?
- Is the AIX workstation to be an endpoint for sessions or is it to be one of the following types of gateways?
 - APPN
 - LU 2
 - TCP/IP
- What types of physical links will the network use?

1.4.2 Determining How to Configure Communications Server for AIX Products

To determine how an AIX workstation running Communications Server for AIX is to function, you should first decide how work is to flow through the network. The following questions should be considered:

- What resources (such as applications) should be available through the network?
- How many users need access to remote resources?
- How frequently is each resource accessed?
- How can users get access to the network?
- How will user requests be routed through the network?

You can configure Communications Server for AIX products to support many functions, including, for example, the following:

- APPN network node server for Intermediate Session Routing (ISR)
- APPN end node (which communicates autonomously with adjacent nodes but uses APPN network node services to communicate with nonadjacent peer nodes)

- Low-entry network (LEN) node (which communicates directly with adjacent nodes or nodes configured to appear adjacent)
- Host terminal emulation using LU 2
- Use of LU 0, LU 1, LU 2, LU 3, and LU 6.2 (dependent and independent)
- SNA Gateway connection between hosts and workstations
- TN Server connection between TN3270 IP clients and SNA hosts

One or more functions can be configured for a given node, depending on your needs. For example, you can configure an AIX workstation as an APPN network node to provide routing services and ISR. You can also use the same node as an SNA gateway for routing dependent LU sessions, such as LU 2. Another example of a multi-functional node is an AIX workstation running TN Server, and supporting a DB2/6000 database as well as an independent LU 6.2 connection to a host DATABASE 2 (DB2) database.

1.4.3 Identifying Resource Requirements for Installation and Operation

To estimate support for the functions of Communications Server for AIX products, the following questions must be answered:

- What personnel skills do I need?
- What model RISC System/6000 workstation do I need?
- What transport media do I anticipate using?
- What are the installation requirements for the configuration I select?
- How much memory and paging space do I need for operation?
- What response time can I expect with the anticipated traffic level?

Answering these questions will help you identify the types of resources that an AIX workstation uses when it is configured to support any one or more of the many functions described in 1.4.1, “Identifying Functional Requirements for the Network” on page 5. Answering the questions will also help you to understand the relationship between Communications Server for AIX functions, AIX resources, and network resources.

How you allocate resources to nodes determines how the network will perform.

1.4.3.1 Personnel Requirements

Tuning, as well as installing and operating SNA products for AIX, requires the following personnel:

- Network administrators, who plan the network, add new devices, and maintain or enhance overall network performance

- System administrators, who install and maintain workstations and SNA products, and who configure systems for network connection

Administrators should be familiar with AIX workstations, the AIX operating system, SMIT, and TCP/IP. They must know the networks to which various systems are connected and understand SNA concepts in general.

1.4.3.2 RISC System/6000 Models

RISC System/6000 workstations are available in many models that have different capacities for memory, disk space, I/O adapters, and CPU speeds. They are packaged as follows:

- Desktop models, such as the Model 250 or 43P. The desktop models have the smallest capacity for memory, disk drives, and I/O adapters (such as token-ring). They are best suited as client or endpoint systems.
- Deskside models, such as the Model 590 or J30. The deskside models have more capacity for memory, disk drives, and I/O adapters. They are suitable for small- to medium-sized server or multi-user environments.
- Rack Mount models, such as the Model 990 or R24 or Scalable POWERparallel Systems (SP2). The rack mount models have the most memory, disk, and I/O capacity, and are suitable for large server or multi-user environments. SP2 systems are rack mounted and contain up to 128 processors, making them particularly suited to applications that can take advantage of parallel processing.

1.4.3.3 Transport Media

Communications Server for AIX may need to share the underlying transport medium (such as SDLC, token-ring, or FDDI) with other communication protocols. Therefore, physical layer bandwidth requirements must accommodate all the protocols and applications sharing the transport medium.

Note: Communications Server for AIX can share the RISC System/6000 token-ring, Ethernet, and FDDI adapters with other protocols such as TCP/IP. You may need to specify unique service access point (SAP) addresses for each protocol to use. Communications Server for AIX can also share a Multiprotocol Quad Port (MPQP) adapter, but not an MPQP port.

1.4.3.4 Installation Requirements

The function that you assign to the RISC System/6000 workstation (from 1.4.1, “Identifying Functional Requirements for the Network” on page 5) also determines the installation requirements. This section provides an overview of the computer resources required for installing Communications Server for AIX products.

Link Hardware

Link hardware consists of a communication adapter and a matching cable for attachment to the network. The RISC System/6000 workstation accepts the following adapters:

- IBM Token-Ring High-Performance Network Adapter or ISA Token-Ring adapter for connecting the RISC System/6000 workstation to a token-ring LAN or IBM PCI Token-Ring adapter or ATM Adapter with ATM LAN Emulation.
- Ethernet LAN adapter or ISA Ethernet LAN adapter for connecting the RISC System/6000 workstation to Ethernet Version 2 or IEEE 802.3 Ethernet LAN or IBM PCI Ethernet adapter or IBM 100 Mbps Ethernet adapter or ATM Adapter with ATM LAN Emulation.
- SDLC links for connecting the RISC System/6000 workstation to a modem
 - IBM 4-Port Multiprotocol Communications Controller and 4-Port Multiprotocol Interface Cable. The 4-port adapter provides four independent communication interfaces that support only SDLC protocols.
 - IBM Single Port Multiprotocol Adapter/A and appropriate cables for establishing an EIA232D SDLC connection for external clocking.
- IBM X.25 adapters
 - X.25 Interface Co-Processor/2
 - Portmaster adapters supported by the AIX Link X.25 LPP
- IBM Fiber Distributed Data Interface Adapters for attachment to an FDDI network:
 - Fiber Single-Ring Adapter
 - Fiber Dual-Ring Upgrade
 - STP Single-Ring Adapter
 - STP Dual-Ring Upgrade
- Frame relay connections are supported by utilizing the TPS SoftFRAD product from TPS Systems, Inc.. For complete adapter support information refer to the TPS Systems, Inc. Web pages at <http://www.tpssys.com>. Frame relay adapters include:
 - PortMaster
 - Artic

Frame relay adapters tested during the production of this book are:

- TPS Systems 8-Port PortMaster Adapter/A
- IBM 4-Port Selectable Portmaster Adapter/A

- IBM 4-Port RS232/4 Port RS-422 Multiport/2 Adapter
- ATM:
 - IBM AIX ATM adapter
- Channel adapters:
 - ESCON
 - Block Multiplexer

Software

Communications Server for AIX Version 5 Release 0 runs on the AIX base operating system (BOS) Version 4.1.5 or later. This system supports the symmetric multi-processing (SMP) platform, which enables multiple processes to proceed in parallel on a given RISC System/6000 workstation.

Communications Server for AIX also requires appropriate DLC support for the links that it will use in operation. The DLC support can be found in the bos.dlc package, which is part of the aforementioned AIX BOS.

Memory and Storage

To support a full range of configurations and services, at least 32 MB of memory.

Note: We strongly recommend 64 MB of memory. In addition, 2 MB of temporary storage is required during installation.

If you decide to install the documentation for CS/AIX, you need additional fixed-disk storage for each additional softcopy book installed. To install all softcopy books, you need 8 MB of disk space.

Note: Memory and fixed-disk requirements for other licensed programs, user applications, and data are not included in these requirements.

Disk space requirements are shown below. This table has been updated to show the requirements documented in the README file shipped with CS/AIX.

Description	Fileset	Storage Requirements
CS/AIX base code (required)	sna.rte	26 MB
Messages, per language (required; exact number depends on language)	sna.msg. <i>Language</i> .rte	3.0 MB to 5.0 MB

Description	Fileset	Storage Requirements
Filesets for backwards compatibility (required)	sna.anynet.base, sna.anynet.snaip, sna.anynet.socksna, sna.gw, sna.toolkit.3270, sna.toolkit.aftp, sna.toolkit.aname, sna.toolkit.basic, sna.toolkit.misc and sna.toolkit.rte	120 KB
Motif administration program	sna.xsna	8 MB
LU 0	sna.lu0	600 KB
Reference manual pages for SNA commands	sna.man.Language.rte.data	176 KB
SNA Application Development Toolkit (SNAPI)	sna.snapi	1.2 MB
Online publications in HTML	sna.html.Language.data	14 MB to 18 MB
Host Access Class Library	sna.ecl.data	3.3 MB
SNA Channel Attachment Feature	sna.dlcchannel and sna.dlcmpc	3.5 MB
HCON	hcon.rte	3.5 MB

1.4.4 Memory and Storage Requirements for Operation

This section describes the memory and storage requirements of a RISC System/6000 workstation running Communications Server for AIX products.

STREAMS Buffers

CS/AIX uses STREAMS buffers for internal communications. The STREAMS buffer pool is a shared resource that must be managed at the system level. CS/AIX is just one of the subsystems that uses STREAMS buffers; its STREAMS resource usage can affect the performance of other subsystems. Coordinate with your network administrator in determining STREAMS buffer requirements.

Memory Buffers (Mbufs)

AIX uses memory buffers (mbufs) to enable communication between the communication subsystems and the AIX DLCs. The mbuf pool is a shared resource that must be managed at the system level. Communications Server for AIX uses mbufs to send and receive data

across the network, but it is just one of the subsystems that uses mbufs. Communications Server for AIX's mbuf resource utilization can affect the performance of other subsystems, such as TCP/IP, NFS, or AFS. Coordinate with your network administrator in determining mbuf requirements.

Memory, Disk, and Paging Storage

Communications Server for AIX uses primary storage (also known as main memory and Random Access Memory), permanent disk storage, and paging disk storage (also known as paging space):

- Control blocks for SNA resources—such as LUs, links, and sessions—consume primary storage.
- Memory buffers consume only primary storage.
- Communications Server for AIX executables, configuration profiles, and softcopy publications consume permanent disk storage.
- Application programs, sessions, and related processes have paging space requirements that consume permanent disk storage.

