Signaling Gateway User Manual

Part No. 160 - 2001 -01

Release 1.6.0

April 8, 2009

494 Bridgeport Avenue, #326
Shelton, CT 06484-4748

Technical Assistance: (800) 416-1624
(408) 432-2600
fax: (408) 428-3548
email: support@newnet.com

TRADEMARKS

NewNet®, and AccessMANAGER® are registered trademarks of NewNet Communications Technologies, LLC

NewNet AccessMANAGER™, NewNet Connect7™, NewNet Disributed7™, NewNet Easy7™, NewNet SG™, NewNet SGC™, NewNet OTAserver™ and NewNet SMserver™ are trademarks of NewNet Communication Technologies, LLC

INFORMIX C-ISAM is a registered trademark of Informix Software, Inc. in the U.S.A. and other countries.

Solaris® is a registered trademark of Sun Microsystems, Inc.

Sun™, Sun-3™, Sun-4™, Sun386i™, SunInstall™, SunOS™, and SPARC Sun Microsystems™, and Sun Workstations™ are trademarks of Sun Microsystems, Inc.

SPARC® is a registered trademark of SPARC International, Inc. SPARC CPU-2CE™ is a trademark of SPARC International, Inc. licensed to FORCE COMPUTERS, Inc.

Motorola® and the Motorola logo are registered trademarks of Motorola, Inc. in the U.S.A. and other countries.

FX Series™ is a trademark of Motorola Computer Group.

AIX®, PowerPC®, RS/6000®, and ARTIC960® are registered trademarks of IBM, Inc.

UNIX® is a registered trademark of UNIX Systems Laboratories, Inc. in the U.S.A. and other countries.

All the brand names and other products or services mentioned in this document are identified by the trademarks or service marks of their respective companies or organizations.

SUCCESSOR IN INTEREST

NewNet Communication Technologies, LLC . is the successor in interest to EBS, Inc.; NewNet, Inc.; ADC Enhanced Services Division; ADC ESD, Inc.; Centigram Communications Corporation and SS8 Networks. Any rights or title to the marks or copyrights of these entities, unless otherwise disclosed, are the property of NewNet Communication Technologies, LLC.

NOTICES AND WARRANTY INFORMATION

The information in this document is subject to change without notice and should not be construed as commitment by NewNet Communications Technologies, LLC. NewNet Communications Technologies assumes no responsibility or makes no warranties for any errors that may appear in this document and disclaims any implied warranty of merchantability or fitness for a particular purpose.

COPYRIGHT INFORMATION

PERFORMANCE SPECIFICATIONS

NewNet Communications Technologies, LLC reserves all the rights to change the equipment performance specifications stated herein at any time without notice. For OEM components, NewNet Communications Technologies relies on the specifications supplied by the OEM vendors.

Table of Contents

Table of Contents

Table of Contents

Table of Contents

List of Figures

List of Tables

Revision History

The following table lists the revision history of this manual. The Date column shows the date a revised manual was published. The Associated Software Release column shows the software release number for which the updated manual was published

.

Date	Pages Replaced	Description of Changes	Assoc SW Release
9/5/2007	Chapter 2	Edited section 2.4 Software Components Sun Solaris versions supported. Edited section 2.6 Computing Platform for Intel AMD and Sun Solaris versions supported.	1.5.0.1
	Chapter 4	Edited section 4.2.4 Application Processes, Table 4-3 and Signaling Gateway Core Processes - new bullet added. Edited section 4.3.1 Starting the Software, #3 Signaling Gateway Core process - added sctpd for Solaris 8/9. Edited bullet 3 with Solaris 8/9 and sctpd info. Edited section 4.5.1.3 SCTP Configuration, added Solaris 8/9 (sctpd) process to step #4.	
	Chapter 6	Section 6.1 Chapter Overview, added sctpd to table 6-1. Section 6.4.1 sctpd, new section added. Section 6.4.2 sgpd, added support for Solaris 8/9 under -a and -p.	
7/6/07	Chapter 4	Updated section 4.5.1.3.	1.5.0 GA
	Chapter 5	Updated section 5.2.4 Route Set.	
6/15/07	Chapter 1	Updated section 1.5 Document and References.	1.5.0 beta
2/7/07	All	Deleted references to SCTPD.	1.5.0 beta
	Chapter 3	Changed Solaris 2.8 references to Solaris 10.	
	Chapter 6	Deleted 6.4.1 sctpd process.	
7/31/06	Chapter 2	Table 2-2: removed references to gnu.	1.4.0
	Chapter 5	Section 5.1.3: removed reference to History command in line 5. Table 5-4: removed histbuf row from MMLCONF entry. Section 5.4.6 MMLCONF: removed histbuf parameter. Removed Section 5.4.10 History.	
	Chapter 6	Section 6.2.12 mml: removed out-of-date listings from Files. Section 6.2.13 mmi: removed out-of-date listings from Files.	
7/22/05	All	Revised version numbers and dates for D7 1.4.0.	1.1.1
	Chapter 2	Updated Table 2-1 Capacity. Updated Table 2-2 Host Platform Options. Updated Table 2-3 Available SS7 Controller Options.	

Date	Pages Replaced	Description of Changes	Assoc SW Release
2/21/03	Chapter 2	Updated Table 2-1 Capacity.	1.1.1
		Section 2.6.2 SS7 Controller Options — Updated footnote 9 to include ARTIC1000/2000 boards and updated Table 2-3	
	Chapter 3	Updated D7 release to 1.3.0; updated Disk Space Requirements, Section 3.2.1; deleted note after step 11 in Installation, Section 3.2.2.2	
	Chapter 4	Figure 4-7 Distributed7 MIB MTP Tree 3 — Updated to include new link parameters. L2ECM, PCRN1 and PCRN2	
		Figure 4-8 Distributed7 MIB Hardware Tree — Added pmlink and ctbusTable branches from ss7board	
		Section 4.5.4.2 Local SGP — Updated SGSGP MO command examples and added note about sctpport=2905	
		Section 4.5.4.3 Remote ASP — Updated command examples, deleting sctpport=2905	
		Section 4.5.4.4 Remote Peer SG and SGP — Updated command examples, deleting scpport=2905	
		Section 4.5.4.5 IP Destination Point Code — Updated remote AS command examples	
		Section 4.5.4.6 SS7 Destination Point Code — Updated command example	
	Chapter 5	Table 5-2 MTP Configuration Managed Objects — Modified LINK object with new parameters L2ECM, PCRN1 and PCRN2; added CTBUS object; modified ROUTE, SS7 BOARD, LINE, LINEHIST, LINESTAT, PORT, TIMESLOT	
		Table 5-6 Passive Monitor Managed Object — Added new table	
		Section 5.2.1 Link (LINK) — Modified to include L2ECM, PCRN1 and PCRN2	
		Section 5.2.5 SS7Board (SS7BOARD) — Added ***pm*** to the parameters section of command descriptionl; updated commands, parameters, errors and examples for CT BUS and artic8260	
		Section 5.2.10 Line (LINE) and Section 5.2.15 Port (PORT) — Modified sections	
		Section 5.2.19 Time Slot (TIMESLOT) — Updated commands, parameters, errors and examples for CT BUS and artic8260	
		Section 5.2.21 CT Bus (CTBUS) — Added new section	
		Section 5.3.1 Concerned Point Code (CPC) — Edited commands	
		Section 5.3.2 Global Title (GT) — Edited names, commands, parameters, errors and examples	
		Section 5.3.3 Global Title Entry (GTENTRY) — Edited commands, parameters, errors and examples	
		Section 5.3.4 Mate — Edited commands and errors	
		Section 5.3.6 SCCP Signalling Point — Edited commands and errors	
		Section 5.3.7 Subsystem — Edited commands and errors	
		Section 5.5.3 Application Server (SGAS) — Modified ADD command, ***mode*** and ***asplist*** parameters, examples and sample output	

Revision History

Date	Pages Replaced	Description of Changes	Assoc SW Release
2/21/03	Chapter 5	Section 5.5.4 Application Server Process (SGASP) — Modified ***sctpport*** parameter and examples	1.1.1
		Section 5.5.9 SS7 Destination Point Code (SGDPC) — Modified ***mode*** parameter and examples	
		Section 5.5.10 Peer SG (SGPSG) — Modfied ADD command and ***mode*** parameter	
		Section 5.5.11 Peer SGP (SGPSGP) — Modfied ADD command, ***sctpport*** parameter and examples	
		Section 5.6 Passive Monitor MML Commands — Added new section	
	Chapter 6	Section 6.2.4 AccessSNMP — Updated daemon	
		Section 6.2.12 MML — Updated errors	
		Section 6.2.13 MMI — Updated errors	
		Section 6.2.14 netd — Updated ***netd*** daemon, which now takes '-i' command-line option for Solaris IP network multipathing (IPMP) support	
		Section 6.2.16 spmd — Added artic8260 to the list of boards in the third paragragh of DESCRIPTION	
	Chapter 7	Artic8260 board references added throughout chapter (Sections 7.2.2, 7.2.3, 7.2.6, 7.2.8, 7.2.19, 7.3.5 and Table 7-3)	
		Table 7-1 User Command Summary — Added ***ebs_dbconfig*** to table	
		Section 7.2.5 ebs_dbconfig — Added new section	
		Section 7.2.17 ebs_sysinfo — Updated ***ebs_sysinfo*** utility, which now provides '-l' command-line option	
	Appendix B	Section B.1 Chapter Overview — Added ARTIC8260 board	
		Table B-2 MTBF Ratings — Added MTBF figures for ARTIC1000 and ARTIC2000 boards	
		Section B.2.4 Installing a new PCIbus card — Removed section	
		Section B.2.7 ARTIC8260 Boards (ARTIC1000 and ARTIC2000) — Added new section for new boards	
		Section B.2.8 Installing a new PCIbus card — Modified steps 2 and 7 to include -f command	
		Section B.3.3 ARTIC1000 Compact PCI Bus Configuration — New section added	
	Appendix C	Section C.1.2 Alarm Groups and Trouble Tables — Added bullet for GSM-A Interface; updated range of alarms for ETMOD, ISUP, OMAP, DKM and ISUPMOD categories; added bullets for new alarm groups PMMOD and PMON	
		Tables C-1 through C-15 — Updated alarm messages and levels	
		Table C-16 GSM-A Alarm Group — Added new alarm table	
		Table C-17 PMMOD Alarm Group — Added new alarm table	
		Table C-18 PMON Alarm Group — Added new alarm table	

Date	Pages Replaced	Description of Changes	Assoc SW Release
11/25/02	Chapter 1	Changed product name to Signaling Gateway; added bullets to Section 1.5, Related Documents and References	1.1.0
	Chapter 2	Changed product name to Signaling Gateway; modified Signaling Gateway and Signaling Gateway Process descriptions in Section 2.1.2, SIGTRAN Overview; modified paragrah text (for peer SG) in Section 2.1.3 NewNet's Signaling Gateway Suite; updated Section 2.3, Signaling Gateway Features and Capabilities with new description of functionality and new Figure 2-6 Signaling Gateway's SS7-IP Interworking	
	Chapter 3	Changed product name to Signaling Gateway; updated patch in Caution statement under Section 3.2.2, Installation Steps; updated D7 release to 1.2.1, directories and steps in Section 3.2.2.2, Installation	
	Chapter 4	Changed product name to Signaling Gateway; modified the procedures in Section 4.3.1, Starting the Software and Section 4.3.2, Stopping the Software; updated the SCTP Daemon Configuration File in Section 4.5.1.3, SCTP Configuration; changed ANSI_92 to ANSI_96 in Section 4.5.3.1, MTP Configuration; updated Section 4.5.4, Signaling Gateway Configuration for peer SG/SGP; updated Figure 4-11 Signaling Gateway MIB Tree	
	Chapter 5	Changed product name to Signaling Gateway; added HOSTNAME parameter to Table 5-4 Signaling Gateway Configuration Managed Objects SGSGP and SGASP and added new MOs SGPSG, SGPSGP and SGDPC; updated Signaling Gateway Managed Objects in Section 5.5; added new sections 5.5.9, SS7 Destination Point Code (SGDPC), 5.5.10, Peer SG (SGPSG) and 5.5.11, Peer SGP (SGPSGP)	
	Chapter 6	Changed product name to Signaling Gateway; updated directory paths in Section 6.4.1, sctpd and Section 6.4.2, sgpd	
	Chapter 7	Changed product name to Signaling Gateway; updated utility information in Section 7.4.2, sg_start amd Section 7.4.3 sg_stop; changed sgp case in Section 7.4.5, sg_trace	
	Chapter 8	Changed product name to Signaling Gateway; updated alarms in Table 8-1: SGP Alarm Group; changed directory path in Section 8.3.2, Log Files; corrected Caution statement in Section 8.3.3, Runtime Tracing	
11/24/02	Chapter 1	Updated the version of M3UA and the document numbers of the SGC manuals, and added the new SGC API Reference Manual in Section 1.5, Related Documents and References.	1.1.0 Beta
11/24/02	Chapter 2	Updated all sections of this chapter to reflect the new functionality introduced in this release, including multiple hosts, SCCP, GTT and IPsec support, and increases in capacity. Also moved Section 3.2, Computing Platform from Chapter 3 to the end of Chapter 2.	1.1.0 Beta
11/24/02	Chapter 3	Moved Section 3.2 Computing Platform to the end of Chapter 2, updated the disk space requirements, and divided the steps into Installation Preparation, with separate headings for CD and Tape, and Installation Steps.	1.1.0 Beta
11/24/02	Chapter 4	Updated the configuration steps to include the latest SS7 Distributed7, SCCP, GTT and IPsec support,	1.1.0 Beta
11/24/02	Chapter 5	Updated the MTP, and System MML commands with the SS8 Distributed7 1.2.Ø changes.	1.1.0 Beta
		Changed the SG, SGP, AS, ASP, ASSTATUS and RK MML commands, added the SGRKRNG and SGSPMC MML commands, and deleted the CONN and M3UATRC MML commands.	
11/24/02	Chapter 6	Added a new Chapter 6, System Processes documenting the SS8 Distributed7 and Service-Controller 1510 daemons..	1.1.0 Beta
11/24/02	Chapter 7	Added ebs_brdfinfo, ebs_audit, ebs_lob, ebs_sysinfo and all the apm utilities to the User Commands chapter. Updated the sg_start, sg_stop and sg_trace commands.	1.1.0 Beta
4/30/02	Chapter 3	Updated the installation steps to reflect the changes for the GA release.	1.0.0
4/30/02	Chapter 5	Updated the MML commands from SS8 Distributed7 1.2.Ø Beta to the GA version, which includes information about new boards.	1.0.0

Revision History

Date	Pages Replaced	Description of Changes	Assoc SW Release
4/30/02	Chapter 6	Added the AccessMOB, AccessStatus, AccessAlarm and AccessOMAP commands to the chapter.	1.0.0
		Added **pci3xapq** as a value for the devname parameter for the ***ebs_cfgbrd***, ***ebs_dnlbrd***, and ***ebs_mnglbrd*** commands.Also added the **PCI370APQ** and **PCI372APQ** values to the Board Types in the ***getcfg*** command.	
4/30/02	Appendix B	Removed all Sbus and PCI370/372 board information. Updated the PCI370PQ/372PQ board information to PCI370APQ/372APQ, noting that the only difference is that the PCI370APQ/372APQ boards have more memory (32 MB instead of 8MB on the PCI370PQ/372PQ boards).	1.0.0

Chapter 1: Introduction

1.1 DocumentOverview

This document describes the architecture and interface of Signaling Gateway Signaling Gateway, which will be interchangably referenced as SG..

1.2 Scope

The manual is organized into the following chapters describing the Signaling Gateway:

- Chapter 1: Introduction - introduces this manual, including specific notations and conventions used in the manual, and related documents and references.
- Chapter 2: Overview - provides an overview of the Signaling Gateway that describes the architecture, key concepts, and features of the Signaling Gateway.
- Chapter 3: Installation - has the steps to install the Signaling Gateway software.
- Chapter 4: Operations, Administration and Maintenance - provides an in depth discussion of how the Signaling Gateway system operates. Also included is a sample configuration section that describes and provides examples of the usage of MML sessions and commands for SS7 and M3UA configuration.
- Chapter 5: MML Commands - provides reference information for the Signaling Gateway service provisioning, statistics, general system, and Managed Objects including syntax and usage and a set of common UNIX commands
- Chapter 6: System Processes - provides descriptions of the system processes (daemons) and their configuration files.
- Chapter 7: User Commands - documents the user commands (scripts) that come with the Signaling Gateway software.
- Chapter 8: Maintenance and Troubleshooting - identifies the Signaling Gateway alarms and troubleshooting tools.
- Appendix A: Glossary- lists frequently used acronyms and definitions,
- Appendix B: Hardware Installation- has detailed information about the hardware requirements and installation.
- Appendix C: Alarms- lists the NewNet Distributed7 alarms by alarm group.

1.3 How to Use this Manual

The *Signaling Gateway User Manual* is intended for use by the operator during operation, configuration, and maintenance of the software and to aid with application development. This manual and all documents referenced within are required for proper operation.

1.4 Notations and Conventions

This paragraph describes the notations and conventions that are used in this manual. Conventions have been established for commands and file names, window names and heading names, acronyms, mnemonics, and alert messages. Each of these conventions is described in a subsection that follows.

1.4.1 Revisions and Updates

NewNet Communications Technologies seeks to provide total quality and customer satisfaction through continuous improvement. Toward that goal, revisions to the manual will occur from time to time due to software enhancements or documentation enhancements. Documentation changes will be included in the release notes for software point releases.

1.4.2 Alert Messages

ANSI A535 specifications define specific words and icons that alert the reader to dangerous situations that may result in injury to a person or the equipment. These have been adapted for use with NewNet Communications Technologies software.

***Important**: Recommendations, guidance, hints, tips, and shortcuts to alert readers to situations and procedures that, if not followed properly, may complicate or prevent proper operation of the software.*

***Caution:** Situations that, if not avoided, may corrupt the software or stop it entirely.*

***Notice**: Situations that, if not avoided, may cause damage to the equipment.*

1.5 Related Documents and References

This paragraph lists the related documents that are beyond the scope of the Signaling Gateway manual set. The documents listed below are referenced from this manual, and contain additional information that may be helpful to the user:

- Framework Architecture for Signaling Transport, RFC 2719, Oct 1999
- SS7 MTP3-User Adaptation Layer (M3UA), RFC 3332, September 2002
- SS7 MTP3-User Adaptation Layer (M3UA), RFC 4666, September 2006
- M3UA Implementer's Guide, draft-ietf-sigtran-m3ua-implementors-guide-01.txt
- M3UA SG-SG Communication, draft-bidas-sigtran-sgsg-01.txt, September 2002
- Stream Control Transmission Protocol (SCTP), RFC 2960, Oct 2000
- Stream Control Transmission Protocol (SCTP) Implementers Guide, draft-ietf-tsvwg-sctpimpguide-06.txt, May 2002
- Stream Control Transmission Protocol (SCTP) Checksum Change, RFC 3309, September 2002
- SNMPv1, RFC 1157
- SNMPv2 RFC 1905, RFC 1906
- NewNet Distributed7™ Manual Set CD-ROM, 2700-1796-01
- Signaling Gateway Client™ User Manual, 160-3001-01
-

Chapter 2: Overview

2.1 Introduction

This chapter provides background information on the SS7 and SIGTRAN protocols, which are the foundation of understanding Signaling Gateway as bridging the SS7 and IP networks.

2.1.1 SS7 Overview

Signaling System Number 7 (SS7) is the protocol for sending signals in a digital network instead of using voice connections. This protocol is a set of rules that define the exchange of Switched Circuit Network (SCN) native signaling information between SS7 network elements and the Public Switched Telephone Network (PSTN). The SS7 protocol defines the sequence in which SCN signaling is sent using transmission paths called *links*. Figure 2-1 shows the SS7 network architecture and is followed by brief explanations of the diagram labels.

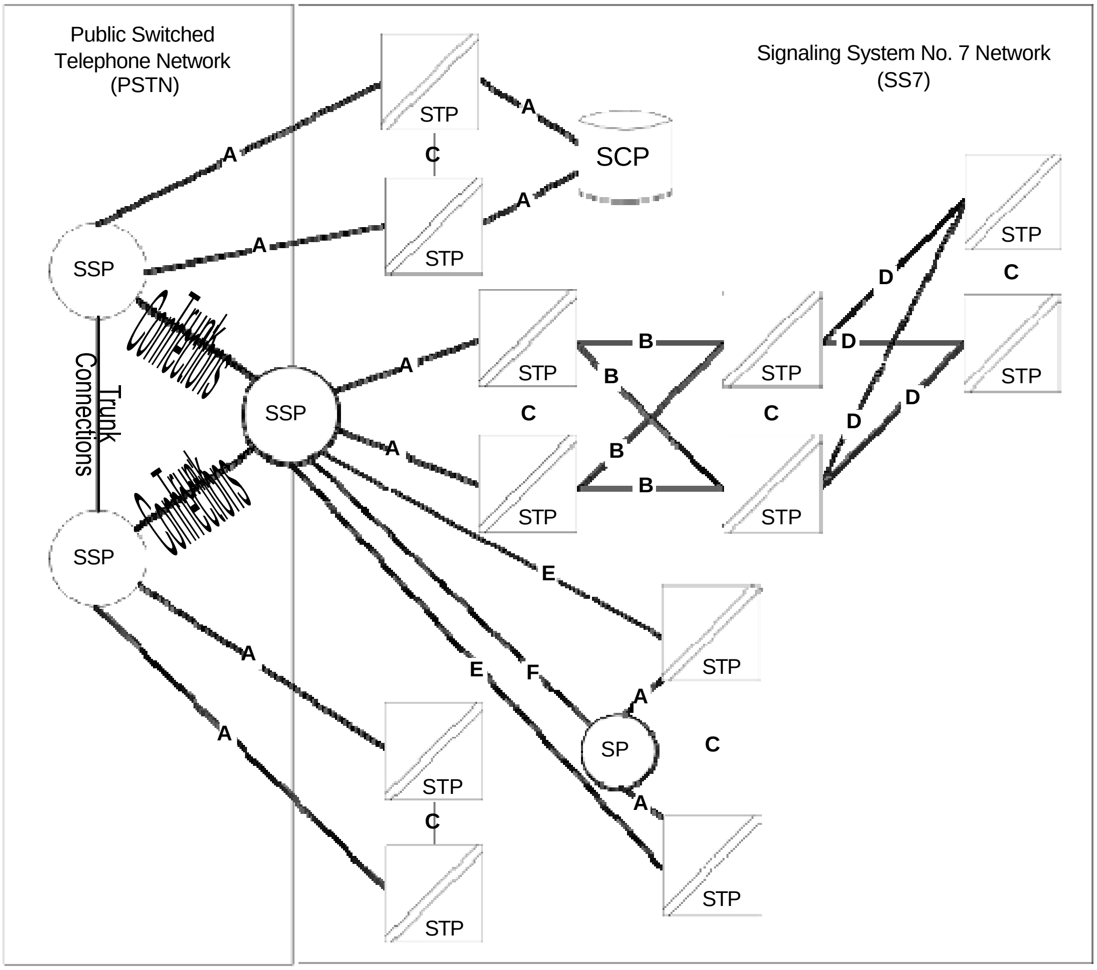


Figure 2-1: SS7 Network Architecture

The following describes the acronyms and labels used in Figure 2-1:

- **STP** or **Signaling Transfer Point** is the router in an SS7 network that sends and receives SS7 signals to and from other signaling points. STPs are always paired for redundancy, so that if one of the pair is not functioning then the other one handles the load.
- **SP** or **Signaling Point**, is any node in an SS7 network that is capable of sending and receiving SS7 signals. It is a generic term that is used when what a node does is not important. STP, SSP and SCP are all signaling points. STPs must be located in different geographic locations to ensure redundancy, but any other SP may be in the same geographic location as an STP. Each SP always has its own **S**ignaling **P**oint **C**ode (SPC), which is its network address.
- **SSP** or **Service Switching Point**, is a signaling point that handles call set up, stops call processing if necessary, queries databases (such as 800 numbers or credit card verification), and responds appropriately to the result of a database query.
- **SCP** or **Service Control Point**, is a signaling point that acts as the front end of a database by handling queries and sending the answer back to the requesting node. An SCP has a

mechanism to retrieve data from a database in a format that the requesting node can use. An SCP may or may not be in the same location as the database.

***Note**: A Signaling End Point, SEP, is any location in the SS7 network that originates and/ or terminates SS7 messages. This generic term is useful only to discuss SS7 architecture without the need to define the specific functions of a node. Therefore, other terms (SSP, SCP, IP, etc.) more fully define what happens at the node, but all are SEPs.*

- **Links** are the transmission paths between locations in an SS7 network. They are referred to by the letters **A through F**, and these letters represent what the links do:
  - **A** Links **Access Links** give signaling points *access* to the SS7 network.
  - **B** Links **Bridge Links** connect pairs of STPs in every possible way to create a *bridge* between the two.
  - **C** Links **Cross Links** are the links that connect STP pairs and allow messages to *cross* over from one to the other. They are required to provide an STP's redundancy.
  - **D** Links **Diagonal Links** connect two different levels in an SS7 network in every possible way. The hierarchy in SS7 is conceptual and represented in diagrams by placing STPs at a higher level, above STPs at a lower level. Going from local to regional STPs, or regional to national STPs are examples of SS7's conceptual hierarchy. The links are always *diagonal* lines to connect the two levels, one higher than the other in drawings.
  - **E** Links **Extended Links** connect an SP to an STP in another area to *extend* a signaling point's routing flexibility. An E link provides access to the network like an A link, but is called an E link because it *extends* an SP's routing flexibility beyond the access an A Link provides.
  - **F** Links **Fully Associated Links** connect two locations to specifically isolate their communication from the network so that it is a direct connection.

2.1.1.1 SS7 Protocol Stack

Figure 2-2 is a standard diagram of the first three functional layers of SS7 that deal with a message just about to be transmitted over the links, or one just received from the links. These three levels are Message Transfer Part (MTP) Levels 1, 2 and 3, or MTP1, MTP2 and MTP3. Stacked above the MTP 1, 2 and 3 layers of the User Parts are the Level 4 User Parts, as shown in Figure 2-2:

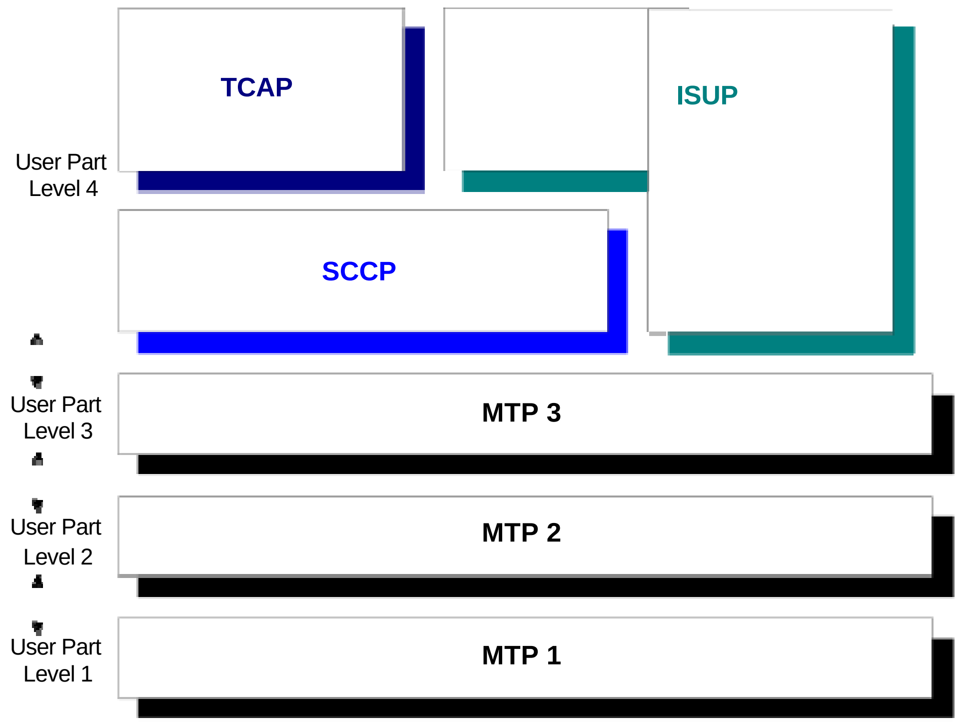


Figure 2-2: SS7 Protocol Stack

Message Transfer Part (MTP)

SS7's MTP layers are the base of the SS7 stack:

- **MTP Level 1** defines the physical, electrical and functional characteristics of the signaling link. This level requires that a link must be bi-directional.
- **MTP Level 2** provides the reliable transfer of signaling information between two adjacent signaling points. It is the last to handle messages being transmitted and the first to handle messages being received. It monitors links and reports on their status, checks the integrity of messages, discards bad messages, requests copies of discarded messages, acknowledges good messages, places links in service and restores links that have been taken out of service. It reports most of what it does to MTP Level 3.
- **MTP Level 3** defines signaling message handling procedures and signaling network management functions. It is responsible for:
  - *Signal Message Handling* controls message discrimination (routing to and from MTP Level 2) and message distribution (routing to and from Level 4 User Parts).
  - *Signaling Network Management* manages traffic, links, routing and congestion flow control.

Level 4 User Parts

The SS7 Level 4 User Parts define the messages and procedures for a particular application running on top of MTP. The following describes the standard Level 4 User Parts:

- **Signaling Connection Control Part** (SCCP) - extends the addressing capability of MTP and establishes non-circuit related signaling connections. It also controls database redundancy.
- **Transaction Capabilities Application Part** (TCAP) - defines non-circuit related protocol over SCCP, such as database queries.
- **Integrated Services Digital Network User Part** (ISUP) - defines the protocol to control circuit-switched services in the Integrated Services Digital Network (ISDN). This part provides the functions for handling telephone-call-related messaging that is sent from switch to switch in a voice network.

The following diagram illustrates one example of how the SS7 stack works with applications.

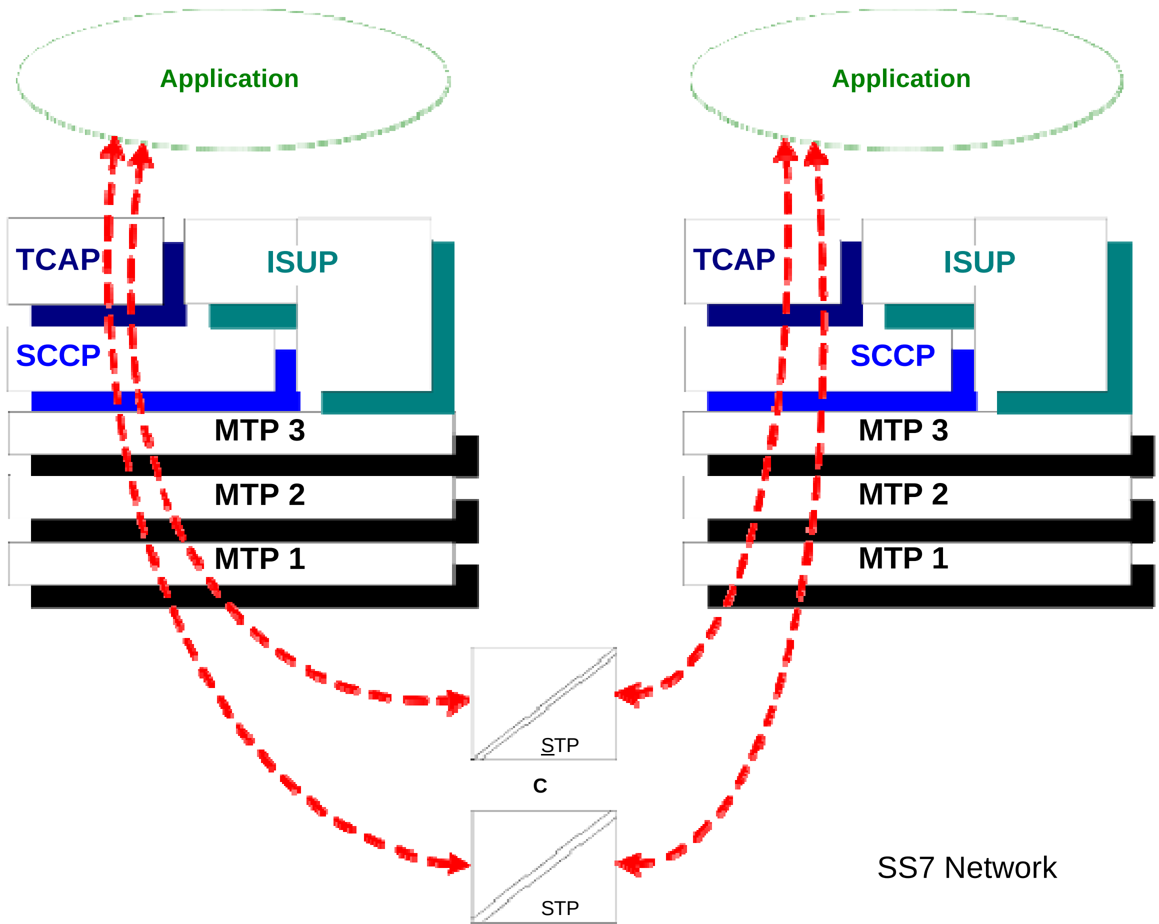


Figure 2-3: SS7 Stack Working with Applications

***Note**: Figure 2-3 does not show the message flow passing through ISUP because the example does not involve switching in a voice network.*

2.1.2 SIGTRAN Overview

The Internet Engineering Task Force (IETF) SIGTRAN MTP3 User Adaptation Layer (M3UA) draft protocol defines:

• the conversion of MTP3 messages to M3UA messages, without affecting MTP3 user signaling, such as ISUP and SCCP, and vice versa.

• the procedures for transporting M3UA messages between two end points using Streams Control Transmission Protocol (SCTP). SCTP is a reliable transport protocol operating on top of a connectionless packet network, such as the IP network.

The SIGTRAN protocol uses the concepts and terms listed in the following subsections.

2.1.2.1 Signaling Gateway (SG)

A Signaling Gateway is a signaling agent that receives and sends SCN native signaling between the SS7 and IP networks. It appears to the SS7 network as an SS7 Signaling Point and is addressable with a unique Point Code (PC). It also represents a set of nodes in the IP domain to the SS7 network for routing purposes between the two networks.

- A Signaling Gateway has one or more Signaling Gateway Processes (SGP), and usually one or more is actively processing traffic.
- A Signaling Gateway is capable of routing SS7 user traffic destined for one or more logical entities, which are referred to as Application Servers (AS) in the IP domain, or of routing traffic to another peer SG.

2.1.2.2 Signaling Gateway Process (SGP)

An SGP is a process instance of an SG. It uses M3UA and SCTP to communicate with the Application Server Process (ASP) in the IP domain, or another SGP.

2.1.2.3 Application Server (AS)

An AS is a logical entity serving a unique route key. A route key consists of a set of SS7 parameters that define a range of traffic handled by the AS. An example of an AS is a virtual switch element handling all call processing for a unique range of PSTN trunks identified by an SS7 route key combination of SIO/DPC/OPC/CIC range. An AS consists of one or more Application Server Processes (ASP), of which one or more is normally actively processing traffic.

2.1.2.4 Application Server Process (ASP)

An ASP is a process instance of an AS. It uses M3UA and SCTP to communicate with the SGP. An ASP can serve as an active or standby process in an AS. It can also serve more than one AS (i.e. route key).

2.1.2.5 SCTP Association

The SGP and ASP are viewed as signaling process end points between which an SCTP association is established. The association provides the transport for the delivery of M3UA messages between SGP and ASP.

2.1.2.6 Signaling Point Management Cluster (SPMC)

An SPMC is a complete set of ASs represented to the SS7 network under one specific SS7 Point Code of one specific Network Appearance. The Network Appearance uniquely identifies an SS7 network. See RFC 3332 for more information on the IETF SIGTRAN.

2.1.3 NewNet Communications Technologies' Signaling Gateway Suite

Signaling Gateway is an open-architecture, real-time, high availability, scalable, high-performance telecommunications application platform that provides a rapid deployment environment for service providers. Signaling Gateway is a part of NewNet Communications Technologies' Signaling Gateway Suite which includes the Signaling Gateway Client.

Together, the two products provide a complete SS7 over IP interworking solution. Both Signaling Gateway and Signaling Gateway Client are ANSI/ITU SS7 and IETF SIGTRAN compliant.

- Signaling Gateway is a gateway between the SS7 and IP networks and it extends MTP service to remote SS7 User Parts using M3UA over SCTP. It also provides SS7 offloading via IP by communicating with a peer SG.
- Signaling Gateway Client provides SS7 User Part services, such as ISUP, SCCP and TCAP, to IP-based applications such as the Media Gateway Controller (MGC) or an IP-based Home Location Register (HLR). The Signaling Gateway Client solution offers user applications in the IP domain the ability to use SS7 services without having to focus on protocol conversion. Please see the *Signaling Gateway Client™ User Manual* for more information to develop your company's applications.

Although part of the Suite, Signaling Gateway can also be deployed without the Signaling Gateway Client. It is designed to integrate with an SS7 network that is ANSI/ITU compliant, and with any ASPs that are IETF SIGTRAN compliant.

The following illustrates the Signaling Gateway Suite's internetworking solution for communicating between the SS7 and IP networks:

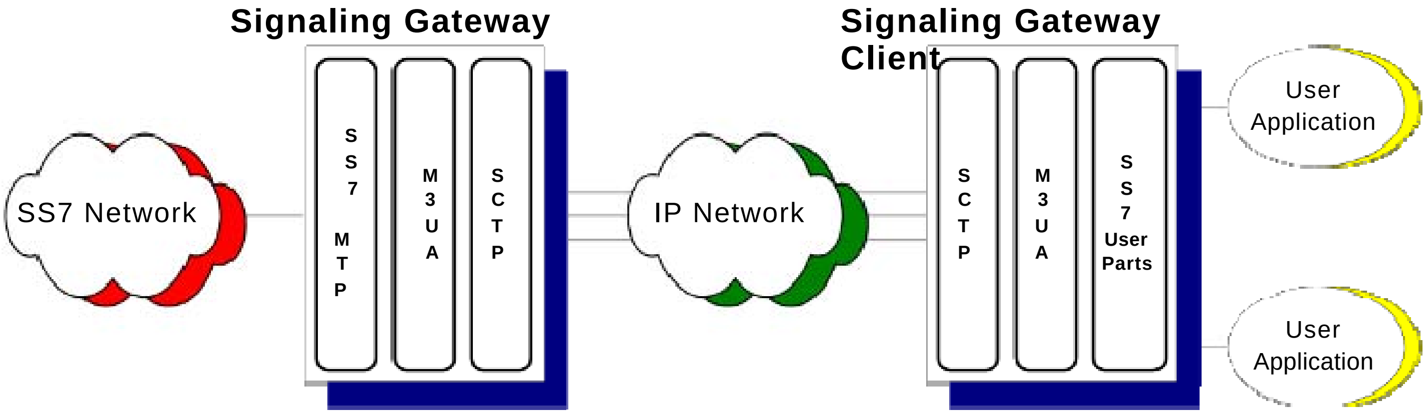


Figure 2-4: NewNet Communications Technologies' Signaling Gateway Suite

2.2 Signaling Gateway Interfaces

Signaling Gateway runs on a distributed platform that supports up to eight hosts, which form a cluster. Figure 2-5 illustrates the four interfaces of Signaling Gateway:

- SS7 network interface (A)
- SCTP network interface (B)
- Cluster LAN interface (C)
- OAM interface (D)

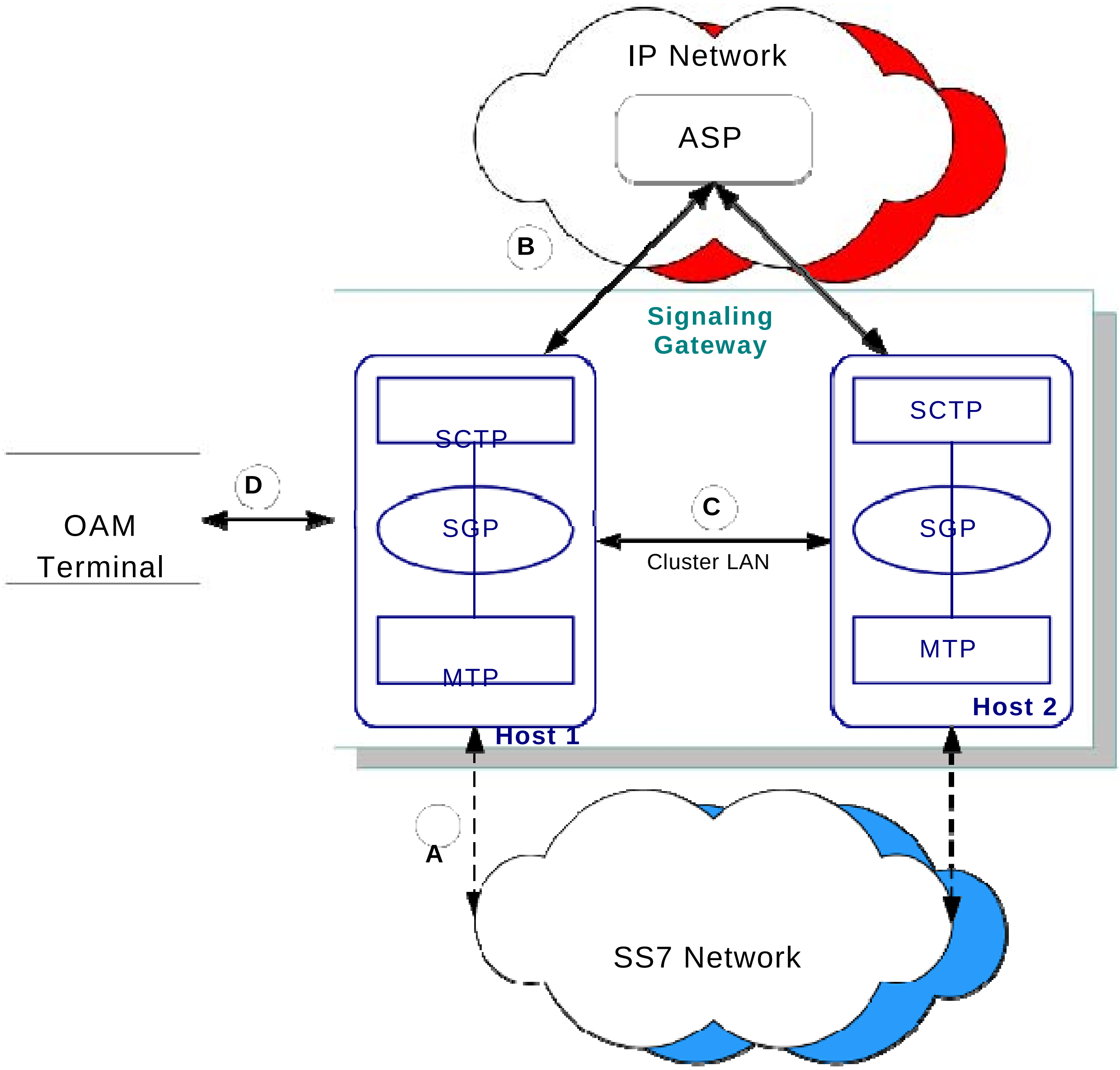


Figure 2-5: Signaling Gateway Interfaces

2.2.1 SS7 Interface

Depending on the configuration, SG connects to the SS7 network using A Links or F Links. The SS7 links can be distributed across the hosts in the signaling gateway cluster.

SG is an addressable physical node in the SS7 network, and it allows representation of up to eight signaling points with unique SPCs in the SS7 network. MTP layers and the ASPs run the MTP user part layers (ISUP, SCCP, TCAP). The communication between the user part layers is done using M3UA over SCTP (see Figure 2-6). Signaling Gateway can also be configured to run the SCCP layer to provide Global Title Translation service.

2.2.2 SCTP Interface

The SG communicates with the ASPs through SCTP. SCTP provides reliable packet transport over IP. IPsec, which is supported by the Solaris OS, may be used for enhanced security.

With the multihoming support SCTP provides, an SGP and an ASP can communicate over an SCTP association using more than one IP address. Any failure of an IP address causes a fail-over to alternative addresses. Each host can have multiple IP addresses specified for this purpose.

2.2.3 OAM Interface

The SG provides OAM interfaces for system configuration and administration. Protocol parameters and application attributes are modeled as Managed Objects (MO) and can be managed through the following interfaces:

- Man Machine Interface (MMI) - a text-based interface based on the standard ITU Z-315 MML syntax. OAM functions can be performed from a system console, such as a dumb terminal, connected to the serial port of the host, or from a remote virtual terminal connection such as a telnet or rlogin session.
- Simple Network Management Protocol (SNMP) versions 1 and 2 - OAM functions can be performed from an SNMP management console.
- AccessMOB - A GUI tool that enables configuration of the MOs.

In addition, there are several utilities and scripts provided for system startup, shutdown, software upgrade, tracing, etc.

2.2.4 Cluster LAN Interface

The distributed hosts in the cluster are connected to each other through a private LAN based on TCP/IP. The distributed hosts exchange data related to service distribution, database synchronization and management messages through the cluster interface.

For added redundancy, the cluster interface is made up of a dual LAN network, in which each host has access to both LANs through separate Ethernet interfaces.

It is recommended that the cluster LAN be separated from the SCTP interface or any other network to be sure that the performance and integrity of the distributed platform is not affected.

2.3 Signaling Gateway Features and Capabilities

2.3.1 SS7-IP Interworking

Signaling Gateway (SG) provides an SS7-IP interworking solution based on IETF SIGTRAN framework.

MTP user traffic received at the SG is packaged into M3UA messages in the Nodal Interworking Function (NIF) of an SGP. The M3UA messages are then delivered to the ASP in a remote IP node using SCTP over IP. On the IP side, M3UA messages are translated back to MTP user traffic and delivered to the user application. Similarly, MTP user traffic from an IP application is delivered to an SS7 destination through the same conversion process.

With this seamless interworking solution, an IP application can use MTP services from a remote SG as if they were from its local stack.

The following figure illustrates the SS7-IP interworking solution provided by SG when it is interworking with an ASP.

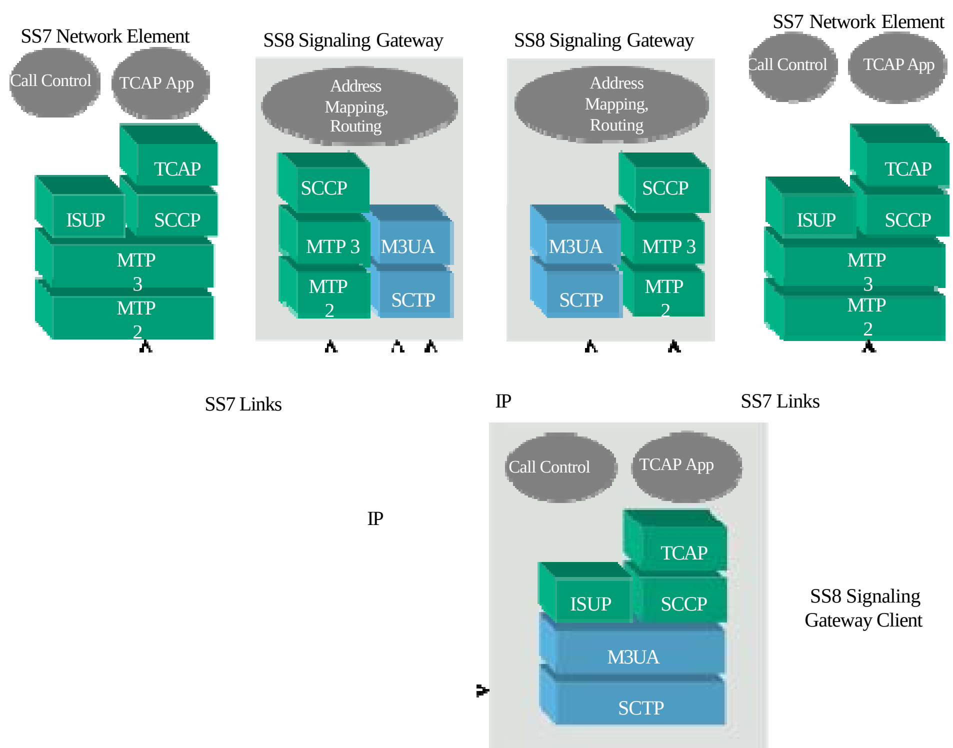


Figure 2-6: SS7-IP Interworking

2.3.2 Message Routing

2.3.2.1 SS7 to IP

SG distributes incoming SS7 messages to peer SGs, based on the DPC field in the routing label. The routing table is a list of route keys where each route key maps to a unique AS. A route key can be of one of the following combinations:

- DPC
- DPC-OPC
- DPC-OPC-CIC range
- DPC-SIO
- DPC-CIC-SIO
- DPC-OPC-SIO
- DPC-SSN

***Note**: DPC-SSN routing keys have limited support:*

- *Connection-oriented SCCP messages are not supported. A Route Key match fails when a connection-oriented SCCP message is received and the message is discarded.*
- *SCCP management messages have the reserved SSN value of 1 and fail to match a DPC-SSN route key, even though they are connectionless. The workaround for this is to define a second route key and AS to route the SCCP management messages to the correct destinations. See page 5-146 for more information.*

The routing table is provisioned by the administrator through the OAM interface.

The SG analyses the incoming SS7 message from the SS7 network, extracts the necessary information from the message unit, and searches the internal routing key database.

- The SG routes the message to the matching AS if there is a match.
- If no AS match is found, the SG tries to find a peer SG to route the message based on the DPC field.
- The message is discarded if there is no match.

An AS contains a set of ASPs processing traffic defined by the routing key at the SG. These ASPs may work in *override* or *loadsharing* mode. The mode of operation is configurable at the SG. Depending on the mode of operation, configuration changes, failover mechanisms and availability status of the ASPs, the number of active and inactive ASPs in an AS changes dynamically. The SG monitors the status of all ASPs and performs the management tasks as described in the M3UA specification.

- When ASPs are working in override mode, the SG always directs the SS7 traffic to the active ASP.
- When ASPs are working in loadsharing mode, the SG distributes the traffic to the actively available ASPs. The distribution is done according to an algorithm based on the SLS value. Hence, the ordered delivery of the messages to the ASPs is guaranteed. Also while forwarding messages to peer SGs, override and loadsharing modes are supported.

The following figure illustrates a network topology of SG and ASPs. In this scenario, SG is connected to ASP1, ASP2 and ASP3. Both ASP1 and ASP2 serve AS1 and are running in the active/standby mode, while ASP2 and ASP3 serve AS2 and are running in the load sharing mode. The traffic destined to AS1 is routed to ASP1 (active ASP), whereas the traffic destined to AS2 is distributed to both ASP2 and ASP3.

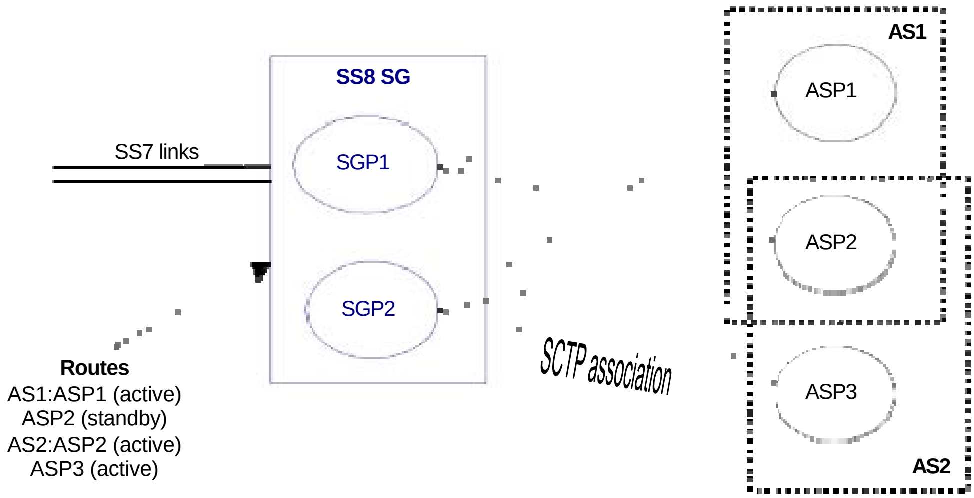


Figure 2-7: SG, ASP and AS Relationship

2.3.2.2 IP to SS7

The messages sent by the ASPs are received at the SGP. Depending on the SG configuration there may be one or more active SGP instances to process the traffic. In the standalone configuration there is only one active SGP, whereas in the distributed configuration there may be multiple SGPs, which are equally capable of processing the incoming traffic. SGPs convert the M3UA messages back to MTP3 messages, and forward them to the MTP3 layer, which sends the messages out on the appropriate signaling link. MTP3 is capable of rerouting the message to the other redundant SG host where a link may be available if there is no link available on a host.

2.3.3 Running as SEP and Gateway STP

SG can be configured to run as a Signaling End Point (SEP) or as a gateway Signaling Transfer Point (STP).

2.3.3.1 SEP

When configured as an SEP, SG shares its point code with a set of ASs in the IP domain. The SG and ASs together form a Signaling Point Management Cluster (SPMC) both using the same point code.

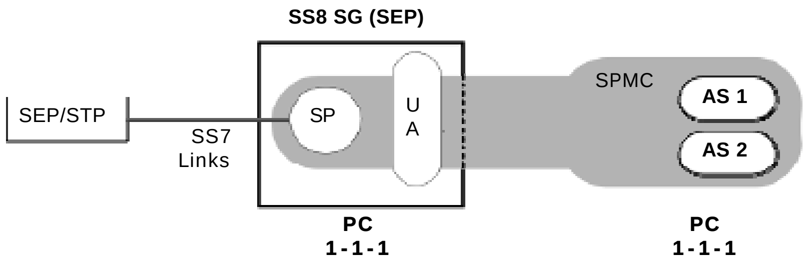


Figure 2-8: SG as an SEP

2.3.3.2 STP

When configured as an STP gateway, SG does not share its point code with any AS and it does not belong to the SPMC. The SPMC point codes are viewed as *gateway capability point codes* of the SG. SG is capable of routing MTP messages with Destination Point Codes (DPC) that are different than its own, operating just like a normal STP.

STP support offers more flexibility and redundancy in routing. It also responds more gracefully to the failure of any SPMC - it is capable of informing the SS7 network of the statuses of the signaling points represented in the IP domain. For example:

- When the SPMC is congested, it sends MTP Transfer Controlled (TFC) messages to the relevant nodes in the SS7 network.
- When the SPMC is inaccessible, which happens when all ASPs are unavailable, it sends MTP Transfer Prohibited (TFP) messages to the relevant nodes in the SS7 network.
- When an unavailable SPMC becomes accessible again, it sends MTP Transfer Allowed (TFA) messages to the relevant nodes in the SS7 network.

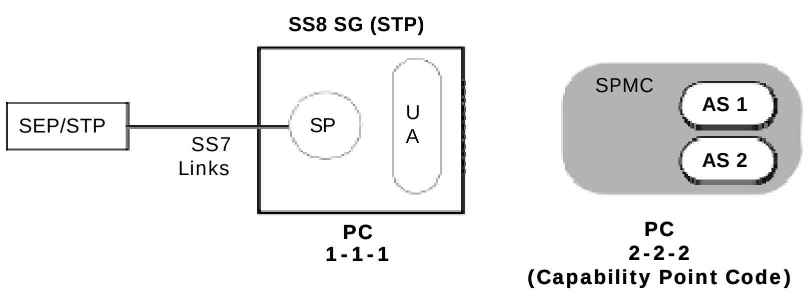


Figure 2-9: SG as an STP

2.3.4 SCCP Services and Global Title Translation Support

It is possible to run the SCCP layer to provide SCCP services, such as Global Title Translation (GTT). The following illustrates a typical configuration supporting GTT:

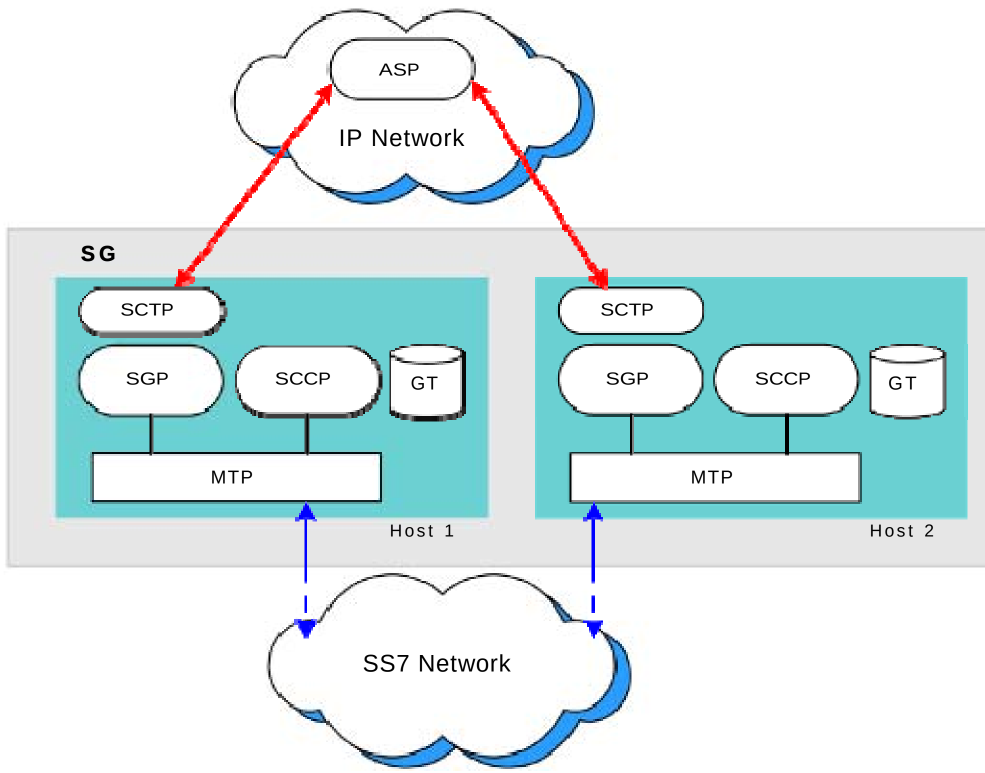


Figure 2-10: SG with GTT Service

In this configuration SG is configured as an STP on the SP where the SCCP layer runs. The typical use of this configuration is as follows:

- GT queries originate from the SS7 network and are processed by the local SCCP.
  - If the query results in an address in IP domain, then the message is forwarded to the relevant ASP through SGP.
  - If the query results in an address in SS7 domain, then the message is forwarded to the relevant SS7 node.
  - If the query results in failure, then the SCCP procedures are followed.
- It is also possible for an IP node to initiate GT queries. In this case the query arrives at the local SCCP from an ASP.
  - If the query results in an address in the IP domain, then the message is routed back to the relevant IP node.
  - If the query results in an address in the SS7 domain, then the message is forwarded to the relevant SS7 node.
  - Otherwise a response is sent back to the originating IP node per SCCP procedures.

2.3.5 Reliability

The two most important aspects of reliability are high availability and fault tolerance.

2.3.5.1 High Availability

Signaling Gateway offers high availability by distributing traffic and configuration data over multiple hosts:

- MTP distribution - MTP service is distributed across all hosts. When MTP service is unavailable in one of the hosts, traffic is automatically rerouted to the other host.
- SGP distribution - Signaling Gateway runs one SGP process on each host. When an SGP fails, then remaining SGPs running on other hosts are capable of taking over traffic from SS7's MTP and SCTP
- Configuration distribution - all configuration databases are replicated and synchronized across all hosts, and are accessible from any host in the cluster through the OAM interface.

2.3.5.2 Fault Tolerance

Signaling Gateway is fault tolerant in the following ways:

- Redundant LAN Cluster - Dual LAN interface is provided for cluster communication to increase redundancy. Communication is possible through the redundant LAN should any LAN fail.
- SCTP Multihoming - SCTP supports multihoming where multiple IP addresses can be defined for each SGP-ASP association. The redundant interface prevents single-point-of-failure should any IP interface fail.
- Redundant SGP and internal message routing among SGPs - when an SGP loses its connection to the ASP, messages are re-routed to the other SGPs. This avoids a single-point-of failure should any SGP lose its SCTP connection or the connection is terminated.

