IBM CICS Universal Client
Version 7 Release 0 Modification 1

Migrating an SNA connection from TCP62 to Enterprise Extender

GC34-6889-00

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 53.

Edition notice

This edition applies to Version 7, Release 0, Modification 1 of the CICS Universal Client, program number 5724-I81, and to all subsequent versions, releases, and modifications until otherwise indicated in new editions. Make sure that you are using the correct edition for the level of the product.

Preface

This book tells you how to migrate a CICS® Universal Client for Windows® TCP62 configuration to an Enterprise Extender connection.

Before you can use this book, you must have a good understanding of IBM® CICS Universal Client for Windows, IBM CICS Transaction Gateway, IBM Communications Server, Linux® and Microsoft® Windows software.

This book is written for anyone who is responsible for migrating an SNA connection from TCP62 to Enterprise Extender.

You can use this book in four ways:

- As an overview of how to migrate a CICS Universal Client for Windows TCP62 configuration to an Enterprise Extender connection. The section "Enterprise Extender" on page 2 of the book provides this information.
- As a straightforward step by step guide to performing the configuration tasks to migrate from the TCP62 protocol to an Enterprise Extender connection. The table "Configuration information" on page 6 provides links for these steps.
- As a guide to collecting the necessary parameter information to migrate from the TCP62 protocol to an Enterprise Extender connection. The table "Cross-product configuration definitions" on page 39 provides guidance on the key definitions that are required.
- As a reference guide for the configuration files to edit when migrating from the TCP62 protocol to an Enterprise Extender connection. The appendix section of this book provide supporting "Example configurations," on page 39.

The contents of this book are applicable to the IBM CICS Transaction Gateway and the IBM CICS Universal Client products.

Contents

Figures

Tables

Chapter 1. Introduction

The IBM CICS® Universal Client TCP62 protocol enables a client application to use an LU6.2 SNA connection to a CICS region, encapsulating the SNA data in TCP/IP packets. This encapsulation works with the AnyNet® component of VTAM® provided in Communication Server for z/OS®. However, with the announced withdrawal of AnyNet in z/OS V1R8, migration from TCP62 to Enterprise Extender connections will be necessary if you cannot migrate to the use of native TCP/IP connections.

This migration guide tells you how to migrate a CICS Universal Client for Windows TCP62 configuration to an Enterprise Extender connection using a Remote API Client connection to a Communications Server for Linux on System z™. Then you configure the link from Communications Server on Linux to connect to VTAM using an HPR/IP connection.

The contents of this guide are also applicable to the IBM CICS Transaction Gateway product.

Enterprise Extender

Enterprise Extender, also known as HPR/IP, is part of IBM Communications Server, which enables IP links between configured SNA nodes for LU6.2 sessions. It is similar to AnyNet and provides additional support in terms of APPN High Performance Routing (HPR). It is easier to configure than AnyNet.

You can choose from two principal topologies when migrating from TCP62 to Enterprise Extender:

- Communications Server Remote API Client (split stack)
- Full Communications Server installed on the local computer (full SNA stack)

The key differences between these two are as follows:

1. Split stack - Remote API Client:

- Supports TCP/IP, HTTP and HTTPS connection options.
- Provides a more efficient model, in terms of host resources, when large numbers of SNA clients connect to the data center.
- Provides ability to consolidate SNA network traffic within the data center.
- Allows freedom of choice for connections into VTAM, meaning that the VTAM node can continue to run as a subarea network if necessary, without migration to APPN.
- Has a lightweight install footprint on the client workstation.
- Uses TCP/IP connections between API clients and IBM Communications Server nodes.

2. Full SNA stack - Communications Server:

- Provides a simpler and more efficient SNA configuration, in terms of host resources, when using small numbers of clients.
- Requires the z/OS node to support APPN and Virtual Internet Protocol Addresses (VIPA).
- Uses UDP connections between SNA nodes.

Existing TCP62 architecture

The diagram below shows the existing TCP62 architecture. The configuration for the CICS Universal Client uses the TCP62 protocol to connect to AnyNet APPC over IP on the z/OS system.

Table 1. Existing TCP62 architecture

Client	Protocol	Remote System
CICS Universal Client on Microsoft Windows	TCP62	AnyNet APPC over IP, VTAM, CICS on z/OS

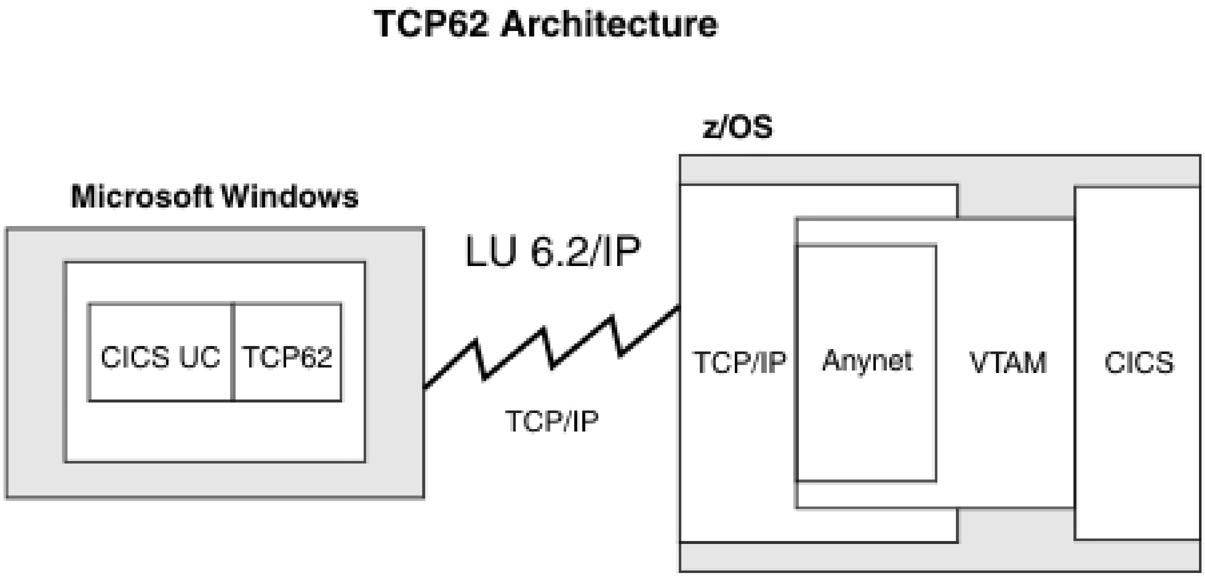


Figure 1. TCP62 Architecture

Split stack - Remote API Client

In the migrated Remote API Client architecture, the CICS Universal Client uses the TCP/IP protocol to connect to a Communications Server for Linux. The Communications Server also uses TCP/IP to connect to the Enterprise Extender on the z/OS system, although other link protocols can be used for this connection.

Table 2. Enterprise Extender, Remote API Client

Client	Protocol	Communications Server	Protocol	Remote System
CICS Universal Client on Microsoft Windows (configured for SNA)	TCP/IP	IBM Communications Server for Linux on System z	HPR/IP	Enterprise Extender, VTAM, CICS on z/OS

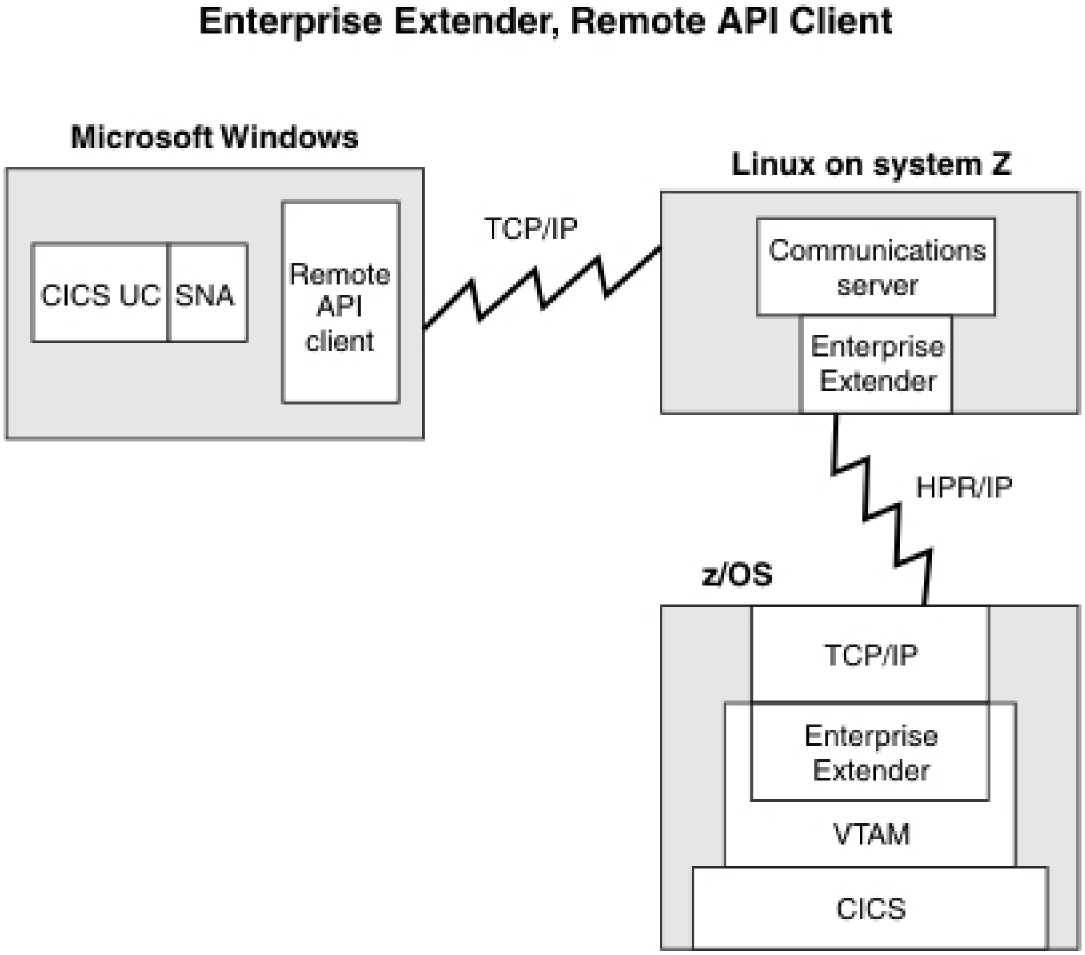


Figure 2. Enterprise Extender, Remote API Client

Full SNA stack - Communications Server

If you do not want to use a Remote API Client, you can configure a full SNA stack HPR/IP connection on Microsoft Windows to connect directly to VTAM, which is the full stack option. This option is not described further in this migration guide; however, the information provided is sufficient to help you configure this option.

