



# BEA WebLogic Java Adapter for Mainframe

## SNACRM Administration Guide

BEA SNACRM Administration Guide  
Document Edition 4.1  
October 2000

## Copyright

Copyright © 2000 BEA Systems, Inc. All Rights Reserved.

## Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

## Trademarks or Service Marks

BEA, ObjectBroker, TOP END, and Tuxedo are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Connect, BEA Manager, BEA MessageQ, BEA Jolt, M3, eSolutions, eLink, WebLogic, WebLogic Enterprise, WebLogic Commerce Server, and WebLogic Personalization Server are trademarks of BEA Systems, Inc.

All other company names may be trademarks of the respective companies with which they are associated.

### BEA SNACRM Administration Guide

Document Edition	Part Number	Date	Software Version
4.1	N/A	October 2000	BEA WebLogic Java Adapter for Mainframe Version 4.1

---

# Contents

## About This Document

What You Need to Know .....	viii
How to Use The Documentation .....	viii
e-docs Web Site .....	viii
Related Information.....	ix
Documentation Conventions .....	ix
Contact Us! .....	xi

## 1. Understanding the SNACRM

About the SNACRM .....	1-1
System Configuration.....	1-2
Local Configuration .....	1-3
Distributed Configurations .....	1-4
Remote Host Domain Configuration.....	1-5
Establishing the VTAM Configuration .....	1-6
Configuring the CICS/ESA LU.....	1-6
Creating Connections at the Remote Host .....	1-6
Defining the Session at the Remote Host.....	1-7
View Connection and Session Status.....	1-8
Completing Cross-Platform Definitions.....	1-8
VTAM Cross-Platform Definitions .....	1-8
OS/390 Platform Definitions .....	1-12
Microsoft SNA Cross-Platform Definitions .....	1-13
Setting SNACRM and Stack Traces .....	1-16

## 2. Sample Configurations and JCL

Sample VTAM Configurations .....	2-1
----------------------------------	-----

---

Local Environment .....	2-2
Remote Environment .....	2-2
Sample Environments .....	2-2
Machine Attributes (LAN Descriptions) .....	2-2
Local Environment for HPSNAPLus2 Configuration .....	2-5
Remote Environment Configurations .....	2-10
Sample JCL for an MVS Platform .....	2-17
Sample JCL for the SNACRM Command .....	2-17
Sample JCL for the CRMLKON Command .....	2-18
Sample JCL for the CRMLKOFF Command .....	2-20
Sample JCL for the CRMDOWN Command .....	2-21
Sample JCL for the CRMLOGS Command .....	2-22

### 3. Administering the SNA Components

Starting the SNACRM on Unix or Windows NT .....	3-2
Starting the PU2.1 Server .....	3-2
Starting the SNACRM .....	3-3
Starting the SNACRM on OS/390 MVS .....	3-3
Setting Environment Variables .....	3-3
Using the SNACRM Monitor .....	3-4
Prerequisite for Running the JSNACRM Utility on an NT Platform .....	3-5
Running the Java Applet Version .....	3-5
Prerequisites for Running the Java Applet Version .....	3-5
Starting the Java Applet .....	3-6
Running the Java Applet Version .....	3-6
Running the Java Application Version .....	3-10
Activating and De-Activating Links .....	3-10
Using the crmlkon Command .....	3-11
Using the crmlkoff Command .....	3-11
Stopping the SNACRM on OS/390 Unix .....	3-12
Stopping the SNACRM on OS/390 MVS .....	3-13

### A. Command Reference Pages

CRMLOGS .....	A-1
crmlkon .....	A-2

---

Synopsis .....	A-2
Description .....	A-2
Example .....	A-3
Diagnostics .....	A-3
crmlkoff .....	A-4
Synopsis .....	A-4
Description .....	A-4
Example .....	A-5
Diagnostics .....	A-5
crmdown .....	A-5
Synopsis .....	A-5
Description .....	A-5
Example .....	A-6
Diagnostics .....	A-6
SNACRM .....	A-6

## **B. Error Messages**

### **Glossary**

### **Index**



---

# About This Document

This document explains how the BEA SNA Communications Resource Manager (SNACRM) provides the emulation allowing Customer Information Control System (CICS), Distributed Program Link (DPL), and Information Management System (IMS) protocols to flow into and out of the Java environment. This document also describes how to administer application domains with the SNACRM.

This document covers the following topics:

- [Chapter 1, “Understanding the SNACRM,”](#) describes the SNACRM and provides information about remote host domain configurations and sample Virtual Telecommunications Access Method (VTAM) configurations.
- [Chapter 2, “Sample Configurations and JCL,”](#) describes some sample VTAM configurations and sample JCL for running the SNACRM commands on an MVS operating system.
- [Chapter 3, “Administering the SNA Components,”](#) describes how to use the SNACRM for administering communications resources.
- [Appendix A, “Command Reference Pages,”](#) lists and describes SNACRM system commands.
- [Appendix B, “Error Messages,”](#) describes SNACRM error messages.
- [Glossary](#)

---

# What You Need to Know

This document is intended mainly for system administrators and operators who will use the SNACRM to monitor and link communications resources between mainframe and UNIX or Windows NT applications.

## How to Use The Documentation

The Documentation CD included in the package with your product software CD contains an HTML Web User Interface (WUI). The WUI links to HTML versions and PDF versions of BEA WebLogic Java Adapter for Mainframe (JAM) documentation, along with this guide. The WUI should be viewed on an online browser. The PDF versions should be used for printing. (Information on how to view the online documentation is available in the release notes accompanying your product software.)

**Note:** The WUI requires a Web browser that supports HTML 3.0, along with Netscape Navigator 4.0 or later, or Microsoft Internet Explorer 4.0 or later.

You must have the Adobe Acrobat Reader to print the PDF file. If you do not have this reader, you can obtain it free of charge from the Adobe Systems Incorporated home site at [www.adobe.com](http://www.adobe.com). (The WUI contains a hot link to this site.)

## e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the “e-docs” Product Documentation page at <http://e-docs.beasys.com>.



---

# Related Information

The following documents contain information that is relevant to using the SNACRM.

- *BEA WebLogic Java Adapter for Mainframe Installation Guide Version 4.1*
- *BEA WebLogic Java Adapter for Mainframe User Guide Version 4.1*
- *BEA WebLogic Java Adapter for Mainframe Release Notes Version 4.1*

# Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
<b>boldface text</b>	Indicates terms defined in the glossary.
<a href="#">blue text</a>	Indicates hypertext links in PDF documents.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.
<i>italics</i>	Indicates emphasis or book titles.
monospace text	<div>Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard.</div> <div><i>Examples:</i></div> <div>#include &lt;iostream.h&gt; void main ( ) the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</div>

Convention	Item
<b>monospace</b> <b>boldface</b> <b>text</b>	Identifies significant words in code. <i>Example:</i> void <b>commit</b> ( )
<i>monospace</i> <i>italic</i> <i>text</i>	Identifies variables in code. <i>Example:</i> String <i>expr</i>
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. <i>Examples:</i> LPT1 SIGNON OR
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.
[ ]	Indicates optional items in a syntax line. The brackets themselves should never be typed. <i>Example:</i> buildobjclient [-v] [-o name ] [-f file-list]... [-l file-list]...
	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.
...	Indicates one of the following in a command line: <ul style="list-style-type: none"> <li>■ That an argument can be repeated several times in a command line</li> <li>■ That the statement omits additional optional arguments</li> <li>■ That you can enter additional parameters, values, or other information</li> </ul> The ellipsis itself should never be typed. <i>Example:</i> buildobjclient [-v] [-o name ] [-f file-list]... [-l file-list]...
.	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.

---

# Contact Us!

Your feedback on the BEA SNACRM Administration Guide is important to us. Send us e-mail at **docsupport@bea.com** if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the SNACRM documentation.

In your e-mail message, please indicate that you are using the documentation for the SNA Communications Resource Manager, Document Edition 1.0.

If you have any questions about this version of the BEA WebLogic Java Adapter for Mainframe, or if you have problems installing and running the SNACRM, contact BEA Customer Support through BEA WebSupport at **www.bea.com**. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages



# 1 Understanding the SNACRM

This section discusses the following topics:

- [About the SNACRM](#)
- [System Configuration](#)
- [Remote Host Domain Configuration](#)

## About the SNACRM

The System Network Architecture Communications Resource Manager (SNACRM) is used with the BEA WebLogic Java Adapter for Mainframe (JAM) to manage communications resources. The SNACRM coordinates the flow of data between Java applications running on a WebLogic Server platform and applications running on a mainframe. The mainframe applications may use the following protocols: Customer Information Control System/Enterprise System Architecture (CICS/ESA), Distributed Program Link (DPL), or Information Management System (IMS). The SNACRM runs as a separate native process providing emulation that allows CICS/ESA, DPL, and IMS protocols to flow into and out of the Java environment.

Version 4.1 of the SNACRM has been enhanced with the following new features:

- May be installed on an OS/390 platform, both Unix and MVS, eliminating the need for an SNA stack.

- Deploys independently of the WebLogic Server, allowing the WebLogic Server to be started and shutdown while the SNACRM is running.
- Persistent operation in OS/390 installations. If the JAM gateway goes down, the SNACRM does not have to be restarted.

As in prior releases, the SNACRM supports non-transactional Information Management System (IMS) programs using implicit Application Program-to-Program (APPC) support for IMS. Implicit APPC is similar to the CICS/ESA DPL. Any IMS program that sends and receives messages to and from the IMS message queue can be used as either a client or a server without change.

To use the implicit APPC capabilities of IMS, you must modify the `APPCM` file in the `SYS1.PARMLIB` Partitioned Data Set (PDS) under OS/390. The configuration parameters in this file associate the Logical Unit (LU) with the IMS scheduler. You must identify the LU representing the application name used by JAM to access the IMS region and the IMS system ID that provides scheduling for inbound requests. Consult with your mainframe support personnel before making changes to the `APPCM` file.

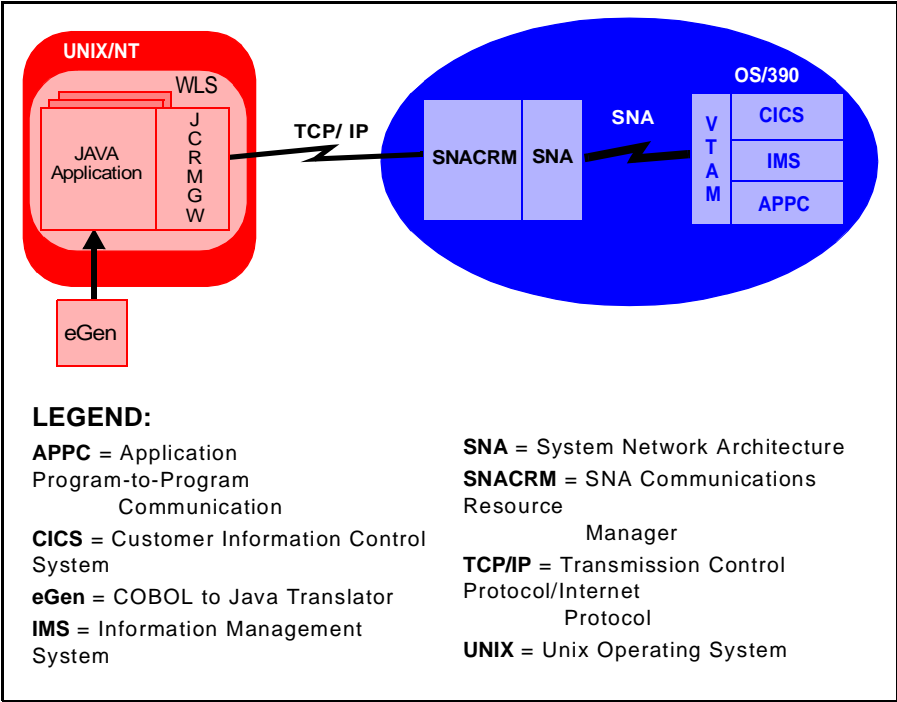
# System Configuration

The SNACRM system may be configured as either a local configuration or a distributed configuration running on Unix, Windows NT, or a mainframe operating system. For a complete list of operating systems, refer to the *BEA WebLogic Java Adapter for Mainframe Release Notes*. If the SNACRM is not run on a mainframe, it must run on the same platform as the SNA stack. If the gateway is running on a platform other than the one the SNACRM is on, then the SNACRM should already be started and monitoring the address specified in the gateway configuration.

# Local Configuration

The local configuration, illustrated in [Figure 1-1](#), combines the Java application, JCRMGW, WebLogic Server, and the SNACRM with the stack (PU2.1 server) on the same Unix or Windows NT platform. It employs the IBM proprietary SNA protocol for communication with the mainframe via the stack.

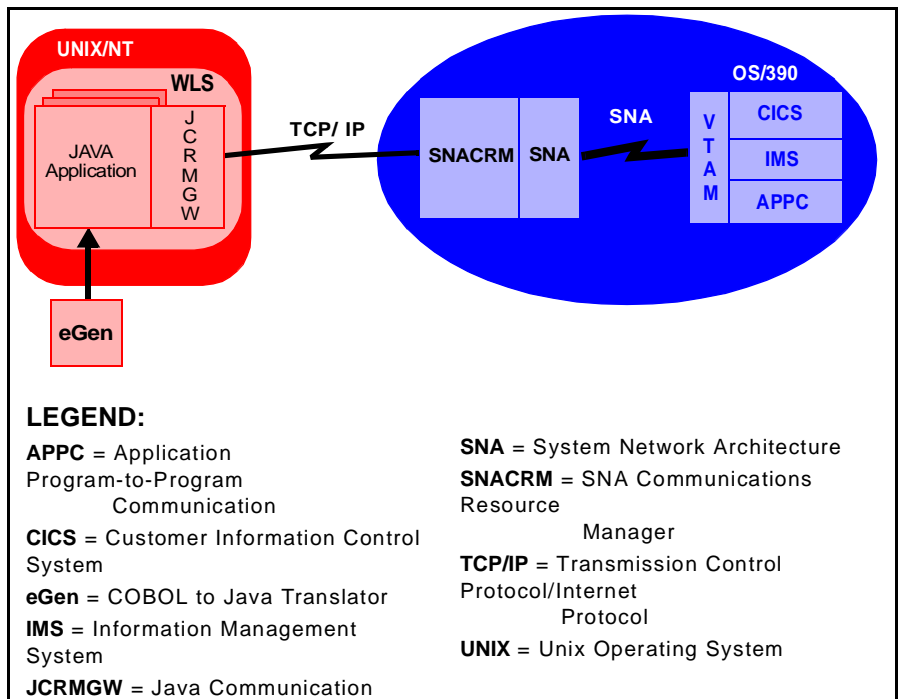
**Figure 1-1 JAM Local Configuration**



## Distributed Configurations

One type of distributed configuration separates the Java applications and JCRMGW from the SNACRM by installing the SNACRM to the IBM OS/390 Mainframe. See [Figure 1-2](#). This configuration eliminates the need for a third-party stack on the Unix or NT machine. Note that this configuration requires a one-to-one relationship between the local JCRMGW and the remote SNACRM.