2.3.6 Fault Management

Signaling Gateway has the following fault management functions:

- Application Process Management (APM) - the APM feature provides automatic coordination of platform startup and shutdown. APM also provides continuous process monitoring such that whenever a process fails, it is restarted automatically and instantaneously without human intervention.
- Automatic startup - automatic application startup can be enabled so that all application processes start automatically following a system reboot. This is useful when a power failure occurs.
- Alarm reporting - all components of Signaling Gateway have built-in fault detection. An alarm is generated and the faulty event is logged whenever a software, operating system or network failure is detected. Alarm reporting can be configured so that the alarm message is sent either to the console, log file, or as an SNMP trap to a management station.

2.3.7 Troubleshooting Tools

Signaling Gateway has alarm reporting, daily event log file and runtime tracing utility for troubleshooting.

2.3.8 Security

Signaling Gateway has the following security features:

- **User Access** - restricted system access is provided through the UNIX password-protected user account. Only the administrator or the *superuser* has the privileges to administer and maintain the system.
- **Transport Security** - the use of SCTP allows the support of transport related security features such as protection against Blind Denial of Service Attacks, flooding or blind masquerade.
- **ASP Access** - Connection requests from an ASP will not be accepted unless its IP address or hostname is configured. This prevents unauthorized ASP access to the Signaling Gateway.

When the network involves more than one party, it is recommended to use IPsec to ensure confidentiality of user payload.

2.4 SoftwareComponents

The Signaling Gateway system has the following software components:

- NewNet Distributed7 - a distributed, open-architecture, high-performance, real-time and scalable SS7 application development platform.
- Signaling Gateway Core - an IETF SIGTRAN-based signaling gateway application that provides M3UA and SCTP services for transporting SS7 user traffic between the SS7 and IP networks.

Signaling Gateway run on the Sun Solaris 8/9/10 operating system. The following illustrates the high-level architecture:

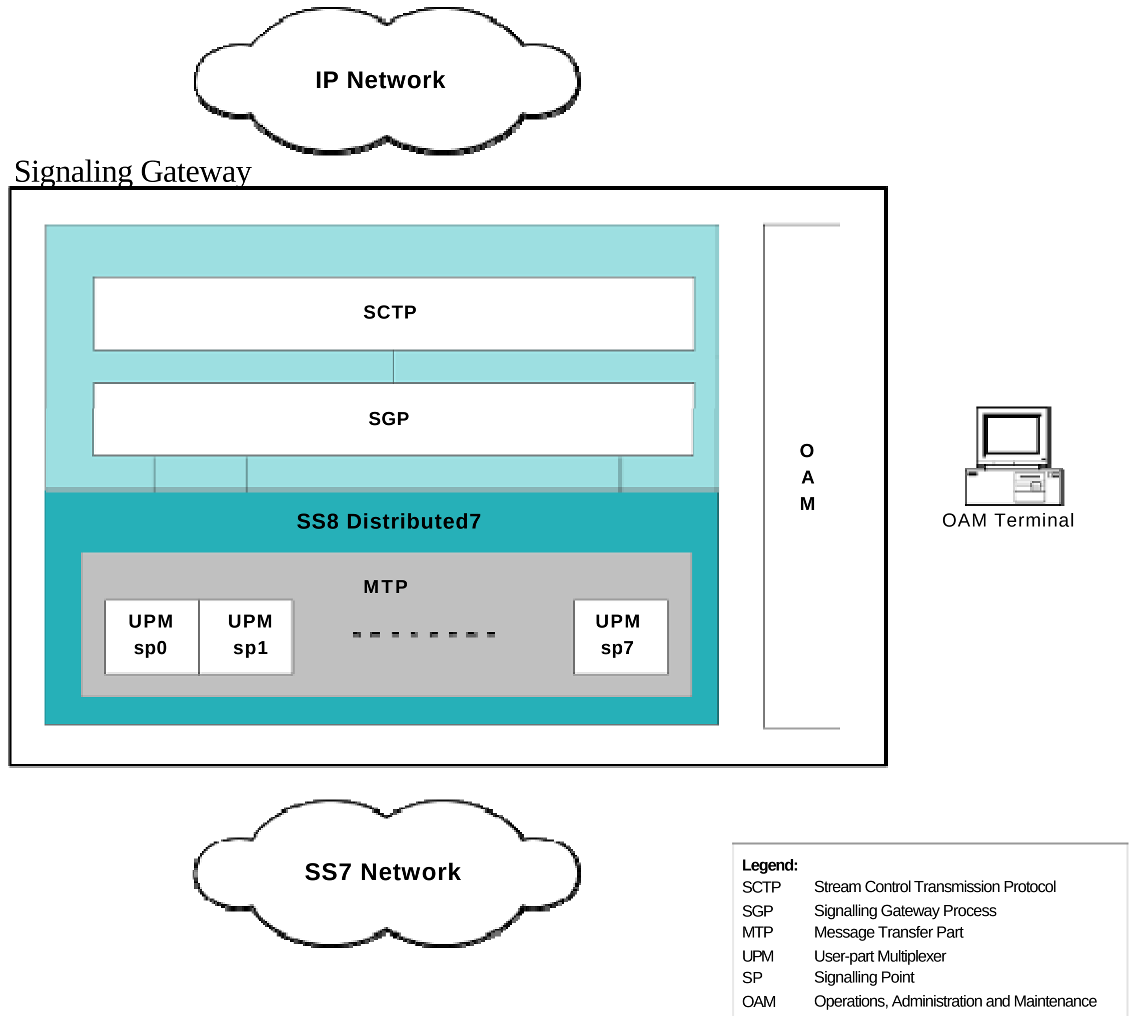


Figure 2-11: Signaling Gateway Software Architecture

2.5 Capacity

Signaling Gateway supports a distributed, load sharing platform over a maximum of eight hosts. The following table lists the various capacity factors:.

Table 2-1: Capacity

	Description	ANSI	ITU
MTP	destination point codes (SS7 + capability)	2048	2048
	destinations behind link sets	2048	2048
	links	511	511
	link sets	64	64
	routes per destination	16	16
SCCP	destination point codes	8192	8192
	subsystems per destination	256	256
	concerned point codes per subsystem	8192	8192
	global title types	16	16
	translation types per global title type	256	256
	simultaneous open SCCP connections	16384	16384
	simultaneous reassembly processes per system	16	16
Maximum number of cluster hosts		8	8
Maximum number of IP addresses per host for SCTP interface		2	2
Maximum number of AS		255	255

2.6 ComputingPlatform

Signaling Gateway software runs on the Sun Microsystem's SPARC and Intel/AMD's x86 architectures and Solaris 8/9/10 operating system. It is recommended that Signaling Gateway be deployed on telco grade NEBS level-3 compliant rack mountable system, such as the Sun Netra or Sun Enterprise series.

Caution *See Appendix B for information about the hardware installation and configuration that must be done before installing the software.*

2.6.1 Host Platform Options

For information regarding Host Platform Options, refer to the SG/SGC Release Notes.

2.6.2 SS7 Controller Options

Table 2-2: Available SS7 Controller Options

Bus Architecture	Physical Interface	Ports per Controller for Links up to 64 kbps	Ports per Controller for High Speed Links	Numberof Controllers per System[1]
Sbus	RS-449	Up to 4		8
	V.35	Up to 4		
	T1	Up to 4		
	E1	Up to 4		
PCIbus	RS-449	Up to 4		
	V.35	Up to 4		
	T1	Up to 4[2] Up to 24[3,8] Up to 64[9]	Up to 4[10]	4
	E1	Up to 4[4] Up to 24[5,8] Up to 64[9]		
CompactPCI bus	T1	Up to 24[6,8] Up to 64[9]		8
	E1	Up to 24[7,8] Up to 64[9]		

1. Also limited by the number of available slots on the bus.

2. Available with PCI370 board.

3. Available with PCI370PQ and PCI370APQ boards.

4. Available with PCI372 board.

5. Available with PCI372PQ and PCI372APQ boards.

6. Available with CPC370PQ board.

7. Available with CPC372PQ board.

8. Although PCI3xPQ, PCI3xAPQ and CPC3xPQ boards allow configuration of up to 24 links, use of more than 16 with the PCI3xPQ card is not recommended for systems requiring full bandwidth on all configured links.

9. Available with PMC8260 and ARTIC1000/2000 boards.

10. Available with the PMC4539F board.

Figure 2-12 is a sample hardware setup for Signaling Gateway deployment. For redundancy's purpose, it is recommended that each host be equipped with two Ethernet interfaces for the cluster dual LAN, and two Ethernet interfaces for SCTP access. Each pair of redundant Ethernet interfaces should connect to different Ethernet hubs on different subnets to prevent single-point-of-failures. Similarly, the host where SS7 links are present should be equipped with at least two SS7 interface cards to increase the reliability.

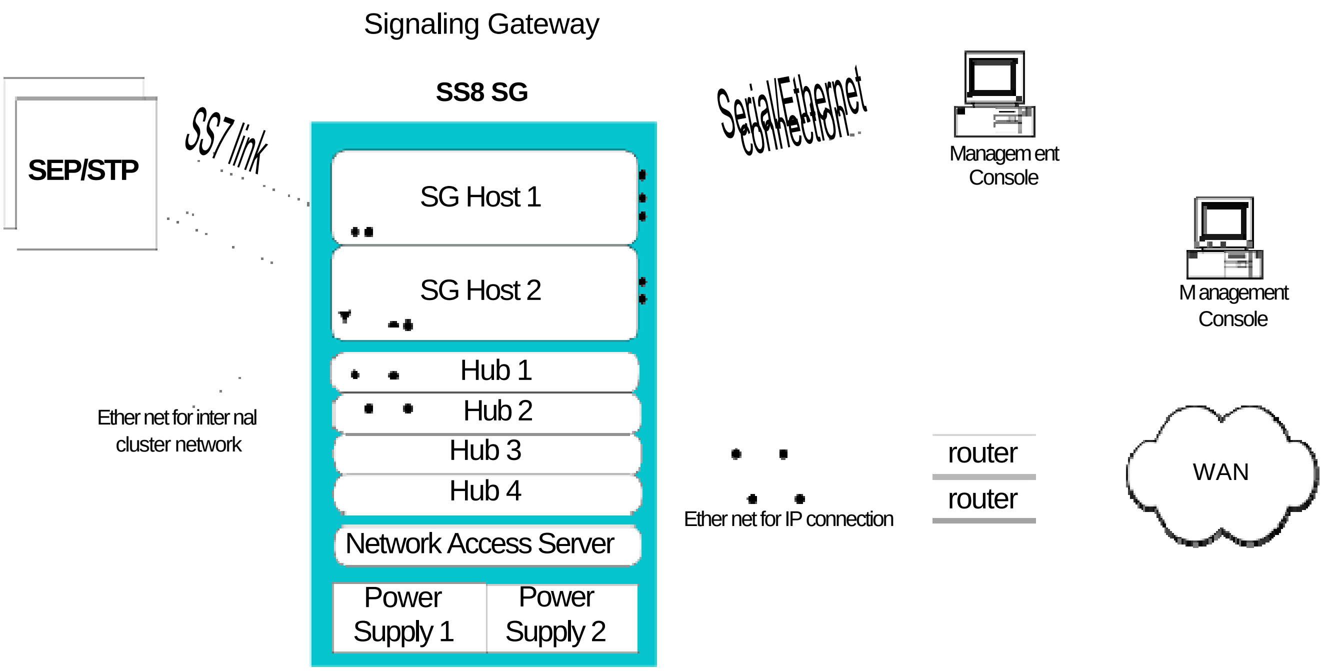


Figure 2-12: Signaling Gateway Hardware Layout

This page is intentionally blank.

Chapter 3: Installation

3.1 Overview

This chapter describes installation of the Signaling Gateway software on a Sun platform. Signaling Gateway uses NewNet Distributed7 for its SS7 MTP and SCCP functions, and requires that distributed NewNet Distributed7 packages be installed first.

Important: *You need a license file to run the NewNet Communications Technologies software package. This license file is already on hardware systems purchased and staged by NewNet Communications Technologies and delivered with the software already installed. You must have the following information to obtain the license file from the NewNet Communications Technologies' Technical Assistance Center (TAC):*

- *License number from the label on the CD and/or installation tape.*
- *Host ID of your machine. Enter the following in the Solaris command window to find out the host id:*

  ```
  /usr/sbin/sysdef -h
  ```

- *Call TAC at (877) 698-5583 US, (203) 647-0580 International, or email the information to support@newnet.com.*
- *The license file is then emailed to you or put up on the FTP site for you to download.*

The key file must be copied into the ***$EBSHOME/access/etc*** *directory with the file name of* ***license.dat*** *(see Step 10 on page 3-10).*

3.2 Software Installation

The Signaling Gateway software is installed from either a CD ROM or is downloaded from the FTP site. The software contains object files, include files, sample programs, libraries, utility files, and executables.

The steps in this section have the instructions to install Signaling Gateway using a *packaged* format. The packages, or filesets, are dependent on other packages as shown in Table 3-1. Packages must be installed in order of their dependencies, and removed in reverse order. For instance, *D7core1* is the first package to be installed and the last one to be removed.

Multiple versions of the software can be installed on a system.

Correct operations of the software may require manipulation of one or more of the following configuration files on all involved hosts. These files are used to locate a variety of databases associated with host names and network services. Information about a specific entry may come from a number of sources, such as UNIX files, NIS, NIS+.

- **`/etc/nsswitch.files`** This configuration file instructs the network services library functions to rely on the information specified in appropriate UNIX files only.
- **`/etc/nsswitch.nis`** This configuration file instructs the network services library functions to rely on the information specified in appropriate UNIX files and the NIS database.
- **`/etc/nsswitch.nisplus`** This configuration file instructs the network services library function to rely on the information specified in appropriate UNIX files and the NIS+ database.

Important: *Users must be aware that the* ***netdbase*** *service type, which relies on the use of connection-oriented TCP/IP protocol, has been defined in the* ***/etc/services*** *file on the system and correct operations of the Signaling Gateway software require that the* ***netd*** *daemon can invoke the* ***netdir_getbyname()*** *function to retrieve information about this service type. Therefore, it is up to the user to manipulate the sample configuration files listed previously to be sure that it is possible to retrieve this information from the* ***/etc/services*** *file. Failure to do so results in the* ***netd*** *daemon terminating its execution prematurely with the reason for termination listed in the master log file.*

Important: *By default, NewNet Distributed7 software expects the official host name of the machine to be used. Customers interested in using alias names are required to run the* ***ebs_config*** *script following initial software installation. This script prompts the customer for the host name, and the host name that the customer specifies there is saved in the* ***/etc/amgrhost*** *file. From then on, when NewNet Distributed7 system software is started, the host name listed in the* ***/etc/amgrhost*** *file is used. This allows customers to configure the NewNet Distributed7 product to make use of alias names for their host machines, and therefore provides flexibility in product configuration, e.g., when a particular host machine is part of multiple public/private networks.*

Example

Incorportate the following lines into the `/etc/nsswitch.conf` file that has ***files*** as the first entry for the hosts and services entries:

```
hosts: files dns nisplus [NOTFOUND=return] files
services: files nisplus [NOTFOUND=return] files
```

***Important**: For more information about the format of the entries listed in the file, and how each entry is interpreted, please see the online manual pages for the **nsswitch.conf** configuration file.*

3.2.1 Disk Space Requirements

Table 3-1 lists the disk space requirements for each Signaling Gateway software package, and the total disk space required:

Table 3-1: Required Packages for Signaling Gateway

Description	Package Name	Disk Space (KB)	Prerequisite
Core	D7core1	26,808	-
Application Process Management	D7core2	13,408	D7core1
Distributed Memory Management	D7core3	10,672	D7core1, 2
Alarm Management	D7core4	2,384	D7core1, 2, 3
Graphical User Interface	D7core5	21,456	D7core1-4
Measurements Collection (OMAP)	D7core6	1,616	D7core1-5, D7ss7p1
MTP	D7ss7p1	28,648	all D7core packages
SCCP	D7ss7p3	6,744	all D7core packages, D7ss7p1
Signaling Gateway Core	SGcore	14,000	All of the above
Total Disk Space Required		125,736	

3.2.2 Installation Steps

This section has the steps to install the Signaling Gateway software. Installation process has the following requirements:

- Superuser privileges are required to complete the installation.
- Multiple releases of Signaling Gateway can be installed on a single host. Use the **sg_setrelease** command to switch between different releases when there is more than one release installed on a host.
- The installation procedure should be repeated on all the hosts of the SG cluster. It is possible to install Signaling Gateway as standalone or distributed.
  - For multiple hosts "distributed" mode must be selected in the installation process.
  - For one host operation "standalone" must be selected.

***Note**: It is possible to change the Distributed/Standalone mode of operation after completing the installation. Installation directory names need not be same across all hosts in the cluster.*

- Go to Step 1 if you have the license file. The following information is required to obtain the license file from TAC, which must be copied to the host in Step 10:
  - The Serial number from the label on the installation CD or tape.
  - The Host ID of the machine.

***Note**: Enter `hostid` in the Solaris command window to get the Host ID.*

  - Give the information to the NewNet Communications Technologies Technical Assistance Center by calling (800) 416-1624 US, (408) 432-2600 International, or emailing the request to nnsupport@ss8.com. The license file is emailed to you or put up on the FTP site for you to download.

***Important**: Installation instructions are updated in the Release Notes and README file. Be sure to check these for the latest information.*

Caution *Be sure that the following Solaris 2.8 patch is installed on your system:*

108528-18 (cluster patch)

Enter the following command to find out the installed patches on your system:

/bin/showrev -a

Although it is better to install the patches before installing Signaling Gateway, it is possible to apply the patches any time AFTER installing the NewNet Communications Technologies software. The NewNet Communications Technologies software may have abnormal behavior or you may *experience fatal kernel panics/crashes if the patches are not installed!*

The patches and their installation instructions can be obtained from Sun Support (http://sunsolve.sun.com/pub-cgi/show.pl?target=home):

3.2.2.1 Installation Preparation

From a CD

1. Log in as root.
2. Create the directory where the Signaling Gateway software packages are to be installed. This is the *base directory* that is referenced in later installation steps.

   mkdir /NewNet Ret

   - These steps use NewNet as the *base directory* example, but you can create the base directory with any name.

Caution: *It is strongly recommended that the base directory be on the same host as the installation. Also there should be enough free disk space to install all the packages and to accommodate the log files which may increase substantially in size during normal operation.*

   - The startup script specifies your *base directory* entry as the $EBSHOME and $SGHOME environment variables when logged in as **sgadm**.

3. Put the CD into its drive.

Important*: You may install the software directly from the CD or copy the CD files into a directory (but* ***NOT*** *the tmp directory). This allows for a faster software installation from the hard disk, and the files are readily available to reinstall something without the CD. However, there must be enough disk space to copy all the files and install the software. Complete the following substeps to make a directory and copy the files from a CD. Go to the next step if you do not want to do this.*

   a. Enter the following to create a directory in the root of the hard disk (/SG110Bpkg in this example):

   mkdir /SG110Bpkg Ret

   cd /SG110Bpkg Ret

   b. Enter the following to copy the files from the CD to the /SG110Bpkg directory:

   cp -r *<cd_device_name>*/*** /SG110Bpkg** Ret

   where `<cd_device_name>` is the name of the CD drive being used—`/cdrom/cdrom0` for example.

4. Complete the following ***IF*** a previous version of Signaling Gateway is installed on the machine:

   a. Modify the **`/var/sadm/install/admin/default`** file contents so that the value of the **instance** field is set from ***overwrite*** to ***unique***.

This insures that an already installed software package, such as D7core1, is not overwritten when the new version is installed. It also allows different versions of a particular software package to co-exist in the same directory, such as D7core1, D7core1.2, and SGcore.2. This change does not prevent re-installation of a particular version of a software package in the same *base directory*.

***Note**: Use the* **`pkginfo`** *command to check all existing versions if there are multiple versions of a software package installed on one machine.*

5. Install the software packages either directly from the CD, or from the directory created in Step 3.a on page 3-5:

Install directly from a CD:

`pkgadd -d devname`

where **`devname`** is the device name of the CD drive being used.

OR

Install directly from source directory created in Step 3.a on page 3-5

`pkgadd -d pathname`

where **`pathname`** is the full UNIX path name of the directory to which the files were copied from the CD (/SG110Bpkg in this example).

From a Tape

1. Log in as root.
2. Create the directory where the Signaling Gateway software packages are to be installed. This is the *base directory* that is referenced in later installation steps.

mkdir /NewNet

- These steps use NewNet as the *base directory* example, but you can create the base directory with any name.

Caution: *It is strongly recommended that the base directory be on the same host as the installation. Also there should be enough free disk space to install all the packages and to accommodate the log files which may increase substantially in size during normal operation.*

- The startup script specifies your *base directory* entry as the $EBSHOME and $SGHOME environment variables when logged in as **sgadm**.

3. Put the tape into its drive. Be sure to set a tape's write protection away from *safe*.

Important**: You may install the software directly from the tape, or copy the files into a directory (but* ***NOT *the tmp directory). This allows for a faster software installation from the hard disk, and the files are readily available to reinstall something without the tape. However, there must be enough disk space to copy all the files and install the software. Complete the following substeps to make a directory and copy the files from a tape. Go to the next step if you do not want to do this.*

a. Enter the following to create a directory in the root of the hard disk (/SG110Bpkg in this example):

mkdir /SG110Bpkg Ret

cd /SG110Bpkg Ret

b. Enter the following to copy the files from the tape to the /SG110Bpkg directory:

tar xvf <*tape_device_name*> Ret

where `<tape_device_name>` is the name of the tape drive being used—*/dev/rmt/0* for example.

4. Complete the following ***IF*** a previous version of Signaling Gateway is installed on the machine:

a. Modify the **`/var/sadm/install/admin/default`** file contents so that the value of the **instance** field is set from ***overwrite*** to ***unique***.

This insures that an already installed software package, such as D7core1, is not overwritten when the new version is installed. It also allows different versions of a particular software package to co-exist in the same directory, such as D7core1, D7core1.2, and SGcore.2. This change does not prevent re-installation of a particular version of a software package in the same *base directory*.

Note*: Use the* **`pkginfo`** *command to check all existing versions if there are multiple versions of a software package installed on one machine.*

5. Install the software packages either directly from the tape, or from the directory created in Step 3.a on page 3-7:

Install directly from a tape:

`pkgadd -d devname` Ret

where **`devname`** is the device name of the tape drive being used.

OR

Install directly from source directory created in Step 3.a on page 3-5

`pkgadd -d pathname` Ret

where **`pathname`** is the full UNIX path name of the directory to which the files were copied from the tape (/SG110Bpkg in this example).

3.2.2.2 Installation

1. The system displays a menu of the Signaling Gateway packages that can be installed. Install all software packages in the order they are listed:

```
The following packages are available:
 1  D7core1     Distributed7 Core 1.3.0.2
           (sparc) 1.3.0.2
 2  D7core2     Distributed7 Process Management 1.3.0.2
           (sparc) 1.3.0.2
 3  D7core3     Distributed7 Memory Management 1.3.0.2
           (sparc) 1.3.0.2
 4  D7core4     Distributed7 Alarm Management 1.3.0.2
           (sparc) 1.3.0.2
 5  D7core5     Distributed7 GUI 1.3.0.2
           (sparc) 1.3.0.2
 6  D7core6     Distributed7 OMAP 1.3.0.2
           (sparc) 1.3.0.2
 7  D7ss7p1     Distributed7 MTP 1.3.0.2
           (sparc) 1.3.0.2
 8  D7ss7p3     Distributed7 SCCP 1.3.0.2
           (sparc) 1.3.0.2
 9  SGcore      Signaling Gateway Core 1.1.1
           (sparc) 1.1.1

Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

- Press Return after this menu to install all the packages.
- Install just the SGcore package if the same version of NewNet Distributed7 packages are already installed. Type **9** and press Return to do this.
- Use **pkginfo** command to see which versions of the packages are already installed on the system.

2. During installation, the following prompt asks you to select *distributed* or *simplex* configuration:

```
Distributed7 product can be configured to support:

+ distributed product configurations where two or more host
machines are connected to each other via a local area network.

+ simplex product configurations where the Distributed7
environment comprises isolated [stand-alone] host machines.

Which mode do you want [(1)distributed (2)simplex]?  [?]
```

If you plan to use Signaling Gateway in standalone mode select *simplex*, otherwise select *distributed*.

- Enter **1** to select a ***distributed*** system.
- Enter **2** to select a ***simplex*** (standalone) system.

Important: *Run ebs_config to change the Signaling Gateway configuration whenever you need to change the mode between* ***distributed*** *and* ***simplex****. Running the ebs_config command is part of the configuration in* *Chapter 4: Operations, Administration and Maintenance**.*

3. The following prompt for the NewNet Distributed7 base directory appears to prepare for the installation of its packages:

Enter installation directory base: [?,q] **NewNet** Ret

This example shows ***NewNet*** as the entry for the NewNet Distributed7 base directory.

4. The following prompt for the Signaling Gateway Core base directory appears to prepare for the installation of its package:

Enter Signaling Gateway Core base directory (default: /NewNet) [?,q]

Notice that the default base directory is what was entered in the prompt for the NewNet Distributed7 base directory. While a different directory can be entered, it is recommended that Signaling Gateway be installed in the same base directory as NewNet Distributed7.

Important: *The startup script automatically sets the EBSHOME and SGHOME environment variables to this base directory when logged in as* ***sgadm****. It must be set manually if logged in using a different account.*

5. During installation of the individual packages, there are several prompts for the **`setuid/setgid`** permissions of different executables. Enter **`y`** (YES) to all prompts. The following is an example of one of these prompts:

```
The following files are being installed with setuid and/or setgid
permissions:
  /NewNet/access.1.3.0.2/bin/logd <setuid root>
  /NewNet/access.1.3.0.2/bin/mml <setuid root>
  /NewNet/access.1.3.0.2/bin/netd <setuid root>
  /NewNet/access.1.3.0.2/bin/snmp_i <setuid root>
  /NewNet/access.1.3.0.2/bin/snmp_p <setuid root>
  /NewNet/access.1.3.0.2/bin/spmd <setuid root>
  /NewNet/access.1.3.0.2/install/cron.restart <setuid root>
  /NewNet/access.1.3.0.2/install/ebs_modunload <setuid root>

Do you want to install these as setuid/setgid files [y,n,?,q]
```

6. Enter the password for **sgadm** when prompted to do so. Enter it a second time to confirm that it was entered correctly.

```
Please supply a password for sgadm.
New password:
Re-enter new password:
passwd (SYSTEM): passwd successfully changed for sgadm
```

7. The system returns to the `pkgadd` menu after all Signaling Gateway software packages are installed. Quit this menu.

8. Enter the following to switch to the **sgadm** account:

`su - sgadm`

9. Go to Step 10 if this is the first time that Signaling Gateway have been installed on the machine—or if there is no need to activate the release just installed. Otherwise complete the following to activate the releases:

sg_setrelease <*sgrlsnum*>

where *sgrlsnum* is the activated Signaling Gateway release number, e.g., 1.1.1.

Run the **sg_setrelease** command again to change the activated release(s) whenever needed.

10. Enter the following to save the license file to ***$EBSHOME/access/etc*** as ***license.dat***:

cp <licensefile> $EBSHOME/access/etc/license.dat

11. Enter the following to adjust the system parameters required to run the Signaling Gateway software:

ebs_tune

12. Enter the following to reboot the system:

sync

reboot

The Signaling Gateway software is installed and ready to configure. Please review the files listed in the following sections before continuing.

NewNet Distributed7

The following NewNet Distributed7 files are located in the ***$EBSHOME/access*** directory:

- **README** - contains an outline of the NewNet Distributed7 directories and the files in them
- **BUGS** - lists the known deficiencies in the release

Signaling Gateway Core

The following Signaling Gateway files are located in the ***$SGHOME/sg*** directory:

- **README** - contains an outline of the Signaling Gateway Core directories and the files in them
- **BUGS** - lists the known deficiencies in the release

3.2.3 Live Upgrade

This section describes user considerations and procedures when installing and/or activating a new release of the Signaling Gateway software on one or more hosts in a cluster while the rest of the hosts in the cluster continue to run on an earlier release. This live upgrade capability allows the user to validate the behavior of the new software before upgrading all the machines in the cluster. Upgrades include from patch level, that is 1.0.0_B to 1.0.0.1, to full version installs, such as 1.0.0. to 1.1.0.

***Important**: The Signaling Gateway upgrade scripts include **ALL** the necessary upgrades for NewNet Distributed7 packages.*

3.2.3.1 User Considerations

The following list describes system operations and user choices at the time of installation of the new release of the Signaling Gateway software:

- **`pkgadd`** results in the full installation/activation of the Signaling Gateway software the very first time they are installed. No additional action is necessary.
- If a version of Signaling Gateway is already installed on the target machine, **`pkgadd`** results in a passive installation of the new release. The release that is already installed on that machine is not removed or deactivated. In this case, users must run the **`sg_setrelease`** command to activate the newly installed release. The **`sg_setrelease`** command allows users to switch between different Signaling Gateway releases installed on the same machine.
- The following sections apply to multiple versions of either software package installed on one machine.

NewNet Distributed7 Packages

- When multiple versions of NewNet Distributed7 packages are installed on one machine, an *access* tree is created under the base product installation directory for each new version, such as **`access.1.0.0_B`**, **`access.1.0.0.1`**. To make it easy to switch between the different versions of NewNet Distributed7, the **`$EBSHOME/access`** entry is maintained as a symbolic link to the NewNet Distributed7 version that is currently in use.
- When multiple versions of NewNet Distributed7 packages co-exist:
  - The **pkginfo** command lists all such packages. Users should issue the **ebs_setrelease -i** command to find out which NewNet Distributed7 version is currently running.
  - It is possible to remove obsolete versions using the **`pkgrm`** command. Alternately, the **ebs_pkgrm** command removes all software packages associated with a particular NewNet Distributed7 version.

***Caution** SGcore package **must** be removed BEFORE **NewNet Distributed7** packages are removed!*

Signaling Gateway

- When multiple versions of Signaling Gateway Core (SGcore) software are installed on one machine, an *sg* tree is created under the base product installation directory for each new version, such as `sg.1.0.0_B`, `sg.1.0.0`. To make it easy to switch between the different versions of SGcore, the `$SGHOME/sg` entry is maintained as a symbolic link to the version that is currently in use.
- When multiple versions of an SGcore software package co-exist:
  - The **pkginfo** command lists all such packages. Users should run the **sg_setrelease -i** command to find out which NewNet Distributed7 version is required by the active Signaling Gateway Core software.
  - It is possible to remove obsolete versions using the `sg_pkgrm` command. Alternately, the **sg_pkgrm** command removes all software packages associated with a particular Signaling Gateway version.

Caution: *SGcore package **must** be removed BEFORE **NewNet Distributed7** packages are removed!*

3.2.3.2 Live Upgrade Steps

The following sections describe the two possible scenarios for a live upgrade of the software—both NewNet Distributed7 and Signaling Gateway Core are upgraded at the same time, or only Signaling Gateway Core is upgraded.

Upgrading NewNet Distributed7 and Signaling Gateway Core at the Same Time

The following is an example of upgrading both NewNet Distributed7 and Signaling Gateway Core:

	NewNet Distributed7	Signaling Gateway Core
Existing Release		
New Release		

- All packages must be installed in this scenario, following the Installation Steps on page 3-4.
- Then run **sg_setrelease** to upgrade the Signaling Gateway Core files.

Upgrading Only Signaling Gateway Core

The following is an example of upgrading just Signaling Gateway Core:

	NewNet Distributed7	Signaling Gateway Core
Existing Release		
New Release		

- Only the SGcore packages should be installed in this scenario.
- Use the `pkgadd` command to install **SGcore**.
- Run **sg_setrelease** to upgrade the Signaling Gateway files.

Caution *DO **NOT** run the **ebs_setrelease** command unless it is required to fix problems. Always use the **sg_setrelease** command to upgrade Signaling Gateway.*

Related Information

- See Chapter 4: Operations, Administration and Maintenance for information about the startup and configuration of your system.
- Table 4-1, Directories and Files, on page 4-2 to see the directory structure that exists after successfully installing the software

Related Commands

- ebs_config on page 7-8
- ebs_setrelease on page 7-27
- sg_setrelease on page 7-70

3.2.4 Software Removal

The software must be removed by reversing the order in which it was installed—Signaling Gateway Core must be removed BEFORE NewNet Distributed7 packages are removed.

Caution: Before software is removed, make sure the version to be removed is NOT running!

Complete the following steps to remove the Signaling Gateway Core software package:

1. Login as **sgadm**.
2. Determine the software version to be removed. Run the following command if you are not sure of the version:

   `sg_setrelease -i`

3. Run the following command to remove the software:

   `sg_pkgrm` *`version`*

   where *version* is the software version number.

Complete the following steps to remove the NewNet Distributed7 software packages:

1. Login as **sgadm**.
2. Determine the software version to be removed. Run the following command if you are not sure of the version:

   `ebs_setrelease -i`

3. Run the following command to remove the software:

   `ebs_pkgrm version`

   where *version* is the software version number.

The **ebs_pkgrm** command automatically removes ALL NewNet Distributed7 packages listed in Table 3-1 in proper order.

Chapter 4: Operations, Administration and Maintenance

4.1 Overview

This chapter describes the operations, administration and maintenance of the Signaling Gateway system. The following areas are covered:

- operating environment of the platform
- general operations
- getting started with a new system
- configuration

4.2 OperatingEnvironment

This section describes the software platform and environment in which the Signaling Gateway operates.

4.2.1 User Account

The **sgadm** UNIX account is created for software administration purposes during installation. The **sgadm** account has privileges to start, stop and provision the Signaling Gateway system.

A set of environment variables required for starting the software are set in the default `.cshrc` file of the **sgadm** account. Always switch the user to **sgadm**, using the following command, to be sure that you are working with the proper environment settings:

```
su - sgadm
```

***Note**: This user account has root, or superuser privileges, and must be used carefully.*

4.2.2 Directory Structure and Files

All NewNet Distributed7 software files are located under the ***$EBSHOME/access.X*** directory, and Signaling Gateway Core software files are located under the **$*SGHOME/ sg.X*** directory, where **X** indicates the software version number.

The following table shows the directory structure of the Signaling Gateway software:

Table 4-1: Directories and Files

Software Package	Software Directories	Contents
NewNet Distributed7	$EBSHOME/access/bin	Executable files
	$EBSHOME/access/demo	Demo applications
	$EBSHOME/access/drv	Device drivers
	$EBSHOME/access/etc	License and miscellaneous data files
	$EBSHOME/access/gui	Command File Navigator files
	$EBSHOME/access/help	Help text for Managed Object browser
	$EBSHOME/access/include	Header files for NewNet Distributed7 API
	$EBSHOME/access/install	Scripts related to installation
	$EBSHOME/access/lib	Libraries for NewNet Distributed7 API
	$EBSHOME/access/manpages	Man pages
	$EBSHOME/access/sample	Sample programs using NewNet Distributed7 API
	$EBSHOME/access/RUN/DBfiles	Non-SS7 Managed Object database files
	$EBSHOME/access/RUN/alarmlog	Alarm log files
	$EBSHOME/access/RUN/alog	Alternative log files
	$EBSHOME/access/RUN/mlog	Master log files
	$EBSHOME/access/RUN/config/ALARM	Alarm configuration files
	$EBSHOME/access/RUN/config/MML	MML configuration files
	$EBSHOME/access/RUN/config/PMGR	Application Process Management configuration files
	$EBSHOME/access/RUN/config/SNMP	SNMP MIBs and configuration templates
	$EBSHOME/access/RUNx/DBfiles	Managed Object databases for signaling point x
	$EBSHOME/access/RUNx/alog	Alternative log files for signaling point x
	$EBSHOME/access/RUNx/mlog	Master log files for signaling point x
	$EBSHOME/access/RUNx/backup/MML	MML command log file (contains detailed execution history)
	$EBSHOME/access/RUNx/config/MML	MML command history file
	$EBSHOME/access/RUNx/config/SNMP	SNMP Agent configuration for signalling point x

Table 4-1: Directories and Files (Continued)

Software Package	Software Directories	Contents
ServiceController 1510 Core	$SGHOME/sg/bin	Executable files
	$SGHOME/sg/etc	Miscellaneous files
	$SGHOME/sg/install	Installation scripts
	$SGHOME/sg/manpages	Man pages
	$SGHOME/sg/RUN/config/ALARM	Alarm configuration template file
	$SGHOME/sg/RUN/config/MML	MML configuration template file
	$SGHOME/sg/RUN/config/PMGR	Application Process Management configuration template file
	$SGHOME/sg/RUN/config/SCTP	SCTP configuration file
	$SGHOME/sg/RUN/config/SNMP	SNMP configuration template files
	$SGHOME/sg/RUN/config/AUTOSTART	Autostart configuration file
	$SGHOME/sg/RUN/config/SGP	Default SGP configuration file

4.2.3 Environment Variables

The following table lists the environment variables required for Signaling Gateway software:

Table 4-2: Environment Variables

Environment Variable	Description
EBSHOME	Base installation directory for NewNet Distributed7
SGHOME	Base installation directory for Signaling Gateway Core

4.2.4 Application Processes

The application software processes are categorized in three groups, as shown in the following table:

Table 4-3: Application Processes

Process Type	Process Name	Description
Basic Platform Processes	apmd	Application Process Manager daemon
	mlogd	Log daemon
	spmd	Service Provider Module daemon
	netd	Network daemon
	alarmd	Alarm daemon
	dsmd	Distributed Shared Memory daemon
	dkmd	Distributed Kernel Memory daemon
SS7 Node Processes	upmd	User Part Multiplexer daemon
	scmd	SCCP daemon
Signaling Gateway Core Processes	sctpd	SCTP (Streams Control Transmission Protocol) daemon, only for Solaris 8/9.
	sgpd	Signaling Gateway Process daemon

- **Basic Platform Processes** - processes that facilitate the common platform services such as process management, logging, alarms, etc. These processes must be running at all times as long as any signaling point is active.
- **SS7 Node Processes** - provides SS7 services.
  - The *upmd* process multiplexes user part traffic, such as ISUP and SCCP. The process must be running in order to bring the SS7 MTP for a signaling point into service. There is one instance of *upmd* for each signaling point in the Signaling Gateway system.
  - The *scmd* process provides SCCP services, such as GTT. There is one instance of *scmd* for each signalling point in the Signaling Gatewaysystem.
- **Signaling Gateway Core Processes** - processes that interwork between the SS7 and IP networks.
  - The sctpd process provides SCTP service to the SGP on Solaris 8/9. Only one instance of the SCTP process is allowed per host.
  - The *sgpd* process translates SS7 MTP messages into M3UA format and transports them to the IP network through SCTP over IP. Only one instance of the *sgpd* process is allowed per host.

4.3 GeneralOperations

This section describes the general operations for administering and maintaining the Signaling Gateway system. Only the **sgadm** user is allowed to perform all administrative and maintenance operations.

4.3.1 Starting the Software

The user may choose to start application processes in different phases. For example, start all processes, including the basic platform, SS7 and Signaling Gateway Core processes, or start only a subset of processes for maintenance. The processes must be started in the following order:

1. Basic Platform processes (*apmd, mlogd, spmd, netd, dsmd, dkmd*)
2. SS7 Node processes (*upmd, scmd*)
3. Signaling Gateway Core process (*sctpd for Solaris 8/9, sgpd*)
4. Miscellaneous processes (*omapd*)

Additionally, each process has certain dependencies on the existence of a set of processes. When starting a particular process, the processes on which that process depend are started automatically.

Process hierarchy is explained below:

- Basic platform processes are the lowest in the hierarchy and are always started in one step. All processes in groups 2, 3 and 4 depend on basic platform process. Hence, whenever any process in groups 2, 3 or 4 is started, group 1 processes are also started if they are not already running.

- P (*scmd*) process depends on MTP3 (*upmd*) process for successful initialization. So whenever they are started *upmd* is also started if it is not already running.

- On Solaris 8/9, SGP (*sgpd*) process depends on SCTP (sctpd) for successful initialization. Whenever SGP is started, sctpd is also started unless it is already running.

Independent from the process hierarchy explained above, it is recommended to start a process before the ones which have higher group numbers (and not automatically started within thehierarchy rules explained above).

•Use the **sg_start** command to start the application processes listed previously.

EXAMPLES

sg_start d7 starts only the Basic Platform Processes.

sg_start sp0 starts the MTP3 process (*upmd*) for signaling point 0. It also starts the Basic Platform Processes if they have not been started already.

sg_start sgp starts the Signaling Gateway Core processes, including SGP and SCTP. It also starts the Basic Platform processes if they have not been started already. However, the SS7 Node processes are not started automatically unless explicitly specified in the **sg_start** command.

sg_start sp0 sp1 sgp starts the Basic Platform processes if they have not been started already, then starts MTP3 for signaling points 0 and 1, and the SCTP and the SGP processes.

Related Commands

- sg_start on page 7-67

4.3.2 Stopping the Software

Use the **sg_stop** command to stop the processes. Like the **sg_start** command, there are arguments that stop one or more processes based on each one's place in the hierarchy. Stopping a process at a specific level in the hierarchy also stops the processes that are higher in the hierarchy.

EXAMPLES

sg_stop asp stops the SGP process.
sg_stop ip stops the SCTP and the SGP processes.
sg_stop sp0 sp1 stops the MTP3 Processes at signaling points 0 and 1.
sg_stop d7 stops the Basic Platform Processes and all other processes.

Related Information

- Starting the Software on page 4-5

Related Commands

- sg_stop on page 7-69

4.3.3 Automatic Startup

The automatic startup of Signaling Gateway can be enabled so that the software starts automatically following a system reboot. Complete the following steps to enable automatic startup:

1. Enter the following to copy `$SGHOME/sg/install/initsg` to `/etc./init.d:`

   cp $SGHOME/sg/install/initsg /etc/init.d

2. Enter the following to create a link for the auto start script:

   cd /etc/rc3.d

   ln /etc/init.d/initsg S99initsg

3. Create a link for the auto stop script:

   cd /etc/rc0.d

   ln /etc/init.d/initsg K00initsg

4. Make the following edits in the `$SGHOME/sg/RUN/config/AUTOSTART/sgstart.conf` configuration file:

   a. Be sure that the SG_AUTO_START value is ON.

   b. Specify the startup states that you want in the SG_START_OPT parameter, using the sg_start command arguments.

5. Save and Close the file.

6. The auto start and stop is now configured. Reboot the system to test it and to verify that Signaling Gateway starts the way you want it to after rebooting.

***Note**: The S99 and K00 numbers specified in Steps 2 and 3 specify the boot sequence and may be changed to match the sequence you want.*

4.3.4 Using MML for Configuration and Query

Most software configurations are done using the Man Machine Interface (MMI) or Simple Network Management Protocol (SNMP). The Man Machine Interface (MMI) uses Managed Objects (MOs) to configure the software. A managed object defines an entity's set of parameters, and which operations can be performed on the managed object (add, delete, modify and/or display).

The ITU Z-315 Man Machine Language (MML) syntax is used with the MMI. An MML session can be started for each logical signaling point.

- Run the following command at the command line to start an MML session:

  mml sp

  where ***sp*** is the signaling point to be configured or queried.

- Type the following in the MML command prompt to display help information for a particular MML command:

  help:;

Then, type the MML command for which help is needed in the MML help prompt, as shown in the following example:

modify-sgp:;

- Type the following in the MML command prompt to terminate an MML session:

exit:;

Related Information

- Chapter 5: MML Commands

4.4 GettingStarted

Complete the following steps to configure the system after all the required software has been installed. These steps must be completed before Signaling Gateway can successfully operate and process traffic.

Important: *This section only serves as a step-by-step guide line, the details of each step of configuration are provided in Software Configuration on page 4-9:*

1. Log in or switch the user to sgadm:

su - sgadm

2. Configure the TCP/IP interfaces on each host (TCP/IP Interface Configuration on page 4-11).

3. Configure the SCTP parameters on each host (SCTP Configuration on page 4-13).

4. Run **ebs_config** on each host to configure the distributed platform:

a. Type the following in the command line:

ebs_config

b. When prompted for the hostname, enter the hostname of the primary cluster LAN interface (this should have been assigned in Step 2).

c. Enter **1** to configure a *distributed system*
OR.
Enter **2** to configure a *simplex system* when prompted for the operation mode

5. Enter the following on each host to start the Basic Platform Processes:

sg_start d7

Only a subset of the Basic Platform Processes start if this is a distributed system and if the cluster LAN has not been configured. Execution is suspended here until the cluster LAN is configured, and a banner is displayed.

6. Run **ebs_ps** and verify that the *apmd*, *mlogd*, *spmd*, and *netd* processes are listed:

ebs_ps

7. Complete this step only if you selected 1 for a *distributed system* in Step 4. c.
Configure the cluster LAN on each host (Section 4.5.1.4, Cluster LAN Configuration on page 4-19).

When the cluster LAN configuration is complete, execution of the rest of the Basic Platform processes resumes. Continue with the SS7 boards and time slots, MTP and Signaling Gateway configurations using MML commands from any host that is connected to the distributed network.

8. From any distributed host, configure the SS7 boards and time slots (Section 4.5.2, SS7 Board and Time Slot Configuration on page 4-22).

9. Configure MTP:

a. On each host, start the MTP3 process for the interested signaling points:

sg_start sp0 sp1 Ret

where sp0, sp1, etc. are signaling point numbers ranging from 0-7.

b. Configure MTP from any distributed host. (Section 4.5.3.1, MTP Configuration on page 4-24)

10. Configure SCCP:

a. On each host, start the SCCP for the interested signaling points:

sg_start sc0 sc1 Ret

where sc0, sc1, etc. are SCCP processes for the signaling point numbers ranging from 0-7.

11. Configure Signaling Gateway:

a. On each host, enter the following to start the SGP process:

sg_start sgp Ret

Note: This command also starts SCTP automatically.

b. Configure Signaling Gateway from any distributed host (Section 4.5.4, Signaling Gateway Configuration on page 4-27).

12. Configure SNMP on each host (Section 4.5.6, SNMP Configuration on page 4-33).

13. Configure the alarms on any distributed host (Section 4.5.5, Alarm Configuration on page 4-31).

4.5 Software Configuration

This section describes the software configuration of Signaling Gateway. It includes instructions and examples on configuring the LAN and SCTP interface, MTP, SIGTRAN, SNMP and alarm handling.

4.5.1 Network Configuration

The Signaling Gateway operates on two different networks:

- LAN for internal cluster communication - used for internal communication among the distributed hosts for data synchronization.
- LAN that connects to the WAN using SCTP over IP - used to communicate between Signaling Gateway and IP nodes.

The two networks should be isolated from each other so that the external traffic passing through the system does not interfere with the internal communication among the distributed hosts.

Figure 4-1 illustrates the recommended network layout for a 4-node distributed Signaling Gateway system. For redundancy purposes, both the cluster LAN and SCTP network have dual Ethernet interfaces. Therefore, a total of five ethernet interfaces/IP addresses are required per host, with two for each network, and one for the primary interface (comes as default with the system):

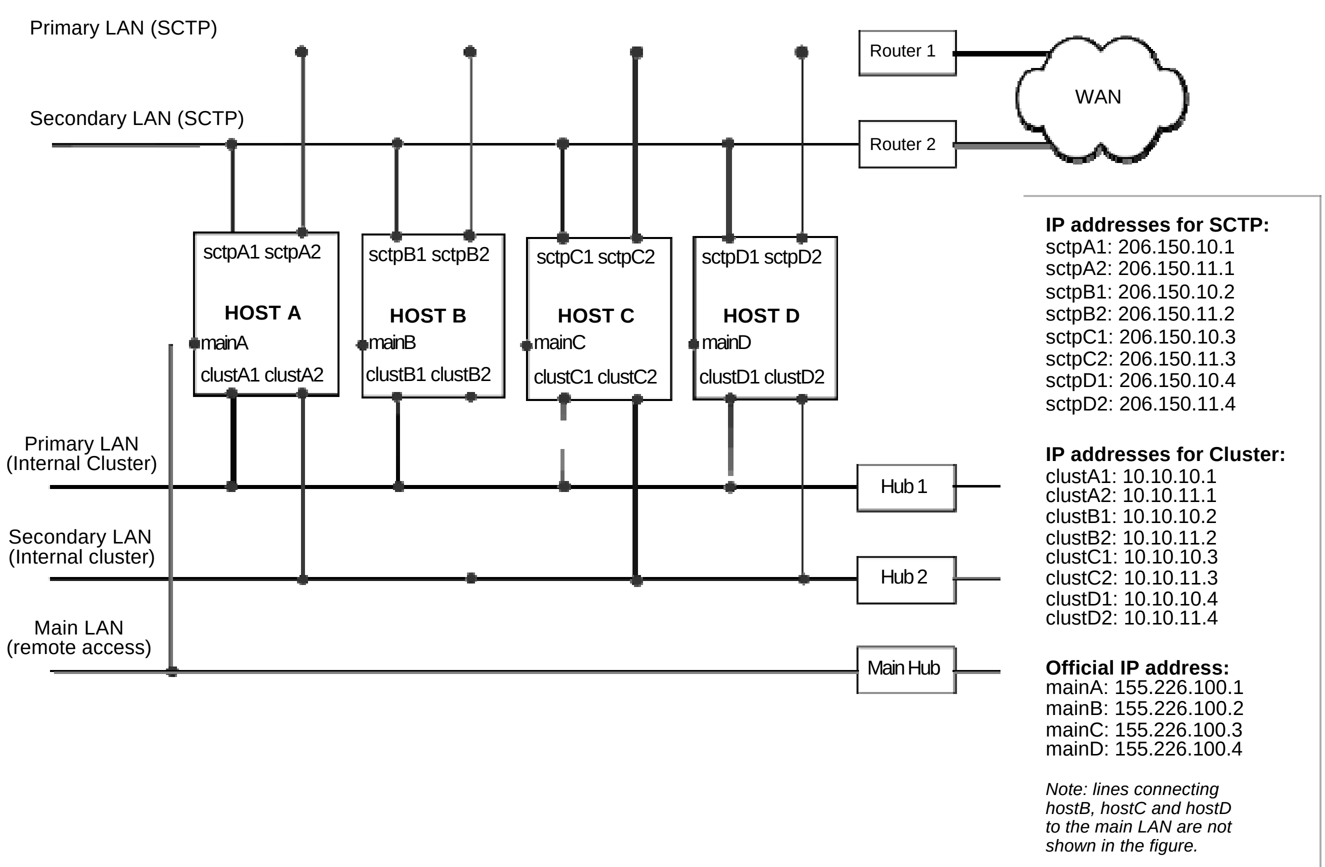


Figure 4-1: LAN Connections

4.5.1.1 TCP/IP Interface Configuration

1. Complete the following before proceeding with the configuration:

a. Design the network topology, similar to the example in Figure 4-1.

b. Assemble the required network hardware according to the topology designed (installing Network Interface Cards (NIC), connecting Ethernet cables, hubs, routers, etc). Each machine must have at least five Ethernet ports:

- two for SCTP access
- two for internal cluster
- one for primary access on the main LAN

c. Assign IP addresses and host names to all network interfaces on each host involved in the network. At least five IP addresses are required for each host:

- two for SCTP
- two for the internal cluster
- one for the main LAN

2. Assign Ethernet interface names:

Each Ethernet interface on a machine is associated with an IP address or a host name. For each Ethernet interface installed on a host, create an `/etc/hostname.`*`interface`* file, where *interface* is the name associated with the Ethernet interface. Refer to the documentation that comes with the NICs for appropriate interface names.

The Solaris installation program automatically creates the `/etc/hostname.`*`interface`* file for the primary network interface that comes with the system. Additional interface files must be created for the interfaces that are installed after the initial Solaris installation.

Repeat the following steps for each of the SCTP and cluster LAN interfaces on each host:

a. With a text editor, create a file named `/etc/hostname.`*`interface`* for the network interface. Using Figure 4-2, Sample Network Configuration, on page 4-24 as an example, if the network interface associated with sctpA1 is qfe0, the name of the file to create on hostA is `/etc/hostname.qfe0`.

b. Type the network interface's name or IP address in the file created in Step 2. a., then save the file. Using the previous example, type *sctpA1* as a one-line entry in the `/etc/hostname.qfe0` file.

3. Configure the IP addresses in the `/etc/hosts` file:

On each host, add all five IP addresses and names of each host involved in the network to the `/etc/hosts` file. Based on the example shown in Figure 4-1, LAN Connections, on page 4-10, the `/etc/hosts` file may look like the following:

```
127.0.0.1       localhost


155.226.100.1   mainA       # hostA: official name
155.226.100.2   mainB       # hostB: official name
155.226.100.3   mainC       # hostC: official name
155.226.100.4   mainD       # hostD: official name
```

```
206.150.10.1   sctpA1     # hostA: SCTP primary
206.150.11.1   sctpA2     # hostA: SCTP secondary
206.150.10.2   sctpB1     # hostB: SCTP primary
206.150.11.2   sctpB2     # hostB: SCTP secondary
206.150.10.3   sctpC1     # hostC: SCTP primary
206.150.11.3   sctpC2     # hostC: SCTP secondary
206.150.10.4   sctpD1     # hostD: SCTP primary
206.150.11.4   sctpD2     # hostD: SCTP secondary

10.10.10.1     clustA1     # hostA: cluster primary
10.10.11.1     clustA2     # hostA: cluster secondary
10.10.10.2     clustB1     # hostB: cluster primary
10.10.11.2     clustB2     # hostB: cluster secondary
10.10.10.3     clustC1     # hostC: cluster primary
10.10.11.3     clustC2     # hostC: cluster secondary
10.10.10.4     clustD1     # hostD: cluster primary
10.10.11.4     clustD2     # hostD: cluster secondary
```

4. Configure `nsswitch.conf` to select the appropriate name services:

a. Open the `/etc/nsswitch.conf` file with a text editor.

b. Be sure that ***files*** is specified as the primary name service for the ***hosts*** and ***services*** network databases. ***Files*** must be listed as the first entry in the list of name services, as shown in the following example:

```
hosts:  files   nisplus [NOTFOUND=return] files
services:   files nisplus [NOTFOUND=return] files
```

These instructions cover basic network configuration. Consult the Solaris System Administration Guides for more details.

4.5.1.2 IP Security Configuration

IP security architecture (IPsec) protects the IP datagrams. It is possible to configure IPsec on the interfaces reserved for SCTP communication. This provides additional security for the SIGTRAN traffic at the IP level. Please consult the Solaris Administration Guide to learn how to configure IPsec.

4.5.1.3 SCTP Configuration

The default SCTP configuration file does not need to be modified for default operation. However, complete the following steps to open the file, customize the parameters, and implement the changes:

1. Open the SCTP configuration file (`$SGHOME/sg/RUN/config/SCTP/sctpd.conf`) with a text editor.
2. The ***sctpd.conf*** has comments explaining each configuration parameter in the file. The parameter values can be edited to meet the user needs.
3. Save the file, and quit the text editor.
4. On Solaris 10, if the sgpd process is running, get the process ID, and send the HUP signal to the sgpd process to implement the changes in the revised ***sctpd.conf*** file. The following is an example of doing this:

   ps -eaf | grep sgpd

   (get the process id)

   kill -HUP <pid>

   On Solaris 8/9, if the sctpd process is running, get the process id and send HUP signal to the process to implement the changes in the revised sctpd.conf file. The following is an example of doing this:

   ps -eaf | grep sctpd

   (get the process id)

   kill -HUP <pid>

Repeat this step on each of the distributed hosts that is running Signaling Gateway.

The following is a printout of the default ***sctpd.conf*** file:

```
##################################################################################
# SCTP Configuration File
##################################################################################

#
# Parameters can be changed in run-time by sending a HUP signal to the sctp daemon process
# on Solaris 8/9, or the asp/sgp daemon process on Solaris 10.
#
#   example:
#
#    On Solaris 8/9
#      prompt> ps -eaf | grep sctpd
#           (note the pid of sctpd in the output screen)
#      prompt> kill -HUP <pid of sctpd>
#
#    On Solaris 10
#      prompt> ps -eaf | grep aspd
#           (note the pid of aspd in the output screen)
#      prompt> kill -HUP <pid of aspd>
#
# To change a configuration item uncomment the example line and modify the
# configuration value according to the acceptable values.
#
# Parameters are divided into two main parts. Changes in PART I parameters
#   are applicable only to the associations opened after the update.
#   Changes in PART II parameters are effective to all associations
#   immediately.
#
# Refer to RFC 2960 for further info on the following parameters.
#
##################################################################################
# PART I
##################################################################################
#
# max_init_timeo <msec>
#   <msec> : INIT Message Time-Out (RTO) in milliseconds. (default = 3000)
#
#   This value is used as the initial value of the Time-Out when sending INIT
#   Message to a destination for setting up the SCTP association.
#
# example:
#   max_init_timeo 3000
#
#
# rto_init <msec>
#   <msec> : Initial Retransmission Time-Out (RTO) in milliseconds. (default = 1000)
#
#   This value is used as the initial value of the RTO before sending traffic
#   to a destination after association setup or when a destination path
#   becomes active.
#
# example:
#   rto_init 1000
#
```



```
#
# rto_min <msec>
#   <msec> : Minimum Retransmission Time-Out (RTO) in milliseconds. (default = 1000)
#
#   Defines the minimum allowed RTO value when SCTP stack adjusts RTO.
#
# example:
#   rto_min 1000
#
#
# rto_max <msec>
#   <msec> : Maximum Retransmission Time-Out (RTO) in milliseconds. (default = 1000)
#
#   Defines the maximum allowed RTO value when SCTP stack adjusts RTO.
#
# example:
#   rto_max 1000
#
#
# valid_cookie_life <msec>
#   <msec> : Valid cookie life in milliseconds. (default = 60000)
#
#   Defines the lifetime of cookie after which the cookie is
#   considered invalid (i.e. stale).
#
# example:
#   valid_cookie_life 60000
#
#
# assoc_max_retrans <attempts>
#   <attempts> : Association max retransmissions. (default = 5)
#
#   The association is closed automatically when number of
#   retransmissions exceeds assoc_max_retrans.
#
# example:
#   assoc_max_retrans 5
#
#
# path_max_retrans <attempts>
#   <attempts> : Path max retransmissions. (default = 2)
#
#   On Solaris 10, the parameter can't be changed in run-time, and
#   it will take effect after aspd/sgpd restarts.
#
#   The path becomes INACTIVE after number of retransmissions to
#   a path (i.e. destination) exceeds path_max_retrans.
#
# example:
#   path_max_retrans 2
#
```

```
#
# max_init_retrans <attempts>
#   <attempts> : Maximum init retransmissions. (default = 5)
#
#   Association setup fails when number of (INIT or COOKIE-ECHO) retransmissions exceeds
# init_max_retrans.
#
# example:
#   max_init_retrans 5
#
#
# hb_interval <msecs>
#   <msecs> : Heartbeat interval in milliseconds. (default = 1000)
#
#   On Solaris 10, the parameter can't be changed in run-time, and it will take effect after aspd/sgpd
# restarts.
#
#   When a path is idle for hb_interval milliseconds a HEARTBEAT message is sent to the destination.
#
#
# example:
#   hb_interval 1000
#
#
# num_in_streams <value>
#   <value> : Default number of incoming streams per assoc. (default = 32)
#
# example:
#   num_in_streams 32
#
#
# num_out_streams <value>
#   <value> : Default number of outgoing streams per assoc. (default = 32)
#
# example:
#   num_out_streams 32
#
#
# arwnd_size <value>
#   <value> : Advertised receive window size in bytes. (default = 264000)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
# example:
#   arwnd_size 264000
#
#
```