Memory and disk storage requirements depend on various factors that widely differ from one environment to another. Transaction programs (TPs) and SNA sessions are the dominant consumers of memory and disk storage.

A TP is a program that uses the SNA application programming interface (API) to access the network. A session is a temporary logical channel between two LUs on partner nodes. TPs use this channel to communicate with each other.

The consumption of memory and disk storage is largely affected by the overall number of sessions, the number of allocated sessions (conversations), the traffic intensity of the conversations, and the number of active user TPs. Memory requirements are mostly affected by the data-traffic intensity and the resulting usage of mbufs. Disk storage requirements stem from the paging space requirements of the TPs, sessions, and related processes.

For all LU types, a distinction is made between an idle session and an allocated session. An idle session is an active session that is not being used by TPs. An allocated session is one that supports an ongoing conversation. To estimate the total memory requirement, you need to estimate the traffic intensity and the portion of the overall number of sessions that will be, on average, allocated at any point in time. You can also choose to configure the system to handle peak demand. These estimates vary among environments. Consequently, the numbers given in the sections that follow should be taken as rough

estimates. These estimates are meant to avoid system paging. You could, however, increase the session load beyond system memory capacity and incur the cost of paging.

Processes

Communications Server for AIX requires a minimum of nine AIX processes. No additional kernel processes are created for LU 1, 2, 3, or 6.2 sessions (this is different from previous versions of CS/AIX). The LU 0 facility can create up to 33 processes, depending on the number of active primary and secondary servers. This does not include the processes necessary for Communications Server for AIX.

Response Time

Estimating response time for anticipated traffic levels is an impossible task because of the large number of networking conditions that exist. Because response time also cannot be expressed in a concise formula, it is best obtained from experimental data, which you can extrapolate to your network.

To help you in your estimates, Appendix B, "Performance Tests," presents different network configurations, and shows how the response time of each network is affected by such factors as the model of the RISC System/6000 workstation, and the number of ongoing sessions. For more information about response time, see Appendix B, "Performance Tests" on page 41.

1.4.5 Estimating Resource Requirements for Operation

This section presents guidelines for estimating resource requirements for various types of sessions. In all cases, the resource requirements are broken down into units of one session. To calculate your network's overall resource needs, do the appropriate multiplication.

For all examples below, the paging space requirements should be figured by taking the memory requirement and doubling it (paging space=memory*2).

1.4.5.1 3270 Dependent LU Sessions

Consider the following when planning resources required for 3270 dependent LU sessions.

Session	Memory Required
1 SSCP-LU	2 KB
1 LU-LU	6 KB

In the case of a 3270 HCON display session, the resource consumption of the AIXwindows sessions must also be factored in.

1.4.5.2 SNA Gateway

Consider the following when planning resources required for the SNA Gateway function.

Session	Memory Required
1 SSCP-LU	4 KB
1 LU-LU w/traffic	6 KB

For guidance on the model RS/6000 required to support the planned sessions and traffic load, see 1.4.5.7, "SPECint95 Ratings for TN Server and SNA Gateways" on page 14.

1.4.5.3 TN Server

Consider the following when planning resources required for TN3270(E) sessions.

Session	Memory Required
1 SSCP-LU	5 KB
1 LU-LU w/traffic	10 KB

For guidance on the model RS/6000 required to support the planned sessions and traffic load, see 1.4.5.7, "SPECint95 Ratings for TN Server and SNA Gateways" on page 14.

1.4.5.4 APPC

Consider the following APPC resource requirements when planning your network.

Session	Memory Required
1 LU-LU	11 KB

To make the above estimates complete, add the resources required by the TPs you plan on using. If this information is not available, use 100 KB as a guess for memory and paging space consumption.

1.4.5.5 AnyNet APPC/IP and Sockets/SNA

Consider the following AnyNet resource requirements when planning your network.

Session	Memory Required
1 Sockets/SNA	58 KB
1 APPC/IP	15 KB

1.4.5.6 APPN ISR Sessions

Since a single process manages all APPN intermediate session routing, the memory required is restricted to mbuf consumption. The paging space requirement per session is close to 0 KB. Mbuf consumption is restricted by the number of available adapters and should be estimated by allotting 4 MB per adapter. The cost per session is therefore as little as 1 KB per session used for internal session control blocks.

Estimate 4KB for each end node supported by the network node.

1.4.5.7 SPECint95 Ratings for TN Server and SNA Gateways

Number of Sessions	Minimum SPECint95	Processor
500 - 1500	3.3	R20
1500 - 2500	6.7	595
2500 - 3500	9.4	E30
3500 - 4500	12.9	43P-140
4500 - 6000	12.9	43P-240 (2 processors)

It is difficult to give precise recommendations, because response time is sensitive to the specific nature and traffic intensity of the active sessions. If you suspect the intensity of traffic to be consistent and high, then selecting a higher end RS/6000 would be recommended.

1.5 Ensuring Compatibility among Cross-Platform Configurations

SNA products running on different platforms, such as Communications Server for Windows NT (CS/NT) or Communications Server for OS/2 (CS/2) or on Intel platforms or VTAM NCP on a host can function with Communications Server for AIX if certain requirements are met.

In general, the current release of an SNA networking product can function with Communications Server for AIX as long as it supports PU 2.1 nodes and at least one of the link types supported by Communications Server for AIX. However, some very old releases (such as VTAM V2) may not function reliably. The capabilities of each networking product are documented in its corresponding user's guide.

Other factors to consider include the following:

- Because default values for DLC window sizes and timers vary from device to device, you must ensure that your remote devices use the correct value. For example, a DLC window size that gives good performance with a CS/2 node may not perform well with a 3172 node. For more information about window size, see 2.4, “Using DLC-Level Pacing” on page 28.
- When working in a heterogeneous environment (TCP/IP and SNA protocols on the same LAN) with multiple LAN segments, make sure that your LAN interconnect devices can “route” TCP/IP and “bridge” SNA frames at the same time.

1.6 Naming Conventions

You can use network IDs to logically segment your physical network. Also, if you plan to connect to other networks, we highly recommend that you register your network IDs to avoid network name conflicts.

You can define network and LU names as follows:

Network names

You can define different network names (network IDs) to provide segmentation of APPN networks. Segmentation limits the size of network topology databases and the frequency of broadcast *LOCATE* requests through each network.

To ensure the uniqueness of a network ID, a network administrator can register the network’s ID with the IBM worldwide registry. The IBM registry ensures that each network ID is unique among those registered with it. Registry standards are consistent with Open Systems Interconnection (OSI) standards, including OSI country codes, as established by the International Organization for Standards (ISO). For more information about registration, see *User’s Guide for SNA Network Registry*.

LU names

You can use wildcards for LU names to minimize system definition and network searches.

Chapter 2. Tuning Communications Server for AIX

Network tuning is not an easy task. Even the simplest case of a two-node network involves many hardware and software parameters whose interdependencies can significantly impact network performance. The performance of the Communications Server for AIX product family also depends on the interactions of the individual products with the AIX operating system, as well as with the network software and hardware.

Instead of considering every possible scenario, this chapter broadly discusses performance-related topics with respect to tuning a node running Communications Server for AIX products. Topics discussed include the following:

- Flow of data through various network components
- Transformations performed on the data
- Potential bottlenecks that can slow down data transfer
- Allocation of memory buffers
- Session- and DLC-level pacing
- Adapter queues
- AIX 4.1 Symmetric Multiple Processor (SMP) Support

The chapter concludes with a discussion of network tuning and a summary of tuning recommendations.

2.1 Understanding Data Flow

The efficient flow of data through the various network components depends on the availability of system resources. It is also affected by constraints imposed by pacing algorithms.

The following diagram is a simplified overview of the flow of data in a simple two-node network.

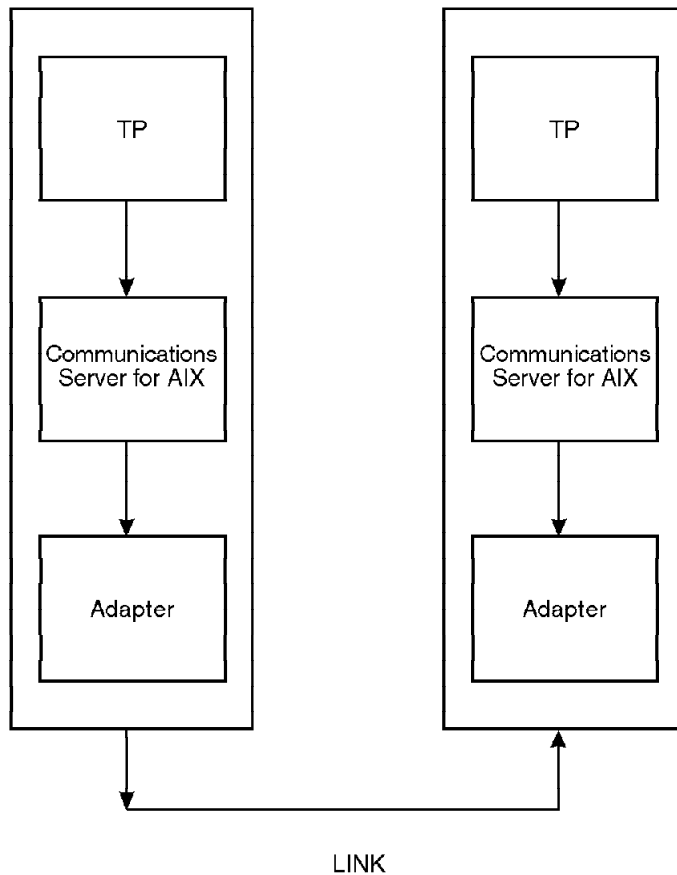


Figure 1. Flow of Data through Network Components

Data generated by the user's TP goes through several transformations as it travels from the local TP to the remote one. It enters the local SNA protocol stack as a data buffer and reaches the remote TP as such. Along the way, however, the data is copied into mbufs, acquires protocol headers and trailers, and possibly is segmented into smaller pieces. When data arrives at the destination node, the transformations performed on it by the sender are reversed by the receiving node. Segmented messages are reassembled, headers and trailers are stripped, and data is passed on to the remote TP.