**Figure 1-2 JAM Distributed Configuration**

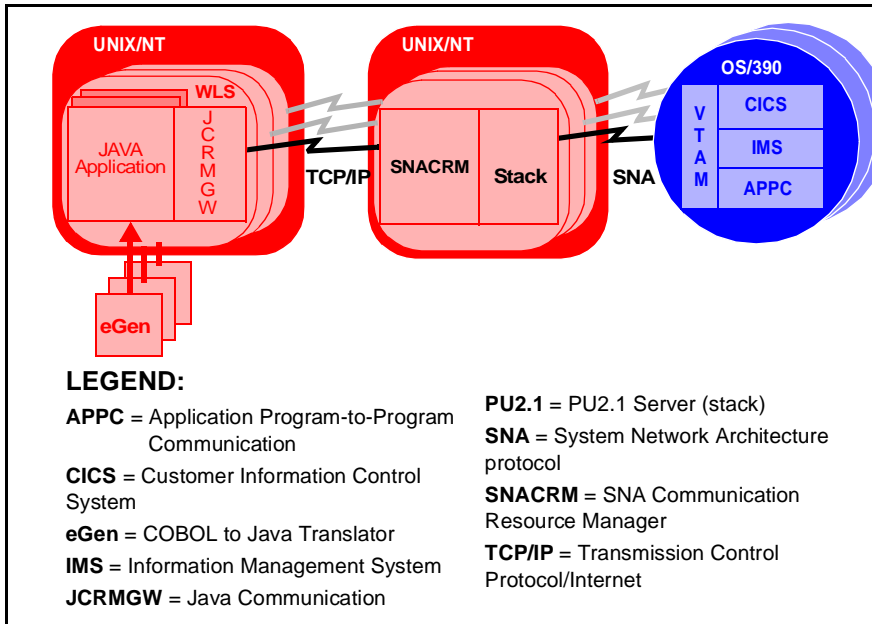


Another type of distributed configuration separates the Java applications and JCRMGW from the SNACRM on different Unix or Windows NT platforms. See [Figure 1-3](#). This configuration employs Transmission Control Protocol/Internet Protocol (TCP/IP) connectivity between the Java applications platform and the SNACRM platforms, as well as the SNA connectivity to the mainframe environment(s). It provides the flexibility to deploy WebLogic Server separately from



the SNACRM at installations that require WebLogic Server on a platform other than the one on which the SNA stack is running. Note that this configuration requires a one-to-one relationship between the local JCRMGW and the remote SNACRM.

**Figure 1-3 Alternate JAM Distributed Configuration**



## Remote Host Domain Configuration

A basic understanding of the mainframe configuration requirements provides a context for understanding the SNACRM functions and configuration requirements.

**Note:** Always consult with your local mainframe system administrator for specific information about your system. Any samples provided illustrate a starting point for configuring your system and do not represent all possibilities. The samples represent one way a mainframe can be configured to work in an Advanced Peer-to-Peer Networking (APPN) Local Area Network (LAN) environment.

You must ensure the CICS/ESA remote domain is prepared to conduct operations with the BEA local domain. This includes:

- [Establishing the VTAM Configuration](#)
- [Configuring the CICS/ESA LU](#)
  - [Creating Connections at the Remote Host](#)
  - [Defining the Session at the Remote Host](#)
- [Completing Cross-Platform Definitions](#)
- [Setting SNACRM and Stack Traces](#)

## Establishing the VTAM Configuration

The SNACRM communicates with the gateway. If your JAM system is used in a Virtual Telecommunications Access Method (VTAM) environment, you must ensure the host configuration supports it. Refer to [“Sample VTAM Configurations,”](#) for some examples based on the requirements for the JAM to be used in a VTAM environment with an Ethernet LAN and an APPN mainframe system.

## Configuring the CICS/ESA LU

Before you can connect to the remote stack, the LU (Logical Unit) configuration must be established. This entails creating connection definitions, creating session definitions, and installing resource definitions.

### Creating Connections at the Remote Host

If a remote connection definition file is not already in place, you must work with the mainframe support personnel to create one. When placed on the remote host, the definition provides a connection with the local domain. Note the following example of a JAM connection definition file:

```
DEFINE CONNECTION(BEA)          GROUP(BEACONN)
DE(JAM EXAMPLE RDO CONNECTION)
ACCESSMETHOD(VTAM)              PROTOCOL(APPC)
NETNAME(**VTAM NETWORK NAME OF REMOTE SYSTEM**)
ATTACHSEC(LOCAL)                AUTOCONNECT(NO)
```

To install the sample connection definition, put it on the host in a separate group. Use the CEDA INSTALL command.

For example:

```
CEDA INSTALL GROUP(BEACONN)
```

## Defining the Session at the Remote Host

If a session definition is not already in place, you must work with the mainframe support personnel to create one. When placed on the remote host, the session definition defines the logical links by which the local domain communicates with the remote host. Note the following example of a JAM session definition:

```
DEFINE SESSION(BEATEST)          GROUP(BEACONN)
CONNECTION(BEA)
DE(JAM EXAMPLE RDO SESSION)
PROTOCOL(APPC)                  AUTOCONNECT(YES)
MODENAME(**MODE**)              MAXIMUM(**SESSNBR**, **WINNER**)
```

AUTOCONNECT indicates how the activation of the session is negotiated. YES enables the CICS/ESA host to negotiate its own winner sessions when a conversation is allocated.

The MODENAME can be either a CICS/ESA-supplied mode name, such as SMSNA100, or your own defined mode name. If another set of session definitions exist for the BEA connection, this mode name must be unique among all sets defined to the connection. The mode name corresponds to the VTAM LOGMODE name.

The MAXIMUM option defines the total number of sessions in the set and the total number of winner sessions. The total number of winner sessions must include those for the host and the remote stack. The WINNER number plus the remote WINNER number should equal the SESSNBR.

View Connection and Session Status

After you have installed group definitions, you can view the status of connections and sessions using the following CICS/ESA system commands:

```
CEMT I CONN(BEA)           **view the status of the connection
CEMT I NET(**Netname**)    **View the status of the sessions
CEMT I MODENAME(**MODE**)  **View the status of the mode
```

Completing Cross-Platform Definitions

Consult with your CICS/ESA remote domain administrator to obtain key parameters in the VTAM definition that must be included in the SNA stack configuration, as well as in other configuration files in the JAM local domain.

VTAM Cross-Platform Definitions

Before installing JAM software, please examine [Table 1-1](#) for a summary of cross-platform definitions. Consult with the VTAM system administrator to obtain the value indicated in the *Name* column and make the corresponding entries shown in the *Needed In* column.

Table 1-1 Summary of Cross-Platform Definitions

Item	Name	Originates In	Needed In
1.	SNA Network ID (e.g. <b>SNANET1</b> )	VTAM configuration	<b>SunLink SNA Stack Configuration:</b> Example: CP NQ_CP_NAME= <b>SNANET1</b> .SPARC1 DLC RMTNQ_CP_NAME= <b>SNANET1.VTAMHOST</b> LU NQ_LU_NAME= <b>SNANET1.L0P0024A</b> PTNR_LU NQ_LU_NAME= <b>SNANET1.CICSSYN</b>
	and VTAM Host ID (e.g. <b>VTAMHOST</b> )		<b>SNaplus2 or IBM CS/AIX Stack Configuration:</b> Example: fqcp_name= <b>SNANET1</b> .SPARC1 adj_cp_name= <b>SNANET1.VTAMHOST</b> fqplu_name= <b>SNANET1.CICSSYN</b>

Table 1-1 Summary of Cross-Platform Definitions

Item	Name	Originates In	Needed In
2.	Mode Name (e.g. <b>SNA62</b> )	VTAM-MODEENT definition	<b>CICS Sessions Definition:</b> Example: MODENAME ( <b>SNA62</b> )  <b>SunLink SNA Stack Configuration:</b> Example: MODE     NAME= <b>SNA62</b>  <b>SNAplus2 Stack Configuration:</b> Example: mode_name= <b>SNA62</b>  <b>JCRMGW Configuration:</b> Example: *JC_SNA_LINKS MODENAME= " <b>SNA62</b> "
3.	Control Point Name CPNAME (e.g. <b>SPARC1</b> )	VTAM-PU definition	<b>SunLink SNA Stack Configuration:</b> Example: CP     NAME= <b>SPARC1</b> CP     NQ_CP_NAME=SNANET1 . <b>SPARC1</b>  <b>SNAplus2 Stack Configuration:</b> Example: fqcp_name=SNANET1 . <b>SPARC</b> cd_alias= <b>SPARC1</b>
4.	Local LU Name (e.g. <b>L0F0024A</b> )	VTAM-LU definition	<b>CICS CONNECTION definition:</b> Example: NETNAME ( <b>L0F0024A</b> ) <b>SunLink SNA Stack Configuration:</b> Example: LU     NAME= <b>L0F0024A</b> LU     NQ_LU_NAME=SNANET1 . <b>L0F0024A</b> PTNR_LU     LOC_LU_NAME= <b>L0F0024A</b> TP     LOC_LU_NAME= <b>L0F0024A</b>  <b>SNAplus2 Stack Configuration:</b> Example: lu_name= <b>L0F0024A</b> lu_alias= <b>L0F0024A</b>  <b>JCRMGW Configuration:</b> Example: *JC_SNA_STACKS LOCALLU= " <b>L0F0024A</b> "

# 1 Understanding the SNACRM

**Table 1-1 Summary of Cross-Platform Definitions**

Item	Name	Originates In	Needed In
5.	CICS LU Name (e.g. <b>CICSSYN</b> )	VTAM-LU definition	<b>SunLink SNA Stack Configuration:</b> Example: PTNR_LU NAME= <b>CICSSYN</b> PTNR_LU NQ_LU_NAME=SNANET1. <b>CICSSYN</b> MODE PTNR_LU_NAME= <b>CICSSYN</b>  <b>SNaplus2 Stack Configuration:</b> Example: fqplu_name=SNANET1. <b>CICSSYN</b> plu_alias= <b>CICSSYN</b>  <b>JCRMGW Configuration:</b> Example: *JC_SNA_LINKS RLUNAME= " <b>CICSSYN</b> "
6.	Terminal Identifier (e.g. <b>05DF0024</b> )	VTAM (IDNUM+IDBLK)	<b>SunLink SNA Stack Configuration:</b> Example: DLC TERM << ALIGN=CHAR  <b>SNaplus2 Stack Configuration:</b> Example: node_id=< <b>05000002</b> >
7.	Local Network Device (e.g. <b>/dev/tr</b> )	UNIX Configuration	<b>SunLink SNA Stack Configuration:</b> Example: TRLINE DEVICE= ' <b>/dev/tr</b> ' or SDLCLINE DEVICE= ' <b>/dev/dcp1</b> '
8.	Local MAC Address (token ring only)	Token ring address of Host	<b>SunLink SNA Stack Configuration:</b> Example: TRLINE Source_Address=X'080020117d7a'
9.	Remote MAC Address (token ring only)	Token ring address of local machine	<b>SunLink SNA Stack Configuration:</b> Example: DLC RMTMACADDR=X'40000101000'  <b>SNaplus2 Stack Configuration:</b> Example: mac_address=<400031720001>
10.	LAN Speed (e.g. <b>4MBs</b> )	Speed of token ring network	<b>SunLink SNA Stack Configuration:</b> Example: TRLINE LAN_rate=RING_ <b>4MBS</b>

**Table 1-1 Summary of Cross-Platform Definitions**

Item	Name	Originates In	Needed In
11.	SDLC parameters (line protocol)	VTAM-line definition	<b>SunLink SNA Stack Configuration:</b> Example: SDLC LINE    DUPLEX=half LINE=switched NRZI=no PAUSE=1 SPEED=4800
12.	Partner Definition (e.g. <b>TUXPART1</b> )	CICS/ESA	<b>Mainframe Client Application:</b> (for example, COBOL with embedded CPI-C to route CICS to appropriate LU for BEA Connect SNA) COBOL Example: MOVE ' <b>TUXPART1</b> ' TO SYM-DEST-NAME CALL 'CMINIT' USING CONVERSATION-ID SYM-DEST-NAME, CM_RETCODE END-CALL
13.	Set LU definition so maximum sync-level allowed corresponds to JCRMGW.CFG entry:	Stacks	<b>SunLink SNA Stack Configuration:</b> Example: SYNC_LVL=SYNCPPT <b>SNAplus2 Stack Configuration:</b> Example: [define local_lu] Syncpt.Support=NO
14.	Map all incoming conversations to BEA Connect SNA gateway (make sure TPs have all privileges available, e.g. CNOS, syncpoint if licensed, service conversations, etc.).	Stacks	<b>SunLink SNA Stack Configuration:</b> Example: TP_HEXNAME=x'2a' <b>SNAplus2 Stack Configuration:</b> Example: Sna_tps <404040...hex representation of 64 EBCDIC spaces...404040> TYPE=QUEUED TIMEOUT=-1 USERID=authorized_user_here GROUP=authroized_group_here LUALIAS=lu_name_here

**Table 1-1   Summary of Cross-Platform Definitions**

Item	Name	Originates In	Needed In
15.	CICS Transaction IDs (e.g. TOUP)	CICS/ESA	<b>JCRMGW Configuration:</b> Example: *JC_REMOTE_SERVICES

## OS/390 Platform Definitions

Before installing JAM software, please examine [Table 1-2](#) for a summary of OS/390 SNA definitions. Consult with the system administrator to obtain the value indicated in the *Name* column and make the corresponding entries shown in the *Needed In* column.

**Table 1-2   Summary of OS/390 SNA Definitions**

Item	Name	Originates In	Needed In
1.	Local LU Name (e.g. <b>BEAAPPL1</b> )	VTAM-LU definition	<b>CICS CONNECTION definition:</b> Example: NETNAME ( <b>BEAAPPL1</b> )  <b>VTAM Configuration:</b> Example: BEASNA VBUILD TYPE=APPL4 BEAAPPL1 APPL ACB=BEAAPPL1 , APPC=YES  <b>JCRMGW Configuration:</b> Example: *JC_SNA_STACKS LOCALLU= " <b>BEAAPPL1</b> "



**Table 1-2 Summary of OS/390 SNA Definitions**

Item	Name	Originates In	Needed In
2.	Mode Name (e.g. <b>SNA62</b> )	VTAM-MODEENT definition	<b>CICS Sessions Definition:</b> Example: MODENAME ( <b>SNA62</b> )  <b>JCRMGW Configuration:</b> Example: *JC_SNA_LINKS MODENAME= " <b>SNA62</b> "  <b>VTAM Configuration (not required):</b> Example: MODEENT= <b>SNA62</b>
3.	CICS LU Name (e.g. <b>CICSSYN</b> )	VTAM-LU definition	<b>JCRMGW Configuration:</b> Example: *JC_SNA_LINKS RLUNAME= " <b>CICSSYN</b> "
4.	Set LU definition so maximum sync-level	Stacks	<b>VTAM Configuration:</b> Example: SYNCLVL=CONFIRM
5.	CICS Transaction IDs (e.g. TOUP)	CICS/ESA	<b>JCRMGW Configuration:</b> Example: *JC_REMOTE_SERVICES

## Microsoft SNA Cross-Platform Definitions

It is important to communicate with the administrator of the CICS/ESA remote domain to obtain key parameters in the VTAM definition that must be included in the Microsoft SNA Server configuration, as well as in other configuration files in the JAM local domain.

Before installing JAM software, please examine the following general procedure for configuring the Microsoft SNA Server. Use the Microsoft SNA Server Manager GUI. Sample values are shown in parenthesis. Consult with the VTAM system administrator to obtain the proper values.