```
#
# t_sack <msecs>
#   <msecs> : SACK delay timeout in milliseconds. (default = 200)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
#   Delayed acknowledgement timer. If set to 0 delayed acknowledgement is disabled. The timer value
# must not be greater than 500 msecs.
#
# example:
#   t_sack 200
#
#
# t_shutdown_guard <msecs>
#   <msecs> : Shutdown guard timer in milliseconds. (default = 5000)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
#   Graceful shutdown procedure is aborted when guard timer expires.
#
# example:
#   t_shutdown_guard 5000
#
#
# t_path_mtu <msecs>
#   <msecs> : Path mtu raise timeout in milliseconds. (default = 600000)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
#   In accordance with path discovery procedures the path maximum transmission unit (MTU)
# is increased when this timer expires.
#
# example:
#   t_path_mtu 600000
#
#
# report_unsent <value>
#   <value> : Enable/disable sending of unsent messages in the transmit buffer to the ULP when an
# association is closed. (default = 1)
#           0:disable 1:enable
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
# example:
#   report_unsent 1
#
#
# init_tsn <value>
#   <value> : Initial TSN value for the associations. (default = -1)
#           -1:generated by SCTP randomly
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
# example:
#   init_tsn -1
#
```

```
#
# tx_queue_length <value>
#   <value> : transmit queue length (default = 3000)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
#   Maximum number of data chunks that can be queued up before transmission.
# Overflowed chunks are discarded.
#
# example:
#   tx_queue_length 3000
#
#
# ecn <value>
#   <value> : enabled(1)/disabled(0) (default = 0)
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
#   Explicit Congestion Notification support. (rfc3168, rfc2960)
#
# example:
#   ecn 0
#
################################################################################
# PART II
################################################################################
#
# chksum <value>
#   <value> : checksum algorithm (default = 1)
#           0:Adler32 1:CRC32
#
#   The parameter is valid for sctpd on Solaris 8/9.
#
# example:
#   chksum 1
#
#
# ip_tos <value>
#   <value> : IP Type Of Service value (default = 0)
#
#   ip_tos value is copied to the tos field of outgoing IP packets.
#   meaningful values:
#
#   IPTOS_PREC_NETCONTROL       224
#   IPTOS_PREC_INTERNETCONTROL  192
#   IPTOS_PREC_CRITIC_ECP       160
#   IPTOS_PREC_FLASHOVERRIDE    128
#   IPTOS_PREC_FLASH            96
#   IPTOS_PREC_IMMEDIATE        64
#   IPTOS_PREC_PRIORITY         32
#   IPTOS_LOWDELAY              16
#   IPTOS_THROUGHPUT            8
#   IPTOS_RELIABILITY           4
#   IPTOS_PREC_ROUTINE          0
#
```



```
#
# example:
#   ip_tos 16
#


#
# ip_ttl <value>
#   <value> : IP Time To Live value (default = 64)
#
#   ip_ttl value is copied to the ttl field of outgoing IP packets.
#
# example:
#   ip_ttl 255
#
```

4.5.1.4 Cluster LAN Configuration

The cluster LAN must be configured to connect all the distributed hosts into one network. This must be done on each new system immediately after starting the basic platform processes. For increased reliability, it is highly recommended to deploy the dual-LAN model as illustrated in Figure 4-1 on page 4-10. Complete the following steps on each host to configure the cluster LAN (based on the example in Section 4.5.1 on page 4-10):

The following subsections provide sample configurations for 2-node and 4-node clusters.

2-Node Cluster

The following configuration steps are based on a 2-host example, which is a subset taken from Figure 4-1, LAN Connections, on page 4-10. They also assume that the network has hostA and hostB connected on a dual LAN.

1. Configure hostA:
   a. Log in to hostA and start MML: **mml** 0
   b. Define the primary and secondary cluster host names for hostA:
      modify-ntwk:hostname=clustA1,dualhost=clustA2;
   c. Define the netmasks used for the cluster LAN on hostA:
      modify-ntwk:hostname=clustA1,dualhost=clustA2,netmask1=x, netmask2=y;
   d. Define the remote primary and secondary cluster host names that are reachable from hostA:
      add-host:hostname=clustA1,rmthost=clustB1,alias=clustB2,conf=ON;
2. Configure hostB:
   a. Log in to hostB and start MML.
   b. Define the primary and secondary cluster host names for hostB:
      modify-ntwk:hostname=clustB1,dualhost=clustB2;
   c. Define the netmasks used for the cluster LAN on hostB:

modify-ntwk:hostname=clustB1,dualhost=clustB2,netmask1=x, netmask2=y;

d. Define the remote primary and secondary cluster host names that are reachable from hostB:

add-host:hostname=clustA1,rmthost=clustB1,alias=clustB2,conf=ON;

3. After the hosts are configured, run the **ebs_showlink** command on each host to verify that it is connected to each host in the distributed network.

Note: *For single LAN configuration, follow the same steps as above, except that you do not need to specify the secondary host name. For example:*

modify-ntwk:hostname=clustB1;

modify-ntwk:hostname=clustB1,netmask1=x;

add-host:hostname=clustA1,rmthost=clustB1,conf=ON;

4-Node Cluster

The following configuration steps are based on the details in Figure 4-1, LAN Connections, on page 4-10.

1. Configure hostA:

a. Log in to hostA and start MML: **mml 0**

b. Define the primary and secondary cluster host names for hostA:

modify-ntwk:hostname=clustA1,dualhost=clustA2;

c. Define the netmasks used for the cluster LAN on hostA:

modify-ntwk:hostname=clustA1,dualhost=clustA2,netmask1=x, netmask2=y;

d. Define all remote primary and secondary cluster host names that are reachable from hostA:

add-host:hostname=clustA1,rmthost=clustB1,alias=clustB2,conf=ON;

add-host:hostname=clustA1,rmthost=clustC1,alias=clustC2,conf=ON;

add-host:hostname=clustA1,rmthost=clustD1,alias=clustD2,conf=ON;

2. Configure hostB:

a. Log in to hostB and start MML.

b. Define the primary and secondary cluster host names for hostB:

modify-ntwk:hostname=clustB1,dualhost=clustB2;

c. Define the netmasks used for the cluster LAN on hostB:

modify-ntwk:hostname=clustB1,dualhost=clustB2,netmask1=x, netmask2=y;

d. Define the remote primary and secondary cluster host names that are reachable from hostB:

add-host:hostname=clustB1,rmthost=clustA1,alias=clustA2,conf=ON;

add-host:hostname=clustB1,rmthost=clustC1,alias=clustC2,conf=ON;

add-host:hostname=clustB1,rmthost=clustD1,alias=clustD2,conf=ON;

3. Configure hostC in similar fashion:

modify-ntwk:hostname=clustC1,dualhost=clustC2;

modify-ntwk:hostname=clustC1,dualhost=clustC2,netmask1=x, netmask2=y;

add-host:hostname=clustC1,rmthost=clustA1,alias=clustA2,conf=ON;

add-host:hostname=clustC1,rmthost=clustB1,alias=clustB2,conf=ON;

add-host:hostname=clustC1,rmthost=clustD1,alias=clustD2,conf=ON;

4. Configure hostD in similar fashion:

modify-ntwk:hostname=clustD1,dualhost=clustD2;

modify-ntwk:hostname=clustD1,dualhost=clustD2,netmask1=x, netmask2=y;

add-host:hostname=clustD1,rmthost=clustA1,alias=clustA2,conf=ON;

add-host:hostname=clustD1,rmthost=clustB1,alias=clustB2,conf=ON;

add-host:hostname=clustD1,rmthost=clustC1,alias=clustC2,conf=ON;

5. After the hosts are configured, run the **ebs_showlink** command on each host to verify that it is connected to each host in the distributed network.

***Note**: For single LAN configuration, follow the same steps as above, except that you do not need to specify the secondary host name. See Note after 2-Node Cluster on page 4-20.*

Related Information

- ebs_showlink on page 7-28

Related MML

- Network (NTWK) on page 5-126
- Host (HOST) on page 5-111

Changing the LAN Configuration

Stop the entire platform and complete the following steps if you need to switch from a single-LAN to dual-LAN configuration, or vice versa:

1. Enter the following to stop the entire platform:

sg_stop d7

2. Enter the following to remove all local NTWK, HOST and TCPCO Managed Object databases:

ebs_config -u

3. Enter the following to restart Signaling Gateway software:

sg_start d7

4. Enter the following to start MML:

mml 0

5. Repeat the steps at the beginning of Section 4.5.1.4 on page 4-19 to configure a single or dual-LAN.

Related MML

- ebs_config on page 7-8
- sg_start on page 7-67

4.5.2 SS7 Board and Time Slot Configuration

Each SS7 board installed on a Signaling Gateway host requires that SS7BOARD and TIMESLOT managed objects be added. The following commands provide sample configuration steps for adding an SS7BOARD and configuring its time slots:

1. **SS7 Board**
   a. Enter the following to add an SS7 board, installed in slot 0, with 16 ports:

      add-ss7board:hostname=hostA,boardnm=pci3xpq,inst=0, ports=16;

   b. Modify the clock mode:

      modify-ss7board:hostname=hostA,boardnm=pci3xpq,inst=0, clockmode=LINE,clockspan=1;

   c. The board must be deactivated before the time slots are configured:

      modify-ss7board:hostname=hostA,boardnm=pci3xpq,inst=0,conf=OFF;

2. **SS7 Time Slots**
   a. Configure each time slot on the board:

      modify-timeslot:hostname=hostA,boardnm=pci3xpq,inst=0, desttype=HDLC,destslot=0, origtype=LINE,origspan=1,origslot=1;

      modify-timeslot:hostname=hostA,boardnm=pci3xpq,inst=0, desttype=HDLC,destslot=1, origtype=LINE,origspan=1,origslot=2;

      modify-timeslot:hostname=hostA,boardnm=pci3xpq,inst=0, desttype=LINE,destspan=1,destslot=1,origtype=HDLC,origslot=0;

      modify-timeslot:hostname=hostA,boardnm=pci3xpq,inst=0, desttype=LINE,destspan=1,destslot=2,origtype=HDLC,origslot=1;

3. **Activate Board Configuration**

   modify-ss7board:hostname=hostA,boardnm=pci3xpq,inst=0, conf=ON;

Related MML

- SS7Board (SS7BOARD) on page 5-51
- Time Slot (TIMESLOT) on page 5-85

4.5.3 Signaling Network Configuration

This section describes configuring Signaling Gateway as a signaling gateway that relays SS7 signaling messages between the SS7 and IP networks.

All explanations in the following subsections are based on the sample network in Figure 4-2, Sample Network Configuration, on page 4-24. In this example:

- Signaling Gateway operates on a 2-host distributed platform where one SGP runs on each host. Signaling Gateway has SP0 and SP1 in service, and both SPs are operating in STP mode.
- SP0
  - SP0's PC is 1-1-1, and is connected to an SS7 destination with PC 1-1-10
  - SP0 is assigned NA 100 and serves a remote AS that has PC 1-1-20. Two loadsharing ASPs serve this remote AS.
- SP1
  - SP1's PC is 2-2-2, and is connected to an SS7 destination with PC 2-2-10
  - SP1 is assigned NA 200 and serves two remote ASs:
    - one (AS2) with PC 2-2-20
    - the other (AS3) with PC 2-2-21.
    - A single ASP (ASP3) serves both remote AS2 and AS3.

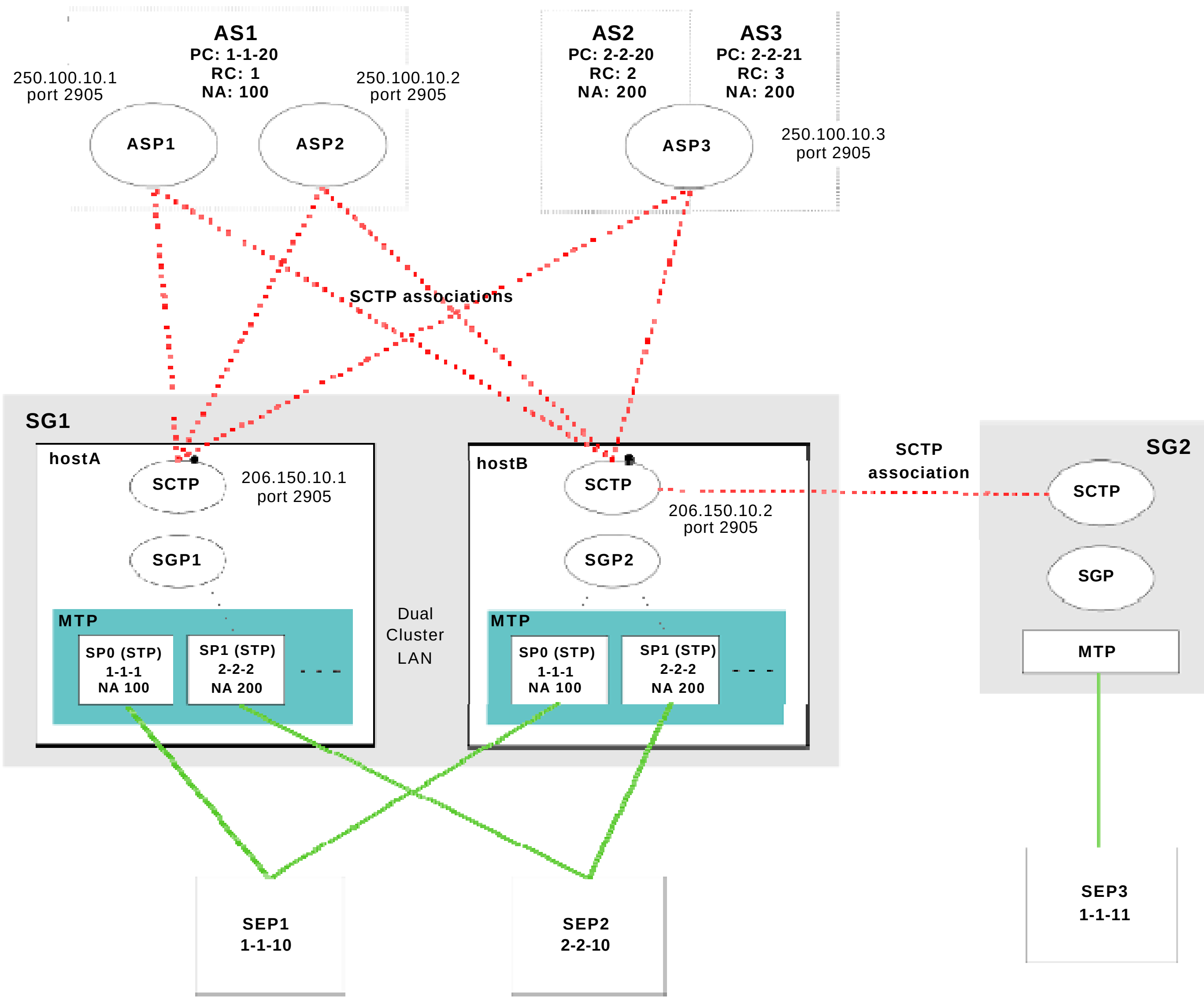


Figure 4-2: Sample Network Configuration

4.5.3.1 MTP Configuration

The Signaling Gateway system must be configured as an SS7 network node in order to process SS7 signaling messages. This section provides a sample configuration steps for configuring SP0 of the SS7 MTP layer, as shown in Figure 4-2, Sample Network Configuration, on page 4-24:

1. **Signaling Node**

   Enter the following to define the protocol information used by SP0:

   add-mtp:protocol=ANSI_96,pcsize=24_BIT; Ret

   After the MTP instance is added, an instance of the SP is created automatically, and is modified in the next step.

2. Point Code

Each addressable SS7 node must be assigned a unique SS7 point code. Assign a point code to SP0 and specify the network indicator and node type:

modify-sp:spc=1-1-1,ni=NATIONAL,type=STP;

3. Route Set

A route set defines a destination that is reachable from a signalling point either directly or indirectly. The following adds a route set for SP0 that leads to SEP 1-1-10:

add-rtset:rtset=RS-SEP1,dpc=1-1-0;

4. Link Sets

Link sets are signaling pathways to an adjacent destination that is reachable directly via SS7 links. Each link set can contain up to 16 links that carry the signaling traffic. The following adds a link set that contains two physical links for route set RS-SEP1:

add-lset:lset=LS-SEP1,dpc=1-1-0,type=ALINK,loaded=2, active=15;

5. Links

Links are physical entities that carry the signaling traffic. Multiple links are grouped into link sets which identify different adjacent destinations. Based on the example in Figure 4-2 on page 4-24, add the following 2 links for link set LS-SEP1:

***Note**: The hostname parameter is assigned the primary cluster network interface as shown in Figure 4-2:*

add-link:link=l1,lset=LS-SEP1,boardnm=sbs334,slot=1,port=1,slc=0, priority=0,hostname=clustA1;

add-link:link=l2,lset=LS-SEP1,boardnm=sbs334,slot=1,port=2,slc=1, priority=1,hostname=clustA1;

6. Routes

For all reachable remote nodes, a route must be created which defines the link set to be used in order to reach that destination. Based on the example in Figure 4-2 on page 4-24, add the following routes leading to SEP1:

add-route:rtset=RS-SEP1,lset=LS-SEP1;

7. Activating Links and Link Sets

Links and Link set can be activated, after they are defined, with the following commands:

modify-lsetstat:lset=LS-SEP1,status=SET_ACT;

modify-linkstat:link=l1,status=SET_ACT;

modify-linkstat:link=l2,status=SET_ACT;

Related MML

- Message Transfer Part (MTP) on page 5-43
- Signaling Point (SP) on page 5-81
- Route Set (RTSET) on page 5-47

- Link Set (LSET) on page 5-40
- Link (LINK) on page 5-36
- Route (ROUTE) on page 5-49
- Link Set Status (LSETSTAT) on page 5-74
- Link Status (LINKSTAT) on page 5-72

4.5.3.2 SCCP Configuration

The SCCP services must be configured if SCCP or TCAP applications are supported.

1. **SCCP Network Provisioning**

   The SCCP network is a subset of the MTP network. The SCCP node is defined *after* a signaling point is defined for MTP. For example:

   add-snsp:spc=1-1-20;

2. **Subsystem Provisioning**

   SCCP users can communicate only with subsystems provisioned in the SCCP network. In ITU WHITEBOOK networks, the SCCP management subsystem is created by SCCP and has a predefined value of SSN=1. The following example adds a subsystem for a particular SCCP node:

   add-subsys:spc=1-1-20,ssn=254;

3. **Concerned SP Provisioning**

   A concerned signaling point is a remote signaling point that accesses the local subsystem and queries its local status. SCCP management uses the concerned point code information to determine which remote signaling points must be notified of state changes in the local subsystem. The following command adds a concerned point code for a signaling point with PC 1-1-20:

   add-cpc:spc=1-1-1,cpc=1-1-20,ssn=254;

4. **Global Title (GT) Provisioning**

   A global title is an alias address that can be translated to a point code, a point code and a subsystem number, or a new global title. A global title address contains information that allows routing in the signaling network. The GT table is used to define the GT entries which are translated in the GTENTRY table. A maximum of 131,072 (256 translation types x 512 GTs per translation type) entries can be entered. The following command add GT1 as a global title for 2039251111:

   add-gt:gt=GT1,GTIE=4,trtype=0,addrinfo=12039251111;

5. **Global Title Entry (GTENTRY) Provisioning**

   The GTENTRY table defines the GT translation for the SCCP routing module. Two types of GTs are maintained: incoming and outgoing translations. The incoming GT translates messages coming from the network, whereas the outgoing GT translates messages going to the network. Cycles in GT are not allowed. A maximum of 131072 (256 translation types x 512 GTs per translation type) entries can be entered. The following command adds a GT entry that SCCP uses to translate GT1:

   add-gtentry:IO=incoming,GT=GT1,spc=1-1-20,ssn=254;

6. **Mated Subsystem Provisioning**

SCCP subsystems can be mated with each other as pairs to provide redundancy in case of failure. The signaling points, subsystem numbers, and concerned point codes have to be designed prior to mating two subsystems. For example, the following command defines a mated pair of subsystem 20 and 254:

add-mate:spc=1-1-1,ssn=20,mspc=1-1-20,mssn=254;

4.5.4 Signaling Gateway Configuration

The following serves as a step-by-step guideline for configuring Signaling Gateway. It is recommended that these guidelines be followed in the order provided:

1. Configure signaling point and network appearance mapping
2. Configure local SGP
3. Configure remote ASP
4. Configure remote peer SG and SGP - only if SG-SG feature is used
5. Configure DPC in IP domain
6. Configure DPC in SS7 domain

4.5.4.1 Signaling Point and Network Appearance Mapping

Signaling Gateway can support up to eight signaling points (SP), each connecting to a different SS7 network. Each SP uniquely identifies a Network Appearance in the SIGTRAN term. When traffic for multiple networks are carried on the same SCTP association, the network appearance ID is used to separate the traffic.

The mapping between SP and NA must be defined, using the SGSPNA MO, before any route key and AS can be defined. An SGSPNA MO must be created for each SP that will be in service, and a unique NA ID must be assigned to each SP. For example:

add-sgspna:spno=0,naid=100;

add-sgspna:spno=1,naid=200;

The SP must be activated so that it can be in service after the mapping between NAID and SP is defined:

mod-sgspna:spno=0,operst=ACT;

mod-sgspna:spno=1,operst=ACT;

Related MML

- Signalling Point to Network Appearance Mapping (SGSPNA) on page 5-135

4.5.4.2 Local SGP

The SGSGP managed object defines the local SGP process's IP addresses and SCTP port number used for SCTP communication. Two IP addresses can be specified for SCTP multihoming purposes.

By default, the SGSGP managed object is created automatically for each SGP process, but has no IP addresses or port definitions initially.

Modify the SGSGP MO for each host to set the IP addresses and port number used by the SGP for SCTP communication:

modify-sgsgp:sgpid=hostA,ip1=206.150.10.1,ip2=206.150.11.1;
modify-sgsgp:sgpid=hostB,ip1=206.150.10.2,ip2=206.150.11.2;

***Note**: When **sctpport** is absent, a default value of 2905 is set.*

***Important**: The SGP ID is pre-assigned with the machine's official host name when the SGSGP MO is first created. Use the UNIX **uname -n** command to display a machine's official host name.*

Related MML

- Signaling Gateway Process (SGSGP) on page 5-138

4.5.4.3 Remote ASP

The SGASP managed object defines the remote ASP's IP addresses and SCTP port numbers used for SCTP communication. An SGASP managed object must be added on the Signaling Gateway system for each ASP instance that is to connect to the SGP.

Two IP addresses can be specified for SCTP multi-homing purposes. Each ASP must be assigned a unique SCTP port number for communication with SCTP if multiple ASPs are running on the same host. Be sure that the IP addresses and port numbers configured on the Signaling Gateway side are identical to that of the ASP side to ensure successful connection.

The following commands configure the remote ASPs as shown in Figure 4-2, Sample Network Configuration, on page 4-24:

add-sgasp:aspid=ASP1,ip1=250.100.10.1,ip2=250.100.11.1;
add-sgasp:aspid=ASP2,ip1=250.100.10.2,ip2=250.100.11.2;
add-sgasp:aspid=ASP3,ip1=250.100.10.3,ip2=250.100.11.3;

Related MML

- Application Server Process (SGASP) on page 5-143

4.5.4.4 Remote Peer SG and SGP

The SG-SG feature support allows the Signaling Gateway to be configured to route messages to and from an SS7 destination point code via another remote peer SG using M3UA and SCTP protocols. In this scenario, SS7 messages received at the local SGP are transported using M3UA and SCTP to a peer SGP to be delivered to the remote SS7 destination. If the SG-SG feature is deployed, the remote peer SG and SGP must be configured as described in the following paragraph.

The SGPSG MO defines the attributes of a remote peer SG. For each peer SG that will communicate with the local Signaling Gateway, an SGPSG instance must be added, for example:

add-sgpsg:sgid=sg2;

The SGPSGP MO defines the attributes of an SGP that runs in the remote peer SG. For each peer SGP that will connect to the local SGPs, an SGPSGP instance must be added, for example:

add-sgpsgp:psgpid=psgp1,sgid=sg2,ip1=155.226.145.143;
add-sgpsgp:psgpid=psgp2,sgid=sg2,ip1=155.226.145.144;

Related MML

Peer SG (SGPSG) on page 5-155 and Peer SGP (SGPSGP) on page 5-157.

4.5.4.5 IP Destination Point Code

The attributes of the SS7 MTP-User peer in the IP domain is modeled as Application Server (AS) and Route Key in SIGTRAN terms. An AS is a logical entity serving a specific route key, in which a route key describes a set of SS7 parameters that uniquely defines a range of signalling traffic served by that particular AS. There is a one-to-one mapping between an AS and a route key. First, a route key must be defined, then it is attached to an AS. The following subsections describe how the AS and route key are configured.

Route Key

The SGRK MO defines the route key type and the destination point code of a remote AS. An SGRK instance must be created before it is attached to an AS. The following types of route keys are supported:

DPC route by DPC only

DPC_OPC route by DPC and OPC

DPC_OPC_CIC route by DPC, OPC and CIC ranges

DPC_OPC_SIO route by DPC, OPC and SIO

DPC_SIO route by DPC and SIO

DPC_CIC_SIO route by DPC, SIO and CIC ranges

DPC_SSN route by DPC and SSN

The following examples add three route keys, one for each remote AS, as shown in Figure 4-2: on page 4 - 24:

add-sgrk:rkid=r1,type=DPC,dpc=1-1-20;
add-sgrk:rkid=r2,type=DPC_SIO,dpc=2-2-20;
add-sgrk:rkid=r3,type=DPC_CIC_SIO,dpc=2-2-21;

If traffic is to be routed according to a set of SS7 parameters in addition to the DPC (i.e. route key type other than DPC), then the SGRKRNG instances must be defined to specify the traffic range parameters. As in the above example, route key r2 and r3 both have types other than DPC, their SGRKRNG instances are defined as follows:

add-sgrkrng:rkid=r2,SI=ISUP;
add-sgrkrng:rkid=r3,SI=ISUP,cicmin=0,cicmax=1000;
add-sgrkrng:rkid=r3,SI=ISUP,cicmin=2000,cicmax=3000;

Application Server

The SGAS MO defines the attributes of a remote AS. Each SGAS instance is associated with an instance of SGRK which must be created in advance. In addition, each AS is also associated with a unique numeric value called *route context* (RC). There is a one-to-one mapping between an AS and a RC ID. The RC ID provisioned on the Signaling Gateway side must be identical to that of the ASP side to ensure successful message routing.

An AS can be assigned a list of up to five ASPs, operating in either override or load sharing mode. These ASPs must have been created before they can be in the list of assigned ASPs (Remote ASP on page 4-28).

Signaling Gateway supports a total of 255 ASs, and each SP can be configured to serve more than one AS. When adding the SGAS MO, the SPID is specified to associate the AS with the SP.

Using Figure 4-2, Sample Network Configuration, on page 4-24 as an example, SP0 serves AS1 in the network with NA 100, whereas SP1 serves AS2 and AS3 in the network with NA 200. The remote ASs are configured as follows:

add-sgas:asid=AS1,spid=0,rkid=r1,rcid=1, asplist=ASP1/ASP2; Ret

add-sgas:asid=AS2,spid=1,rkid=r2,rcid=2, asplist=ASP3; Ret

add-sgas:asid=AS3,spid=1,rkid=r3,rcid=3, asplist=ASP3; Ret

Related MML

- Application Server (SGAS) on page 5-140

4.5.4.6 SS7 Destination Point Code

All SS7 destination point codes (DPC) reachable by the local SG must be configured for proper message routing. The SS7 destination point codes are categorized as the following types:

- SS7 - DPC that is reachable via conventional SS7 route set
- IPSS7 - DPC that is reachable via another remote peer SG using M3UA and SCTP over IP. This type of point code is configured only when the SG-SG feature is used.

The SGDPC MO defines the SS7 DPC that is reachable by the Signaling Gateway. For DPC of type IPSS7, a list of SGs through which the DPC is reached, as well as the traffic mode of the list of SGs for that DPC, must also be specified. For example:

add-sgdpc:dpc=1-10-10,naid=100,type=SS7; Ret
add-sgdpc:dpc=1-10-11,naid=100,type=IPSS7,sglist=sg2; Ret

4.5.4.7 Monitoring AS Traffic Status

Since an ASP may serve multiple ASs, it has the different traffic statuses for all the ASs it serves. For example, an ASP may be active for AS1 (actively processing traffic for AS1), but inactive for AS2 (not processing traffic). The relation between the ASP, AS and the traffic statuses are represented in the SGASTFC MO. The following command displays the current traffic status:

display-sgastfc:;

The following command displays the traffic statuses of all ASs served by ASP1:

display-sgastfc:aspid=ASP1;

The following command displays the traffic statuses for AS1, including all the ASPs serving it:

display-sgastfc:asid=AS1;

4.5.5 Alarm Configuration

4.5.5.1 Alarm Type and Severity

Alarms are generated to alert the operators of any system faults or major events that require attention. Alarms are categorized in the following groups:

- EVENT - informational alarms that are generated usually for event logging purposes
- SET_ALARM - alarms that are set in the system and remain until it is manually cleared by an operator
- CLR_ALARM - alarms that are generated to automatically clear an alarm of SET_ALARM type.

All alarms are logged and kept in the directory `$EBSHOME/access/RUN/alarmlog`. Alarms of SET_ALARM type can be queried through MML. Some alarms of SET_ALARM type are self-clearing—they are cleared automatically when the error condition is corrected. Non-self-clearing alarms must be cleared manually by the operator.

Alarms are rated according to the following four severity levels:

- INFO - informational alarm, reporting events such as system startup, shutdown, etc. These alarms do not affect the functional state of the system.
- MINOR - alarms that indicate minor error conditions that do not impair the continued operation of the system, such as configuration errors, invalid messages received, etc. These alarms should not be ignored as they may indicate an impending problem.
- MAJOR - alarms that indicate major error conditions that affect the normal operation of the system, such as traffic congestion, resource utilization reaching maximum threshold, etc. These alarms require immediate attention to prevent potential service interruption.
- CRITICAL - alarms that indicate critical error conditions that is interrupting the normal operation of the system, such as process failure, loss of network connection, hardware faults, etc.

4.5.5.2 Alarm Groups

The system alarms are categorized into various groups. Each group is responsible for different types of functions. The alarm groups are:

- TRMOD (translation module)
- SPM (signaling point management)
- UPM (user part multiplexer)

- NIMOD (connection management)
- DKM (distributed kernel memory)
- Signaling Gateway (signaling gateway application)

Related Information

- Chapter 8: Maintenance and Troubleshooting
- Appendix C: Alarms

4.5.5.3 Alarm Display

By default, all alarms are displayed on the console and written to the alarm log file. Run the following command to toggle the display option:

modify-alarm:display=option;

where *option* is either ON or OFF.

The alarms that are displayed can be filtered according to their severity levels. For example, to display alarms that have severity level that are equal to or higher than MINOR, Run the following command MML command:

modify-alarm:cons_thrs=MINOR;

Related MML

- Alarm (ALARM) on page 5-116

4.5.6 SNMP Configuration

Signaling Gateway supports Simple Network Management Protocol (SNMP) versions 1 and 2 for network management. In a network management system, there can be multiple network elements such as hosts, routers, terminal servers, etc., each with a processing agent called an *SNMP agent*. A centralized location called the *network management station* or *SNMP manager* is used to manage these individual network elements. The manager remotely controls or monitors the network elements by accessing their *management information*.

Management information is simply a collection of managed objects residing in a virtual information store, called a Management Information Base (MIB). MIBs are defined by the Structure of Management Information (SMI), which is a subset of OSI's Abstract Syntax Notation One (ASN.1).

Figure 4-3 illustrates the SNMP architecture involving an SNMP Agent and an SNMP Manager.

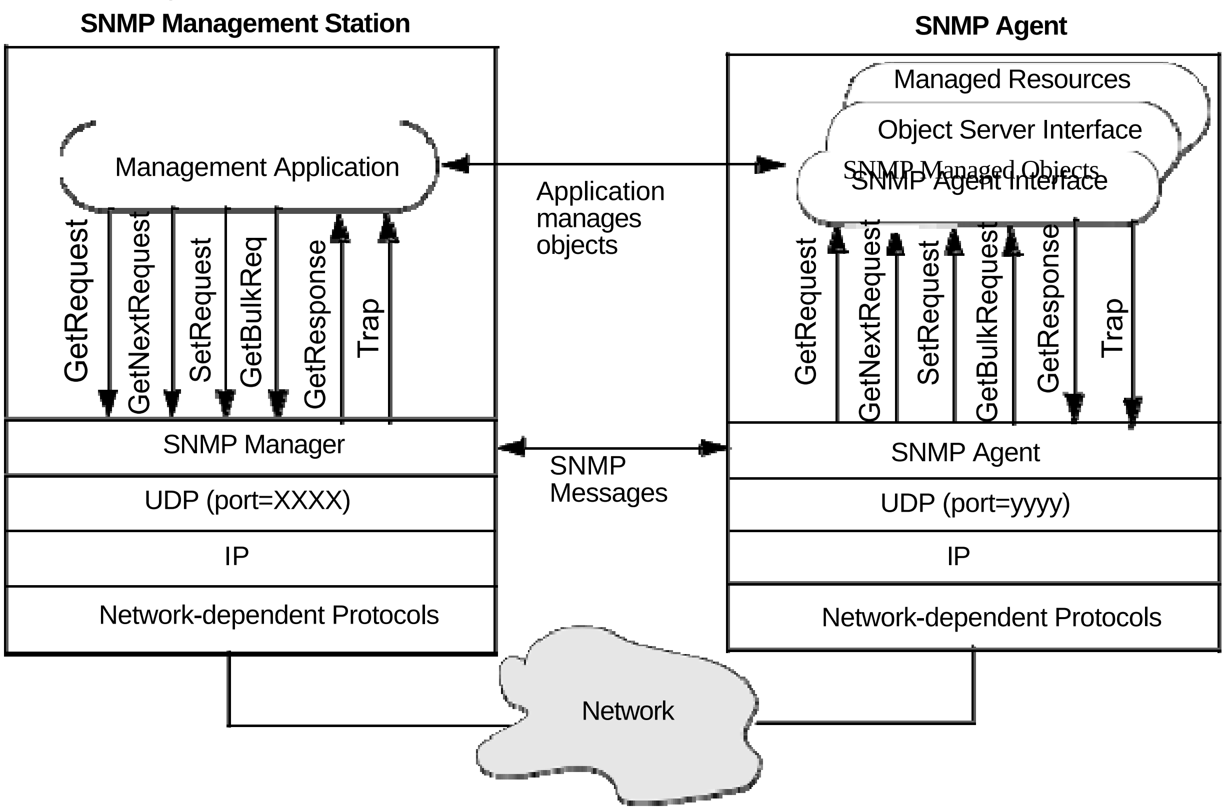


Figure 4-3: SNMP Architecture

The Signaling Gateway software package comes with an SNMP Agent that supports both SNMP v1 and v2. All Signaling Gateway MIBs are defined by the *ebs* branch under the .internet.private.enterprises subtree. The NewNet Distributed7 MIBs are defined by the *.ebs.accessMANAGER* branch, while Signaling Gateway MIBs are defined by the *.ebs.sg* branch. The following figures illustrate the MIB hierarchy on Signaling Gateway system (the key fields are bolded).

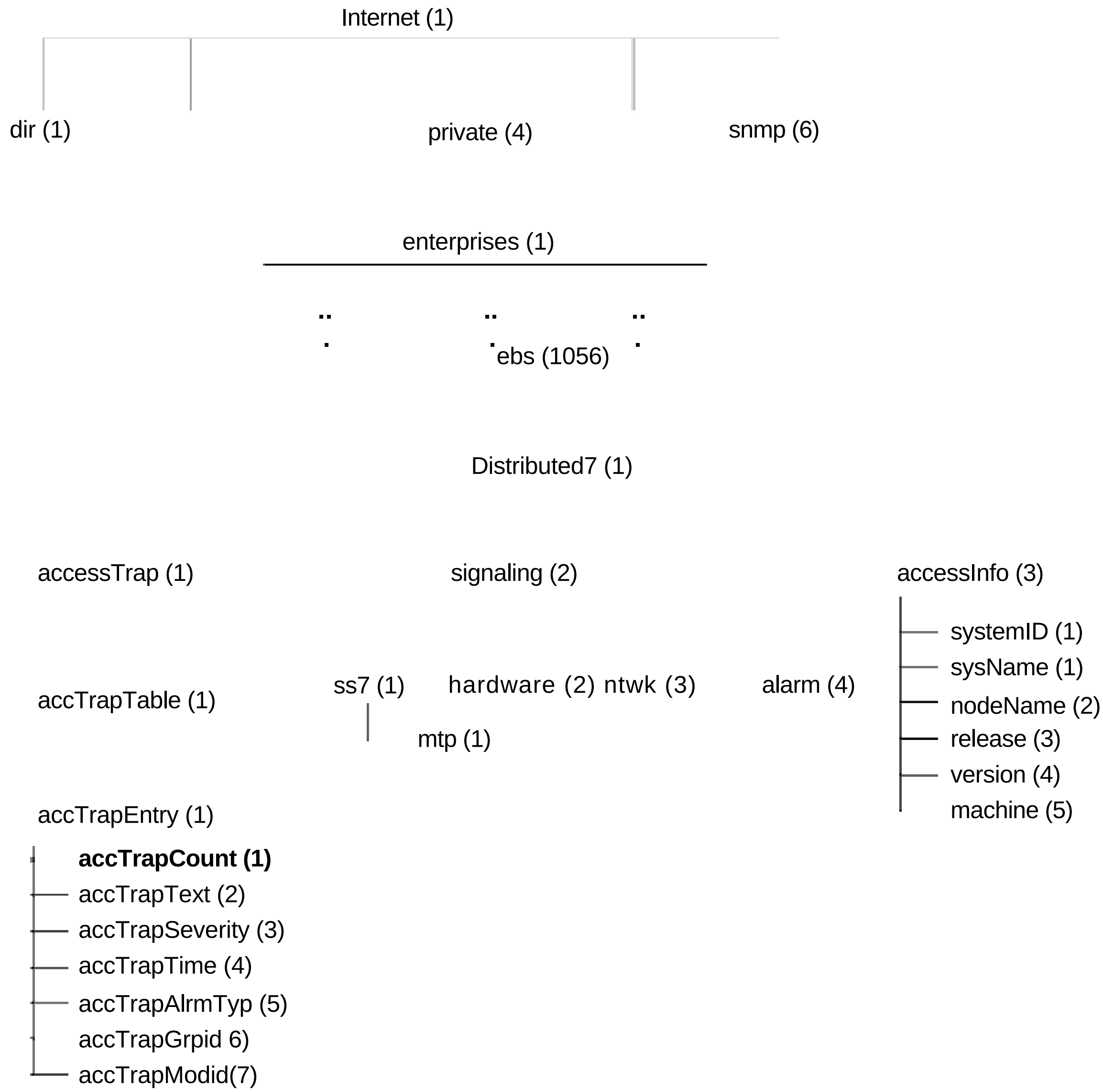


Figure 4-4: Internet MIB Tree

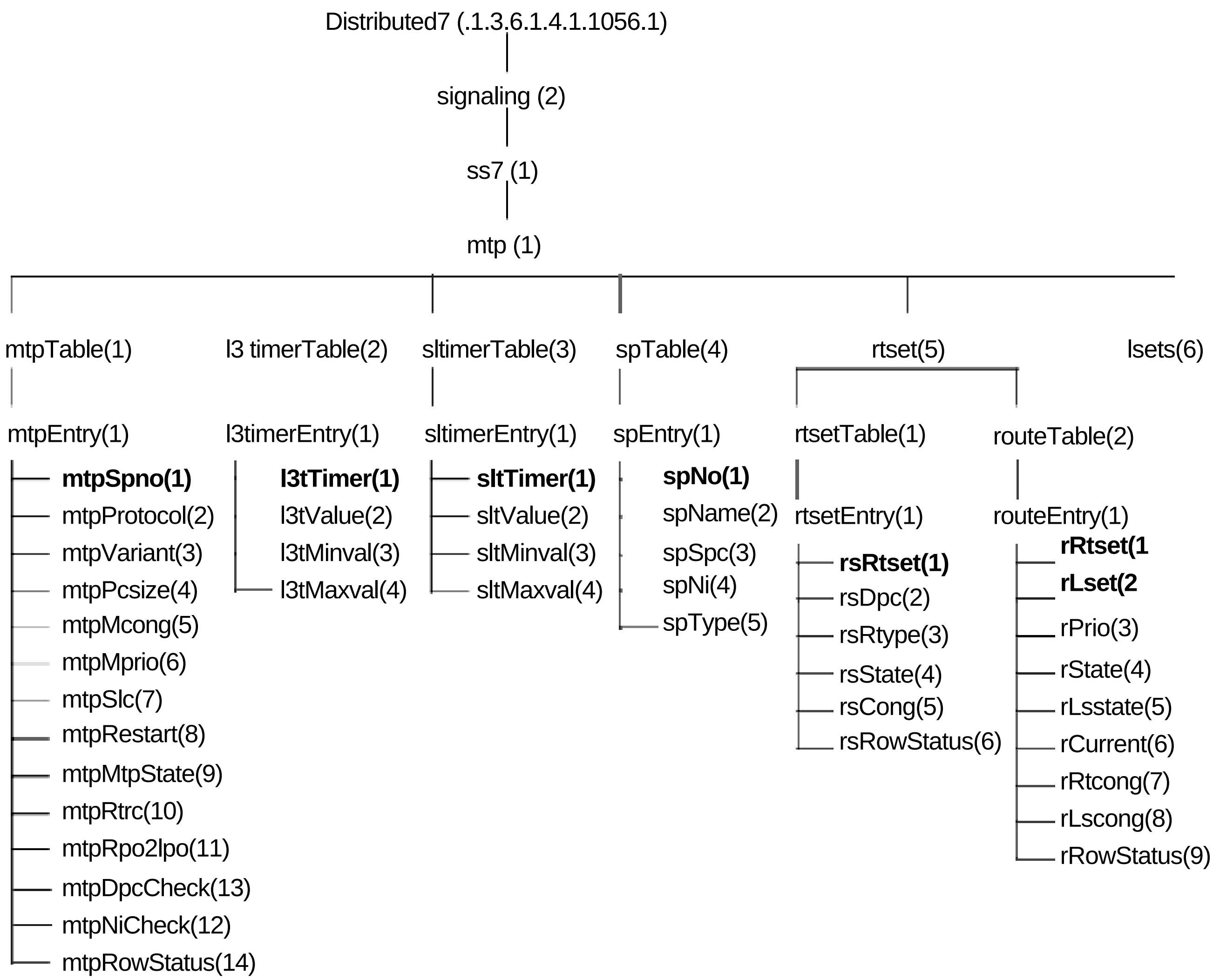


Figure 4-5: NewNet Distributed7 MIB - MTP Tree 1

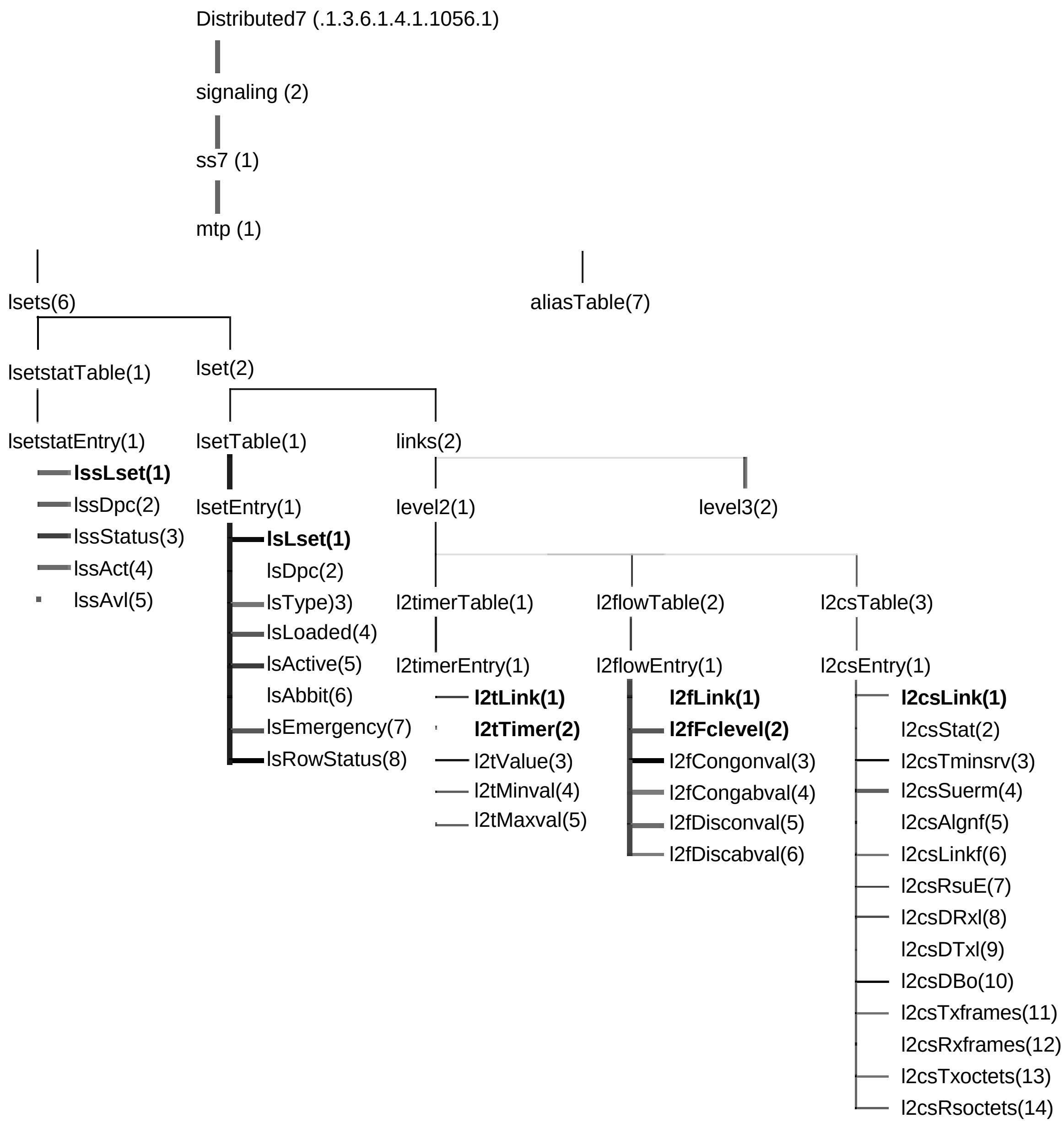


Figure 4-6: NewNet Distributed7 MIB - MTP Tree 2

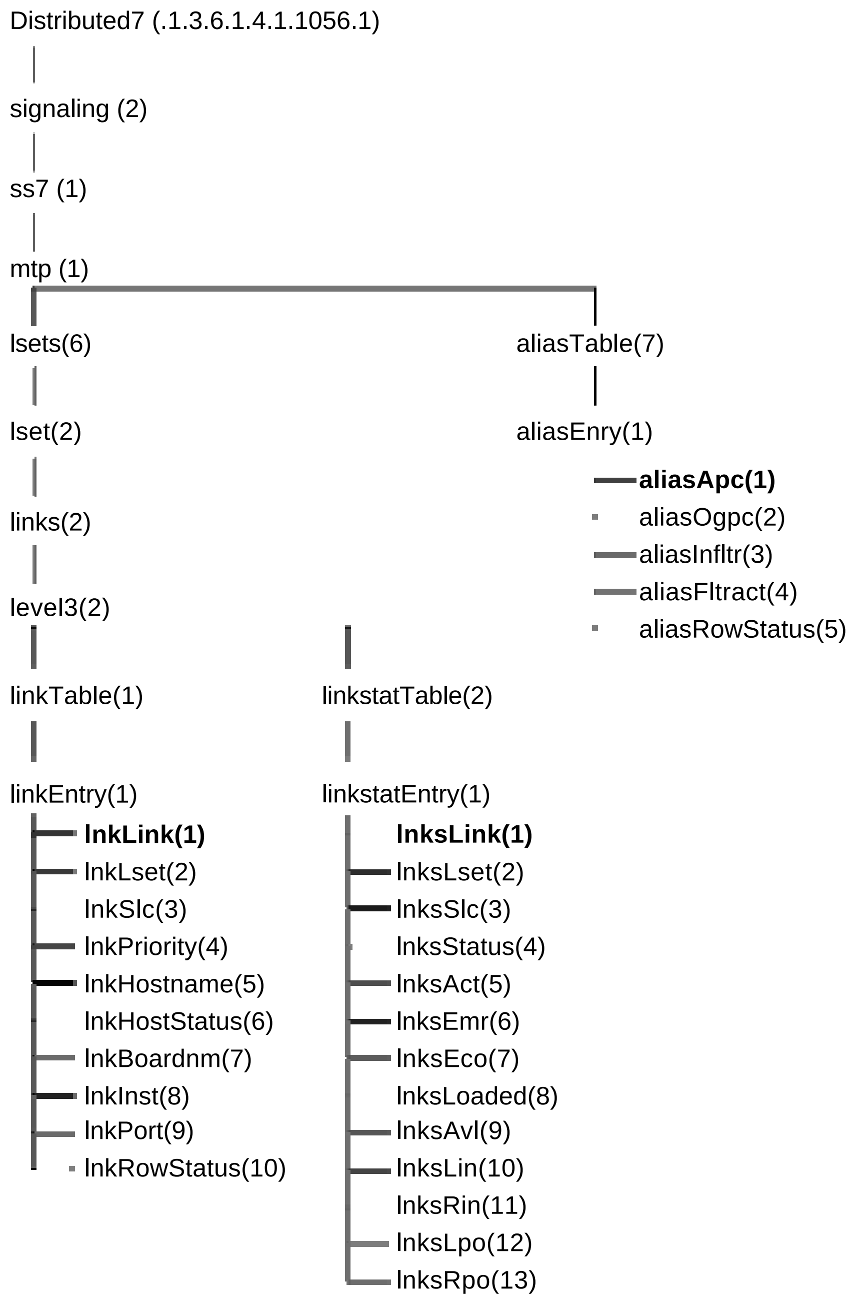


Figure 4-7: NewNet Distributed7 MIB - MTP Tree 3

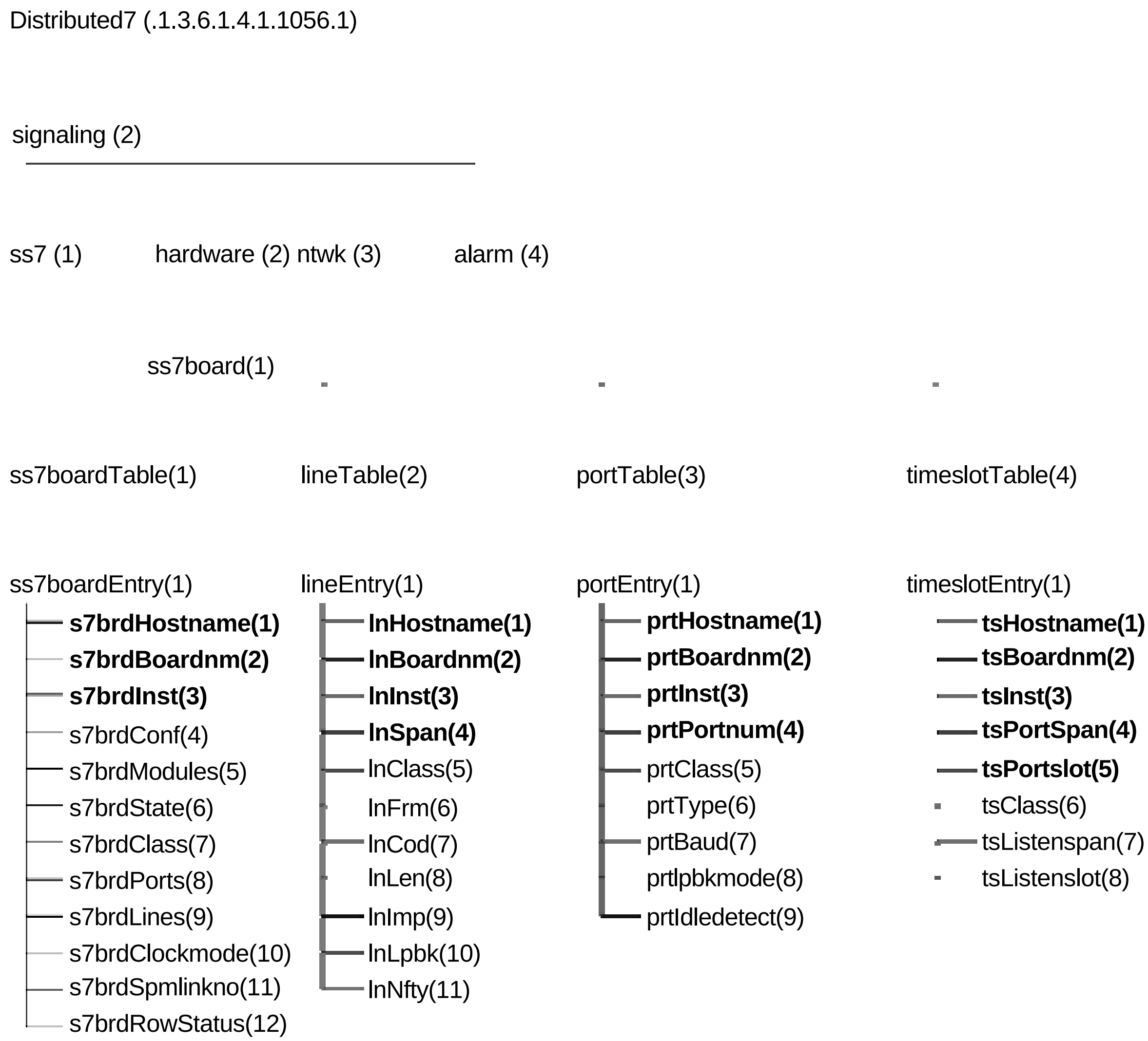


Figure 4-8: NewNet Distributed7 MIB - Hardware Tree

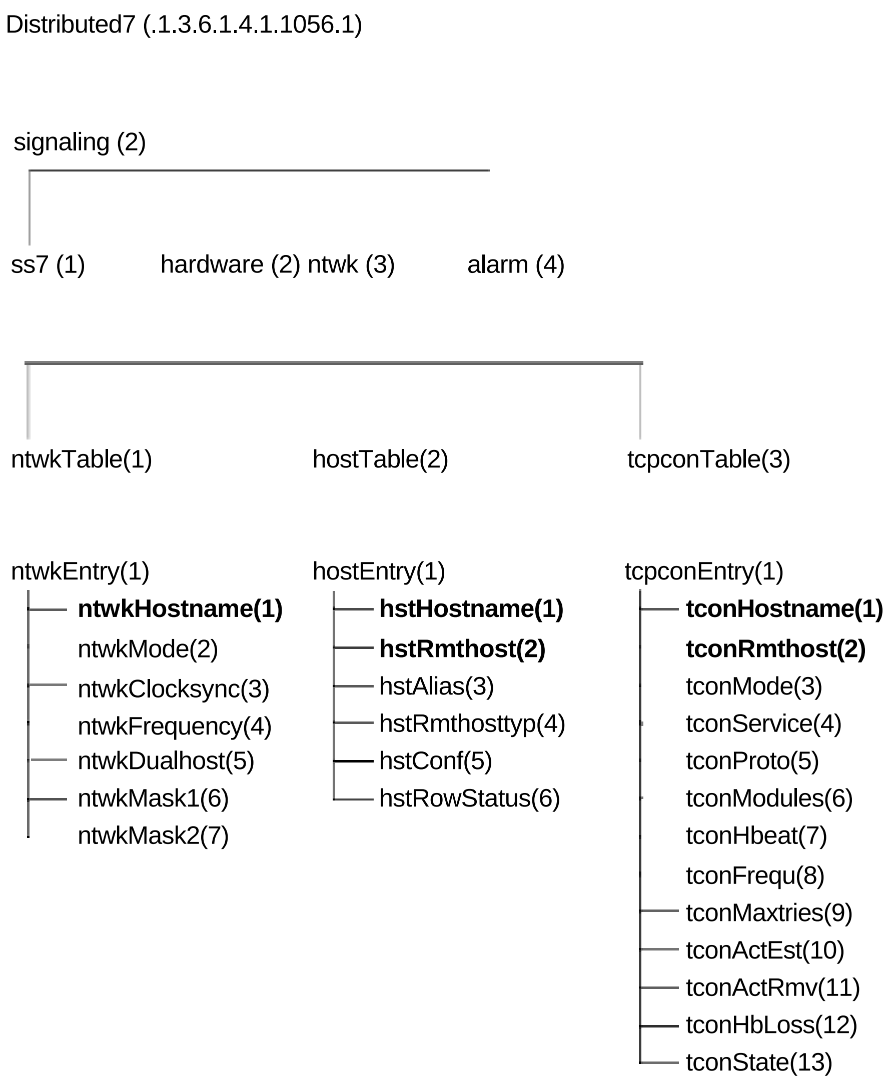


Figure 4-9: NewNet Distributed7 MIB - Network Tree

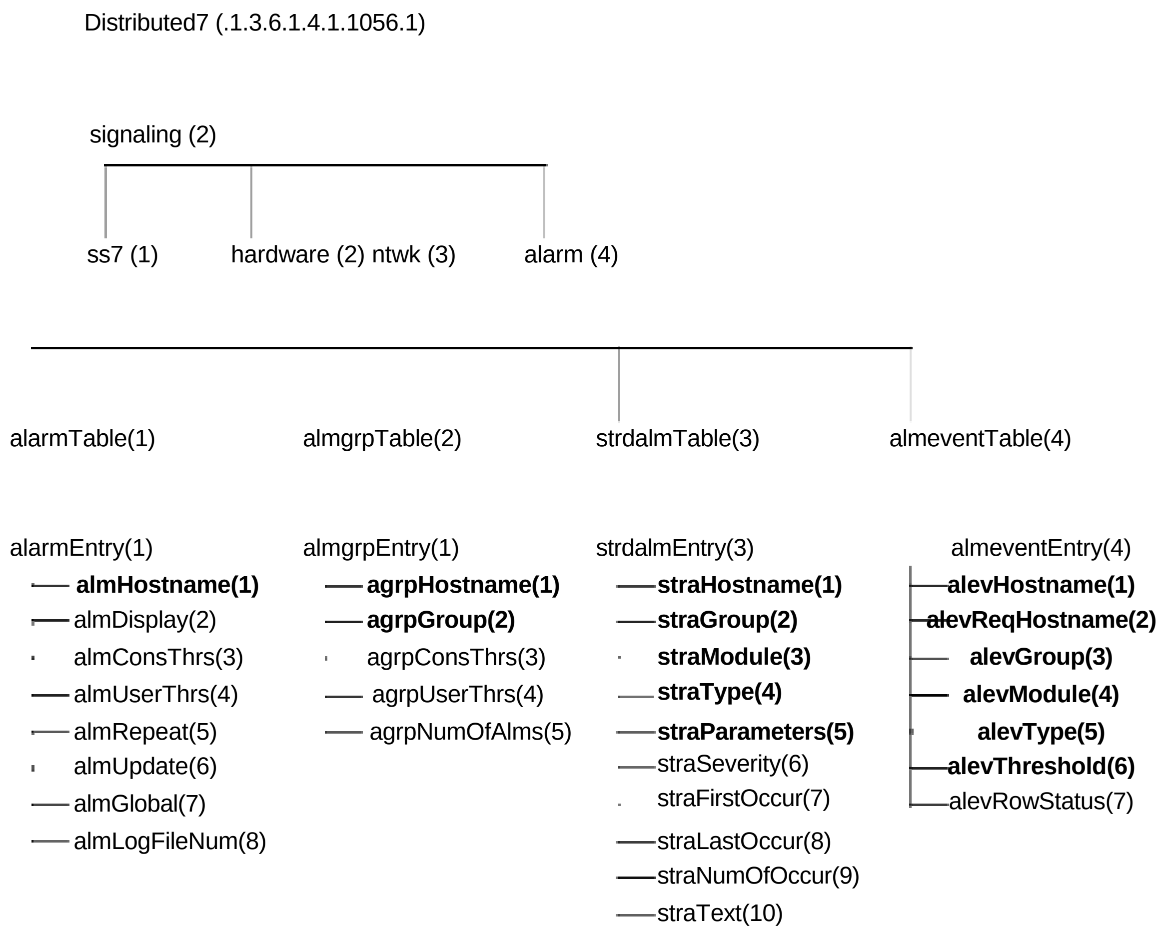


Figure 4-10: NewNet Distributed7 MIB - Alarm Tree

Figure 4-11: Signaling Gateway MIB Tree

4.5.6.1 Configuring SNMP Agent

The MIBs and SNMP agent configuration files must be properly configured and placed in the appropriate directories to manage Signaling Gateway using SNMP. The following table describes the location and meaning of all the configuration files involved.

Table 4-4: SNMP Configuration Directories and Files

Directory	Configuration File	Description
$EBSHOME/access/RUN/config/SNMP	mib_text.v1	MIB modules for SNMP v1. No modification is required in this file.
	mib_text.v2	MIB modules for SNMP v2. No modification is required in this file.
	snmp_cmnd.tbl	SNMP and MO mapping table. No modification is required in this file.
$EBSHOME/access/RUN/config/SNMP/etc	trap.ini (SNMP v1)	Template that defines where SNMP traps must be sent. Must be copied to $EBSHOME/access/RUNx/config/SNMP/trap.conf.
	community.ini (SNMP v1)	Template that defines the community attributes. Must be copied to $EBSHOME/access/RUNx/config/SNMP/community.conf.
	party.ini (SNMP v2)	Template that defines the party attributes. Must be copied to $EBSHOME/access/RUNx/config/SNMP/party.conf.
	context.ini (SNMP v2)	Template that defines the context attributes. Must be copied to $EBSHOME/access/RUNx/config/SNMP/context.conf.
	view.ini (SNMP v2)	Template that defines the supported MIB view. Must be copied to $EBSHOME/access/RUNx/config/SNMP/view.conf.
	acl.ini (SNMP v2)	Template that defines the access policy. Must be copied to $EBSHOME/access/RUNx/config/SNMP/acl.conf.
$EBSHOME/access/RUN*x*/config/SNMP where *x* is the signalling point number.	trap.conf	Defines where traps are sent.
	community.conf	Defines community attributes.
	party.ini	Defines party attributes.
	context.ini	Defines context attributes.
	view.ini	Defines MIB view.
	acl.ini	Defines access policy.

- Determine which version of SNMP, v1 or v2, you are using before configuring anything.
- Then, based on Table 4-4, copy the necessary configuration templates named as **$EBSHOME/access/RUN/config/SNMP/etc/*.ini**
  to
  $EBSHOME/access/RUNx/config/SNMP/*.conf,
  where **x** is the signaling point number on which the SNMP Agent should run.

- If both SNMP v1 and v2 are used, then copy all the configuration templates. Otherwise, only copy the set of files needed for the version you are using. Based on the version, complete the following instructions to edit the configuration files.

SNMPv1 Configuration Files

Two SNMPv1 configuration files exist. For SNMPv1, security is only at the community information level. Each file should be edited as described below.

community.conf

Contains the port numbers for listening SNMPv1 requests. Up to 4 ports can be defined for listening SNMPv1 requests on the first line. The community names to be accepted by the SNMP agent are defined in the second line. After initialization, port number ‘7778’ and community name ‘public’ are defined for your system by default. Both lines can be modified. Sample lines from the file appear below:

```
#local-port-nums for listening snmpV1 managers (maximum 4 ports)
#defined community-names for identifying snmpV1 managers (maximum 4 communities)
7778 yyy zzz
public
```

trap.conf

Contains the community information, network manager IP address, and port number to which the SNMPv1 traps will be sent. Sample lines from the file appear below:

```
#community name        remote manager net address remote port number
public                 yyy.yyy.yyy.yyy  xxx
```

SNMP v2 Configuration Files

The four SNMPv2 configuration files incorporate three security concepts - party concept, context concept and access policy concept. Each file should be edited as described.

party.conf

Contains party concept elements. The required elements are party friendly name, party object identifier, domain name, IP address, and port number. The file must be edited to replace xxx.xxx.xxx.xxx with the network address of the agent station and yyy.yyy.yyy.yyy with the network address of the management station. Port numbers are set to defaults, but they can be modified. If the chosen party is not using authentication, then there is no need to configure any other elements. If the chosen party is using *SNMPv2 md5* authentication, then **snmpv2md5Auth** must be placed in *authProtocol* and the *authPrivate* text string key has to be verified to be the same as the manager party *authPrivate* key. Sample lines from the file follow:

```
# FriendlyName PartyID
# TDomain      IP-address     UDP-port
# authProtocol   privProtocol
# lifetime maxmessagesize
# clock
# authPrivate authPublic
# privPrivate privPublic

agent_accessMANAGER                 .1.3.6.1.6.3.3.1.3.xxx.xxx.xxx.xxx.1
snmpUDPDomain                   xxx.xxx.xxx.xxx        7777
noAuth                      noPriv
300                        484
00000000
00000000000000000000000000000000      Null
00000000000000000000000000000000      Null

manager_accessMANAGER               .1.3.6.1.6.3.3.1.3.xxx.xxx.xxx.xxx.2
snmpUDPDomain                   yyy.yyy.yyy.yyy        7788
noAuth                      noPriv
300                        484
00000000
00000000000000000000000000000000      Null
00000000000000000000000000000000      Null
```

context.conf

Defines the contexts between agent and manager parties. In this file, xxx.xxx.xxx.xxx refers to the network address of the agent station. No other information needs editing. Sample lines from the file appear below:

```
# contextName   contextIdentity
# contextViewIndex contextLocalEntity contextLocalTime
# contextDstPartyIndex  contextSrcPartyIndex    contextProxyContext

context_1_accessMANAGER        .1.3.6.1.6.3.3.1.4.xxx.xxx.xxx.xxx.1
1                    Null        currentTime
2              1                        .0.0

context_2_accessMANAGER        .1.3.6.1.6.3.3.1.4.xxx.xxx.xxx.xxx.2
1                    Null        currentTime
4              3                        .0.0
```

view.conf

Defines the supported MIB view. The Signaling Gateway SNMP Agent supports the following subtrees:

- accessMANAGER (1.1.3.6.1.4.1.1056.1) - NewNet Distributed7-related MIBs
- sg (1.1.3.6.1.4.1.1056.2) - Signaling Gateway application-related MIBs

acl.conf

Defines the access privileges for each source party, destination party, and context triple. If the default agent manager party definitions are used, then there is no need to edit this file. Whenever new parties are defined, the required privilege information must be added to this file. Sample lines from the file appear below:

 # targetParty sourceParty context privileges
*# and privileges is [gnsrt]**
 # where g = get, n = getnext, s = set, r = get Response,
 # b = bulk, i = inform, u = trap2

agent_accessMANAGER manager_accessMANAGER context_1_accessMANAGER gnb

manager_accessMANAGER agent_accessMANAGER context_1_accessMANAGER ru

4.5.6.2 Starting the SNMP Agent

The SNMP Agent must be started ***BEFORE*** the system can be managed remotely from a management station. Run the following command to start an SNMP Agent for a particular signaling point:

AccessSNMP -v version sp

where

- *version* is the SNMP version, either **1** or **2**
- *sp* is the signaling point number from 0 to 7.

4.6 Using AccessMOB

4.6.1 Introduction

The Graphical User Interface (GUI) for Signaling Gateway is called the Managed Object Browser (MOB). This interface displays the hierarchical model of the Signaling Gateway on the screen and permits convenient access through *point-and-click* mouse sequences instead of through typed commands. Managed Objects (MO) presented in tree form, can be selected at will for viewing or modifying.

4.6.1.1 Requirements

To access the Managed Object Browser, you will need the following:

- Signaling Gateway software;
- X Window System Release 5 (X11R5) server software, or equivalent, e.g., OpenWindows for Solaris, and the corresponding shared (run-time) libraries;
- Motif Version 1.2.4 shared (run-time) libraries. (See the *Environment Variables* section);
- Motif or another compatible window manager (e.g., OPEN LOOK for Sun systems).

For additional details on using the Motif environment, refer to the Open Software Foundation's *OSF/Motif User's Guide*.

For details on using OPEN LOOK, refer to your Sun system documentation.

4.6.1.2 Environment Variable Settings

Environment variables must be set before running the Managed Object Browser, if they were not set at installation. The following commands are given for the C Shell.

1. The ***$EBSHOME*** variable must be set to the directory where the Distributed7 software was installed. If it is not set, enter the command:

```
setenv EBSHOME <install_directory>
```

(`<install_directory>` should be replaced with the actual directory path where the software is installed.)

***Important**: **$EBSHOME** can be up to 1024 characters.*

2. Check that the path in your ***.chrc*** file is set up to run the Distributed7 software. If not, set the ***$PATH*** variable with the following command:

```
setenv PATH ${PATH}:$EBSHOME/access/bin
```

3. Set the ***$DISPLAY*** variable to the host name of your machine (network node name) using the command: `setenv DISPLAY <hostname>:0.0`
This command can be placed in your ***.cshrc*** file.

4. The ***$LD_LIBRARY_PATH*** environment variable MAY have to be set to access shared libraries at run-time. For example, if an error such as *fatal: libXm.so.4: can't open file: errno=2* occurs, then the ***libXm.so*** shared library for Motif Version 1.2.4 cannot be found. In this case, the variable must be set according to one of the two methods below.

a. For a variable that has other settings, enter:

```
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:$MOTIFHOME/lib
```

b. For a variable that has not been set previously, enter:

```
setenv LD_LIBRARY_PATH $MOTIFHOME/lib
```

*(**$MOTIFHOME** is an environment variable which represents the location of the Motif installation directory for your particular systeme.g. **/usr/dt** for Sun platforms Consult your system administrator.)*

***Important**: The X and Motif shared libraries used by AccessMOB are **libXm.so**, **libXt.so**, **libX11.so**, and **libXext.so**. If any of these libraries reside in non-standard directories on your system, their location must be determined and the location <u>must</u> be added to the **$LD_LIBRARY_PATH** variable as described above. (Consult your system administrator for help in finding the location.)*

5. Place these settings permanently in your ***.cshrc*** file.

4.6.1.3 Conventions

The following are the conventions used within this chapter:

- File names and dialog box buttons within paragraphs appear in *Bold Italic*;
- Commands that must be typed in appear in **`Courier Bold`** while the options of the command appear in `Courier` that is not bold;
- Menu names and options appear in Bold.

- Point codes are defined as Network-Cluster-Members, for ANSI versions. For ITU/CCITT versions, point codes are defined as Zone-Network-Signaling Points;
- Left and right mouse buttons will be referred to as LEFT and RIGHT.
- DOUBLE CLICK means a rapid press-release-press-release of the mouse button.

Important*: If the mouse buttons or other functions do not seem to operate as described in this manual, you can reset the entire environment to use the default behavior. To do this, press these four keys simultaneously:* ALT + CTRL + Shift + !

4.6.1.4 Starting the Managed Object Browser

To run the Managed Object Browser, the Signaling Gateway software must be running. The Managed Object Servers that control the MTP and SCCP managed objects will be running. Other user part managed objects, such as the ones for ISUP, can also be browsed with the MOB if their Managed Object Servers are running. (See the Initial Configuration chapter.)

The command to start the Managed Object Browser (MOB) is:

```
AccessMOB sp
```

The **sp** argument is the signaling point number (0, 1, 2, 3, 4, 5, 6, or 7) of a Signaling Gateway logical node that is already running and needs to be configured. It should be a number that was used with an ***upmd*** command.

The Managed Object Browser registers non-exclusively with the Signaling Gateway environment. Multiple copies of the MOB can exist for the same signaling point, either on the local host or across the distributed network.

The Managed Object Browser can be stopped with the Ctrl c key combination or the **Exit** option under the **File** menu. It will also be stopped automatically when the software is stopped with the ***ebs_stop*** command.

Note*: If the status of a Managed Object Server changes (for example, one of the daemons is started after AccessMOB), AccessMOB is automatically updated. It does* not *have to be restarted.*

4.6.2 Managed Object Browser

The managed object browser consists of a main window and dialog boxes. The operation mode and the managed object are selected at the main window. Then, dialog boxes appear to specify the unique managed object instance and perform the operation.

All of the dialog boxes have the same components and display information in the same manner. The parameters of a managed object are shown in a list, with a text field next to each parameter. The fields show the present values of the parameters. This value may or may not be changed by the user depending on the context. If an parameter's value cannot be changed, the parameter is shown in grey and it cannot be clicked on with the mouse.

Generally, dialog boxes pop up at full size but can be resized smaller, if desired. When necessary, scroll bars are provided to allow viewing lists that are too large to fit in a normal window. The scroll bars are activated with the mouse.

4.6.2.1 Window Managers

While window manager functions will not be discussed in this manual, knowledge of them will allow the most flexible use of the Managed Object Browser. For example, the main window of the Managed Object Browser can be *iconized* to free up screen space while viewing other subwindows. All viewing windows of the Managed Object Browser can be sized and arranged as desired. Minimizing and resizing is done using the window border or the border menu of the window manager.

The window manager controls the screen and the inputs from the mouse and keyboard to the Managed Object Browser. For example, to accept an input, the desired window of the Managed Object Browser must be selected. To select the window, the user would either click on the window with the mouse or simply move the mouse pointer inside the window, depending on the window manager. A selected window will have a color change in the window border or some other visual indication.

The program for the window manager can be started by entering the name at the bottom of the ***.xinitrc*** file that exists in the home directory. The name of the Motif Window Manager is ***mwm***. The name of the OPEN LOOK Window Manager is ***olwm***. The program must ***NOT*** be run in the background (***do not*** use the *&* with the command).

4.6.2.2 Accessing Menus

The menus of the MOB can be accessed using the mouse to click and drag. The menus may also be accessed by using the keyboard. However, if NUM LOCK is on, the keyboard will appear to be disabled.

Two ways exist to access menus using the keyboard instead of the mouse. The first method pops up a menu so that a selection can be chosen visually. By pressing the ◆ (*Meta* or diamond) key and then the letter key of the menu name (e.g. F for File), that menu will display its choices. When the menu choices have been displayed (either by the mouse or a key combination) a choice can be selected by pressing the key of the underlined letter (e.g. E for Exit).

In a menu, keyboard actions also include using arrow keys (↑ ↓ ← →) to move the cursor, the Ret or Space keys to activate, and Esc to cancel.

The second method of accessing menus is through *menu accelerators*. Menu items can be directly selected WITHOUT going to the menu by using [Ctrl] key combinations. The combinations are identified in the menus next to the associated menu item for which they apply. They are also provided in the following subsections. The Main Window must be selected as the current window for the key combinations to work. If [CAPS LOCK] is on while running the program, the [Ctrl] key menu accelerators will be disabled. They require lower case.

4.6.2.3 Using the Mouse

The following list summarizes the valid mouse actions:

- Clicking the LEFT mouse button once activates an operation in the current mode.
- Clicking the MIDDLE mouse button shows or hides the subtree of a managed object node.
- Pressing the RIGHT mouse button brings up a menu to choose an operation from a mode that is not in the current mode.
- Double clicking the LEFT mouse button opens a view box of all instances when in View mode.
- Pressing [Shift] and clicking the LEFT button, when in View mode, opens a view box of all instances.
- Pressing [Shift] and clicking the RIGHT button, when in a mode other than View, opens the popup menu for selection of the view operation to view all instances.

Important*: If the mouse buttons or other functions do not seem to operate as described in this manual, you can reset the entire environment to use the default behavior. To do this, press these four keys simultaneously:* [ALT] + [CTRL] + [Shift] + [!]

4.6.2.4 Entering Data in the Dialog Box

Dialog boxes are used to enter data for a managed object selected from the main window. First, a unique instance of the managed object is identified in a key dialog box. Then, another dialog box will appear in which to perform an operation. The data entry into any of the dialog boxes follows the same general rules.

A field can be selected by clicking on it with the mouse. Movement between the fields can also be accomplished using the [Tab] key to go down the list or the [Shift] + [Tab] combination to go up the list. The [Ret] key acts the same as the [Tab] key. Each time [Ret] is hit, the next field down on the list is made active for input.

Within the field, the [BackSpace], [Delete], [Insert], [Home], [End] and the arrow keys ([↓] [↑] [→] [←]) may be used for editing. The mouse may also be used to point and click directly. Copy and Paste operations are available using the LEFT and MIDDLE buttons of the mouse or by using the keyboard.

A *Range* or *Set* menu that appears at the end of the field can also be used to input the data. Please see Range Menus and Set Menus and Set Type Values for more details.

After data is entered or changed, the ***Apply*** button must be selected to complete the operation. The mouse can be used to click on the button. If the ***Apply*** button is indicated as the current button by an outline around it, either the Ret or Space key can be pressed to complete the operation.

Selecting the ***Cancel*** button closes the dialog box without making any changes. Usually this is done by clicking on the button with the mouse. However, if ***Cancel*** is the outlined button, then either the Ret or Space key can be used.

Range Menus

A *Range* menu occurs on the right side of an integer field if the allowed range of values is small enough (e.g. 1 - 32). It lists the allowed values, from minimum to maximum. The *Range* menu is viewed by clicking on the ***Range*** button with the LEFT mouse button or by pressing the ◆ R (Meta or diamond key and R) key combination when the desired field is the active field.

A value can be selected directly from the menu by using the mouse. That value is transferred into the text box, eliminating the need to type it. Within the *Range* menu, the up and down arrow keys (↑, ↓), Ret, Space, and Esc may be used.

Set Menus and Set Type Values

A *Set* menu occurs on the right side of a string or numeric field if it is restricted to a small number of valid values (e.g. ON, OFF). The *Set* type is provided to reduce the possibilities of error and the amount of typing required for a string value. The *Set* menu is viewed by clicking on the ***Set*** button with the LEFT mouse button or by pressing the ◆ R (Meta or diamond key and R) key combination when the desired field is the active field.

A value can be selected from the menu using the mouse. That value is transferred into the text box, eliminating the need to type it. Within the *Set* menu, the up and down arrow keys (↑, ↓), Ret, and Space may be used. Esc

When typing in the value for this type of field, only the initial characters which uniquely identify a value from the set are required. The value will be automatically completed and accepted. For example, in the set {ON, OFF} it is necessary to type the first two letters to identify the choice.

For long strings, the Esc key may be used to perform a partial *completion* while you are typing a set value. For example, in the following set,

{sbs332, sbs334, sbs370, sbs372, ax7000, pri200}

typing an s followed by Esc will automatically provide the substring **sbs3**. This is the maximum substring identified by the given character. Then, you must type the rest of the characters needed to uniquely identify one element of the set. The Esc key may complete the set value if that is possible. However, using the Esc key to complete a value is not necessary. The automatic completion occurs when you move to a different text entry field or when the ***Apply*** button is selected. If a value could not be completed to form a valid set member value, an error message will appear.

***Important**: For values which have multiple completions (i.e. the set {I, II, III}), any substring which is entered will be accepted. Be careful to enter the appropriate value.*

4.6.2.5 Managed Objects Parameters

Managed objects are a functional or physical resource of the system, such as subsystems or link sets. For example, each box in the Main Window of the Managed Object Browser (Figure 4-12) is a managed object. Each managed object has a set of operations (add, modify, delete, view) that are allowed to be performed on it. An individual instance of a managed object, such as a specific link set, is defined by its parameters. They provide the managed object with a unique identity.

A more in-depth description of managed objects can be found in *Chapter 2: Overview*. However, information about parameters are provided in the following subsections to provide a better understanding of data entry in the Managed Object Browser.

Key Parameters

Key parameters are identified by a key symbol as seen in Figure 4-13 on page 4-57. In the key selection dialog box, only the key parameters are listed for input by the user. The other dialog boxes show the key values, but do not allow them to be changed. They cannot be changed because they act as the *title* of a particular instance.

Data Types

The data type of an parameter identifies whether it must be a numerical value, a point code value, or a general alphabetical string. The data type can either be a *Set* type or a *Range* type which is deduced from the range information in the **Range** popup menu next to the parameter's value field. Information on sets or ranges can also be found in the MMI/MML chapters of this manual. Menus can be viewed by clicking on the ***Range*** or ***Set*** button with the LEFT mouse button (see Figure 4-13). These menus list the complete set of allowed values or the range of values, from minimum to maximum. Values can be selected directly from these menus (see Range Menus).

Values entered in the fields are checked for the data type and the range. Error messages will indicate any illegal values that need to be corrected. The constraints of each data type are described below. Chapter 2: Overview of this manual contains tables which identify the data types of all managed object parameters.

Integer

- Must be an integer value within the range shown; the minimum and maximum values are valid.
- May include a K or M suffix (lower or upper case) after the value to indicate thousands. *This is NOT binary (1024).*

Point Code

- Must be three sets of integers with a dash between each set.
- Must be within the range and format required by the protocol version (ANSI or CCITT) and identified by the Range popup menu.

Set Type

- Must be chosen from a given list of numeric or alphanumeric values.
- Displays allowed values in the Set popup menu.
- Requires only initial characters to be typed to identify a value (see Set Menus and Set Type Values).

String

- May be any set of alphanumeric characters, except the asterisk *. (See the Wild Cards section.)
- Must have a character length within the specified range.

Wild Cards

The special character, *, represents a *wild card* value. A wild card means that ALL existing values of a particular parameter are selected. It can only be used for KEY-type parameters chosen in the **Keys** selection dialog. The * can be used for viewing instances of a Managed Object as a group. If there is one key, then all of them are viewed. If there are several keys, the instances may be viewed by category.

The * character can be used for any data type, *Integer, Point Code*, *Set Type,* or *String*. However, the following limitations on wild card usage exist:

- Existing Managed Objects only accept ONE wild card in the key values list, if there are multiple keys. Refer to the specific MMI/MML commands to see which Managed Object parameter key will accept a wild card as a value.
- Wild cards can only be used in the View operation. The other operations require full specification of a unique instance.
- Wild cards can only be used for a key parameter.

Access Types

An access type determines the type of access a user has to a parameter of a managed object. They identify which operations the user is allowed to use on a parameter - view, add, delete, or modify. Illegal operations will result in an error message. *Chapter 2: Overview* of this manual contains tables which identify the access types of all managed object parameters. The chapters on MMI/MML commands also identify which parameters are valid for a particular operation. The four access types are defined as follows:

READ-WRITE

- Parameter is always displayed.
- Parameter can be modified in Add and Modify dialog boxes.
- Entry of a value can be optional.

READ-ONLY

- Parameter is always displayed.
  (Usually status information from the Managed Object Server)
- Parameter ***cannot*** be modified.

READE-CREATE

- Parameter is always displayed.
- Parameter can be defined in Add dialog boxes.
- Parameter ***cannot*** be modified in Modify dialog boxes.
- Entry of a value can be optional.
- Parameter is or behaves like a key parameter but does not have to be one.

WRITE-ONLY

- Parameter can be supplied by the user in Add, Delete, and Modify dialog boxes.
- Entry of a value can be optional (default value exists).
- Entry is usually a setting for an operation. (e.g. a range of instances to Add or Delete)

The special WRITE-ONLY parameter type is identified by a *pen* symbol on the left side of the text entry field.

4.6.3 Managed Object Browser Windows and Operation

4.6.3.1 The Main Window

Figure 4-12 is an example of the Managed Object Browser's main window. If other MO Servers, such as ISUP, are running, the associated managed objects will appear.

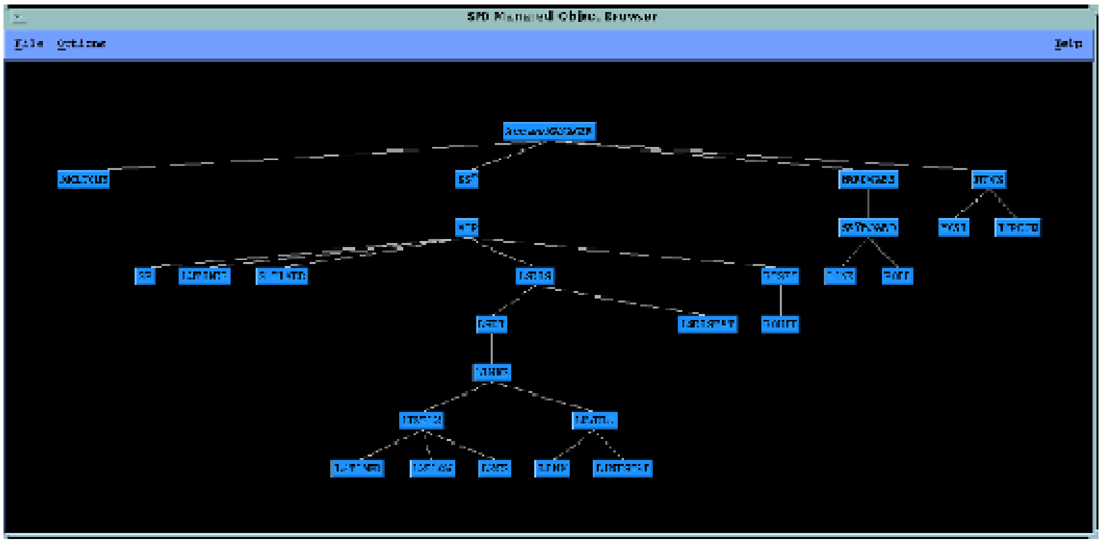


Figure 4-12: MOB Main Window

The main window of the MOB contains a tree representing the managed objects in your Signaling Gateway environment. The subtree of a managed object node can be shown by a single click of the MIDDLE mouse button on the desired node. A displayed subtree can be hidden by the same action. Subtrees can also be shown or hidden by holding down the RIGHT mouse button while over the node and selecting the **Show/Hide** choice from the five-color popup menu that appears.

The main window supports the following actions and inputs:

- Pull down menus selected with the LEFT mouse button
- Keyboard *quick key* combinations that perform the menu actions
- Mouse point and click operations on the nodes of the tree

Three menus exist at the top of the Managed Object Browser main window. They are *File*, *Options*, and *Help*. A menu is accessed by placing the mouse on the menu name and clicking the LEFT mouse button. The available options will be displayed.

File Menu

The only choice in the **File** menu is **Exit**. When **Exit** is selected, all open dialog windows are closed and the program is ended. The Ctrl Key combination can also be used.

Options Menu

The **Options** menu allows the user to select the mode and to select the display style for the tree. The key combination for the choice is shown beside it. The choices are:

- View - to choose the mode for viewing managed objects (Ctrl V)
- Modify - to choose the mode for modifying an instance of a managed object (Ctrl M)
- Add - to choose the mode for adding an instance of a managed object (Ctrl A)
- Delete - to choose the mode for deleting an instance of a managed object (Ctrl D)
- Refresh Tree - to refresh the managed object tree when the managed object configuration has changed *(the tree is normally checked and automatically refreshed while the program is running, so using this selection is unnecessary)*
- V Tree - to choose a vertical display of the tree
- H Tree - to choose a horizontal display of the tree
- Dialog Auto Place - to enable an alternate method of positioning new *View* dialogs on the screen (Ctrl P). When this option is set (square indicator appears at left), *View* dialogs are popped up around the edge of the screen, instead of being placed according to your window manager's default placement.
- Change Title - to set a new title for windows (Ctrl T)
  Windows already open when the window title is changed will not display the change but new windows that appear after the change is made will have the new title.

Help Menu

The **Help** menu provides information about the main window and modes. The choices are:

- Help on AccessMOB - describes the main application window
- Managed Object Tree - describes the mouse actions pertaining to the tree
- Keys Dialog - describes the dialog box for entering key choices
- View Dialog - describes the *view* mode and its dialog box
- Modify Dialog - describes the *modify* mode and its dialog box
- Add Dialog - describes the *add* mode and its dialog box
- Delete Dialog - describes the *delete* mode and its dialog box

4.6.3.2 Selecting an Operation Mode

The Managed Object Browser operates in one of four modes, each identified by a specific color. The modes are View (blue), Modify (yellow), Add (green), and Delete (red). When a given mode is active, the managed object tree is shown in the associated color. After selecting a mode, an operation is initiated by a single click of the LEFT mouse button on the desired managed object.

The mode is set using the **Options** menu or a key combination. The combinations are Ctrl A for Add, Ctrl M for Modify, Ctrl D for Delete, and Ctrl V for View, as shown in the menu. These key combinations are called *menu accelerators*. See *Accessing Menus on page 4-48*.

One mode can be quickly accessed from another for a single operation (e.g. to modify a single managed object while in view mode). To access a mode in this way, the RIGHT mouse button should be held down while the cursor is over the desired managed object. A five-color popup menu appears over the node while the button is still held down. The cursor should be moved to the desired operation and the button released to choose that operation, similar to a menu. The overall mode does not change after the operation is complete.

Important*: <CAPS LOCK> cannot be used for the above actions. If <CAPS LOCK> is on while running the program, the* Ctrl *key menu accelerators will be disabled. They require lower case.*

Important*: <NUM LOCK> cannot be set while the Managed Object Browser is running. If <NUM LOCK> is on, the keyboard will appear to be disabled.*

4.6.3.3 Selecting Managed Objects

Once the mode is selected, you must pick the managed object to perform an operation on. An operation is initiated by a single click of the LEFT mouse button on the desired managed object box in the main window. Remember, if the desired managed object is in a subtree that is hidden, simply click the MIDDLE mouse button on the node of the managed object, then select the managed object.

Note*: Some managed objects in the tree have no associated operations (example Distributed7 at the top of the tree). These objects only serve as parents for other managed objects. If one of these objects is selected, an error message popup window will appear.*

For Modify and Delete operations, a managed object instance can also be selected from the managed object's *View* dialog list. The view list displays all instances of a managed object. This method is described in *Selecting Other Modes From the View Dialog Box*.

After selecting a valid managed object, a popup key selection dialog box similar to Figure 4-13 appears. The dialog box shows the managed object name that was selected and the key parameter(s) for which a value must be supplied. Only the key parameters are listed in this box. Other dialog boxes will show the key values, but do not allow them to be changed. In this dialog box, space is available next to each key parameter name to enter the key value. If more than one key parameter appears in the list, all must be specified in order to uniquely identify the single instance. Entering data in a dialog box is covered in Section 4.6.2.4 on page 4-49.

The *Range* or *Set* menu on the right side of each key field contains the possible values that can be entered. The field's menu is viewed by clicking on the button with the LEFT mouse button. The field's value should be a new value when in Add mode, or a known value when in View, Delete, or Modify modes. Normally, only one instance may be added, modified, or deleted at a time.

The managed object instance is chosen by clicking on the ***Apply*** button of the dialog box. When this dialog box closes, an operation dialog box appears containing information for the chosen managed object instance.

The selection can be ended by clicking on the ***Cancel*** button.

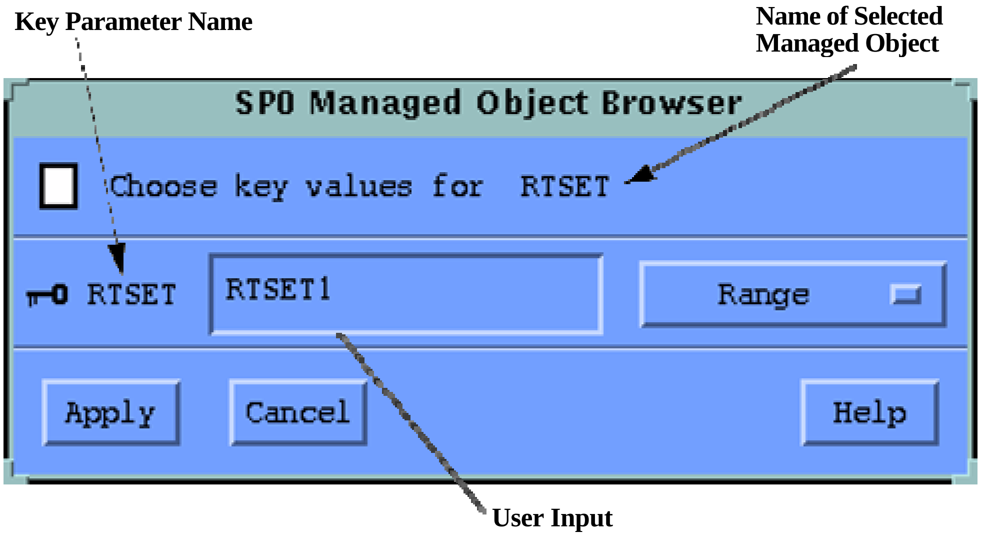


Figure 4-13: Key Selection Dialog Box

For the view operation, managed object instances can be viewed all at once instead of individually through the key dialog box. By double clicking the LEFT mouse button or pressing Shift and clicking a mouse button on the desired managed object class, the keys dialog box will be bypassed and a window will appear showing ALL existing instances of a managed object. This viewing operation can be invoked at any time, even with a key dialog box open.

Important*: Only one Modify, one Add, and one Delete dialog box can be open at one time, but an unlimited number of View dialogs can be open simultaneously.*
While a Key dialog box remains open, no other dialogs can be opened except View dialogs for ALL instances.

4.6.3.4 Operation Dialog Boxes

The following sections show the four types of dialog boxes that exist. Each type of dialog box will have unique action buttons. During use, the contents of the dialog boxes will differ based on the managed object that was selected. However, the functions remain the same.

Each dialog box type shows the color code associated with the operation at the top of the box, to the left of the operation name. The managed object class is identified to the right of the operation name.

Specific characteristics of each dialog box are described following each figure.

Add Dialog Box

Figure 4-14 shows a sample Add Dialog Box for the managed object, *RTSET*. The box will be similar for any managed object. The list of parameters and which ones cannot be changed will be unique to each managed object.

Figure 4-14: Add Dialog Box

When the Add Dialog Box first appears, the key parameter values and any defined default values are shown. Default values can be accepted or changed. Some empty fields require an entry, while others do not. The chapters on MMI/MML commands will identify those parameters that have default values or are optional.

Key field values can be changed. If a different key value is desired, select the field using the mouse, the Tab key, or Shift Tab then edit the value. Any parameters shown in grey (e.g. CONG in Figure 4-14) cannot be set by the user.

The ***Apply*** button completes the Add operation to create an instance of the managed object. The ***Cancel*** button exits from the Add operation without making any changes. The ***Reset*** button clears the entries in all fields and resets the key fields to their original values.

To add a managed object instance:

1. Select the Add operation mode in the main window. Either select **Add** from the *Options* menu or press Ctrl A
2. Click the LEFT mouse button once on the managed object in the main window.

3. Enter values for all fields in the key selection dialog box that pops up. (Figure 4-13 on page 4-57)
4. Enter the values for all required and desired fields in the Add operation dialog box.
5. Click the ***Apply*** button.

Modify Dialog Box

Figure 4-15 shows a sample Modify Dialog Box for the managed object, *LSET*. The box will be similar for any managed object. The list of parameters and which ones cannot be changed will be unique to each managed object.

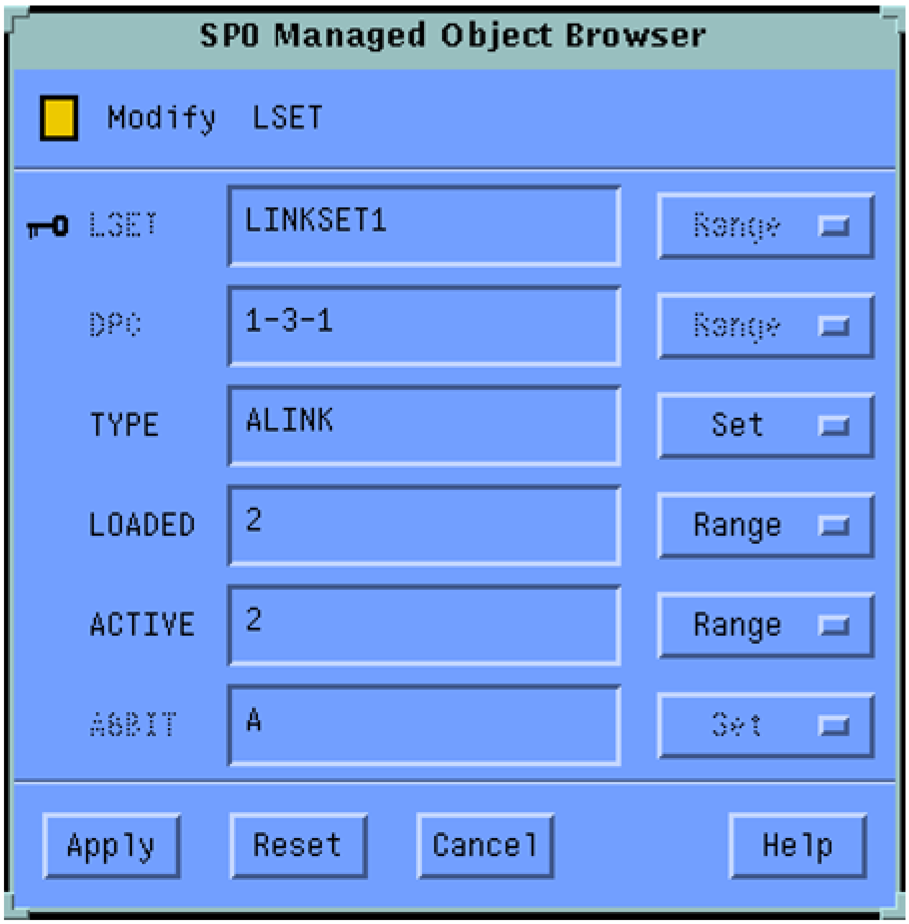


Figure 4-15: Modify Dialog Box

When the Modify Dialog Box appears, the current parameter settings are displayed in the fields. Any parameters shown in grey ***cannot*** be changed by the user. The chapters on MMI/ MML commands describe the parameter fields and the valid settings.

The ***Apply*** button completes the Modify operation. The ***Cancel*** button exits from the Modify operation without making any changes. The ***Reset*** button sets all entries in the fields back to their original settings before any changes were made.

To modify a managed object instance:

1. Select the Modify operation mode in the main window. Either select **Modify** from the *Options* menu or press M
2. Click the LEFT mouse button once on the managed object in the main window.
3. Identify the instance through the key selection dialog box that pops up. (Figure 4-13 on page 4-57)
4. Enter the values to be modified in the Modify operation dialog box.
5. Click the ***Apply*** button.

Delete Dialog Box

Figure 4-16 shows a sample Delete Dialog Box for the managed object, *LSET*. The box will be similar for any managed object. The list of parameters will be unique to each managed object.

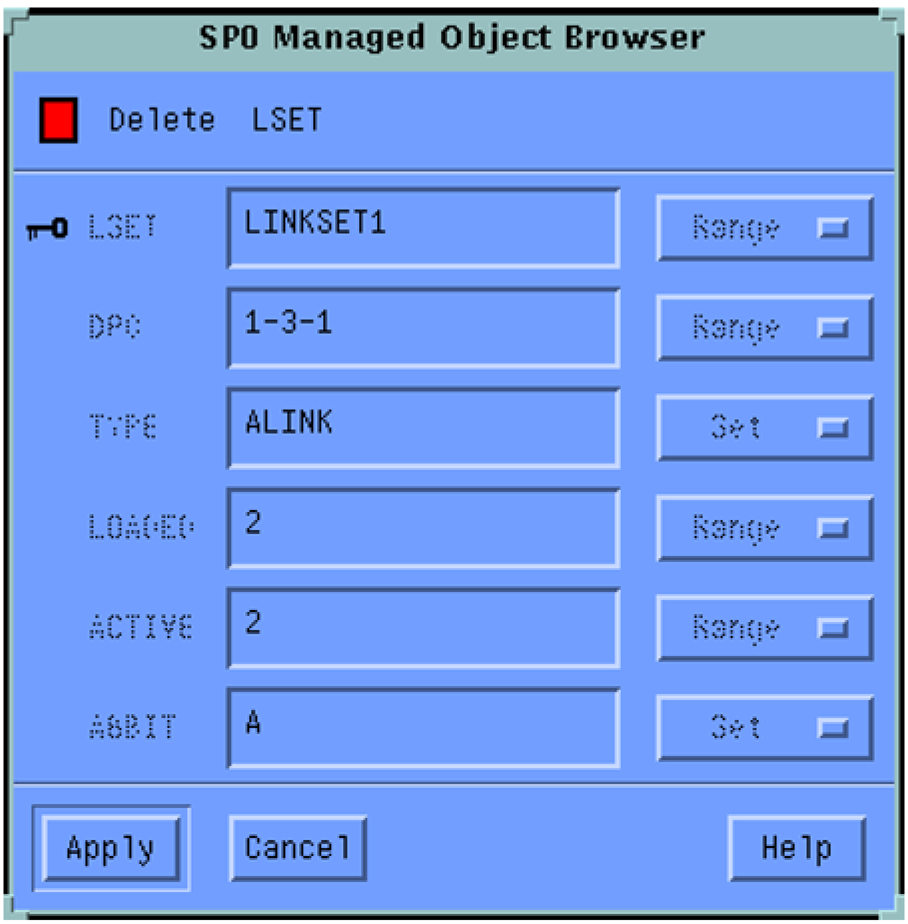


Figure 4-16: Delete Dialog Box

When the Delete Dialog Box appears, all parameter settings are displayed, but shown in grey. However, if WRITE-ONLY parameters exist, a value can be entered (e.g. a range to be deleted).

The only action to take is to select the ***Apply*** button to delete the instance of the managed object, or to select the ***Cancel*** button to exit from the Delete operation without deleting the instance.

To delete a managed object instance:

1. Select the Delete operation mode in the main window. Either select **Delete** from the *Options* menu or press Ctrl D.
2. Click the LEFT mouse button once on the managed object in the main window.
3. Identify the instance through the key selection dialog box that pops up. (Figure 4-13 on page 4-57)
4. Enter any values that are writable in the Delete operation box (e.g. range to be deleted)
5. Click the ***Apply*** button.

View Dialog Box

The view operation can be performed for a single instance of a managed object or for all instances of a managed object.

To view a single instance:

1. Select the View operation mode in the main window. Either select **View** from the *Options* menu or press Ctrl V.
2. Click the LEFT mouse button once on the managed object in the main window.
3. Identify the instance through the key selection dialog box that pops up. (Figure 4-13 on page 4-57)

A window will appear showing the parameters of the single instance.

To view all instances:

1. Select the View operation mode in the main window. Either select **View** from the *Options* menu or press Ctrl V.
2. Double click the LEFT mouse button or press Shift and click the LEFT mouse button on the desired managed object in the main window. The Shift key must remain held down until the mouse button is released.

A window will appear showing ALL existing instances of a managed object. This viewing operation can be invoked at any time, even with a key dialog box open.

When in other operation modes, a View dialog with all instances can be opened by holding down the Shift key while clicking the RIGHT mouse button and then selecting the View operation from the menu that appears.

Figure 4-17 shows a sample View Dialog Box for the managed object, *LSET*. The box will be similar for any managed object, but the output will differ.

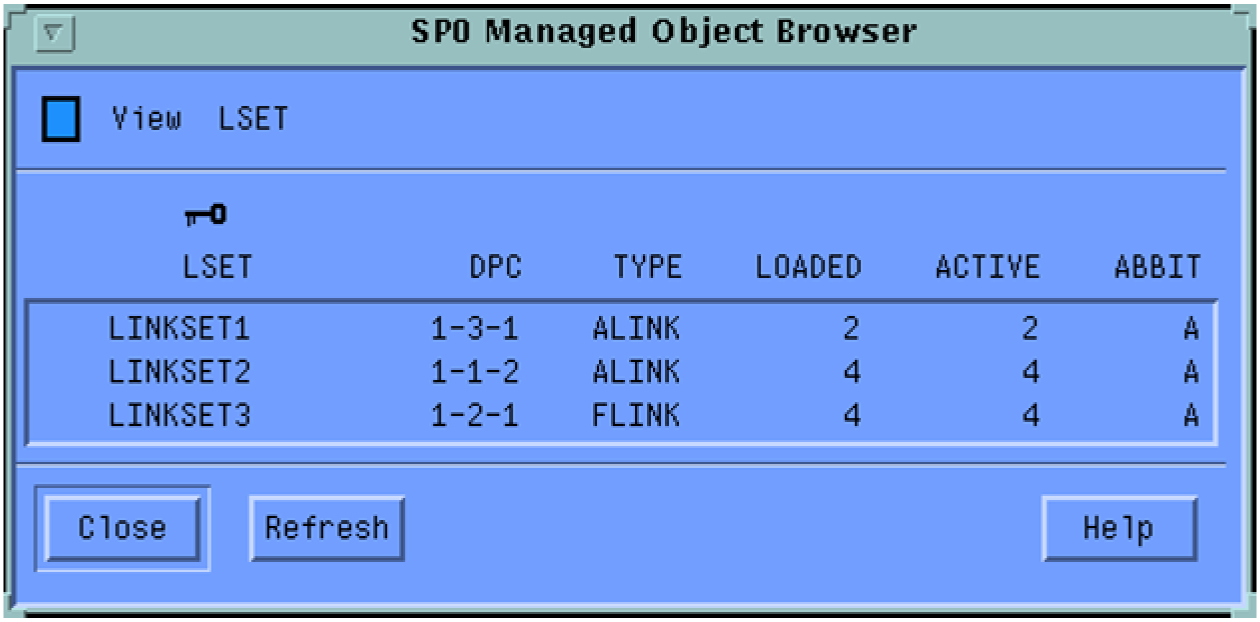


Figure 4-17: View Dialog Box

The View Dialog Box can display one or multiple instances of a managed object. No information can be changed on this window.

A refresh of the screen occurs at a predetermined time interval to retrieve any changes in current information from the Managed Object Server. The ***Refresh*** button can also be selected by a mouse click to force a refresh.

The window will stay open until the ***Close*** button is selected. The window may stay open while other dialog boxes are being used.

Note*: View Dialog Boxes are positioned on the screen according to the* ***Dialog Auto Place*** *option in the* ***Options*** *menu (**Options Menu on page 4-55**). When the option is set* ***on*** *(the default), View Dialog boxes will be popped up around the edge of your screen in a tiled, non-overlapping manner, for convenience of viewing. When set* ***off****, the dialog boxes will be placed according to your window manager's current default placement.*

Selecting Other Modes From the View Dialog Box

The View Dialog Box allows other dialog boxes to be called up for a selected instance in the box. Selecting a displayed instance from a View Dialog Box will create a new dialog box for that individual instance in the current operation mode of the main tree (View, Modify, Add, or Delete). This selection action is allowed in any View Dialog Box, whether there is one instance displayed or a list of instances. For example, an instance could be selected for modification from a *view-all* list, without having to enter its key values.

To create a new dialog box in the current mode from a view box, click on any instance in the dialog display area to select it. Then, click on it again to create a new dialog. A double click combines these two actions.

To create a dialog in any chosen mode, press on the instance with the RIGHT mouse button but do not release it. A popup menu will appear. Use the cursor to choose View, Modify, Add, or Delete. When the mouse button is released, a dialog box in that mode for the selected instance will appear.

The following keys and key combinations may also be used:

- The Tab key or Shift Tab can be used to move between the action buttons and the display area of the dialog box.
- The , , ↑ ↓ Ctrl Home and Ctrl End keys are used to move through the list to select an instance.
- The Ret or Space key creates the new dialog for the currently highlighted instance.

4.6.4 Error Messages

Error messages are issued either by the Managed Object Browser or by the Managed Object Servers. The Managed Object Browser performs syntactic and range checks on the entered values and produces error messages when problems occur.

If no syntax or range errors occur at the MOB level, the information is sent to the Managed Object Server. If the operation could not be performed, an error message will be returned indicating a failure and the reason for the failure.

Managed Object Browser Error Messages

Cannot attach to apmd environment

SPM connection error

Either the signaling point software or the APM daemon is not running. Start the software before executing AccessMOB.

Inapplicable operation: <operation>

The selected operation, ADD, MODIFY, or DELETE, is not permitted or not meaningful for this Managed Object.

Privileged operation: <operation>

The selected operation, VIEW, ADD, MODIFY, or DELETE, is not permitted for this user on this Managed Object. The operation is protected and can be performed only by a user on the Managed Object's access control list.

No operations defined for <MO-name>

The node that was selected on the managed object tree does not correspond to a managed object that can be viewed, created, or modified by the user.

<MO-name> Managed Object Server not available

The daemon process responsible for <MO-name> is not running (*upmd, snmd, scmd, or isupd)*. The required daemon should be started.

<MO-name> Managed Object Server communication timeout

Communication failed between the Managed Object Browser and the daemon process responsible for <MO-name>. To resolve, the operation can be retried, the MOB can be restarted with AccessMOB, or network/system problems can be investigated.

<MO-name> has no instances - Press OK to close View dialog

A view dialog box became invalid when an automatic refresh (or forced refresh) occurred. The box becomes invalid when instances of a managed object no longer exist; for example, all instances of a Managed Object may have been deleted since the box was last updated. The dialog box will be closed once *OK* is selected.

Managed Objects changed - Press OK to close invalid dialogs

One or more dialog boxes became invalid when an automatic (or forced) refresh of the managed object tree occurred. This means that the tree has changed and managed objects that were being accessed are no longer available. The affected dialog boxes will be closed once OK is selected, but other dialog boxes will remain open.

Instance already exists

An ADD operation was attempted using one or more key parameters that match an existing instance.

<operation>: Wildcard not allowed

A wildcard is only permitted for the VIEW operation. For ADD, MODIFY, or DELETE, a specific instance must be chosen by supplying all key values.

Wildcard not allowed for: <parameter-name>

The named parameter does not accept a wildcard value. Only certain keys accept a wildcard, which is a characteristic defined for the specific Managed Object.

Value out of range: <value>

The value that was entered is not within the specified range. The Range popup menu and the MML chapter identify the valid range of values.

Integer required for: <parameter-name>

The value entered for the named parameter is not a valid integer or an integer suffixed with *K* or *M*.

Point code out of range or incorrect format: <value>

The value entered is not a valid point code for the protocol standard (ANSI or CCITT) being used. The Range popup menu shows the correct format and valid minimum and maximum values.

Ambiguous set choice: <value>
Not a valid set choice: <value>

The value must be chosen from the given list of values. Type the last or remaining characters required to uniquely identify the value, or select the item directly from the Set popup menu.

String length out of range: <value>

The string value that was entered is too short or too long. The Range popup menu identifies the valid string length.

No new values were entered

Values were not changed since the last time *Apply* was selected, or a null MODIFY operation is being attempted. The *Cancel* button should be used to exit.

Could not open help file

The applicable *.info* file is not available in the *access/help* directory, or does not have the correct permissions. Check the directory.

Managed Object Server Error Messages

Messages from the Managed Object Server are presented in one of the following forms, depending on the requested operation. The message that is displayed is specific to the actual error and is self-explanatory.

GET VALUES: <message>

MODIFY: <message>

ADD: <message>

DELETE: <message>

Related Information

- Section 6.2.5, AccessStatus on page 6-11

This page is intentionally blank.

Chapter 5: MML Commands

5.1 ChapterOverview

Signaling Gateway software has a set of operating MML commands to configure and maintain the system that are described in this chapter.

There are two basic types of commands used: MML commands and shell commands.

- All MML commands are entered at the **`MML_TH>`** prompt.
- The shell commands are entered at the UNIX command prompt and are used for such things as installing the software, and starting and stopping the system.

This chapter documents just the MML commands and lists some common UNIX commands at the end.

5.1.1 MML Commands

Use the **mml** utility command to start the MML interface from which the operator can enter the commands to configure, administer, and maintain the system.

The MML commands are grouped as the following managed objects**:**

- ***MTP*** Managed Objects configure the MTP layer. The operator must define these parameters before using Signaling Gateway.
  - Table 5-2 lists all MTP managed objects, parameter names, default values, units, value ranges, and operation commands.
- ***SCCP*** configure SCCP user part. The operator must define these parameters before using Signaling Gateway.
  - Table 5-3 lists all SCCP managed objects, parameter names, default values, units, value ranges, and operation commands.
- ***System*** Managed Objects configure and maintain the system.
  - The operator uses these functions to manage hosts and their operation in a distributed network, view and maintain alarms and TCP/IP connections, and manage an MML session.
  - Table 5-4 lists all managed objects, parameter names, default values, units, value ranges, and operation commands.
- ***Signaling Gateway*** Managed Objects configure, maintain and monitor the Signaling Gateway.
  - Table 5-5 lists all managed objects, parameter names, default values, units, value ranges, and operation commands.

Use the **HELP** command to view online information about specific MML commands.

5.1.2 MML Network Element Labels

The MML command labels for network elements are defined as follows:

- Link set, Link, Route set, and Signaling Point name: These are 12-character labels that can include any number of alphanumeric characters, numbers and hyphens.
- Hostname: This is a 15-character label that can include any number of alphanumeric characters, numbers and hyphen.
- Point Codes (DPC, OPC and SPC): Signaling point code are 11-characters labels, formatted as three numbers separated by hyphens (xxx-yyy-zzz), where the numbers represent the following IDs based on the standard:

  ANSI xxx is the Network ID
  yyy is the Cluster ID
  zzz is the Member ID

  ITU xxx is the Zone ID
  yyy is the Area ID
  zzz is the SP ID

ANSI and 24-bit ITU:

Fields	Network/Zone	Cluster/Area	Member/SP
Format	8 bits	8 bits	8 bits
Value Range	0 - 255	0 - 255	0 - 255

16-bit ITU:

Fields	Zone	Area	SP
Format	5 bits	4 bits	7 bits
Value Range	0 - 31	0 - 15	0 - 127

14-bit ITU:

Fields	Zone	Area	SP
Format	3 bits	8 bits	3 bits
Value Range	0 - 7	0 - 255	0 - 7

***Note**: Leading zeros (0) are not necessary.*

5.1.3 Syntax Rules for the Command Line

The following rules apply to the format of command line entries:

1. Decimal values must be typed directly and should only be digits.
2. Hexadecimal values must begin with ***H'***.
3. Octal values must begin with ***O'***.
4. Parameter names can only have alphanumeric characters.
5. Commands have the form: **<Operation>-<Managed Object>**. However, commands such as HELP do not have a managed-object component.

5.1.4 String-Constant Data Entry Method

The data entry of a string constant variable is slightly different than other data entry. Since some of the constant strings are long, a utility is introduced which allows shortened versions of the constants. Users can simply type the first characters that uniquely identify that constant string instead of typing the whole string. For example, it is possible to type **DT** for *DTE*, **N** for *NATIONAL*, or **A** for *ALINK*. Note that the abbreviated part cannot be ambiguous.

MODIFY-SP:NI=INT; (VALID)

MODIFY-SP:NI=I; (VALID)

MODIFY-MTP:SLTC=O; (INVALID)

MODIFY-MTP:SLTC=OF; (VALID)

MODIFY-MTP:SLTC=ON; (VALID)

5.1.5 Case Sensitivity

MML is ***NOT*** case sensitive with respect to command and parameter names. MML converts this input into upper case. However, the values entered for a parameter are case-sensitive. The following figure illustrates the rules of case-sensitivity in MML commands.

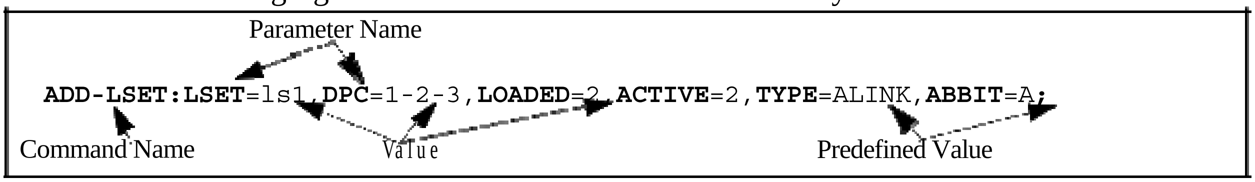


Figure 5-1: Case Sensitivity in MML Commands

- The **command name** and **parameter names** can be typed in either upper or lower case.
- The **values** ARE case sensitive.
  For example, `ADD-LSET:LSET=ls1`... and `ADD-LSET:LSET=LS1`... create two separate and unique link sets called, ls1 and LS1.
- **Predefined values** listed for a parameter, such as ALINK, must be typed in upper case.

5.1.6 Command Syntax

The following explains how the syntax for each command is documented in this manual:

•The **COMMANDS** section lists the commands that can be used with each managed object: ADD, MODIFY, DELETE and/or DISPLAY.

• Each command is followed by the command syntax:

- The *command* and *parameters* are in ***UPPER CASE, BOLD ITALICS*** typeface.

- The *values* for a parameter are lower case, regular typeface.

- These lower case values are listed in the **PARAMETERS** section with a description of what the parameter does, and what can be entered as the value.

- Optional parameters are in square brackets, ***[]***, and are listed after the required parameters.

- The order in which the parameters are entered does not matter.

The following is an example of the command syntax for the SG command:
MODIFY-SG:SPID=spid[,NAID=naid];

Any read-only parameters that appear in the output of the DISPLAY command are NOT listed in the **PARAMETERS** section, but are in a table after the **SAMPLE OUTPUT**. The following is the table that appears after the SG **SAMPLE OUTPUT:**

Table 5-1: SG DISPLAY VALUES

NASTD	MODE
The Network Appearance standard defines the M3UA standard. The value is set based on the protocol that is configured in MTP, and can be one of the following values: • ITU • ANSI • CHINESE	This read-only parameter is the mode of the SG, which can be one of the following values: • **SEP** - SG is running as an SEP • **STP** - SG is running as an STP

5.1.7 Output Messages

MML prints "`<SUCCESS>`" when a command runs successfully. It prints "`<ERROR>::`*`error message`*." if a command fails. MML performs syntactic and range checks on MML commands. MML displays an appropriate error message if a command is syntactically wrong or parameters have an out-of-range value. The following are some of the common error messages and their meanings:

1. `<ERROR>::Could not locate information for command` ***`commandname`*** `in database.`
   The MO Server does not work or MML cannot reach the database files related to this MO. Check the ***$EBSHOME*** environment variable.
2. `<ERROR>::Parameter` ***`[param]`*** `not defined.`
   The wrong parameter was entered for this MML command.
3. `<ERROR>::Missing value of` ***`[param]`*** `parameter.`
   A value should have been entered after = for this parameter.
4. `<ERROR>::Alias` ***`alias`*** `not found.`
   The parameter value, ***alias***, is considered an alias but it cannot be found in alias tables. It should be defined as a new alias.
5. `<ERROR>::Value syntax error.`
   The value of the parameter is syntactically wrong.
6. `<ERROR>::Invalid entry.`
   The command is syntactically wrong.
7. `<ERROR>::Unknown command.`
   This command is not defined. Check the command that was entered again.
8. `<ERROR>::Parameter` ***`[param]`*** `should be in range` *`n-m`*`.`
   The parameter should be in the given range.
9. `<ERROR>::Parameter` ***`[param]`*** `should be less than or equal to` *`m`* `in length.`
   Check the length of this string parameter.
10. `<ERROR>::Point code is not valid.`
    The point code you entered is not suitable for the protocol you are using. For example, 9-1-9 is invalid for ITU.)
11. `<ERROR>:: Managed Object Server not available.`
    The daemon process responsible for the command is not running (upmd, snmd, scmd, or isupd). The required daemon must be started.
12. `<ERROR>::lset1 should be A and lset2 should be B PLANE`
    The link sets must be in different planes. *(For Japanese networks only.)*
13. `<ERROR>::MTP protocol is not set yet.`
    The ADD-MTP command sets the protocol of the signaling point so the point code operations can only be performed after running the ADD-MTP command.

Table 5-2: MTP Configuration Managed Objects

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
LINK (LINK Managed Object - see Section 5.2.1 on page 5-36)	LINK	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	LSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	
	SLC	-	unsigned integer	0 - 15	
	PRIORITY	0 is the highest priority and the highest available priority is the default if this parameter is not entered.	unsigned integer	0 - 15	
	L2ECM	BASIC/PCR	-	-	
	PCRN1	-	unsigned integer	0 - 127	
	PCRN2	-	unsigned integer	0 - 9999	
	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	BOARDNM	sbs334 pci334 pci3xpq pci3xapq cpc37xpq pmc8260 artic8260 vbrd	alphanumeric characters	-	
	INST	-	unsigned integer	0 - 7	
	PORT	depends upon the BOARDNM: sbs334 0 - 3 pci334 0 - 3 pci3xpq 0 - 23 pci3xapq 0 - 23 cpc37xpq 0 - 23 pmc8260 0 - 63 artic8260 0 - 63 vbrd 0 - 31	unsigned integer	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
LSET (Link set Managed Object - see Section 5.2.2 on page 5-40)	LSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	DPC	-	alphanumeric characters	Format: x-y-z (see page 5-2 for more information about the point code format).	
	TYPE	ALINK BLINK CLINK DLINK ELINK FLINK	characters	-	
	EMERGENCY	*OFF - off* ON - on	characters	-	
	LOADED	<= value in ACTIVE	unsigned integer	0 - 128	
	ACTIVE	-	unsigned integer	0 - 128	
	ABBIT (Only used with Japan protocol)	A B	-	A - B	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
MTP (Message Transfer Part Managed Object - see Section 5.2.3 on page 5-43)	SPNO	-	unsigned integer	0 - 7	ADD MODIFY DELETE DISPLAY
	PROTOCOL	ITU_93 ITU_97 ANSI_92 ANSI_96	alphanumeric characters	-	
	VARIANT	*GENERIC* NEW_ZEL AT & T GT E BELL ETSI97	alphanumeric characters	-	
	PCSIZE	14_BIT 16_BIT 24_BIT	alphanumeric characters	-	
	MCONG	*OFF - off* ON - on	characters	-	
	MPRIO	*OFF - off* ON - on	characters	-	
	SLTC	*OFF - off* ON - on	characters	-	
	RTRC	*OFF - off* ON - on	characters	-	
	RSRT	*OFF - off* ON - on	characters	-	
	RPO2LPO	*OFF - off* ON - on	characters	-	
	NICHECK	OFF - off *ON - on*	characters	-	
	DPCCHECK	OFF - off *ON - on*	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
RTSET (Route Set Managed Object - see Section 5.2.4 on page 5-47)	RTSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	DPC	-	alphanumeric characters	Format: x-y-z (see page 5-2 for more information about the point code format).	
	RTYPE	*MEMBER* CLUSTER NETWORK CAPABILITY	characters	-	
	STATE	ACC *INACC* REST	characters	-	
ROUTE (Route Managed Object - see Section 5.2.5 on page 5-49	RTSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	LSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	
	PRIORITY	0 is the highest priority and the highest available priority is the default if this parameter is not entered.	unsigned integer	0 - 7 (Up to two load sharing routes can have the same priority)	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SS7BOARD (SS7 Board Managed Object - see Section 5.2.6 on page 5-51	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	BOARDNM	sbs334 pci334 pci3xpq pci3xapq cpc37xpq pmc8260 artic8260 vbrd	alphanumeric characters	-	
	INST	-	unsigned integer	0 - 7	
	CONF	*OFF - off* ON - on	characters	-	
	PM	OFF - off ON - on	characters	-	
	MODULES	*trmod*	alphanumeric characters	-	
	PORTS	-	unsigned integer	1 - 64, where the maximum depends upon the maximum number of ports on the board.	
	CLOCKMODE	*LINE* INTERNAL EXTERNAL REMOTE	characters	-	
	CLOCKSPAN	-	unsigned integer	1 -8, required only if CLOCKMODE is LINE.	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
L2FLOW (MTP Level-2 Flow Managed Object - see Section 5.2.7 on page 5-54)	LINK	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	FCLEVEL	Depends on the ON/ OFF value in the MCONG parameter of the MTP MO: ON1 - 3 OFF1	unsigned integer	1 - 3	
	CONGONVAL	Depends upon the FCLEVEL value. See Table 5-13 on page 5-54 for more information.	unsigned integer	0 - 127	
	CONGABVAL	Depends upon the FCLEVEL value. See Table 5-13 on page 5-54 for more information.	unsigned integer	0 - 127	
	DISCONVAL (requires that MTP's MPRIO value be ON)	Depends upon the FCLEVEL value. See Table 5-13 on page 5-54 for more information.	unsigned integer	0 - 127	
	DISCABVAL (requires that MTP's MPRIO value be ON)	Depends upon the FCLEVEL value. See Table 5-13 on page 5-54 for more information.	unsigned integer	0 - 127	
L2TIMER (MTP Level-2 Timer Managed Object - see Section 5.2.8 on page 5-57)	LINK	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	MODIFY DISPLAY
	TIMER	See Integer in Table 5-14 on page 5-58.	unsigned integer	0 - 8	
	VALUE	See Range in Table 5-14 on page 5-58.	seconds or milliseconds	-	
L3TIMER (MTP Level-3 Timer Managed Object - Section 5.2.9 on page 5-59)	TIMER	See Integer in Table 5-15 on page 5-60.	unsigned integer	1 - 31	MODIFY DISPLAY
	VALUE	See Range in Table 5-15 on page 5-60.	minutes or seconds	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
LINE (Line Managed Object - see Section 5.2.10 on page 5-62)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	BOARDNM	sbs334 pci334 pci3xpq pci3xapq cpc37xpq pmc8260 artic8260 vbrd	alphanumeric characters	-	
	INST	-	unsigned integer	0-7	
	SPAN	Class II/III boards - 1,2 Class IV boards - 1 to 8	unsigned integer	1 - 8	
	LINE_FRMMOD	E1: *E1CRC4* E1FEBE E1BASIC T1: *T1ESF* T1ZBTSI T1SFRM T1SF4	alphanumeric characters	-	
	LINE_COD	E1: *E1HDB3* AMI T1: *T1B8ZS* T1B7ZS AMI	alphanumeric characters	-	
	LINE_LEN	Class III:*L133* L266 L399 L533 L655 Class IV:*L110* L220 L330 L440 L550 L660 LB000 LB075 LB150 LB225	alphanumeric characters	-	
	LINE_IMP	E1: *I120* I75	alphanumeric characters	-	
	LINE_LPBK	*NONE* LOCAL REMOTE	characters	-	
	LINE_NTFY	*OFF - off* *ON - on*	characters	-	
	LINE_TYP	E1 *T1*	characters	-	

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
LINEHIST (Line History Managed Object - see Section 5.2.11 on page 5-66)	LINE_ACCESS	FRONT=front access REAR=rear access	characters	-	MODIFY DISPLAY
	HOSTNAME	-	alphanumeric characters	-	
	BOARDNM	pci3xpq pci3xapq cpc37xpq pmc8260 artic8260	alphanumeric characters	-	
	INST	-	integer	0 - 7	
	SPAN	-	integer	1 - 8	
	RESET	YES NO	characters	-	
LINESTAT (Line Statistics Managed Object - see Section 5.2.12 on page 5-69)	HOSTNAME	-	alphanumeric characters	-	MODIFY DISPLAY
	BOARDNM	pci3xpq pci3xapq cpc37xpq pmc8260 artic8260	alphanumeric characters	-	
	INST	-	integer	0 - 7	
	SPAN	-	integer	1 - 8	
	ERREVENTS	-	integer	-	
LINKSTAT (Link Status Managed Object - see Section 5.2.13 on page 5-72)	LINK	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	MODIFY DISPLAY
	STATUS	SET_ACT CLR_ACT CLR_EMR SET_EMR CLR_ECO SET_ECO CLR_INH SET_INH CLR_LPO SET_LPO TEST_SLTM	characters	-	
LSETSTAT (Link Set Status Managed Object - see Section 5.2.14 on page 5-74)	LSET	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	MODIFY DISPLAY
	STATUS	SET_ACT CLR_ACT	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
PORT (Port Managed Object - see Section 5.2.15 on page 5-76)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	BOARDNM	sbs334 pci334 pci3xpq pci3xapq cpc37xpq pmc8260 artic8260 vbrd	alphanumeric characters	-	
	INST	-	unsigned integer	0-7	
	PORTNUM	-	unsigned integer	0 - 63 (depends upon the system's configuration)	
	TYPE	*DTE* DCE	characters	-	
	BAUD	600 1200 2400 4800 7200 9600 *16000 19200 *32000 38400 *48000 *56000 **64000* * E1/T1 boards support only these baud rates.	unsigned integer	-	
	LPMODE	*NONE* LOCAL REMOTE	characters	-	
	IDLEDETECT	OFF - off *ON - on*	characters	-	
SLTIMER (SLTM Timer Managed Object - see Section 5.2.16 on page 5-79)	TIMER	1 2	unsigned integer	1 - 2	MODIFY DISPLAY
	VALUE	See Table 5-21 on page 5-79 for valid values.	unsigned integer	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SP (Signaling Point Managed Object - see Section 5.2.17 on page 5-81)	SPNO	-	unsigned integer	0 - 7	MODIFY DISPLAY
	NAME	-	alphanumeric characters	String of 1 - 10 alphanumeric characters	
	SPC	-	unsigned integer	Format: x-y-z, based on the PCSIZE parameter of the MTP MO. (see page 5-2 for more information about the point code format).	
	NI	INTERNATIONAL SPARE NATIONAL RESERVED	characters	-	
	TYPE	STP SEP SEPWRT	characters	-	
ALIAS (Alias Point Code Managed Object - see Section 5.2.18 on page 5-83)	APC	-	unsigned integer	Format: x-y-z, based on the PCSIZE parameter of the MTP MO. (see page 5-2 for more information about the point code format).	ADD MODIFY DELETE DISPLAY
	OGPC	OFF - off *ON - on*	characters	-	
	INFLTR	OFF *SPC* APC	characters	-	
	FLTRACT	ALARM *UPU*	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
TIMESLOT (Time Slot Managed Object - see Section 5.2.19 on page 5-85)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	BOARDNM	sbs334 pci334 pci3xpq pci3xapq cpc37xpq pmc8260 artic8260 vbrd	alphanumeric characters	-	
	INST	-	unsigned integer	0-7	
	DESTTYPE	LINE HDLC CTBUS	characters	-	
	DESTSPAN	LINE (II/III 1-2) (IV 1-available spans on board) CTBUS (0 -31)	integer	0 - 31	
	DESTSLOT	LINE: Class II (E1) 0 -31 Class III (T1) 0 - 23 HDLC: 0 - maximum ports on the board CTBUS 0 -31	integer		
	ORIGTYPE	LINE HDLC NOCONNECT	characters	-	
	ORIGSPAN	LINE (II/III 1-2) (IV 1-available spans on board) CTBUS (0 -31)	integer	0 - 31	
	ORIGSLOT	LINE: Class II (E1) 0 -31 Class III (T1) 0 - 23 HDLC: 0 - maximum ports on the board NOCONNECT: 0	integer	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
L2CS (MTP Level-2 Status - see Section 5.2.20 on page 5-89)	LINK	-	alphanumeric characters	String of 1 - 12 alphanumeric characters	MODIFY DISPLAY
CTBUS (CTbus Managed Object- see 5.2.21 on page 5 - 90)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	BOARDNM	sbs334pci334pci3xpqpci3xapqcpc37xpqpmc8260artic8260	alphanumeric characters	-	
	INST	-	unsigned integer	0-7	
	REFCLK	C8AC8BNETREF1NETREF2SCSA2SCSA4SCSA8MVIPHMVIP	alphanumeric characters	-	
	REFINV	OFFON	characters	-	
	FBMODE	C8AC8BNETREF1NETREF2INTERNALLINE	alphanumeric characters	-	
	FBSPAN	-	unsigned integer	1-8	
	FB	OFFON	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-2: MTP Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
CTBUS (CTbus Managed Object- see 5.2.21 on page 5 - 90)	COMP	OFFON	characters	-	MODIFY DISPLAY
	C8A	OFFON	characters	-	
	C8B	OFFON	characters	-	
	NRMODE	NETREF1NETREF2INTERNALLINE	alphanumeric characters	-	
	NRSPAN	-	unsigned integer	1-8	
	NR8KHZ	OFFON	characters	-	
	NRINV	OFFON	characters	-	
	NRACT	OFFON	characters	-	
	NR1	OFFON	characters	-	
	NR2	OFFON	characters	-	
	GRP_A	OFF204840968192	alphanumeric characters	-	
	GRP_B	OFF204840968192	alphanumeric characters	-	
	GRP_C	OFF204840968192	alphanumeric characters	-	
	GRP_D	OFF204840968192	alphanumeric characters	-	
	GRP_E	OFF204840968192	alphanumeric characters	-	
	GRP_F	OFF204840968192	alphanumeric characters	-	
	GRP_G	OFF204840968192	alphanumeric characters	-	
	GRP_H	OFF204840968192	alphanumeric characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-3: SCCP Configuration Managed Objects

Option	Parameter Name	Value	Unit	Range		Command Operation
CPC (Concerned Point Code Managed Object - see Section 5.3.1)	SPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	ADD DELETE DISPLAY
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
	SSN	-	numeric	2 to 255		
	CPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
				for entire list: *		
GT (Global Title Managed Object - see Section 5.3.2)	GT	-	bits	8		ADD DELETE DISPLAY
	GTIE	-	-	1 to 15		
	TRTYPE	-	-	0 to 255		
	NUMPLAN	1	-	-		
	NATOFADDR	(replaces TRTYPE field when GTIE=1 or 4)[4]	-	-		
	ADDRINFO	-	each digit=1 byte	character string		
	LOADSHARE	ON OFF	-	character string		

Note: *Italics denotes the default*

Table 5-3: SCCP Configuration Managed Objects (Continued)

Option	Parameter Name	Value	Unit	Range	Command Operation
GTENTRY (Global Title Entry Managed Object - see Section 5.3.3)	IO	INCOMING OUTGOING	-	-	ADD DELETE DISPLAY
	GT	-	-	1 to 131,072	
	ENTRYTYPE	PRIMARY SECONDARY	-	-	
	XLATE_ID	-	Alpha Numeric characters	1 to 12 characters	
	SPC	-	-	for CCITT networks: Zone/ Network/ SPID (3-8-3 format)	
				for ANSI networks: Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks: (5-4-7 format)	
	SSN	-	-	2 to 255	
	NEWGT	-	numeric	1 to 4	
	WILDCARD	YES *NO*	-	-	

Note: *Italics denotes the default*

Table 5-3: SCCP Configuration Managed Objects (Continued)

Option	Parameter Name	Value	Unit	Range		Command Operation
MATE (Mate Managed Object - see Section 5.3.4)	SPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	ADD DELETE DISPLAY
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
	SSN	-	-	2 to 255		
	MSPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
	MSSN	-	-	2 to 255		

Note: *Italics denotes the default*

-

Table 5-3: SCCP Configuration Managed Objects (Continued)

Option	Parameter Name	Value	Unit	Range		Command Operation
SCCP (SCCP Managed Object - see Section 5.3.5)	SPNO	-	integer	0 to 7		DISPLAY MODIFY
	PROTOCOL	*DEFAULT* ANSI_92 ANSI_96 ITU_93 ITU_97	-	-		
	VARIANT	NONE ATT APLUS SNET	-	-		
	PCIND	YES NO	-	-		
	T_CONN_EST	-	decimal (in milliseconds)	-		
	T_IAS	-		-		
	T_IAR	-		-		
	T_REL	-		-		
	T_GUARD	-		-		
	T_RESET	-		-		
	T_SEGMENT	-		-		
SNSP (SCCP Signaling Point Managed Object - see Section 5.3.6)	SPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	ADD DELETE DISPLAY
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
Note: *Italics denotes the default*						

Table 5-3: SCCP Configuration Managed Objects (Continued)

Option	Parameter Name	Value	Unit	Range		Command Operation
SUBSYS (Subsystem (Managed Object - see Section 5.3.7)	SPC	-	-	for CCITT networks:	Zone/ Network/ SPID (3-8-3 format)	ADD DELETE DISPLAY
				for ANSI networks:	Network/ Cluster/ Member (8-8-8 format)	
				for Japanese networks:	(5-4-7 format)	
	SSN	-	numeric	2 to 255		
LOCALSUBSYS (Local Subsystem Managed Object - see Section 5.3.8)	-	-	-	-		DISPLAY
CONNECTION (Connection Managed Object - see Section 5.3.9)	ID	-	-	0 to 16383 or * for all		DISPLAY
Note: *Italics denotes the default*						

Table 5-4: System Configuration Managed Objects

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
HOST (Host Managed Object - see Section 5.4.1 on page 5-111)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	ADD MODIFY DELETE DISPLAY
	RMTHOST	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	ALIAS	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	RMTHOSTTYP	*AMGR* OTHER	characters	-	
	CONF	ON - on *OFF - off*	characters	-	
STRDALM (Stored Alarm Managed Object - see Section 5.4.2 on page 5-113)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	DELETE DISPLAY
	GROUP	DKM ETMOD ISUP* ISUPMOD* MTPL1 MTPL2 APM NIMOD OMAP SCCP* SPM TCAP* TCMOD* UPM SG PMON PMMOD *Signaling Gateway does not report these alarms	characters	-	
	MODULE	-	unsigned integer	Middle two digits of the alarm(s)	
	TYPE	-	unsigned integer	Last two digits of the alarm(s)	
	LAST_OCCUR	-	alphanumeric characters	Format: hh:mm:ss@ MM/DD/YY	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-4: System Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
ALARM (Alarm Managed Object - see Section 5.4.3 on page 5-116)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	DISPLAY	*ON - on* OFF - off	characters	-	
	CONS_THRS	NONE *INFO* MINOR MAJOR CRITICAL FATAL	characters	-	
	USER_THRS	NONE *INFO* MINOR MAJOR CRITICAL FATAL	characters	-	
	REPEAT	*3*	unsigned integer	0 - 999	
	GLOBAL	ON - on OFF - off	characters	-	
	UPDATE	ON - on OFF - off	characters	-	
ALMEVENT (Alarm Event Managed Object - see Section 5.4.4 on page 5-119)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	DISPLAY
	REQ_HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	GROUP	-	unsigned integer	0 - 255	
	MODULE	-	unsigned integer	0 - 255	
	TYPE	-	unsigned integer	Last two digits of the alarm ID	
	THRESHOLD	NONE *INFO* MINOR MAJOR CRITICAL FATAL	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-4: System Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
ALMGRP (Alarm Group Managed Object - see Section 5.4.5 on page 5-121)	GROUP	DKM ETMOD ISUP* ISUPMOD* MTPL1 MTPL2 APM NIMOD OMAP SCCP* SPM TCAP* TCMOD* UPM SG PMON PMMOD *Signaling Gateway does not report these alarms	characters	-	MODIFY DISPLAY
	CONS_THRS	NONE *INFO* MINOR MAJOR CRITICAL FATAL	characters	-	
	USER_THRS	NONE *INFO* MINOR MAJOR CRITICAL FATAL	characters	-	
MMLCONF (MML Configuration Managed Object- see Section 5.4.6 on page 5-124)	LOG	ON - on OFF - off	characters	-	MODIFY DISPLAY
	TIMEOUT	*15000*	milliseconds	0 - 240000	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-4: System Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
NTWK (Network Managed Object- see Section 5.4.7 on page 5-126)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	MODE	STNDLN *DSTRBTD*	characters	-	
	CLOCKSYNC	*ON - on* OFF - off	characters	-	
	FREQUENCY	*0* - stand alone *1000* - distributed	milliseconds	60 - 10000 (for distributed mode)	
	DUALHOST	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	NETMASK1	*7f000000* Class A *3fff0000* Class B *1fffff00* Class C	-	32-bit mask in hex format used to extract primary network ID	
	NETMASK2	*7f000000* Class A *3fff0000* Class B *1fffff00* Class C	-	32-bit mask in hex format used to extract secondary network ID	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-4: System Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
TCPCON (TCP/IP Connections Managed Object - see Section 5.4.8 on page 5-128)	HOSTNAME	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	MODIFY DISPLAY
	RMTHOST	-	alphanumeric characters	String of 1 - 15 alphanumeric characters	
	MODE	*AUTO* MASTER SLAVE	characters	-	
	SERVICE	*NETDBASE*	-	-	
	PROTO	*TCP*	-	-	
	MODULES	*NIMOD - nimod*	-	-	
	HBEAT	ON - on OFF - off	characters	-	
	FREQU	*1000 if HBEAT is ON* *0 if HBEAT is OFF*	milliseconds	0 - 1000	
	MAXTRIES	*-1* (unlimited tries)	unsigned integers	-1 - 5000	
	ACT_EST	IGNORE - ignore *INFORM - inform*	characters	-	
	ACT_RMV	IGNORE - ignore *INFORM - inform*	characters	-	
	HB_LOSS	NOACTION - noaction SYNCDATA - syncdata	characters	-	
SET-LOG (Log Managed Object - see Section 5.4.11 on page 5-133)	TO	NMDOBJ SS7OBJ	characters	-	-
	NAME	-	characters	String of 1 to 14 characters maximum	
	SPID	-	unsigned integer	0 - 7	
	UPID	0 - MTP (SNM) 3 - SCCP 5 - ISUP	unsigned integer	-	
	SSN	-	unsigned integer	2 - 255	
	INST	-	unsigned integer	-	
	LOG	ON - on *OFF - off*	characters	-	

1 *Note: When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

Table 5-5: Signaling Gateway Configuration Managed Objects

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SGSPNA (Signaling Gateway Managed Object - see Section 5.5.1 on page 5-135)	SPID	-	integer	0 - 7	ADD MODIFY DELETE DISPLAY
	NAID	-	integer	0 - 0x7fffffff	
	OPERST	ACT *INACT*	alphanumeric characters	-	
SGSGP (Signaling Gateway Process Managed Object - see Section 5.5.2 on page 5-138)	SGPID	-	alphanumeric characters	String of up to 15 characters, formatted as hostname-sp[4]	MODIFY DISPLAY
	IP1	-	alphanumeric characters	String of 1 to 15 characters in dot notation format[3]	
	HOSTNAME	-	alphanumeric characters	String of 1 to 128 characters	
	IP2	-	alphanumeric characters	String of 1 to 15 characters in dot notation format[3]	
	SCTPPORT	2905	unsigned integer	1000 - 99999	
SGAS (Application Server Managed Object - see Section 5.5.3 on page 5-140)	ASID	-	alphanumeric characters	String of 1 to 15 characters	ADD MODIFY DELETE DISPLAY
	SPID	-	integer	0 - 7	
	RKID	-	alphanumeric characters	String of 1 to 15 characters	
	RCID	-	integer	0 - 0x7fffffff	
	MODE	LOADSHARE OVERRIDE	characters	-	
	MINASP	-	integer	0 - 5	
	ASPLIST	-	alphanumeric characters	String of up to 40 characters that specifies up to five ASP IDs, separated by a slash, /	
	PDTIMER *5* seconds	0 - 120			

1 ***Note:*** *When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 ***Note:*** *Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

3 ***Note:*** *An Internet address in dot notation has one to four numbers, and this must be in decimal format. It represents a 32 bit address, where each leading number is eight bits of the address (high byte first) and the last number is the rest. The following is an example of the decimal format: 146.169.22.42.*

4 ***Note:*** *The* ***hostname*** *is the official hostname. The* ***sp*** *is the signaling point on which the SGP runs.*

Table 5-5: Signaling Gateway Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SGASP (Application Server Process Managed Object - see Section 5.5.4 on page 5-143)	ASPID	-	alphanumeric characters	String of 1 to 15 characters	ADD MODIFY DELETE DISPLAY
	IP1	-	alphanumeric characters	String of 1 to 15 characters in dot notation format[3]	
	IP2	-	alphanumeric characters	String of 1 to 15 characters in dot notation format[3]	
	HOSTNAME	-	alphanumeric characters	String of 1 to 128 characters	
	SCTPPORT	-	integer	1000 - 99999	
	NWASPID	*0x7fffffff*	integer	0 - 0x7fffffff	
	SGPID	-	alphanumeric characters	String of 1 to 15 characters	
	OPERST	*DISCONN - disconn* LOCK - lock UNLOCK - unlock	characters	-	
SGRK (Route Key Managed Object - see Section 5.5.5 on page 5-146)	RKID	-	alphanumeric characters	String of 1 to 15 characters	ADD MODIFY DELETE DISPLAY
	TYPE	DPC DPC_OPC DPC_OPC_CIC DPC_SIO DPC_CIC_SIO DPC_SSN DPC_OPC_SIO	characters	-	
	DPC	-	alphanumeric characters	Format: x-y-z, based on the PCSIZE parameter of the MTP MO. (see page 5-2 for more information about the point code format).	

1 ***Note***: *When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 ***Note***: *Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

3 ***Note***: *An Internet address in dot notation has one to four numbers, and this must be in decimal format. It represents a 32 bit address, where each leading number is eight bits of the address (high byte first) and the last number is the rest. The following is an example of the decimal format: 146.169.22.42.*

4 ***Note***: *The **hostname** is the official hostname. The **sp** is the signaling point on which the SGP runs.*

Table 5-5: Signaling Gateway Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SGRKRNG (Route Key Range Managed Object - see Section 5.5.6 on page 5-148)	RKID	-	alphanumeric characters	String of 1 to 15 characters	ADD DELETE DISPLAY
	SI	SCCP ISUP TUP	characters	-	
	OPC	-	alphanumeric characters	Format: x-y-z, based on the PCSIZE parameter of the MTP MO. (see page 5-2 for more information about the point code format).	
	CICMIN	-	alphanumeric characters	0-65535	
	CICMAX	-	alphanumeric characters	0-65535	
	SSN	-	integer	2-255	
SGASTFC (AS Traffic Status Managed Object - see Section 5.5.7 on page 5-150)	SGPID	-	alphanumeric characters	String of up to 15 characters	MODIFY DISPLAY
	ASPID	-	alphanumeric characters	String of up to 15 characters	
	OPERST	INACT - inactive ACT - activate	characters	-	
	ASID	-	alphanumeric characters	String of up to 15 characters	
SGSPMC SPMC Status Managed Object - see Section 5.5.8 on page 5-152	SGPID	*<hostname of running SGP>*	alphanumeric characters	String of up to 15 characters	DISPLAY
	DPC	-	alphanumeric characters	Format: x-y-z, based on the PCSIZE parameter of the MTP MO. (see page 5-2 for more information about the point code format).	
SGPSG Peer SG Managed Object - see Section 5.5.10 on page 5-155	SGID	-	alphanumeric characters	String of 1 to 15 characters	ADD, MODIFY, DELETE, DISPLAY
	MODE	OVERRIDE, LOADSHARE	characters	-	

1 **Note**: *When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*
2 **Note**: *Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*
3 **Note**: *An Internet address in dot notation has one to four numbers, and this must be in decimal format. It represents a 32 bit address, where each leading number is eight bits of the address (high byte first) and the last number is the rest. The following is an example of the decimal format: 146.169.22.42.*
4 **Note**: *The* **hostname** *is the official hostname. The* **sp** *is the signaling point on which the SGP runs.*

Table 5-5: Signaling Gateway Configuration Managed Objects (Continued)

Name	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
SGPSGP Peer SGP Managed Object - see Section 5.5.11 on page 5-157	PSGPID	-	alphanumeric characters	String of 1 to 15 characters	ADD, MODIFY, DELETE, DISPLAY
	SGID	-	alphanumeric characters	String of 1 to 15 characters	
	IP1	-	alphanumeric characters	String of 1 to 15 characters	
	IP2	-	alphanumeric characters	String of 1 to 15 characters	
	HOSTNAME	-	alphanumeric characters	string of 1 to 128 characters	
	SCTPPORT	-	integer	1000-99999	
	LSGPID	-	alphanumeric characters	String of 1 to 15 characters	
	OPERST	DISCONN, CONN, DOWN, UP	characters	String of 1 to 15 characters	

1 ***Note:*** *When applicable, default values are shown in italics. Values that ARE NOT case sensitive appear in both upper and lower case.*

2 ***Note:*** *Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

3 ***Note:*** *An Internet address in dot notation has one to four numbers, and this must be in decimal format. It represents a 32 bit address, where each leading number is eight bits of the address (high byte first) and the last number is the rest. The following is an example of the decimal format: 146.169.22.42.*

4 ***Note:*** *The **hostname** is the official hostname. The **sp** is the signaling point on which the SGP runs.*

Managed Object	Parameter Name	Value[1,2]	Unit	Range[2]	Command Operation
PMLINK(Passive Monitor Link - see 5.6.1 on page 5 - 161)	HOSTNAME	-	alphanumeric characters	String of 1-12 alphanumeric characters	ADD DELETE DISPLAY MODIFY
	BOARDNM	pci3xpq pci3xapq pmc8260 artic8260	alphanumeric characters	String of 1-12 alphanumeric characters	
	INST	-	unsigned integer	0 - 7	
	PORT	-	unsigned integer	Depends on the BOARDNUM: pci3xpq 0 - 32 pci3xapq 0 - 32 pmc8260 0 - 63 artic8260 0 - 63	
	ADMINSTAT	ACTIVATE DEACTIVATE	alphanumeric characters	-	
	OPERSTAT	SHUTOFF INACTIVE IDLE/OOS ALIGNING INSERVICE PROC-OUT	alphanumeric characters	-	
	LINKF	-	unsigned integer	-	
	RXFRAMES	-	unsigned integer	-	
	RXOCTETS	-	unsigned integer	-	
	RSU_E	-	unsigned integer	-	
	D_RXL	-	unsigned integer	-	
	D_BO	-	unsigned integer	-	

1 *Note: When applicable, default values are shown in italics. Values NOT case sensitive appear in both upper and lower case.*
2 *Note: Decimal numbers are typed without a prefix, octal numbers are preceded by O', hexadecimal numbers are preceded by H'.*

5.2 MTP MML Commands

5.2.1 Link (LINK)

NAME

LINK Adds, modifies, deletes, or displays information about a link.

COMMANDS

ADD Adds a signaling link to an existing link set. The CONF parameter of the SS7BOARD managed object must be set to ON prior to this command. The following are created when a new LINK is added:

- an instance of LINKSTAT, which includes the status of the link.
- instances of the L2TIMER managed object (eight timers).
- instances of the L2FLOW managed object (4 flow control level).

ADD-LINK:LINK=link***,LSET***=lset***,SLC***=slc***,PRIORITY***=priority***,***
[L2ECM=l2ecm***,][PCRN1***=pcrN1***,][PCRN2***=pcrN2***,]***
HOSTNAME=hostname***,BOARDNM***=boardnm***,INST***=inst***,***
PORT=port***;***

__Note__: The PROTOCOL parameter of the MTP managed object determines the instance values of L2TIMER and the number of L2FLOW instances.

MODIFY Modifies the priority of an existing link.

MODIFY-LINK:LINK=link ***[,PRIORITY***=priority***]***
[,L2ECM=l2ecm***][,PCRN1***=pcrN1***][,PCRN2***=pcrN2***];***

DELETE Deletes a link from its link set.

DELETE-LINK:LINK=link***;***

__Note__: The link must be deactivated with MODIFY-LINKSTAT __BEFORE__ it can be deleted. Also, the corresponding LINKSTAT instance is deleted with this command.

DISPLAY Displays configuration information on one link, all links in a link set, or all existing links.

DISPLAY-LINK:[LINK=link***][,LSET***=lset***];***

PARAMETERS

link Link identification. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display every link.

lset Link set identification. It is a string of 1 to 12 alphanumeric characters maximum.

slc Signaling link code. It is an integer from 0 to 15.

priority The priority of the signaling link. It is an integer from 0 to 15, where 0 is the highest priority. The highest available priority is assigned if no value is entered.

l2ecm MTP Level 2 Error Correction Method for the link. This parameter is optional and can be entered as one of the following values:

- **BASIC** basic method (default)
- **PCR** preventive cyclic retransmission method

Note: *The preventive cyclic retransmission method applies for intercontinental signaling links where one-way propagation delay is greater than or equal to 15ms and for all signaling links established via satellite. (ITU-T Q.703, Section 1.4: Error Correction)*

pcrN1 Maximum number of MSUs available for retransmission. ***pcrN1*** is applicable to preventive cyclic retransmission method. This parameter is optional and ignored if basic method is selected. It is an unsigned integer from 0 to 127. (Default value is 127).

pcrN2 Maximum number of MSU octets available for retransmission. ***pcrN2*** is applicable to preventive cyclic retransmission method. This parameter is optional and ignored if basic method is selected. It is an unsigned integer from 0 to 9999. (Default value is 2000).

hostname The name of the host to which the link is physically connected. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **sbs334** common name for 4-port Sbus boards (sbs334/sbs37x)
- **pci334** common name for 4-port PCI bus boards (pci334/pci37x)
- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)
- **cpc3xpa** common name for 24-port CompactPCI bus boards (cpc370pq/cpc372pq)
- **pmc8260** common name for 64-port CompactPCI bus boards (pmc8260)
- **artic8260** common name for 64-port CompactPCI bus boards (artic1000 and artic2000)
- **vbrd** 32-port virtual board driver

Note: *Although PCI3xPQ, PCI3xAPQ and CPC37xPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full bandwidth on all configured links.*

inst Physical instance number of the board. It is an integer from 0 to 7.

port Port number of the link, entered as a numerical value. Valid range depends on the ***board*** type:

- sbs334 0 to 3
- pci334 0 to 3

- pci3xpq 0 to 23
- pci3xapq 0 to 23
- cpc3xpa 0 to 23
- pmc8260 0 to 63
- vbrd 0 to 31
- artic8260 **0** to **63**

ERRORS

<ERROR>::No room for new entry

<ERROR>::LINK MO instance already exists

<ERROR>::Missing LINK parameter

<ERROR>::Missing SLC parameter

<ERROR>::Missing PRIORITY parameter

<ERROR>::Missing HOSTNAME parameter

<ERROR>::HOSTNAME is not defined in the network

<ERROR>::Missing LSET parameter

<ERROR>::LSET MO instance does not exist

<ERROR>::Missing BOARDNM parameter

<ERROR>::Missing INST parameter

<ERROR>::Missing PORT parameter

<ERROR>::Pre-used HOSTNAME+BOARDNM+INST+PORT combination

<ERROR>::Pre-used SLC value

<ERROR>::Pre-used PRIORITY value

<ERROR>::cnfg library error

<ERROR>::Device not configured

<ERROR>::Link information can not be retrieved

<ERROR>::Invalid stream no is retrieved from spmd

<ERROR>::mtpl2 MO operation failed

<ERROR>::LINK is activated.

<ERROR>::Nothing to list.

EXAMPLES

ADD-LINK:LINK=Link_11***,LSET=***Lset_1***,SLC=***0***,PRIORITY=***0***,***
HOSTNAME=chicago***,BOARDNM=***sbs334***,INST=***0***,PORT***=0***;***

ADD-LINK:LINK=Link_11***,LSET***=Lset_1***,SLC***=0***,PRIORITY***=0***,L2ECM=***PCR***,***
HOSTNAME=chicago***,BOARDNM=***sbs334***,INST=***0***,PORT***=0***;***

MODIFY-LINK:LINK=ls102030-01***,PRIORITY***=3***;***

MODIFY-LINK:LINK=ls102030-01***,PRIORITY***=3,***PCRN1=***120***, PCRN2=***2001***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LINK:;

-------------------------------------------------------------------------------------
 LINK   LSET SLC PRIORITY L2ECM PCRN1 PCRN2    HOSTNAME   HOSTSTATUS   BOARDNM INST PORT
-------------------------------------------------------------------------------------
l1   ls1   0        0  BASIC    -     -     chicago    AVAILABLE   sbs334    0    0
l2   ls1   1        1  BASIC    -     -     chicago    AVAILABLE   sbs334    0    1
l3   ls1   2        2  BASIC    -     -     chicago    AVAILABLE   sbs334    0    2
l4   ls1   3        3    PCR  127  2000     chicago    AVAILABLE   sbs334    0    3
<SUCCESS>:: 4 records found
```