Table 3. Enterprise Extender, Communications Server

Client	Protocol	Remote System
CICS Universal Client on Microsoft Windows and IBM Communications Server for Windows (configured for SNA)	HPR/IP	Enterprise Extender, VTAM, CICS on z/OS

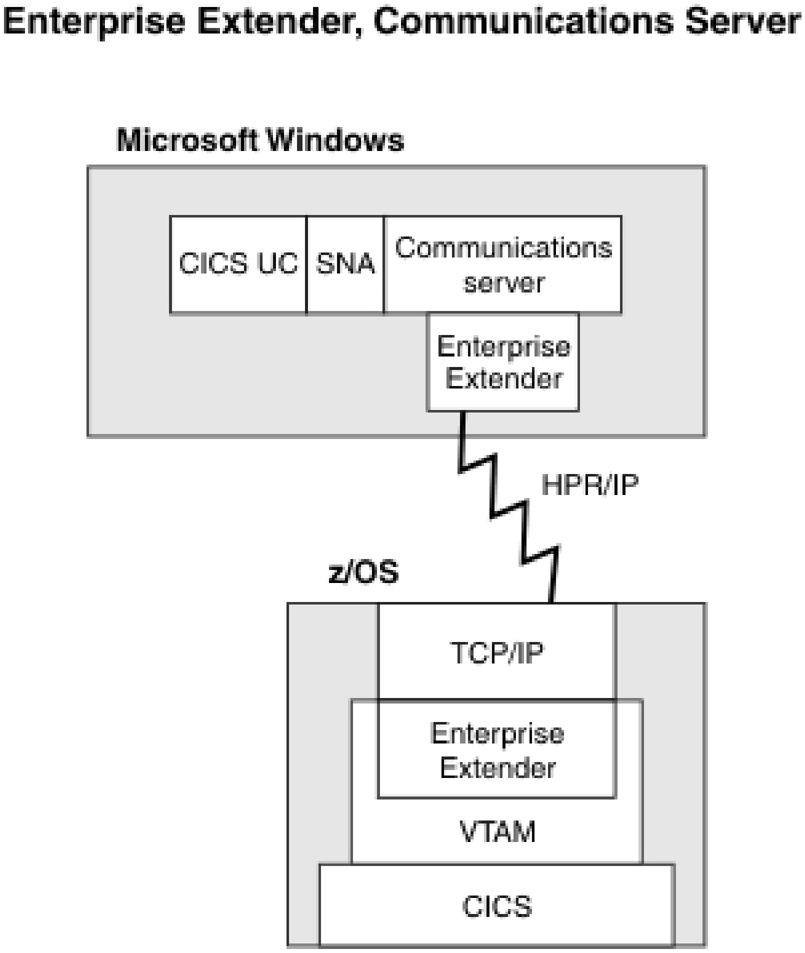


Figure 3. Enterprise Extender, Communications Server

Configuration information

Configuration information that you must update is listed in the following table:

Table 4. Configuration information references

Chapter	**Title**	**Reference**
Chapter 2	Chapter 2, “Configuring TCP/IP and VTAM on z/OS,” on page 7	TCP/IP and VTAM configurations
Chapter 3	Chapter 3, “Installing and configuring IBM Communications Server for Linux on System z,” on page 13	IBM Communications Server on Linux
Chapter 4	Chapter 4, “Configuring and testing CICS Universal Client on Windows,” on page 25	Remote API client on Windows and the ctg.ini file
Chapter 5	“Cross-product configuration definitions” on page 39	Table of example definitions

You do not have to change CICS definitions for this migration, because the CICS LU6.2 connectivity mechanism is unchanged.

Operating environments

CICS Universal Client specified operating environments for hardware and software can be found at:

http://www.ibm.com/software/htp/cics/cuc/reqs/

Chapter 2. Configuring TCP/IP and VTAM on z/OS

Enterprise Extender is a combination of IP and VTAM component configuration on IBM Communications Server. Both components have individual configuration files, which, when started together, provide the Enterprise Extender SNA over IP link. Two configuration tasks achieve this link.

- **TCP/IP**. You require IUTSAMEH and VIPA device definitions and an update of the HOME interface. You then configure the routing daemon to advertise the presence of the VIPA.
- **VTAM**. You require External Communications Adaptor (XCA) major node and switched major node definitions.

Configuring TCP/IP

On z/OS, use of Enterprise Extender requires a Virtual Internet Protocol Address (VIPA).

TCP/IP configuration - add device entries

Add a device entry to the TCP/IP Profile data set for the VIPA and IUTSAMEH definitions. These device entries define the connection from the TCP/IP stack to the VTAM instance running on the same host, using the IUTSAMEH device to move data between VTAM and TCP/IP and vice versa.

Here is a sample IUTSAMEH definition:

```
; VIPA link for EE
     DEVICE VIPAEE  VIRTUAL 0
      LINK   VLINKEE VIRTUAL 0 VIPAEE

      ; VTAM link to VIPA
      DEVICE IUTSAMEH MPCPTP
      LINK   EELINK   MPCPTP IUTSAMEH
      START IUTSAMEH
```

Figure 4. Example IUTSAMEH definition

TCP/IP configuration - update the HOME interface

Add the VIPA link to the list of HOME interfaces. Here is a sample HOME statement:

```
HOME
        10.20.125.52   VLINKEE
```

Figure 5. Example HOME statement

TCP/IP configuration - advertise the presence of the VIPA

To best exploit the capabilities of VIPA on z/OS, you can use the OMPROUTE routing daemon or ARP takeover to provide fault tolerance at the adapter level.

For further details refer to *z/OS V1R8.0 Communications Server IP Configuration Guide, SC31-8775-10*, and the *z/OS V1R8.0 Communications Server IP Configuration Reference, SC31-8776-11*.

Configuring VTAM

You require VTAM configuration to create an HPR/IP link through Enterprise Extender.

The VTAM IPADDR start option <*C*> contains the value of the VIPA TCP/IP address that Enterprise Extender uses to listen for IP requests.

The table “Cross-product configuration definitions” on page 39 provides guidance on the IPADDR start option and other key definitions that are required.

Here is an example configuration:

```
* Sample VTAM start opts setup
***********************************************************************
CONFIG=2C,                          CONFIGURATION LIST 2C             *
CPCP=YES,                           FOR APPN                          *
IPADDR=10.20.125.52,                VIPA ADDRESS FOR EE               *
NETID=GBIBMIYA,                     NETWORK NAME                      *
NODETYPE=NN,                        FOR APPN N/W NODE FUNCTIONS
NQNMODE=NQNAME,   X-NET RESOURCES DEFINED N/W QUALIFIED NAME
SSCPNAME=IYCWCDRM,                  SAME AS CDRM AND CPNAME(APPN)
SSCPID=52,                          DEFAULT SSCPID SUBAREA
IOBUF=(100,496,18,,42,30),
CONNTYPE=APPN
```

Figure 6. Example VTAM start options

Other important start options, which dictate the use of APPN, are as follows:
- CPCP=YES
- NODETYPE=NN or EN
- CONNTYPE=APPN

For further details see *z/OS V1R8.0 Communications Server SNA Resource Definition Reference*, SC31-8778-07.

Configuring an XCA Major Node

You configure one XCA major node for each LPAR to define the connection between VTAM and TCP/IP on the z/OS system.

```
* External communications adapter (XCA) VTAM major node defines the IP port
 * connection to the z/OS TCP/IP stack that VTAM will use for EE connections
 * SYSP.VTAMLST(XCAEE)
 XCAEE1          VBUILD TYPE=XCA
 EEPORT   PORT   MEDIUM=HPRIP,
                 LIVTIME=10,
                 IPTOS=(20,40,80,C0),
                 SAPADDR=4,
                 SRQRETRY=3,
                 SRQTIME=15
 EEGROUP  GROUP  DIAL=YES
                 DYNPU=NO,
                 AUTOGEN=(100,EEXL,EEXP),
                 CALL=INOUT
```

Figure 7. Example XCA major node VTAM configuration

The Autogen parameter configures the maximum number of connections available to clients for simultaneous connectivity. Set it to the maximum number of remote SNA nodes that will connect using HPR/IP.

Configuring a Switched Major Node

You can use a switched major node to provide additional security to prevent unknown SNA nodes from dynamically connecting to VTAM. Set the value of the DYNPU parameter to NO (DYNPU=NO) on the XCA major node. Ensure that each remote node has a corresponding switched PU definition. On the host, you need a switched major node definition for the CP Name of the server. For example:

```
SWLNX VBUILD TYPE=SWNET *
PULNXS01 PU       CPNAME=CTGUZL1
                  CPCP=YES
                  CONNTYPE=APPN
                  HPR=YES  *
```

Figure 8. Example switched major node definition on the host

Troubleshooting TCP/IP connectivity

Four tests can confirm HPR/IP connectivity between the client workstation and VTAM.

1. Ping from the client to the VIPA address:

```
C:\>ping winmvs2caa

Pinging winmvs2caa [10.20.125.52] with 32 bytes of data:

Reply from 10.20.125.52: bytes=32 time<1ms TTL=61
Reply from 10.20.125.52: bytes=32 time<1ms TTL=61
Reply from 10.20.125.52: bytes=32 time<1ms TTL=61
Reply from 10.20.125.52: bytes=32 time<1ms TTL=61

Ping statistics for 10.20.125.52:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

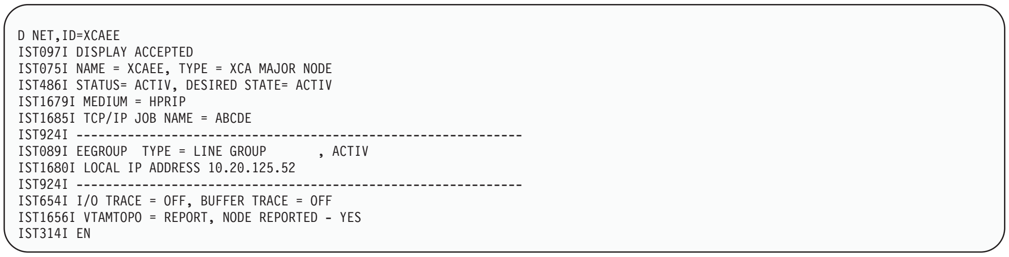
Figure 9. Ping from the client to the VIPA address

2. Check that the IUTSAMEH definition is ACTIVE in VTAM:

```
   Display  Filter  View  Print  Options  Help
 ----------------------------------------------------------------------------------------------------------------------------
 SDSF SYSLOG  26852.110 MV2C MV2C 05/14/2007 31W  24230      COLUMNS   1 132
 COMMAND INPUT ===> /D NET,ID=IUTSAMEH                         SCROLL ===> CSR
NC0000000 MV2C     07134 11:42:53.03 ABCDEFG  00000280  D NET,ID=IUTSAMEH,E
NR0000000 MV2C     07134 11:42:53.04 STC26892 00000090  IST097I DISPLAY ACCEPTED
MR0000000 MV2C     07134 11:42:53.04 STC26892 00000080  IST075I NAME = IUTSAMEH, TYPE = TRLE 574
DR                                        574 00000080  IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
DR                                        574 00000080  IST087I TYPE = LEASED             , CONTROL = MPC , HPDT = YES
DR                                        574 00000080  IST1954I TRL MAJOR NODE = ISTTRL
DR                                        574 00000080  IST1715I MPCLEVEL = HPDT       MPCUSAGE = SHARE
DR                                        574 00000080  IST1717I ULPID = TCPIP
DR                                        574 00000080  IST1717I ULPID = AP2CXCAE
DR                                        574 00000080  IST1500I STATE TRACE = OFF
ER                                        574 00000080  IST314I END
```

Figure 10. IUTSAMEH status in VTAM

3. Check the details of the XCA major node:



```
D NET,ID=XCAEE
IST097I DISPLAY ACCEPTED
IST075I NAME = XCAEE, TYPE = XCA MAJOR NODE
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST1679I MEDIUM = HPRIP
IST1685I TCP/IP JOB NAME = ABCDE
IST924I --------------------------------------------------------------
IST089I EEGROUP  TYPE = LINE GROUP       , ACTIV
IST1680I LOCAL IP ADDRESS 10.20.125.52
IST924I --------------------------------------------------------------
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST1656I VTAMTOPO = REPORT, NODE REPORTED - YES
IST314I EN
```

Figure 11. XCA major node details

4. In z/OS v1r8 there is new D NET, EEDIAG command to test for connectivity between nodes. This will work only when the Linux SNA node is configured. An example of the syntax is:

D NET,ID=EEDIAG,TEST=YES,IPADDR=(10.20.125.52,10.20.125.53)

This tests UDP through ports 12000-12004.

For further details see *z/OS V1R8.0 Communications Server Quick Reference*, SX75-0124-06 and *z/OS V1R8.0 Communications Server SNA Operations*, SC31-8779-06.

Chapter 3. Installing and configuring IBM Communications Server for Linux on System z

This chapter describes the tasks required to configure the IBM Communications Server for Linux on System z (Communications Server). The configuration tasks create an HPR/IP connection to VTAM on z/OS and allow a remote API connection from the CICS Universal Client on Windows.

The tasks are the configuration of the following Communications Server definitions:

1. "Configuring the node" on page 14
2. "Configuring the port" on page 15
3. "Configuring the link station to VTAM" on page 16
4. "Configuring the local LU" on page 17
5. "Configuring the partner LU" on page 18
6. "Configuring the mode" on page 19

Installation of Communications Server

To install and configure the IBM Communications Server for Linux on System z software, follow the installation process information in the Communications Server product readme file and documentation.

During the installation process, you are prompted for the name of the server that will be the master server in the Communications Server domain. In a single Communications Server computer configuration this name will be the fully qualified hostname of the Communications Server host.

If you do not provide a master server name, Remote API Client requests from CICS Universal Client are unable to connect.

You can amend the master server name at a later stage using the **snaetutil** command-line utility.

For further details of the master server name, see the *IBM Communications Server for Linux Administration Guide Version 6.2.2, SC31-6771-02.*

Configuring Communications Server

This section describes how to configure the node, port, link station, Local LU, Partner LU, and mode definitions of the Communications Server.

Start the SNA service using the **sna start** command before running **xsnaadmin**.

Use the **xsnaadmin** X-Windows GUI utility or **snaadmin** command-line utility to configure the Communications Server.

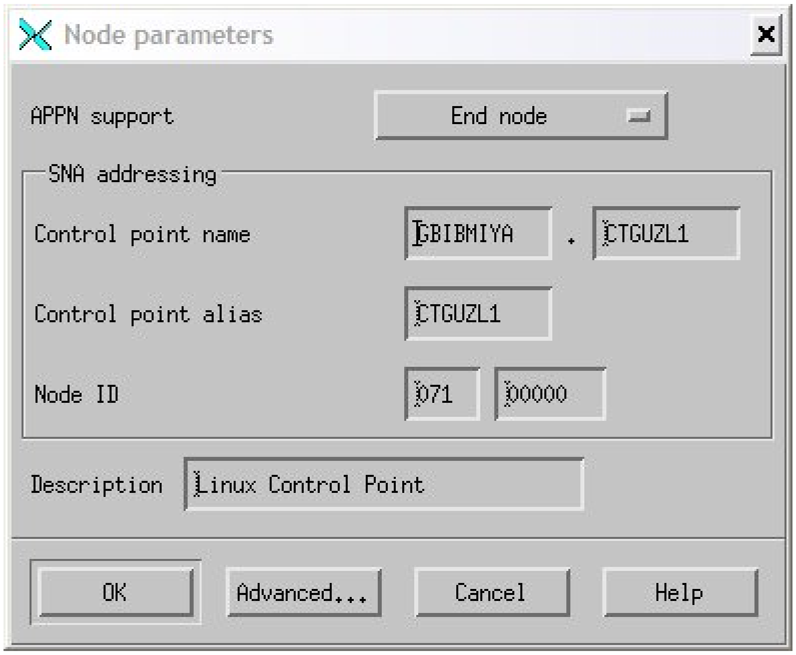
Configuring the node

In the **xsnaadmin** GUI screen, to configure the node:

1. Click **Selection** on the Menu bar
2. Click **Node Properties**

A new window appears to enable configuration of the SNA node:

1. Click **Services** on the Menu bar.
2. Click **Configure node parameters**.
3. Enter the Control point name *<D>*.
4. Enter the Control point alias *<E>*.
5. Select **End node** from the **APPN support** drop-down list box.
6. Leave the other parameters at the default values.
7. Click **OK**.



APPN support: End node

Control point name: GBIBMIYA . CTGUZL1

Control point alias: CTGUZL1

Node ID: 071 00000

Description: Linux Control Point

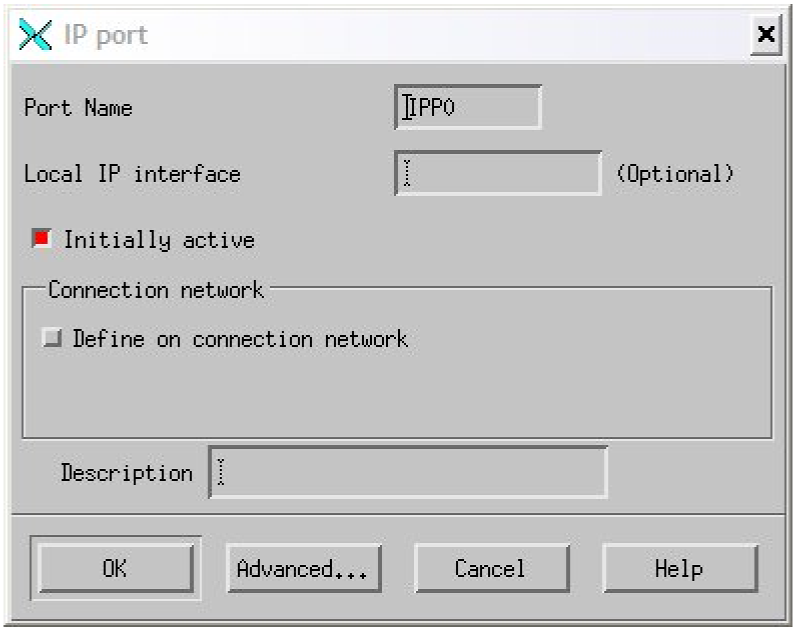
Figure 12. Node configuration window

Configuring the port

You can use a variety of link protocols to connect from Communications Server on Linux to VTAM. In this example, HPR/IP (Enterprise Extender) is selected.

In the **xsnaadmin** GUI screen, to configure the port:

1. Click **Services** on the Menu bar.
2. Click **Connectivity**.
3. Click **new port**.
4. Select **Enterprise Extender** from the **port** field using drop-down list box.
5. Leave the default port name of IPPO.
6. Click **OK**.



Port Name: IPPO

Local IP interface: (Optional)

■ Initially active

☐ Define on connection network

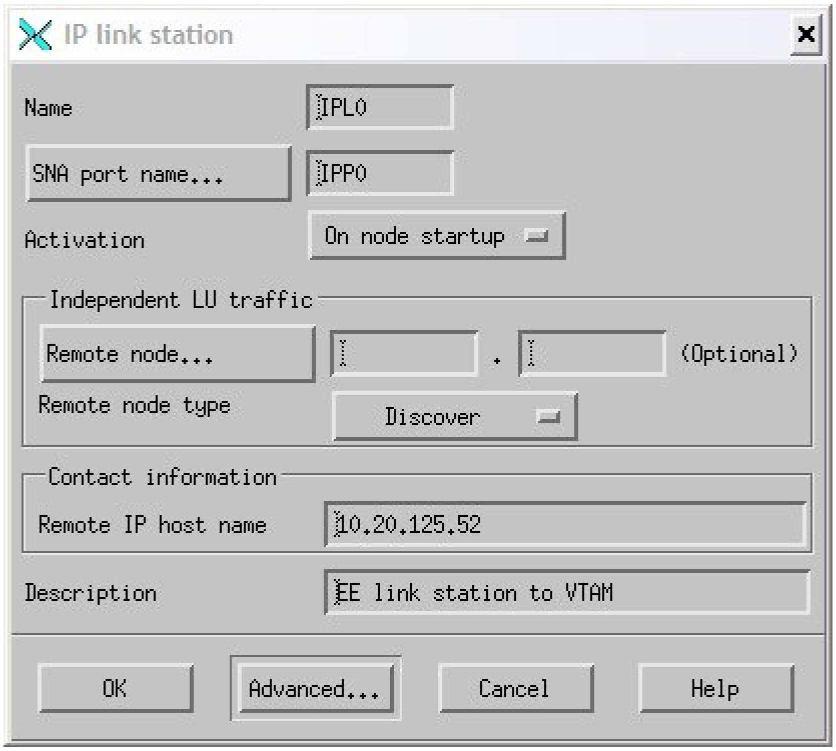
Description:

Figure 13. IP Port configuration window

Configuring the link station to VTAM

In the **xsnaadmin** GUI screen, to configure the link station to VTAM:

1. Click **Services** on the Menu bar.
2. Click **Connectivity -> New link station**.
3. Leave the **SNA port name** setting as the default IPPO and click **OK**.
4. Select **Activation** as **On node startup** using the drop down list box.
5. Leave the **Remote node** fields blank because this parameter is discovered by the system.
6. Select **Remote node type** as **Discover**.
7. Enter the **Remote IP host name** <C> used by the XCA major node on VTAM.
8. Click **OK**.



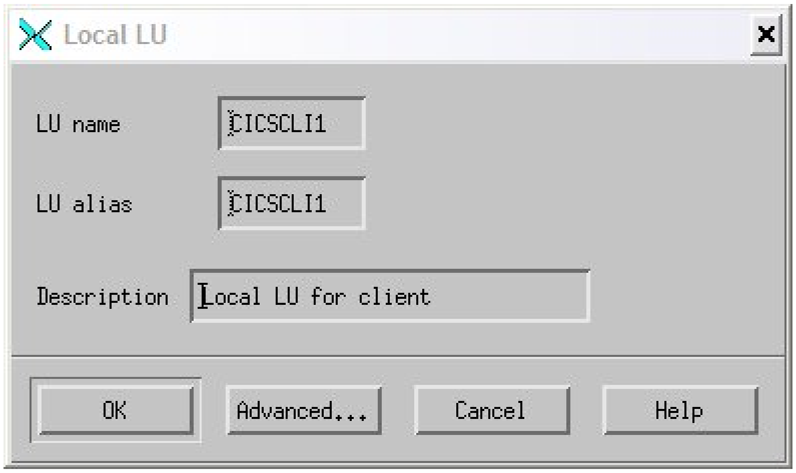
Name: IPL0
SNA port name...: IPP0
Activation: On node startup
Remote node type: Discover
Remote IP host name: 10.20.125.52
Description: EE link station to VTAM

Figure 14. IP link station configuration window

Configuring the local LU

In the **xsnaadmin** GUI screen, to configure the local LU:

1. Click **Services** on the Menu bar.
2. Click **Appc -> New independent local LU**.
3. Enter the local LU name <*F*> in the **LU name** field.
4. The **LU alias** field defaults to a lower case version of the LU name. Update the LU alias to uppercase <*F*>.
5. Click **OK**.



LU name CICSCLI1

LU alias CICSCLI1

Description Local LU for client

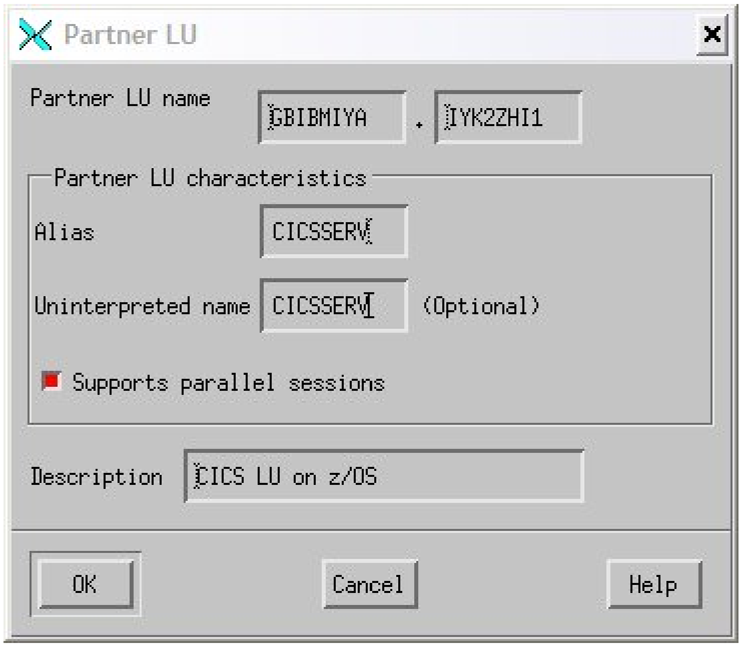
Figure 15. Local LU configuration window

For each additional client workstation running CICS Universal Client you will need a separate Local LU definition.

Configuring the partner LU

In the **xsnaadmin** GUI screen, to configure the partner LU:

1. Click **Services** on the Menu bar.
2. Click **Appc -> New partner LU -> Partner LU alias**.
3. Enter the **Partner LU name** <*D*> <*G*>.
4. The **Alias** field defaults to lower case. Update the alias to uppercase <*G*>.
5. Click **OK**.



Partner LU name GBIBMIYA . IYK2ZHI1

Alias CICSSERV

Uninterpreted name CICSSERV (Optional)

■ Supports parallel sessions

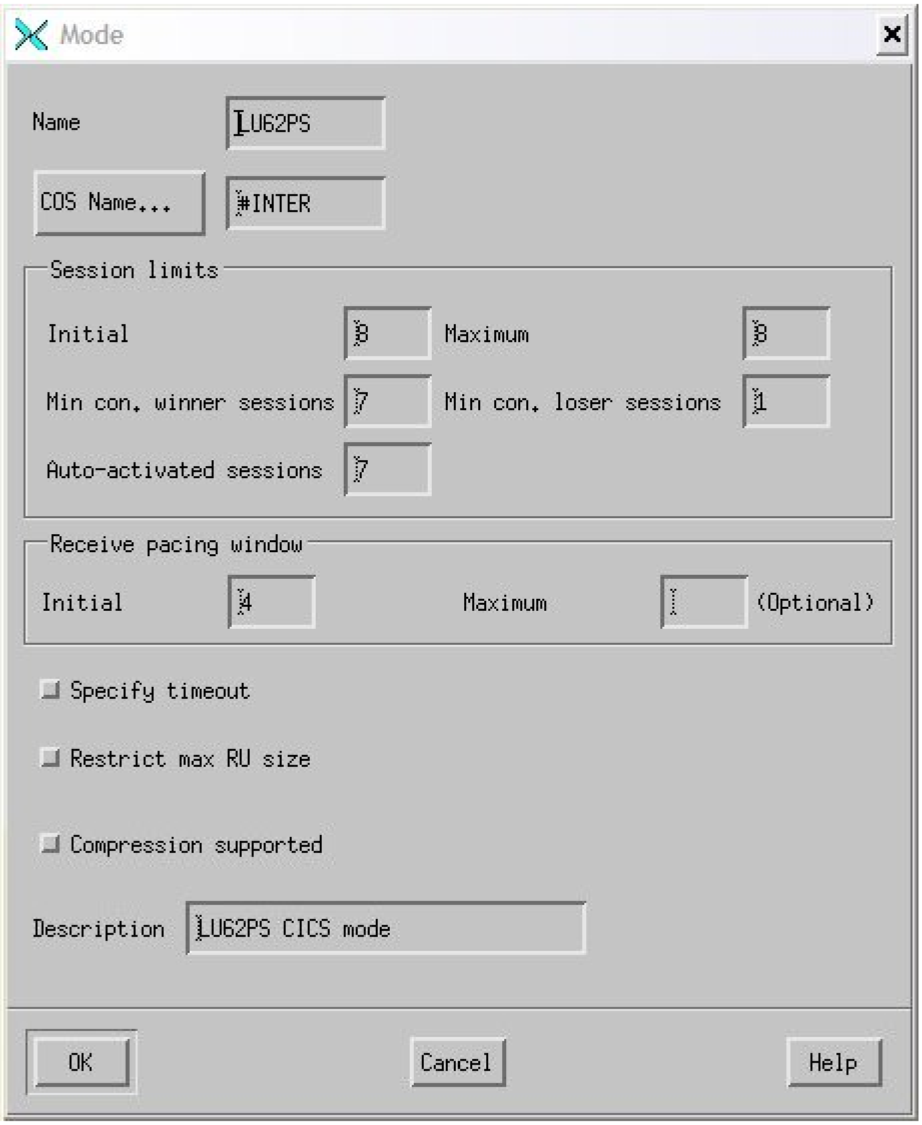
Description CICS LU on z/OS

Figure 16. Partner LU configuration window

Configuring the mode

In the **xsnaadmin** GUI screen, to configure the mode:

1. Click **Services** on the Menu bar.
2. Click **APPC -> Mode**.
3. Click **New** and enter the CICS mode **Name** <*H*>.
4. Select the **COS Name** (Class of Service Name) as **#INTER**.
5. Enter the **Initial** and **Maximum** session limits as <*I*>.
6. Enter the **Min con. winner sessions**. This is <*I*> minus <*J*>.
7. Enter the **Min. con. loser sessions** <*J*>. Set this entry to at least 1 if CICS Universal Client is using ATI requests. This figure is dependent on the ratio of outbound and inbound requests expected when ATI requests are being used by CICS Universal Client. For further details see “Configuring Automatic Transaction Initiation” on page 34.
8. Enter the **Auto-activated sessions** as **1**.
9. Click **OK**.



Name LU62PS

#INTER

Session limits

Initial 8 Maximum 8

Min con. winner sessions 7 Min con. loser sessions 1

Auto-activated sessions 7

Receive pacing window

Initial 4 Maximum (Optional)

☐ Specify timeout

☐ Restrict max RU size

☐ Compression supported

Description LU62PS CICS mode

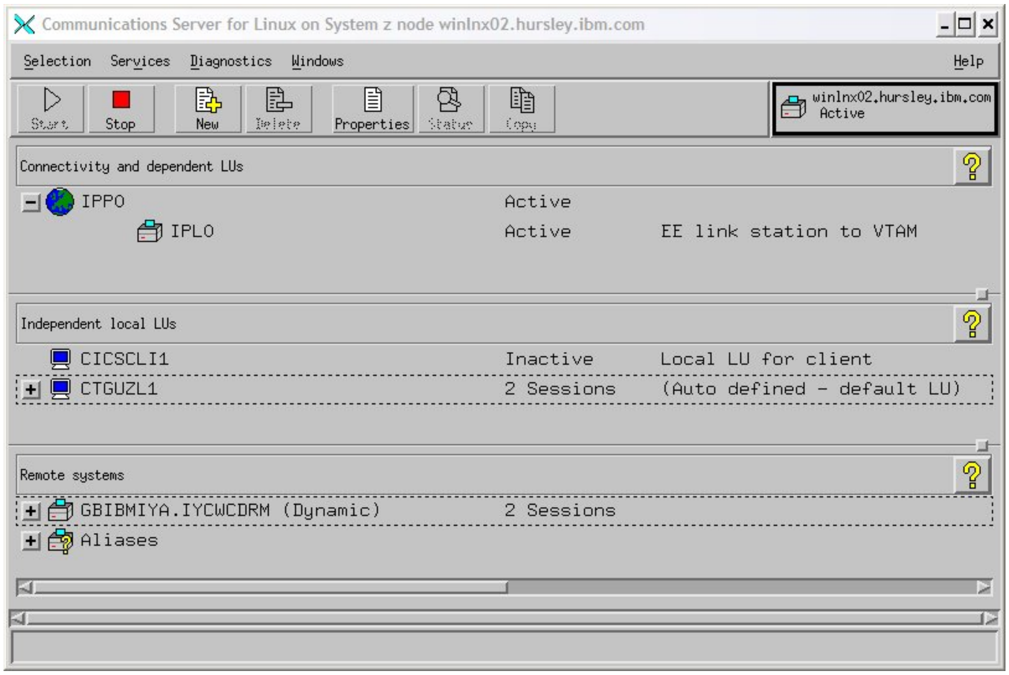
Figure 17. Mode configuration window

Verifying link station connectivity to VTAM

This section describes how to test connectivity from the Communications Server to VTAM.

1. From the **xsnaadmin** screen, in the **Connectivity and dependent LUs** panel, click the port **IPPO.**
2. Click **Start**. The status changes from Inactive to Active and the link station IPLO will start.

When the link station has activated, you know that connectivity is established from Communication Server on Linux to VTAM over the HPR/IP connection.