1. Start Microsoft SNA Server Manager from the Start button on the Task Bar.

2. When a server is automatically created (MVSNT1), note the configuration values displayed in the Server Properties window:

Server: MVSNT1  
Subdomain: MVSNT1  
Server Role: Primary  
Network Transports: TCP/IP

3. Under Link Services, define a link service (SNADLC1):

In the Link Service Properties, define DLC 802.2 Link Service Configuration:

Title: DLC 802.2 Link Service #1  
Adapter: <your ethernet adapter>  
Local Service Access Point (SAP): 0x4  
Use Fixed SAP

4. Under SNA Service, Connections, define an 802.2 connection (MVSNT1):

In the MVSNT1 Properties, define:

### General

Name: MVSNT1  
Link Service: SnaDlc1  
Remote End: Peer System  
Allowed Directions: Both Directions  
Activation: On Server Startup  
Supports Dynamic Remote APPC LU Definition

### Address

Remote Network Address: <host MAC address>  
Remote SAP Address: <host SAP address>

### System Identification

Local Node Name  
Network Name: <mynetwork>  
Control Point Name: MVSNT1  
Local Node ID: <xxx nnnn>XID Type: Format 3  
Remote Node Name  
Network Name: <hostnetwork>  
Control Point Name: <vtamcpname>  
Remote Node ID: Peer DLC Role: Negotiable  
Compression Type: None

### 802.2 DLC

Take Defaults

5. Under Local APPC LUs (SNA Service: Connections: Insert: APPC: Local LU), define a local LU (LUNT1A) in the LUNT1A Properties:

General

LU Alias: LUNT1A  
Network Name: <mynetwork>  
LU Name: LUNT1A

Advanced

Take Defaults

6. Under Remote APPC LUs, define a remote LU (CICS1) in the CICS1 Properties:

General

Connection: MVSNT1  
LU Alias: CICS1  
Network Name: <hostnetwork>  
LU Name: CICS1  
Uninterpreted Name: CICS1

Options

Take Defaults

7. Under APPC Modes, define a mode (SMSNA100) in the SMSNA100 Properties:

General

Mode Name: SMSNA100

Limits

Parallel Session Limit: <max sessions>  
Minimum Winner Contention Limit: <min winners>  
Partner Min Winner Contention Limit: <max sessions - min winners>  
Automatic Activation Limit: 0

Characteristics

Take Defaults

Partners

Add partnership for Server Name: MVSNT1 between Local LU: LUNT1A  
and Partner LU: CICS1

Compression

Take Defaults

# Setting SNACRM and Stack Traces

Trace capabilities are provided for the SNACRM process. To initiate stack tracing, you include the `-t` option for the `SNACRM` command along with a number to specify the level of tracing. Refer to [“Trace Options”](#) for the `SNACRM` command in the [“Command Reference Pages.”](#)

The APPC Stack API trace is either enabled or disabled using the `-s` option of the `SNACRM` command (refer to [“General Options”](#) for the `SNACRM` command in the [“Command Reference Pages”](#)). If enabled, it generally shows the parameters and results of all API calls. Depending on the stack being used, other options (such as vendor-specified environment variables) may have to be activated for `SNACRM` to enable the trace. Consult the vendor publications for instructions on how to set up stack tracing.

# 2 Sample Configurations and JCL

This section discusses the following topics:

- [Sample VTAM Configurations](#)
  - [Local Environment](#)
  - [Remote Environment](#)
  - [Sample Environments](#)
- [Sample JCL for an MVS Platform](#)
  - [Sample JCL for the SNACRM Command](#)
  - [Sample JCL for the CRMLKON Command](#)
  - [Sample JCL for the CRMLKOFF Command](#)

## Sample VTAM Configurations

The following sections provide sample environments showing how JAM software can be configured for use with an Ethernet LAN and an APPN System 390. Considerations for token ring and subarea-style configurations are included. It is assumed that hardware and operating system installations have been completed.

An environment properly configured for the JAM system involves two general components, a local environment and a remote environment.

### Local Environment

A local environment is a Unix-based or Windows NT-based machine running JAM software. The JAM is a fully bidirectional program, supporting the local system as either a client or server. This environment consists of the following components:

- Hardware: any workstation and network interface supported by the required software
- Platform operating systems with protocol stacks (PU 2.1 servers)

### Remote Environment

A remote environment is an IBM mainframe that may or may not be on the same local network. As in the local environment, JAM is a fully bidirectional program, supporting the remote system as either a client or server. This environment consists of the following components:

- Hardware: any workstation and network interface supported by the required software
- Software: any supported host software

### Sample Environments

Samples of each environment are provided to illustrate a starting point when first configuring your system. These samples are not intended to be used without modifications. Any similarity between them and any actual system is coincidental.

### Machine Attributes (LAN Descriptions)

The attributes of the sample environment machines are listed in the following tables for reference. The subsequent sample configurations refer to these attributes as required. Sample configurations are presented for example only and may contain components that are not supported for your system

**Table 2-1 SPARCstation 5**

<b>Name</b>	<b>Attribute</b>
OS	Solaris 2.6 or 2.7
SNA	SunLink 9.1
APP	Tuxedo 6.5
MAC	08:00:20:7C:47:50
IP	206.189.43.14
NAME	beasun2

**Table 2-2 SPARCstation 5**

<b>Name</b>	<b>Attribute</b>
OS	Solaris 2.6 or 2.7
SNA	SunLink 9.1
APP	Tuxedo 6.5
MAC	08:00:20:87:47:2d
IP	206.189.43.54
NAME	dalsun4

**Table 2-3 HP 9000/847**

<b>Name</b>	<b>Attribute</b>
OS	HP-UX B.10.20 or 11.00 (patches:PHNE_9663,9761,9889)
SNA	HPSNAPLus2 5.1 or 6.0
APP	Tuxedo 6.5
MAC	08:00:09:30:24:77

**Table 2-3 HP 9000/847**

Name	Attribute
IP	206.189.43.13
NAME	dalhp10

**Table 2-4 P390 Server 500**

Name	Attribute
OS	MVS 5.22 9510
SNA	VTAM 4.3
APP	CICS 4.1 / IMS DC 5.1
MAC	10:00:5a:d4:3e:8e
IP	206.189.43.98
NAME	beavs

**Table 2-5 P390 Server 500**

Name	Attribute
OS	OS/390 1.2
SNA	VTAM 4.3
APP	CICS 4.1
MAC	10:00:5a:d4:c1:e0
IP	206.189.43.96
NAME	dalvs2



**Table 2-6 Windows NT Server**

Name	Attribute
OS	Windows NT Server 4.0, SP2
SNA	MS SNA Server 4.0, SP3
APP	CICS 4.1
MAC	10:00:5a:d4:c1:e0
IP	206.189.43.99
NAME	dalnt

## Local Environment for HPSNAPPlus2 Configuration

HPSNAPPlus2 configurations are usually setup using the HP xSnapAdmin utility, resulting in the configuration text file `/etc/opt/sna/sna_node.cfg`. This file can be manually created and/or maintained using a text editor; however, using the HP `xsnapadmin` utility is recommended. The following example is the `sna_node.cfg` file for the sample environment.

### Listing 2-1 HPSNAPPlus2 Configuration

---

```
[define_node_config_file]
major_version = 5
minor_version = 1
update_release = 1
revision_level = 116

[define_node]
node_name = dalhp10
description = snacrm development
node_type = END_NODE
fqcp_name = BEALAN.DALHP10
cp_alias = dalhp10
mode_to_cos_map_supp = NO
mds_supported = YES
node_id = <05ffffff>
max_locates = 100
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
dlur_support = YES
pu_conc_support = NO
nn_rar = 128
ptf_flags = NONE
```

### **[define\_ethernet\_dlc]**

```
dlc_name = ETHER0
description = ""
neg_ls_supp = YES
initially_active = NO
adapter_number = 0
```

### **[define\_ethernet\_port]**

```
port_name = ethl0
description = 1st ethernet adapter
dlc_name = ETHER0
port_type = PORT_SATF
port_number = 1
max_rcv_btu_size = 1033
tot_link_act_lim = 64
inb_link_act_lim = 0
out_link_act_lim = 0
ls_role = LS_NEG
act_xid_exchange_limit = 9
nonact_xid_exchange_limit = 5
ls_xmit_rcv_cap = LS_TWS
max_ifrm_rcvd = 7
target_pacing_count = 7
max_send_btu_size = 1033
mac_address = <000000000000>
lsap_address = 0x08
```

```
implicit_cp_cp_sess_support = NO
implicit_limited_resource = NO
implicit_deact_timer = 0
effect_cap = 3993600
connect_cost = 0
byte_cost = 0
security = SEC_NONSECURE
prop_delay = PROP_DELAY_LAN
user_def_parm_1 = 0
user_def_parm_2 = 0
user_def_parm_3 = 0
initially_active = YES
test_timeout = 5
test_retry_limit = 2
xid_timeout = 5
xid_retry_limit = 2
tl_timeout = 5
tl_retry_limit = 5
```

```
[define_ethernet_ls]
ls_name = P390HP10
description = P390 - beavs
port_name = eth10
adj_cp_name = P390.USS3270
adj_cp_type = LEARN_NODE
mac_address = <0020af543176>
lsap_address = 0x08
auto_act_supp = NO
tg_number = 0
limited_resource = NO
solicit_sscp_sessions = NO
pu_name = <0000000000000000>
disable_remote_act = NO
default_nn_server = NO
dspu_services = NONE
dspu_name = <0000000000000000>
dlus_name = <00000000000000000000000000000000>
bkup_dlus_name = <00000000000000000000000000000000>
link_deact_timer = 0
use_default_tg_chars = YES
ls_attributes = SNA
adj_node_id = <00000000>
local_node_id = <00000000>
cp_cp_sess_support = YES
effect_cap = 3993600
connect_cost = 0
byte_cost = 0
security = SEC_NONSECURE
prop_delay = PROP_DELAY_LAN
```

```
user_def_parm_1 = 0
user_def_parm_2 = 0
user_def_parm_3 = 0
target_pacing_count = 7
max_send_btu_size = 1033
ls_role = USE_PORT_DEFAULTS
initially_active = NO
react_timer = 30
react_timer_retry = 65535
test_timeout = 5
test_retry_limit = 2
xid_timeout = 5
xid_retry_limit = 2
tl_timeout = 5
tl_retry_limit = 5

[define_local_lu]
lu_name = LUHP10A
description = Test LU #1
lu_alias = LUHP10A
nau_address = 0
syncpt_support = YES
lu_session_limit = 0
default_pool = NO
pu_name = <00000000000000000>
sys_name = ""
timeout = -1
back_level = NO

[define_local_lu]
lu_name = LUHP10B
description = Test LU #2
lu_alias = LUHP10B
nau_address = 0
syncpt_support = YES
lu_session_limit = 0
default_pool = NO
pu_name = <00000000000000000>
sys_name = ""
timeout = -1
back_level = NO

[define_local_lu]
lu_name = LUHP10C
description = Test LU #3
lu_alias = LUHP10C
nau_address = 0
syncpt_support = YES
lu_session_limit = 0
```

```
default_pool = NO
pu_name = <0000000000000000>
sys_name = ""
timeout = -1
back_level = NO

[define_partner_lu]
description = APPC MVS LU for IMS
fqplu_name = P390.MVSLU01
plu_alias = MVSLU01
plu_un_name = MVSLU01
max_mc_ll_send_size = 32767
conv_security_ver = NO
parallel_sess_supp = YES

[define_partner_lu]
description = backend cics #1
fqplu_name = P390.C410XB01
plu_alias = C410XB01
plu_un_name = C410XB01
max_mc_ll_send_size = 32767
conv_security_ver = NO
parallel_sess_supp = YES

[define_partner_lu]
description = Second backend cics
fqplu_name = P390.C410XB02
plu_alias = CICS2
plu_un_name = C410XB02
max_mc_ll_send_size = 32767
conv_security_ver = NO
parallel_sess_supp = YES

[define_mode]
mode_name = SMSNA100
description = Sessions: 10 -- 5,5
max_ru_size_upp = 1024
receive_pacing_win = 4
default_ru_size = YES
max_neg_sess_lim = 256
plu_mode_session_limit = 10
min_conwin_src = 5
cos_name = #eLink
cryptography = NONE
auto_act = 0
```

---

### Remote Environment Configurations

You must involve your mainframe system support personnel early in the process of setting up your configuration. In a large shop there will most likely be different individuals responsible for Multiple Virtual Storage (MVS), VTAM, CICS, and IMS. Make sure everyone is involved. Most of the configuration for your mainframe may have already been done.

These samples are provided for illustration. Mainframe technical support is not trivial, and this information is not intended to explain all of the possible configurations. These samples represent one way a P390 can be configured to work in an APPN LAN environment.

#### ATCSTRxx VTAM Start List

The following example is the VTAM start list for the BEAVS P390 machine. It supports both the subarea and APPN environments.

#### **Listing 2-2 VTAM Start List for BEAVS P390**

---

```
* ----- *
* VTAM START LIST FOR SYS1 *
* ----- *
BN=YES,
BNDYN=FULL,
XNETALS=YES,
SSCPID=06,NOPROMPT,
CONFIG=00,MAXSUBA=31,SUPP=NOSUP,
SSCPNAME=USS3270,
NETID=P390,
NODETYPE=NN,
HOSTSA=6,
CRPLBUF=(208,,15,,1,16),
IOBUF=(100,512,19,,1,20),
LFBUF=(104,,0,,1,1),
LPBUF=(64,,0,,1,1),
SFBUF=(163,,0,,1,1)
```

---

## XCA Major Node Defines the LAN Adapter for SYS1

This definition is set up for use with a 3172 (emulated) for connecting an APPN network node to another APPN node. Note that it is for an Ethernet LAN, and the SAPADDR specified must be the same as the LSap specified for the local link station.

### Listing 2-3 XCA Major Node

---

```
XETH2LP1  VBUILD TYPE=XCA  ** EXTERNAL COMMUNICATION ADAPT**
PORTE2    PORT  ADAPNO=1,      ** 3172 RELATIVE ADAPTER NUMBER**
           CUADDR=E22,        ** CHANNEL UNIT ADDRESS          **
           MEDIUM=CSMACD,    ** LAN TYPE=ETHERNET            **
           SAPADDR=8,        ** SERVICE ACCESS POINT ADDRESS**
           TIMER=120         ** CHANNEL ACTIVATE RESP TIME    **

*
G1ETH2    GROUP DIAL=YES,      ** YES required for putype 2  **
           DYNPU=YES,
           CALL=INOUT,
           ANSWER=ON,
           ISTATUS=ACTIVE

LETH20    LINE
PETH20    PU
LETHE3    LINE
PETHE3    PU
LETHF3    LINE
PETHF3    PU
```

---

## Switched Network (SWNET) Definitions

The three switched network definition examples in this section specify the VTAM PU (Physical Unit), representing the local link stations that expect to connect with the host machine. The IDBLK and IDNUM definitions are provided to support 3270 traffic and must be unique, as well as match the values specified in the local link definition.

The APPLID definition shown in [Listing 2-6](#) replaces the switched network definition for VTAM as well as any local stack configuration for Unix because a stack on a local Unix machine is no longer needed.