Table 5-7: LINK Display Values

HOSTSTATUS
This displays the current state of the host and can be only of the following values: • UNAVAILABLE • AVAILABLE • CONFLICT

5.2.2 Link Set (LSET)

NAME

LSET Adds, modifies, deletes, or displays information pertaining to a link set.

COMMANDS

ADD Adds a link set. Upon adding a new LSET instance, an LSETSTAT managed object instance, which includes the status information of the link set, will also be created.

ADD-LSET:LSET=lset***,DPC***=dpc***,TYPE***=type***,LOADED***=loaded***,***
ACTIVE=active***[,ABBIT***=abbit***][,EMERGENCY***=emergency***];***

MODIFY Modifies an existing link set.

MODIFY-LSET:LSET=lset***[,TYPE***=type***][,LOADED***=loaded***]***
[,ACTIVE=active***][,EMERGENCY***=emergency***];***

__Important__: The __LOADED__ and __ACTIVE__ parameters are optional but at least one of them must be entered in the MODIFY-LSET command. The LSET __must__ be deactivated __BEFORE__ modifying LOADED or ACTIVE parameters.

DELETE Deletes a link set from the network.

DELETE-LSET:LSET=lset***;***

__Note__: The link set __must__ be deactivated with MODIFY-LSETSTAT, and the route(s) and route set for the link set __must__ be deleted __BEFORE__ it can be deleted.

All LINK and LINKSTAT instances associated with this link set (if any) are deleted with this command. The LSETSTAT instance is also deleted with this command.

DISPLAY Displays the status of one link, all links in a link set, or all existing links.

DISPLAY-LINKSTAT:[LSET=lset***][,LINK***=link***];***

PARAMETERS

lset Link set identification. It is string of 1 to 12 alphanumeric characters maximum, or an * to display every link set.

dpc The adjacent destination point code to which this link set connects. It is entered in the x-y-z format. See page 5-2 for more information about this point code format.

type Link set type. It is a character string that accepts the following values:

- ALINK Access link
- BLINK Bridge link
- CLINK Cross link
- DLINK Diagonal link
- ELINK Link between a Signaling End Point (SEP)and a member of a remote Signaling Transfer Point (STP) pair
- FLINK Link between two Signaling End Points (SEPs)

loaded	Number of links in the link set that will carry traffic, and it ***MUST*** be less than or equal to the number in ***active***. It is usually equal. It is an integer from 0 to 128.
active	Number of links in the link set that are to be aligned and ready for service at all times. It is an integer from 0 to 128.
abbit	AB plane value, which is used only with the Japan protocol. It is a character string that accepts the following values: • A — Plane A link set • B — Plane B link set
emergency	Emergency alignment of the first link in the link set. It is a character string that accepts the following values: • ON — Enables emergency link alignment • OFF — Disables emergency link alignment (*default*)

ERRORS

<ERROR>::Missing LSET parameter

<ERROR>::No room for new entry

<ERROR>::LSET MO instance already exists

<ERROR>::Missing DPC parameter

<ERROR>::Own point code is the same

<ERROR>::An LSET instance exists with the same Point Code

<ERROR>::Parameter LOADED is greater than parameter ACTIVE

<ERROR>::Missing TYPE parameter

<ERROR>::Missing LOADED parameter

<ERROR>::Missing ACTIVE parameter

<ERROR>::Missing ABBIT parameter

<ERROR>::LSET MO instance does not exist.

<ERROR>::LSET is being used by a LINK instance.

<ERROR>::LSET is being used by a ROUTE instance.

<ERROR>::LSET is activated.

<ERROR>::Nothing to list.

<ERROR>::ABBIT parameter is valid for japan protocol only.

EXAMPLES

ADD-LSET:LSET=Lset_1***,DPC***=2-3-4***,TYPE***=ALINK***,LOADED***=4***,ACTIVE***=4***,***
ABBIT=A***;***
MODIFY-LSET:LSET=ls102675***,LOADED***=4***;***
DELETE-LSET:LSET=ls000102***;***
DISPLAY-LSET:LSET=****;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LSET:LSET=*;

---------------------------------------------------------------------

LSETDPCTYPELOADEDACTIVE ABBITEMERGENCY

--------------------------------------------------------------------

LS12-2-2ALINK4 4AOFF

<SUCCESS>:: 1 records found.
```

***Important**: The MTP protocol automatically activates another link when the total number of aligned links falls below the value of the ACTIVE parameter. This activation may occur because of a signaling link test (SLT) signal failure during an activation attempt, or when a link is deactivated by a management request, i.e., LINKSTAT STATUS set to CLR_ACT. The links are automatically activated in order of their priority values, with the one with the highest priority value activated first. The number of links configured in the link set must be greater than, or equal to, the number in the ACTIVE parameter of the LINKSET managed object.*

***Important**: NA, for NOT APPLICABLE, is displayed in the ABBIT parameter if the protocol is not Japan.*

5.2.3 Message Transfer Part (MTP)

NAME

MTP Adds, modifies, displays or deletes the configuration of the MTP layer.

COMMANDS

ADD Adds an MTP. The following are also created automatically upon adding a new MTP:

- L3TIMER with a value determined by the PROTOCOL parameter
- SLTIMER with a value determined by the PROTOCOL parameter
- SP with these parameters:
  - NAME: AMGR
  - SPC: 0-0-1
  - NI: INTERNATIONAL
  - TYPE: SEP

ADD-MTP:PROTOCOL=protocol***,PCSIZE***=pcsize***[SPNO***=spno***]***
[,VARIANT=variant***][,MCONG***=mcong***][,MPRIO***=mprio***]***
[,SLTC=sltc***][,RTRC***=rtrc***][,RPO2LPO***=rpo2lpo***]***
[,NICHECK=nicheck***][,DPCCHECK***=dpccheck***];***

MODIFY Modifies MTP layer 3 parameters

MODIFY-MTP:[SPNO=spno***][,VARIANT***=variant***][,MCONG***=mcong***]***
[,MPRIO=mprio***][,SLTC***=sltc***][,RTRC***=rtrc***]***
[,RPO2LPO=rpo2lpo***][,RSRT***=rsrt***][,NICHECK***=nicheck***]***
[,DPCCHECK=dpccheck***];***

DELETE Deletes an MTP instance.

DELETE-MTP:[SPNO=spno***];***

Note*: Be sure that all related managed objects, such as LSET, LINK, RTSET, etc., are deleted when deleting an MTP.*

DISPLAY Displays variant and congestion information for the Message Transfer Part (MTP) layer of the signaling point.

DISPLAY-MTP:[SPNO=spno***];***

PARAMETERS

Note*: Unless otherwise specified, the default value is used if an optional parameter is not entered.*

spno Signaling Point number entered as a numerical value from 0 to 7 or an * to display every number. If this optional parameter is not entered, the number of the SP that the MML is started with is used.

protocol	Protocol of the MTP layer entered with one of the following values: • ITU_93 ITU 1993 specifications • ANSI_92 ANSI 1992 specifications • ANSI_96 ANSI 1996 specifications
variant	Variant of the MTP layer. It is a character string that accepts the following values: • GENERIC No variant set *(default)* • BELL BELL variant (valid for ANSI) • NEW_ZELNew Zealand variant (reserved for now) • AT&T AT&T variant (reserved for now) • GTE GTE variant (reserved for now)
pcsize	Point code size of the MTP layer. It is a character string that accepts the following values: • 14_BIT 14 bit point code • 16_BIT 16 bit point code • 24_BIT 24 bit point code
mcong	Multiple Congestion flag. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)* • ON this function *is* operational
mprio	Multiple Priority flag. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)* • ON this function *is* operational
sltc	Signaling Link Test Control flag. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)* • ON this function *is* operational
rtrc	Transfer Restricted Control flag. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)* • ON this function *is* operational
rsrt	MTP Restart flag, which initiates a manual MTP restart operation. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)* • ON this function *is* operational
rpo2lpo	Remote and Local Processor Outage flag. It is a character string that accepts the following values: • OFF this function *is not* operational *(default)*

• ON this function *is* operational

nicheck Optional Network Indicator check flag for incoming messages. It is a character string that accepts the following values:

•ON *default*)

•OF

dpccheck Optional Destination Point Code check flag for incoming user part messages. This check applies to SEPs only, and not to STPs. It is a character string that accepts the following values:

•ON *default*)

•OF

***Important**: If **PROTOCOL** is set to ANSI_92 or ANSI_96 then the **PCSIZE** value must be 24_BIT, the **MCONG** value must be ON, **MPRIO** value must ON, **RTRC** value must be ON and **RPO2LPO** value must be OFF.*

ERRORS

<ERROR>::MTP MO instance already exists

<ERROR>::Missing PROTOCOL parameter

<ERROR>::Missing PCSIZE parameter

<ERROR>::RESTART parameter is not allowed in add operation

<ERROR>::Parameters are incompatible with ansi protocols

<ERROR>::MTP MO instance does not exits

<ERROR>::PROTOCOL parameter can not be modified

<ERROR>::PCSIZE parameter can not be modified

<ERROR>::At least one routeset is in use

<ERROR>::Nothing to list

<ERROR>::Invalid SPNO for this upmd MOS

<ERROR>::Mgmt, MTP_STATE RESTARTING

<ERROR>::Mgmt, MTP_STATE RESTARTED

<ERROR>::Invalid VARIANT for this protocol.

EXAMPLES

ADD-MTP:PROTOCOL=ITU***,PCSIZE***=14_BIT***,MCONG***=OFF***,MPRIO***=OFF***,***
SLTC=OFF***;***
MODIFY-MTP:SLTC=ON***,MPRIO***=OFF***;***
DELETE-MTP:;
DISPLAY-MTP:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-MTP:;
--------------------------------------------------------------------------------
 SPNO PROTOCOL VARIANT PCSIZE MCONG MPRIO SLTC   MTP_STATE RTRC RPO2LPO NICHECK DPCCHECK
```

```
-------------------------------------------------------------------------------------
    0  ANSI_92 GENERIC 24_BIT    ON     ON  OFF   RESTARTED   ON      OFF      ON       ON
<SUCCESS>:: 1 record found
```

Table 5-8: MTP Display Values

MTP_STATE
This parameter is the current state of the MTP layer and may be one of the following values: • CREATED • ISOLATED • RESTARTING • RESTARTED

SAMPLE OUTPUT

MML_TH>**DISPLAY-MTP:;**

```
-----------------------------------------------------------------------------------------
SPNOPROTOCOL VARIANT PCSIZE MCONG MPRIO SLTC MTP_STATE RTRC RPO2LPONICHECK DPCCHECK
-----------------------------------------------------------------------------------------
0 ANSI_92 GENERIC 24_BIT ON ON OFF CREATED ON OFF ON ON
<SUCCESS>:: 1 record found.
```

Table 5-9: MTP Display Values

MTP_STATE
CREATED ISOLATED RESTARTING RESTARTED

5.2.4 Route Set (RTSET)

NAME

RTSET Adds, deletes, displays, or modifies a route set, or information about a route set.

COMMANDS

ADD Adds a route set.

ADD-RTSET:RTSET=rtset***,DPC***=dpc***[,RTYPE***=rtype***][,CAPABILITY***=*capability*]***;***

MODIFY Modifies a route's STATE if CAPABILITY is set to ON. This command is functional only for GATEWAY processes.

MODIFY-RTSET:RTSET=rtset***,STATE***=state***;***

DELETE Deletes a route set from the network.

DELETE-RTSET:RTSET=rtset***;***

Important: *Be sure that none of the existing SS7 components, such as ISUPNODE, SNSP, etc., use the destination point code (dpc) defined by this route set* ***BEFORE*** *deleting a route set.*

DISPLAY Displays information about one or more route sets.

DISPLAY-RTSET:[RTSET=rtset***];***

PARAMETERS

rtset Route set name. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display all route sets.

dpc Destination point code for this route set. It is entered in a X-X-X format where the sum of the Xs must equal the bits in the PCSIZE parameter of the MTP MO. See page 5-2 for more information about this point code format.

Sample DPC	PCSIZE Parameter	Sum
3-8-3	14_BIT	3+8+3=14
5-4-7	16_BIT	5+4+7=16
8-8-8	24_BIT	8+8+8=24

Important: *The 0-0-2 point code is reserved for internal use. Users* ***MUST NOT*** *use this point code to create a route set.*

rtype Routing type of the route set. This parameter must be set to **MEMBER** in ITU networks. Valid values are:

• MEMBER Routing based on the full point code *(default).*

- CLUSTERRouting based on the network and cluster portions of the point code.
- NETWORKRouting based on the network portion of the point code.
- ON Capability routing attribute is set for gateway application.
- OFF Capability routing attribute is not set.

state State of a CAPABILITY route set. It is a character string that accepts the following values:

- ACC activates the capability route set
- INACC deactivates the capability route set *(default)*
- RESTR changes a CAPABILITY route set to a restricted state.

ERRORS

<ERROR>::Missing RTSET parameter

<ERROR>::No room for new entry

<ERROR>::RTSET MO instance already exists

<ERROR>::Missing DPC parameter

<ERROR>::Own point code is the same

<ERROR>::A routeset instance exists with the same Point Code

<ERROR>::Only member routing is valid for ITU protocols

<ERROR>::RTSET MO instance does not exist.

<ERROR>::Instance is marked by another MOS.

<ERROR>::Nothing to list.

EXAMPLES

ADD-RTSET:RTSET=denver***,DPC***=2-32-0***,RTYPE***=CLUSTER***;***
ADD-RTSET:RTSET=takoma***,DPC***=2-32-6***;***
DELETE-RTSET:RTSET=philpa13***;***
DISPLAY-RTSET:RTSET=****;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-RTSET:RTSET=*;
-----------------------------------------------
RTSETDPCTYPECAPABILITYSTATECONG
-----------------------------------------------
rs12-3-4CLUSTEROFFINACCOFF
rs2240-23-2MEMBERONACCOFF
<SUCCESS>:: 2 records found.
```


Table 5-10: RTSET Display Values

CONG	STATE
Congestion Status: • ON • OFF	State of Routeset: • ACC Accessible • INACC Inaccessible • REST Restricted

5.2.5 Route (ROUTE)

NAME

ROUTE Adds, deletes, or displays a route, or information about a route.

COMMANDS

ADD Adds a route to a route set.

ADD-ROUTE:RTSET=rtset***,LSET***=lset***,[PRIORITY***=priority***];***

DELETE Deletes a route from an existing route set.

DELETE-ROUTE:RTSET=rtset***,LSET***=lset***;***

DISPLAY Displays information about one or more routes in a route set.

DISPLAY-ROUTE:[RTSET=rtset***][,LSET***=lset***];***

PARAMETERS

rtset Route set name. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display all route sets.

lset Link set identification. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display all route sets.

priority Priority of the link set. It is an integer from 0 to 7, where 0 is the highest priority. The first available priority value is assigned if no priority is entered. The maximum number of routes that can have the same priority in a route set is two. These are load-sharing routes.

ERRORS

<ERROR>::Missing RTSET parameter

<ERROR>::Missing LSET parameter

<ERROR>::ROUTE MO instance already exists

<ERROR>::RTSET MO instance does not exist

<ERROR>::No room for new ROUTE entry in this RTSET

<ERROR>::No more room for equal priority routes

<ERROR>::LSET MO instance does not exist

<ERROR>::ROUTE MO instance does not exist.

<ERROR>::Nothing to list.

EXAMPLES

ADD-ROUTE:RTSET=chicago***,LSET***=chicago_1***;***
ADD-ROUTE:RTSET=chicago***,LSET***=chicago_2***,PRIORITY***=2***;***
DELETE-ROUTE:RTSET=trumct00cl3***,LSET***=Is111111***;***
DISPLAY-ROUTE:RTSET=rs1***,LSET***=****;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-ROUTE:RTSET=rs1;
-------------------------------------------------------------------------
RTSET   LSET  PRIORITY  STATE   LSSTATE  CURRENT  RTCONG  LSCONG
-------------------------------------------------------------------------
  rs1    ls1         0     NI        UA       OFF      OFF     OFF
<SUCCESS>:: 1 records found.
```

Table 5-11: ROUTE Display Values

STATE	LSSTATE	CURRENT	RTCONG	LSCONG
State of route: • PR prohibited • NI Not initiated • RS restricted	State of route (link set). • A available • UA unavailable	Availability of link set as a route to destination: • OFF • ON	Congestion of route: • OFF • ON	Congestion of link set: • OFF • ON

5.2.6 SS7Board (SS7BOARD)

NAME

SS7BOARD Adds, modifies, deletes, or displays an SS7 board and information about its configuration in the system.

COMMANDS

ADD Adds and configures a board in the system.

ADD-SS7BOARD:HOSTNAME=hostname***,BOARDNM***=boardnm***,***
INST=inst***[,PORTS***=ports***][,MODULES***=modules***]***
[,CLOCKMODE=clockmode***][,CLOCKSPAN***=clockspan***]***
[,CONF=conf***][,PM***=pm***];***

MODIFY Changes the configuration settings of a board that is defined.

MODIFY-SS7BOARD:HOSTNAME=hostname***,***
BOARDNM=boardnm***,INST***=inst***[,MODULES***=modules***]***
[,CLOCKMODE=clockmode***][,CLOCKSPAN***=clockspan***]***
[,CONF=conf***];***

DELETE Deletes an SS7 board from the system.

DELETE-SS7BOARD:HOSTNAME=hostname***,***
BOARDNM=boardnm***,INST***=inst***;***

DISPLAY Displays the configuration information of the SS7 boards in the system.

DISPLAY-SS7BOARD:[HOSTNAME=hostname***]***
[,BOARDNM=boardnm***][,INST***=inst***];***

PARAMETERS

hostname Name of the host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **sbs334** common name for 4-port Sbus boards (sbs334/sbs37x)
- **pci334** common name for 4-port PCI boards (pci334/pci37x)
- **pci3xpq** common name for 24-port PCI boards (pci370pq/372pq)
- **pci3xapq** common name for 24-port PCI boards (pci370apq/372apq)
- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)
- **artic8260** common name for 64-port CompactPCI bus boards (artic1000 and artic2000)
- **vbrd** 32-port virtual board driver (vbrd)

__Note:__ Although PCI3xPQ, PCI3xAPQ and CPC3xPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full bandwidth on all configured links.

inst Identifies the SS7 board driver instance number. The *getcfg* command provides the configured SS7 board slot, driver, and instance information.

pm passive monitoring option. it is used to select the software layer that will be downloaded on to the board. Set type, possible values are:

- **ON**: download Passive Monitor software
- **OFF**: download MTPL2 protocol software *(default)*

conf Determines the configuration of the host. It is a character string that accepts the following values:

- ON: establish this connection
- OFF: wait for some other configurations *(default)*

modules Specifies an ordered list of STREAMS modules to be pushed over this device connection. ***trmod*** is the default module name.

ports Maximum number of available ports on this board. The default value is determined by the physical board hardware. The value ranges from 1 to 32, for different types of boards, where the maximum is the maximum number of ports on the board.

clockmode Clock source of board—class II , class III or class IV type. It is a character string that accepts the following values:

- LINE for class II, III and IV boards, LINE-1 maps to the older LINEB *(default),* and LINE-2 maps to the older LINEA.
- INTERNALfor class II, III and IV boards
- EXTERNALfor class II, and III boards
- REMOTE for class IV board, this designates that the board is a CT bus clock slave. In all other clockmodes, except EXTERNAL, the board is a CT bus clock master with its internal clock synchronized to either LINE 1 through 8, or self-synchronized, which is the case for INTERNAL.

clockspan Clock source span number if the clockmode parameter is LINE. This parameter is ignored if the clockmode is not LINE. It accepts the following values:

CLOCKSPAN VALUE	BOARD
1 - 2	Class II and III
1 - number of physical spans	Class IV

ERRORS

<ERROR>::Missing BOARDNM attribute

<ERROR>::Missing INST attribute

<ERROR>::Board already configured

<ERROR>::Board is in use

<ERROR>::No such a SS7BOARD MO instance

<ERROR>::Can not add board MO

<ERROR>::Can not get board attributes

<ERROR>::CLOCKMODE value must be one of {LINE, INTERNAL, EXTERNAL}

<ERROR>::PORTS value is greater than number of available ports on the board

<ERROR>::Board is OFFLINE

EXAMPLES

ADD-SS7BOARD:HOSTNAME=host-A***,BOARDNM***=sbs334***,INST***=0***, MODULES***=trmod***,PORTS***=4***,CLOCKMODE***=LINE***,CLOCKSPAN***=1***, CONF***=ON***;***
MODIFY-SS7BOARD:HOSTNAME=host-A***,BOARDNM***=sbs334***,INST***=0***, MODULES***=trmod***,CLOCKMODE***=LINE***,CLOCKSPAN***=1***,CONF***=ON***;***
DELETE-SS7BOARD:HOSTNAME=ultra5***,BOARDNM***=pci334***,INST***=0***;***
DISPLAY-SS7BOARD:;
DISPLAY-SS7BOARD:HOSTNAME=host-A***,BOARDNM***=sbs334***;***

SAMPLE OUTPUT

```
MML_TH>dis-ss7board:;
---------------------------------------------------------------------------------------------------
HOSTNAME BOARDNM INST CONF PM MODULES STATE CLASS PORTS LINES CLOCKMODE CLOCKSPAN SPMLINKNO
---------------------------------------------------------------------------------------------------
ziwo     pci334    0  ON   OFF trmod  READY  I     4    0     NOTUSED      -       0
ziwo     pci3xpq   0  ON   OFF trmod  READY  III   24   2     LINE         1       1
ziwo     pci3xapq  0  ON   ON  pmmod  READY  III   24   2     INTERNAL     -       2
ziwo     pmc8260   0  ON   OFF trmod  READY  IV    64   4     LINE         2       3
ziwo     artic8260 0  ON   OFF trmod  READY  IV    64   4     LINE         1       4
<SUCCESS>:: 5 records found
```

Table 5-12: SS7BOARD Display Values

CLASS	LINES	STATE	SPMLINKNO
This read-only parameters identifies the board's preassigned hardware class type. The board's hardware determines the value. **I** class 1 64kbits/s board **II** class 2 E1 board **III**class 3 T1board **IV**class 4 E1/T1 configurable board	This read-only parameter identifies the number of lines on this board. • For E1/T1 boards, this parameter is the number of E1/T1 spans on the board hardware. • For 64K boards, this parameter is 0.	This read-only parameter is the state of this board, which is one of the following: • DETACHED •ATTACHED • CDWNLOADED (mtp/l2 downloaded) • READY •RESET	This read-only parameter corresponds to the stream number underneath the SPM multiplexer and is of interest because it shows exactly where the corresponding device driver has been linked. It is an integer from 0 to 7.

5.2.7 Level-2 Flow (L2FLOW)

NAME

L2FLOW — Modifies or displays MTP Level-2 flow control information.

COMMANDS

MODIFY — Modifies level-2 flow control values for a specific link. All links can have different flow control ***values***, however, all links on a signaling point must have the same flow control ***levels***.

MODIFY-L2FLOW:LINK=link***,FCLEVEL***=fclevel ***[,CONGONVAL***=congonval***][,CONGABVAL***=congabval***] [,DISCONVAL***=disconval***][,DISCABVAL***=discabval***];***

DISPLAY — Displays flow control information.

DISPLAY-L2FLOW:[LINK=link***][,FCLEVEL***=fclevel***];***

Table 5-13: Default Flow Control Values

FCLEVEL	CONGONVAL	CONGABVAL	DISCONVAL	DISCABVAL
Multiple Congestion and Multiple Priority: (e.g. ANSI)				
1	20	0	38	18
2	56	36	74	54
3	92	72	110	90
Standard ITU				
1	76	42	n/a	n/a

PARAMETERS

link — Link identification entered as a 12-character alphanumeric label.

fclevel — Numerical value for the flow control level or an * to display all values. The valid range depends on the setting of the ***mcong*** parameter in the MTP managed object:

- ON 1 to 3
- OF 1

congonval — The number of messages in the queue that indicate the onset of congestion. It is an integer from 0 to 127. Default values for the ***fclevels*** are in Table 5-13.

congabval — The number of messages in the queue at which congestion ends. It is an integer from 0 to 127. Default values for the ***fclevels*** are in Table 5-13.

disconval — The number of messages in the queue at which to begin discarding messages. It is an integer from 0 to 127. Default values for the ***fclevels*** are in Table 5-13.

discabval	The number of messages in the queue at which to stop discarding messages. It is an integer from 0 to 127. Default values for the ***fclevels*** are in Table 5-13.

***Important**: DISCONVAL and DISCABVAL are applicable only if the **MPRIO** parameter of MTP MO is set to ON.*

ERRORS

<ERROR>::threshold value is out of range

<ERROR>::threshold level is out of range

<ERROR>::Missing FCLEVEL parameter.

<ERROR>::Missing LINK parameter

<ERROR>::CONGONVAL is less than previous level value.

<ERROR>::CONGONVAL is greater than next level value.

<ERROR>::CONGONVAL is less than CONGABVAL.

<ERROR>::CONGONVAL is greater than DISCONVAL.