```
IPPO                         Active
    IPLO                     Active        EE link station to VTAM
```

```
CICSCLI1                     Inactive      Local LU for client
CTGUZL1                      2 Sessions    (Auto defined - default LU)
```

```
GBIBMIYA.IYCWCDRM (Dynamic)  2 Sessions
Aliases
```

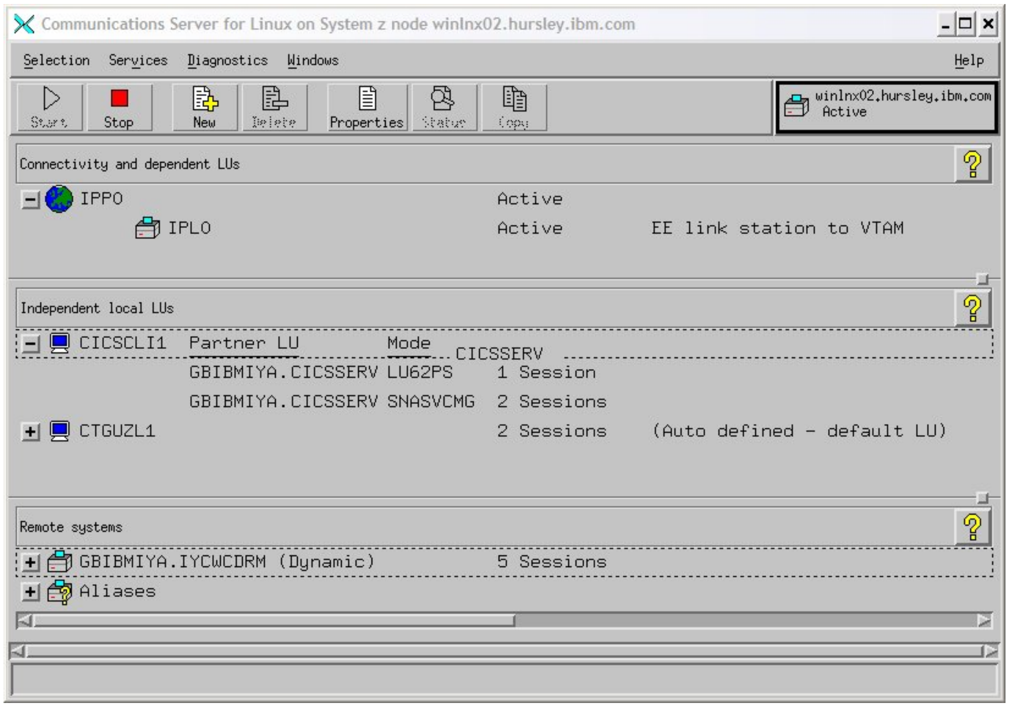
Figure 18. Link station active

Verifying the binding of APPC sessions to CICS

This section describes how to test connectivity from Communications Server to CICS by binding APPC sessions.

1. With the IPLO link station active, select the independent local LU that was just defined <*F*>.
2. Click **Start.** A pop-up menu appears to activate a session.
3. Select the **Alias** to specify the CICS Partner LU name <*G*>.
4. Select the mode <*H*>.
5. Click **OK** to bind the sessions.

In the **xsnaadmin** panel, you now see a number of sessions created under the local LU name, including two SNA session manager sessions, SNASVCMG, and at least one session in the user-defined mode group.