2.1.1 TP Data Transformation

The transformation of TP data is accomplished as follows:

1. The local TP sends a data buffer to the local SNA subsystem. SNA transforms the data buffer into one or more BIUs. The BIU (basic

information unit) consists of an RH header followed by an RU (request unit) whose size is determined from user-supplied profiles. The BIU is the fundamental unit passed between origin and destination half-sessions. If the amount of user data does not fit in a single RU, it will result in multiple BIUs.

2. In the next phase, a BIU is transformed into one or more PIUs (physical information unit). PIU size is determined by the BTU (basic transmission unit) size. BTU size is the size of the information field contained in a BLU (basic link unit). If a BIU is small enough to fit in a PIU, a TH (transmission header) is added to it, forming a single PIU. Otherwise, the BIU is segmented into multiple BIU segments, each of which gets a TH header added to it, forming multiple PIUs.
3. BIUs and PIUs are put into mbufs and mbuf clusters. The process of transforming TP data into BIUs and PIUs consumes mbufs and mbuf clusters. If they are not available, delays can occur.
4. Once there is a full RU worth of data, the BIU is ready to move on to the SNA send queue (an RU of less than maximum size can be sent when the TP issues FLUSH, PREPARE_TO_RECEIVE, CONFIRM, or DEALLOCATE calls). The PIUs that make up the BIU are placed on the queue. The queuing of BIUs continues as long as the pacing count is larger than 0. If pacing count is reduced to 0, BIU transfer is blocked until a new pacing window is granted by the adjacent node.
5. From the SNA send queue, PIUs are passed on to the adapter send queue. Prior to transmission, the PIU (which is now a BTU) is enveloped in a link header (LH) and trailer (LT) to form a BLU. The transmission of frames (BLUs) is subject to DLC-level pacing constraints, which will be discussed later. At this point, the frame is ready for transmission.
6. Upon arrival at the receiving node, the steps taken on the send path are reversed on the receive path.
 - a. The frame gets stripped of its header and trailer, gets copied into an mbuf, and gets placed on the adapter receive queue.
 - b. From there the BTUs (PIUs) are passed to the SNA receive queue and assembled into whole BIUs.
 - c. A BIU containing a complete RU is then passed on to the receiving TP.

At times, the receiving end of the frame transmission is not the BIU destination but a network node whose function is to route the BIU forward to its final destination. If the inbound BTU size and the outbound BTU size are identical, the incoming BIU is moved quickly from the inbound queue to the outbound one. If not, segmentation or

reassembling of the BIU takes place and the mbufs containing the inbound BIUs are copied into outbound BIUs.

To help you understand such data transformations, the following diagram illustrates format evolution for the simple case where the TP buffer fits into a single BIU, which in turn fits into a single PIU.

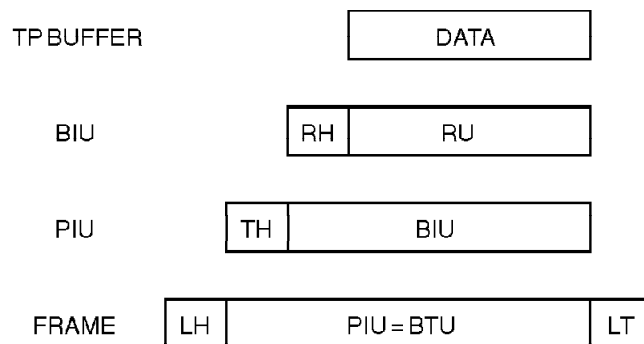


Figure 2. Format Evolution

2.1.2 Potential Bottlenecks

There are five potential bottlenecks that can slow down data transfer:

- CPU load
- Shortage of mbufs
- Pacing constraints
- Finite queues
- Link speed

High CPU utilization is sometimes unavoidable and link speed is a given. However, some measures can be taken to resolve or alleviate the other three potential bottlenecks. Performance can also suffer in the reverse situation where network resources are underutilized due to a poor choice of system parameters. These issues are discussed in the sections that follow.

2.1.3 U-Shaped Sessions

Testing your applications with U-shaped sessions can show whether the bottleneck is the physical network or another factor like the CPU speed or mbuf utilization. With U-shaped sessions, both the local and remote TPs are within the same AIX machine and no actual network traffic is generated. If your TPs perform well with U-shaped sessions and perform poorly with a real remote machine, you need to look at the network or the remote machine for the bottleneck.

2.2 Allocating and Using Control Blocks and Message Buffers

CS/AIX uses four different types of kernel memory:

- Control block memory (non-pinned kernel heap)
- STREAMS buffers
- Mbufs
- Externally allocated STREAMS buffers

STREAMS buffers and mbufs are both allocated out of a pool of memory reserved by AIX for network use. The maximum size of this pool is controlled by a configurable AIX system parameter, `thewall`. See the `network option (no)` command for more details.

System STREAMS buffer and mbuf usage can be monitored using the `netstat -m` command. Each buffer type takes up two rows of output: mbufs are made out of mbufs and mclusters; STREAMS buffers are made out of mblks and mblkdata.

```

105 mbufs in use:
32 mbuf cluster pages in use
154 Kbytes allocated to mbufs
0 requests for mbufs denied
0 calls to protocol drain routines

Kernel malloc statistics:

***** CPU 0 *****
By size      inuse    calls failed    free  hiwat  freed
32           290     12114    0     94    640    0
64           102      817     0     26    320    0
128          156    175908    0    132    160    18
256          456   3994478    0    328    384    12
512          374   877194    0     10     40    57
1024         5    120542    0     15     20   131
2048         0     1294     0     10     10     7
4096         89     701     0     84    120     0
8192         1      22     0      1     10     0
16384        1       2     0     18     24     7

By type      inuse    calls failed  memuse  memmax  mapb
mbuf       115  3918068  0  29440 160256  0
mcluster  32   953688  0  131072 180224  0
socket       373    23464    0  102176  110784    0
pcb          312    20118    0  76032  80640     0
routetbl    17      19     0   2272   2912     0
fragtbl     0      270     0     0     32     0
ifaddr      8       8     0   2112   2112     0
mblk      133  175823  0  24320 88448  0
mblkdata  68   88148  0  49152 301056  0
strhead     26      39     0   8992   8992     0
strqueue    43      76     0  22016  22016     0
strmodsw    24      24     0   1536   1536     0
strosr      2      236     0    512    768     0
strsyncq    52     158     0   5984   6048     0
streams     153    185     0  19296  19296     0
devbuf      4       4     0  12608  12608     0
kernel table 67     564     0  77152  79712     0
locking     3       3     0    384    384     0
temp        63     127     0 211552 216160     0
mcast opts  0       4     0     0    128     0
mcast addr  2       2     0    128    128     0

Streams mblk statistic failures:
0 high priority mblk failures
0 medium priority mblk failures
0 low priority mblk failures

```

STREAMS buffers and mbufs are relatively expensive. They are allocated out of pinned memory and therefore may not be swapped out to the paging file. To reduce the amount of pinned memory it uses, CS/AIX does the following:

- Only uses mbufs to communicate with the DLCs and within the AnyNet component
- Uses externally allocated STREAMS buffers for all but the smallest messages


```
csaix1:/ > snaadmin query_kernel_memory_limit
```

```
-----  
limit = 0  
actual = 231596  
max_used = 231596  
reset_max_used = NO
```

CS/AIX by default limits itself to using at most 50% of the physical RAM for control blocks and 25% of the physical RAM for STREAMS buffers. You can change these limits with the following snaadmin commands:

- set_kernel_memory_limit
- set_buffer_availability

2.3 Using Session-Level Pacing

In an ideal setting, there is no need for a pacing mechanism; as long as the producer (sender) and consumer (receiver) of data operate at the same rate, the data transfer rate does not need to be regulated.

The networking environment, however, is rarely ideal. Often, the rates of producing and consuming data differ among conversation partners. Different CPU speeds and CPU loads are two of the most common causes for the discrepancy between them. Pacing is the remedy for this discrepancy; its purpose is to prevent the producer from overwhelming the consumer with a flood of data. Pacing comes in two forms: session-level pacing managed by the LU component, and DLC-level pacing at the adapter component. Understanding the behavior of these pacing algorithms is important for performance tuning.

Communications Server for AIX supports two types of session-level pacing: adaptive and fixed.

- Adaptive pacing, used in LU6.2 sessions, dynamically adjusts the pacing window size based on the availability of system resources. The size of a new pacing window granted by the partner node ranges anywhere from 0 to X , where X is specified by the user's mode as *upper bound for adaptive receive pacing window*.
- In fixed pacing, the pacing window size can take only the values 0 and X , where X is specified in the mode as *receive pacing window*. The pacing windows for a session are negotiated when it is started. They then stay fixed for the life of the session. CS/AIX pre-reserves memory for the pacing windows. If memory is not available then the sessions will not be started.

Adaptive pacing is used if both conversation partners support adaptive pacing.

2.3.1 Session-Level Window Size

The session-level window size (pacing count, or PC) represents the number of RUs that can be transmitted before an acknowledgment must be received. The sender keeps track of how many RUs have been sent. The residual window size (residual pacing count, or RPC) variable indicates how many RUs are left in the current window. When a session is initialized, PC and RPC are set to 1. The first RU in every window indicates to the receiver that a new window is requested. If at any point $RPC=0$, and there are still RUs to send, the RU requesting a new window also indicates that this new window should be larger than the previous one. The receiver can grant a larger window, same-size window, or a smaller window depending on the availability of memory on the receiving machine.

The user can influence the session-level pacing algorithm by placing an upper bound on the size of the pacing window. How high to set this upper bound depends on the relative speeds of the producer and consumer, as well as on memory availability.

If the receiver can consume RUs as fast or faster than the sender can produce, then setting the *upper bound for adaptive receive pacing window* on the receiver node at larger values yields higher throughput (defined in units of KBps) and faster response time. If the receiver is slower than the sender, there is no need to set a smaller adaptive session pacing limit. Pacing will be adjusted automatically in order to avoid overwhelming the receiver.