<ERROR>::CONGABVAL is less than previous level value.

<ERROR>::CONGABVAL is greater than next level value.

<ERROR>::CONGABVAL is greater than CONGONVAL.

<ERROR>::CONGABVAL is greater than DISCABVAL.

<ERROR>::DISCONVAL is less than previous level value.

<ERROR>::DISCONVAL is greater than next level value.

<ERROR>::DISCONVAL is less than DISCABVAL.

<ERROR>::DISCONVAL is less than CONGONVAL.

<ERROR>::DISCABVAL is less than previous level value.

<ERROR>::DISCABVAL is greater than next level value.

<ERROR>::DISCABVAL is greater than DISCONVAL.

<ERROR>::DISCABVAL is less than CONGABVAL.

<ERROR>::LINK MO instance does not exist

<ERROR>::Nothing to list

EXAMPLES

MODIFY-L2FLOW:LINK=Link_1_1***,FCLEVEL***=2***,CONGONVAL***=60***;***
DISPLAY-L2FLOW:LINK=l1***,FCLEVEL***=****;***

SAMPLE OUTPUT

(MCONG=OFF, MPRIO=OFF)

```
MML_TH>DISPLAY-L2FLOW:LINK=11;
-------------------------------------------------------------------------
LINKFCLEVELCONGONVALCONGABVALDISCONVALDISCABVAL
-------------------------------------------------------------------------
11 1 76 42
<SUCCESS>:: 1 records found.
```

(MCONG=ON, MPRIO=OFF)

```
MML_TH>DISPLAY-L2FLOW:LINK=11;
-------------------------------------------------------------------------
LINKFCLEVELCONGONVALCONGABVALDISCONVALDISCABVAL
-------------------------------------------------------------------------
11 1 20 0 3818
11 2 56 36 7454
11 3 92 72 11090
<SUCCESS>:: 3 records found.
```


5.2.8 Level-2 Timer (L2TIMER)

NAME

L2TIMER Modifies or displays the MTP Level-2 timer values.

COMMANDS

MODIFY Modifies MTP Level 2 timer values for a specific link from their defaults. All links can have different level-2 timer values.

MODIFY-L2TIMER:LINK=link***,TIMER***=timer***,VALUE***=value***;***

DISPLAY Displays the MTP Level 2 timer values.

DISPLAY-L2TIMER:[LINK=link***][,TIMER***=timer***];***

PARAMETERS

link Link identification entered as a 12-character alphanumeric label.

timer It is an integer from 0 to 8, as listed in Table 5-14 on page 5-58, or an * to display all timers for a given LINK.

value Numerical value specifying the timer value in milliseconds. It must be in the range defined in Table 5-14.

ERRORS

<ERROR>::Missing LINK parameter

<ERROR>::Missing TIMER parameter.

<ERROR>::LINK MO instance does not exist.

<ERROR>::Missing VALUE parameter.

<ERROR>::parameter value out of range.

<ERROR>::Nothing to list

EXAMPLES

MODIFY-L2TIMER:LINK=l11***,TIMER***=1***,VALUE***=120***;***
DISPLAY-L2TIMER:TIMER=l1***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-L2TIMER:LINK=l1;
---------------------------------------------
LINKTIMERVALUEMINVALMAXVAL
---------------------------------------------
l1  0  160  0  500
l1 1 130001300013000
l1 2 118001150023000
l1 3 115001150011500
l1 4 230023002300
l1  5  80  80  120
l1 6 5000300012000
l1 7 2000500 6000
l1 8 600 600 600
<SUCCESS>:: 9 records found
```

***Note**: Level 2 timers have a precision of 20 ms and are rounded to the nearest multiple-of-20-ms value. Therefore, the values are multiples of 20.*

Table 5-14: MTP-L2 Timer Definitions

L2 Timers	Alias string	Integer	Range[1]		
			ITU-1992[2]	ANSI-1992[3]	ANSI-1996[4]
Timer for link idle-detect	T0	0	0 - 500 ms	0 - 500 ms	0 - 500 ms
Timer for aligned/ready	T1	1	40 s - 600 s	13.0 s	12.9 s - 16 s
Timer for not aligned/ waiting for destination activation	T2	2	5 s - 150 s	11.5 s - 23.0 s	5 s - 30 s
Timer for aligned/ waiting for alignment completion	T3	3	1 s - 2 s	11.5 s	5 s - 14 s
Normal Proving period timer	T4N	4	7.5 s - 120 s	2.3 s	2.3 s (+/- 10%)
Emergency Proving period timer	T4E	8	400 ms - 8s	600 ms	600 ms (+/- 10%)
Timer for sending SIB	T5	5	80 ms - 120 ms	80 ms - 120 ms	80 ms - 120 ms
Timer for monitoring remote congestion	T6	6	3 s - 12 s	3 s - 12 s	1 s - 2 s
Timer for excessive delay of acknowledgment	T7	7	500 ms - 6 s	500 ms - 6s	500 ms - 6 s

1 For ranges: s=seconds; ms=milliseconds

2 Timer values are rate-dependent; refer to CCITT Blue Book, Vol. 6, Fascicle VI.7-Rec. Q.703, Para.12

3 Timer values are rate-dependent; refer to ANSI T1.111-1992, Para.12.3

4 Timer values are rate-dependent; refer to ANSI T1.111-1996, Para.12.3

5.2.9 Level-3 Timer (L3TIMER)

NAME

L3TIMER Modifies or displays MTP Level-3 timer values.

COMMANDS

MODIFY Modifies MTP Level 3 timer values from their defaults.

MODIFY-L3TIMER:TIMER=timer***,VALUE***=value***;***

DISPLAY Displays information about the MTP level 3 timer values.

DISPLAY-L3TIMER:[TIMER=timer***];***

PARAMETERS

timer Numerical value for the MTP Level 3 timer, ranging from 1 to 31, as listed in Table 5-15 on page 5-60, or an * to display all timers.

value Numerical value specifying the timer value in milliseconds. It must be in the range defined in Table 5-15.

ERRORS

<ERROR>::Missing TIMER parameter.

<ERROR>::L3TIMER MO instance does not exist.

<ERROR>::Missing VALUE parameter.

<ERROR>::parameter value out of range.

<ERROR>::Nothing to list

EXAMPLES

MODIFY-L3TIMER:TIMER=1***,VALUE***=120***;***
DISPLAY-L3TIMER:TIMER=8***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-L3TIMER:TIMER=8;
------------------------------------
TIMERVALUEMINVAL MAXVAL
------------------------------------
8 1200 800   1200
<SUCCESS>:: 1 records found.
```

***Note**: Level 3 timers have a precision of 10 ms and are rounded to the nearest multiple-of-10-ms value. Therefore the values are multiples of 10.*

Table 5-15: MTP-L3 Timer Definitions

L3 Timers	Alias String	Integer	ITU[1] Range[2]	ANSI-92/96[3] Range[2]
Delay to avoid message mis-sequencing on changeover	T1	1	0.5 - 1.2 s	0.5 - 1.2 s
Wait for changeover ACK	T2	2	0.7 - 2 s	0.7 - 2 s
Time-controlled delay on changeover	T3	3	0.5 - 1.2 s	0.5 - 1.2 s
First Wait for changeback ACK	T4	4	0.5 - 1.2 s	0.5 - 1.2 s
Second Wait for changeback ACK	T5	5	0.5 - 1.2 s	0.5 - 1.2 s
Delay to avoid message mis-sequencing on re-routing	T6	6	0.5 - 1.2 s	0.5 - 1.2 s
Wait for signaling data link connection ACK	T7	7	1 - 2 s	1 - 2 s
Transfer-prohibited inhibited timer (transient solution)	T8	8	0.8 - 1.2 s	0.8 - 1.2 s
Wait to repeat signaling route set test message	T10	10	30 - 60 s	30 - 60 s
Transfer-restricted timer	T11	11	30 - 90 s	30 - 90 s
Wait for uninhibit ACK	T12	12	0.8 - 1.5 s	0.8 - 1.5 s
Wait for force uninhibit	T13	13	0.8 - 1.5 s	0.8 - 1.5 s
Wait for inhibit ACK	T14	14	2 - 3 s	2 - 3 s
Wait to start route set congestion test	T15	15	2 - 3 s	2 - 3 s
Wait for route set congestion status update	T16	16	1.4 - 2 s	1.4 - 2 s
Delay to avoid oscillation of initial alignment failure and link re-start	T17	17	0.8 - 1.5 s	0.8 - 1.5 s
Wait for available links at re-starting STP (ITU) Repeat TFR once by response method (ANSI-96)	T18	18	40 s	2 s - 20 s
Wait for all traffic re-start messages at re-starting STP after T18	T19	19	67 - 69 s	480 - 600 s
Wait to broadcast traffic re-start allowed messages at re-starting STP after T19	T20	20	59 - 61 s	90- 120 s
Wait to re-start traffic through adjacent SP at re-starting SP having no STP function	T21	21	63 - 65 s	90 - 120 s
Local inhibit test timer (ITU) Waiting for signaling links become available at restart-ing SP (ANSI)	T22	22	3 min - 6 min	90 - 120 s
Remote inhibit test timer Waiting to receive all TRA message (ANSI)	T23	23	3 min - 6 min	10 s
Stabilizing after removal of LPO (ITU) Waiting to broadcast all TRA messages (ANSI)	T24	24	500ms	10 s
Waiting for traffic restart allowed message (ANSI)	T25	25	0 (not used)	30 - 35 s
Waiting for repeat traffic restart waiting message (ANSI	T26	26	0 (not used)	12 - 15 s
Minimum duration of unavailability for full restart	T27	27	0 (not used)	2 - 5 s

1 *Timer values are rate-dependent; refer to ITU white book 1992. Rec. Q.704, Para. 16.8*

2 *For ranges: s=seconds; ms=milliseconds; min=minutes*

3 *Timer values are rate-dependent; refer to ANSI T1.111-1992/1996, Para. 16.7*

Table 5-15: MTP-L3 Timer Definitions (Continued)

L3 Timers	Alias String	Integer	ITU[1] Range[2]	ANSI-92/96[3] Range[2]
Waiting for traffic restart waiting message (ANSI)	T28	28	0 (not used)	3 - 35 s
Timer started when TRA sent in response to unexpected TRA or TRW (ANSI)	T29	29	0 (not used)	60 - 65 s
Timer to limit sending of TFPs and TFRs in response to unexpected TRA or TRW (ANSI)	T30	30	0 (not used)	30 - 35 s

1 Timer values are rate-dependent; refer to ITU white book 1992. Rec. Q.704, Para. 16.8

2 For ranges: s=seconds; ms=milliseconds; min=minutes

3 Timer values are rate-dependent; refer to ANSI T1.111-1992/1996, Para. 16.7

5.2.10 Line (LINE)

NAME

LINE — Modifies or displays the configuration of an SS7 line on a board.

COMMANDS

MODIFY — Modifies a class instance of an unconfigured SS7 board. LINE cannot be modified unless the SS7 board configuration is set to OFF (MODIFY-SS7BOARD:CONF=OFF;).

MODIFY-LINE:HOSTNAME=hostname***,BOARDNM***=boardnm***,***
INST=inst***,SPAN***=span***[,LINE_FRMMOD***=line_frmmod***]***
[,LINE_COD=line_cod***][,LINE_LEN***=line_len***]***
[,LINE_IMP=line_imp***][,LINE_LPBK***=line_lpbk***]***
[,LINE_NTFY=line_ntfy***][,LINE_ACCS***=line_accs***];***

DISPLAY — Displays class instance(s).

DISPLAY-LINE:[HOSTNAME=hostname***][,BOARDNM***=boardnm***]***
[,INST=inst***][,SPAN***=span***];***

PARAMETERS

hostname — Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm — Board type, entered as one of the following values:

- **sbs334** common name for 4-port Sbus boards (sbs37x)
- **pci334** common name for 4-port PCI bus boards (pci37x)
- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)
- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)
- **artic8260** common name for 64-port CompactPCI bus boards (artic1000 and artic2000)
- **vbrd** 32-port virtual board driver (vbrd)

Note: *Although PCI3xPQ and PCI3xAPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full bandwidth on all configured links.*

inst — Identifies the SS7 board driver instance number. It is an integer from 0 to 8.

span — Identifies the span number. It is an unsigned integer from 1 to 8 that is based on the board:

- Class II/E1 and III/T1 boards**1** or **2**
- Class IV E1 or T1 boards**1** to **8**

line_frmmod Line framemod is a class II(E1) and class III(T1) type board parameter, entered as one of the following values:

• E1:	E1CRC4 *(default)*
	E1FEBE
	E1BASIC
• T1:	T1ESF *(default)*
	T1ZBTSI
	T1SLC96
	TISF4
	T1SFRM

line_cod Line code is a class II (E1) and class III (T1) type board parameter, entered as one of the following values:

• E1:	E1HDB3 *(default)*
	AMI
• T1:	T1B8ZS *(default)*
	T1B7ZS
	AMI

line_len Line length is a class III or class IV type board parameter, entered as one of the following values:

Class III boards:

- L133 *(default)*
- L266
- L399
- L533
- L655

Class IV boards:

- L110 *(default)*
- L220
- L330
- L440
- L550
- L660
- LB000
- LB5
- LB150
- LB225

line_imp Line impedance is a class II (E1) type board parameter entered as one of the following values:

- E1: I120 *(default)*
  I75

line_lpbk Loopback function is set with one of the following values:

- NONE *(default)*
- LOCAL
- REMOTE

line_ntfy Turns on or off the notification of framer line alarms with one of the following values:

- ON
- OF *(default)*

line_typ Identifies the Primary Rate Interface. This parameter sets the line interface for class IV (E1/T1) type boards. This is a read-only parameter for other boards (class II/E1) and class III/T1). It is a character string that accepts the following values:

- E1 interface at 2048 kbit/sec
- T1 interface at 1544 kbit/sec

line_accs Selects front or rear access for line interfaces on ARTIC1000/2000 boards. For boards other than ARTIC1000 and ARTIC2000 an error will be returned. The default value is FRONT for SBS334, PCI334, PCI3XPQ, PCI3XAPQ, CPC3XPQ, PMC8260-F variant, ARTIC2000 and ARTIC1000 without RTB. The LINE_ACCS default value is REAR for PMC8260-R variant and ARTIC1000 with RTB. Acceptable values are:

- **FRONT**
- **REAR**

ERRORS

<ERROR>::Missing BOARDNM attribute

<ERROR>::Missing INST attribute

<ERROR>::Missing SPAN attribute

<ERROR>::LINE_LEN is class III (T1) attribute

<ERROR>::LINE_IMP is class II (E1) attribute

<ERROR>::No such a LINE MO instance

<ERROR>::Can not add line MOs

<ERROR>::Can not delete line MOs

<ERROR>::ACCESS value can not be modified for this board type

EXAMPLES

MODIFY-LINE:HOSTNAME=ultra5***,BOARDNM***=pci334***,INST***=0***,SPAN***=B***,***
LINE_FRMMOD=E1CR4***,LINE_COD***=E1HDB3***,LINE_IMP***=I120***,***
LINE_LPBK=NONE***,LINE_NTFY***=OFF***;***
DISPLAY-LINE:;
DISPLAY-LINE:HOSTNAME=ultra5***,BOARDNM***=pci334***,INST***=0***,SPAN***=B***;***

SAMPLE OUTPUT

```
MML_TH> DISPLAY-LINE:;
------------------------------------------------------------------------------------------------------
HOSTNAME BOARDNM INST SPANCLASS LINE_TYPLINE_FRMMODLINE_CODLINE_LENLINE_IMPLINE_LPBKLINE_NTFY LINE_ACCS
------------------------------------------------------------------------------------------------------
diablo   pmc8260    0    1  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF     FRONT
diablo   pmc8260    0    2  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF     FRONT
diablo   pmc8260    0    3  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF     FRONT
diablo   pmc8260    0    4  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF     FRONT
diablo   artic8260  0    1  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF      REAR
diablo   artic8260  0    2  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF      REAR
diablo   artic8260  0    3  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF      REAR
diablo   artic8260  0    4  IV       T1       T1ESF   T1B8ZS     L110  I100   NONE    OFF      REAR

 <SUCCESS>:: 4 records found
```

Table 5-16: LINE Display Values

CLASS	
This read-only parameters identifies the board’s preassigned hardware class type. The board’s hardware determines the value.	
I	class 1 64kbits/s board
II	class 2 E1 board
III	class 3 T1board

5.2.11 Line 24-Hour Performance Data (LINEHIST)

NAME

LINEHIST Modifies or displays 24-hour performance data of an SS7 line on a board.

COMMANDS

MODIFY Modifies the 24-hour performance data of an SS7 line on a board.

MODIFY-LINEHIST: HOSTNAME=hostname***,***
BOARDNM=boardnm***,INST***=inst***,SPAN***=span***,RESET***=reset***;***

DISPLAY Displays the 24-hour performance data of an SS7 line on a board.

DISPLAY-LINEHIST:[HOSTNAME=hostname***]***
[,BOARDNM=boardnm***][,INST***=inst***][,SPAN***=span***];***

PARAMETERS

hostname Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)
- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)

Note*: Although PCI3xPQ, PCI3xAPQ and CPC3xPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full band-width on all configured links.*

inst Identifies the SS7 board driver instance number. The getcfg command provides the configured SS7 board slot, driver, and instance information.

span Identifies the span number. It is an unsigned integer from 1 to 8 that is based on the board:

- Class II/E1 and III/T1 boards**1** or **2**
- Class IV E1 or T1 boards**1** to **8**

reset Reset option, entered as either **YES** or **NO**. If entered as **YES** all of the the history will be deleted.

ERRORS

<ERROR>:: No such SS7BOARD MO instance

<ERROR>:: Board name must be one of class II or III boards

<ERROR>:: Missing BOARDNM attribute

<ERROR>:: Missing INST attribute

<ERROR>:: Missing SPAN attribute

<ERROR>:: Can not get 24-Hour performance data

<ERROR>:: Can not modify 24-Hour performance data

EXAMPLES

MODIFY-LINEHIST:HOSTNAME=ultra5***,*** ***BOARDNM***=pci3xpq***,*** ***INST***=0***,*** ***SPAN***=1***,*** ***RESET***=YES***;***
DISPLAY-LINEHIST:HOSTNAME=ultra5***,*** ***BOARDNM***=pci3xpq***,*** ***INST***=0***,*** ***SPAN***=1***;***
DISPLAY-LINEHIST:HOSTNAME=ultra5***,*** ***BOARDNM***=pci3xpq***,*** ***INST***=0***;***
DISPLAY-LINEHIST:HOSTNAME=ultra5***,*** ***BOARDNM***=pci3xpq***;***
DISPLAY-LINEHIST:HOSTNAME=ultra5***;***
DISPLAY-LINEHIST:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LINEHIST:HOSTNAME=ultra5,BOARDNM=pci3xpq,INST=0,SPAN=1;
------------------------------------------------------
        HOSTNAME   BOARDNM INST SPAN INTERVAL   ES UAS
------------------------------------------------------
          ultra5   pci3xpq    0    1        1    0   0
          ultra5   pci3xpq    0    1        2    0   0
          ultra5   pci3xpq    0    1        3    0   0
          ultra5   pci3xpq    0    1        4    0   0
          ultra5   pci3xpq    0    1        5    0   0
          ultra5   pci3xpq    0    1        6    0   0
          ultra5   pci3xpq    0    1        7    0   0
          ultra5   pci3xpq    0    1        8    0   0
          ultra5   pci3xpq    0    1        9    0   0
          ultra5   pci3xpq    0    1       10    0   0
          ultra5   pci3xpq    0    1       11    0   0
          ultra5   pci3xpq    0    1       12    0   0
          ultra5   pci3xpq    0    1       13    0   0
          ultra5   pci3xpq    0    1       14    0   0
          ultra5   pci3xpq    0    1       15    0   0
          ultra5   pci3xpq    0    1       16    0   0
          ultra5   pci3xpq    0    1       17    0   0
          ultra5   pci3xpq    0    1       18    0   0
          ultra5   pci3xpq    0    1       19    0   0
          ultra5   pci3xpq    0    1       20    0   0
          ultra5   pci3xpq    0    1       21    4  15
```

```
<SUCCESS>:: 21 records found
```

Table 5-17: LINEHIST Display Values

INTERVAL	ES	UAS
15-minute interval index. Interval 1 represents the most recent 15-minute interval.	Number of errored seconds. An errored second is a second with one or more error events.	Number of unavailable seconds. An unavailable second occurs when the signal state is unavailable.

5.2.12 Line STATISTICS (LINESTAT)

NAME

LINESTAT Modifies or displays statistics and performance parameters of an SS7 line on a board.

COMMANDS

MODIFY Modifies the statistics and performance parameters of an SS7 line on a board.

MODIFY-LINESTAT: HOSTNAME=hostname***,***
BOARDNM=boardnm***,INST***=inst***,SPAN***=span***,***
ERREVENTS=errevents***;***

*Note: **errevents** is the only parameter that can be modified.*

DISPLAY Displays statistics and performance parameters of an SS7 line on a board.

DISPLAY-LINESTAT:[HOSTNAME=hostname***]***
[,BOARDNM=boardnm***][,INST***=inst***][,SPAN***=span***];***

PARAMETERS

hostname Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)
- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)

***Note**: Although PCI3xPQ, PCI3xAPQ and CPC3xPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full band-width on all configured links.*

inst Identifies the SS7 board driver instance number. The getcfg command provides the configured SS7 board slot, driver, and instance information.

span Identifies the span number. It is an unsigned integer from 1 to 8 that is based on the board:

- Class II/E1 and III/T1 boards**1** or **2**
- Class IV E1 or T1 boards**1** to **8**

errevents Error events counter, entered as **0** to reset the counter. An error event is defined as the occurrence of a CRC6 error or an Out of Frame error.

ERRORS

<ERROR>:: No such SS7BOARD MO instance

<ERROR>:: Board name must be one of class II or III boards

<ERROR>:: Missing BOARDNM attribute

<ERROR>:: Missing INST attribute

<ERROR>:: Missing SPAN attribute

<ERROR>:: Can not get line statistics

<ERROR>:: Can not modify line statistics

EXAMPLES

MODIFY-LINESTAT:HOSTNAME=ultra5***,BOARDNM***=pci3xpq***,INST***=0***,***
SPAN=1***, ERREVENTS***=0***;***
DISPLAY-LINESTAT:HOSTNAME=ultra5***, BOARDNM***=pci3xpq***,INST***=0***,***
SPAN=1***;***
DISPLAY-LINESTAT:HOSTNAME=ultra5***,BOARDNM***=pci3xpq***,INST***=0***;***
DISPLAY-LINESTAT:HOSTNAME=ultra5***,BOARDNM***=pci3xpq***;***
DISPLAY-LINESTAT:HOSTNAME=ultra5***;***
DISPLAY-LINESTAT:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LINESTAT:;
-----------------------------------------------------------------------------------------------
HOSTNAME BOARDNM INST SPAN ERREVENTS CURSTATUS CURTIMER CUR-ES CUR-UAS 24H-ES 24H-UAS VLDINTTOTAL
-----------------------------------------------------------------------------------------------
  ultra5   pci3xpq    0    1 6018    SIG-AV      266      0       0      4      15          96
  ultra5   pci3xpq    0    2 979    SIG-AV      265      0       0      4       0          96
  ultra5  pci3xapq    0    1 0    SIG-AV      263      0       0      1      15          96
  ultra5  pci3xapq    0    2 4      SIG-AV      263      0       0      1       0          96
<SUCCESS>:: 4 records found
```

Table 5-18: LINESTAT Display Values

CURSTATUS	CURTIMER	CUR-ES
Current signal status. An unavailable signal state is declared after 10 consecutive seconds each has 320 or more CRC6 error events OR one or more Out of Frame error. • SIG-AV =Available • SIG-UNAV= Unavailable	Current 15-minute interval timer.	Number of errored seconds in current 15-minute interval. An errored second is a second with one or more error events.

CUR-UAS	24H-ES	24H-UAS
Number of unavailable seconds in current 15-minute interval. An unavailable second is a second when the signal state is unavailable.	Number of errored seconds in previous 24-hour period.	Number of unavailable seconds in previous 24-hour period.

Table 5-18: LINESTAT Display Values

VLDINTTOTAL
Number of valid 15-minute intervals in previous 24-hour period.

5.2.13 Link Status (LINKSTAT)

NAME

LINKSTAT Modifies or displays the state of an existing link.

COMMANDS

MODIFY Modifies the state of an existing link.

MODIFY-LINKSTAT:LINK=link***,STATUS***=status***;***

DISPLAY Displays status information about one or all existing links.

DISPLAY-LINKSTAT:[LINK=link***];***

PARAMETERS

link Link identification. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display all links.

status State of the link, entered as:

- **SET_ACT**activate link
- **CLR_ACT**deactivate link
- **CLR_EMR**clear emergency link alignment
- **SET_EMR**set emergency link alignment
- **CLR_ECO**clear emergency changeover
- **SET_ECO**set emergency changeover
- **CLR_INH**clear inhibit
- **SET_INH** set inhibit
- **CLR_LPO**clear local processor outage
- **SET_LPO**set local processor outage
- TEST_SLTMsend a single SLTM message over the link

ERRORS

<ERROR>::Missing LINK parameter.

<ERROR>::LINKSTAT MO instance does not exist.

<ERROR>::Parameter value is out of range.

<ERROR>::Nothing to list.

EXAMPLES

MODIFY-LINKSTAT:LINK=ls102030-01***,STATUS***=SET_ACT***;***
DISPLAY-LINKSTAT:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LINKSTAT:;
---------------------------------------------------------------
LINKLSETSLCLOADEDACTAVLEMRECOLINRINLPO RPO
---------------------------------------------------------------
l1 ls10 ON ON ONOFF OFFOFFOFF OFF OFF
l2 ls11 OFFOFFOFFOFFOFFON OFF OFF OFF
l3 ls12 OFFOFFOFFOFFOFFOFFON OFF OFF
l4 ls13 OFFOFFOFFON ONOFF OFF OFF OFF
<SUCCESS>:: 4 records found.
```

Table 5-19: LINKSTAT Display Values

LSET	SLC	LOADED	ACT	AVL
Link Set label.	Signaling Link Code, which is an integer 0 to 5	Link state: • **ON**=Link is loaded • **OFF**= link is not loaded	Activation state: • **ON**=activated • **OFF**= not activated	Availability state: • **ON**=available • **OFF**= not available

EMR	ECO	LIN	RIN	LPO
Emergency alignment state: • **ON**=emergency alignment set • **OFF**= not set	Emergency change over state: • **ON**=emergency change over set • **OFF**= not set.	Locally inhibited state: • **ON**=locally inhibited • **OFF**= not locally inhibited.	Remotely inhibited state. • **ON**=remotely inhibited • **OFF**= not remotely inhibited.	Local processor outage state. • **ON**=local processor outage • **OFF**= no local processor outage.

RPO
Remote processor outage state. • **ON**=remote processor outage • **OFF**= no remote processor outage.

5.2.14 Link Set Status (LSETSTAT)

NAME

LSETSTAT Modifies or displays the state of an existing link set.

COMMANDS

MODIFY Modifies the state of an existing link set.

MODIFY-LSETSTAT:LSET=lset***,STATUS***=status***;***

DISPLAY Displays status information about one or more link sets.

DISPLAY-LSETSTAT:[LSET=lset***];***

PARAMETERS

lset Link set identification. It is a string of 1 to 12 characters maximum, or an * to display all link sets.

status State of the link set, entered as:

- **SET_ACT** activate link set
- **CLR_ACT**deactivate link set

ERRORS

<ERROR>::Missing LSET parameter.

<ERROR>::LSETSTAT MO instance does not exist.

<ERROR>::Parameter value is out of range

<ERROR>::Nothing to list.

EXAMPLES

MODIFY-LSETSTAT:LSET=ls102675***,STATUS***=SET_ACT***;***
DISPLAY-LSETSTAT:LSET=ls1***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-LSETSTAT:;

--------------------------------

LSETDPC ACTAVL

--------------------------------

ls12-2-2ONOFF
```

```
<SUCCESS>:: 1 records found.
```

Table 5-20: LSETSTAT Display Values

DPC	ACT	AVL
Destination Point Code	Activation state: • ON=activated • OFF= not activated	Availability state: • ON=available • OFF= not available

5.2.15 Port (PORT)

NAME

PORT Modifies or displays the configuration of an SS7 port on a board.

COMMANDS

MODIFY Modifies the configuration of the SS7 port on a board. PORT cannot be modified until the SS7 board configuration is set to OFF. (MODIFY-SS7BOARD:CONF=OFF;)

MODIFY-PORT:HOSTNAME=hostname***,BOARDNM***=boardnm***,***
INST=inst***,PORTNUM***=portnum***[,TYPE***=type***][,BAUD***=baud***]***
[,LPBKMODE=lpbkmode***][,IDLEDETECT***=idledetect***];***

DISPLAY Displays the configuration settings of ports.

DISPLAY-PORT:[HOSTNAME=hostname***,BOARDNM***=boardnm***]***
[,INST=inst***][,PORTNUM***=portnum***];***

PARAMETERS

hostname Name of the host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **sbs334** common name for 4-port Sbus boards (sbs334/sbs37x)
- **pci334** common name for 4-port PCI bus boards (pci334/pci37x)
- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)
- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)
- **artic8260** common name for 64-port CompactPCI bus boards (artic1000 and artic2000)
- **vbrd** 32-port virtual board driver (vbrd)

Note: *Although PCI3xPQ, PCI3xAPQ and CPCxPQ boards allow configuration of up to 24 links, use of more than 16 for these boards is not recommended for systems requiring full bandwidth on all configured links.*

inst Identifies the SS7 board driver instance number. The *getcfg* command provides the configured SS7 board slot, driver, and instance information. It is an integer from **0** to **7**.

portnum Port number in the SS7 board, entered as a numerical value from **0** to **63** (system configuration dependent).

type Clocking type of the port, only entered for class I boards:

•DTE *(default)*

•DCE

baud Baud rate for the link on the port, entered as one of the following:

• 600

• 1200

• 2400

• 4800

• 7200

• 9600

• 16000*

• 19200

• 32000*

• 38400

• 48000*

• 56000*

• **64000*** *(default)*

Note: *Ports on E1/T1 boards support only the E1/T1 sub-channeling baud rates that have an *.*

lpmode Loopback mode is an all class parameter. It is a character string that accepts the following values:

• **NONE** *(default)*

• LOCAL

•REMOTE

idledetect Sets the line idle-detection facility of the port ON or OFF. When idledetect is ON, the board software monitors the port receive side for the occurrence of a line idle state and informs the port about a possible line idle state which leads the link on that port to go out of alignment. This mechanism allows board software to detect line disconnects or other party transmission problems which cannot be detected easily in some cases. It is a character string that accepts the following values:

• **OFF** turns off idle-detection

• **ON** turns on idle-detection *(default)*

This parameter enables/disables idle-detection through the board configuration commands. MTPL3 has also the MODIFY-L2TIMER command to enable/disable idle-detection by modifying the T0 value.

ERRORS

<ERROR>::Missing BOARDNM attribute

<ERROR>::Missing INST attribute

<ERROR>::Missing PORTNUM attribute

<ERROR>::Can not add port MOs

<ERROR>::Can not delete port MOs

<ERROR>::Port is already inuse

<ERROR>::Port is already free

<ERROR>::No such a PORT MO instance

EXAMPLES

MODIFY-PORT:HOSTNAME=host-A***,BOARDNM***=sbs334***,INST***=0***, PORTNUM***=0***,BAUD***=64000***,LPBKMODE***=NONE***,IDLEDETECT***=ON***;***
DISPLAY-PORT:;
DISPLAY-PORT:HOSTNAME=ultra5***,BOARDNM***=pci334***,INST***=0***, PORTNUM***=0***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-PORT:;
----------------------------------------------------------------------
HOSTNAMEBOARDNMINSTPORTNUMCLASSTYPEBAUDLPBKMODEIDLEDETECT
----------------------------------------------------------------------
ziwopci3xpq00 IIINOTUSED64000NONEON
ziwopci3xpq01 IIINOTUSED64000NONEON
ziwopci3xpq02 IIINOTUSED64000NONEON
ziwopci3xpq03 IIINOTUSED64000NONEON
<SUCCESS>:: 4 records found
```


5.2.16 MTP SLTM Timer (SLTIMER)

NAME

SLTIMER Modifies or displays the MTP SLTM timer values.

COMMANDS

MODIFY Modifies MTP SLTM (Q.707) timer values.

MODIFY-SLTIMER:TIMER=timer***,VALUE***=value***;***

DISPLAY Displays information about the MTP SLTM (Q.707) timer values.

DISPLAY-SLTIMER:[TIMER=timer***];***

PARAMETERS

timer MTP SLTM (Q.707) timer id entered as an integer from 1 to 2, or an * to display all MTP SLTM timers.

value Timer value entered as an integer. Valid ranges are in Table 5-21 on page 5-79.

ERRORS

<ERROR>::SLTIMER MO instance does not exist

<ERROR>::missing TIMER parameter

<ERROR>::missing VALUE parameter

<ERROR>::parameter value is out of range

<ERROR>::nothing to list

EXAMPLES

MODIFY-SLTIMER:TIMER=1***,VALUE***=120***;***
DISPLAY-SLTIMER:;

Note*: Level 3 timers have a precision of 10 ms and are rounded to the nearest multiple-of-10-ms value. Therefore the values shall be multiples of 10.*

Table 5-21: SLTIMER Definitions

SLTM Timers	Alias String	Integer	ITU[1] Range[2]	ANSI-92/96[3] Range[2]
Supervision timer for signaling link test acknowledgment message	T1	1	4 - 12 s	4 - 12 s
Interval timer for sending signaling link test messages	T2	2	30 -90 s	30 - 90 s

1 Timer values are rate-dependent; refer to ITU white book 1992. Rec. Q.707, Para.5.5
2 For ranges: s=seconds
3 Timer values are rate-dependent; refer to ANSI T1.111.7-1992/1996, Para.5.5

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SLTIMER:;
-------------------------------
TIMER   VALUE    MINVAL MAXVAL
-------------------------------
1    6000      4000  12000
2   90000     30000  90000
<SUCCESS>:: 2 records found
```

5.2.17 Signaling Point (SP)

NAME

SP Modifies or displays signaling point parameters.

COMMANDS

MODIFY Modifies the parameters of a logical signaling point.

MODIFY-SP:[SPNO=spno***][,NAME***=name***][,SPC***=spc***][,NI***=ni***]***
[,TYPE=type***];***

DISPLAY Displays network and signaling information about the signaling point.

DISPLAY-SP:[SPNO=spno***];***

PARAMETERS

spno Signaling point number entered as an integer from 0 to 7, or an * to display all signaling points. The SP number at which this command is entered is used if this optional parameter is not entered with the DISPLAY command.

name Name of signaling point, entered as a 10-character alphanumeric label.

spc Own signaling point code. It is entered in the x-y-z format. See page 5-2 for more information about this point code format.

ni Network indicator. It is a character string that accepts the following values:

- INTERNATIONAL (international)
- SPARE (international with spare bits set)
- NATIONAL (national)
- RESERVED (national with reserved bits set).

type Type of signaling point. It is a character string that accepts the following values:

- STP (Signaling Transfer Point)
- SEP (Signaling End Point)
- SEPWRT (Signaling End Point with Routing Option).

***Important**: All the parameters are optional, but at least one of them must be entered.*

ERRORS

<ERROR>::SP MO instance does not exist.

<EUUPEXISTS>::In order to modify SPC/NI parameters all user parts must terminate.

EXAMPLES

MODIFY-SP:NI=NATIONAL***,SPC***=10-20-10***,TYPE***=STP***;***
DISPLAY-SP:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SP:;
------------------------------------------------
SPNONAMESPCNITYPE
------------------------------------------------
0 DAMGR254-1-23NATIONALSEP
<SUCCESS>:: 1 records found.
```

5.2.18 Alias Point Code (ALIAS)

NAME

ALIAS Adds modifies, deletes, or displays alias point code information.

COMMANDS

ADD Adds second point code to the node.

ADD-ALIAS:APC=apc***[,OGPC***=ogpc***][,INFLTR***=infltr***]***
[,FLTRACT=fltract***];***

MODIFY Modifies any of the attributes of alias point code.

MODIFY-ALIAS:[APC=apc***][,OGPC***=ogpc***][,INFLTR***=infltr***]***
[,FLTRACT=fltract***];***

DELETE Deletes an alias point code.

DELETE-ALIAS:[APC=apc***];***

DISPLAY Displays the configuration information of alias point code.

DISPLAY-ALIAS:[APC=apc***];***

PARAMETERS

apc Alias point code of the node. It is entered in the x-y-z format. See page 5-2 for more information about this point code format.

ogpc Outgoing point code. Based on this parameter, decision is taken as to which point code is to be populated in the originating point code of the routing label. Values are:

- OFF Originating Point Code field of the routing label is populated with the Signaling Point Code (SPC) of the node.
- ON Originating Point Code field of the routing label is populated with the Alias Point Code (APC) of the node. *(default)*

infltr Incoming message filter. This parameter indicates if any incoming user part messages are to be filtered. Values are:

- OFF Incoming user part messages to either SPC or APC are not filtered.
- SPC Incoming user part messages to SPC are filtered. *(default)*
- APC Incoming user part messages to APC are filtered.

fltract Filter action. This parameter indicates the action to be taken for the filtered messages. Values are:

- ALARM Filtered messages are discarded with an alarm.
- UPU Filtered messages are discarded with an alarm, and a "User Part Unavailable" message is sent to the originator. *(default)*

ERRORS

<ERROR>::ALIAS MO does not exist

<ERROR>::Nothing to list

<ERROR>::ALIAS MO instance already exists

<ERROR>::MTP protocol not set

EXAMPLES

ADD-ALIAS:APC=2-3-4***,OGPC=***OFF***,INFLTR=***APC***,FLTRACT=***ALARM***;***
DISPLAY-ALIAS:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-ALIAS:;
--------------------------------------------------------------------------
ALIASOGPCINFLTRFLTRACT
--------------------------------------------------------------------------
2-3-4ONSPCUPU
<SUCCESS>:: 1 records found
```

5.2.19 Time Slot (TIMESLOT)

NAME

TIMESLOT Modifies or displays the configuration of an SS7 time slot on a board.

COMMANDS

MODIFY Modifies the configuration of the SS7 time slot on the board. A time slot cannot be modified until the SS7 board configuration is set to OFF (MODIFY-SS7BOARD:CONF=OFF;). The initial state for all time slots is ORIGTYPE=NOCONNECT and ORIGSLOT=0. This is because there is a time slot conflict detection mechanism that is activated with the MODIFY-TIMESLOT command. If any two time slots have the same ORIGTYPE and ORIGSLOT value pair (except ORIGTYPE=NOCONNECT) an error is returned informing the user about the conflict. This mechanism prevents accidental misuse of time slot mapping.

NOCONNECT timeslots are no longer kept in the TIMESLOT database. Therefore, only timeslots with an origtype other than NOCONNECT are displayed and if a timeslot origtype is modified to NOCONNECT its record is deleted from database.

For CTBUS spans the number of available slots changes with the data rate of a span that can be done with MODIFY-CTBUS command. Please refer to Section 5.2.21 on page 5-90 for more detail about this issue.

MODIFY-TIMESLOT:HOSTNAME=hostname***,***
BOARDNM=boardnm***,INST***=inst***,DESTTYPE***=desttype***,***
DESTSPAN=destspan***,DESTSLOT***=destslot,
ORIGTYPE=origtype***,ORIGSLOT***=origslot***,***
ORIGSPAN=origspan***;***

DISPLAY Displays the configuration settings for time slots.

DISPLAY-TIMESLOT:[HOSTNAME=hostname***]***
[,BOARDNM=boardnm***][,INST***=inst***][,DESTTYPE***=desttype***]***
[,DESTSPAN=destspan***][,DESTSLOT***=destslot***]***
[,ORIGSPAN=origspan***];***

PARAMETERS

hostname Name of the host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **sbs334** common name for 4-port Sbus boards (sbs334/sbs37x)
- **pci334** common name for 4-port PCI bus boards (pci334/pci37x)
- **pci3xpq** common name for 24-port PCI bus boards (pci37xpq)
- **pci3xapq** common name for 24-port PCI bus boards (pci37xapq)

- **cpc3xpq** common name for 24-port CompactPCI boards (cpc370pq/372apq)
- **pmc8260** common name for 64-port PCI boards (pmc8260)
- **artic8260** common name for 64-port CompactPCI bus boards (artic1000 and artic2000)
- **vbrd** 32-port virtual board driver (vbrd)

Identifies the SS7 board driver instance number. The *getcfg* command provides the configured SS7 board slot, driver, and instance information. It is an integer from 0 to 7.

desttype Identifies the destination span, the span to which PCM time slot data is directed. Desttype must be one of the following values:

- LINE Spans 1 to 2 on the class II, III boards
  Spans 1 to 8 on class IV boards.
  For class II, III and IV boards, LINE-1 maps to the older LINEB *(default)*, and LINE-2 maps to the older LINEA.
- **HDLC** HDLC controllers on the E1/T1 board
- **CTBUS** CT bus spans 0-31 on the class IV board

destspan Identifies the destination span number if the ***desttype*** parameter is either **LINE** or **CTBUS**. Ignored for ***desttype*** values other than **LINE** and **CTBUS**. For class II and III boards the range is 1 to 2. For class IV boards the range is 1 to the number available physical spans on the board if ***desttype*** is **LINE**, or 0 to 31 if ***desttype*** is **CTBUS**.

destslot Identifies the destination slot, which is the slot of the destination span to which PCM time slot data is directed.

- For LINE[1-8] spans:
  - (E1) board values must be in the range of **0** to **31**
  - (T1) board values must be in the range of **0** to **23**
- For HDLC spans, the destslot is in the range of **0** to the maximum number of ports on board.
- For CTBUS spans, the destslot is in the range of **0** to **127**

origtype Identifies the source span, the span from which PCM time slot data is coming from. Origtype can be one of the following values:

- LINE Spans 1 to 2 on the class II, III boards
  Spans 1 to 8 on class IV boards.
  For class II, III and IV boards, LINE-1 maps to the older LINEB *(default)*, and LINE-2 maps to the older LINEA.
- **HDLC** HDLC controllers on the E1/T1 board

• **NOCONNECT**No connection designator, not a real span

• **CTBUS** CT bus spans 0-31 on the class IV board

origspan Identifies the destination span number if the ***origtype*** parameter is either **LINE** or **CTBUS**. Ignored for ***origtype*** values other than **LINE** and **CTBUS**. For class II and III boards the range is 1 to 2. For class IV boards the range is 1 to the number available physical spans on the board if ***origtype*** is **LINE**, or 0 to 31 if ***origtype*** is **CTBUS**.

origslot Identifies the source slot, the slot of the source span from which PCM time slot data is coming from.

• For LINE[1-8] spans:

- (E1) board values must be in the range of **0** to **31**
- (T1) board values must be in the range of **0** to **23**

• For HDLC spans, the destslot is in the range of **0** to the maximum number of ports on board.

• For CTBUS spans, the destslot is in the range of **0** to **127**

• NOCONNECT spans must be **0**

ERRORS

<ERROR>::Missing BOARDNM attribute

<ERROR>::Missing INST attribute

<ERROR>::Missing DESTTYPE attribute

<ERROR>::Missing DESTSLOT attribute

<ERROR>::Missing ORIGTYPE attribute

<ERROR>::Missing ORIGSLOT attribute

<ERROR>::Can not add timeslot MOs

<ERROR>::Can not delete timeslot MOs

<ERROR>::Self connection in TIMESLOT assignment not allowed

<ERROR>::ORIGSLOT value must be 0 for NOCONNECT

<ERROR>::No such a TIMESLOT MO instance

<ERROR>::HDLC timeslot (port) listens to NOCONNECT

<ERROR>::HDLC timeslot (port) not listened by a timeslot

<ERROR>::No timeslot record(s) found, timeslot(s) in NOCONNECT state

EXAMPLES

MODIFY-TIMESLOT:HOSTNAME=host-A***,BOARDNM=****sbs334****,INST=****0****, DEST-TYPE=****HDLC****,DESTSLOT=****0****,ORIGTYPE=****LINEB****, ORIGSLOT=***0***;***
DISPLAY-TIMESLOT:;

SAMPLE OUTPUT

```
MML_TH>dis-timeslot:;
----------------------------------------------------------------------------------------------------
HOSTNAME BOARDNM INST DESTTYPE DESTSPAN DESTSLOT CLASS ORIGTYPE ORIGSPAN ORIGSLOT
----------------------------------------------------------------------------------------------------
ziwo     pmc8260 0    LINE      1         1       IV    HDLC      -       0
ziwo     pmc8260 0    LINE      1         2       IV    HDLC      -       1
ziwo     pmc8260 0    LINE      2         1       IV    HDLC      -       2
ziwo     pmc8260 0    LINE      2         2       IV    HDLC      -       3
ziwo     pmc8260 0    HDLC      -         0       IV    LINE      1       1
ziwo     pmc8260 0    HDLC      -         1       IV    LINE      1       2
ziwo     pmc8260 0    HDLC      -         2       IV    LINE      2       1
ziwo     pmc8260 0    HDLC      -         3       IV    LINE      2       2
ziwo     pmc8260 0    HDLC      -         4       IV    CTBUS     0       0
ziwo     pmc8260 0    HDLC      -         5       IV    CTBUS     0       1
ziwo     pmc8260 0    CTBUS     0         0       IV    HDLC      -       4
ziwo     pmc8260 0    CTBUS     0         1       IV    HDLC      -       5
```

Table 5-22: TIMESLOT Display Values

CLASS
This read-only parameters identifies the preassigned hardware class type of the board to which this time slot belongs. Board hardware determines the class parameter value, which can be one of the following: II : class 2 E1 type line III : class 3 T1 type line

5.2.20 MTP Level-2 Status (L2CS)

NAME

L2CS Displays MTP Level-2 status information.

COMMANDS

DISPLAY Displays MTP Level-2 cumulative status information.

DISPLAY-L2FLOW:[LINK= link ***];***

PARAMETERS

link Link name. It is a string of 1 to 12 alphanumeric characters maximum, or an * to display all links.

ERRORS

<ERROR>::LINK MO instance does not exist

<ERROR>::nothing to list

EXAMPLES

DISPLAY-L2CS:LINK= l1***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-L2CS:LINK=l1;
---------------------------------------------------------------------
LINKSTATTMINSRVSUERMALGNFLINKFRSU_ED_RXLD_TXLD_BOTXFRAMESRXFRAMES
---------------------------------------------------------------------
l11IS6 02 0 15 0 0 03 3
      ---------------------
      TXOCTETSRSOCTETS
      ---------------------
      7171
<SUCCESS>:: 1 records found
```

5.2.21 CT Bus (CTBUS)

NAME

CTBUS Modifies or displays the configuration of the CT Bus logic on a board. For more information about CT Bus (H.110/H.100), please refer to documents 'H.110/H.100 Hardware Compatibility Specification: CT Bus, revision 1.0, 1997, Enterprise Computer Telephony Forum (ECTF)'. These documents can be downloaded from web site of ECTF at http://www.ectf.org/.

COMMANDS

MODIFY Sets the configuration of the CT Bus on the board. The CT Bus cannot be modified until the SS7 board configuration is set to OFF (MODIFY-SS7BOARD:CONF=OFF;).

CTBUS MO applies to class IV boards like PMC8260, ARTIC1000 and ARTIC2000. Use of MODIFY-CTBUS command for boards other than the mentioned above will return an error.

MODIFY-CTBUS:HOSTNAME=hostname***,BOARDNM***=boardname***,*** ***INST***=instance***[,REFCLK***=reflck***][,REFINV***=refinv***][,FBMODE***=fbmode***]*** ***[,FBSPAN***=fbspan***][,COMP***=comp***][,C8A***=c8a***][,C8B***=c8b***]*** ***[,NRMODE***=nrmode***][,NRPSAN***=nrspan***][,NR8KHZ***=nr8khz***][,NRINV***=nrinv***]*** ***[,NR1***=nr1***][,NR2***=nr2***][,GRP_A***=grp_a***][,GRP_B***=grp_b***][,GRP_C***=grp_c***];*** ***[,GRP_D***=grp_d***][,GRP_E***=grp_e***][,GRP_F***=grp_f***][,GRP_G***=grp_g***]*** ***[,GRP_H***=grp_h***];***

DISPLAY Displays the configuration settings and status for ctbus. The fallback status and the netref clock output status are shown if the board is configured ON otherwise a '-' will be printed. When fallback status ***FB*** column is **ON** the fallback clock selection is active. When netref clock output status **NRACT** is **OFF** netref output has been shut off by the board due to a reference clock loss.

DISPLAY-CTBUS:[HOSTNAME=hostname***][,BOARDNM***=boardnm***][,INST***=inst***];***

PARAMETERS

hostname Name of the host. It is a string of 1 to 15 alphanumeric characters maximum.

boardnm Board type, entered as one of the following values:

- **pmc8260** - common name for 64-port CompactPCI bus boards
- (pmc8260)
- **artic8260** - common name for 64-port CompactPCI bus boards
- (artic8260)

inst Identifies the SS7 board driver instance number. The **getcfg** command provides the configured SS7 board slot, driver, and instance information.

refclk Identifies primary reference clock source when board is in ***REMOTE*** clock mode. Refclk must be one of the following values:

- **C8A,C8B** ECTF H.1x0 bus 8.192Mhz A or B clocks.
- **NETREF1,NETREF2** ECTF H.1x0 bus network reference clocks.
- **SCSA2,SCSA4,SCSA8** SCSA bus 2.048, 4.096 or 8.192 Mhz clocks.
- **MVIP,HMVIP** MVIP or H-MVIP bus clocks, H.100 bus only.

refinv Identifies primary reference clock inversion switch value. Refinv must be one of the following values:

- **ON** invert primary reference clock.
- **OFF** do not invert primary reference clock.

fbmode Identifies fallback clock mode activated when primary reference clock failed. Fbmode must be one of the following values:

- **C8A,C8B** ECTF H.1x0 bus 8.192Mhz A or B clocks.
- **NETREF1,NETREF2** ECTF H.1x0 bus network reference clocks.
- **INTERNAL** Internal board clock.
- **LINE** Line recovered clock, requires FBSPAN to be entered.

fbspan Identifies fallback clock span when FBMODE value is LINE. Fbspan must be one of the following values: **1,2,3,4,5,6,7,8.**

comp Identifies compatibility clocks output mode selection. Comp must be one of the following values:

- **ECTF** disables compatibility clocks.
- **SCSA2,SCSA4,SCSA8** enables 2.048, 4.096 or 8.192 MHz SCSA clocks.
- **MVIP,HVMIP** enables MVIP or HMVIP clocks, H.100 bus only.

c8a Identifies ECTF H.1x0 C8A clock output enable/disable switch value. C8a must be one of the following values:

- **ON** enable C8A clock output.
- **OFF** disable C8A clock output.

c8b Identifies ECTF H.1x0 C8B clock output enable/disable switch value. C8b must be one of the following values:

- **ON** enable C8B clock output.
- **OFF** disable C8B clock output.

nrmode Identifies network reference clock source mode. Nrmode must be one of the following values:

- **NETREF1,NETREF2** ECTF H.1x0 bus network reference clocks.
- **INTERNAL** Internal board clock.
- **LINE** Line recovered clock, requires NRSPAN to be entered.

nrspan Identifies network reference clock span when NRMODE value is LINE. Nrspan must be one of the following values: **1,2,3,4,5,6,7,8**.

nr8khz Identifies network reference clock 8KHz output switch value. Nr8khz must be one of the following values:

- **ON** enable 8KHz netref clock output.
- **OFF** disable 8KHz netref clock output.

nrinv Identifies network reference clock output inversion switch value. Nrinv must be one of the following values:

- **ON** enable netref clock output inversion.
- **OFF** disable netref clock output inversion.

nr1 Identifies network reference clock 1 output enable/disable switch value. Nr1 must be one of the following values:

- **ON** enable netref 1 clock output.
- **OFF** disable netref 1 clock output.

nr2 Identifies network reference clock 2 output enable/disable switch value. NR2 must be one of the following values:

- **ON** enable netref 2 clock output.
- **OFF** disable netref 2 clock output.

grp_a..grp_h Identifies span group A to H data rate, each group consists of 4 ctbus spans. Group A is 0-3, group B is 4-7, etc. Grp_x must be one of the following values:

- **OFF** disable (tri-state) spans in a group.
- **2048** span group rate is 2.048 Mbps, 32 timeslots.
- **4096** span group rate is 4.096 Mbps, 64 timeslots.
- **8192** span group rate is 8.192 Mbps, 128 timeslots.

***Important**: In order to be able to do the TIMESLOT configuration for CTBUS spans user must first enable corresponding CTBUS span with the GRP_x attribute by setting it to an appropriate value the application demands. Otherwise, the user will get an error stating that the timeslot value is out of the acceptable range.*

ERRORS

<ERROR>::Missing BOARDNM attribute

<ERROR>::Missing INST attribute

<ERROR>::Can not add ctbus MO

<ERROR>::Can not delete ctbus MO

<ERROR>:: Cannot modify FALLBACK attribute

<ERROR>:: Can not modify ctbus MO

<ERROR>:: Can not get ctbus status

<ERROR>::No such a ctbus MO instance

EXAMPLES

MODIFY-CTBUS:HOSTNAME=host-A***,BOARDNM***=artic8260***,INST***=0***,***
REFCLK=C8A***,FBMODE***=C8B***,COMP***=ECTF***,GRP_A***=2048***;***
DISPLAY-CTBUS:;

SAMPLE OUTPUT

```
MML_TH>dis-ctbus:;
----------------------------------------------------------------------------------------------------
HOSTNAME BOARDNM INST REFCLK REFINV FBMODE FBSPAN FB COMP C8A C8B NRMODE NRSPAN NR8KHZ NRINV NRACT NR1
----------------------------------------------------------------------------------------------------
diablo    pmc8260 0   C8A    OFF  DISABLED   -    -  ECTF OFF OFF  INTERNAL -   ON     OFF   -    OFF
diablo    pmc8260 1   C8A    OFF  DISABLED   -    -  ECTF OFF OFF  INTERNAL -   ON     OFF   -    OFF
diablo    pmc8260 2   C8A    OFF  DISABLED   -    -  ECTF OFF OFF  INTERNAL -   ON     OFF   -    OFF
diablo    pmc8260 3   C8A    OFF  DISABLED   -    -  ECTF OFF OFF  INTERNAL -   ON     OFF   -    OFF
diablo  artic8260 0   C8A    OFF  DISABLED   -    -  ECTF OFF OFF  INTERNAL -   ON     OFF   -    OFF
----------------------------------------------------
NR2 GRP_A GRP_B GRP_C GRP_D GRP_E GRP_F GRP_G GRP_H
----------------------------------------------------
OFF   2048  OFF   OFF   OFF   OFF   OFF   OFF   OFF
OFF   2048 4096  8192   OFF   OFF   OFF   OFF   OFF
OFF   2048  OFF   OFF   OFF   OFF   OFF   OFF   OFF
OFF   2048  OFF   OFF   OFF   OFF   OFF   OFF   OFF
OFF   2048  OFF   OFF   OFF   OFF   OFF   OFF   OFF
<SUCCESS>:: 5 records found
```

5.3 SCCPMMLCommands

5.3.1 Concerned Point Code (CPC)

NAME

CPC — Adds, deletes, or displays a Concerned Point Code, or information about a Concerned Point Code.

COMMANDS

ADD — Adds a new Concerned Point Code (CPC) to the subsystem of a Signaling Point Code (SPC) defined in the SCCP network database. The SP and the subsystem must already exist.

ADD-CPC:SPC=spc***,SSN***=ssn***,CPC***=cpc***;***

DELETE — Deletes the CPC from the Subsystem of the SPC defined in the SCCP network database.

DELETE-CPC:SPC=spc***,SSN***=ssn***,CPC***=cpc***;***

DISPLAY — Displays the CPC for the subsystems defined for the SPC from the SCCP network database.

DISPLAY-CPC:[SPC=spc***,SSN***=ssn***,CPC***=cpc***;]***

PARAMETERS

spc — Signaling point code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127

ssn — A subsystem number with a range of 2 to 255.

cpc — Concerned point code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127
- Asterisk (*****) for the entire list

ERRORS

<ERROR>:: Missing SPC parameter.

<ERROR>:: Missing SSN parameter.

<ERROR>:: Missing CPC parameter.

<ERROR>:: Wildcard cannot be used with this command.

<ERROR>:: sp not defined in sccp network.

<ERROR>:: subsystem not defined for sp.

<ERROR>:: sp cannot be concerned for itself.

<ERROR>:: cpc not defined in sccp network.

<ERROR>:: subsystem has a mate at cpc.

<ERROR>:: cpc already defined for subsystem.

<ERROR>:: cpc not defined for subsystem.

EXAMPLES

ANSI:	***ADD-CPC:SPC***=0-23-255***,SSN***=4***,CPC***=224-245-123***;***
CCITT:	***ADD-CPC:SPC***=3-125-6***,SSN***=4 ***CPC***=1-2-3***;***
ANSI:	***DELETE-CPC:SPC***=0-23-255***,SSN***=4***,CPC***=224-245-123***;***
CCITT:	***DELETE-CPC:SPC***=3-125-6***,SSN***=4***,CPC***=1-2-3***;***
ANSI:	***DISPLAY-CPC:SPC***=0-23-255***,SSN***=4***,CPC***=224-245-123***;***
CCITT:	***DISPLAY-CPC:SPC***=3-125-6***,SSN***=4***,CPC***=1-2-3***;***
Both:	***DISPLAY-CPC:SPC***=3-125-6***,SSN***=4***,CPC***=****;***
	DISPLAY-CPC:SPC=0-23-255***,SSN***=4***,CPC***=****;***

SAMPLE OUTPUT

```
------------------------------
SSN   SPC          CPC
------------------------------
254   1-1-1        2-2-2
<SUCCESS>:: 1 records found.
```

5.3.2 Global Title (GT)

NAME

GT — Adds, Deletes, Modifies or Displays a Global Title in the SCCP database.

COMMANDS

ADD — Provisions a global title into the SCCP database.

ADD-GT:GT=gt***,GTIE***=gtie***,TRTYPE***=trtype ***[, NUMPLAN***=numplan***,***
NATOFADDR=natofaddr,***LOADSHARE***=loadshare***],***
ADDRINFO=addrinfo***;***

DELETE — Deletes or removes one or more global titles from the SCCP database.

DELETE-GT: GT=gt***,GTIE***=gtie***,TRTYPE***=trtype***,***
[NATOFADDR=natofaddr,***NUMPLAN=numplan],ADDRINFO=***
addrinfo***;***

MODIFY — Modifies the LOADSHARE parameter of GT.

MODIFY-GT: GT=gt***,LOADSHARE***=loadshare***;***

DISPLAY — Displays one or more global titles in the SCCP database.

DISPLAY-GT:GT=gt***,GTIE=***gtie***,TRTYPE=***trtype***,***
[NATOFADDR=natofaddr,***NUMPLAN=numplan],ADDRINFO=***
addrinfo***];***

PARAMETERS

gt — Global title alias (8 bits).

gtie — Global Title Encoding (from 1 to 15).

trtype — Translation type (from 0 to 255).

numplan — Numbering plan (optional for ADD command, default value=1)

natofaddr — Nature of address indicator (optional) replaces ***trtype*** field when gti=1 or 4. (for ITU only)

addrinfo — Addressing information. Enter as a character string (each digit = 1 byte). All global titles which begin with or are exactly equal to this string of digits will be translated to the specified SPC. Global titles with fewer digits will not be translated/deleted/displayed by this entry. To delete/ display all, enter ‘*’.

loadshare — Loadsharing option for global title translation. It is a character string that accepts the following values:

•ON

•OF

If load-share is ON, GT related traffic is shared among related gt-entries If set to OFF gt-entries are used in an active/standby manner

ERRORS

<ERROR>:: Missing GTIE parameter.
<ERROR>:: Missing TRTYPE parameter.
<ERROR>:: TRTYPE undefined for GTIE=1 (CCITT).
<ERROR>:: Missing NATOFADDRIND parameter.
<ERROR>:: NATOFADDRIND only defined for GTIE=1 (CCITT).
<ERROR>:: Missing ADDRINFO parameter.
<ERROR>:: ADDRINFO too long.
<ERROR>:: Wildcard cannot be used with this command.
<ERROR>:: Address specified already provisioned.
<ERROR>:: Trtype specified not provisioned.
<ERROR>:: Invalid ADDRINFO.
<ERROR>:: Address specified not provisioned.
<ERROR>:: Only LOADSHARE can be modified.
<ERROR>:: DB inconsistency for GT and GTENTRY.

EXAMPLES:

ADD-GT:GT=GT1,**GT I E** =4,**TRTYPE**=0,**ADDRINFO**=12039251111;
DELETE-GTENTRY:GT=2,***GTIE***=4,***TRTYPE***=253, ***ADDRINFO***=8001234567;
DELETE-GTENTRY:GT=3,***GTIE***=4,***TRTYPE***=253,***ADDRINFO***=*;
MODIFY-GT:GT=GT1,***LOADSHARE***=ON;
DISPLAY-GT:IO=OUTGOING,**GT I E** =4,***TRTYPE***=253, ***ADDRINFO***=8001234567;
DISPLAY-GT:IO=INCOMING,**GT I E** =4,***TRTYPE***=253,***ADDRINFO***=*;
DISPLAY-GT:IO=OUTGOING,***GTIE***=4,***TRTYPE***=253,***ADDRINFO***=8003;

SAMPLE OUTPUT