### Listing 2-4 SWNET Major Node (DALHP10)

---

```
SWNETHHP    VBUILD    TYPE=SWNET,MAXNO=3,MAXGRP=3
P390HP10    PU    ADDR=02,
              IDBLK=05F,
              IDNUM=FFFFF,
              PUTYPE=2,
              NETID=BEALAN,
              CPNAME=DALHP10,
              MAXPATH=3,
              DWACT=YES,
              CONNTYPE=APPN,
              CPCP=YES,
              DYNLU=YES

* -----
*  SNA SAP & HP10 MAC ADDRESS BIT REVERSED FOR TRFMT
* -----
PATHHP      PATH    DIALNO=00041000900C24EE,
              GRPNM=G1ETH2
LUHP10A LU    LOCADDR=0
LUHP10B LU    LOCADDR=0
LUHP10C LU    LOCADDR=0
```

---

### Listing 2-5 SWNET Major Node (SUN2)

---

```
SWNETH2     VBUILD    TYPE=SWNET,MAXNO=3,MAXGRP=3
P390ETH2    PU    ADDR=04,
              IDBLK=019,
              IDNUM=10092,
              PUTYPE=2,
              NETID=BEALAN,
              CPNAME=SUN2,
              MAXPATH=3,
              DWACT=YES,
              CONNTYPE=APPN,
              CPCP=YES,
              DYNLU=YES
```



```
* -----
* SNA SAP & SUN2 MAC ADDRESS BIT REVERSED FOR TRFMT
* -----
PATH01      PATH DIALNO=00081000043EE20A,
              GRPNM=G1ETH2
LUSUN2A LU   LOCADDR=0
LUSUN2B LU   LOCADDR=0
LUSUN2C LU   LOCADDR=0
```

### Listing 2-6 APPLID Definition (OS/390)

```
BEASNA VBUILD TYPE=APPL
BEAAPPL1 APPL ACBNAME=BEAAPPL1,
              AUTH=(ACQ,PASS),
              APPC=YES,
              SYNCLVL=SYNCPT,
              PARSESS=YES
```

## VTAM Application Major Nodes for CICS Regions

These examples represent the partner LU definitions to be accessed from the local environment. The APPL names must match those specified in the partner LU definitions on the local machine.

### Listing 2-7 Partner LU Definitions

```
BEACICS VBUILD TYPE=APPL                      APPLICATION MAJOR NODE
* APPL DEFINITION STATEMENTS FOR CICS
* CICS 4.10 BACKEND REGION #1 SYSID=B41A
C410XB01 APPL EAS=64,                          ESTIMATED CONCURRENT SESSIONS
              MODETAB=ISTINCLM,                  MAKE SURE DEFAULT MODETAB
              PARSESS=YES,
              AUTH=(ACQ,BLOCK,PASS) CICS CAN ACQUIRE & PASS TMLS
                                   CICS CAN REQUEST BLOCKED INPUT

C410XB02 APPL EAS=64,                          ESTIMATED CONCURRENT SESSIONS
              MODETAB=ISTINCLM,                  MAKE SURE DEFAULT MODETAB
              PARSESS=YES,
              AUTH=(ACQ,BLOCK,PASS) CICS CAN ACQUIRE & PASS TMLS
                                   CICS CAN REQUEST BLOCKED INPUT
```

## 2 Sample Configurations and JCL

---

```
C410XB03 APPL EAS=64, ESTIMATED CONCURRENT SESSIONS
          MODETAB=ISTINCLM, MAKE SURE DEFAULT MODETAB
          PARSESS=YES,
          AUTH=(ACQ,BLOCK,PASS) CICS CAN ACQUIRE & PASS TMLS
                                CICS CAN REQUEST BLOCKED INPUT
* #####
* END OF BEACICS APPL DEF
* #####
```

---

### CICS Resource Definition Entries (RDO)

CICS connection and session definitions map the VTAM path definitions for the CICS application. Each connection represents one local LU definition in the local SNA configuration. Therefore, the names must match.

CICS session definitions associate a VTAM mode with the LU specified in the connection. The mode names and session count characteristics must match those specified in the mode definitions for the local SNA configuration. Note that these definitions set `AUTOCONNECT` to `YES`, allowing automatic session acquisition for a CICS client application.

#### Listing 2-8 CICS Resource Definition Entries

---

```
LIST GROUP(BEAHP10) OBJECTS
GROUP NAME: BEAHP10
-----
CONNECTIONS:          FHPA      FHPB      FHPC
SESSION              FHPA      FHPB      FHPC

CONNECTION(FHPA)          GROUP(BEAHP10)
DESCRIPTION(1ST HP SNAP2+ CONNECTION)
CONNECTION-IDENTIFIERS
  NETNAME(LUHP10A)          INDSYS( )
REMOTE-ATTRIBUTES
  REMOTESYSTEM( )  REMOTENAME( )  REMOTESYSNET( )
CONNECTION-PROPERTIES
  ACCESSMETHOD(VTAM)          PROTOCOL(APPC)  CONNNTYPE( )
  SINGLESESS(NO)              DATASTREAM(USER)  RECORDFORMAT(U)
  QUEUELIMIT(NO)              MAXQTIME(NO)
OPERATIONAL-PROPERTIES
  AUTOCONNECT(NO)              INSERVICE(YES)
SECURITY
  SECURITYNAME( )              ATTACHSEC(LOCAL)  BINDSECURITY(NO)
```

```

        USEDFLTUSER(NO)
    RECOVERY
        PSRECOVERY(SYSDEFAULT)

```

```

CONNECTION(FHPB)          GROUP(BEAHP10)
DESCRIPTION(2ND HP SNA+ 2 CONNECTION)
CONNECTION-IDENTIFIERS
    NETNAME(LUHP10B)          INDSYS( )
REMOTE-ATTRIBUTES
    REMOTESYSTEM( )          REMOTENAME( )    REMOTESYSNET( )
CONNECTION-PROPERTIES
    ACCESSMETHOD(VTAM)        PROTOCOL(APPC)    CONNNTYPE( )
    SINGLESESS(NO)            DATASTREAM(USER) RECORDFORMAT(U)
    QUEUELIMIT(NO)            MAXQTIME(NO)
OPERATIONAL-PROPERTIES
    AUTOCONNECT(NO)          INSERVICE(YES)
SECURITY
    SECURITYNAME( )          ATTACHSEC(LOCAL) BINDSECURITY(NO)
    USEDFLTUSER(NO)
RECOVERY
    PSRECOVERY(SYSDEFAULT)

```

```

CONNECTION(FHPC)          GROUP(BEAHP10)
DESCRIPTION(3RD HP SNA+ 2 CONNECTION)
CONNECTION-IDENTIFIERS
    NETNAME(LUHP10C)          INDSYS( )
REMOTE-ATTRIBUTES
    REMOTESYSTEM( )          REMOTENAME( )    REMOTESYSNET( )
CONNECTION-PROPERTIES
    ACCESSMETHOD(VTAM)        PROTOCOL(APPC)    CONNNTYPE( )
    SINGLESESS(NO)            DATASTREAM(USER) RECORDFORMAT(U)
    QUEUELIMIT(NO)            MAXQTIME(NO)
OPERATIONAL-PROPERTIES
    AUTOCONNECT(NO)          INSERVICE(YES)
SECURITY
    SECURITYNAME( )          ATTACHSEC(LOCAL) BINDSECURITY(NO)
    USEDFLTUSER(NO)
RECOVERY
    PSRECOVERY(SYSDEFAULT)

```

```

SESSIONS(FHPA)           GROUP(BEAHP10)
DESCRIPTION(1ST HP SNAP2+ SESSION)
SESSION-IDENTIFIERS
    CONNECTION(FHPA)          SESSNAME( )    NETNAMEQ( )
    MODENAME(SMSNA100)
SESSION-PROPERTIES
    PROTOCOL(APPC)            MAXIMUM(32,16) RECEIVEPFX( )
    RECEIVECOUNT( )          SENDPFX( )    SENDCOUNT( )
    SENDSIZE(4096)            RECEIVESIZE(4096) SESSPRIORITY(0)

```

## 2 Sample Configurations and JCL

---

```
PRESET-SECURITY
  USERID( )
OPERATIONAL-PROPERTIES
  AUTOCONNECT(YES)          BUILDCHAIN(YES)    USERAREALEN(0)
  IOAREALEN(0,0)            RELREQ(NO)         DISCREQ(NO)
  NEPCLASS(0)
RECOVERY
  RECOVOPTION(SYSDEFAULT)

SESSIONS(FHPB)              GROUP(BEAHP10)
                             DESCRIPTION(2ND HP SNAP2+ SESSION)
SESSION-IDENTIFIERS
  CONNECTION(FHPB)          SESSNAME( )        NETNAMEQ( )
  MODENAME(SMSNA100)
SESSION-PROPERTIES
  PROTOCOL(APPC)            MAXIMUM(32,16)    RECEIVEPFX( )
  RECEIVECOUNT( )          SENDPFX( )        SENDCOUNT( )
  SENDSIZE(4096)            RECEIVESIZE(4096)  SESSPRIORITY(0)
PRESET-SECURITY
  USERID( )
OPERATIONAL-PROPERTIES
  AUTOCONNECT(YES)          BUILDCHAIN(YES)    USERAREALEN(0)
  IOAREALEN(0,0)            RELREQ(NO)         DISCREQ(NO)
  NEPCLASS(0)
RECOVERY
  RECOVOPTION(SYSDEFAULT)

SESSIONS(FHPC)              GROUP(BEAHP10)
                             DESCRIPTION(3RD HPSNAP2+ SESSION)
SESSION-IDENTIFIERS
  CONNECTION(FHPC)          SESSNAME( )        NETNAMEQ( )
  MODENAME(SMSNA100)
SESSION-PROPERTIES
  PROTOCOL(APPC)            MAXIMUM(10,5)    RECEIVEPFX( )
  RECEIVECOUNT( )          SENDPFX( )        SENDCOUNT( )
  SENDSIZE(4096)            RECEIVESIZE(4096)  SESSPRIORITY(0)
PRESET-SECURITY
  USERID( )
OPERATIONAL-PROPERTIES
  AUTOCONNECT(YES)          BUILDCHAIN(YES)    USERAREALEN(0)
  IOAREALEN(0,0)            RELREQ(NO)         DISCREQ(NO)
  NEPCLASS(0)
RECOVERY
  RECOVOPTION(SYSDEFAULT)
```

---

# Sample JCL for an MVS Platform

Job Control Language (JCL) is used on an OS/390 Multiple Virtual Storage (MVS) platform to set the environment and invoke SNACRM commands. The following sections provide samples of JCL that may be used for your MVS operating system. The sample SET commands may not reflect the configuration of your system. You must customize the SET commands for your environment. Refer to your System Administrator for more information about your particular set up. Refer to [Appendix A, “Command Reference Pages,”](#) for more information about the SNACRM, CRMLKON, CRMLKOFF, CRMDOWN, and CRMLOGS commands.

## Sample JCL for the SNACRM Command

Following is an explanation of the SET commands and an example of JCL that can be used when you run the SNACRM command.

SET SNACMD	Sets the SNACRM command line parameters. Refer to <a href="#">SNACRM in Appendix A, “Command Reference Pages”</a> for more information about the command line parameters.
SET OBJLIB	Indicates the name of the PDSE library where the SNACRM executable is installed.
SET DATA1	Indicates the name of the PDS library where the SNACRM required parameter file FMB was installed.
SET DATA2	Indicates the dataset containing the ENVFILE.
SET ENVFILE	Indicates the name of the PDS member that contains the environment variables for the SNACRM. A sample member, ENV, is delivered with your product.
SET SIZE	Defines the region size for the running SNACRM task. It is suggested that the size be set to 0M to allow the SNACRM to start up and level out to the size it requires.
SET ENV	Indicates the ENVFILE DD name.
SET CEE	Specifies the high-level qualifier for the LE runtime library. CEE should be set to the prefix of the IBM Language Environment data sets. Language Environment is required to run the SNACRM.

## 2 Sample Configurations and JCL

---

SET CBC Specifies the high-level qualifier for the C/C++ runtime library. CBC should be set to the prefix of the IBM C++ data sets.

### Listing 2-9 Sample JCL for SNACRM Command

---

```
//*****
**
/** USE THE SET STATEMENTS TO SET THE APPROPRIATE VALUES      *
/** SNACMD IS THE SNACRM COMMAND LINE                          *
/** DATA1 IS THE DATA SET CONTAINING FMB                      *
/** DATA2 IS THE DATA SET CONTAINING ENVFILE                  *
/** ENVFILE IS THE FILE CONTAINING ENVIRONMENT VARIABLES      *
/** ENV SETS THE ENVFILE DD NAME                               *
/** RUNOPTS SETS ANY DESIRED LE RUNTIME OPTIONS                *
/** SIZE SETS THE REGION SIZE FOR THE SNACRM PROCESS           *
/** OBJLIB IS THE LOAD LIBRARY CONTAINING THE JAM PROGRAM OBJECTS*
/** CEE IS THE HLQ FOR THE LE RUNTIME LIBRARY                  *
/** CBC IS THE HLQ FOR THE C/C++ RUNTIME LIBRARY               *
//*****
**
// SET SNACMD=' '//<address>:<port>" <group>'
// SET OBJLIB=
// SET DATA1=
// SET DATA2=
// SET ENVFILE=ENV
// SET SIZE=0M
// SET ENV='ENVAR("_CEE_ENVFILE=DD:ENV")'
// SET CEE=CEE,CBC=CBC
// SNACRM EXEC PGM=SNACRM,REGION=&SIZE,
// PARM='POSIX(ON) &ENV &RUNOPTS/&SNACMD'
// STEPLIB DD DSN=&OBJLIB,DISP=SHR
//          DD DSN=&CEE..SCEERUN,DISP=SHR
//          DD DSN=&CBC..SCLBDLL,DISP=SHR
//FLDTBL DD DSN=&DATA1,DISP=SHR
//ENV DD DSN=&DATA2(&ENVFILE),DISP=SHR
//
```

---

## Sample JCL for the CRMLKON Command

Following is an explanation of the SET commands and an example of JCL that can be used when you run the CRMLKON command.

SET LNKCMD	Sets the CRMLKON command line parameters. Refer to <a href="#">crmlkon</a> in <a href="#">Appendix A, “Command Reference Pages”</a> for more information about the command line parameters.
SET OBJLIB	Indicates the name of the PDSE library where the CRMLKON executable is installed.
SET DATA1	Indicates the name of the PDS library where the CRMLKON required parameter file FMB was installed.
SET DATA2	Indicates the dataset containing the ENVFILE.
SET ENVFILE	Indicates the name of the PDS member that contains the environment variables for the CRMLKON. A sample member, ENV, is delivered with your product.
SET SIZE	Defines the region size for the running CRMLKON task.
SET ENV	Indicates the ENVFILE DD name.
SET CEE	Specifies the high-level qualifier for the LE runtime library. CEE should be set to the prefix of the IBM Language Environment data sets. Language Environment is required to run CRMLKON.
SET CBC	Specifies the high-level qualifier for the C/C++ runtime library. CBC should be set to the prefix of the IBM C++ data sets.

### Listing 2-10 Sample JCL for CRMLKON Command

```

//*****
**
//* LINKCMD INDICATES THE DISTRIBUTED SNACRM ADDRESS AND LINKNAME*
//* RUNOPTS SETS ANY DESIRED LE RUNTIME OPTIONS                      *
//* SIZE SETS THE REGION SIZE FOR THE SNACRM PROCESS                *
//* OBJLIB IS THE LOAD LIBRARY CONTAINING THE JAM PROGRAM OBJECTS*
//* CEE IS THE HLQ FOR THE LE RUNTIME LIBRARY                      *
//* CBC IS THE HLQ FOR THE C/C++ RUNTIME LIBRARY                  *
//*****
**
// SET LINKCMD='-n'//<host name>:<port>" <linkname>'
// SET OBJLIB=
// SET RUNOPTS=
// SET DATA1=
// SET DATA2=
// SET SIZE=1M
// SET ENV='ENVAR( "_CEE_ENVFILE=DD:ENV" )'
// SET CEE=CEE,CBC=CBC

```

```
//CRMLKON EXEC PGM=CRMLKON,REGION=&SIZE,  
// PARM=' POSIX(ON) &ENV &RUNOPTS/&LINKCMD '  
//STEPLIB DD DSN=&OBJLIB,DISP=SHR  
// DD DSN=&CEE..SCEERUN,DISP=SHR  
// DD DSN=&CBC..SCLBDLL,DISP=SHR  
//FLDTBL DD DSN=&DATA1,DISP=SHR
```

---

## Sample JCL for the CRMLKOFF Command

Following is an explanation of the SET commands and an example of JCL that can be used when you run the CRMLKOFF command.