The impact of session-pacing window size on throughput was measured in the (NN → NN) network topology. Maximum session-pacing window sizes varied over 2, 4, 8, 16, 32, and 63. This is an example of file transfer between two nodes of equal speed. Figure 3 on page 26 displays the impact of maximum session-pacing window size on throughput. The TP send buffer size was 8192 bytes and receive buffer size was 20000.

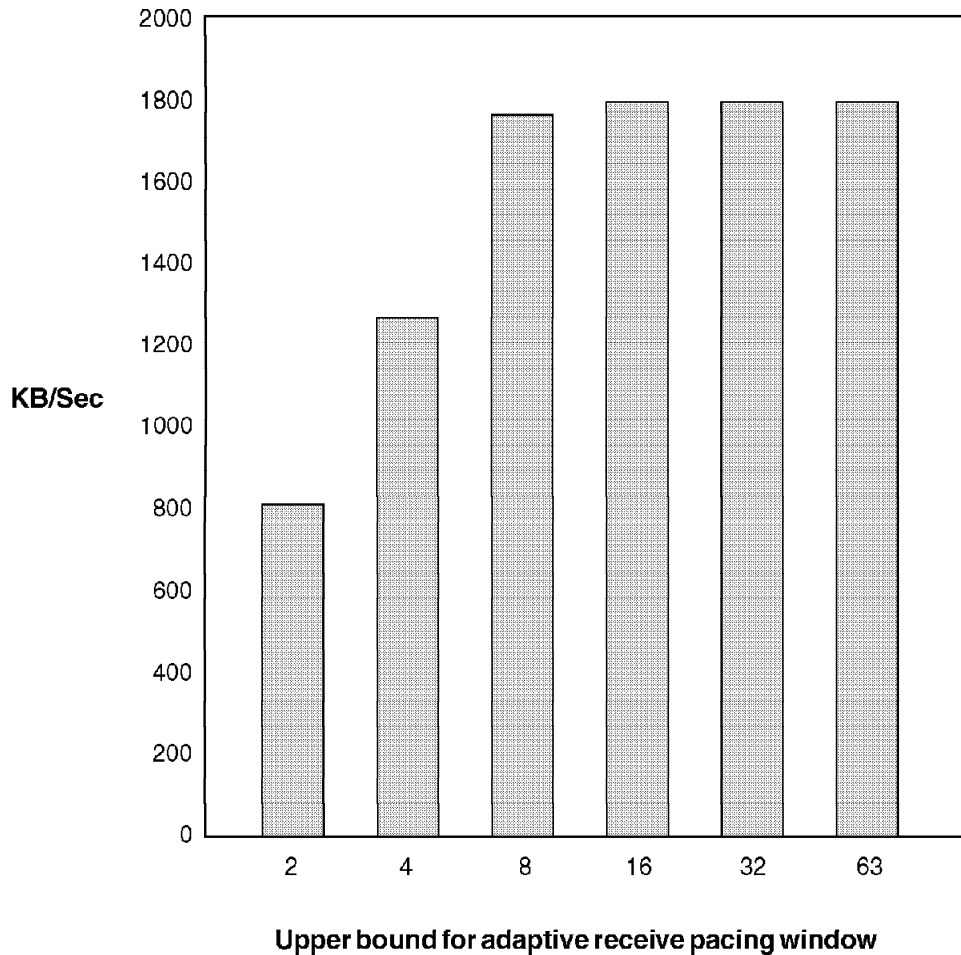


Figure 3. Throughput (KBps) as a Function of Session Pacing

2.3.2 Improving Throughput

As evident from Figure 3, significant gain in throughput is achieved when raising session-pacing window size from 2 to 4 to 8. Going above 8 approaches the level of diminishing returns. Notice, however, that the maximum throughput was achieved with *upper bound for adaptive receive pacing window* set at 63.

The data in Figure 3 can also suggest that larger pacing-window size corresponds to higher throughput. This is not always the case. Large window size will increase memory usage on the receiving node. If this gets too high, the receiving node will reduce the window size on all sessions requesting a new pacing window. This reduction of pacing-window size

alleviates the memory usage, but results in lower throughput and higher response time for those sessions whose window size was reduced. However, the node keeps track of the available memory and how much it has committed to each session. It should be able to detect indications of buffer congestion early and gradually scale back the pacing windows accordingly.

The best way to check if buffer memory is limiting performance is to look at the output from `snaadmin query_buffer_availability`. The interesting fields are:

throttle_status is an indication of whether CS/AIX is currently allowing pacing windows to grow, keeping them the same or cutting them back. The value is based on available memory and previous buffer usage patterns. The values reported are on a scale between 0 and 100, where 0 means pacing windows are allowed to grow quickly and 100 means pacing windows are being cut back to nothing. Having said that, the usual value is 30.

Anything above 30 indicates that pacing windows are being restricted by memory limits. As the value gets higher, the impact on performance increases.

buf_use_status is the current buffer congestion status of CS/AIX. This takes values of CONGESTED and UNCONGESTED. CONGESTED indicates that performance may be limited by memory availability.

max_buf_use_status indicates the worst value of `buf_use_status` since the statistics were last reset.

Another scenario in which large window size can be detrimental is when token-ring or Ethernet queues are fairly small, and the sending TP and LU generate BTUs faster than the local adapter can process them. The adapter queue fills up and further attempts to write to it are rejected. Writing to the queue is attempted again after a brief delay, causing a performance penalty. This particular bottleneck can be resolved by increasing the size of the local-adapter send queue and the remote-adapter receive queue. If the problem persists, then a reduction in maximum receive pacing will ease congestion on the adapter queues.

A reduction in session pacing is a trade-off. It can slow response time of some TPs in exchange for reduced memory usage.

The upper limit on the size of the receive-pacing window is set in the mode. To detect the current size of the send and receive window of an LU6.2 session, use the following command:

- `snaadmin query_session,list_options=DETAIL + FIRST_IN_LIST`

This command will return information on the current and maximum pacing windows.

max_send_pac_win

Maximum size of the send pacing window on this session,

cur_send_pac_win

Current size of the send pacing window on this session,

max_rcv_pac_win

Maximum size of the receive pacing window on this session,

cur_rcv_pac_win

Current size of the receive pacing window on this session,

The `snaadmin query_session` command can be qualified to a specific session. See the *IBM eNetwork Communications Server for AIX Command Reference* for more information.

2.4 Using DLC-Level Pacing

DLC-level pacing has a significant impact on performance and, if set incorrectly, can become a major throughput bottleneck. Although an adaptive-pacing algorithm, DLC-level pacing is significantly different from session-level pacing.

2.4.1 Local Area Network (LAN) Connections

To help you understand the impact of various adapter parameters on performance, this section first describes DLC-level pacing for the token-ring, Ethernet, and FDDI connections.

2.4.1.1 Understanding Response and Acknowledgment Times

When the sending adapter transmits a frame, a *response timer* is started. If the timer expires before an acknowledgment is received, a retransmission can occur and degrade performance. (This is an adapter-level acknowledgment, not a session-level acknowledgment.) The response timer value should, therefore, be greater than the total delay that the frame might encounter within the sending adapter, the network, and the receiving adapter. Normal settings for the response timer are in the 1- to 2-second range.

The receiving adapter maintains a corresponding *acknowledgment timer*, as well as a count of the number of outstanding frames that need to be acknowledged. To improve on link utilization, the receiving adapter does not necessarily acknowledge every frame but rather tries to acknowledge a group of frames with a single acknowledgment. When a frame arrives at the receiving adapter, the acknowledgment timer is started (if it is not already on) and the count of unacknowledged frames is incremented by one. An acknowledgment is sent under one of two conditions:

- The acknowledgment timer has expired.
- The number of outstanding (unacknowledged) frames reaches the maximum set by the local *Receive window count* parameter.

The expiration-time value of the acknowledgment timer on the receiver must be less than that of the response timer on the sender. This is to ensure that the sending adapter receives an acknowledgment before the expiration of its response timer. Throughput is seriously degraded if acknowledgments are sent as a result of acknowledgement timer expiration. Parameter values should be selected so that acknowledgments result from reaching the *Receive window count* value.

The sending adapter can send only a limited number of frames before it must stop and wait for an acknowledgment. This limit is set by the *Transmit window count* parameter.

The *Transmit window count* and the *Receive window count* are defined on the linkstation and port definitions with the following parameters:

Transmit window count
target_pacing_count

Receive window count
max_ifrm_rcvd

2.4.1.2 Window Count Size Guidelines

It is the relative sizes of *Receive* and *Transmit window count* values, rather than their absolute magnitudes, that significantly affect performance. To achieve high throughput, the receiver's *Receive window count* should always be less than or equal to the sender's *Transmit window count*. If the Receive window is larger than the Transmit window, it is likely the sender will halt after sending a full window while the receiver has less than a full window received; therefore, acknowledgment is not sent until the acknowledgment timer expires. It is thus important to set the *Transmit window count* on the sender to be larger than the *Receive window count* of the receiver.

For token-ring, Ethernet, and FDDI, use the following rule: the *Transmit window count* parameter on node A should always be larger than or equal to the *Receive window count* on Node B (for every two nodes A and B that are connected to one another). This holds even if A or B is not a RISC System/6000 workstation.

2.4.1.3 Adapter-Level Pacing over a Token-Ring

To test the preceding guidelines, a number of experiments were done using token-ring connections.

Note: These experiments and their results apply equally well to Ethernet and FDDI. In the first experiment, the Transmit window count on the sender side was fixed at 32, while the Receive window count on the receiver side varied from 1 to 127. Throughput of an 8MB file transfer TP was measured for each value of the Receive window count in the set (1, 2, 4, 8, 16, 32, 64, 127). Figure 4 on page 31 shows the relationship of the DLC pacing values to throughput.