```
----------------------------------------------------------------
GT GTIE  TRTYPE NATOFADDRIND      ADDRINFO
----------------------------------------------------------------
1  1     N/A    4             0f3a(HEX)
2  1     N/A    4             8006661234
<SUCCESS>:: 2 records found.
```

5.3.3 Global Title Entry (GTENTRY)

NAME

GTENTRY Adds, deletes, or displays a Global Title entry.

COMMANDS

ADD Provisions a global title into the SCCP database.

ADD-GTENTRY:IO=io***,GT***=gt***,SPC***=spc***[,SSN***=ssn***][,NEWGT***=newgt***],[ENTRYTYPE***=entrytype***] [,WILDCARD***=wildcard***][,XLATE_ID***=xlate_id***;***

DELETE Removes a global title entry.

DELETE-GTENTRY:IO=io***,GT***=gt***,[ENTRYTYPE***=entrytype***], [XLATE_ID***=xlate_id***]***

MODIFY Modifies SPC, SSN or NEWGT parameters of global title entry.

MODIFY-GTENTRY:IO=io***,GT***=gt***, ENTRYTYPE***=entrytype ***[,XLATE_ID***=xlate_id***] [,SPC***=spc***][,SSN***=ssn***][,NEWGT***=newgt***];***

DISPLAY Display one or more global title entries.

DISPLAY-GTENTRY:IO=io***,GT***=gt***,[ENTRYTYPE***=entrytype***[,XLATE_ID***= xlate_id***]];***

PARAMETERS

io Incoming or outgoing table. It must be either INCOMING or OUTGOING.

gt Global Title index (1 to 131,072)

entrytype Priority of SCCP entity set in the global translation table. It is a character string that accepts the following values:

- PRIMARY (default)
- SECONDARY

This attribute is optional in all command types.

xlate_id Defines a unique name for the gtentry. It can only be specified for gt-entries of entrytype **SECONDARY**. When the load-share attribute of the global title is set to **OFF**, gt-entries are used in an active/multi-standby manner and the xlate-id defines the order in which secondary gt-entries are used for translation (when the translation -spc, ssn- in the primary record becomes unavailable, secondary entries are used in alphabetical order to provide another available translation for the gt).

This attribute is optional in all command types and the default value is empty string.

spc Signaling Point Code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127

ssn	Subsystem number, value from **2** to **255** (optional parameter for ADD command).
newgt	New Global Title for translation
wildcard	Indicates whether the entry should be used for wildcard matches: •YES • NO (default)

ERRORS

<ERROR>:: Missing IO parameter.

<ERROR>:: IO must be either INCOMING or OUTGOING.

<ERROR>:: Wildcard cannot be used with this command.

<ERROR>:: Missing SPC parameter.

<ERROR>:: SP not defined in sccp network.

<ERROR>:: Subsystem not defined for sp.

<ERROR>:: Address specified already provisioned.

<ERROR>:: Primary GTENTRY not defined.

<ERROR>:: Secondary record exists.

<ERROR>:: WILDCARD can NOT be modified.

EXAMPLES:

ADD-GTENTRY:IO=INCOMING***,GT***=GT 1 ***,ENTRYTYPE***=SECONDARY***,***
XLATE_ID=xlate1***,SPC***=1-1-2***,SSN***=254***;***
DELETE-GTENTRY:IO=INCOMING***,GT***=GT 1 ***;***
MODIFY-GTENTRY:IO=INCOMING***,GT***=GT 1 ***,ENTRYTYPE***=SECONDARY***,***
XLATE_ID=xlate1***,SPC***=1-2-3***;***
DISPLAY-GTENTRY:IO=INCOMING***,GT***=GT 1 ***;***

5.3.4 Mate (MATE)

NAME

MATE — Adds, deletes, or displays two subsystems of different SPs as mates.

COMMANDS

ADD — Mates two subsystems at different SPs in the SCCP network database. Both of the SPs and the subsystems must exist in the database.

ADD-MATE:SPC=spc***,SSN***=ssn***,MSPC***=mspc***,MSSN***=mssn***;***

DELETE — Deletes the mate relationship between the SSNs.

DELETE-MATE:SPC=spc***,SSN***=ssn***,MSPC***=mspc***,MSSN***=mssn***;***

DISPLAY — Displays the mate of a subsystem defined for the Signaling Point Code (SPC) from the SCCP network database.

DISPLAY-MATE:[SPC=spc***,SSN***=ssn***];***

PARAMETERS

spc — Signaling point code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127

ssn — A subsystem number with a range of **2** to **255**.

mspc — Mate signaling point code entered in the same format as ***spc***.

mssn — A subsystem number with a range from **2** to **255**.

ERRORS

<ERROR>:: Missing SPC parameter.

<ERROR>:: Missing SSN parameter.

<ERROR>:: Missing MSPC parameter.

<ERROR>:: Missing MSSN parameter.

<ERROR>:: Sp not defined in sccp network.

<ERROR>:: Subsystem not defined for sp.

<ERROR>:: Mate sp not defined in sccp network.

<ERROR>:: Mate subsystem not defined for mate sp.

<ERROR>:: Subsystems of same sp cannot be mated.

<ERROR>:: Subsystem of sp already has a mate.

<ERROR>:: Mate sp defined as own id

<ERROR>:: Subsystem of mate sp already has a mate

<ERROR>:: Subsystems are not mated.

EXAMPLES

ANSI:	***ADD-MATE:SP***=0-23-255***,SSN***=4***,MSPC***=224-245-123***,MSSN***=8***;***
CCITT:	***ADD-MATE:SP***=3-125-6***,SSN***=4***,MSPC***=1-2-3***,MSSN***=8***;***
ANSI:	***DELETE-MATE:SPC***=0-23-255***,SSN***=4***,MSPC***=224-245-123***, MSSN*** =8***;***
CCITT:	***DELETE-MATE:SPC***=3-125-6***,SSN***=4***,MSPC***=123***,MSSN***=8***;***
ANSI:	***DISPLAY-MATE:SPC***=0-23-255***,SSN***=4***;***
CCITT:	***DISPLAY-MATE:SPC***=3-125-6***,SSN***=4***;***

SAMPLE OUTPUT

```
------------------------------------
SSN   SPC         MSSN  MSPC
------------------------------------
111   3-3-3       254   1-1-1
<SUCCESS>:: 1 records found.
```

5.3.5 SCCP (SCCP)

NAME

SCCP	Displays or modifies information of the SCCP

COMMANDS

DISPLAY Displays protocol-specific information of the working SCCP. This includes sp, variant, management address, and timer information.

DISPLAY-SCCP:;

MODIFY Modifies protocol-specific information of the working SCCP. Allowed fields consist of management format and timer values.

MODIFY-SCCP:[PCIND=pcind***][,PROTOCOL***=protocol***]***
[,VARIANT=variant***][,T_IAS***=t_ias***][,T_CONN_EST***=t_conn_est***]***
[,T_IAR=t_iar***][,T_REL***=t_rel***][,T_INT***=t_int***]***
[,T_GUARD=t_guard***][,T_RESET***=t_reset***]***
[,T_SEGMENT=t_segment***][,T_A***=t_a***][,T_D=t_d]***
[,T_CON=t_con***];***

PARAMETERS

pcind Include point code for SCCP management messages

•YES

•NO

protocol Protocol of the SCCP

• DEFAULT

• ANSI_92

• ANSI_96

• ITU_93

• ITU_97

variant Variant of the SCCP

•NONE

•TT

•APLUS

•SNET

t_conn_est	Connection T_CONN_EST timer value
t_ias	Connection T_IAS timer value
t_iar	Connection T_IAR timer value
t_rel	Connection T_REL timer value
t_int	Connection T_INT timer value
t_guard	Connection T_GUARD timer value

t_reset	Connection T_RESET timer value
t_segment	Segmented messsage T_SEGMENT timer value
t_a	Restriction level T_A timer value (ITU only)
t_d	Restriction level T_D timer value (ITU only)
t_con	SCCP/subsystem congestion level T_CON timer value (ITU only)

Note: *Timer values are decimal values in milliseconds. For an exact description of the timers, refer to the SCCP specifications.*

EXAMPLES

ANSI	***MODIFY-SCCP:PCIND***=YES***,T_IAS***=300***,T_REL***=700***,*** ***T_GUARD***=500***;***
ITU	***MODIFY-SCCP:PCIND***=YES***,T_IAS***=300***,T_REL***=700***,*** ***T_GUARD***=500***;***

SAMPLE OUTPUT

```
---------------------------------------------------------------------------------------------------------
 SPNO PROTOCOL VARIANT PCIND T_CONN_EST T_IAS T_IAR T_REL T_INT T_GUARD T_RESET T_SEGMENT T_A T_D T_CON
---------------------------------------------------------------------------------------------------------
 0    ANSI_96  NONE    NO    27000      45000 99000 1500  6000  6000    3000    3000      300  5000 5000
<SUCCESS>:: 1 record found
```

5.3.6 SCCP Signaling Point (SNSP)

NAME

SNSP Adds, deletes, or displays a Signaling Point in the SCCP network.

COMMANDS

ADD Adds a new signaling point to the SCCP network.The SPC must already be provisioned in the MTP network. When an SPC is added to the SCCP network, the SCCP management subsystem (SSN=1) is automatically created by the SCCP in order to display remote SCCP status in ITU WHITEBOOK networks. When a remote user part (SCCP) is unavailable, only one SST message is sent to the remote SCCP for SSN=1 until the remote SCCP is up. Subsystem SSN=1 can only be displayed by users to monitor the remote SCCP's status. It cannot be modified by users.

ADD-SNSP:SPC=spc;

DELETE Deletes the Signaling Point Code (SPC) from the SCCP network database. This command fails if the SPC does not exist in the database. The SCCP management subsystem (SSN=1) was automatically created by the SCCP when the first SPC was added. Subsystem SSN=1 exists to display remote SCCP status in ITU WHITEBOOK networks. When a remote user part (SCCP) is unavailable, only one SST message is sent to the remote SCCP for SSN=1 until the remote SCCP is up. Subsystem SSN=1 can only be displayed by users to monitor the remote SCCP's status. It cannot be modified by users. When the last SPC is removed from the SCCP network, the management subsystem (SSN=1) is also removed automatically.

DELETE-SNSP:SPC=spc***;***

DISPLAY Displays the Signaling Point Codes (SPC) from the SCCP network database. In ITU WHITEBOOK networks, the management subsystem (SSN=1) always exists and can only be displayed. When subsystem (SSN=1) is PROHIBITED, it means that the remote SCCP user part is unavailable.

DISPLAY-SNSP:[SPC=spc***];***

PARAMETERS

spc Signaling point code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127
- Asterisk (*****) for the entire list

ERRORS

<ERROR>:: Missing SPC parameter.

<ERROR>:: SPC not defined in MTP network.Add routeset first.

<ERROR>:: SP already defined in sccp network.

<ERROR>:: SP defined as own ID.

<ERROR>:: Wildcard cannot be used with this command.

<ERROR>:: SP has defined subsystems.

<ERROR>:: SP not defined in sccp network.

<ERROR>:: SP defined as concerned.

<ERROR>:: Nothing to list.

EXAMPLES

	ADD-SNSP:SPC=0-3-2;
ANSI:	***DELETE-SNSP:SPC=0-23-255;***
CCITT:	***DELETE-SNSP:SPC=3-125-6;***
ANSI:	***DISPLAY-SNSP:SPC=0-23-255;***
CCITT:	***DISPLAY-SNSP:SPC=3-125-6;***
Both:	***DISPLAY-SNSP:SPC=*;***

SAMPLE OUTPUT

```
-----------------------------------------------------------------
SPC STATUS XLATE CONCERNED SUBSYSTEMS
-----------------------------------------------------------------
3-125-6 ACCESSIBLE PRIMARY NO               YES
<SUCCESS>:: 1 records found.
```

Table 5-23: SNSP Display Values

STATUS	XLATE	CONCERNED	SUBSYSTEMS	RL
Status of the signaling point: ACCESSIBLE INACCESSIBLE	PRIMARY SECONDARY	Whether the signaling point is a concerned point code.	Whether subsystems are provisioned.	SCCP restriction level

RSL	CLS
SCCP restriction sub-level	SCCP congestion level

5.3.7 Subsystem (SUBSYS)

NAME

SUBSYS — Adds, Deletes, or Displays a subsystem or subsystem information for a Signaling Point Code (SPC)

COMMANDS

ADD — Adds a new subsystem to a SPC defined in the SCCP network database.

ADD-SUBSYS:SPC=spc***,SSN***=ssn***;***

DELETE — Deletes the subsystem from the SPC defined in the SCCP network database. The command fails if the subsystem or SPC does not exist in the database.

DELETE-SUBSYS:SPC=spc***,SSN***=ssn***;***

DISPLAY — Displays the subsystems defined for the SPC from the SCCP network database.

DISPLAY-SUBSYS:[SPC=spc***,[SSN***=ssn***];***

PARAMETERS

spc — Signaling point code entered as one of the following:

- Zone-Network-SPid (3-8-3) for CCITT networks
  Example:1-222-3
- Network-Cluster-Member (8-8-8) for ANSI networks
  Example:10-20-30
- 5-4-7 bit format for Japanese networks
  Example: 31-25-127

ssn — Subsystem number entered as one of the following

- number in the range of **2** to **255**
- asterisk (*****) for the entire list

ERRORS

<ERROR>:: Missing SPC parameter.

<ERROR>:: Missing SSN parameter.

<ERROR>:: Wildcard cannot be used with this command.

<ERROR>:: SP not defined in sccp network.

<ERROR>:: Subsystem already defined for sp.

<ERROR>:: SP defined as own id.

<ERROR>:: Subsystem not defined for sp.

<ERROR>:: Subsystem has defined cpc's.

<ERROR>:: Subsystem has a mate.

<ERROR>:: Invalid SSN.

<ERROR>:: Given element not in the sccp database.

EXAMPLES

ANSI: ***ADD-SUBSYS:SPC***=0-23-255***,SSN***=4***;***
CCITT: ***ADD-SUBSYS:SPC***=3-125-6***,SSN***=4***;***
ANSI: ***DELETE-SUBSYS:SPC***=0-23-255***,SSN***=4***;***
CCITT: ***DELETE-SUBSYS:SPC***=3-125-6***,SSN***=4***;***
ANSI: ***DISPLAY-SUBSYS:SPC***=0-23-255***,SSN***=4***;***
CCITT: ***DISPLAY-SUBSYS:SPC***=3-125-6***,SSN***=4***;***
Both: ***DISPLAY-SUBSYS:SPC***=1-1-1***,SSN***=****;***

SAMPLE OUTPUT

```
-------------------------------------------------------------------
SSN SPC MSSN MSPC SSN_STATUS XLATE  CONCERNED
-------------------------------------------------------------------
253 1-11-1 0 0-0-0 ALLOWED PRIMARY NO
254 1-1-1  0 0-0-0 PROHIBITED  PRIMARY NO
<SUCCESS>:: 1 records found.
```

Table 5-24: SUBSYS Display Values

SSN, SPC	MSSN	MSPC	SSN_STATUS	XLATE	CONCERNED
See description in synopsis	Mate SSN, if any.	Mate point code, if any.	Status of the SSN: ALLOWED PROHIBITED	Translation: PRIMARY SECONDARY	Whether the signaling point is a concerned point code.

5.3.8 Local Subsystem (LOCALSUBSYS)

NAME

LOCALSUBSYS Displays a local subsystem or local subsystem information for a Signaling Point Code (SPC)

COMMANDS

DISPLAY Displays the local subsystems defined for the SPC from the SCCP network database.

DISPLAY-LOCALSUBSYS:;

PARAMETERS

none

EXAMPLES

ANSI: ***DISPLAY-LOCALSUBSYS:;***

CCITT: ***DISPLAY-LOCALSUBSYS:;***

Table 5-25: LOCALSUBSYS Display Values

SSN, SPC	MSSN	MSPC	SSN_STATUS	XLATE	CONCERNED
See description in synopsis	Mate SSN, if any.	Mate point code, if any.	Status of the SSN: UNEQUIPPED, ALLOWED PROHIBITED	Translation: PRIMARY SECONDARY	Whether the signaling point is a concerned point code.

5.3.9 Connection (CONNECTION)

NAME

CONNECTION Displays the state of a connection-oriented SCCP connection.

COMMANDS

DISPLAY Displays the state of a connection-oriented SCCP connection.

DISPLAY-CONNECTION:ID=id***;***

PARAMETERS

id Connection ID ranging from 0 up to 16383, or asterisk (*****) for all. If * entered, then all the connection states but the IDLE ones. are displayed.

ERRORS

<ERROR>:: Missing ID parameter.

<ERROR>:: All connections are in IDLE state.

EXAMPLES

DISPLAY-CONNECTION:ID=126;

DISPLAY-CONNECTION:ID=*;

SAMPLE OUTPUT

```
------------------
ID     STATUS
------------------
126    IDLE
<SUCCESS>:: 1 records found.
```

The connection states can be one of following:

- IDLE
- CONNECTION_PENDING_OUTGOING
- CONNECTION_PENDING_INCOMING
- CONNECTION_PENDING
- WAIT_CONNECTION_CONFIRM
- ACTIVE
- DISCONNECT_PENDING
- DISCONNECT_PENDING_BOTHWAY
- DISCONNECT_PENDING_INCOMING
- DISCONNECT_PENDING_OUTGOING
- MAINTENANCE_BLOCKING

- RESET_OUTGOING
- RESET_INCOMING
- BOTHWAY_RESET
- WAIT_FOR_SENDING_EA_MESSAGE

5.4 System MML Commands

Use the system MML commands to configure system-wide managed objects.

5.4.1 Host (HOST)

NAME

HOST Adds, deletes, displays, or modifies a host instance.

COMMANDS

ADD Adds a new host instance.

ADD-HOST:HOSTNAME=hostname***,RMTHOST***=rmthost ***[,ALIAS***=alias***][,RMTHOSTTYP***=rmthosttyp***][,CONF***=conf***];***

MODIFY Modifies a host instance information.

MODIFY-HOST:HOSTNAME=hostname***,RMTHOST***=rmthost ***[,RMTHOSTTYP***=rmthosttyp***][,CONF***=conf***];***

DELETE Deletes a host instance.

DELETE-HOST:HOSTNAME=hostname***][,RMTHOST***=rmthost***];***

DISPLAY Displays a specific host instance or all instances.

DISPLAY-HOST:[HOSTNAME=hostname***][,RMTHOST***=rmthost***];***

Note: *MML commands to disconnect host B from host A cannot be entered from host A as follows:*
MML_TH> ***MODIFY-HOST: HOSTNAME=host-B,RMTHOST=host-A,CONF=OFF;***

This command fails with the following error string:
<ERROR>:: MODIFY-HOST operation must be performed on local hosts

This means that the same command should have been issued from host-B. Only the connection from the host-A side can be disconnected from host-A, for example:
MML_TH> ***MODIFY-HOST:HOSTNAME=host-A,RMTHOST=host-B,CONF=OFF;***
MML_TH> ***MODIFY-HOST:HOSTNAME=host-A,RMTHOST=host-D,CONF=OFF;***

PARAMETERS

hostname Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

rmthost Name of remote host. It is a string of 1 to 15 alphanumeric characters maximum.

alias Alias name for the remote system, if it is a multi-homed host.

rmthosttyp Remote system host type. It is a character string that accepts the following values:

- AMGR: NewNet Distributed7-type system *(default)*

• OTHER: other than NewNet Distributed7-type system

conf Determines the configuration of the host. It is a character string that accepts the following values:

• ON Establish this connection

• OFF Wait for more configurations

ERRORS

<ERROR>:: MO does not exist

<ERROR>:: Hostname is not defined in the network

<ERROR>:: No such an instance

<ERROR>:: Can not add host in standalone mode

<ERROR>:: Missing HOSTNAME parameter

<ERROR>:: Missing RMTHOST parameter

<ERROR>:: Local hostname can not be used as remote or alias

<ERROR>:: DUALHOST is not configured in NTWK MO

<ERROR>:: Alias host is not in the same network of DUALHOST

<ERROR>:: TCPCON entry does not exist

<ERROR>:: Missing CONF parameter

<ERROR>:: Same CONF value for this entry

<ERROR>:: CONF parameter of the entry is ON

EXAMPLES

ADD-HOST:HOSTNAME=tweety-priv***;RMTHOST***=sylvester-priv***,CONF***=OFF***;***
MODIFY-HOST:HOSTNAME=tweety-priv,***CONF***=ON***;***
DELETE-HOST:HOSTNAME=tweety-priv***;***
DISPLAY-HOST:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-HOST:;
-------------------------------------------------------------------
         HOSTNAME          RMTHOST           ALIAS RMTHOSTTYP CONF
-------------------------------------------------------------------
      tweety-priv   sylvester-priv               -       AMGR   ON
   sylvester-priv      tweety-priv               -       AMGR   ON
<SUCCESS>:: 2 records found
```


5.4.2 Stored Alarm (STRDALM)

NAME

STRDALM Deletes and displays alarms set in the system.

COMMANDS

DELETE Clears an alarm that is currently *set* in the system. Alarms that have occurred which are of type SET_ALARM are tracked by the alarm process until they are cleared. An alarm can be cleared when the CLR_ALARM type alarm that is associated with the alarm occurs or when this command is run with the alarm specified individually or as part of a group. The current list of set alarms can be displayed with the DISPLAY-STRDALM command.

DELETE-STRDALM:GROUP=group***[,HOSTNAME***=hostname***]***
[,MODULE=module***][,TYPE***=type***][,LAST_OCCUR***=last_occur***];***

DISPLAY Displays the current alarms that are *set* in the system. Alarms that have occurred which are of type SET_ALARM are tracked by the alarm process until they are cleared. These alarms are the ones shown in the output of this command. EVENT and CLR_ALARM alarms are never shown in this output. All alarms are displayed to the console as they occur, depending on the display settings.

DISPLAY-STRDALM:GROUP=group***[,HOSTNAME***=hostname***]***
[,MODULE=module***][,TYPE***=type***][,LAST_OCCUR***=last_occur***];***

PARAMETERS

hostname Name of the host whose alarms are to be deleted or displayed. Default is the local host.

group Alarm group name. It is a character string that accepts the following values, or an ***** for all groups:

- DKM - Distributed kernel memory
- NIMOD - connection management
- MTPL1 - MTP Level 1
- MTPL2 - MTP Level 2
- APM - Application Process management
- ETMOD - Ethernet test module
- OMAP - operation, maintenance, and administration part
- SPM - signaling point (SP) management
- TRMOD - translation module
- UPM - user part management, such as MTP Level 3
- SG - SGP lrms

The following have no meaning in Signaling Gateway, although they are acceptable values:

- ISUP - ISUP management
- ISUPMOD - ISUP management - distributed
- SCCP - service connection control part management
- TCAP - TCAP driver
- TCMOD - TCAP over TCP/IP connection management

module Module number (middle two digits) of the alarm or alarms to be deleted or displayed for the specified ***GROUP***.

type Last two digits of the alarm number to be deleted or displayed for the specified ***GROUP*** and ***MODULE***.

last_occur Date and time of the alarm or alarms to be deleted. It can be used with or without the other parameters. The time stamp must be specified in the format: **hh:mm:ss@MM/DD/YY**

- hh (hour) can be from 01 to 24
- mm (minutes) can be from 00 to 59
- ss (seconds) can be from 00 to 59
- MM (month) can be from 01 to 12
- DD (day) must be a valid two-digit number for the given month in the range of 1 to 31
- YY (year) can be from 00 to 99

ERRORS

<ERROR>:: Mgmt, MTP_STATE either RESTARTING or RESTARTED

EXAMPLES

DISPLAY-STRDALM:GROUP=*;***
DISPLAY-STRDALM:GROUP=SPM***,MODULE=***2***,TYPE=***3***;***
DISPLAY-STRDALM:LAST_OCCUR=11:15:32@02/15/97***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-STRDALM:;
--------------------------------------------------------------------------------
HOSTNAME : tweety-p
GROUP    : SG         MODULE : 4   TYPE   : 4   SEVERITY : NONE
PARAMETERS:*0
FIRST_OCC: Wed Oct 17 14:09:22 2001   LAST_OCC: Wed Oct 17 14:09:22 2001
             NUM_OF_OCCUR : 1
ALM_TEXT : SGP: SPMC for network appearance 0 is down

<SUCCESS>:: 1 record found
```


Table 5-26: STRDALM Display Values

SEVERITY	FIRST_OCCUR	LAST_OCCUR	NUM_OF_OCCUR	TEXT
Severity of the alarm: • INFO • MINOR • MAJOR • CRITICAL • FATAL	Date and time that the alarm happened for the first time in hh:mm:ss@ MM/DD/YY format.	Date and time of the last time that an alarm number happened in hh:mm:ss@ MM/DD/YY format.	Number of occurrences of alarm number.	Text associated with the alarm number.

5.4.3 Alarm (ALARM)

NAME

ALARM Displays or modifies the alarm configuration.

COMMANDS

MODIFY Modifies the configuration of the alarm managed object to have new alarm-display characteristics, to be the global alarm process in the distributed system, or to be updated with configuration files. Regardless of display settings, alarms are always logged to a file in the ***$EBSHOME/access/AlarmLogs*** directory.

MODIFY-ALARM:[HOSTNAME=hostname***][,DISPLAY***=display***]***
[,CONS_THRS=cons_thrs***][,USER_THRS***=usr_thrs***]***
[,REPEAT=repeat***][,GLOBAL***=global***][,UPDATE***=update***];***

DISPLAY Displays the current alarm configuration, including the console threshold, external console threshold, current alarm log file name, and current configuration files. It also displays the lower severity thresholds of alarms to be received by console and user for each alarm group.

The output of this command (see Sample Output) displays the current settings for the console output and for the default severity-level threshold settings of alarm output to the external console and the external user. The full path and file names are shown for the alarm group definition file, the alarm configuration file, and the current log file for alarms. The alarm log file holds all the generated alarms and can be used for an extensive examination of the alarms that have occurred in the system.

The output also displays all the alarm group IDs (GR#), in decimal, from the alarm group definition file. The GRP_NAME column specifies the name associated with the group ID. The USR-THR column shows the minimum threshold of alarm severity that will be displayed to the user interface for that group (e.g. all those alarms from that group that are above INFO in severity). The CONS-THR column shows the minimum threshold of alarm severity that will be displayed to the console, if it is enabled.

DISPLAY-ALARM:;

PARAMETERS

hostname Name of the host to be modified, entered as an 15-character alphanumeric label.

display Indicator of display status. Values can be ON, to display alarms to the console, or OFF, to turn the display of alarms off. At start-up, it is set to ON by default. The alarm process must be running in order to display alarms.

cons_thrs Minimum severity level of alarms to be displayed to the console. It is a character string that accepts the following values:

•NONE

• INFO *(default)*

•MINOR

• MAJOR

•CRITICAL

•FTAL

user_thrs Minimum severity level of alarms to be displayed to a user-defined external alarm interface function. It is a character string that accepts the following values:

•NONE

• INFO *(default)*

•MINOR

• MAJOR

•CRITICAL

•FTAL

repeat Counter threshold for displaying a repeated alarm. If an incoming alarm is exactly the same as the immediately preceding alarm, the counter is increased incrementally but the alarm is not displayed unless the counter equals the setting specified in this parameter. It is an integer from 0 to 999. The default is 3, so that if four identical alarms are received, only two are displayed. The counter is reset to 0 when the threshold is reached. The value is ignored if the DISPLAY parameter is OFF.

global Indicator of an attempt to become the global alarm process. It is a character string that accepts the following values:

•ON

•OF

update Indicator to update the system with modified alarm group definition and alarm text files. It is a character string that accepts the following values:

•ON

•OF

ERRORS

<ERROR>::Network is down.

<ERROR>::No such process.

<ERROR>::Illegal address.

<ERROR>::Parameter value is out of range.

<ERROR>::Cannot update configuration.

EXAMPLES

MODIFY-ALARM:DISPLAY=ON***;***
MODIFY-ALARM:DISPLAY=OFF***;***
MODIFY-ALARM:CONS_THRS=MINOR***;***
MODIFY-ALARM:USER_THRS=MINOR***;***
MODIFY-ALARM:REPEAT=5***;***
MODIFY-ALARM:GLOBAL=ON***;***
MODIFY-ALARM:UPDATE=ON***;***

SAMPLE OUTPUT

Alarm output with DISPLAY on and REPEAT at a threshold of 3:

```
ALARM $880703 $00000000 $00000000 LVL: Info
MTP: Link Set LS-ls0 available
  - - - 3 alarms came - - -
   Fri Mar 27 14:31:03 1996
   Last alarm repeated 3 times more
  - - - 3 alarms came - - -
   Fri Mar 27 14:31:42 1996
   Last alarm repeated 3 times more
  - - - 3 alarms came - - -
   Fri Mar 27 14:32:22 1996
   Last alarm repeated 3 times more

MML_TH>DISPLAY-ALARM:;
```

```
---------------------------------------------------------------------------
HOSTNAME      : sylvester        DISPLAY      : ON
CONS_THRS     : INFO             USER_THRS    : INFO          REPEAT  : 3
LOG_FILE_NUM  : AccessAlarms.0   GLOBAL       : OFF           UPDATE  : OFF

HOSTNAME      : tweety           DISPLAY      : ON
CONS_THRS     : INFO             USER_THRS    : INFO          REPEAT  : 3
LOG_FILE_NUM  : AccessAlarms.0   GLOBAL       : OFF           UPDATE  : OFF

<SUCCESS>:: 2 records found
```


5.4.4 Alarm Event (ALMEVENT)

NAME

ALMEVENT Displays alarm events for all hosts or a specified host.

***Note**: Application programs can use the **alm_notify()** function to express interest in a particular set of alarm events that can occur while operating under the Distributed7 (a.k.a. AccessMANAGER) environment, and specify what action should take place if and when one of the pending alarm events occurs on local, a remote, or all hosts.*

COMMANDS

DISPLAY Displays alarm events.

DISPLAY-ALMEVENT:HOSTNAME=hostname***,***
REQ_HOSTNAME=req_hostname***,GROUP***=group***,***
MODULE=module***,TYPE***=type***,THRESHOLD***=threshold***;***

***Important**: Only the MO server alarm daemon is allowed to add and delete alarm events.*

PARAMETERS

hostname Name of the host whose alarm event information is requested. It is a string of 1 to 15 alphanumeric characters maximum, and the * is not allowed. Alarm event information for all hosts is displayed if HOSTNAME and other parameters are not specified.

req_hostname Name of the host from which an application wants to be notified when a specified alarm occurs. It is a string of 1 to 15 alphanumeric characters maximum, or an * to be notified from all hosts.

group ID of the alarm group, or an * for all alarm groups.

module Alarm module ID in alarm group, or an * for all modules.

type Alarm type in alarm group and module, or an * for all types.

threshold Severity of the alarm. If the specified alarm occurs with this severity, then the application is notified. Wild cards are not allowed. It is a character string that accepts the following values, which are in ascending order of severity:

•NONE

• INFO *(default)*

•MINOR

• MAJOR

•CRITICAL

•F TAL

ERRORS

<ERROR>:: Mgmt, MTP_STATE either RESTARTING or RESTARTED

EXAMPLES

DISPLAY-ALMEVENT:;
DISPLAY-ALMEVENT:HOSTNAME=vortex,***REQ_HOSTNAME***=atomic***;***
DISPLAY-ALMEVENT:HOSTNAME=vortex,***REQ_HOSTNAME***=atomic, ***GROUP***=*;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-ALMEVENT:;
---------------------------------------------------------------------------
HOSTNAME   REQ_HOSTNAME   GROUP   MODULE   TYPE   THRESHOLD
---------------------------------------------------------------------------
galaxya-1galaxya-113122MINOR
galaxya-1galaxya-113122CRITICAL
galaxya-1galaxya-113232INFO
galaxya-1galaxya-113232MAJOR
galaxya-1galaxya-113122INFO
galaxya-1galaxya-113232FATAL
galaxya-1galaxya-113122MAJOR
galaxya-1galaxya-113122FATAL
galaxya-1galaxya-113232MINOR
galaxya-1galaxya-113232CRITICAL
<SUCCESS>:: 10 records found
```


5.4.5 Alarm Group (ALMGRP)

NAME

ALMGRP Displays or modifies the alarm group settings.

COMMANDS

MODIFY Modifies the severity thresholds for the display of alarms for individual alarm groups.

MODIFY-ALMGRP:GROUP=group***,[CONS_THRS***=cons_thrs***]***
[,USER_THRS=user_thrs***];***

DISPLAY Displays the threshold settings for an alarm group.

DISPLAY-ALMGRP:GROUP=group***;***

PARAMETERS

group Alarm group name that specifies which group's alarms to display, with the minimum severity level required for display controlled with the CONS_THRS and the USER_THRS parameters. It is a character string that accepts the following values, or an ***** to display all groups:

- DKM - Distributed kernel memory
- NIMOD - connection management
- OMAP - operation, maintenance, and administration part
- SPM - signaling point (SP) management
- TRMOD - translation module
- UPM - user part management, such as MTP Level 3
- SG - SGP lrms

The following have no meaning in Signaling Gateway, although they are acceptable values:

- ISUP - ISUP management
- ISUPMOD - ISUP management - distributed
- SCCP - service connection control part management
- TCAP - TCAP driver
- TCMOD - TCAP over TCP/IP connection management

cons_thrs Minimum severity level of alarms to be displayed to the console. It is a character string that accepts the following values, which are in ascending order:

- NONE
- INFO *(default)*
- MINOR
- MAJOR
- CRITICAL
- FTAL

user_thrs Minimum severity level of alarms to be displayed to a user-defined external alarm interface function. It is a character string that accepts the following values, which are in ascending order:

- •NONE
- •INFO *(default)*
- •MINOR
- • MAJOR
- •CRITICAL
- •FTAL

Note*: Alarms with a lower severity for a specific group in this command are* ***NOT*** *sent if the CONS_THRS or USER_THRS settings in MODIFY-ALARM are higher.*

ERRORS

<ERROR>::Network is down.

<ERROR>::No such process.

<ERROR>::Illegal address.

<ERROR>::Parameter value is out of range.

<ERROR>::Cannot update configuration.

EXAMPLES

MODIFY-ALMGRP:GROUP=SPM,CONS_THRS=MAJOR;
MODIFY-ALMGRP:GROUP=SPM,USER_THRS=MAJOR;
DISPLAY-ALMGRP:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-ALMGRP:;

----------------------------------------------------------------
          HOSTNAME         GROUP CONS_THRS USER_THRS NUM_OF_ALMS
----------------------------------------------------------------
    sylvester-priv           UPM      NONE      NONE           0
    sylvester-priv           DKM      NONE      NONE           0
    sylvester-priv         NIMOD      NONE      NONE           0
    sylvester-priv           SPM      NONE      NONE           3
         sylvester         TRMOD      NONE      NONE           0
         sylvester         ETMOD      NONE      NONE           0
         sylvester           DKM      NONE      NONE           0
         sylvester            SG      NONE      NONE           0
         sylvester           UPM      NONE      NONE           0
    sylvester-priv         TRMOD      NONE      NONE           0
```


sylvester	NIMOD	NONE	NONE	0
sylvester-priv	ETMOD	NONE	NONE	0
sylvester	SPM	NONE	NONE	0
sylvester-priv	SG	NONE	NONE	1
tweety-priv	UPM	NONE	NONE	0
tweety-priv	SPM	NONE	NONE	0
tweety-priv	NIMOD	NONE	NONE	0
tweety	SPM	NONE	NONE	0
tweety	NIMOD	NONE	NONE	0
tweety-priv	DKM	NONE	NONE	0
tweety-priv	TRMOD	NONE	NONE	0
tweety	TRMOD	NONE	NONE	0
tweety-priv	ETMOD	NONE	NONE	0
tweety	ETMOD	NONE	NONE	0
tweety	DKM	NONE	NONE	0
tweety	SG	NONE	NONE	0
tweety	UPM	NONE	NONE	0
tweety-priv	SG	NONE	NONE	0

<SUCCESS>:: 28 records found

5.4.6 Configuration (MMLCONF)

NAME

MMLCONF Displays or modifies the MML session configuration values.

COMMANDS

MODIFY Configures the MML process with settings for command logging and response timeout. The configuration values are stored and maintained as the settings for MML on a particular signaling point even after the current session ends. Subsequent MML sessions, ***on the same signaling point***, use the same configuration until it is modified. MML sessions for different signaling points use their own configurations (e.g. *mml 0* and *mml 1* may have different configurations).

MODIFY-MMLCONF:[LOG=log***][,TIMEOUT=***timeout***];***

DISPLAY Displays the configuration values for the MML session. All settings can be displayed by just entering the command without parameters. Individual settings are displayed by specifying a parameter name.

DISPLAY-MMLCONF:[LOG=log***][,TIMEOUT=***timeout***];***

PARAMETERS

log State of MML command logging, ON or OFF. Default is ON.

While the LOG option is on, MML logs all commands that are issued, except DISPLAYs, into the file, ***$EBSHOME/access/RUN***<sp#>***/backup/MMLcmnds.current***. The user name, user ID, and time of execution are included with the command in the log that is written to this file. The LOG option can be turned off and on at any time with this command.

timeout Value of timeout for communications with processes, between 0 and 240000 milliseconds. The default is 15000 milliseconds.

When MML sends a message to the daemon processes, such as, ***upmd***, ***isupd***, etc., it waits for acknowledgments until the TIMEOUT setting has expired. If an acknowledgment is not received, MML displays an SPM error message.

EXAMPLES

MODIFY-MMLCONF:;

MODIFY-MMLCONF:LOG;

MODIFY-MMLCONF:LOG=OFF;

MODIFY-MMLCONF:TIMEOUT=3000;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-MMLCONF:;
```



```
------------------------------
  CONFNAME TIMEOUT LOG
------------------------------
      CFG0   15000  ON
<SUCCESS>:: 1 record found
```

5.4.7 Network (NTWK)

NAME

NTWK — Displays or modifies the operation mode of hosts in the distributed network.

COMMANDS

DISPLAY — Displays the operation mode of hosts in the distributed network.

DISPLAY-NTWK:[HOSTNAME=*hostname****];***

MODIFY — Configures the Signaling Gateway system as stand-alone or part of a distributed network.

MODIFY-NTWK:HOSTNAME=hostname***[,MODE=***mode***]***
[,CLOCKSYNC=clocksync***][,FREQUENCY=***frequency***]***
[,DUALHOST=dualhost***][,NETMASK1=***netmask1***]***
[,NETMASK2=netmask2***];***

PARAMETERS

hostname — Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

mode — Specifies how the host operates in the network. It is a character string that accepts the following values:

- STNDLN stand alone mode
- DSTRBTDdistributed mode *(default)*

clocksync — Specifies whether or not the ability to synchronize the network clock is available. It is a character string that accepts the following values:

- ON *(default)*
- OF

frequency — Specifies, in milliseconds, how often to check the system clock on all hosts if CLOCKSYNC is **ON**. It is an integer from 0 to 10000. The default value is 0 if running in the stand alone mode, or 1000 in the distributed mode. THe range for the distributed mode is 60 to 10000.

dualhost — Specifies the alternate host name, if any, of the local host on a secondary host in a dual-LAN network. If dual-LAN is not in use, then this field must contain the local host name specified in the HOSTNAME parameter. It is a string of 1 to 15 alphanumeric characters maximum.

netmask1 — Specifies the 32-bit mask, in hex format, that is used to extract the network ID on the primary network. The following default values are initialized on the basis of class type associated with the corresponding network:

- 7f000000 *(Class A default)*
- 3fff0000 *(Class B default)*
- 1fffff00 *(Class C default)*

netmask2	Specifies the 32-bit mask, in hex format, that is used to extract the network ID on the secondary network, if any. The following default values are initialized on the basis of class type associated with the corresponding network: • 7f000000 *(Class A default)* • 3fff0000 *(Class B default)* • 1fffff00 *(Class C default)*

ERRORS

<ERROR>:: MO does not exist

<ERROR>:: Hostname is not defined in the network

<ERROR>:: No such an instance

<ERROR>:: NTWK MO cannot be modified - HOST entries exist

<ERROR>:: Product is not configured as distributed

EXAMPLES

MODIFY-NTWK:HOSTANME=vortex***,MODE***=STNDLN***;***
DISPLAY-NTWK:HOSTNAME=vortex***;***
DISPLAY-NTWK:;

SAMPLE OUTPUT

```
MML_TH> display-ntwk:;
-------------------------------------------------------------------------
HOSTNAMEMODECLOCKSYNCFREQUENCYDUALHOSTNETMASK1NETMASK2
-------------------------------------------------------------------------
sylvester-pDSTRBTDON1000sylvester-p7f0000007f000000
tweety-pDSTRBTDON1000tweety-p7f0000007f000000
<SUCCESS>:: 2 records found
```

5.4.8 TCP/IP Connections (TCPCON)

NAME

TCPCON Displays or modifies the TCP/IP connection information.

COMMANDS

MODIFY Configures TCP/IP connections.

MODIFY-TCPCON:HOSTNAME=hostname***, RMTHOST***=rmthost
[,MODE=mode***][,SERVICE***=service***][,PROTO***=proto***]***
[,MODULES=modules***][,HBEAT***=hbeat***][,FREQU***=frequ***]***
[,MAXTRIES=maxtries***][,ACT_EST***=act_est***]***
[,ACT_RMV=act_rmv***][,HB_LOSS***=hb_loss***];***

DISPLAY Displays information on TCP/IP connections.

DISPLAY-TCPCON: [HOSTNAME=hostname***]***
[,RMTHOST=rmthost***];***

PARAMETERS

hostname Name of host. It is a string of 1 to 15 alphanumeric characters maximum.

rmthost Name of remote host. It is a string of 1 to 15 alphanumeric characters maximum.

mode TCP/IP connection type. It is a character string that accepts the following values:

- AUTO The system sets the mode *(default)*
- MASTER The local host always tries to establish the connection
- SLAVE The local host always waits for connection requests.

service Internet service name. The only value accepted is **NETDBASE**, which is the default.

proto Identifies transportation layer protocol. The only value accepted is **TCP**, which is the default.

modules Specifies the ordered list of STREAMS modules to push over this TCP/IP connection. The only value accepted is ***nimod***, which is the default.

hbeat The heartbeat mechanism verifies the TCP/IP connection with the active host. It is a character string that accepts the following values:

- ON Activate heartbeat
- OFF Deactivate heartbeat

frequ Specifies, in milliseconds, how often to check the TCP/IP connections if HBEAT is **ON**. It is integer from 0 to 1000. The default value is 0 if the HBEAT parameter is OFF, and 1000 if the HBEAT parameter is ON.

maxtries Specifies the maximum number of consecutive times to try to establish the specified TCP/IP connection. It is integer from -1 to 5000. The default value, **-1**, specifies that there is no maximum limit, and to keep trying to establish a connection until successful.

act_est Specifies the action to take when the connection is established. It is a character string that accepts the following values:

- IGNORE
- INFORM *(default)*

act_rmv Specifies the action to take when the connection is broken. It is a character string that accepts the following values:

- IGNORE
- INFORM *(default)*

hb_loss Specifies the action to take when a remote host fails to respond to a heartbeat request. It is a character string that accepts the following values:

- NOACTION
- SYNCDATA *(default)*

ERRORS

<ERROR>:: MO does not exist

<ERROR>:: Hostname is not defined in the network

<ERROR>:: No such an instance

<ERROR>:: Missing HOSTNAME parameter

<ERROR>:: Missing RMTHOST parameter

<ERROR>:: TCPCON entry does not exist

<ERROR>:: CONF parameter of the entry is ON

EXAMPLES

MODIFY-TCPCON:HOSTNAME=tweety-priv***,*** ***RMTHOST***=sylvester-priv, ***MODE***=auto***,HBEAT***=ON***,ACT_EST***=IGNORE***,ACT_RMV***=IGNORE***,*** ***HB_LOSS***=SYNCDATA***;***
DISPLAY-TCPCON:;

SAMPLE OUTPUT

MML_TH>DISPLAY-TCPCON:;

```
----------------------------------------------------------------------------------
HOSTNAMERMTHOSTMODESERVICEPROTOMODULESHBEATFREQUMAXTRIES
----------------------------------------------------------------------------------
tweety-privsylvester-privAUTONETDBASETCPnimodON1000-1
sylvester-privtweety-privAUTONETDBASETCPnimodON1000-1
       ----------------------------------------------
       ACT_ESTACT_RMVHB_LOSSSTATE
       ----------------------------------------------
       IGNOREIGNORESYNCDATAESTBLSHD
       IGNOREIGNORESYNCDATAESTBLSHD
       <SUCCESS>:: 2 records found
```

5.4.9 Exit (EXIT)

NAME

EXIT Ends the MML session and returns to the Unix prompt.

COMMANDS

EXIT Exits the MML session.

EXIT:;

ERRORS

None

EXAMPLES

EXIT:;

5.4.10 Help (HELP)

NAME

HELP — Displays a list of commands for which Help is available by typing the command in the HELP command line prompt. A command can also be entered with the HELP command to display help about a specific command.

COMMANDS

HELP — Switches to the help mode and displays names of commands or a man page for the specific command entered.

HELP:;

HELP:CMD=*command_name****;***

PARAMETERS

command_name The name of the command about which to display the MML help information.

ERRORS

None

EXAMPLES

HELP:;
HELP:CMD=ADD-LINK***;***

5.4.11 Log (SET-LOG)

NAME

SET-LOG — Turns the generation of log messages for a specified process ON or OFF. After enabling log generation, each message into or out of the named object or SS7 object is duplicated and saved to the LOG process. Logging is disabled by entering the command with OFF in the LOG parameter.

COMMANDS

SET-LOG — Configures the generation of log messages.

SET-LOG:TO=to***,LOG***=log***[,NAME***=name***][,SPID***=spid***][,UPID***=upid***]***
[,SSN=ssn***][,INST***=inst***];***

PARAMETERS

to — Type of process. It is a character string that accepts the following values:

- NMDOBJ
- SS7OBJ

log — Turns the logging function on and off. It is a character string that accepts the following values:

- ON
- OF *(default)*

name — The name of any registered named object. This parameter is required when the TO parameter is **NMDOBJ**. It is a string of 1 to 14 characters maximum.

spid — The Signaling Point identification number. It is an integer from 0 to 7 that is required when the TO parameter is **SS7OBJ**.

upid — The User Part identification number is an integer value as defined in SS7 protocol (0 for MTP, 3 for SCCP, 5 for ISUP). This parameter is required to log messages to any SS7 object.

ssn — Subsystem number of the SS7 object or an * for all SSNs.

inst — The instance number of the subsystem number of the SS7 object.

ERRORS

<ERROR>::Network is down.

<ERROR>::No such process.

<ERROR>::Illegal address.

<ERROR>::Parameter value is out of range.

<ERROR>::MML syntax error.

EXAMPLES

SET-LOG:TO=NMDOBJ***,NAME***=XYZ***,LOG***=ON***;***
SET-LOG:TO=SS7OBJ***,SP***=0***,UP***=3***,SSN***=254***,INST***=2***,LOG***=ON***;***

5.5 Signaling Gateway Managed Objects

5.5.1 Signalling Point to Network Appearance Mapping (SGSPNA)

NAME

SGSPNA — Defines a mapping between a Network Appearance (NA) and Signaling Point (SP).

COMMANDS

ADD — Adds a mapping between an NA and an SP.

ADD-SGSPNA:SPID=spid***,NAID***=naid***;***

MODIFY — Modifies the NA and SP mapping or status of a particular SP.

MODIFY-SGSPNA:SPID=spid***[,NAID***=naid***][,OPERST***=operst***]***
[,SGPID=spgid***];***

Note*: The SGSPNA's status must be INACT (inactive) before modifying the NAID.*

DELETE — Deletes an NA to SP mapping.

DELETE-SGSPNA:SPID=spid***;***

Note*: The SGSPNA's status must be INACT (inactive) before deleting the SGSPNA instance. Also, there must not be any AS configured for the associated SP.*

DISPLAY — Displays an NA to SP mapping.

DISPLAY-SGSPNA:[SPID=spid***][,SGPID***=sgpid***];***

PARAMETERS

spid — The Signaling Point identification number to which an NA ID maps. It is an integer from 0 to 7.

naid — The NA identification number. Valid values are 0 to 0x7fffffff. Each SP must have a unique NAID.

operst — The operational state of the SP. It is a character string that accepts the following values:

- ACT - activate an SP, so that if MTP has been configured as an STP, SGP registers to MTP as a gateway process for the specified SP.
- INACT - inactivate an SP, by deregistering SGP from MTP for the specified SP.

sgpid The SGP ID to which the OPERST is applied, if it is specified. The OPERST is applied to all SGPs if this optional parameter is not included in the command.

ERRORS

Database operation failed

Configuration limit exceeded

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Failed to establish communication with UPMD

SP is active

SP is inactive

MTP not ready or not configured

Failed to query MTP (check upmd status)

Protocol mismatch between SPs with same NA

NA is used in other SP

AS exists for SP %d

SP activation in progress, please wait

EXAMPLES

MODIFY-SGSPNA:SPID=0***,NAID***=0***;***
DISPLAY-SGSPNA:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SGSPNA:;
-----------------------------------------------
 SPID        NAID  OPERST STATUS          SGPID
-----------------------------------------------
    0           0      ACT    ACT     sylvester
    0           0      ACT    ACT          chip
    0           0      ACT    ACT          dale
    0           0      ACT    ACT        tweety
```



```
<SUCCESS>:: 4 records found
```

Table 5-27: SG Display Values

STATUS
This read-only parameter is the registration status of the SP, which can be one of the following values: • ACT - SGP is registered to MTP • INACT - SGP is not registered to MTP

5.5.2 Signaling Gateway Process (SGSGP)

NAME

SGSGP — Configures a Signaling Gateway Process (SGP). By default, an instance of SGSGP is automatically created for each SGP.

COMMANDS

MODIFY — Modifies the configuration of an SGP.

MODIFY-SGSGP:SGPID=sgpid*[**,IP1***=ip1*]**[,IP2***=ip2***][,HOSTNAME***=hostname***] [,SCTPPORT***=sctpport***];***

DISPLAY — Displays the configuration of an SGP.

DISPLAY-SGSGP:[SGPID=sgpid***];***

PARAMETERS

sgpid — SGP ID. It is a string of 15 characters, and is pre-assigned with the hostname of the machine where SGP is running.

ip1 — Primary IP address of SGP used for communication over SCTP. Address is entered in dot notation format. A '-' can be entered to de-assign a previously assigned address.

ip2 — Secondary IP address of SGP used for communication over SCTP. Address is entered in dot notation format. A '-' can be entered to de-assign a previously assigned address.

hostname — Domain name of the system where the local SGP runs. This field can be used in place of the IP addresses if Internet Domain Name System (DNS) is available. Input is a string of 128 characters. A '-' can be entered to de-assign a previously assigned address.

sctpport — Local port number to which SCTP is bound. It is an integer from 1000 to 99999. The IETF SIGTRAN M3UA recommends that port 2905 be used for listening for new connections, and this is the default value. However, each SGP must be assigned a *unique* port number if there are other SCTP clients running on the same host.

***Note**: The IP address and hostname fields are mutually exclusive, i.e. only either IP addresses OR hostname can be specified at the same time, but not both.*

ERRORS

Database operation failed

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Invalid SGP ID

Invalid IP address

IP addresses cannot be identical

Error configuring M3UA stack

EXAMPLES

MODIFY-SGSGP:SGPID=hostA***,IP1=***155.226.100.100***;***
DISPLAY-SGSGP:;
DISPLAY-SGSGP:SGPID=hostA***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgsgp:;
---------------------------------------------------------------------------
SGPID: sylvester
HOSTNAME:
IP1: 155.226.145.142      IP2:                     SCTPPORT: 2905
---------------------------------------------------------------------------
SGPID: tweety
HOSTNAME:
IP1: 155.226.145.143      IP2:                     SCTPPORT: 2905

<SUCCESS>:: 1 record found
```

5.5.3 Application Server (SGAS)

NAME

SGAS Configures a remote Application Server (AS) for a specific Signaling Point (SP).

COMMANDS

ADD Adds a remote AS configuration for a particular Signalling Point.

ADD_SGAS:ASID=asid,SPID=spid,RKID=rkid,RCID=rcid,ASPLIST =asplist [, MODE=mode] [,MINASP=minasp][PDTIMER=pdtimer];

MODIFY Modifies the configuration of an AS.

MODIFY-SGAS:ASID=asid*[,**SPID***=spid*]**[,RKID***=rkid***][,RCID***=rcid***] [,MODE***=mode***][,ASPLIST***=asplist***] [,PDTIMER***=pdtimer***];***

DELETE Deletes an AS configuration.

DELETE-SGAS:ASID=asid***;***

***Note**: An AS must have no registered ASP actively processing traffic BEFORE an AS can be deleted.*

DISPLAY Displays the configuration of one or more AS(s).

DISPLAY-SGAS:[ASID=asid***];***

PARAMETERS

asid The identification of the Application Server (AS). It is a string of up to 15 characters.

spid The Signaling Point identification. It is an unsigned integer from 0 to 7. An SP can serve up to 256 ASs.

rkid The identification of the route key. The route key must be created before it can be associated with an AS. It is a string of up to 15 characters.

rcid The Route Context associated with the route key. It is a unique value assigned to represent the route key provisioned on both the ASP and SG sides. Both the ASP and the SG ***must*** use exactly the same route context that is associated with a particular route key used by an AS. This is required to route messages correctly.

mode Traffic mode in which the ASPs for this AS are operating. It is a character string that accepts the following values:

- LOADSHARE - ASPs are operating in a load sharing mode and the traffic is distributed to all active ASPs.
- OVERRIDE - ASPs are operating in an override mode, in which there is only one, primary ASP that is actively processing traffic, and the other ASPs are in standby mode.

	Default mode is set to LOADSHARE.
minasp	The minimum number of ASPs in this AS that must be active to handle load and keep the AS active. This parameter is required only when the MODE parameter is LOADSHARE. It is an integer from 0 to 5.
asplist	List of ASP IDs for the ASPs that belong to this AS. A maximum of five ASPs can be specified. It is a string of up to 40 characters maximum. The format of input is a list of ASP IDs, each separated by a slash, /, as the delimiter. For example, 1/4/5/6/7, or 2/3. Default minimum asp value is set to 1.
pdtimer	The pending timer is the amount of time, in seconds, to wait for an ASP to become active before deleting any messages in the pending queue. It is an integer from 0 to 120. The default is 5 seconds.

ERRORS

Licensed number of AS reached

Configuration limit exceeded

Database operation failed

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Error configuring M3UA stack

AS does not exist

SP is used by other AS

NA for the specified SP does not exist

Route key is used by other AS

AS is active

Maximum number of ASPs exceeded

Insufficient number of ASPs

ASP exists in AS

ASP does not exist

ASP is connected

EXAMPLES

ADD-SGAS:ASID=0,SPID=0,RKID=0,RCID=0,ASPLIST=0/1;
ADD-SGAS:ASID=0,SPID=0,RKID=0,RCID=0,ASPLIST=0/1,MODE=OVER-RIDE;
ADD-SGAS:ASID=1***,SPID=1,RKID=***1,***MODE=***LOADSHARE***,MINASP=***1***, ASPLIST=***0/1/2***;***
MODIFY-SGAS:ASID=1***,SPID=1,RKID=***1,***MODE=***OVERRIDE***;***

MODIFY-SGAS:ASID=1***,SPID=1,ASPLIST***=0/1***;***
DELETE-SGAS:ASID=2***;***
DISPLAY-SGAS:ASID=1***;***
DISPLAY-SGAS:;

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SGAS:;
--------------------------------------------------------------------------
ASIDSPIDRKIDRCIDMODEMINASPASPLISTPDTIMERSTATUSSGPID
--------------------------------------------------------------------------
as10 r1 1LOADSHARE1vortex5ACTchip
as10 r1 1LOADSHARE1vortex5ACTdale
as10 r1 1LOADSHARE1vortex5INACTsylvester
as10 r1 1LOADSHARE1vortex5INACTtweety
<SUCCESS>:: 4 records found
```

Table 5-28: SGAS Display Values

STATUS	SGPID
This is the status of the global traffic on the AS, which can be one of the following values: • INACT - inactive • ACT - active • PEND - pending • INSUFRES - insufficient number of active ASPs	The SGP identification.

5.5.4 Application Server Process (SGASP)

NAME

SGASP Configures a remote Application Server Process (ASP).

COMMANDS

ADD Adds a remote ASP.

ADD-SGASP:ASPID=aspid***[,IP1***=ip1***][,IP2***=ip2***][,HOSTNAME***=hostname***] [,SCTPPORT***=sctpport***][,NWASPID***=nwaspid***];***

MODIFY Modifies the configuration of an ASP.

MODIFY-SGASP:ASPID=aspid***[,IP1***=ip1***][,IP2***=ip2***][,HOSTNAME***=hostname***] [,SCTPPORT***=sctpport***][,NWASPID***=nwaspid***] [,OPERST***=operst***][,SGPID***=sgpid***];***

DELETE Deletes an existing ASP.

DELETE-SGASP:ASPID=aspid***;***

DISPLAY Displays the configuration of one or more ASPs.

DISPLAY-SGASP:[ASPID=aspid***][,SGPID***=sgpid***];***

PARAMETERS

aspid The identification of the Application Server Process (ASP). It is a string of up to 15 characters.

ip1 The main IP address used by the ASP to communicate with the SG. It is a string of up to 15 characters that is entered in dot notation format.

ip2 The redundant IP address used by the ASP to communicate with the SG. It is a string of up to 15 characters that is entered in dot notation format.

hostname Domain name of the system where the remote ASP runs. This field can be used in place of the IP addresses if Internet Domain Name System (DNS) is available. Input is a string of 128 characters.

sctpport The remote port number to which SCTP is bound. It is an integer from 1000 to 99999. Default port is set to 2905.

nwaspid Network ASP ID. It is a unique value that identifies an ASP within an AS. Valid values are 0 to 0x7fffffff. 0x7fffffff indicates that it is not used and is the default value if the NWASPID is not specified.

operst The operation performed on a remote ASP from the specified SGP. It is a character string that accepts the following values:

- DISCONN - disconnect the SCTP association between an SGP and an ASP. *(default)*

- LOCK - lock an ASP. An ASP can be locked only when it is not in UP state. Once locked, it cannot transition to UP state until it is unlocked.
- UNLOCK - unlock an ASP so that it can transition to UP state.

sgpid SGP identification. This parameter is used only with the MODIFY and DISPLAY commands. It is a string of 1 to 15 characters maximum.

- MODIFY - This field is applicable only when OPERST is specfied. It indicates the SGP in which OPERST is performed. If not specified, OPERST applies to all SGPs.
- DISPLAY - This displays the status of the remote ASP as viewed from the specified SGP.

***Note**: The IP address and hostname fields are mutually exclusive, i.e. only either IP addresses OR hostname can be specified at the same time, but not both.*

ERRORS

Database operation failed

Licensed number of ASP reached

Configuration limit exceeded

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Invalid IP address

IP addresses cannot be identical

Invalid operation state request

Error configuring M3UA stack

Connection is already down

Connection exists or is being established

ASP is down

EXAMPLES

ADD-SGASP:ASPID=0,IP1=155.226.147.226;
MODIFY-SGASP:ASPID=0***,OPERST***=LOCKED***;***
DELETE-SGASP:ASPID=0***;***
DISPLAY-SGASP:ASPID=0***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgasp:;
----------------------------------------------------------------------
ASPID: chip
HOSTNAME:
```



```
IP1: 155.226.145.163      IP2: SCTPPORT: 2905 ASSOCID: 0 NWASPID: 1 STATUS: DISCONN SGPID: sylvester

ASPID: dale
HOSTNAME:
IP1: 155.226.145.164      IP2: SCTPPORT: 2905 ASSOCID: 0 NWASPID: 2 STATUS: DISCONN SGPID: sylvester

---------------------------------------------------------------------------------
ASPID: chip
HOSTNAME:
IP1: 155.226.145.163      IP2: SCTPPORT: 2905 ASSOCID: 0 NWASPID: 1 STATUS: DISCONN SGPID: tweety

ASPID: dale
HOSTNAME:
IP1: 155.226.145.164      IP2: SCTPPORT: 2905 ASSOCID: 0 NWASPID: 2 STATUS: DISCONN SGPID: tweety

<SUCCESS>:: 2 records found
```

Table 5-29: SGASP Display Values

STATUS
ASP status in relation to a remote SGP: • DISCONN - ASP is disconnected • CONN - ASP is connected (SCTP association is established) • UP - ASP is connected, up, and available. • CONG - ASP is congested • CONN-LOCK - ASP is connected and locked • DISCONN-LOCK - ASP is disconnected and locked

5.5.5 Route Key (SGRK)

NAME

SGRK — Defines the route key associated with an AS that is served by the SG. A route key is a combination of the basic SS7 routing elements. The SG uses RK to route messages to the ASs.