- |             |   |
|-------------|---|
| SET LNKCMD  | Sets the CRMLKOFF command line parameters. Refer to <a href="#">crmlkoff</a> in <a href="#">Appendix A, “Command Reference Pages”</a> for more information about the command line parameters. |
| SET OBJLIB  | Indicates the name of the PDSE library where the CRMLKOFF executable is installed.  |
| SET DATA1   | Indicates the name of the PDS library where the CRMLKOFF required parameter file FMB was installed.   |
| SET DATA2   | Indicates the dataset containing the ENVFILE.   |
| SET ENVFILE | Indicates the name of the PDS member that contains the environment variables for the CRMLKOFF. A sample member, ENV, is delivered with your product.  |
| SET SIZE    | Defines the region size for the running CRMLKOFF task.  |
| SET ENV     | Indicates the ENVFILE DD name.  |
| SET CEE     | Specifies the high-level qualifier for the LE runtime library. CEE should be set to the prefix of the IBM Language Environment data sets. Language Environment is required to run CRMLKOFF.   |
| SET CBC     | Specifies the high-level qualifier for the C/C++ runtime library. CBC should be set to the prefix of the IBM C++ data sets.   |

---

### Listing 2-11 Sample JCL for CRMLKOFF Command

---

```
//*****  
**
```



---

```

/* LINKCMD INDICATES THE DISTRIBUTED SNACRM ADDRESS AND LINKNAME*
/* RUNOPTS SETS ANY DESIRED LE RUNTIME OPTIONS *
/* SIZE SETS THE REGION SIZE FOR THE SNACRM PROCESS *
/* OBJLIB IS THE LOAD LIBRARY CONTAINING THE JAM PROGRAM OBJECTS*
/* CEE IS THE HLQ FOR THE LE RUNTIME LIBRARY *
/* CBC IS THE HLQ FOR THE C/C++ RUNTIME LIBRARY *
/*****
**
// SET LINKCMD='-n'//<host name>:<port>" <linkname>'
// SET OBJLIB=
// SET RUNOPTS=
// SET DATA1=
// SET DATA2=
// SET SIZE=1M
// SET ENV='ENVAR( "_CEE_ENVFILE=DD:ENV" )'
// SET CEE=CEE,CBC=CBC
//CRMLKOFF EXEC PGM=CRMLKOFF,REGION=&SIZE,
// PARM='POSIX(ON) &ENV &RUNOPTS/&LINKCMD'
//STEPLIB DD DSN=OBJLIB,DISP=SHR
// DD DSN=&CEE..SCEERUN,DISP=SHR
// DD DSN=&CBC..SCLBDLL,DISP=SHR
//FLDTBL DD DSN=&DATA1,DISP=SHR

```

---

## Sample JCL for the CRMDOWN Command

Following is an explanation of the SET commands and an example of JCL that can be used when you run the CRMDOWN command.

SET LNKCMD	Sets the CRMDOWN command line parameters. Refer to CRMDOWN in <a href="#">Appendix A, “Command Reference Pages”</a> for more information about the command line parameters.
SET OBJLIB	Indicates the name of the PDSE library where the CRMDOWN executable is installed.
SET DATA1	Indicates the name of the PDS library where the CRMDOWN required parameter file FMB was installed.
SET DATA2	Indicates the dataset containing the ENVFILE.
SET ENVFILE	Indicates the name of the PDS member that contains the environment variables for the CRMDOWN. A sample member, ENV, is delivered with your product.
SET SIZE	Defines the region size for the running CRMDOWN task.

## 2 Sample Configurations and JCL

---

- SET ENV     Indicates the ENVFILE DD name.
- SET CEE     Specifies the high-level qualifier for the LE runtime library. CEE should be set to the prefix of the IBM Language Environment data sets. Language Environment is required to run CRMDOWN.
- SET CBC     Specifies the high-level qualifier for the C/C++ runtime library. CBC should be set to the prefix of the IBM C++ data sets.

### Listing 2-12 Sample JCL for CRMDOWN Command

---

```
// *****
**
//*  LINKCMD INDICATES THE DISTRIBUTED SNACRM ADDRESS          *
//*  RUNOPTS SETS ANY DESIRED LE RUNTIME OPTIONS              *
//*  SIZE SETS THE REGION SIZE FOR THE SNACRM PROCESS         *
//*  OBJLIB IS THE LOAD LIBRARY CONTAINING THE JAM PROGRAM OBJECTS*
//*  CEE IS THE HLQ FOR THE LE RUNTIME LIBRARY                *
//*  CBC IS THE HLQ FOR THE C/C++ RUNTIME LIBRARY             *
// *****
**
//  SET LINKCMD=' -n'//<host name>:<port>' '
//  SET OBJLIB=
//  SET RUNOPTS=
//  SET DATA1=
//  SET DATA2=
//  SET SIZE=1M
//  SET ENV=' ENVAR( "_CEE_ENVFILE=DD:ENV" ) '
//  SET CEE=CEE,CBC=CBC
//CRMDOWN EXEC PGM=CRMDOWN,REGION=&SIZE,
//  PARM=' POSIX(ON) &ENV &RUNOPTS/&LINKCMD'
//STEPLIB DD DSN=&OBJLIB,DISP=SHR
//          DD DSN=&CEE..SCEERUN,DISP=SHR
//          DD DSN=&CBC..SCLBDLL,DISP=SHR
//FLDTBL DD DSN=&DATA1,DISP=SHR
```

---

## Sample JCL for the CRMLOGS Command

Following is an explanation of the SET commands and an example of JCL that can be used when you run the CRMLOGS command.

SET LNKCMD	Sets the CRMLLOGS command line parameters. Refer to <a href="#">CRMLLOGS</a> in <a href="#">Appendix A, “Command Reference Pages”</a> for more information about the command line parameters.
SET OBJLIB	Indicates the name of the PDSE library where the CRMLLOGS executable is installed.
SET DATA1	Indicates the name of the PDS library where the CRMLLOGS required parameter file FMB was installed.
SET DATA2	Indicates the dataset containing the ENVFILE.
SET ENVFILE	Indicates the name of the PDS member that contains the environment variables for the CRMLLOGS. A sample member, ENV, is delivered with your product.
SET SIZE	Defines the region size for the running CRMLLOGS task.
SET ENV	Indicates the ENVFILE DD name.
SET CEE	Specifies the high-level qualifier for the LE runtime library. CEE should be set to the prefix of the IBM Language Environment data sets. Language Environment is required to run CRMLLOGS.
SET CBC	Specifies the high-level qualifier for the C/C++ runtime library. CBC should be set to the prefix of the IBM C++ data sets.

### Listing 2-13 Sample JCL for CRMLLOGS Command

```
//*****
**
/** THIS JOB IS USED TO CHECK THE RECOVERY LOGS FOR          *
/** OUTSTANDING TRANSACTION DATA. SEE USER GUIDE FOR MORE INFO *
/**
/** @(#) $Id: crmllogs.jcl,v 1.3 2000/05/17 16:20:27 crount Exp $
/** Copyright (c)2000 BEA Systems, Inc., all rights reserved.  *
/*******
**
/** SNAGRP IS THE GROUP NAME OF THE DISTRIBUTED SNACRM          *
/** DATA1 IS THE DATA SET CONTAINING FMBDEF                  *
/** DATA2 IS THE DATA SET CONTAINING ENVFILE                  *
/** ENVFILE IS THE FILE CONTAINING ENVIRONMENT VARIABLES       *
/** ENV SETS THE ENVFILE DD NAME                                *
/** RUNOPTS SETS ANY DESIRED LE RUNTIME OPTIONS                *
/** SIZE SETS THE REGION SIZE FOR THE SNACRM PROCESS           *
/** OBJLIB IS THE LOAD LIBRARY CONTAINING THE EAM PROGRAM OBJECTS*
/** CEE IS THE HLQ FOR THE LE RUNTIME LIBRARY                  *
```

## 2 Sample Configurations and JCL

---

```
//*   CBC IS THE HLQ FOR THE C/C++ RUNTIME LIBRARY                               *
//*****
**
//   SET SNACMD=<group>
//   SET OBJLIB=
//   SET DATA1=
//   SET DATA2=
//   SET RUNOPTS=
//   SET SIZE=10M
//   SET ENV='ENVAR( "_CEE_ENVFILE=DD:ENV" ) '
//   SET CEE=CEE,CBC=CBC
//CRMLOGS EXEC PGM=CRMLOGS,REGION=&SIZE,
//   PARM='POSIX(ON) &ENV &RUNOPTS/&SNAGRP'
//STEPLIB DD DSN=&OBJLIB,DISP=SHR
//          DD DSN=&CEE..SCEERUN,DISP=SHR
//          DD DSN=&CBC..SCLBDLL,DISP=SHR
//MSGFILE DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//FLDTBL DD DSN=&DATA1,DISP=SHR
//ENV DD DSN=&DATA2(ENV),DISP=SHR
//
```

---

# 3 Administering the SNA Components

The topics in this section cover activities an administrator performs with the System Network Architecture Communications Resource Manager (SNACRM) to maintain BEA WebLogic Java Adapter for Mainframe (JAM) applications.

The interface to the stack administration and configuration is dependent on the stack provider and thus cannot be covered in this guide. Refer to vendor publications for the stack(s) used in your environment.

This section discusses the following topics:

- [Starting the SNACRM on Unix or Windows NT](#)
- [Starting the SNACRM on OS/390 MVS](#)
- [Using the SNACRM Monitor](#)
- [Activating and De-Activating Links](#)
- [Stopping the SNACRM on OS/390 Unix](#)

# Starting the SNACRM on Unix or Windows NT

The SNACRM is a server that communicates directly with the PU 2.1 server to provide SNA connectivity. These servers can be started manually. The PU 2.1 server must always be started before the SNACRM. Both servers must be started before starting the associated SNA domain gateway. Before you start the SNACRM or the PU 2.1 server, you must set several environment variables.

## Starting the PU2.1 Server

Refer to the operational documentation provided by your SNA stack vendor for information about starting the PU2.1 server. The SNA stack must be running and active before you start the SNACRM.

## Starting the SNACRM

### To start the SNACRM on Unix:

- Enter the SNACRM command on the command line. Refer to “[Command Reference Pages](#)” for more detailed information about the SNACRM command.

When you start the SNACRM from the Unix command line, the SNACRM Command Line Console puts its prompt in a window, and if exited, shuts down all of the active links.

### To start the SNACRM from a Command Prompt Window on Windows NT:

- Open a Command Prompt window.
- Navigate to the drive where the SNACRM executable is installed.
- Enter SNACRM to start SNACRM.exe.

On Windows NT, the SNACRM can be started in its own window, however, no prompt is displayed and no console commands are available. Refer to “[Command Reference Pages](#),” for more detailed information about the SNACRM command.

## Starting the SNACRM on OS/390 MVS

## Setting Environment Variables

The following environment variables must be set in the environment where the SNACRM is started. A sample file is delivered (ENV)

```
FLDTBLDIR32=JAM Installation Directory/lib  
FIELDTBLS32=JAM Installation Directory/fmb.def  
APPDIR=<wherever>
```

These environment variables can be added as ‘set’ or ‘export’ commands in a script file used to start a SNACRM. Or they can be added to the `startweblogic` script for use when a SNACRM is spawned on startup.

The OS/390 MVS platform sets the environment and invokes the SNACRM through Job Control Language (JCL).

- Run a SNACRM job using JCL written for your system.

Refer to [“Sample JCL for the SNACRM Command”](#) and [“Command Reference Pages,”](#) for more information about sample SNACRM JCL and the SNACRM command.

**Note:** If the SNACRM is installed on an OS/390 MVS platform, it does not have to be restarted if the JAM gateway goes down. Only the OS/390 MVS version and the OS/390 Unix version of the SNACRM have this persistent feature.

## Using the SNACRM Monitor

You can use the SNACRM monitor to set trace levels for a selected SNACRM and the associated APPC stacks. You also can observe link activity and display trace status, link status, and link statistics.

**Notes:** The SNACRM monitor does not show trace data. This data is captured in a file under the APPDIR directory (where APPDIR is the variable name associated with the application directory). Please contact BEA Customer Support for help in locating the trace file(s) and interpreting them.

The JAM software includes a `jsnacrm` utility that launches and executes the SNACRM monitor. The `jsnacrm` utility is designed for Java platforms and supplies both a Java-based application and an applet.

**Note:** The `jsnacrm` utility cannot be run from an OS/390 mainframe platform. It must be run on a Unix or Windows NT platform to connect to an OS/390 SNACRM.

The following discussion relates to the Java-based SNACRM monitor only.

The BEA eLink Java Adapter for Mainframe software CDROM contains the following files associated with the `jsnacrm` utility:

- `bealogo.gif`
- `jsnacrm.html`
- `jsnacrm.jar`



- `moncrm.jar`
- `moncrm.x509`

## Prerequisite for Running the JSNACRM Utility on an NT Platform

The `jsnacrm` utility is written in Java as both an application and an applet. The application launches and executes like any other Java application and can be set up so it is accessible from the Windows desktop. The applet launches and executes from a network browser.

## Running the Java Applet Version

The following sections describe how to set up and run the Java applet version of the `jsnacrm` utility.

### Prerequisites for Running the Java Applet Version

You must have either Netscape Communicator 4.x or Internet Explorer 4.x or 5.x installed on your NT Windows system. You also must have the Java plug-in installed on your system. You can download this plug-in from the following internet location:

<http://java.sun.com/products/plugin>

**Note:** If the Java plug-in is not already installed on your system, when you attempt to open the `jsnacrm.html` file, the program prompts you for an automatic download of the plug-in by the browser.

Next, you must set up your system to accept code signed by the identity `moncrm`. To do this, perform the following steps:

1. Create the identity `moncrm` in your JDK 1.1 identity database. By entering the parameter `true`, you establish `moncrm` to be a trusted identity.

```
javakey -c moncrm true
```

2. Import the `moncrm` certificate into your identity database. To associate the certificate with the identity, use the nickname `moncrm` as the first argument to the `javakey` command.

```
javakey -ic moncrm %TUXDIR%\bin\moncrm.x509
```

## Starting the Java Applet

To start the Java applet in an existing browser, open the file:

```
<tuxedo-path>\bin\jsnacrm.html
```

To build a shortcut to start the Java applet using a separate instance of your network browser, enter the following command:

```
<browser-pathname> %TUXDIR%\bin\jsnacrm.html
```

## Running the Java Applet Version

Set up your applet version to monitor either a local or remote SNACRM. To do this, you make selections on the Java Plug-in Properties control panel. This control panel is automatically downloaded with the plug-in and is initiated from the Windows Start Programs pop-up menu. Refer to online documentation about the control panel at the following Internet location:

```
http://java.sun.com/products/plugin/1.1.1/docs
```

When the Monitor screen displays ([Figure 3-1](#)), enter the address of the SNACRM that you want to monitor in the field at the top of the screen.

To monitor a local SNACRM, select Applet Host from the Network Access drop-down menu. Type the following in the Enter SNACRM Address panel:

```
//localhost:port
```

where:

`localhost`

Explicitly specifies the local host.

`port`

Specifies the port number of the SNACRM on the local host.