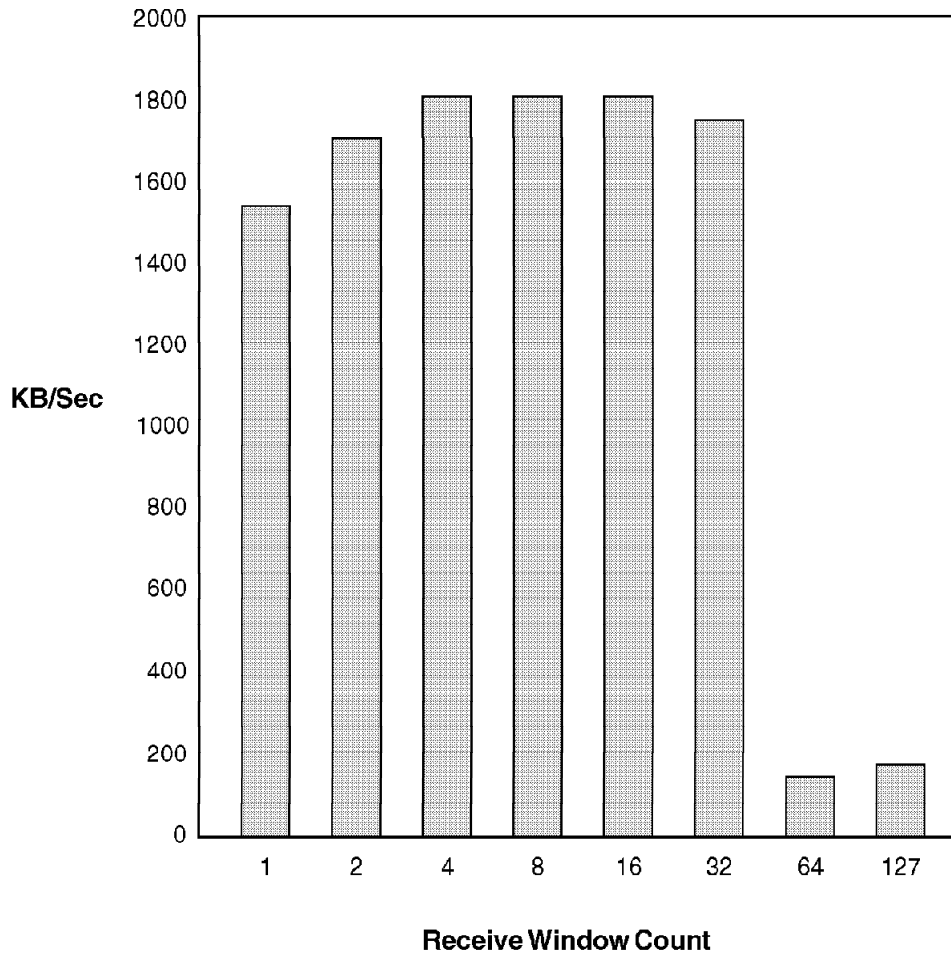


Figure 4. Ratio Test—Throughput (KBps) as a Function of DLC-Level Pacing Counts on a Token-Ring Connection

These results support the previous discussion that Receive window count on the receiver should not exceed Transmit window count on the sender. Since Transmit window count was fixed at 32, a dramatic drop-off in throughput occurs once Receive window count exceeds this value. Best throughput was achieved when Receive window count was set to 16, half the value of Transmit window count. When both were equal at 32, throughput was slightly degraded. From this experiment, the conclusion is that for best throughput, Receive window count should be set to half the value of Transmit window count.

With the ratio of these two values settled, what about their absolute magnitude? Is the combination 32:16 better than 16:8? This was tested in

another experiment fixing the ratio of Transmit window count to Receive window count at 2:1, and gradually increasing their values. Figure 5 on page 32 shows throughput as a function of the window counts.

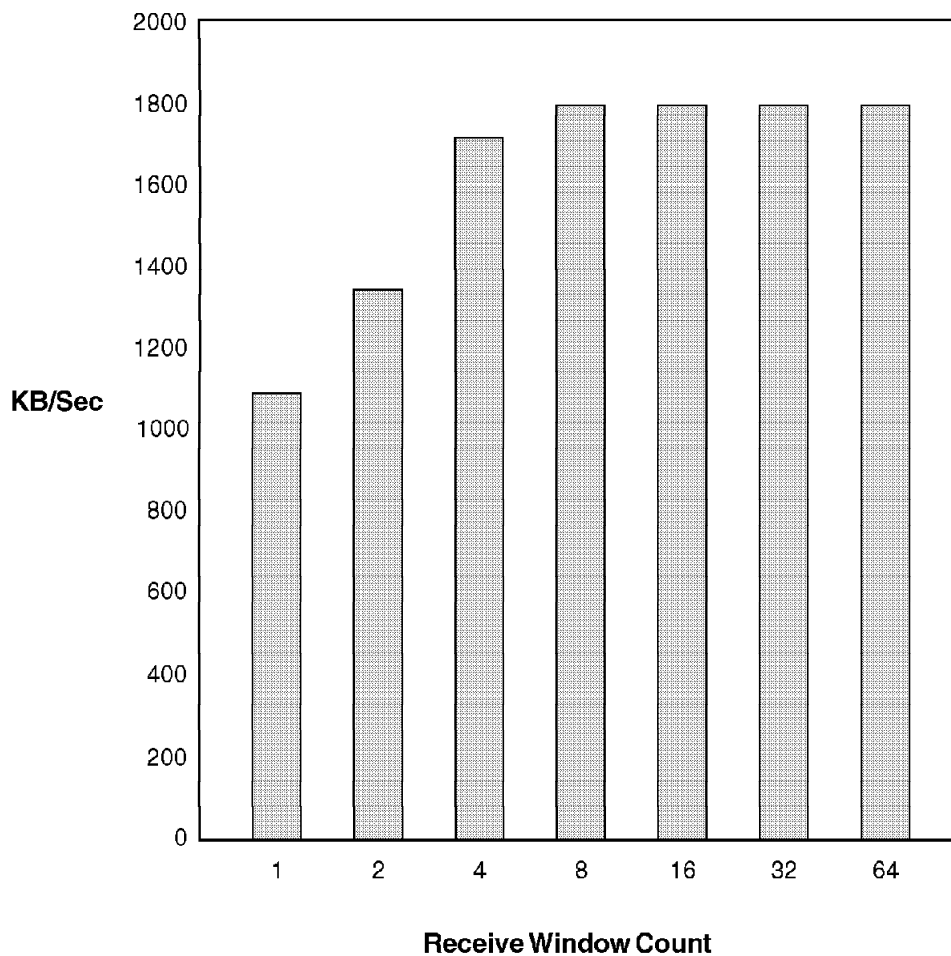


Figure 5. Magnitude Test—Throughput (KBps) as a Function of DLC-Level Pacing Counts on a Token-Ring Connection

Significant throughput gains were made up to the 16:8 (Transmit:Receive) window count values. Beyond that point, no additional throughput was achieved by doubling the window counts.

2.4.2 SDLC Connections

SDLC connections (EIA232D, smart modem, X.21, EIA422A, V.25bis, and V.35) use a slightly different DLC-level pacing algorithm. There is no Receive window count; instead, the number of unacknowledged frames that can be

sent is regulated by Transmit window count only. Also, SDLC connections distinguish between primary and secondary link stations. A link station (primary or secondary) sends frames until it empties its send queue or until it reaches the limit imposed by Transmit window count. The last frame sent is marked as such to indicate that the sender is expecting an acknowledgment.

Upon receiving this marked frame, the receiving link station acknowledges it. If the sender is the primary link station, a timer is started after sending the last frame. This timer, called *Primary repoll frequency*, indicates how long it waits for an acknowledgment. (For the secondary link station, the equivalent timer is called *Inactivity time-out*.) If an acknowledgment has not been received within that specified time, the primary link station activates a recovery procedure that degrades performance. To avoid this delay, the Primary repoll frequency and Inactivity time-out timer must allow enough time for the largest frame to travel to and from the partner link station. In a point-to-point connection, if the link speed is X bits-per-second and your largest frame is Y bits, then the Primary repoll frequency and Inactivity time-out timers should be larger than $2 * Y \div X$ seconds. For example, assume link speed of 9600 bits per second and the largest frame is 256 bytes (which is 2048 bits). The Primary repoll frequency and Inactivity time-out timers should be larger than $2 * 2048 \div 9600 = 0.426$ seconds. In this case, setting the timers at 0.5 seconds suffices.

2.5 Adapter Queues

The token-ring, Ethernet, and FDDI adapters maintain receive and send queues of finite size. This finiteness can cause a throughput and response time bottleneck under some circumstances.

2.5.1 Receive Queue Overflow

When the receive queue fills up, the adapter is not able to receive additional frames; data flow over that link, consequently, will suffer significant delays. The following factors can cause this situation:

- Slow CPU speed. The process reading from the receive queue does not consume queue items fast enough.
- Heavily utilized CPU. The process reading from the receive queue does not get to execute often enough.
- The remote adapter transmits more frames than the local adapter can handle. The value in the Transmit window count field can be too large in this case.

2.5.2 Send Queue Overflow

Once the receive queue overflows, the remote adapter is not granted permission to send additional frames. This, in turn, can cause the remote send queue to overflow if the sending LU places BTUs on the send queue faster than the adapter can send them. Another cause of send queue overflow can be a congested LAN. The sending adapter may not be able to transmit often enough on a heavily utilized LAN.

2.5.3 Adjusting the Sizes of Adapter Queues

When the LU attempts to write to a full adapter queue, the write operation fails and is retried by the LU after a brief delay. If the size of the adapter queue is a limiting factor, an error message is logged in the system error log. (You can view the system error log by issuing the `errpt -a` command.) You can adjust the size of the adapter queues to resolve this particular bottleneck. Instructions for modifying these values are given in 2.6.1, “Modifying SNA and System Parameters.” In testing, the default queue size (30) was a limiting factor for 16 Mbps token-ring, 4 Mbps token-ring, and Ethernet. It is therefore recommended to increase the size of the adapter queues to their maximum value of 150 (send and receive).

2.6 Network Tuning Decisions

This section outlines the factors to consider when tuning a network to certain performance specifications. It also explains how to display and modify SNA and system parameters and summarizes tuning recommendations.