```
IPPO                                        Active
        IPLO                                Active       EE link station to VTAM

CICSCLI1    Partner LU          Mode    CICSSERV
            GBIBMIYA.CICSSERV   LU62PS      1 Session
            GBIBMIYA.CICSSERV   SNASVCMG    2 Sessions
CTGUZL1                                     2 Sessions    (Auto defined - default LU)

GBIBMIYA.IYCWCDRM (Dynamic)                 5 Sessions
Aliases
```

Figure 19. APPC sessions activated

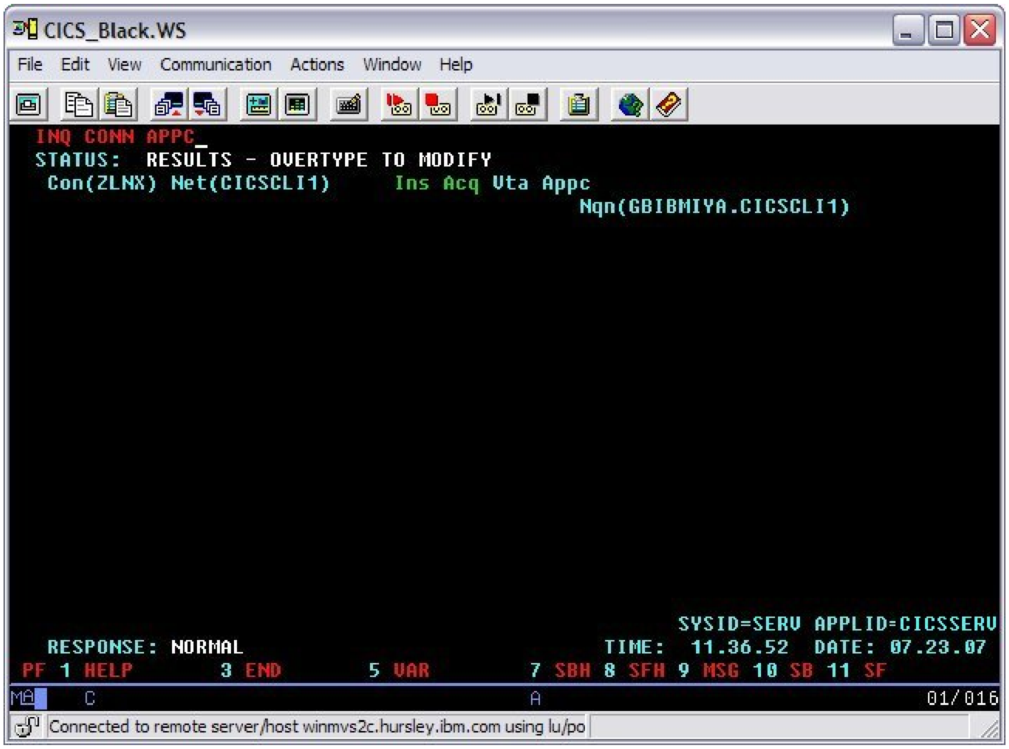
If the session activation fails, check the CICS CSMT log, the VTAM console, and the Communications Server error log.

Use the **aping** command as an alternative to the GUI, to start a session. You must have the `APINGD` tool installed on the target system.

Verifying the binding of APPC sessions from CICS

This section describes an additional test of APPC connectivity from CICS to the Communications Server, without involving the Remote API Client configuration or the CICS Universal Client.

Enter the `CEMT INQ CONN APPC` command from a CICS terminal. When an APPC connectivity test has previously been run, the connection should be in acquired state. The value **Acq** displayed against the relevant connection definition shows this state.



```
 INQ CONN APPC_
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Con(ZLNX) Net(CICSCLI1)     Ins Acq Vta Appc
                                             Nqn(GBIBMIYA.CICSCLI1)

                                                    SYSID=SERV APPLID=CICSSERV
  RESPONSE: NORMAL                            TIME:  11.36.52  DATE: 07.23.07
PF 1 HELP       3 END       5 VAR         7 SBH 8 SFH 9 MSG 10 SB 11 SF
MA     C                                 A                              01/016
```

Connected to remote server/host winmvs2c.hursley.ibm.com using lu/po

Figure 20. CICS APPC connection acquired

If you must shutdown the connection using **xsnaadmin**, the connection status in the CEMT display changes to released **Rel**. Over-typing **Rel** with **Acq** will re-activate the sessions.

The `SNASVCMG` control sessions are always active if the connection is acquired; however, the user mode session might not be activated until traffic starts to flow. This behavior depends on the autoactivate settings in the mode definition and the CICS session definition.

Memory virtualization error

When verifying the connectivity between the Communications Server and Enterprise Extender, you might receive a memory utilization error message similar to the following:

```
2006-11-17 15:00:27 GMT 512-253(0-10) P (winlnx02.hursley.ibm.com)
Not enough memory to start the node.
```

Figure 21. Memory virtualization error

You can view the SNA login details in the error log file, sna.err, in the /var/opt/ibm/sna directory.

If you see a `Not enough memory to start node` message when you first use the product after installation, reboot the server.

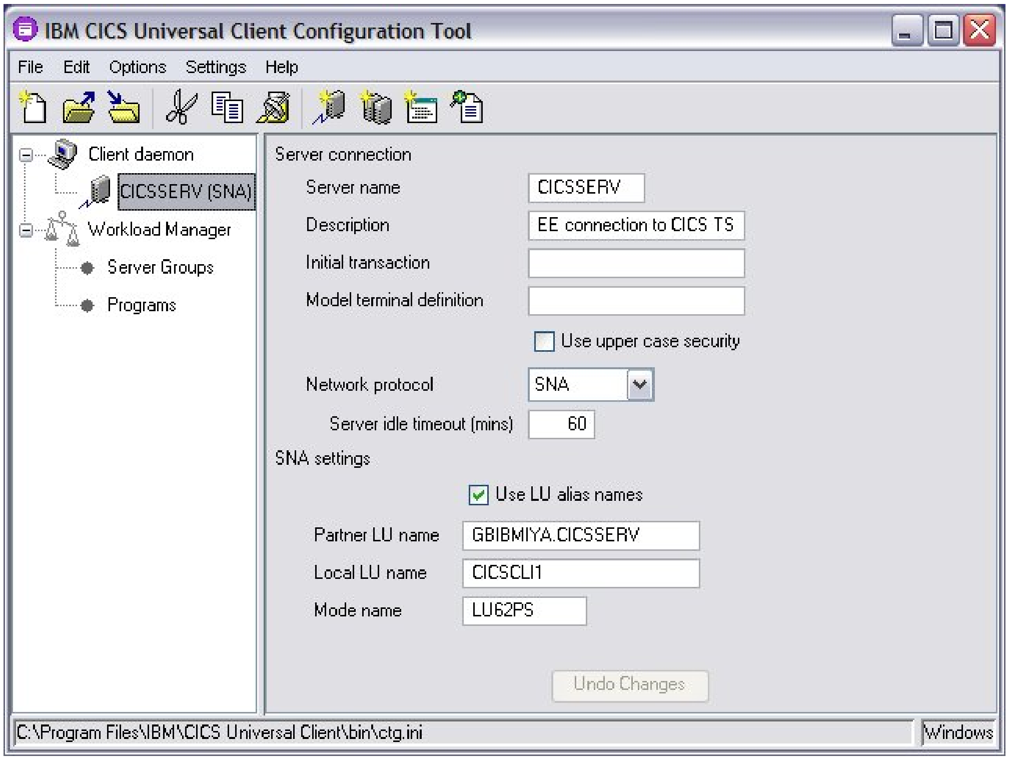
Chapter 4. Configuring and testing CICS Universal Client on Windows

This chapter describes the tasks required to configure and test the CICS Universal Client connectivity for TCP62 migration to SNA.

Configuring CICS Universal Client on the client workstation

This section describes how to configure an SNA connection using the CICS Universal Client Configuration Tool.

1. Open the CICS Universal Client configuration tool on the client workstation.
2. Select **SNA** as the **Network Protocol** from the drop-down box.
3. Select **Use LU alias names**.
4. Enter the Partner name <*G*>.
5. Enter the Local LU name <*F*>.
6. Enter the Mode name <*H*>.



Server connection

Server name: CICSSERV

Description: EE connection to CICS TS

Initial transaction:

Model terminal definition:

☐ Use upper case security

Network protocol: SNA

Server idle timeout (mins): 60

SNA settings

☑ Use LU alias names

Partner LU name: GBIBMIYA.CICSSERV

Local LU name: CICSCLI1

Mode name: LU62PS

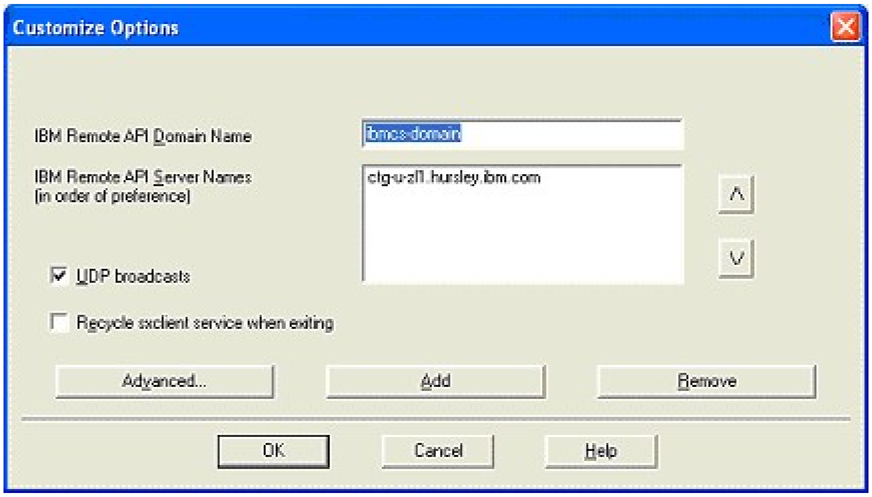
C:\Program Files\IBM\CICS Universal Client\bin\ctg.ini

Figure 22. Client configuration tool

Configuring the Remote API Client on the client workstation

This section describes how to configure the Communications Server Remote API Client on the client workstation using the CICS Universal Client Configuration Tool.

1. Open the Remote API Client configuration utility on the client workstation.
2. Accept the default of **ibmcs-domain** <*B*> for the **IBM Remote API Domain Name** parameter. For enhanced security you might set a new domain name for the Remote API Client domain.
3. Add the IP address name <*A*> of the Communications Server for Linux system to the host list for the **IBM Remote API Server Names**.
4. Uncheck the **UDP broadcasts** check box, to ensure that only the listed Remote API Server is contacted.
5. For the Remote API Client to take account of configuration changes, check the **Recycle sxclient service when exiting** box or, from the Windows services control panel, restart the IBM Remote API Client service.



IBM Remote API Domain Name: ibmcs-domain

IBM Remote API Server Names (in order of preference): ctg-u-zl1.hursley.ibm.com

☑ UDP broadcasts

☐ Recycle sxclient service when exiting

Figure 23. Remote API configuration window - Customize Options

Remote API Client uses ports tcp 1553 and udp 1553 on the IBM Communications Server for Linux on System z server. Enable these ports for firewall access.

Using the **aping** tool tests the configuration from the CICS Universal Client to the Communications Server and then on to VTAM on CICS. Usually, if an **aping** test fails in this configuration, the problem is with the client configuration.

If an **aping** test fails, check the log files on the local client workstation. If no errors are shown locally, check the log files on the Communications Server.

Testing connectivity to Communications Server

This section describes how to use the **aping** tool to test APPC network connectivity.

An APPC client test program, **aping**, is provided with both the Remote API Client and IBM Communications Server for Linux on System z. You can use **aping** to test LU6.2 connections and verify settings. You can run with the same settings on System z and the client workstation, to isolate the computer on which the configuration or network problems can be found.

The format of the **aping** command is:

aping destination_name [–m mode_name]

Where destination name is the partner LU.

An example of the **aping** command syntax is:

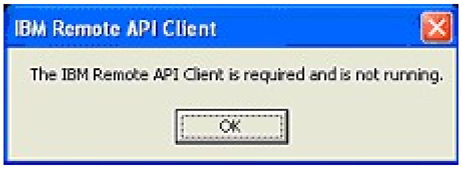
aping GBIBMIYA.CICSSERV -m LU62PS

The command syntax is derived from the parameters <*D*>,<*G*>and <*H*>.

For other options, use the **aping -help** command, or view the command reference documentation from the *IBM Communications Server for Linux Administration Command Reference - Version 6.2.2., SC31-6770-02*.

If **aping** fails, check the log files on the local workstation. If no errors appear locally, check the log files on the IBM Communications Server for Linux on System z server, and the VTAM console.

If the Remote API Client is not running, a window opens:



The IBM Remote API Client is required and is not running.

Figure 24. IBM Remote API Client window

To test connectivity from the Remote API Client to the Communications Server for Linux, enter an **aping** command for the fully qualified CP name of the Linux node:

```
C:\IBMCS\w32cli>aping GBIBMIYA.CTGUZL1
IBM aping version 2.44 APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Allocate duration:      530 ms
Program startup and Confirm duration:      130 ms

Connected to a partner running on: Communications Server for Linux on System z 6.2.2.0
        Duration        Data Sent        Data Rate        Data Rate
        (msec)          (bytes)          (KB/s)           (Mb/s)
        --------        ---------        ---------        ---------
             690              200              0.3            0.002
             130              200              1.5            0.012
Totals:      820              400              0.5            0.004
Duration statistics: Min = 130 Ave = 410 Max = 690
```

Figure 25. Aping command for the fully qualified CP name of the Linux node

To test connectivity from the Communications Server for Linux computer to VTAM, enter the **aping** command on the Linux computer for the fully qualified CP name of the VTAM node:

```
[ctgtest@winlnx02 ~]$ aping GBIBMIYA.IYCWCDRM
IBM aping version 2.44 APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Allocate duration:      410 ms
Program startup and Confirm duration:       40 ms

Connected to a partner running on: (UNKNOWN operating system)
        Duration        Data Sent        Data Rate        Data Rate
        (msec)          (bytes)          (KB/s)           (Mb/s)
        --------        ---------        ---------        ---------
               0              200
              10              200             19.5            0.156
Totals:       10              400
Duration statistics: Min = 0 Ave = 5 Max = 10
```

Figure 26. Aping command on the Linux computer for the fully qualified CP name of the VTAM node

To test connectivity from the Remote API Client to VTAM node, enter the **aping** command on the Remote API Client workstation for the fully qualified CP name of the VTAM node:

```
C:\IBMCS\w32cli>aping GBIBMIYA.IYCWCDRM
IBM aping version 2.44 APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Allocate duration:     2810 ms
Program startup and Confirm duration:      130 ms

Connected to a partner running on: (UNKNOWN operating system)
        Duration        Data Sent        Data Rate        Data Rate
        (msec)          (bytes)          (KB/s)           (Mb/s)
        --------        ---------        ---------        ---------
             150              200              1.3            0.010
             130              200              1.5            0.012
Totals:      280              400              1.4            0.011
Duration statistics: Min = 130 Ave = 140 Max = 150
```

Figure 27. Aping command on the Remote API Client workstation computer for the fully qualified CP name of the VTAM node

To test end-to-end connectivity from the Remote API Client to the CICS LU, use the **aping** command on the Remote API Client workstation, using the ***aping destination_name*** **–m** ***mode_name*** syntax. You must specify a mode that is defined to VTAM for the CICS APPL. The command fails with the following error if the `APINGD TP` is not defined to CICS.

```
C:\IBMCS\w32cli>aping GBIBMIYA.CICSSERV -m LU62PS
IBM aping version 2.44 APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Allocate duration:      540 ms
Problem detected by the client.
The server TP name is not known on the remote system.
```

Figure 28. Aping command on the Remote API Client workstation computer to the CICS LU - APINGD TP is not defined to CICS

However, if the remote CICS LU name is incorrect or cannot be found, the following appears:

```
C:\IBMCS\w32cli>aping GBIBMIYA.CICSSER1 -m LU62PS
IBM aping version 2.44 APPC echo test with timings.
Licensed Materials - Property of IBM
(C) Copyright 1994,1995 by IBM Corp. All rights reserved.

Problem detected by the client.
An allocation failure occurred.
```

Figure 29. Aping command on the Remote API Client workstation computer - Incorrect CICS LU name specified

If the Windows workstation logs are empty, you might see the following message on the Linux on zSeries® server:

```
2006-12-08 11:42:13 GMT 512-533(0-10) P (winlnx02.hursley.ibm.com)
Locate search failed: LU not found.
Sense code           = 0x08400007
Origin CP name       = GBIBMIYA.WINLNX02
Origin LU name       = GBIBMIYA.WINLNX02
Destination LU name  = GBIBMIYA.IYKZ2HI2
```

Figure 30. Locate LU search failed

If the mode name is incorrect, the following error in the Communications Server on Linux computer indicates that an APPN search was unable to locate the specified partner LU.

```
2006-12-08 11:37:01 GMT 512-57(0-10) P (winlnx02.hursley.ibm.com)
Failed to activate a session - mode name not recognized.
Local LU (Alias)   = GBIBMIYA.WINLNX02 (ctguzl1 )
Partner LU (Alias) = GBIBMIYA.CICCSERV (CICSSERV)
Mode name          = #INTER
2006-12-08 11:37:03 GMT 512-57(0-10) P (winlnx02.hursley.ibm.com)
Failed to activate a session - mode name not recognized.
Local LU (Alias)   = GBIBMIYA.WINLNX02 (ctguzl1 )
Partner LU (Alias) = GBIBMIYA.CICSSERV (CICSSERV)
Mode name          = #INTER
```

Figure 31. APPN search was unable to locate the specified partner LU.

The Mode name defaults to `#INTER` if it is not specified. This parameter is not defined on the target computer.

You must configure the Communications Server as the MASTER to allow it to recognize when clients are connecting on the port it listens on. For further details, see “Installation of Communications Server” on page 13.

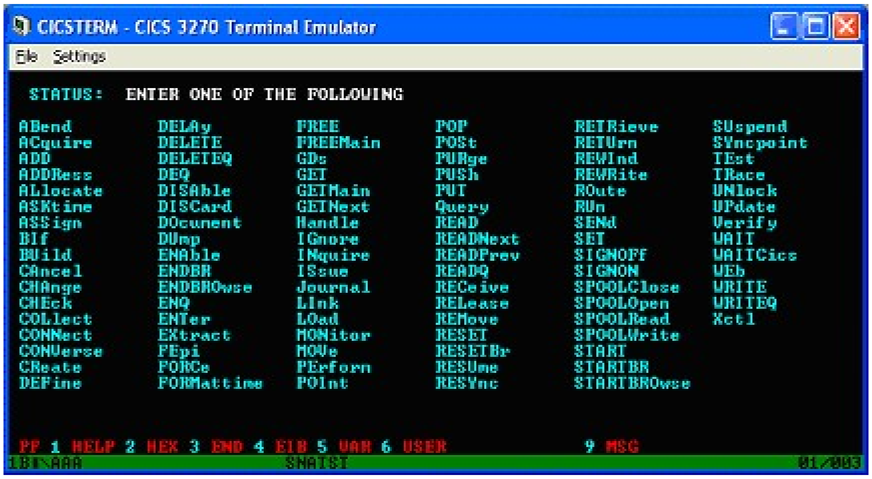
Testing the configuration between the CICS Universal Client and CICS

This section describes how to test the SNA connectivity between a CICS Universal Client and a CICS region.

Table 5. SNA connectivity between a CICS Universal Client and a CICS region

Client	Protocol	Communications Server	Protocol	Remote System
CICS Universal Client on Microsoft Windows XP (configured for SNA).	TCP/IP	IBM Communications Server for Linux on zSeries	TCP/IP	Enterprise Extender, VTAM, CICS on z/OS

1. Enter **cicscli –s=*SERVERNAME*** on the CICS Universal Client command line.
2. Issue the command **cicscli –l** repeatedly. This command should change the server status from connecting to available. If it does not, the server connection has failed to start.
3. If a server connection fails to start, look in the CICS Universal Client error log, CICSCLI.LOG in the c:\Program Files\IBM\CICS Universal Client\bin directory by default. If the log suggests SNA errors, the Remote API Client log might be useful. This log is in the sna.err file in the install directory; by default, C:\IBMCS\w32cli\sna.err.
4. As a further test of the configuration, start a cicsterm terminal to the remote server. Invoke one of the CICS sample transactions, such as CECI, and issue the **cicsterm –s=*SERVERNAME*** command from the command line.



```
 STATUS:  ENTER ONE OF THE FOLLOWING

ABend        DELAy        FREE         POP          RETRieve     SUspend
ACquire      DELETE       FREEMain     POSt         RETUrn       SYncpoint
ADD          DELETEQ      GDs          PURge        REWInd       TEst
ADDRess      DEQ          GET          PUSh         REWRite      TRace
ALlocate     DISAble      GETMain      PUT          ROute        UNlock
ASKtime      DISCard      GETNext      Query        RUn          UPdate
ASSign       DOcument     Handle       READ         SENd         Verify
BIf          DUmp         IGnore       READNext     SET          WAIT
BUild        ENAble       INquire      READPrev     SIGNOFF      WAITCics
CAncel       ENDBR        ISsue        READQ        SIGNON       WEb
CHAnge       ENDBROwse    Journal      RECeive      SPOOLClose   WRITE
CHEck        ENQ          LInk         RELease      SPOOLOpen    WRITEQ
COLlect      ENTer        LOad         REMove       SPOOLRead    Xctl
CONNect      EXtract      MONitor      RESET        SPOOLWrite
CONVerse     FEpi         MOVe         RESETBr      START
CReate       FORCe        PErform      RESUme       STARTBR
DEFine       FORMattime   POInt        RESYnc       STARTBROwse


PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER          9 MSG
1B\AAA                    SNATST                                 01/003
```

Figure 32. CICSTERM screen

Configuring Automatic Transaction Initiation

This section describes how to configure automatic transaction initiation (ATI).

Automatic transaction initiation occurs when a transaction is pushed to a terminal from a CICS region. To achieve this initiation of a CICS transaction, issue an **EXEC CICS START** command against a terminal owned by the CICS Universal Client. You can add ATI configuration after checking that the connection from CICS Universal Client to CICS over SNA is working.

To configure automatic transaction initiation, enter the IP hostname of the Remote API Client workstation configured to use the Local LU definition in Communications Server. For this reason, it is easier to use workstations with static IP addresses than dynamically allocated IP addresses when configuring for ATI use. If you use DHCP, you might use DDNS as an alternative, because the symbolic hostname remains constant, mapping to a DHCP address dynamically.

Configuring the host name

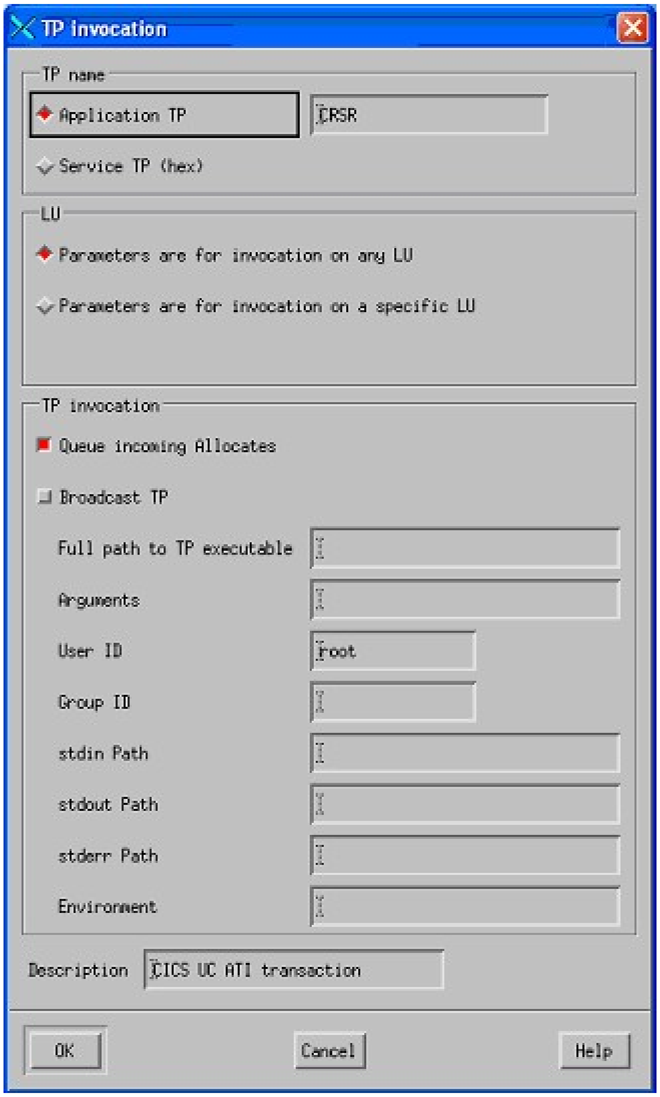
Open the Communications Server for Linux on System z **xsnaadmin** screen.

1. Select **Local LU**.
2. Click **Properties**.
3. In the **Attach Routing section**, click **Advanced**.
4. Enter the host name of the Remote API Client workstation.
5. Leave the other parameters as default.
6. Click **OK**.

Configuring TP invocation

Return to the main xsnaadmin screen.

1. Click **Services**.
2. Click **APPC**.
3. Click **Transaction Programs**.
4. Click **TP invocation**.
5. Click **New** .
6. Enter `CRSR` as the **Application TP**.
7. Enter the **User ID** as an ID that has sufficient privileges to start the TP.
8. Leave the other parameters as default.
9. Click **OK**.



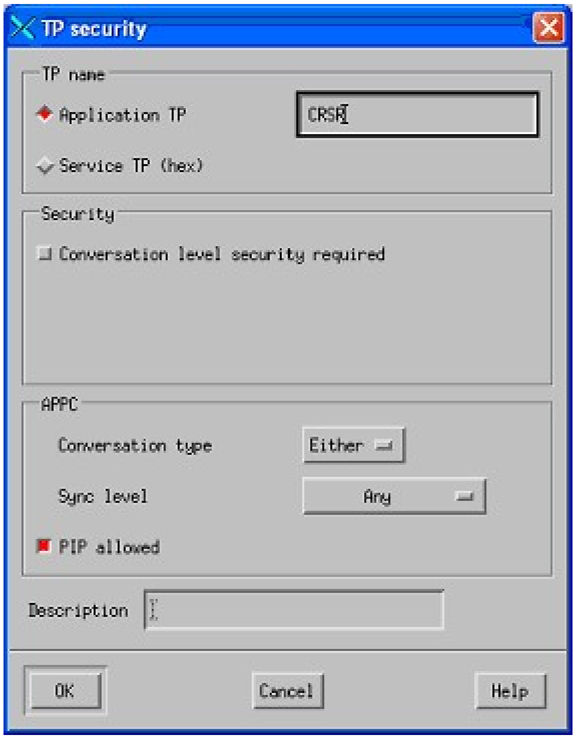
● Application TP: CRSR
○ Service TP (hex)
● Parameters are for invocation on any LU
○ Parameters are for invocation on a specific LU
☒ Queue incoming Allocates
☐ Broadcast TP
User ID: root
Description: CICS UC ATI transaction

Figure 33. TP invocation screen

Configuring TP security

From the main xsnaadmin screen:

1. Click **Services**.
2. Click **APPC**.
3. Click **Transaction programs**.
4. Click **TP security**.
5. Click **New**.
6. Enter `CRSR` as the **Application TP**.
7. Leave the other parameters as default.
8. Click **OK**.



● Application TP: CRSR

○ Service TP (hex)

☐ Conversation level security required

Conversation type: Either

Sync level: Any

☒ PIP allowed

Figure 34. TP security screen

Before running ATIs, restart the node, link, Client daemon, and SNA service.

Remote API Client registry key configuration

Check that the **invoked_tps** registry key is set to `yes` on the client workstation.

1. The **invoked_tps** key is in the registry at **HKEY_LOCAL_MACHINE->SOFTWARE->SNA Client->SxClient->Parameters->Configuration**
2. Restart the Remote API Client service to activate any change made to this registry key.

Testing automatic transaction initiation

To test whether automatic transaction initiation works, use the **cicsterm** command.

1. Start two cicsterms.
2. Note the terminal IDs from the status line at the bottom of the cicsterm sessions.
3. Use CECI to start a transaction against the second terminal. For example, if you started two terminals \AAA and \AAB, you can use terminal \AAA to start an automatic transaction initiation against terminal \AAB
4. Enter `CEMT START TRAN(CEMT) TE(\AAB)` on terminal \AAA.
5. Press **Enter** a few times until the process has completed.
6. You should see the CEMT transaction start running against terminal \AAB.

Appendix. Example configurations

This chapter contains the following examples:

- "Cross-product configuration definitions"
- "Example of a TCP62 enabled ctg.ini file" on page 41
- "Example of an Enterprise Extender enabled ctg.ini file" on page 42
- "Example of an SNA configuration file" on page 43
- "Example CICS connection session definitions" on page 47

Cross-product configuration definitions

This table provides a reference for the values used in the migration from TCP62 to Enterprise Extender.

Table 6. Cross-product configuration definition table.

Key	**Description**	**Example parameter**	**CICS Universal Client (value in ctg.ini)**	**Remote API Client**	**IBM Communications Server for Linux on System z**	**VTAM**	**CICS**
<*A*>	TCP/IP hostname of Linux host	ctg-u-zl1.hursley.ibm.com		ctg-u-zl1.hursley.ibm.com			
<*B*>	Communications Server domain	ibmcs-domain		ibmcs-domain	ibmcs-domain		
<*C*>	TCP/IP address used by VTAM	10.20.125.52			Remote IP host name	IPADDR	
<*D*>	SNA network name	GBIBMIYA			Network name	NETID	
<*E*>	Control point name	CTGUZL1			Control point name and alias		
<*F*>	Local LU name and alias	CICSCLI1	LOCALLUNAME		LU name and alias		NETNAME in CONNECTION defintion
<*G*>	Partner LU name and alias	CICSSERV	NETNAME		Partner LU name and alias		APPLID
<*H*>	Mode name	LU62PS			Mode		MODENAME in SESSIONS defintion
<*I*>	Mode session limit	8			Initial and maximum session limit		MAXIMUM value1 in SESSIONS defintion
<*J*>	Contention losers sessions	1			Minimum contention loser sessions		MAXIMUM value2 in SESSIONS defintion

Example of a TCP62 enabled ctg.ini file

This example shows a configuration file for a CICS Universal Client configured with a TCP62 connection to CICS TS for z/OS.

```
SECTION CLIENT = *
 CPNAME=GBIBMIYA.CPNAME
 CPIPADDRESSMASK=00000000
 DOMAINNAMESUFFIX=hursley.ibm.com
 ENABLEPOPUPS=N
 TERMINSTLOGGING=Y
 LOGFILE=cicscli.log
 LOGFILEINFO=cicscli.log
 TCP62PORT=0
 MAXBUFFERSIZE=32
 MAXREQUESTS=32
 MAXSERVERS=10
 MAXWRAPSIZE=0
 REMOTENODEINACTIVITYPOLLINTERVAL=60
 SRVRETRYINTERVAL=60
 TERMINALEXIT=EXIT
 TRACEFILE=cicscli.bin
 USEOEMCP=Y
 LOADMANAGER=N
ENDSECTION

SECTION SERVER = TCP62CIC
 UPPERCASESECURITY=Y
 LOCALLUNAME=CICSCLI1
 LUIPADDRESSMASK=00000000
 SNAMAXRUSIZE=4096
 SNASESSIONLIMIT=8
 MODENAME=LU62PS
 SNAPACINGSIZE=8
 NETNAME=GBIBMIYA.CICSSERV
 PROTOCOL=TCP62
ENDSECTION

SECTION DRIVER = TCP62
 DRIVERNAME=CCLTCP62
ENDSECTION
```

Figure 35. Example of a TCP62 ctg.ini file

Example of an Enterprise Extender enabled ctg.ini file

This example shows a configuration file for a CICS Universal Client configured with an Enterprise Extender HPR/IP connection to CICS TS for z/OS.

```
SECTION CLIENT = *
 ENABLEPOPUPS=N
 TERMINSTLOGGING=Y
 LOGFILE=cicscli.log
 LOGFILEINFO=cicscli.log
 MAXBUFFERSIZE=32
 MAXREQUESTS=32
 MAXSERVERS=10
 MAXWRAPSIZE=0
 SRVRETRYINTERVAL=60
 TERMINALEXIT=EXIT
 TRACEFILE=cicscli.bin
 USEOEMCP=Y
 LOADMANAGER=N
ENDSECTION

SECTION SERVER = SNAEECIC
 UPPERCASESECURITY=Y
 SRVIDLETIMEOUT=60
 PROTOCOL=SNA
 LOCALLUNAME=CICSCLI1
 MODENAME=LU62PS
 NETNAME=CICSSERV
 LUALIASNAMES=Y
ENDSECTION

SECTION DRIVER = SNA
 DRIVERNAME=CCLWNTSN
ENDSECTION
```

Figure 36. Example of an Enterprise Extender enabled ctg.ini file

Example of an SNA configuration file

This example shows a sample IBM Communications Server for Linux configuration file using HPR/IP connections to VTAM. This file, sna_node.cfg, is located in the /etc/opt/ibm/sna/ directory.

```
[define_node_config_file]
major_version = 5
minor_version = 1
update_release = 1
revision_level = 48

[define_node]
cp_alias = CTGUZL1
description = Linux Control Point
fqcp_name = GBIBMIYA.CTGUZL1
node_type = END_NODE
mode_to_cos_map_supp = YES
mds_supported = YES
node_id = <07100000>
max_locates = 1500
dir_cache_size = 255
max_dir_entries = 0
locate_timeout = 60
reg_with_nn = YES
reg_with_cds = YES
mds_send_alert_q_size = 100
cos_cache_size = 24
tree_cache_size = 40
tree_cache_use_limit = 40
max_tdm_nodes = 0
max_tdm_tgs = 0
max_isr_sessions = 1000
isr_sessions_upper_threshold = 900
isr_sessions_lower_threshold = 800
isr_max_ru_size = 16384
isr_rcv_pac_window = 8
store_endpt_rscvs = NO
store_isr_rscvs = NO
store_dlur_rscvs = NO
cos_table_version = VERSION_0_COS_TABLES
send_term_self = NO
disable_branch_awareness = NO
cplu_syncpt_support = NO
cplu_attributes = NONE
dlur_support = LIMITED_MULTI_SUBNET
pu_conc_support = YES
nn_rar = 128
max_ls_exception_events = 0
max_compress_level = LZ10
clear_initial_topology = NO
ptf_flags = NONE

[define_ip_dlc]
dlc_name = IP0
description = ""
initially_active = NO
udp_port_llc = 12000
udp_port_network = 12001
udp_port_high = 12002
udp_port_medium = 12003
udp_port_low = 12004
ip_precedence_llc = 6
ip_precedence_network = 6
ip_precedence_high = 4
ip_precedence_medium = 2
ip_precedence_low = 1
no_dns_lookup = NO
```

```
[define_ethernet_dlc]
dlc_name = ETHER0
description = ""
neg_ls_supp = YES
initially_active = NO
adapter_number = 0
lan_type = 802_3_DIX