To monitor a remote SNACRM, select Unrestricted from the Network Access drop down menu. Type the following in the Enter SNACRM Address panel:

```
//remotehostname:address
```

where:

`remotehostname`

Specifies the remote host.

`address`

Specifies the network address of the SNACRM on the remote host.

The GUI contains two screen areas that require user entry and four screen areas that display information about the SNACRM being monitored. Status messages are displayed at the bottom of the screen. The GUI screen functions are listed in [Table 3-1](#) and shown in [Figure 3-1](#).

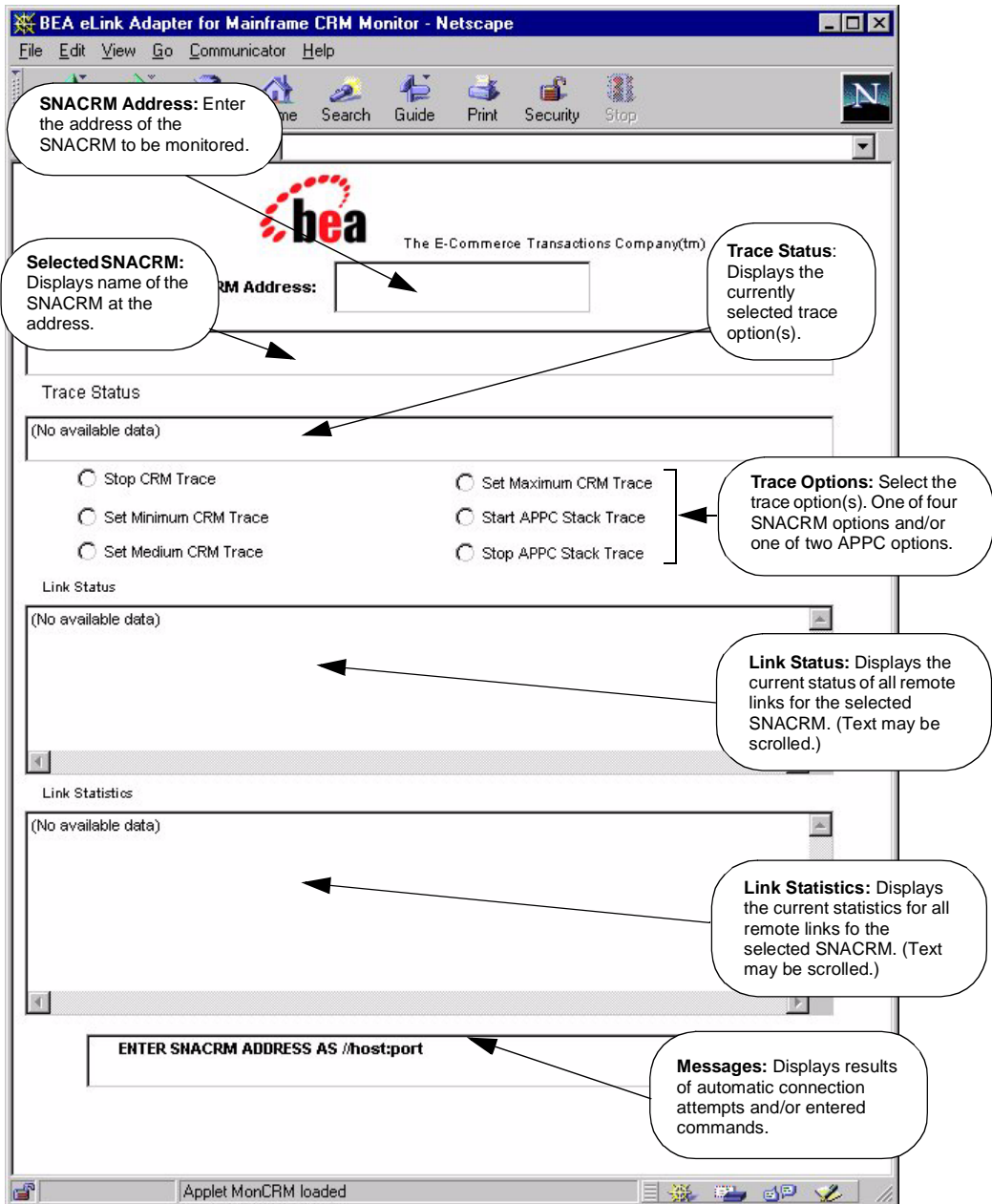
**Table 3-1 SNACRM Monitor Screen Functions**

Display Section	Function
SNACRM Address	This is where you enter the address of the SNACRM to be monitored.
Selected SNACRM	Displays the name of the SNACRM at the address entered in the address field.
Trace Status	Displays the currently selected trace options.
Trace Options	<ul style="list-style-type: none"> <li>■ <b>Stop CRM Trace</b> disables SNACRM tracing and closes the trace file, if it exists.</li> <li>■ <b>Set Minimum CRM Trace</b> establishes tracing of only major events. This level is sufficient only to determine the sequence of application conversations.</li> <li>■ <b>Set Medium CRM Trace</b> establishes minimum tracing plus tracing of all I/O buffers.</li> <li>■ <b>Set Maximum CRM Trace</b> establishes medium tracing plus tracing of all APPC verbs.</li> <li>■ <b>Start APPC Stack Trace</b> establishes tracing of the APPC stack. Generally shows the parameters and results of all API calls. Depending on the stack being used, other options such as vendor-specified environment variables also may have to be activated. May be selected along with any of the previous Trace Options.</li> <li>■ <b>Stop APPC Stack Trace</b> disables APPC stack trace, if established.</li> </ul>

**Table 3-1 SNACRM Monitor Screen Functions**

<b>Display Section</b>	<b>Function</b>
Link Status	Displays the current status of all remote links for the selected SNACRM. (Text may be scrolled.)
Link Statistics	Displays the current statistics for all remote links for the selected SNACRM. (Text may be scrolled.)
Message Line	Displays messages showing either the results of automatic connection attempts or commands issued to change the trace options.

Figure 3-1 The SNACRM Monitor Running as an Applet on a Network Browser



## Running the Java Application Version

The Java application version displays and operates identically to the applet version. Refer to screen definitions and functions discussed under “[Running the Java Applet Version](#).”

To build a shortcut for starting the Java application version, perform the following steps:

1. Enter the command:

```
jrew -classpath %ClassPath%;jsnacrm.jar jsnacrm
```

2. Start the application in the directory %TUXDIR%\bin so it can find its files.

To run from a command window, perform the following steps:

1. Change directory to %TUXDIR%\bin.

2. Enter the command:

```
jrew -classpath %ClassPath%;jsnacrm.jar jsnacrm
```

## Activating and De-Activating Links

The JAM gateway software provides a command line tool you can use to activate and de-activate links that have been defined in the `JM_SNALINKS` section of the `jcrmgw.cfg` file. This tool consists of two commands and their associated parameters: `crmlkon` and `crmlkoff`.

**Note:** If a link to a remote host is de-activated and re-activated by the host, the JAM software normally re-establishes the link automatically. If this does not occur, you can use the `crmlkon` command to re-establish the link.

## Using the `crmlkon` Command

You can start one or more SNA links with this command. Use the following syntax:

```
crmlkon -n<hostname:port> [-v -i -h] <linkname> ...
```

where:

`-n`

indicates that the parameters immediately following are the `hostname` and `port` of the machine running the SNACRM where the `linkname` is located.

`hostname:port`

specifies the IP host name and port of the machine running the SNACRM where the `linkname` is located.

`-v`

specifies that this command displays verbose output.

`-i`

causes the command to ignore errors and attempt to start all links specified on the command line.

`h`

invokes a help screen that shows the syntax usage for this command.

`linkname`

specifies the link name(s) to start.

**Notes:** There is no notification that the link(s) started with the `crmlkon` command are activated. Use the SNACRM monitor to verify a link is active. Refer to [“Using the SNACRM Monitor”](#) for more information.

Refer to [crmlkon](#) in [Appendix A, “Command Reference Pages”](#) for more information on the `crmlkon` utility.

## Using the `crmlkoff` Command

You can stop one or more SNA links with this command. Use the following syntax:

```
crmlkoff -n<hostname:port> [-v -i -h] <linkname> ...
```

where:

`-n`

indicates that the parameters immediately following are the `hostname` and port of the machine running the SNACRM where the `linkname` is located. This is required.

`hostname:port`

specifies the IP host name and port of the machine running the SNACRM where the `linkname` is located.

`-v`

specifies that this command displays verbose output.

`-i`

causes the command to ignore errors and attempt to stop all links specified on the command line.

`h`

invokes a help screen that shows the syntax usage for this command.

`linkname`

specifies the link name(s) to start.

**Notes:** There is no notification that the link(s) stopped with the `crmlkoff` command are de-activated. Use the SNACRM monitor to verify a link is not active. Refer to [“Using the SNACRM Monitor”](#) for more information.

Refer to `crmlkoff` in [Appendix A, “Command Reference Pages”](#) for more information on the `crmlkoff` utility.

## Stopping the SNACRM on OS/390 Unix

If the SNACRM is installed on an OS/390 platform, it does not have to be stopped and restarted if the JAM gateway goes down. Only the OS/390 version of the SNACRM has this persistent feature.

### To stop the SNACRM on Unix:

- Enter the `crmdown` command on the command line. Refer to `crmdown` in [Appendix A, “Command Reference Pages”](#) for more detailed information about the `crmdown` command.



When you stop the SNACRM from the Unix command line, the SNACRM Command Line Console shuts down all of the active links.

## Stopping the SNACRM on OS/390 MVS

The OS/390 MVS platform stops the SNACRM through Job Control Language (JCL). When you are running the SNACRM on an OS/390 operating system, the persistent SNACRM feature allows the SNACRM to continue running, even if the JAM Gateway fails. You must explicitly shutdown the SNACRM if you need to shut it down. Run a `crmdown` job using JCL written for your system.

Refer to [“Sample JCL for the CRMDOWN Command”](#) and [“Command Reference Pages,”](#) for more information about sample `crmdown` JCL and the `crmdown` command.

Refer to `crmdown` in [Appendix A, “Command Reference Pages”](#) for more information on the `crmdown` utility.



---

# A Command Reference Pages

This section discusses the following commands:

- `CRMLOGS`
- `crmlkon`
- `crmlkoff`
- `crmdown`
- `SNACRM`

## CRMLOGS

Displays the content and state of the SNA Communications Resource Manager (SNACRM) log files in the current directory.

Synopsis	<code>CRMLOGS &lt;group&gt; [&lt;crm name&gt;]</code>
Description	You can use the <code>CRMLOGS</code> command to display the contents and state of the two <code>SNACRM</code> log files. <code>RSTRTLOG</code> is the transaction state log used during the recovery process and the <code>BLOBLOG</code> log stores session and link information. Deleting the log files require a cold start for each link involved.

`CRMLOGS` requires the following parameters:

	group	SNA domain group name (required)
	crm name	SNACRM name (default SNACRM)
Diagnostics	CRMLOGS exits with a return code of 0 upon successful completion.	
Examples	To display the RSTRTLOG log file for group2, type:  CRMLOGS GROUP2 SNACRM.GROUP2.RSTRTLOG  To display the BLOBLOG log file for group1, type:  CRMLOGS GROUP1 SNACRM.GROUP1.BLOBLOG	
See Also	SNACRM	

# crmlkon

Start one or more named SNACRM links.

## Synopsis

```
crmlkon -n<hostname:port> [-v -i -h -u<keyfile>] <linkname> ...
```

## Description

`crmlkon` starts all of the SNACRM links named on the command line. This is useful if one or more individual links failed to start when the SNACRM server booted. It can be used from any machine located on the same TCP/IP network as the machine running the SNACRM server. It can be used in a script and will return zero if the command could be sent to the target SNACRM. It will return one if the command could not be sent to the target SNACRM.

**Note:** The `crmlkon` utility cannot be executed on an OS/390 platform.

`-n`  
Names the machine and port running the SNACRM server.

- v Specifies verbose. Normally the command will not produce any messages, facilitating use in a script.
- i Ignores errors. When specifying multiple links, any error encountered when issuing SNACRM commands will cause crmlkon to stop processing links and return. Errors can be ignored for individual links, and processing continued with the next named link.
- u Establishes that process authentication is in effect for communications between this process and the SNACRM.
- <keyfile> Is the location of a file containing a hash key known to both this process and the SNACRM. The file contains a single line specifying a unique hash key (limited to eight characters). The file should be protected.
- <linkname> Names the link to be started. This is the SNALINKS entry in the gateway configurations that defines this link. Multiple link names can be specified.

## Example

To start links `link2` and `cicstest` owned by the SNACRM running on `mach1` at port 5000:

```
crmlkon -n mach1:5000 link2 cicstest
```

## Diagnostics

`crmlkon` only checks the syntax of the command. Separate facilities, either `jsnacrm` or mainframe based facilities must be used to determine if the link actually became active. If the command could not be successfully sent to the SNACRM `crmlkon` prints an error message, if in verbose mode, and exits with error code 1. Upon successful completion, `crmlkon` exits with exit code 0.

# crmlkoff

Stop one or more named SNACRM links.

## Synopsis

```
crmlkoff -n<hostname:port> [-v -i -h -u<keyfile>] <linkname> ...
```

## Description

`crmlkoff` stops all of the SNACRM links named on the command line. This is useful if one or more individual links need to be stopped after the SNACRM server booted. It can be used from any machine located on the same TCP/IP network as the machine running the SNACRM server. It can be used in a script and will return zero if the command could be sent to the target SNACRM. It will return one if the command could not be sent to the target SNACRM.

`-n`

Names the machine and port running the SNACRM server.

`-v`

Specifies verbose. Normally the command will not produce any messages, facilitating use in a script.

`-i`

Ignores errors. When specifying multiple links, any error encountered when issuing SNACRM commands will cause `crmlkoff` to stop processing links and return. Errors can be ignored for individual links, and processing continued with the next named link.

`-u`

Establishes that process authentication is in effect for communications between this process and the SNACRM.

`<keyfile>`

Is the location of a file containing a hash key known to both this process and the SNACRM. The file contains a single line specifying a unique hash key (limited to eight characters). The file should be protected.

<linkname>

Names the link to be stopped. This is the \*DM\_SNALINKS entry in the DMCONFIG which defines this link. Multiple link names can be specified.

## Example

To stop links `link1` and `cicstest` owned by the SNACRM running on `mach` at port 5000:

```
crmlkoff -n mach:5000 link1 cicstest
```

## Diagnostics

`crmlkoff` only checks the syntax of the command. Separate facilities, either `jsnacrm` or mainframe based facilities must be used to determine if the link actually became inactive. If the command could not be successfully sent to the SNACRM `crmlkoff` prints an error message, if in verbose mode, and exits with error code 1. Upon successful completion, `crmlkoff` exits with exit code 0.

# crmdown

Shut down a SNACRM.

## Synopsis

```
crmdown -n<hostname:port> [-v -i -h -u<keyfile>]
```

## Description

`crmdown` shuts down the SNACRM specified on the command line. It can be used from any machine located on the same TCP/IP network as the machine running the SNACRM server. It can be used in a script and will return zero if the command could be sent to the target SNACRM. It will return one if the command could not be sent to the target SNACRM.

<code>-n</code>	Names the machine and port running the SNACRM server.
<code>-v</code>	Specifies verbose. Normally the command will not produce any messages, facilitating use in a script.
<code>-i</code>	Ignores errors.
<code>-u</code>	Establishes that process authentication is in effect for communications between this process and the SNACRM.
<code>&lt;keyfile&gt;</code>	Is the location of a file containing a hash key known to both this process and the SNACRM. The file contains a single line specifying a unique hash key (limited to eight characters). The file should be protected.