Network tuning consists of a number of decisions that combine competing factors. This decision process imposes trade-offs between various important considerations such as link utilization, CPU utilization, memory usage, throughput, and response time. This book highlights the trade-offs that are inherent in specific system and SNA parameter selection. If throughput is the highest priority, then a specific set of parameter values should be selected. If CPU utilization is a higher priority, then an entirely different set of parameter values can be necessary. The choice of parameter values depends on the performance requirements of each network. Typically, higher throughput comes at the expense of higher CPU utilization and memory consumption.

2.6.1 Modifying SNA and System Parameters

This section describes how to modify and display the various SNA and system parameters discussed in 1.4.4, “Memory and Storage Requirements for Operation” on page 10.

Real memory

To display real memory size in kilobytes for a RISC System/6000 workstation, issue the following command:

```
lsattr -El sys0 | grep mem
```

Alternatively, you can display the same information in a different format by using the `svmon -G` command, which reports memory and paging size in pages. Each page is 4 KB.

Paging space

Issue `lpsps -a` or `svmon -G` to obtain the availability and usage of paging space on a RISC System/6000 workstation.

STREAMS buffers

Issue `netstat -m` to obtain statistics on STREAMS buffers.

Mbufs

Issue the following commands to display or modify network parameters:

- `netstat -m` to display statistics on system mbuf usage.
- `no -a` to display the current parameter values.
- `no -o` to modify *thewall*. For example, `no -o thewall=8192` will set thewall at 8192 KB.

Modifications performed by the `no` commands last only until the next system reboot. Use `chdev -l sys0 -a maxmbuf=<value>` to modify thewall permanently. If thewall is modified through SMIT, the new value is permanent and persists across reboot. To modify thewall permanently, invoke SMIT and select the following SMIT panels:

- System Environments
- Change / Show Characteristics of Operating System

The field in the dialogs that corresponds to thewall is called "Maximum Kbytes of real memory allowed for MBUFS."

Session pacing

The upper limit on the window size in session-level pacing is set in the mode. The corresponding field name is "Upper bound for adaptive receive pacing window."

DLC-Level pacing

The Transmit window count, Receive window count, Primary repoll frequency, and Inactivity timeout fields are in the SNA DLC definition.

RU Size

RU size is negotiated by the session partners. If you want to set RU size to a specific value, set the Maximum RU size and Minimum RU size fields to the same value, and make sure that it falls in the

acceptable range for the session partner. The Maximum RU size and Minimum RU size fields are also in the mode.

Adapter Queues

To modify adapter queues, invoke SMIT and select the following panels:

- Devices
- Communications
- Token-ring Adapter
- Adapter (Select the desired adapter from the pop-up menu.)
- Change / Show Characteristics of token-ring adapter

Modify the TRANSMIT queue size and RECEIVE queue size fields to the values you need. Also modify the “Apply change to DATABASE only” field to yes. After these changes are made, reboot the system to have them to take effect.

BTU Size

Changing BTU size can help performance if you are trying to match incoming and outgoing BTU sizes on an NN or gateway node. In situations not involving a network node or gateway, the system-defined defaults are sufficient. A user-defined maximum BTU size can be specified in the SNA DLC definition. Select yes for the “User-defined maximum I-field size?” and enter a new size in the next field.

Memory usage

The following commands can be used to monitor and control CS/AIX memory usage:

- `snaadmin query_kernel_memory_limit`
- `snaadmin set_kernel_memory_limit`
- `snaadmin query_buffer_availability`
- `snaadmin set_buffer_availability`

See the *IBM eNetwork Communications Server for AIX Administration Command Reference* for further information on these commands.

2.6.2 TN Server Tuning Considerations

There are several performance tuning issues that must be addressed in regard to the TN Server function. Specifically, two separate connections must be tuned: The IP connection between the client and server machine, and the SNA connection between the server and the host. In addition, the tuning of the system’s mbuf pool and adapter queues is common to both products as well as IP and SNA. All these issues will be addressed separately in the following sections. The information supplied in this document resulted largely from various tests conducted at our networking lab. Other sources were the documents listed below:

- *AIX Performance Monitoring and Tuning Guide*

- “TCP/IP Performance Tuning for the RISC System/6000” published in *AIXpert* (February 1994) by Rudy Chukran. This article can be found on the Web at <http://www.developer.ibm.com/library/aixpert/feb94/feb94toc.html>.

2.6.2.1 IP Tuning

Mbuf pool parameters as well as adapter queue sizes affect both SNA and IP transport performance. Some parameters, however, are strictly IP related. The `no -a` command displays a few of these TCP/IP variables that have impact on IP performance. These are `sb_max`, `tcp_sendspace`, `tcp_recvspace`, and `ipqmaxlen`. An additional significant parameter was the TCP/IP token-ring packet size. In the test environment, the following parameter values were optimal for maximizing file transfer throughput:

- `sb_max = 131072`
- `tcp_sendspace = 65536`
- `tcp_recvspace = 65536`
- `ipqmaxlen = 150`
- TCP/IP token-ring packet size = 4096

For more information on TCP/IP as well as mbuf pool tuning on the RS/6000, see *AIX Performance Monitoring and Tuning Guide*. Another excellent source is the article published in *AIXpert* titled “TCP/IP Performance Tuning for the RISC System/6000.”

2.6.3 Summary of Tuning Recommendations

The session-level pacing parameter can significantly influence all key performance metrics. The test results show that setting session-level pacing to 8 provides a combination of high throughput with reasonable levels of CPU and memory utilization. The following list summarizes the various tuning recommendations mentioned throughout this book:

- Set the upper bound on session level pacing to 8.
- Use large RUs to improve throughput of file transfer.
- Set the *Transmit_Window_Count* on node A to be equal or greater than *Receive_Window_Count* on node B, if A and B are the endpoints of a communication link. The same rule applies in the reverse direction: *Transmit_Window_Count* on B should be equal or greater than *Receive_Window_Count* on A. This applies to token-ring, Ethernet, and FDDI connections.
- Set the size of the token-ring, Ethernet, or FDDI RECEIVE and TRANSMIT queues to 150.
- If possible, match the incoming and outgoing BTU size on a network to prevent BTU segmentation.

- In short transactions, use allocate-listen remote TPs to significantly improve response time and transaction rate. The use of dynamically attached remote TPs incurs the cost of loading and starting the remote TP.
- Use large TP buffers for sending and receiving a file transfer to improve throughput.

Appendix A. Common Performance and Tuning Problems

This appendix describes the most common problems associated with the tuning of Communications Server for AIX.

When performance problems are encountered, check the following parameters (which are listed in order of importance and frequency of occurrence):

1. Are the Transmit window count and Receive window count values properly set in the link station and port definitions? These counts are not available through the Motif or SMIT interface but can be seen in the `define_port type_ls` and `define_port type_port` definitions. The receive window count is `max_ifrm_rcvd` in the port and link station definitions. The transmit window count is `target_pacing_count` in the port and link station definitions.

The default MAXOUT on NCP is 7. If `max_ifrm_rcvd` on the SNA link station definition on the RISC System/6000 workstation is larger than 7, very poor throughput will be observed. If you do not know the value on the partner node, start by setting `target_pacing_count` to 127 and `max_ifrm_rcvd` to 1. These are safe values, as 1 and 127 are the minimum and maximum values, respectively. These values guarantee that the partner's Transmit window count field value is higher than your Receive window count, and that your Transmit window count is higher than the partner's Receive window count field value.

2. Are the adapter queues large enough?
3. Is the wall large enough?
4. Is session pacing large enough?
5. Is the RU size large enough?

If the size of $(RU+RH+TH+LH+LT)$ is larger than the FRAME size, you will need to do segmenting. In order to make segmenting efficient, a proper value for the RU size must be selected. The goal is to fill up each frame as much as possible, since it is better to have a few large frames than to have many small frames. A good value for the RU size is the following:

$$I * (FRAMEsize - LT - LH - TH - RH)$$

where I is an integer. When $I=1$, no segmenting is done. See Figure 1 on page 18 for the relationship between RU, RH, TH, LH, LT, and FRAME size.

Appendix B. Performance Tests

This appendix presents various performance-related experiments, which were customized to illustrate the effects of different conditions on system performance. It is not reasonable to expect that the same conditions can be or should be duplicated in a customer environment. Instead, the experimental results are used to highlight the principles of efficient data flow, not to set a goal or a standard.

A major portion of the study was dedicated to evaluating the impact of various SNA and system parameters (discussed in 2.6.1, "Modifying SNA and System Parameters" on page 34) on performance. The tests in this study were done using token-ring links to obtain representative data.

B.1 Functions Tested

Performance data was collected for the following Communications Server for AIX functions:

- APPC/CPI-C
- APPN
- SNA Gateway
- AnyNet
- SNA Channel: Block Multiplexer and ESCON
- TN Server

The setup for each of these tests, end result, and observations are included as part of this document.

B.1.1 APPC

For these tests two RISC/6000 J30 SMP machines with 256 MB of memory were used. The mbuf wall was set to 16384 on both test machines. The two RISC/6000 were on an isolated 16 MB token-ring LAN. The objectives of the tests were:

- Measure the performance of the CS/AIX 4.2 APPC API that was ported to Communications Server for AIX. For this test, LU6.2 programs sendto and rcvfrom were used.
- Compare the performance of the CPI-C interface on Communications Server for AIX using AFTP.

The following measurements were based on doing a file transfer of a one megabyte (1.048 MB) text file to the Machine-Under-Test (MUT). The RU size used was 4 KB (3840).

APPC - duration(sec) - data rate(KB/s)	.710 1455.90
CPI-C - duration(sec) - data rate(KB/s)	.770 1336.90

B.1.1.1 CPU Utilization

The following test measured the impact on throughput (1 MB file transfer) as the number of LU-LU sessions running traffic increases. The LU6.2 transaction programs AFTP and AFTPD were used to execute the file transfers and APING and APINGD were used to generate the background sessions and traffic.