[define_ip_port]
port_name = IPP0
description = ""
lsap_address = 0x04
dlc_name = IP0
initially_active = YES
max_rcv_btu_size = 1500
tot_link_act_lim = 4096
inb_link_act_lim = 0
out_link_act_lim = 0
implicit_ls_limit = 0
act_xid_exchange_limit = 9
nonact_xid_exchange_limit = 5
max_ifrm_rcvd = 7
target_pacing_count = 7
max_send_btu_size = 1500
implicit_cp_cp_sess_support = YES
implicit_limited_resource = NO
implicit_deact_timer = 30
implicit_uplink_to_en = NO
effect_cap = 865075200
connect_cost = 0
byte_cost = 0
security = SEC_NONSECURE
prop_delay = PROP_DELAY_LAN
user_def_parm_1 = 128
user_def_parm_2 = 128
user_def_parm_3 = 128
local_ip_interface = ""
react_timer = 30
react_timer_retry = 65535
ack_timeout = 10000
max_retry = 10
liveness_timeout = 10000
short_hold_mode = NO
```

```
[define_ip_ls]
ls_name = IPL0
description = EE link station to VTAM
port_name = IPP0
adj_cp_name = <00000000000000000000000000000000000>
adj_cp_type = LEARN_NODE
max_send_btu_size = 1500
ls_attributes = SNA
cp_cp_sess_support = YES
default_nn_server = NO
lsap_address = 0x04
auto_act_supp = NO
tg_number = 0
limited_resource = NO
disable_remote_act = NO
link_deact_timer = 30
use_default_tg_chars = YES
effect_cap = 865075200
connect_cost = 0
byte_cost = 0
security = SEC_NONSECURE
prop_delay = PROP_DELAY_LAN
user_def_parm_1 = 128
user_def_parm_2 = 128
user_def_parm_3 = 128
target_pacing_count = 7
max_ifrm_rcvd = 0
conventional_lu_compression = NO
branch_link_type = NONE
adj_brnn_cp_support = ALLOWED
initially_active = YES
restart_on_normal_deact = NO
react_timer = 30
react_timer_retry = 65535
remote_ip_host = 10.20.125.52
ack_timeout = 10000
max_retry = 10
liveness_timeout = 10000
short_hold_mode = NO

[define_partner_lu]
plu_alias = CICSSERV
description = CICS LU on z/OS
fqplu_name = GBIBMIYA.CICSSERV
plu_un_name = CICSSERV
parallel_sess_supp = YES
max_mc_ll_send_size = 0
conv_security_ver = NO
```

```
[define_local_lu]
lu_alias = CICSCLI1
list_name = ""
description = Local LU for client
lu_name = CICSCLI1
lu_session_limit = 0
pu_name = <0000000000000000>
nau_address = 0
default_pool = NO
syncpt_support = NO
lu_attributes = NONE
sscp_id = 0
disable = NO
sys_name = ""
timeout = 60
back_level = NO

[define_mode]
mode_name = LU6PS
description = LU62PS CICS mode
max_neg_sess_lim = 8
plu_mode_session_limit = 8
min_conwin_src = 7
min_conloser_src = 1
auto_act = 0
receive_pacing_win = 4
max_receive_pacing_win = 0
default_ru_size = YES
max_ru_size_upp = 1024
max_ru_size_low = 0
cos_name = #INTER
compression = PROHIBITED
max_compress_level = NONE
max_decompress_level = NONE

[define_mode]
mode_name = LU62PS
description = ""
max_neg_sess_lim = 32767
plu_mode_session_limit = 8
min_conwin_src = 7
min_conloser_src = 1
auto_act = 0
receive_pacing_win = 4
max_receive_pacing_win = 0
default_ru_size = YES
max_ru_size_upp = 1024
max_ru_size_low = 0
cos_name = #CONNECT
compression = PROHIBITED
max_compress_level = NONE
max_decompress_level = NONE

[define_tp]
tp_name = CRSR
description = ""
list_name = ""
conv_type = EITHER
security_rqd = NO
sync_level = SYNCPT_NEGOTIABLE
enabled = YES
pip_allowed = YES
tp_instance_limit = 0
incoming_alloc_timeout = 0
```

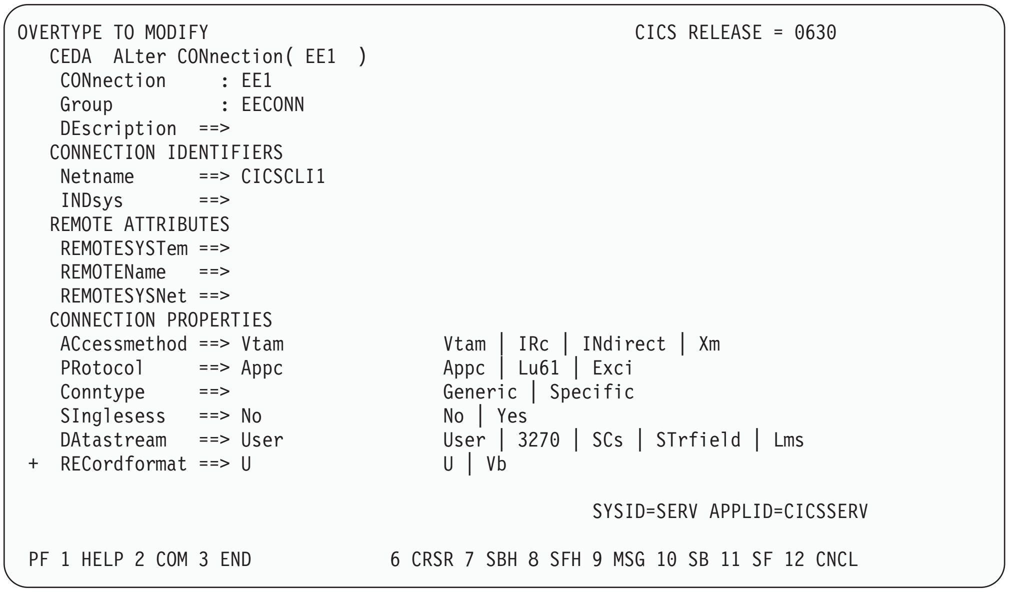
Figure 37. Example of an SNA configuration file

Example CICS connection session definitions

To use an SNA Enterprise Extender connection to CICS, configure a CICS APPC CONNECTION and SESSION definition pair.

If you have APPC connection autoinstall active, using the `DFHZATDY` autoinstall program, you do not have to define any hardcoded definitions because CICS will create definitions dynamically when it receives an incoming bind. Alternatively, if you have an existing set of hardcoded APPC definitions used by TPC62, you can reuse them. At the APPC layer, no difference exists between TCP62 and Enterprise Extender.

The following figure shows an example CONNECTION definition viewed using the command `CEDA VIEW CONNECTION(EE1) GROPUT(EECONN)`.

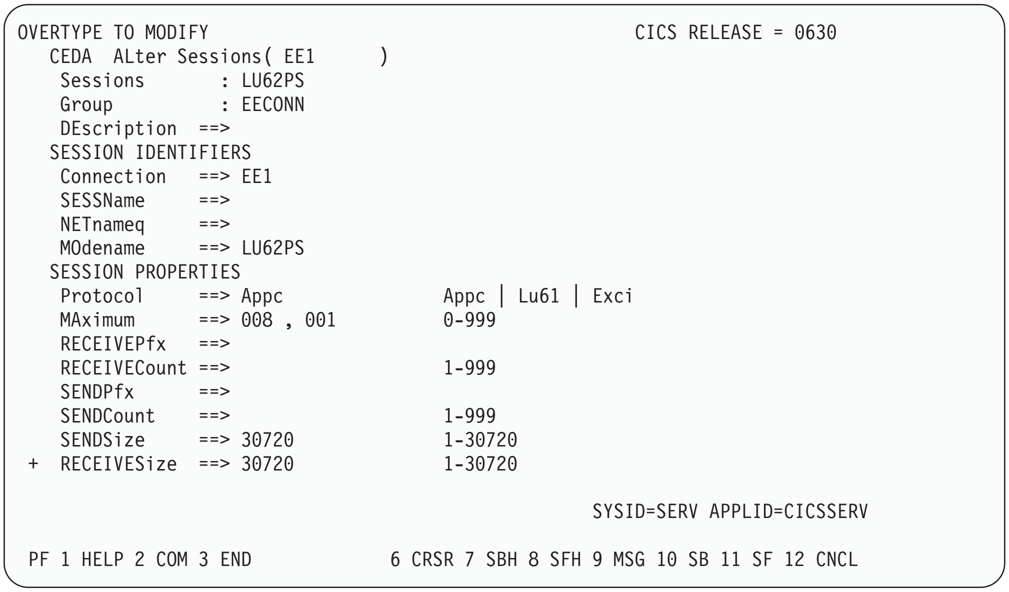


```
OVERTYPE TO MODIFY                                       CICS RELEASE = 0630
   CEDA  ALter CONnection( EE1  )
    CONnection     : EE1
    Group          : EECONN
    DEscription  ==>
   CONNECTION IDENTIFIERS
    Netname      ==> CICSCLI1
    INDsys       ==>
   REMOTE ATTRIBUTES
    REMOTESYSTem ==>
    REMOTEName   ==>
    REMOTESYSNet ==>
   CONNECTION PROPERTIES
    ACcessmethod ==> Vtam              Vtam | IRc | INdirect | Xm
    PRotocol     ==> Appc              Appc | Lu61 | Exci
    Conntype     ==>                   Generic | Specific
    SInglesess   ==> No                No | Yes
    DAtastream   ==> User              User | 3270 | SCs | STrfield | Lms
 +  RECordformat ==> U                 U | Vb

                                                     SYSID=SERV APPLID=CICSSERV

 PF 1 HELP 2 COM 3 END            6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
```

Figure 38. Example CONNECTION definition viewed using the CEDA VIEW CONNECTION command

The CICS connection definition defines the mode parameters for a particular group of sessions bound between two logical units.

The following figure shows an example CONNECTION definition viewed using the command `CEDA VIEW SESSION(LU62PS) GROPUT(EECONN)`.



```
OVERTYPE TO MODIFY                                        CICS RELEASE = 0630
   CEDA  ALter Sessions( EE1      )
    Sessions       : LU62PS
    Group          : EECONN
    DEscription  ==>
   SESSION IDENTIFIERS
    Connection   ==> EE1
    SESSName     ==>
    NETnameq     ==>
    MOdename     ==> LU62PS
   SESSION PROPERTIES
    Protocol     ==> Appc               Appc | Lu61 | Exci
    MAximum      ==> 008 , 001          0-999
    RECEIVEPfx   ==>
    RECEIVECount ==>                    1-999
    SENDPfx      ==>
    SENDCount    ==>                    1-999
    SENDSize     ==> 30720              1-30720
 +  RECEIVESize  ==> 30720              1-30720

                                                      SYSID=SERV APPLID=CICSSERV

 PF 1 HELP 2 COM 3 END             6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
```

Figure 39. Example CONNECTION definition viewed using the CEDA VIEW SESSION command

To install a hardcoded APPC connection and session definition in CICS, you can use the CICS command `CEDA INS GROUP(EECONN)`, where EECONN is the RDO group containing the connection and session definitions.

Accessibility features

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Visit the IBM Accessibility Center for more information about IBM's commitment to accessibility.

Index

Notices

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

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

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

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

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

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

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

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

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 United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

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

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

Trademarks

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

AnyNet
CICS
IBM
IBMLink
System Z
VTAM
z/OS
zSeries

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

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

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

Sending your comments to IBM

If you especially like or dislike anything about this publication, use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness.

Limit your comments to the information in this publication and the way in which the information is presented.

To ask questions, make comments about the functions of IBM products or systems, or to request additional publications, contact your IBM representative or your IBM authorized remarketer.

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

You can send your comments to IBM in any of the following ways:

- By mail, to this address:

  User Technologies Department (MP095)
  IBM United Kingdom Laboratories
  Hursley Park
  WINCHESTER,
  Hampshire
  SO21 2JN
  United Kingdom
- By fax:
  - +44 1962 842327 (if you are outside the UK)
  - 01962 842327 (if you are in the UK)
- Electronically, use the appropriate network ID:
  - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
  - IBMLink™: HURSLEY(IDRCF)
  - Internet: idrcf@hursley.ibm.com

Whichever you use, ensure that you include:

- The publication title and order number
- The topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.