### Example

To stop the SNACRM running on `mach1` at port 5000:

```
crmdown -n mach1:5000
```

### Diagnostics

`crmdown` only checks the syntax of the command. If the command could not be successfully sent to the SNACRM `crmdown` prints an error message, if in verbose mode, and exits with error code 1. Upon successful completion, `crmdown` exits with exit code 0.

## SNACRM

Launches the SNA Communications Resource Manager.

Synopsis	<code>SNACRM [ -t 0 1 2 3 ] [-p&lt;nbr&gt;] [-s] &lt;addr&gt; &lt;group&gt;</code>
Description	When you start <code>SNACRM</code> from the command line, the SNACRM Command Line Console puts its prompt in the window and, if exited, shuts down all of the active links.



You must configure one SNACRM for each JAM gateway, as well as configure one stack for each SNACRM definition. Each stack can manage one or more SNA links.

SNACRM has two types of log files stored in `$APPPDIR`, `RSTRTLOG`, and `BLOBLOG`. `RSTRTLOG` is the transaction state log used during the recovery process, while the `BLOBLOG` log stores session and link information. Deleting the log files requires a cold start for each link involved. You can use the `CRMLOGS` command to display the contents and state of the SNACRM log files.

**Trace Options** When initiating the SNACRM from the UNIX command line, you can specify any of the following SNACRM trace levels using the `-t` option:

- 0 = No tracing. Setting this level effectively disables SNACRM tracing and closes the trace file, if there is one. If tracing is subsequently restarted, a new file is created with an incremented numerical suffix.
- 1 = Minimum tracing. At this level, SNACRM traces only major events and is sufficient only to determine the sequence of application conversations.
- 2 = Medium tracing. At this level, SNACRM also traces all I/O buffers.
- 3 = Maximum tracing. At this level, SNACRM also traces all APPC verbs.

The APPC Stack API trace is either enabled or disabled using the `-s` option of the SNACRM command (refer to “General Options”). If enabled, it generally shows the parameters and results of all API calls. Depending on the stack being used, other options (such as vendor-specified environment variables) may have to be activated for SNACRM to enable the trace.

**Performance Option** The performance option is a tunable parameter used only for the OS/390 Unix and MVS SNACRM. These SNACRM versions use threads to process a request and this parameter indicates how many threads to start for processing SNA requests.

The performance option uses the following format:

```
-p <nbr>
```

where `<nbr>` equals the number of threads to start.

The default for this value is 100 threads. This should correspond to the load of SNA requests that will be made concurrently. If the number of requests exceeds the number of threads, the request will still be executed, however, the completion time could be affected.

It is recommended that you do not exceed 200 threads. The SNACRM is tuned for a maximum of 200 threads. It is recommended that you lower the 100 threads value if you have a restriction on the number of threads that can be active in your system.

General Options    The following parameters apply to this command:

`-s`

APPC Stack API trace (default none)

**Note:** On OS/390 platforms, the General Trace Facility (GTF) is used to capture API trace records under user EID 2EA. The GTF must be active on the OS/390 platform to use the `-s` parameter.

`addr`

Socket listening address (required)

`//host:port group`

Gateway Group Name (required)

Environment Variables    You must set the following environment variables before starting the SNACRM from the UNIX command line:

■ `FIEDLTBLS32` must contain `fmb.def`.

■ `FLDTBLDIR32` must contain the path to the JAM libraries directory (for example, `JAM41/lib`).

■ `APPDIR` must be set to the application directory.

Portability    Refer to the following table for a list of operating systems and stacks that can communicate with the SNACRM. (See the list of the requirements that are needed to NACRM itself on the OS/390 platform.)

**Table A-1 Supported Host Platforms**

Vendor	Classification	Product	Version	Comments
IBM	MF/OS	OS390	1.2, 2.4, 2.5, 2.6	Y2K Certified
IBM	MF/OS	VM	2.3	Y2K Certified
IBM	MF/OS	VSE/ESA	2.3	Y2K Certified
IBM	MF/OLTP	CICS/ESA	4.1	for MVS

**Table A-1 Supported Host Platforms**

Vendor	Classification	Product	Version	Comments
IBM	MF/OLTP	CICS Transaction Server for OS/390	1.2	for MVS
			1.3	
IBM	MF/OLTP	CICS	2.3	for VSE
IBM	MF/OLTP	IMS/ESA Transaction Manager	5.1	for MVS
			6.1	
IBM	MF/TCOM	ACF/VTAM	3.4+	for MVS
IBM	MF/TCOM	ACF/NCP	4.3+	for MVS
IBM	MF/TCOM	ACF/VTAM		for VSE
IBM	MF/TCOM	ACF/NCP		for VSE
IBM	MF/OS	OS390	1.2	Y2K Certified

Interoperability    SNACRM is interactive with the following:

- CICS 4.1 through TS 1.3
- IMS 4.1 or higher
- MVS 5.22 9510 or higher
- OS/390 1.2 or higher
- VTAM for MVS/ESA, version 4.3 or higher
- SNACRM for OS/390 and MVS versions require SecureWay Communications Server, OS/390 2.8 or higher (see *BEA WebLogic Java Adapter for Mainframe Release Notes* for required IBM APAR for version 2.8)

Diagnostics    SNACRM exits with a return code of 0 upon successful completion.

Examples    Following is an example of the SNACRM command:

```
SNACRM -t 0 //myhost:5587 GROUP2 /dev/null>std.out 2>std.err &
```

When you start SNACRM from the UNIX command line, the following SNACRM Command Line Console appears:

```
$ SNACRM -t 0 //myhost:5587 GROUP2
BEA JAM Resource Manager started Thu Dec 11
18:40:49.098 1997
[SNACRM]

Console active. Enter commands
?>
da => Display active tasks
dl => Display remote links
ds => Display link statistics
dt => Display trace status
st => Start all links
sh => Stop all links and terminate
si => Terminate immediately (no quiesce)
```

To launch SNACRM with the console running in the background:

```
$ SNACRM -t 0 //myhost:5587 GROUP2 <dev/null>std.out 2>std.err &
```

To launch SNACRM with detailed tracing and APPC Stack API tracing turned on from the command line using the host/port address, type:

```
SNACRM -t 2 -s //myhost:5587
```

---

# B Error Messages

The following SNA Communications Resource Manager (SNACRM) messages can be displayed in the Tuxedo/Weblogic Enterprise (WLE) console window.

9001:ERROR	<taskname> timed out with failCode <failcode>	
	DESCRIPTION	<p>A conversation has timed out in the SNACRM with the stack return code of &lt;failcode&gt;. A timer event set to watch a conversation has expired.</p> <p>&lt;taskname&gt; may appear as:</p> <p>OB-Conversation #<i>nn</i> (&lt;linkref&gt;) tx #<i>m</i> &lt;tranname&gt;, or</p> <p>IB-Conversation #<i>nn</i> (&lt;linkref&gt;) tx #<i>m</i> &lt;tranname&gt;</p> <p>where:</p> <p><i>nn</i> is an internal APPC conversation number.</p> <p><i>m</i> is the transaction context where -1 signifies non-transactional.</p>
	ACTION	Examine <code>stderr</code> and the ULOG for additional information concerning the failure.
9002:ERROR	Server (<stackref>) Creation Failed	
	DESCRIPTION	SNACRM was unable to instantiate the stack object due to an error.
	ACTION	Check for additional messages in <code>stderr</code> . The shared library for the stack or the stack interface might not have been loaded due to an incorrect library path.

9003:ERROR	Server Failed (<stackref>), Code = <returncode>	
	DESCRIPTION	SNACRM received a bad return code from the stack start-up.
	ACTION	<p>The &lt;returncode&gt; is the value returned by the SNA Stack software. Check the status of the stack, the configuration of the stack, and the gateway configuration.</p> <p><b>Note:</b> Unless you started the SNACRM as a Tuxedo server, you must manually kill the SNACRM process.</p>
9004:ERROR	Configuration change on link <linkref> requires cold start	
	DESCRIPTION	Attempting to do a warm start after changing the domain configuration.
	ACTION	Change start type to COLD and restart.
9005:WARNING	Unrecovered transaction, ID=<txt>, blob dropped. Transaction presumed forgotten.	
	DESCRIPTION	An attempt was made by Tuxedo to recover the specified transaction that was unknown by the SNACRM. It is presumed that it was already committed or aborted prior to the recovery attempt.
	ACTION	None. This message is for information only.
9006:ERROR	Unable to start the recovery task for link <linkref>	
	DESCRIPTION	An error occurred during the warm start of Tuxedo.
	ACTION	Cold start the Tuxedo application.
9008:WARNING	Unknown tranid dropped, id=<txt>	
	DESCRIPTION	Recovery was requested by Tuxedo on a transaction that was already forgotten by the SNACRM.
	ACTION	None. This message is for information only.

9009:ERROR	No blob with recovery request. Transaction dropped, id=<tcctxt>	
	DESCRIPTION	Attempting to warm start after the SNACRM's BLOBLOG has been modified.
	ACTION	Change start type to COLD and restart.
9010:ERROR	<taskname> failed with failCode <failcode>	
	DESCRIPTION	<p>A conversation has failed with the stack return code of &lt;failcode&gt;.</p> <p>&lt;taskname&gt; may appear as:</p> <p>OB-Conversation #nn (&lt;linkref&gt;) tx #m &lt;tranname&gt;, or</p> <p>IB-Conversation #nn (&lt;linkref&gt;) tx #m &lt;tranname&gt;</p> <p>where:</p> <p>nn is an internal APPC conversation number.</p> <p>m is the transaction context where -1 signifies non-transactional.</p> <p>Possible values for the &lt;failcode&gt; are:</p> <ol style="list-style-type: none"> <li>1. Communications - unable to create the APPCserver object.</li> <li>2. MemoryAllocation - internal error allocating memory.</li> <li>3. InvalidObject - a SNACRM object could not be created or has been made invalid by some previous error.</li> <li>4. InputOutput - error occurred during file I/O or an unexpected APPC return code was received.</li> <li>5. Registration - internal task cannot be registered.</li> </ol>
	ACTION	Examine stderr and the ULOG for additional information concerning the failure. For failcode Input/Output, verify that the user starting the SNACRM process has the proper file permissions for the BLOBLOG and RSTRTLOG. If no apparent error is found, contact BEA Customer Support.

## B Error Messages

---

9011:ERROR	Attempt to connect as second master refused!	
	DESCRIPTION	A second GWSNAX is attempting to connect to the SNACRM as a master gateway. Only one master gateway is allowed.
	ACTION	Ensure that multiple Tuxedo configurations do not use the same SNACRM address.
9012:ERROR	Attempt to connect as master in autonomous mode refused!	
	DESCRIPTION	An attempt to connect to the SNACRM as a master gateway was made when the SNACRM was running in autonomous mode.
	ACTION	Ensure that multiple Tuxedo configurations do not use the same SNACRM address.
9013:ERROR	Attempt to connect with incorrect group name (<groupname>) refused!	
	DESCRIPTION	The group name in the gateway configuration file does not match the group name specified in the SNACRM command line.
	ACTION	Correct the group name that is in error and restart.
9014:ERROR	INTERNAL ERROR: memory allocation failed [for new context/data buffer]	
	DESCRIPTION	Internal error allocating memory. No more memory.
	ACTION	Contact BEA Customer Support.
9015:ERROR	INTERNAL ERROR: server registration failed	
	DESCRIPTION	Internal error registering the APPC server. APPC libraries not found. The stack failed.
	ACTION	Contact BEA Customer Support.



9016:ERROR	Link refers to undefined APPC stack (<stackref>)!	
	DESCRIPTION	The stackref in the link configuration is incorrect.
	ACTION	Correct the stackref that is in error, run dmloadcf, and restart.
9017:ERROR	INTERNAL ERROR: link registration failed	
	DESCRIPTION	Internal error registering the link. The stack failed.
	ACTION	Contact BEA Customer Support.
9018:ERROR	Invalid Transition <additional information>	
	DESCRIPTION	<p>An internal state table failure has occurred. The &lt;additional information&gt; will be one of the following:</p> <ol style="list-style-type: none"> <li>1. From &lt;oldstate&gt; to &lt;newstate&gt; for &lt;dir&gt;-bound transaction #&lt;tid&gt;</li> <li>2. To &lt;newstate&gt; for inbound transaction #&lt;tid&gt;</li> <li>3. To &lt;newstate&gt; for outbound transaction #&lt;tid&gt;</li> </ol>
	ACTION	Contact BEA Customer Support.
9019:ERROR	Unknown Service Correlator = <correlator>, message dropped	
	DESCRIPTION	Internal error assigning service correlator values. Message context lost.
	ACTION	Contact BEA Customer Support.
9020:ERROR	Duplicate Service Correlator = <correlator>	
	DESCRIPTION	Internal error assigning service correlator values.
	ACTION	Contact BEA Customer Support.

## B Error Messages

---

9021:ERROR	Invalid Remote Link Name <linkref>	
	DESCRIPTION	The remote link name in a request does not match any defined link name.
	ACTION	Correct the gateway configuration and restart.
9022:ERROR	Invalid transaction context = <txtxt>	
	DESCRIPTION	Internal error assigning transaction context values. Bad transaction.
	ACTION	Contact BEA Customer Support.
9023:ERROR	Unknown Service Correlator = <correlator>, message dropped	
	DESCRIPTION	Internal error assigning service correlator values. Message context lost.
	ACTION	Contact BEA Customer Support.
9024:ERROR	Invalid initial syncpoint received from subordinate, tx#<tid>	
	DESCRIPTION	Syncpoint processing protocol violation. Subordinate member of conversation attempted to initiate a syncpoint. XA does not support syncpoints from subordinate members.
	ACTION	Contact BEA Customer Support.
9025:ERROR	Invalid Input Message Discarded	
	DESCRIPTION	Internal error, bad message sent between GWSNAX and SNACRM. Possibly incompatible GWSNAX and SNACRM.
	ACTION	Contact BEA Customer Support.
9026:ERROR	CNOS Notification Received for unknown partner <partnerLU>	
	DESCRIPTION	Multiple instances of the SNACRM may be using the same local LU.
	ACTION	Ensure that multiple Tuxedo configurations do not use the same local LU.

9027:WARNING	Remote Stop Received for <linkref>	
	DESCRIPTION	The remote host has issued a stop for the specified link.
	ACTION	None. This message for information only.
9028:WARNING	Remote Start Received for <linkref>	
	DESCRIPTION	The remote host has issued a start for the specified link.
	ACTION	None. This message for information only.
9029:ERROR	Undefined Remote LU on link <linkref>	
	DESCRIPTION	The remote LU does not exist as defined.
	ACTION	Check the gateway configuration file and the stack configuration and correct the mis-match.
9030:ERROR	Unable to start session on link <linkref>. Reason=<reason>	
	DESCRIPTION	Link activation failure due to SNA error.
	ACTION	<reason> is the description of the stack return code. Determine the cause and correct.
9031:ERROR	Unable to initialize link <linkref>. Reason=<reason>	
	DESCRIPTION	Link initialization failure due to SNA error.
	ACTION	<reason> is the description of the stack return code. Determine the cause and correct.
9032:ERROR	No Available Session on link <linkref> for context <correlator>	
	DESCRIPTION	Max sessions has been exceeded.
	ACTION	Check session limits in gateway configuration, stack configuration, CICS or VTAM. Increase if necessary.