LU Sessions for background traffic	CPU Idle %	Response Time Sending a 1 MB File
0	99	.640 seconds
16	90	.644 seconds
32	80	.700 seconds
64	65	.730 seconds
128	50	1.010 seconds
256	40	1.000 seconds

The conclusion drawn here was that, as expected, increasing background traffic slightly degraded throughput. However, it was minimal.

B.1.2 APPN

This test measures Intermediate Session Routing (ISR) throughput. Three nodes were used, two end nodes and a network node. One end node was an RS/6000 Model 350 and the other was a Model 59H. The network node was a RS/6000 43P-140. Using 'AFTP', the test involved transferring a 1 MB file between the two end nodes via the network node. Background traffic, generated using TPNS was gradually intensified and the effect on throughput was measured. The five best of six runs were averaged to yield the results shown below:

Background Traffic (Sessions)	Throughput in Seconds
0	1.29
40	1.40
80	1.42

Background Traffic (Sessions)	Throughput in Seconds
160	1.40
800	1.44

The increasing number of background sessions with data traffic had a negligible effect on throughput. CPU utilization, memory, and paging space on the network node stayed constant.

B.1.3 SNA Gateway

The purpose of this test was to measure the throughput (uploads and downloads) through an SNA gateway while varying both the RU size and emulator (in this case, HCON) I/O buffer size. An RS/6000 Model 43P-140 machine with 128 MB of memory was used as the gateway node. The downstream RS/6000 was a Model 250 running CS/AIX 4.2 and HCON. The HCON fxfer program was used to perform the file transfer from the downstream through the gateway to the host. The results are shown below:

Table 1.

RU Size:	Upload			Download		
	HCON I/O Buffer			HCON I/O Buffer		
	2K	4K	32K	2K	4K	32K
2048 4096	24s	16s	7s	24s	12s	7s
8192 16 KB	25s	17s	7s	25s	14s	6s
32 KB	25s	17s	8s	25s	15s	6s

B.1.3.1 CPU Utilization

CPU utilization on the gateway was also measured. With 1524 LU-LU sessions running steady data traffic through the gateway, the CPU usage was at 68%.

Similar to a network node, gateway also performs BTU segmentation if the BTU on the incoming adapter is larger than the BTU on the outgoing adapter. If possible, BTU size on the incoming and outgoing adapters should be made equal. Segmentation will increase mbuf consumption on the gateway node as well as CPU utilization, which translates into degraded response time on the sessions running through the gateway.

B.1.4 AnyNet

The objective of this test was to compare performance among the different variations of AnyNet (APPC/IP, Sockets/SNA) vs. performance in a native SNA or IP network. Two RISC/6000 J30 SMP machines with 256 MB of memory were used. The mbuf wall was set to 16384 on both test machines. The two RISC/6000s were on an isolated 16 MB token-ring LAN. The throughput of transmitting a 1 MB text file was measured. The objectives of the tests were to show comparative performance numbers for the following:

- APPC applications running in their native SNA network
- APPC applications running in an IP network (by using APPC over TCP/IP)
- Sockets applications running in their native IP network
- Sockets applications running in an SNA network (by using Sockets over SNA)

The results of the test were as follows:

APPC/SNA - duration(sec) - data rate(KBps)	.710 1455.90
native CPIC/SNA - duration(sec) - data rate(KBps)	.770 1336.90
APPC/IP - duration(sec) - data rate(KBps)	.950 to 3.540 1077.89 to 289.27
SOS/SNA - duration(sec) - data rate(KBps)	.85 1053.40
native sockets/IP - duration(sec) - data rate(KBps)	same same

In these measurements, GET was used to transfer a file instead of PUT, since FTP “cheats” on a PUT by taking the final timestamp immediately after completion of the last write with no knowledge of the data actually getting to the destination. In fact, the data may not have even left the local box. That problem is circumvented by using GET since the measurement will be completed only after the data has arrived at the destination.

B.1.5 SNA Channel: Block Multiplexer and ESCON

The channel solution expands the RISC System/6000 connectivity to include a direct channel attachment to a host over ESCON and BLKMUX adapters. This document is not a complete performance study but a summary of the performance test experiences. There are facets to this issue that are yet to be inspected. The MVS system was not configured for performance and could not be adjusted for testing purposes, since this host is shared by thousands of users. It would not be surprising to see performance measurements on your premises exceeding those reported here. Also, this report focuses on the throughput of file transfer applications. The tuning recommendations are meant to improve file transfer performance. Other workload types such as interactive traffic were not investigated.

Both the RISC System/6000 workstation and the MVS host need to be tuned to enhance channel performance:

- On the MVS side, the following parameters are most relevant:

RU

Request unit. Contains user data. Define your RU size to be as large as possible.

PIU

Path information unit. Contains the RU (or a portion of it) plus various headers. The channel frame consists of a single PIU plus link headers.

IOBUF

Used for input/output data. Every PIU that enters or leaves VTAM resides in an IOBUF. This buffer must be defined to be as large as the largest PIU to cross the channel connection.

MAXBFRU

Specifies the number of buffer units that VTAM uses to receive data from the physical unit. Always code the MAXBFRU so that the IOBUF size multiplied by MAXBFRU is equal to or a multiple of the maximum RU size used by the RISC System/6000 workstation. VTAM default is 1.

DELAY

Specifies the time in milliseconds that VTAM should wait before transmitting buffered data to the physical unit. To enhance throughput, set this value to 0.

PACING

Use the PACING field of the PU macro to determine how VTAM is to pace the flow of data from the boundary node. This node performs pacing for the channel-attached RISC System/6000 workstation to the

secondary logical unit (SLU). When the SLU and primary logical unit (PLU) are in the same domain, the pacing value is ignored. VTAM default is 1.

VPACING

Use the VPACING field of the PU macro to determine how VTAM is to pace the flow of data from the boundary node that contains the PLU to the boundary node that performs pacing for the channel-attached RISC System/6000 workstation. VTAM default is 1.

For more information about IOBUF, MAXFBRU, and DELAY parameters, see the sections describing coattailing in *VTAM V4R4 Network Implementation Guide*. For more information about PACING and VPACING parameters, see the sections describing pacing and flow control, also in *VTAM V4R4 Network Implementation Guide*.

- On the RISC System/6000 workstation side, the following parameters are most relevant:

thewall

An AIX parameter that specifies the maximum amount of memory that may be used for communication buffers (mbufs). This should be set to 25% of memory.

Session level pacing

The mode profile specifies the maximum value of the session pacing window. A value of 32 should suffice.

Channel Queue Depth

The size of the channel DLC queue. Should be set to the maximum.

For more information on how to set these parameters, look in the *SNA Server for AIX Channel Connectivity User's Guide*.

Throughput was measured in one way: memory-to-memory by using APING. The numbers obtained from APING are a good indicator of channel throughput, as APING avoids bottlenecks that are related to reading from and writing to disk. Table 2 on page 47 describes the APING throughput values that were measured on a RISC System/6000 Model J30 SMP (4-way) and various parameters including IOBUF, which is a VTAM parameter. The actual throughput values are in the last two rows corresponding to the IOBUF values used by ESCON and BLKMUX. For MPC, the IOBUF set to any value does not affect how it performs. MPC performs similar to the IOBUF size equal to 4 KB which will be referenced to the highest IOBUF as defined in Table 2 on page 47. Parameter values for a specific cell throughput are given by the row and column corresponding to that cell. The SPECint95 is a measure of machine speed that is more indicative than simply looking at the clock speed. It should be noted that the tests of the J30 were done using

one MVS system. The MVS system was the difference in the results, not the superior speed of the J30 model. The bottleneck of the AFTP test was identified as the MVS system configuration. Since the testers did not have access to those configurations, they had to test within those constraints. Further work is underway to clearly identify and resolve the bottlenecks encountered with AFTP on the MVS.

The size of the IOBUF should normally make an impact, but it did not in this case because it was masked. By using APING, all bottlenecks related to reading from and writing to MVS storage were circumvented.

Note: APING has the option to do round trip, upload and download, data during its execution, as well as setting the option to do only just uploads or downloads. There are three APING Throughput Values tables, which show values obtained over ESCON and MPC running APING with the option to do upload, download, and round trip as described.

Table 2. APING Throughput Values - Round Trip (uploads and downloads)

Channel Type →	ESCON	MPC	BLKMUX
RISC model	J30	J30	53H
memory	256 MB	256 MB	64 MB
thewall	16 MB	16 MB	16 MB
SPECint95	72	72	28.5 (SPECint92)
iobuf=1256	941 KBps		530 KBps
iobuf=3968	1333 KBps	1292 KBps	1054 KBps

Table 3. APING Throughput Values - Uploads

Channel Type →	ESCON	MPC	BLKMUX
iobuf=1256	653 KBps		530 KBps
iobuf=3968	1066 KBps	1882 KBps	n/a

Table 4. APING Throughput Values - Downloads

Channel Type →	ESCON	MPC	BLKMUX
iobuf=1256	1481 KBps		530 KBps
iobuf=3968	2184 KBps	2163 KBps	n/a

In the APING test, the RISC model as well as IOBUF size had a significant impact on throughput. Better throughput is expected with faster RISC models such as J50 SMP, 595 UP, E30 UP, and more.

APING is a member of the APPC Application Suite. The host was an MVS guest on a two-way VM. CPU utilization on VM was around 60%. Other parameters were:

- VTAM V4R4
- RU size = 32K
- MAXBFRU = 30
- pacing = 60
- VPACING = 60
- DELAY = 0

To summarize, it was concluded that the channel solution is capable of very high throughput when tuned properly. It is largely dependent on the RISC model and VTAM's IOBUF size. Bottlenecks were encountered when trying to take advantage of those throughput values using AFTP. The problem was narrowed down to the MVS system configuration. More work is underway.

B.1.6 TN Server

The TN Server function is used to connect downstream clients using TCP/IP to a host system. It uses the TCP/IP protocol to communicate with the downstream clients and SNA to communicate with the host.