## B Error Messages

---

9033:ERROR	Requested Synclevel not supported by link <linkref> for context <correlator> (synclevel <level>)	
	DESCRIPTION	Attempted to issue a request at sync level <level> on a link that does not support that level.
	ACTION	Correct application or gateway configuration.
9034:ERROR	Service Request at SyncLevel=2 Rejected on PENDING link <linkref> for context <correlator>	
	DESCRIPTION	An attempt to start a new sync level 2 request has been received and the Link is currently processing recovery information.
	ACTION	Wait until recovery is complete to request sync level 2 services.
9035:ERROR	Inbound Request Transform Failed (<status>) for context <correlator>	
	DESCRIPTION	An error has occurred while processing the CICS transform for an inbound DPL request. This normally occurs when the API entry in the gateway configuration for the local service specifies CICS instead of ATMI.
	ACTION	Check gateway configuration for incorrect specification of local service API entry.
9036:ERROR	Inbound Response Transform Failed (<status>) for context <correlator>	
	DESCRIPTION	An error has occurred while processing the CICS transform for an inbound DPL response. This normally occurs when the API entry in the gateway configuration for the local service specifies CICS instead of ATMI.
	ACTION	Check gateway configuration for incorrect specification of local service API entry.

9037:ERROR	Outbound Request Transform Failed (<status>) for context <correlator>	
	DESCRIPTION	An error has occurred while processing the CICS transform for an outbound DPL request. This normally occurs when the API entry in the gateway configuration for the remote service specifies CICS instead of ATMI.
	ACTION	Check gateway configuration for incorrect specification of local service API entry.
9038:ERROR	Outbound Response Transform Failed (<status>) for context <correlator>	
	DESCRIPTION	An error has occurred while processing the CICS transform for an outbound DPL response. This normally occurs when the API entry in the gateway configuration for the remote service specifies CICS instead of ATMI.
	ACTION	Check gateway configuration for incorrect specification of local service API entry.
9039:ERROR	Conversation terminated without confirm for context <correlator>	
	DESCRIPTION	Sync level 2 conversation was terminated with out confirm.
	ACTION	Check application program and correct.
9040:ERROR	Inbound Confirm not supported	
	DESCRIPTION	Host application is requesting an inbound confirm. This is not supported.
	ACTION	Check host application program and correct.
9041:ERROR	Inbound Confirm for multi-ISRT not supported	
	DESCRIPTION	Host IMS application is requesting an inbound confirm and using multiple ISRT commands. This is not supported.
	ACTION	Check host application program and correct.

## B Error Messages

---

9043:ERROR	Missing send last from host (ATMI request/response) for context <correlator>	
	DESCRIPTION	Host application did not issue send last during an outbound request/response service. The host application may have abended.
	ACTION	Check application program and correct.
9044:INFO	DPL program abended with CICS code <abendcode>, program=<progname>	
	DESCRIPTION	The specified host DPL program has abended with the code specified.
	ACTION	None. This message is for information only.
9045:INFO	DPL program failed with CICS rcode <eibrcode>, program=<progname>	
	DESCRIPTION	The specified host DPL program has failed with the eibrcode specified.
	ACTION	None. This message is for information only.
9046:ERROR	Invalid combination for Service Context <correlator>, <combination>	
	DESCRIPTION	The specified <combination> is invalid. It will be one of the following: <ol style="list-style-type: none"><li>1. Sync-Level, function, and API</li><li>2. Function and API</li></ol>
	ACTION	Examine the gateway configuration and make corrections.
9047:ERROR	Sequence number error for Service Context <correlator>, seqno <seqno>	
	DESCRIPTION	There has been a sequence number failure for the specified context. Context is out of sequence.
	ACTION	Contact BEA Customer Support.

9048:ERROR	Invalid conversation task for Service Context <correlator>, task=<task>	
	DESCRIPTION	The conversation has already been terminated.
	ACTION	Contact BEA Customer Support.
9049:ERROR	Invalid task switch for Service Context <correlator>, from <task1> to <task2>	
	DESCRIPTION	An internal protocol violation has occurred.
	ACTION	Contact BEA Customer Support.
9050:ERROR	Transformer creation failed for inbound transaction <trancode>	
	DESCRIPTION	An internal error has occurred. Possibly out of memory.
	ACTION	Contact BEA Customer Support.
9051:ERROR	Transformer failed for inbound transaction <trancode>	
	DESCRIPTION	An internal error has occurred. Resource name is not present. Mainframe compatibility problem.
	ACTION	Contact BEA Customer Support.
9052:WARNING	Inter-task Message dropped (<verbname>), parm=<parm> From: <task1> to <task2>	
	DESCRIPTION	An internal message between two tasks has been dropped.
	ACTION	None. This message is for information only.
9053:ERROR	Attempt to send <nnnnn> bytes (> 32767)	
	DESCRIPTION	The length of a send request exceeded 32767 (including overhead).
	ACTION	Check application program and correct.

## B Error Messages

---

9054:ERROR	Allocation Failure for <trancode> on <remotesysid>: <error>	
	DESCRIPTION	An Allocation error occurred.
	ACTION	The reason for the failure is described by <error>. Correct problem with configuration or application.
9055:ERROR	Invalid Exchange Logs GDS variable received from <remotesysid>	
	DESCRIPTION	The log files for the SNACRM have been incorrectly modified.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.
9056:ERROR	Invalid cold start received from <remotesysid>. Unrecovered local transactions are pending.	
	DESCRIPTION	Attempting to cold start host while warm starting Tuxedo.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.
9057:ERROR	Invalid warm start received from <remotesysid>. Unknown log name.	
	DESCRIPTION	The log files for the SNACRM have been incorrectly modified.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.
9058:ERROR	Invalid Compare States GDS variable received from <remotesysid>	
	DESCRIPTION	The log files for the SNACRM have been incorrectly modified.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.



9059:ERROR	Mixed Heuristic on link <linkref> for <unitofwork> Correlator [<correlator>]	
	DESCRIPTION	One side has reported committed while the other side has reported aborted.
	ACTION	Check the ULOG for any additional messages.
9060:WARNING	Inbound Exchange Logs Rejected for <remotesysid>	
	DESCRIPTION	Link not configured for sync level 2.
	ACTION	None. This message is for information only.
9061:WARNING	Link <linkref> not configured for sync level 2	
	DESCRIPTION	Link specified by <linkref> is not configured for sync level 2.
	ACTION	None. This message is for information only.
9062:ERROR	Exchange Logs Rejected for <remotesysid>, Restart Type or Log Name Mismatch	
	DESCRIPTION	The log files for the SNACRM have been incorrectly modified.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.
9063:ERROR	Exchange Logs failed with <linkref>	
	DESCRIPTION	An error occurred during the exchange logs process.
	ACTION	Run CRMLOGS to examine the SNACRM log file. Cold start the Tuxedo application.
9064:ERROR	Invalid initial syncpoint received from subordinate, %s	
	DESCRIPTION	An internal error has occurred during the commit process. XA does not support syncpoints originating from subordinate members.
	ACTION	Contact BEA Customer Support



---

# Glossary

## A

### Access Control Lists (ACL)

A Tuxedo security feature that controls client access to services by means of lists that are automatically checked each time a service is requested.

### ACID Properties

The essential characteristic of transaction processing systems:

*Atomicity*: All changes that a transaction makes to a database are made permanent, or else are nullified.

*Consistency*: A successful transaction transforms a database from a previous valid state to a new valid state.

*Isolation*: Changes that a transaction makes to a database are not visible to other operations until the transaction completes its work.

*Durability*: Changes that a transaction makes to a database survive future system or media failures.

### Application

A BEA Tuxedo System/T *application* is bounded by the environment described in a single `TUXCONFIG` file. In /Domain, a BEA Tuxedo System/T application can communicate with another application via a domain gateway group.

### Application Domain

When used alone, the term *Domain* can mean a number of things. In order to avoid confusion, the term *application domain* is used to refer to a BEA Tuxedo application bounded by the configuration of a `tmconfig` file. This application domain can be restricted to a single platform or shared memory (SHM) environment, or it can be scaled across multiple machines in a multiple processor (MP) environment.

### Application Programming Interface (API)

1) The verbs and environment that exist at the application level to support a par-

---

ticular system software product. 2) A set of code that enables a developer to initiate and complete client/server requests within an application. 3) A set of calling conventions that define how to invoke a service. A set of well-defined programming interfaces (entry points, calling parameters, and return values) by which one software program utilizes the services of another

### **Application Program-to-Program Communication (APPC)**

An interface to LU6.2 services; provides a set of primitives to conduct conversations in LU6.2 sessions.

## **C**

### **Client**

A program designed to request information from a server.

### **CNOS**

*CNOS* are service programs implemented as part of an LU6.2. The *CNOS* programs negotiate session limits between the two communication LU.

### **Common Programming Interface for Communications (CPI-C)**

An interface to LU6.2 services. It is a simpler set of primitives than the APPC interface and is intended for use in program-to-program communications.

### **Customer Information Control System/Enterprise System Architecture (CICS/ESA)**

An operating environment devised by IBM that provides a foundation upon which to write customer applications programs. Several facilities useful for programming are supplied by the CICS environment, including basic mapping services (BMS), transient data queues (TD), temporary storage files (TS), memory services, etc. Customer applications are built as separate transaction programs, and are invoked as transactional tasks. CICS/ESA is a trademark of International Business Machines (IBM), Inc.

## **D**

### **Distributed Program Link (DPL)**

Function of CICS ISC that supports LINK requests between CICS regions, and is similar to a BEA Tuxedo request/response.

---

### **Distributed Transaction Processing (DTP)**

A CICS intercommunication in which processing is distributed among transactions that communicate synchronously over intersystem or inter-region links. It is roughly equivalent to BEA Tuxedo conversations.

### **E**

### **ESA**

(ESA) Enterprise Systems Architecture is the conceptual structure and functional behavior of the latest range of IBM mainframe computers. ESA/370 is the fourth step in an evolution of which the first three steps were System/360, System/370, and System/370 extended architecture (370-XA).

### **F**

### **Field Manipulation Language (FML)**

A set of C language functions for defining and manipulating storage structures called field buffers. Cooperating processes can send and receive data in fielded buffers.

### **FML Buffer**

A buffer of self-describing data items accessed through the Field Manipulation Language (FML) API.

### **I**

### **Inbound**

A generic term referring to request message direction relative to the server or a response message direction relative to the client.

### **Information Management System (IMS)**

A database manager used by CICS/ESA to allow access to data. IMS provides for the arrangement of data in an hierarchical structure and a common access approach in application programs that manipulate IMS databases.

### **InterSystem Communications (ISC)**

Communication between separate systems by means of SNA networking facilities or by means of the application-to-application facilities. ISC links CICS systems to other systems and may be used for communication between user applications or to transparently execute CICS functions on a remote CICS system.

---

## **L**

### **Logical Unit (LU)**

In SNA, a port through which a user gains access to the services of a network. Also, see System Network Architecture (SNA).

### **LU6.2**

*LU6.2* is a particular SNA logical unit that identifies a specific set of services for program to program communication. Services include syncpoint, mapping of buffers into records, message confirmation, and security.

## **M**

### **MODENAME**

**MODENAME** is a configuration parameter that names a set of characteristics for a group of BEA eLink Java Adapter for Mainframe sessions. In the CICS region, the mode is defined in VTAM and referenced in CICS and the `DMCONFIG` file.

### **mirror task**

CICS/ESA task that services incoming requests that specify a *mirror transaction* (CSMI, CSM1, CSM2, CSM3, CSM5, CPMI, CVMI, or a user-defined mirror transaction identifier).

### **mirror transaction**

CICS/ESA transaction that recreates a request that is function shipped from one system to another, issues the request on the second system, and passes the acquired data back to the first system.

### **Mirror Transaction Identifier Support**

BEA eLink Java Adapter for Mainframe feature which enables BEA Tuxedo clients to invoke host CICS/ESA programs and, conversely, CICS/ESA client programs to invoke BEA Tuxedo services. Based on the IBM CICS/ESA mirror transaction.

### **Multiple Virtual Storage (MVS)**

An operating system for processing systems consisting of one or more mainframe processors.

---

## O

### Outbound

A generic term referring to request message direction relative to the client or response message direction relative to the server.

## P

### PU 2.1

PU 2.1 is an SNA server that provides client programs with access to an SNA network. The PU 2.1 Server's functions include managing the physical connections, providing SNA node device emulation, servicing Logical Units, communicating with client programs, and controlling and monitoring the local SNA resources.

## R

### Resource Definition Online (RDO)

The recommended method of defining resources to CICS/ESA. Resource definitions are created interactively by a CEDA transaction, or by the DFHCSDUP utility. Both methods store definition in the CICS/ESA system definition data set (CSD). At CICS initialization, CSD definitions are selectively installed as CICS system tables controlled by a user-supplied list of definitions. CEDA-defined resource definitions can be installed while CICS is active and used immediately.

## S

### Server

A computer or program that is dedicated to providing information in response to external requests.

### Session

When two LU bind with each other, that is, when they have successfully negotiated how they will communicate, they are said to be in *session*. SNA has fixed limits on the number of sessions configured for an LU type.

### SNA Communication Resource Manager (SNACRM)

A process that provides all of the sync-level two logic for an SNA domain gateway and directly communicates with the PU2.1 server.

---

## **Stack**

Platform vendor-supplied software that provides connectivity to an SNA network.

## **Synchronization Level (sync level)**

The level of synchronization (0, 1, or 2) established for an APPC session between intercommunicating CICS/ESA transactions. Level 0 gives no synchronization support, level 1 allows the exchange of private synchronization requests, and level 2 gives full CICS/ESA synchronization support, with backout of all updates to recoverable resources if failure occurs.

## **System Network Architecture (SNA)**

A seven-layer networking protocol. Each layer of the protocol has a set of associated data communication services. The services of the uppermost layer are embodied in a Logical Unit (LU). Each LU type defined in SNA has its own specific set of services available to an end user for communicating. The end user may be a terminal device, or an application program. The SNA structure enables the end user to operate independently, unaffected by the specific facilities used for information exchange.

## **T**

### **Transaction**

- 1) A complete unit of work that transforms a database from one consistent state to another. In DTP, a transaction can include multiple units of work performed on one or more systems.
- 2) A logical construct through which applications perform work on shared resources (e.g., databases). The work done on behalf of the transaction conforms to the four ACID properties: atomicity, consistency, isolation, and durability.

### **Transaction Processing (TP)**

A form of immediate data processing in which user requests are entered directly to the terminal and on-line programs satisfy the requests; for example, by updating database files and displaying output messages.

## **V**

### **Virtual Telecommunications Access Method (VTAM)**

A set of programs that control communication across a network between terminals and application programs.



---

# Index

## A

- Adobe Acrobat Reader viii
- APPN system 390
  - VTAM configuration 2-1

## C

- CICS/ESA LU
  - configuration 1-6
- crmlkoff command A-4
- crmlkon command A-2
- crmlogs command A-1
- customer support contact information xi

## D

- documentation, where to find it viii

## E

- error messages B-1
- ethernet lan
  - VTAM configuration 2-1

## H

- HTML
  - Web User Interface for this guide viii

## L

- link

- stop command A-4
- log files
  - SNACRM A-1

## P

- PU2.1 server (stack) 1-3

## R

- related information ix

## S

- servers
  - SNACRM A-6
- SNA protocol 1-3
- SNACRM 1-4
  - command A-6
- stack traces 1-16
- support
  - technical xi

## T

- TCP/IP 1-4

## V

- VTAM
  - configuration 1-6

---

## W

Web User Interface (WUI)

HTML interface for this guide viii

WebLogic Server (WLS) 1-4

winner sessions 1-7