This section contains a description of the TN Server performance tests that were run along with the results. A TN3270E client running the WRQ Reflection emulator was connected to the TN Server. An upload/download of a 1 MB file through the TN Server to the host was performed using IND\$FILE. Simultaneously with the file transfer, background traffic was slightly varied using TPNS and RU sizes were varied to assess the effect on throughput. The RS/6000 machine running TN Server was a 43P model with 128 MB of main memory. It was connected to the host via a channel-attached 3745. The test environment was not dedicated to the test, and other unrelated traffic took place during testing.

B.1.6.1 Test Results

Table 5.

Neg. Bind RU Sizes	Upload			Download		
	TNC I/O Buffer			TNC I/O Buffer		
	2K	4K	32K	2K	4K	32K
2048 4096	30s	19s	10s	1:46m	55s	14s
8192 16K	31s	21s	10s	1:52m	59s	18s
1.32K	31s	21s	9s	1:47m	59s	19s

When 1523 LU-LU sessions were started with traffic, the upload time of a 1 MB file took 1:24 seconds. When the load was increased to 2000 LU-LU sessions with traffic, the upload took 1:35 seconds. The number of sessions running traffic clearly has an impact on throughput, but our tests showed that the degradation was no more than what would be reasonably expected. As far as CPU usage, when the load was scaled up to 2000 sessions with traffic, the CPU usage was 85%.

Appendix C. Special Notices

This publication is intended to help system administrators and application programmers who use Communications Server for AIX to tune their networks and workstations. The information in this publication is not intended as the specification of any programming interfaces that are provided by Communications Server for AIX. See the PUBLICATIONS section of the IBM Programming Announcement for Communications Server for AIX for more information about what publications are considered to be product documentation.

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee

that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

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

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

IBM	eNetwork
AIX	RISC System/6000
APPN	AnyNet
VTAM	Portmaster
AIXwindows	RS/6000
ESCON	DB2
DATABASE 2	SP2
OS/2	

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

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

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

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

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

Appendix D. Related Publications

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

D.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see "How to Get ITSO Redbooks" on page 55.

- *IBM eNetwork Communications Server for AIX: Understanding and Migrating to Version 5: Part 1 - Configuration and New Features*, SG24-5215

D.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
Lotus Redbooks Collection	SBOF-6899	SK2T-8039
Tivoli Redbooks Collection	SBOF-6898	SK2T-8044
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
RS/6000 Redbooks Collection (PDF Format)	SBOF-8700	SK2T-8043
Application Development Redbooks Collection	SBOF-7290	SK2T-8037

D.3 Other Publications

These publications are also relevant as further information sources:

- *IBM eNetwork Communications Server for AIX General Information*, GC31-8584
- *IBM eNetwork Communications Server for AIX Installation and Migration Guide*, SC31-8585
- *IBM eNetwork Communications Server for AIX Quick Beginnings*, GC31-8583
- *IBM eNetwork Communications Server for AIX Administration Guide*, SC31-8586

- *IBM eNetwork Communications Server for AIX Administration Command Reference*, SC31-8587
- *IBM eNetwork Communications Server for AIX CPI-C Programmer's Guide*, SC31-8591
- *IBM eNetwork Communications Server for AIX APPC Programmer's Guide*, SC31-8590
- *IBM eNetwork Communications Server for AIX CSV Programmer's Guide*, SC31-8592
- *IBM eNetwork Communications Server for AIX MS Programmer's Guide*, SC31-8594
- *IBM eNetwork Communications Server for AIX NOF Programmer's Guide*, SC31-8595
- *IBM eNetwork Communications Server for AIX Diagnostics Guide*, SC31-8588
- *IBM eNetwork Communications Server for AIX AnyNet Guide to APPC over TCP/IP*, GC31-8598
- *IBM eNetwork Communications Server for AIX AnyNet Guide to Sockets over SNA*, GC31-8597
- *IBM eNetwork Communications Server for AIX APPC Application Suite User's Guide*, SC31-8596
- *IBM eNetwork Communications Server for AIX Channel Connectivity User's Guide*, SC31-8219
- *AIX Performance Monitoring and Tuning Guide*, SC23-2365
- *VTAM Version 4 Release 4 Network Implementation Guide*, SC31-8370

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at <http://www.redbooks.ibm.com/>.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Redbooks Web Site on the World Wide Web**

<http://w3.itso.ibm.com/>

- **PUBORDER** — to order hardcopies in the United States

- **Tools Disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLCAT REDPRINT
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get BookManager BOOKs of redbooks, type the following command:

```
TOOLCAT REDBOOKS
```

To get lists of redbooks, type the following command:

```
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
```

To register for information on workshops, residencies, and redbooks, type the following command:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
```

- **REDBOOKS Category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (<http://www.redbooks.ibm.com/redpieces.html>). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

In United States:	IBMMAIL usib6fpl at ibmmail	Internet usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications Publications Customer Support P.O. Box 29570 Raleigh, NC 27626-0570 USA	IBM Publications 144-4th Avenue, S.W. Calgary, Alberta T2P 3N5 Canada	IBM Direct Services Sortemosevej 21 DK-3450 Allerød Denmark
--	--	--

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States) or (+1)001-408-256-5422 (Outside USA)** — ask for:

Index # 4421 Abstracts of new redbooks
Index # 4422 IBM redbooks
Index # 4420 Redbooks for last six months

- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com/
IBM Direct Publications Catalog	http://www.elink.ibm.link.ibm.com/pbl/pbl

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (<http://www.redbooks.ibm.com/redpieces.html>). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

List of Abbreviations

BIU	Basic information unit	Mbps	Megabits per second
BLU	Basic link unit	Mbuf	Memory buffer
BTU	Basic transmission unit	NN	Network node
DLC	Data link control	PIU	Path information unit
EN	End node	RH	Request/Response header
KB	Kilobyte	RU	Request/Response unit
LH	Link header	SMP	Symmetric Multiprocessor
LT	Link trailer	TH	Transmission header
LU	Logical unit	TP	Transaction program
MB	Megabyte		

Index

A

- abbreviations 59
- acronyms 59
- adapter queues 33, 34, 36, 39
- APPC
 - memory requirements 13
 - performance measurements 41
- APPC over TCP/IP
 - memory requirements 13
 - overview 4
 - performance measurements 44
- APPN ISR
 - memory requirements 14
 - performance measurements 42

B

- bibliography 53
- bottlenecks for traffic 20

C

- channel performance and tuning 45
- commands
 - chdev -l 35
 - lsattr -EL sys0 35
 - netstat -m 21, 35
 - no -a 35
 - no -o 35
 - snaadmin query_buffer_availability 23, 27, 36
 - snaadmin query_kernel_memory_limit 23, 36
 - snaadmin query_session 28
 - snaadmin set_buffer_availability 23, 24, 36
 - snaadmin set_kernel_memory_limit 24, 36

D

- data flow 17

I

- installation requirements for RISC System/6000 workstations
 - DLC support 9
 - link hardware 8
 - memory and storage 9
 - software 9

M

- maxmbuf 35
- mbufs
 - allocation 21, 22, 35
 - overview 10
- memory requirements for
 - 3270 dependent LU session 12
 - AnyNet 13
 - APPC 13
 - APPN ISR sessions 14
 - node installation 9
 - node operation 10
 - SNA gateway 13
 - TN Server 13
- modifying SNA and system parameters
 - adapter queues 36
 - BTU size 36
 - DLC-level pacing 35
 - mbufs 35
 - memory usage 36
 - paging space 35
 - real memory 35
 - RU size 35
 - session pacing 35
 - STREAMS buffers 35

N

- naming conventions 15
- network ID registration 15
- network planning
 - determining configuration 5
 - identifying function 5
 - overview 4

P

pacing

- adaptive pacing 24, 26, 35, 39
- checking memory usage 26, 27
- DLC-level 28, 35
 - Ethernet 30
 - FDDI 30
 - SDLC 32
 - token ring 30
- fixed pacing 24
- session level pacing 25, 37, 46
- window size 25

performance problems 39

personnel requirements 6

R

receive window count 24, 29, 30, 31, 32, 33, 35, 36, 37, 39

RISC System/6000 models 7

S

segmentation of APPN networks 15

SNA Gateway

- memory requirements 13
- overview 4
- performance measurements 43

Sockets over SNA

- memory requirements 13
- overview 4
- performance measurements 44

SPECint95 ratings 14

STREAMS buffers

- allocation 21, 22, 23, 35
- overview 10

system requirements for

- installing RISC System/6000 workstation
 - DLC support 9
 - link hardware 8
 - memory and storage 9

operating Communications Server for AIX products

- disk storage 11
- mbufs 10
- memory storage 11
- paging storage 11

system requirements for (*continued*)

operating Communications Server for AIX

products (*continued*)

- processes 12
- STREAMS buffers 10

T

thewall 21, 23, 35, 39, 46

TN Server

- memory requirements 13
- overview 4
- performance measurements 48
- tuning 36

transmit window count 29, 30, 31, 32, 33, 35, 36, 37, 39

transport media 7

tuning Communications Server for AIX products

- adapter queues 33
- bottlenecks 20
- data flow 17
- DLC-level pacing 28
- LAN connections 28, 29
- mbufs 21, 35
- overview 17
- recommendations 37
- SDLC connections 32
- session-level parameters
 - pacing 24
 - window size 25

SNA and system parameters

- adapter queues 36
- BTU size 36
- DLC-level pacing 35
- mbufs 35
- memory usage 36
- paging space 35
- real memory 35
- RU size 35
- session pacing 35
- STREAMS buffers 35
- throughput 26
- TP data transformation 18

tuning problems 39

U

U-Shaped sessions 20

ITSO Redbook Evaluation

IBM eNetwork Communications Server for AIX: Understanding and Migrating to Version 5:
SG24-2136-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at <http://www.redbooks.ibm.com>
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Which of the following best describes you?

Customer **Business Partner** **Solution Developer** **IBM employee**
 None of the above

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes____ No____

If no, please explain:

What other redbooks would you like to see published?

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

**IBM eNetwork Communications Server for AIX: Understanding and Migrating to Version 5:
Part 2 - Performance**

SG24-2136-00

**SG24-2136-00
Printed in the U.S.A.**