COMMANDS

ADD — Adds a route key.

ADD-SGRK:RKID=rkid***,TYPE***=type***,DPC***=dpc***;***

MODIFY — Modifies the parameters of a route key.

MODIFY-SGRK:RKID=rkid***[,DPC***=dpc***];***

DELETE — Deletes a route key. Only route keys that are not associated with any AS can be deleted.

DELETE-SGRK:RKID=rkid***;***

DISPLAY — Displays the parameters of a route key.

DISPLAY-SGRK:[RKID=rkid***];***

PARAMETERS

rkid — The identification of a route key. It is a string of up to 15 characters.

type — Route key combination type. Every type except DPC requires the creation of an associated SGRKRNG range parameters that the type uses. It is a character string that accepts the following values:

- DPC - route key that is only a DPC
- DPC_OPC - combination of DPC and OPC
- DPC_OPC_CIC - combination of DPC, OPC and CIC range
- DPC_OPC_SIO - combination of DPC, OPC and SIO ranges
- DPC_SIO - combination of DPC and SIO
- DPC_CIC_SIO - combination of DPC, SIO and CIC range
- DPC_SSN - combination of DPC and SSN (connectionless SCCP messages only)

Adding a DPC-SSN route key requires that another route key of the same type, with an SSN equal to 1, be added so that management messages can be routed to the management subsystem. For example, the following route keys must be added to route traffic to DPC 2-2-2, subsystem 250:

add-sgrk:rkid=r1,type=DPC_SSN,dpc=2-2-2;
add-sgrkrng:rkid=r1,ssn=250;
add-sgrk:rkid=ssn1,type=DPC_SSN,dpc=2-2-2;
add-sgrkrng:rkid=ssn1,ssn=1;

The route key with SSN 1 must be associated with another AS that is of the same SP number, for example:

add-sgas:asid=as1,rkid=r1,rcid=1,...;
add-sgas:asid=asssn1,rkid=ssn1,rcid=1,...;

Note*: The DPC-SSN type of Routing Key applies only to connectionless SCCP messages.*

dpc Destination Point Code of the remote AS. It is entered in the x-y-z format. See page 5-2 for more information about this point code format.

ERRORS

Database operation failed
Configuration limit exceeded
Record already exists
Record does not exist
Invalid key
Missing key
Missing mandatory parameters

EXAMPLES

ADD-SGRK:RKID=0***,TYPE***=DPC***,DPC***=2-2-2***,;***
ADD-SGRK:RKID=2***,TYPE***=DPC_OPC_CIC***,DPC***=2-2-2***;***
MODIFY-SGRK:RKID=1***,TYPE***=DPC_OPC***;***
DELETE-SGRK:RKID=4***;***
DISPLAY-SGRK:RKID=0***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SGRK:;
-------------------------------------------
             RKID          TYPE          DPC
-------------------------------------------
               r1           DPC     11-11-11
               xx   DPC_OPC_CIC        9-9-1
               yy       DPC_SIO        8-8-1
<SUCCESS>:: 3 records found
```

5.5.6 Route Key Range (SGRKRNG)

NAME

SGRKRNG Defines a range that is associated with an existing route key (Route Key (SGRK) on page 5-146) that has any TYPE except DPC. One range at a time can be added or deleted. The SGRK entry must be created ***before*** an SGRKRNG can be added.

COMMANDS

ADD Adds a range to an existing route key.

ADD-SGRKRNG:RKID=rkid***[,SI***=si***][,OPC***=opc***][,CICMIN***=cicmin***]*** ***[,CICMAX***=cicmax***][,SSN***=ssn***];***

DELETE Deletes a route key. Only route keys that are not associated with any AS can be deleted.

DELETE-SGRKRNG:RKID=rkid***[,SI***=si***][,OPC***=opc***]*** ***[,CICMIN***=cicmin***][,CICMAX***=cicmax***][,SSN***=ssn***];***

DISPLAY Displays the parameters of a route key.

DISPLAY-SGRKRNG:[RKID=rkid***];***

PARAMETERS

rkid The identification of an existing route key. It is a string of up to 15 characters.

si Service Indicator. It is a character string that accepts the following values:

•SCCP

•SUP

•TUP

opc Origination Point Code of the originating SS7 node that is connected to the SG. It is entered in the x-y-z format. See page 5-2 for more information about this point code format.

cicmin The minimum number in a CIC range. It is a string of 0 to 65535 characters maximum.

cicmax The maximum number in a CIC range. It is a string of 0 to 65535 characters maximum.

ssn Subsystem Number. It is an integer from 2 to 255.

ERRORS

Database operation failed

Configuration limit exceeded

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Route key range already exists

Route key range does not exist

Route key does not exist

CIC range overlaps with other ranges in this RK

CIC range error (CICMAX must be >= CICMIN)

RKRNG not applicable for RK of DPC type

EXAMPLES

ADD-SGRKRNG:RKID=2***,OPC***=1-1-1***,CICMIN***=1000,CICMAX=3000***;***
DELETE-SGRKRNG:RKID=4***;***
DISPLAY-SGRKRNG:RKID=0***;***

SAMPLE OUTPUT

```
MML_TH>DISPLAY-SGRKRNG:;
-----------------------------------------------------------------
             RKID    SI          OPC      CICMIN      CICMAX SSN
-----------------------------------------------------------------
               xx     -        9-9-9        1000        2000   -
               yy  ISUP                        -           -   -
<SUCCESS>:: 2 records found
```

5.5.7 AS Traffic Status (SGASTFC)

NAME

SGASTFC Modifies and displays the traffic status of a remote AS.

COMMANDS

MODIFY Modifies the traffic status of a remote AS.

DISPLAY-SGASTFC:ASPID=aspid***,ASID***=asid***,OPERST***=operst ***[,SPGID***=spgid***];***

DISPLAY Displays the traffic status of a remote AS.

DISPLAY-SGASTFC:[ASPID=aspid***][,ASID***=asid***][,SPGID***=spgid***];***

PARAMETERS

aspid The identification of the Application Server Process (ASP). It is a string of up to 15 alphanumeric characters.

asid The identification of the Application Server (AS). It is a string of up to 15 alphanumeric characters.

operst Traffic control operation performed for the AS. It is a character string that accepts the following values:

- INACT - inactivate traffic processing for the remote AS on the association between the SGP and ASP.
- ACT - activate traffic processing for the remote AS on the assocation between the SGP and ASP.

spgid SGP identification. It is a string of 1 to 15 characters maximum. This field is applicable only when OPERST is specfied. It indicates the SGP in which the OPERST is performed. If not specified, OPERST applies to all SGPs.

ERRORS

Invalid key

Missing key

Missing mandatory parameters

AS traffic not defined

EXAMPLES

MODIFY-SGASTFC:ASPID=asp1***,ASID***=asp1,***OPERST=act,SGPID***=vortex***;***
DISPLAY-SGASTFC:;
DISPLAY-SGASTFC:SGPID=vortex***;***
DISPLAY-SGASTFC:ASID=as1,***SGPID***=vortex***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgastfc:;
-----------------------------------------------------------
ASID             SGPID            ASPID STATUS
-----------------------------------------------------------
as2           tweety          chip  INACT
as2           tweety             dale  INACT
as1           tweety             chip  INACT
as1           tweety             dale  INACT
<SUCCESS>:: 4 records found
```

Table 5-30: SGASTFC Display Values

STATUS
The current traffic status of the AS, which is one of the following values: • DOWN • INACT • ACT • PEND • INSUF_RES • ALT_ASP_ACT

5.5.8 Signaling Point Management Cluster Status (SGSPMC)

NAME

SGSPMC The status of the Signaling Point Management Cluster (SPMC), including the overall point code status and its user-part statuses.

COMMANDS

DISPLAY Displays the status of the SPMC

DISPLAY-SGSPMC:[SGPID=spgid***][,DPC***=dpc***];***

PARAMETERS

sgpid The SGP identification. It is a string of 1 to 15 characters maximum, and has the hostname of the machine that is running the SGP as the default value. This parameter is used to view the SPMC statuses as seen from the individual SGPs.

dpc Point Code representing the SPMC. It is a string of 11 characters in the format of x-y-z, where x is the network ID (zone), y is the cluster ID (area) and z is the member ID (signaling point).

ERRORS

Invalid key

Missing key

EXAMPLES

DISPLAY-SGSPMC:;
DISPLAY-CONN:SGPID=0***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgspmc:;
         ------------------------------------------------------------------
            DPC            NAID    PCST    ISUPST    SCCPST            SGPID
      ---------------------------------------------------------------------
                3-10-0         200  INACC  UNAVAIL  UNAVAIL          tweety
                 2-100-1        200  INACC  UNAVAIL  UNAVAIL          tweety
            1-100-1         100  INACC  UNAVAIL  UNAVAIL          tweety
<SUCCESS>:: 3 records found
```

Table 5-31: SGSPMC Display Values

NAID	PCST	ISUPST	SCCPST
The Network Appearance ID	Status of the SPMC represented by a particular point code.	Status of ISUP in an SPMC.	Status of SCCP in an SPMC.

5.5.9 SS7 Destination Point Code (SGDPC)

NAME

SGDPC — Defines the SS7 Destination Point Code (DPC) that is reachable by the Signaling Gateway.

COMMANDS

ADD — Adds an SS7 DPC that is either reachable via SS7 link or IP.

ADD-SGDPC:DPC=dpc***,NAID***=naid***,TYPE***=type***[,SGLIST***=sglist***,MODE***=mode***];***

DELETE — Deletes an SS7 DPC that is reachable via SS7 link or IP.

DELETE-SGDPC:DPC=dpc***,NAID***=naid***;***

DISPLAY — Displays SS7 DPC that is reachable via SS7 link or IP.

DISPLAY-SGDPC:[DPC=dpc***][,NAID***=naid***][,SGPID***=sgpid***];***

PARAMETERS

dpc — Point Code of the SS7 node that is reacheable either via SS7 link or IP.

naid — Network Appearance ID of the SS7 node. Valid values are 0-0x7fffffff. NAID must already be defined in the SGSPNA entry.

type — Type of point code. Valid types are:

- SS7 - SS7 point code reachable via SS7 link
- IPSS7 - SS7 point code reachable via IP

sglist — List of SGs through which the point code is reached. The list is a string in the format of n1/n2/n3/..., up to 5 SG IDs. This parameter is needed only when the point code type is of IPSS7.

mode — Traffic mode among the SGs for the point code. This parameter is needed only when the point code type is of IPSS7. Valid modes are:

- OVERRIDE - SGs operate in override mode for the point code
- LOADSHARE - SGs operate in loadsharing mode for the point code

Default mode is set to LOADSHARE.

sgpid — SGP ID. It is a string of 15 characters. This field is used to obtain status of the point code as viewed from the specified SGP.

ERRORS

Database operation failed

Configuration limit exceeded

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Error configuring M3UA stack

SG does not exist

NA does not exist

DPC exists in SG

EXAMPLES

ADD-SGDPC:DPC=2-10-1***,NAID=***200***,TYPE=***SS7***;***
ADD-SGDPC:DPC=3-10-0***,NAID=***200***,TYPE=***IPSS7***,SGLIST=***sg2***;***
DELETE-SGDPC:DPC=2-10-1***,NAID=***200***;***
DISPLAY-SGDPC:;
DISPLAY-SGDPC:NAID=200***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgdpc:;
-------------------------------------------------------------------
      DPC    NAID   TYPE       MODE    SGLIST    PCST        SGPID
--------------------------------------------------------------------
   1-10-0     100    SS7          -         -     ACC    sylvester
   1-10-1     100    SS7          -         -     ACC    sylvester
   2-10-0     200    SS7          -         -     ACC    sylvester
   2-10-1     200    SS7          -         -     ACC    sylvester
   3-10-0     200  IPSS7  LOADSHARE       sg2     INACC  sylvester
<SUCCESS>:: 5 records found
```

5.5.10 Peer SG (SGPSG)

NAME

SGPSG Defines the attributes of a remote peer SG.

COMMANDS

ADD Adds a remote peer SG configuration.

ADD-SGPSG:SGID=sgid[,MODE=mode];

DELETE Deletes a remote peer SG configuration.

DELETE-SGPSG:SGID=sgid***;***

MODIFY Modifies the configuration of a remote peer SG.

MODIFY-SGPSG:SGID=sgid***,MODE***=mode***;***

DISPLAY Displays the remote peer SG configuration.

DISPLAY-SGPSG:[SGID=sgid***];***

PARAMETERS

sgpid SGP ID. It is a string of 15 characters.

mode Traffic mode among the SGPs in the SG. Valid modes are:

- OVERRIDE - SGs operate in override mode
- LOADSHARE - SGs operate in loadsharing mode

Default mode is set to LOADSHARE.

ERRORS

Database operation failed

Configuration limit exceeded

License number of remote peer SG reached

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Error configuring M3UA stack

SG does not exist

DPC exists in SG

SGP exists in SG

EXAMPLES

ADD-SGPSG:SGID=sg2***,MODE***=LOADSHARE***;***
MODIFY-SGPSG:SGID=sg2***,MODE***=OVERRIDE***;***

SAMPLE OUTPUT

```
MML_TH>dis-sgpsg:;
----------------------------
             SGID       MODE
----------------------------
              sg2   OVERRIDE
<SUCCESS>:: 1 record found
```

5.5.11 Peer SGP (SGPSGP)

NAME

SGPSGP — Defines the attributes of a remote peer SGP.

COMMANDS

ADD — Adds a remote peer SGP configuration.

ADD-SGPSGP:PSGPID=psgpid***,SGID***=sgid***[,IP1***=ip1***][,IP2***=ip2***][,HOSTNAME***=hostname***] [,SCTPPORT***=sctpport***];***

MODIFY — Modifies the configuration of a remote peer SGP.

MODIFY-SGPSGP:PSGPID=psgpid***[,IP1***=ip1***][,IP2***=ip2***][,HOSTNAME***=hostname***] [,SCTPPORT***=sctpport***] [,OPERST***=operst***][,LSGPID***=lsgpid***];***

DELETE — Deletes a remote peer SGP configuration.

DELETE-SGPSGP:PSGPID=psgpid***;***

DISPLAY — Displays the configuration and status of the remote peer SGPs.

DISPLAY-SGPSGP:[PSGPID=psgpid***][,LSGPID=l***sgpid***];***

PARAMETERS

psgpid — Remote peer SGP ID. It is a string of 15 characters.

sgid — Remote SG ID to which the peer SGP belongs. It is a string of 15 characters. The SGPSG entry referenced by sgid must be created in advanced.

ip1 — Primary IP address of the peer SGP used for communication over SCTP. Address is entered in dot notation format. A '-' can be entered to de-assign a previously assigned address.

ip2 — Secondary IP address of the peer SGP used for communication over SCTP. Address is entered in dot notation format. A '-' can be entered to de-assign a previously assigned address.

hostname — Domain name of the system where the remote peer SGP runs. This field can be used in place of the IP addresses if Internet Domain Name System (DNS) is available. Input is a string of 128 characters. A '-' can be entered to de-assign a previously assigned address.

sctpport — Remote port number to which SCTP is bound. Default port is set to 2905.

lsgpid — Local SGP ID. It is a string of 15 characters. This field is applicable only when OPERST is specified. It indicates the SGP in which OPERST is performed. If not specified, OPERST will be applied in all local SGPs.

operst Operation performed on a remote peer SGP from the specified SGP in local SG. Valid values are:

- DISCONN - disconnect SCTP association between the local and remote SGPs
- CONN - establish SCTP association between the local and remote SGPs
- DOWN - inactivate a remote SGP so that it becomes unavailable
- UP - activate a remote SGP so that it becomes available

Note*: The IP address and hostname fields are mutually exclusive, i.e. only either IP addresses OR hostname can be specified at the same time, but not both.*

ERRORS

Database operation failed

Configuration limit exceeded

Record already exists

Record does not exist

Invalid key

Missing key

Missing mandatory parameters

Error configuring M3UA stack

Invalid IP address

IP addresses cannot be identical

Only IP addresses OR hostname is accepted

EXAMPLES

ADD-SGPSGP:PSGPID=psgp1,SGID=sg1,IP1=10.0.0.1,IP2=10.0.0.2;
MODIFY-SGPSGP:PSGPID=psgp1***,IP1=-,HOSTNAME=***sg1.xyz.com***,SCTP-PORT=***3000***;***
DELETE-SGPSGP:PSGPID=psgp1***;***
DISPLAY-SGPSGP:

SAMPLE OUTPUT

```
MML_TH>dis-sgpsgp:;
-------------------------------------------------------------------------------
PSGPID: vortex     SGID: sg2
HOSTNAME:
IP1: 155.226.145.171   IP2:     SCTPPORT: 2905  ASSOCID: 0  OPERST: UP  STATUS: DISCONN  LSGPID:
sylvester

-------------------------------------------------------------------------------
PSGPID: vortex     SGID: sg2
HOSTNAME:
```


IP1: 155.226.145.171 IP2: SCTPPORT: 2905 ASSOCID: 0 OPERST: UP STATUS: DISCONN LSGPID: tweety

<SUCCESS>:: 2 record found

5.5.12 Common UNIX Commands

Only the MML and Signaling Gateway utility commands are required to maintain and configure the Signaling Gateway software. Nevertheless, familiarity with some common UNIX commands is useful. These commands include the following:

- **pg** to scroll by page
- **cat** to dump a file to the screen
- **cd** to change working directory
- **cp** to copy a file
- **date** to display date and time
- **grep** to search a file for a pattern
- **kill** to terminate a process
- **login** to sign onto a system
- **ls** to list contents of a directory
- **more** to browse through a text file
- **mt** to access the magnetic tape for rewinding, ejecting, etc.
- **pkgadd** to install a new software package
- **pkginfo** to display information about installed software packages
- **pkgrm** to remove installed software packages
- **ps** to report process status
- **rlogin** to remotely sign onto a system
- **rsh** to execute a command shell on a remote system
- **tail** to browse the last part of a file
- **tar** to create tape archives and add or extract files
- **telnet** to remotely sign on to a system
- **time** to measure process execution time
- **vi** to edit text files

Consult UNIX manual pages for information on these commands. For example, enter the following string at a UNIX prompt to see the manual page for the UNIX command **more**:

```
$man more  [Ret]
more(1)                     User Commands                     more(1)
NAME
     more, page - browse or page through a text file

SYNOPSIS
more(1)                     User Commands                     more(1)

NAME

 more, page - browse or page through a text file

SYNOPSIS

     more [ -cdflrsuw ] [ -lines ] [ +linenumber ] [ +/pattern ]
          [ filename ... ]
                               ⋮
```

5.6 Passive Monitor MML Commands

5.6.1 Passive Monitor Link (PMLINK)

NAME

PMLINK — Adds, modifies, deletes a passive monitor link, or displays information about a passive monitor link.

COMMANDS

ADD — Adds a passive monitor link to a ss7board. The ss7board managed object must be added as a passive monitor board (by setting the PM option to ON) and must be configured ON prior to this command.

ADD-PMLINK: HOSTNAME=hostname***, BOARDNM***=boardnm***,***
INST=inst***, PORT***=port***;***

Note*: Proper timeslot and clocking settings must be done before adding a passive monitor link. Please see MML command descriptions in Section 5.2.6, SS7Board (SS7BOARD) and Section 5.2.19, Time Slot (TIMESLOT).*

MODIFY — Modifies the administrative state of a passive monitor link.

MODIFY-PMLINK: HOSTNAME=hostname***, BOARDNM***=boardnm***,***
INST=inst, PORT=port, ADMINSTAT=adminstat;

DELETE — Deletes a passive monitor link.

DELETE-PMLINK: HOSTNAME=hostname, BOARDNM=boardnm,
INST=inst***, PORT***=port***;***

DISPLAY — Displays passive monitor link attributes.

DISPLAY-PMLINK: [HOSTNAME=hostname [,BOARDNM=brdnm
[,INST=inst [,PORT=port]]]];

PARAMETERS

boardnm — Board type, entered as one of the following values:

- pci3xpq common name for 24-port PCI bus boards (pci37xpq)
- pci3xapq common name for 24-port PCI bus boards (pci37xapq)
- pmc8260 common name for 64-port CompactPCI bus boards (pmc8260)
- artic8260 common name for 64-port CompactPCI bus boards (artic1000 and artic2000)

inst — Physical instance number of the board. It is an unsigned integer from 0 to 8

port — Port number of the link, entered as a numerical value. Valid range depends on the board type:

- pci3xpq 0 to 23
- pci3xapq 0 to 23
- pmc8260 0 to 63
- artic8260 0 to 63

adminstat Passive monitor link administrative state. Possible values are:

- DEACTIVATE : Link is deactivated. MSUs are not received on this link.
- ACTIVATE: Link is activated and ready to capture and pass on MSUs.

operstat Passive monitor link operational state. Possible values are:

- SHUTOFF : Link is deactivated by the passive monitor layer due to an internal resource outage.
- INACTIVE : Link is deactivated due to an application's request.
- IDLE : Link is activated but not receiving any MSUs. Either the connection with the SS7 link is lost or the SS7 link is not operational.
- OOS: Out of Service, link is active but only receiving LSSU SIOS signals.
- ALIGNING: Link is active but only receiving LSSU-SIO, SIN, SIE. Link is in alignment period.
- INSERVICE: Link is active and receiving FISU or MSU signals (SS7 link is in service).
- PROC-OUT: Link is active and receiving LSSUs because the sending side of the SS7 link has processor outage.

linkf Number of link failures.

rxframes Number of received MSUs.

rxoctets Number of received MSU octets

rsu_e Number of received signal units in error

d_rxl Number of discarded signal units due to invalid HDLC length

d_bo Number of discarded signal units due to receiver buffer overflow

ERRORS

<ERROR>:: Can not add pmlink

<ERROR>:: Can not perform operation - board not configured

<ERROR>:: Can not delete pmlink

<ERROR>:: MO exists

<ERROR>:: Generic error

<ERROR>:: Can not get pmlink attributes

<ERROR>:: Missing ADMINSTAT attribute

<ERROR>:: Missing BOARDNM attribute

<ERROR>:: Missing INST attribute

<ERROR>:: Missing PORTNUM attribute

<ERROR>:: Missing parameter

<ERROR>:: No such a SS7BOARD MO instance

<ERROR>:: No such a pmlink instance

<ERROR>:: No such a PORT MO instance

<ERROR>:: Nothing to list

<ERROR>:: Link information can not be retrieved

<ERROR>:: Invalid stream no is retrieved from spmd

<ERROR>:: mtpl2 MO operation failed

<ERROR>:: LINK is activated.

<ERROR>:: Nothing to list.

EXAMPLES

ADD-PMLINK: HOSTNAME=chicago***, BOARDNM***=pci3xapq***, INST***=0***, PORT***=0***;***
MOD-PMLINK: HOSTNAME=chicago***, BOARDNM***=pci3xapq***, INST***=0***, PORT***=0***, ADMINSTAT***=ACTIVATE***;***
MOD-PMLINK: HOSTNAME=chicago***, BOARDNM***=pci3xapq***, INST***=0***, PORT***=0***, ADMINSTAT***=DEACTIVATE***;***
DISPLAY-PMLINK: HOSTNAME=chicago***, BOARDNM***=pci3xapq***, INST***=0***;***
DISPLAY-PMLINK: HOSTNAME=chicago***, BOARDNM***=pci3xapq***;***
DISPLAY-PMLINK: HOSTNAME=chicago***;***
DISPLAY-PMLINK:;

SAMPLE OUTPUT

```
MML_TH>dis-pmlink:;

-----------------------------------------------------------------------------------------------------------
         HOSTNAME    BOARDNM INST PORT   ADMINSTAT   OPERSTAT LINKF RXFRAMES RXOCTETS RSU_E D_RXL   D_BO
-----------------------------------------------------------------------------------------------------------
           chicago   pci3xpq    0    0    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    1    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    2    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    3    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    4    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    5    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    6    ACTIVATE  INSERVICE     0        0        0     0     0      0
           chicago   pci3xpq    0    7    ACTIVATE  INSERVICE     0        0        0     0     0      0
```

Chapter 6: System Processes

6.1 ChapterOverview

This chapter provides descriptions of the Signaling Gateway system processes (daemons) and their configuration files. The system processes are summarized in the following table.

Table 6-1: Process Summary

New Command	Description
AccessAlarm (alarmd)	Starts the Alarm handling process
AccessMOB (mob)	Starts the Managed Object Browser graphical user interface
AccessOMAP (omapd)	Starts the Operations, Maintenance, and Administration process (optional)
AccessSNMP	Starts the SNMP agent
AccessStatus	Starts monitoring SS7 links.
AccessMonitor	Starts the system software status monitor
apmd	Starts Application Process Manager (APM) daemon.
dkmd	Distributed Kernel Memory (DKM) manager daemon.
dsmd	Sets up the distributed shared memory manager system process.
logd	Starts the log process (LOG_MNGR) for message logging capabilities
mlogd	Starts master event log daemon.
mmi	Starts the Man-Machine Language user interface
mml	Starts the Man-Machine Language user interface
netd	Sets up connections for inter-machine messaging in distributed environment
scmd	Starts the SCCP user part
sctpd	Starts the Signaling Gateway SCTP process on Solaris 8/9.
sgpd	Starts the Signaling Gateway SG processes
spmd	Starts the NewNet Distributed7 infrastructure
upmd	Starts User Part Multiplexer

6.2 ProcessListing

***Important**: Please refer to Chapter 2: Distributed7 Overview for a list of the user commands with external dependencies to make sure your environment has the necessary software libraries.*

To use NewNet Distributed7, set the $**EBSHOME** environment variable and include ***$EBSHOME/access/bin*** in the command path. The $**EBSHOME** environment variable should be set to the path where the NewNet Distributed7 software is installed.

To set the variable, use a C-shell command similar to this sample:

```
setenv EBSHOME /<samedir>/<mydir>/<mySS7>
```

***Important**: $EBSHOME must be less than or equal to 1024 characters.*

To add the NewNet Distributed7 ***bin*** directory into the command path, use the following command:

```
setenv PATH ${PATH}:$EBSHOME/access/bin
```

On-line reference manuals on all system processes are also available in the NewNet Distributed7 system. These reference manuals are provided in the form of manual pages so that the user can invoke the UNIX standard ***man(1)*** utility to review the information contained in them. The NewNet Distributed7 manual pages are provided within the ***$EBSHOME/access/manpages*** directory. Therefore, the user should set the **MANPATH** environment variable as follows:

```
setenv MANPATH ${MANPATH}:$EBSHOME/access/manpages
```

6.2.1 AccessAlarm (alarmd)

NAME

AccessAlarm or ***alarmd*** Starts the alarm handling process.

SYNOPSIS

AccessAlarm [-o] [-d dir***] [-n*** nfile***] [-m*** msize***]***
alarmd [-o] [-d dir***] [-n*** nfile***] [-m*** msize***]***

DESCRIPTION

AccessAlarm or

alarmd This daemon is responsible for collecting, analyzing, logging, and displaying alarm messages that may be generated by the user/kernel-space components comprising the NewNet Distributed7 system software and/or user application programs running under the NewNet Distributed7 environment.

A list of alarm conditions that may be encountered by the NewNet Distributed7 system software is provided in the form of a set of alarm text files that are all located in the ***$EBSHOME/access/RUN/config/ALARM*** directory. These files contain information about all alarm conditions that may be encountered by a particular software module and about the specifics of the individual alarm conditions, e.g., alarm module identifier, group identifier, severity, alarm text, and associated parameters. The severity and/or text of a particular alarm condition can be modified by the users of the NewNet Distributed7 product simply by editing the information contained in these files and re-starting the ***alarmd*** daemon. Care should be taken, however, not to change the module and group identifier information as well as the number of parameters supplied within individual alarm text messages, as this misleads the ***alarmd*** daemon.

A list of individual alarm groups is provided within the *alarmGroups* file under the ***$EBSHOME/access/RUN/config/ALARM*** directory. Also listed in this directory is the *alarmConfig* file, which contains configuration related information for use by the ***alarmd*** daemon (whether messages collected by the ***alarmd*** daemon should be displayed on the system console in addition to being logged for off-line analysis).

Unless the ***-d dir*** command line option is in use, The log files maintained by the ***alarmd*** daemon are located under the ***$EBSHOME/access/RUN/alarmlog*** directory and they all start with the AccessAlarms prefix. Users interested in reviewing the contents of the alarm log files maintained on specified hosts within a NewNet Distributed7 environment can use the *ebs_report* command-line utility, i.e., for retrieving and displaying selected pieces of information stored in alarm log files.

While in operation, ***alarmd*** will be registered exclusively with the

NewNet Distributed7 environment on the local host machine as a daemon object, under the name ALM_MNGR—a macro defined in the <***api_macro.h***> header file.

-o This option is used to overwrite the time stamp information contained in the alarm message received. By default, the time stamp information is populated when the alarm message is submitted, i.e., at the point of origination. ***alarmd*** uses this information as is.

-d dir Stores alarm log files on specified host machines. By default, alarm log files are located under the ***alarmlog*** directory in the ***$EBSHOME/access/RUN*** directory. If the ***dir*** is specified, the alarm log files are expected to be located under the ***dir/alarmlog*** directory. If ***dir*** directory (or ***alarmlog*** sub-directory) does not exist, ***alarmd*** will make an attempt to create all necessary directories.

-n nfile Allows a maximum number of ***nfile*** log files to co-exist under the log directory at any time. When this count is exceeded, delete the oldest log file to create room for new log files. By default, ***alarmd*** allows up to 10 log files to co-exist, where each log file can grow up to ***msize*** kilobytes in size. The absolute maximum number of log files that can co-exist on a system is 100.

-m msize Allows each log file in log directory to grow to ***msize*** kilobytes in size. By default, ***alarmd*** allows each log file to grow to 512 kilobytes in size before it starts writing into a new log file. When the total number of log files exceeds the limit set via the ***nfile*** argument, ***alarmd*** deletes the oldest log file before creating a new log file.

Note*: The alarm process is automatically started by the apmd process and does not have to be started separately. This command is only needed to activate the ALM_MNGR process if it is deactivated while the system is running.*

Important*: Refer to the Maintenance Manual for trouble reports and alarm definitions.*

FILES

$EBSHOME/access/RUN/config/ALARM/alarmGroups

$EBSHOME/access/RUN/config/ALARM/alarmConfig

$EBSHOME/access/RUN/alarmlog/AccessAlarms.*

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.2.14, **ebs_report** on page 7-25

6.2.2 AccessMOB (mob)

NAME

AccessMOB Starts the graphical user interface.

SYNOPSIS

AccessMOB sp

mob sp

DESCRIPTION

AccessMOB or

mob Starts the Managed Object Browser graphical user interface for node configuration of all managed objects of a node (instead of using MML).

The ***mob*** interface for a specified signalling point can be started automatically by the *apmd* daemon (at system start-up time) provided that there is a corresponding entry in the *apmd* configuration file. Alternatively, it can be started manually from the command line.

While in operation, ***mob*** will be registered exclusively with the NewNet Distributed7 environment on the local host machine as a daemon object, under the name GUI_MNGR(sp) (a macro defined in the <***api_macro.h***> header file).

sp The logical signalling point number used with *upmd*, or *sg_stop*.

***Important**: The upmd process **MUST** be running before executing this command. Use ebs_ps to confirm that these processes exist.*

Related Information

- Section 6.2.12, **mml** on page 6-22
- Section 6.2.17, **upmd** on page 6-28

6.2.3 AccessOMAP (omapd)

NAME

AccessOMAP or

omapd Starts the SS7 OMAP process.

SYNOPSIS

AccessOMAP [-q report_frequency] sp

omap [-q report_frequency] sp

DESCRIPTION

AccessOMAP or

omapd Starts the SS7 Operations, Maintenance, Administration Part (OMAP) process. The ***omapd*** daemon is responsible for collecting and storing various pieces of SS7-specific measurements data associated with a user-specified signalling point (SP). This data includes information about Message Transfer Part (MTP) Level 2 (collected at 30-minute intervals) and Level 3 measurements (collected at 5- and 30-minute intervals), Signalling Connection Control Part (SCCP) measurements (collected at 30-minute intervals), and Transaction Capabilities Application Part (TCAP) measurements (collected at 30-minute intervals).

The log files maintained by the ***omapd*** daemon are located under the ***$EBSHOME/access/RUN[0-7]/omaplog*** directory for the corresponding signalling point. Users interested in reviewing the contents of the log files maintained by the ***omapd*** daemon and generating summary reports can use the `omap_report` command-line utility (or customized versions of it) whose source code listing is given under the ***$EBSHOME/access/sample/omap*** directory. Alternatively, the *oam_retrieve()* function, which is part of the NewNet Distributed7 OAM API Library, can be used to retrieve measurements data collected by the AccessOMAP daemon.

The ***omapd*** daemon for a specified signalling point can be started automatically by the *apmd* daemon (at system start-up time) provided that there is a corresponding entry in the *apmd* configuration file. Alternatively, it can be started manually from the command line.

As an operational practice, the AccessOMAP daemon on each host needs to be started prior to starting the daemon processes associated with the MTP, SCCP, and TCAP layers on that host. If this practice is not followed, then the measurements data collected and stored by the AccessOMAP daemon do not include the first set of statistics reported by the individual SS7 protocol layers. Also, if any of these SS7 protocol layers are instantiated on multiple hosts, then the AccessOMAP daemon for the corresponding signalling point needs to be started on all such hosts in order to be able to collect measurements data through all involved hosts.

While in operation, ***omapd*** is registered exclusively with the NewNet Distributed7 environment on the local host machine as a daemon object, under the name OMAP_MNGR(sp) (a macro defined in the <***api_macro.h***> header file).

-q report_frequency This option is used to specify the time interval in minutes at which all measurements are collected. If this option is not specified, then the measurement interval defaults to 30 minutes

sp The logical signalling point number.

FILES

$EBSHOME/access/RUN[0-7]/omaplog/mtp2.mmddyy

$EBSHOME/access/RUN[0-7]/omaplog/mtp3.mmddyy

$EBSHOME/access/RUN[0-7]/omaplog/mtp3_5min.mmddyy

$EBSHOME/access/RUN[0-7]/omaplog/sccp.mmddyy

$EBSHOME/access/RUN[0-7]/omaplog/tcap.mmddyy

$EBSHOME/access/RUN/omaplog/tcap.mmddyy

$EBSHOME/access/sample/omap/omap_report.c

$EBSHOME/access/sample/omap/Makefile

Important*: The upmd process must be running before executing this command. Use ebs_ps to confirm that these processes exist.*

NOTES

When the TCAP over TCP/IP feature is in use, the TCAP layer will report its OMAP statistics to the AccessOMAP instance associated with the signalling point 0. Unless this instance of AccessOMAP is alive, no OMAP measurements data will be collected for TCAP [over TCP/IP] applications. Also note that since this data is not associated with any signalling point, per se, it will be saved under the ***$EBSHOME/access/RUN/omaplog*** directory as opposed to being saved in ***$EBSHOME/access/RUN[0-7]/omaplog*** directory. The ***omap_report*** utility contains built-in intelligence to search through all appropriate directories to locate the log files maintained by the AccessOMAP daemon.
The ***omapd*** daemon archives the log files under the *omaplog* directory every week by moving them to a *omaplog.mmddyy* directory that is at the same level as the *omaplog* directory. It is highly recommended that users check on the size of accumulated *omaplog* files on their system from time to time and clean up the old log files from their system to guard against excessive disk space consumption.

Related Information

• Section 6.2.7, **apmd** on page 6-14

6.2.4 AccessSNMP

NAME

snmp_p Simple network management protocol interface.

SYNOPSIS

$EBSHOME/access/bin/snmp_p [-a] -v 1|2 sp

$EBSHOME/access/bin/AccessSNMP [-a] -v 1|2 sp

DESCRIPTION

AccessSNMP Starts the SNMP agent which replies to all the SNMPv1 and SNMPv2 requests that come from network managers. It responds according to the MIB view of Distributed7.

sp Identifies the signalling point of interest.

snmp_p This daemon is responsible for establishing a standardized interface between external network management entities and the Distributed7 platform using version 1 or version 2 Simple Network Management Protocol (SNMP). The selection of the SNMPv1 vs. SNMPv2 protocol is via the "-v" command line option provided to ***snmp_p*** at the time of program invocation.

The ***snmp_p*** interface for a specified signalling point can be started automatically by the *apmd* daemon (at system start-up time) provided that there is a corresponding entry in the *apmd* configuration file. Alternatively, it can be started manually from the command line. While in operation, ***snmp_p*** will be registered exclusively with the Distributed7 environment on the local host machine as a daemon object, under the name PRTCL_MNGR(sp) (a macro defined in the <***api_macro.h***> header file). This behavior of SNMP agent can be changed using the "-a" command line option. where SNMP agent will use the Distributed7 hostname assigned to a machine instead of using the official hostname for that machine.

Upon start-up, ***snmp_p*** will perform the following tasks:

- read through the SNMPv1 and SNMPv2 "*.conf" files located under the corresponding ***$EBSHOME/access/RUN*/config/SNMP*** directory
- establish a transport endpoint for communication with external management entities using the User Datagram Protocol (UDP)
- spawn the corresponding ***snmp_i*** daemon to be able to perform Managed Object related tasks if and when it becomes necessary
- wait for UDP datagrams from external network management entities or trap requests initiated by the Distributed7 system software

When the ***snmp_p*** daemon receives a UDP datagram from an external management entity, it will interact with the corresponding ***snmp_i*** daemon (i.e., in order to resolve and process the incoming request) and reply to the originating party with an appropriate response. Alternatively, when the ***snmp_p*** daemon receives a trap request via the ***snmp_i*** daemon, it will relay this request to the external network management entity via a datagram.

Note that it is also possible to define interfaces between the Distributed7 alarm handler and the ***snmp_i*** daemon process such that ***snmp_p*** is informed about selected alarm conditions that occur on the Distributed7 platform and it subsequently notifies the external network management entities to that effect by generating SNMP trap requests. This latter capability involves customizing of the *alarmd* daemon by manipulating the AlmExt.c file provided under the ***$EBSHOME/access/sample/alarm*** directory, re-compiling/linking the source/object codes in that directory, and subsequently replacing the default *alarmd* daemon executable under the ***$EBSHOME/access/bin*** directory with the newly constructed version of it. More information about this procedure can be found in the *Distributed7 User Manual*.

The operations of the ***snmp_p*** daemon are controlled via the SNMPv1 and SNMPv2 Management Information Base (MIB) text files and SNMP command table located under the ***$EBSHOME/access/RUN/config/SNMP*** directory. Following initial system software installation, both of these files can be edited to customize the operations of the Distributed7 SNMP agent (e.g., by modifying the access privileges, defining attributes for additional managed objects, modifying existing SNMP commands and/or introducing additional ones).

Important*: Following initial software installation, all configuration files listed under the* ***$EBSHOME/access/RUN/config/SNMP/etc*** *directory must be copied by the system administrator to all appropriate* ***$EBSHOME/access/RUN*/config/SNMP*** *directories with the "*.conf" extension and hand-edited to specify the Internet Protocol (IP) address of the external network management entity and setup-related system parameters. Once this step is successfully completed, the Distributed7 SNMP agent can be started properly. For further information about these configuration files and their exact use, refer to the Distributed7 documentation.*

FILES

$EBSHOME/access/RUN/config/SNMP/README

$EBSHOME/access/RUN/config/SNMP/mib_text.v1

$EBSHOME/access/RUN/config/SNMP/mib_text.v2

$EBSHOME/access/RUN/config/SNMP/snmp_cmnd.tbl

$EBSHOME/access/RUN/config/SNMP/etc/acl.ini

$EBSHOME/access/RUN/config/SNMP/etc/community.ini

$EBSHOME/access/RUN/config/SNMP/etc/context.ini
$EBSHOME/access/RUN/config/SNMP/etc/party.ini
$EBSHOME/access/RUN/config/SNMP/etc/trap.ini
$EBSHOME/access/RUN/config/SNMP/etc/view.ini
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/acl.conf
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/community.conf
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/context.conf
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/party.conf
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/trap.conf
$EBSHOME/access/RUN<_ s_ p#>/config/SNMP/view.conf
$EBSHOME/access/sample/alarm/AlmExt.c

6.2.5 AccessStatus

NAME

AccessStatus Monitors signalling point configuration, MTP level 2 and level 3 status, and traffic capacity utilization of SS7 links.

SYNOPSIS

AccessStatus sp

DESCRIPTION

AccessStatus Displays a scrollable *tcl* window with one row of information for each SS7 link defined in the corresponding Signalling Point (sp). AccessStatus can be started on each host where the NewNet Distributed7 CORE system is running. This is to say that MTP/L2 or MTP/L3 is not necessary in order to start AccessStatus.

Upon start-up, AccessStatus gets the current link information from the MTP/L3 and displays information only for these links. If a link is added or deleted, MTP/L3 will inform all AccessStatus processes so that the correct link information can be displayed.

sp Signalling point number of the system.

DISPLAY

For each link entry, the following information is displayed:

LinkSet The linkset name of the link

Link The link name

SLC Signalling Link code of the link

L3State MTP/L3 status, can be one of the following:

- *failed* - link is unavailable
- *available* - link is available

Inhibit Inhibition, can be on of the following:

- *local* - link is locally inhibited
- *remote* - link is remotely inhibited
- *loc/rem* - link is locally and remotely inhibited

ProcOut Processor Outage, can be one of the following:

- *local* - local processor outage is set
- *remote* - remote processor outage is set
- *loc/rem* - local and remote processor outage is set

L2State MTP/L2 state of the link, can be one of the following:

- *pow_off* - power off
- *oos* - out of service

- *init_al* - initial alignment
- *alg_ready* - alignment ready
- *alg_not* - alignment not ready
- *is* - in service
- *proc_out* - processor outage

SueCnt	SUERM (Signalling unit Error Rate Monitor) counter
TxFrame	Number of transmitted frames per second
RxFrames	Number of received frames per second
TxBand	Transmit bandwidth usage in percentage
RxBand	Receive bandwidth usage in percentage

EXAMPLE

AccessStatus 0

Starts up AccessStatus for signalling point 0.

6.2.6 AccessMonitor

NAME

AccessMonitor Starts the system software status monitor.

SYNOPSIS

AccessMonitor <sp#>

DESCRIPTION

AccessMonitor Provides the ability to monitor the status of the SS7 protocol stack running on multiple hosts within a NewNet Distributed7 environment, via a selected host, with a Graphical User Interface (GUI). It supports both stand-alone and distributed configurations. When executed under a distributed configuration, it also monitors the health of the kernel-level TCP/IP connections to all remote hosts on an on-going basis.

Upon start-up, AccessMonitor brings up a map of hosts that are currently configured and accessible through the local host. The main window gives general information about the current status of SS7 layers for each host and the TCP/IP connections among all active hosts on the system. By selecting each active layer from the main window, a new window for each layer will be created which displays further, more detailed, status information about daemon processes and STREAMS components.

sp Signalling point number of the system.

EXAMPLE

AccessMonitor 0

Starts up AccessMonitor for signalling point 0.

6.2.7 apmd

NAME

apmd Starts Application Process Manager (APM) daemon.

SYNOPSIS

apmd [-c|s|x] [-f cfgfile***]***

DESCRIPTION

apmd Starts the ***apmd*** daemon, which is a general application process manager. Its primary responsibility is to create and manage processes according to instructions specified in an *apmconfig* configuration file. It is also responsible for processing application requests (placed with APM library calls) for creation of a new process, communication between parent and child processes, detection of child process termination, and network-based inter-process communication via UNIX signals. The ***apmd*** is also capable of communicating with its peers on other hosts in a distributed processing environment (e.g., to spawn a process on a remote host or send signals to remote processes). This communication is message-based and uses the functionality provided by the NewNet Distributed7 SPM library.

-c Starts the AccessCRP (Call Routing Point) version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values prior to program execution. This version supports multiple application domains on a single host. The settings of the above environment variables are used to determine the specific ***apmd*** instance to invoke.

-s Starts the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variable must be set to an appropriate value prior to program execution. This version supports multiple application domains on a single host. The current setting of the ***$DOMID*** environment variable is used to determine the specific ***apmd*** instance to invoke.

-x Starts the normal NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value prior to program execution. This version does not support multiple application domains on a single host.The ***apmd*** daemon executes in a local-exclusive mode, i.e., only one instance of this version of ***apmd*** may be executing on a host.

-f cfgfile Use the configuration file specified by ***cfgfile*** instead of the default. By default, ***apmd*** uses the *apmconfig* or ***apmconfig.old*** configuration file under an appropriate release directory. (See the FILES section.)

Important*: The IPC shared memory segment for the APM trace functionality must be created in advance using the apm_trinit utility. If it is not created, then* ***apmd*** *will terminate.*

The ***apmd*** supports different types of application environments, as described by the options (c, s, x). If the user does not explicitly specify one of the options in the command, ***apmd*** will determine the appropriate environment based on environment variable settings and the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes an AccessCRP environment.
- If none of the above environment variables are set but ***$EBSHOME*** is set, it assumes a basic NewNet Distributed7 environment.

The AccessSERVICES and AccessCRP versions of ***apmd*** support multiple application domains on a single host. Multiple instances of either version of the ***apmd*** daemon can be invoked on a single host to create and manage independent sets of application processes. The basic NewNet Distributed7 version does not support multiple application domains. Only one instance of this version can exist on a host. However, all three versions of the daemon may coexist on a given host.

Similar to the UNIX *init(1)* daemon, ***apmd*** maintains an internal run state during its life cycle. This run state corresponds to a software configuration which specifies a group of processes that should be spawned by ***apmd***. The configuration of active processes for different run states is defined in the *apmconfig* file. The run state of ***apmd*** changes either when it executes a pre-defined scenario from the *apmconfig* file which puts it in a new state or when the *apm_setstate* utility is used to request state change from the command line.

Once started, ***apmd*** reads all entries listed in *apmconfig* and copies the information to an internal process control table. From then on, ***apmd*** will use the information stored in the internal process control table. It will not consult the *apmconfig* file until the execution of the *apm_update* utility causes it to re-read the configuration file.

The first run state is retrieved by ***apmd*** from *apmconfig* during start-up. When ***apmd*** starts, it searches the *apmconfig* file for an *initdefault* entry. If one exists, ***apmd*** uses the run state specified in that entry as the initial state to enter. If an *initdefault* entry does not exist in the *apmconfig* file, then ***apmd*** enters a run state as follows:

- In AccessCRP environments, it moves to the **D** state.
- In AccessSERVICES environments, it moves to the **A** state.
- In NewNet Distributed7 environments, it moves to the **init** state.

When ***apmd*** has its initial state, and whenever its run state changes, ***apmd*** scans all entries listed in its internal process control table and executes each line that has an execution state which matches its current state. If the execution state of a particular process entry does not match the current run state, this entry is completely ignored. When an entry instructs ***apmd*** to spawn a process, ***apmd*** reads the entry for the process first, and then identifies the host where the process should be created. If the host is not local, ***apmd*** contacts its peer on the appropriate remote host and submits the process entry to that ***apmd*** for processing. For entries specifying the local host, ***apmd*** creates a new child process and executes the user-specified program.

The ***apmd*** can be configured to do more than simply spawning and terminating processes. Each entry in the process control table also specifies the behavior of ***apmd*** during the start-up and life of a child process. For example, ***apmd*** can be programmed to wait a specified time for an acknowledgment that a process has started (or terminated) before executing the next entry. It can also be programmed to take certain actions if and when a child process terminates. During the life of a process, ***apmd*** can be programmed to continually monitor the process through *heartbeat request* messages, which require *heartbeat response* messages from the process.

An entry in the process table may also instruct the ***apmd*** to change its run state to a new state after executing the line. This action will cause ***apmd*** to again scan the process control table. But now the lines with execution states matching this new state will be executed, possibly executing new scenarios, e.g., spawning a new set of processes and/or terminating an existing set of processes.

During its operations, ***apmd*** reports all process-related activities and run state changes to the *mlogd* process. The *mlogd* process creates a permanent record of these activities in the master log files on the local host.

When ***apmd*** is requested to terminate processes, it takes the actions described in the *apm_stop*, *apm_kill*, and *apm_killall* utilities.

FILES

The default configuration file names are as follows:

NewNet Distributed7 version: ***$EBSHOME/access/RUN/config/PMGR/apmconfig***

AccessSERVICES version: ***$EBSHOME/access/RUN/config/PMGR/apmconfig.old***

AccessCRP version: ***$APPHOME/apmconfig.old***

Related Information

- Section 6.2.11, **mlogd** on page 6-20
- Section 7.3.8, **apm_start** on page 7-49
- Section 7.3.9, **apm_stop** on page 7-51
- Section 7.3.3, **apm_kill** on page 7-38
- Section 7.3.4, **apm_killall** on page 7-40
- Section 7.3.16, **apm_update** on page 7-65
- Section 7.3.5, **apm_ps** on page 7-42
- Section 6.3.1, **apmconfig** on page 6-30

6.2.8 dkmd

NAME

dkmd Distributed Kernel Memory (DKM) manager daemon.

SYNOPSIS

dkmd [-ps] [-m module(s)]

DESCRIPTION

dkmd This daemon is responsible for setting up, maintaining, and tearing down the Distributed Kernel Memory (DKM) and Distributed Record Access (DRA) infrastructures that are available as part of this release of the NewNet Distributed7 system software. The DKM infrastructure allows kernel-resident applications executing under a distributed NewNet Distributed7 environment to share kernel-space information in an efficient manner by reading/writing through local copies of kernel-space data that are replicated on all hosts involved. The DRA infrastructure builds upon the DKM and is intended to fulfill the needs of database-oriented kernel-resident applications. It is through the DRA framework, a kernel application can view its kernel-space data in the form of a distributed database and operate on it.

The ***dkmd*** daemon expects the *netd* daemon on the local host to be up and running when it starts its execution. The only exception to this rule is when the ***dkmd*** daemon is started in the stand-alone mode, using the ***-s*** command-line argument. The ***dkmd*** daemon on a particular host is normally started automatically by the *apmd* daemon on that host provided that there is an entry in the *apmd* configuration file. Alternatively, it can be started manually from the command-line.

While in operation, the ***dkmd*** daemon on each host will be registered exclusively with the NewNet Distributed7 environment on that host as a daemon object, under the name DKM_MNGR (a macro that is defined in the **<*api_macro.h*>** header file).

-p Collect performance statistics by running a small set of performance benchmark tests upon start-up. The collected statistics can be retrieved at a later time simply by sending a SIGUSR1 signal to the ***dkmd*** daemon.

-s Execute in standalone mode. Note that in this mode of operation, ***dkmd*** daemon acts as a lock manager only, which simply facilitates protected access to a kernel-resident memory space allocated on the local host.

-m module Push STREAMS modules specified over the DKM multiplexer in the specified order.

Related Information

- Section 6.2.7, **apmd** on page 6-14

• Section 6.2.14, **netd** on page 6-24

6.2.9 dsmd

NAME

dsmd Sets up the distributed shared memory manager system process.

SYNOPSIS

dsmd [-s]

DESCRIPTION

dsmd Allocates, maintains, and de-allocates resources associated with the distributed shared memory (DSM) subsystem that is available as part of the Signaling Gateway Release 1.0.0 system software. The ***dsmd*** process performs these functions on behalf of application programs that issue requests through the ***libdsm*** API library calls.

-s Executes the process in stand-alone mode. This mode of operation causes the ***dsmd*** process to act as a lock manager to facilitate protected access to the IPC shared memory segments allocated on the local host.

A distributed shared memory segment allows processes executing on the individual hosts of a NewNet Distributed7 environment to attach local copies of the segment to their virtual address space. In other words, UNIX IPC shared memory is replicated on each host. The processes share information by reading/writing from/to these segments. To prevent inconsistencies and/or collisions when reading/writing through a DSM segment, processes must always use synchronization variables (e.g. read/write locks) when reading/writing through local copies of the segment.

The ***dsmd*** process is normally started automatically by the *apmd* process on a particular host. To be started automatically, the command must be an entry in the *apmd* configuration file (e.g. *apmconfig* in *6.3*) for the host. The ***dsmd*** process can also be started from the command-line. In either case, the *netd* process must already be running, unless the **-s** option is used. While running, the ***dsmd*** process on each host is registered exclusively with the NewNet Distributed7 environment on the local host as a daemon object called, DSM_MNGR [a macro defined in the **<*api_macro.h*>** header file].

Once the ***dsmd*** process starts, it contacts its peers across the network to find out if any DSM segment has already been created. If one has, ***dsmd*** will request information about all segments so it can synchronize itself. After it is synchronized, it starts servicing application requests placed by processes executing on the local host. Most DSM requests require communication between the ***dsmd*** processes on the individual hosts.

***Important**: The **dsmd** daemon features a built-in auditing mechanism that has been designed to ensure the consistency of various pieces of dynamic data that are associated with the DSM subsystem and maintained by the **dsmd** daemon on behalf of application processes. The frequency of these audits varies based on the nature of the data being audited (e.g., lock records are audited more frequently than other pieces of dynamic data in an effort to identify leftover locks as quick as possible). While this auditing mechanism is deemed as essential for correct operation of the DSM subsystem in unsupervised mode, users can still disable/re-enable this mechanism by sending a SIGUSR2 signal to the **dsmd** daemon process. Each time this signal is received, **dsmd** will toggle a software flag that controls whether automatic audits should be performed. By default, this flag is enabled. SIGUSR1 signal, on the other hand, causes the **dsmd** daemon to display the current values of its operational parameters. Both signals are instrumental for debugging purposes only; therefore, should not be used under normal circumstances.*

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.14, **netd** on page 6-24

6.2.10 logd

NAME

logd Starts the Log process (LOG_MNGR).

SYNOPSIS

logd [-d] [-a|h|o] [-f logfile]

DESCRIPTION

logd Activates the LOG_MNGR process which outputs the messages logged from NewNet Distributed7 processes that have had the message logging capability activated with *ebs_log* or *spm_log()*. It is registered exclusively as a daemon object to the NewNet Distributed7 environment on the local host. By default, ***logd*** outputs the messages to the standard ouput, but can also send them to a file.

The information displayed for each message includes a message identifier, a date/time stamp, source and destination addresses, message type, message size, priority-level, and contents of the data portion of the message. The destination address information is *always* expressed in the L_IPCKEY format. The message size field contains information about both the overall message size and the size of the data portion of the message.

-d Displays additional information. An extra line of output is displayed which shows the origination point where message logging for a particular message took place, i.e., host and STREAMS multiplexer.

-a	Displays message contents in ASCII format.
-h	Displays message contents in hexadecimal format (default).
-o	Displays message contents in octal format.
-f logfile	Displays information to the standard output and also stores it in the file named ***logfile***. Without this option, ***logd*** will only display information to the standard output. If ***logd*** is initially invoked with this option, output to the file can be enabled/disabled by sending a SIGUSR2 signal to the ***logd*** process. Each time this signal is received, ***logd*** will toggle its current log status.

Related Information

- Section 7.2.7, **ebs_log** on page 7-13
- spm_log() from the SS8 SGC API Reference Manual

6.2.11 mlogd

NAME

mlogd Starts master event log daemon.

SYNOPSIS

mlogd [-c*|****s****|****x] [-v] [-d*** dir***] [-m*** msize***] [-a*** asize***]***

DESCRIPTION

mlogd	Starts the ***mlogd*** daemon which collects and processes log messages generated by system and application processes, messages are generated using the APM API library log macros. The ***mlogd*** daemon is normally started by the *apmd* daemon, but it could be started from the command line.
-c	Starts the AccessCRP version of ***mlogd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values prior to program execution.
-s	Starts the AccessSERVICES version of ***mlogd***. The ***$DOMID*** environment variable must be set to an appropriate value prior to program execution.
-x	Starts the normal NewNet Distributed7 version of ***mlogd***. The ***$EBSHOME*** environment variable must be set to an appropriate value prior to program execution.
-o	This option is used to overwrite the time stamp information contained in the log message received. By default, the time stamp information is populated when the log message is submitted, i.e., at the point of origination. ***mlogd*** uses this information as is

-v Runs ***mlogd*** in *verbose* mode. This mode causes ***mlogd*** to print the specifics of log messages received from other processes on the system console.

-d dir Stores master/alternative log files on specified host machines. By default, master/alternative log files are stored under ***mlog*** and ***alog*** directories, respectively, in the ***$EBSHOME/access/RUN*** directory. If the ***dir*** is specified, the master/alternative log files are stored under the ***dir/mlog*** and ***dir/alog*** directories, respectively. If ***dir*** directory (or one of its sub-directories) does not exist, ***mlogd*** will make an attempt to create all necessary directories.

-m msize Allows ***mlog*** directory to grow to ***msize*** kilobytes in size. By default, ***mlogd*** allows ***mlog*** directory to grow to 8192 kilobytes (i.e., 8 megabytes) in size before cleaning up dated ***mlog*** files.

-a asize Allows ***alog*** directory to grow to ***asize*** kilobytes in size. By default, ***mlogd*** allows ***alog*** directory to grow to 8192 kilobytes (i.e., 8 megabytes) in size before cleaning up dated ***alog*** files.

The ***mlogd*** daemon also supports multiple versions of *apmd*. If the user does not explicitly specify one in the command, then ***mlogd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

The ***mlogd*** daemon is normally started by the *apmd* daemon as the very first process. Only if it is the first process can it collect and process ALL log messages generated by processes that are subsequently started. The ***mlogd*** analyzes the log options in the incoming log messages to send them to the specified destinations. It also formats each message. The destinations may include the master or alternate log files on the local host, the system console, the local printer, the local alarm handler daemon, and/or MMI agents such as MML and GUI. Log messages that carry the **E_MLog** and/or **E_ALog** options are stored in local log files permanently.

FILES

The master log files generated and maintained by ***mlogd*** are named according to the following convention:

$EBSHOME/access/RUN/mlog/MLog.mmddyy

The alternate [secondary] log files generated and maintained by ***mlogd*** are named as follows:

$EBSHOME/access/RUN/alog/ALog.mmddyy

The **mmddyy** string is the current month, day, and year, each expressed in a two-digit numeric format.

NOTES

mlogd will delete the oldest log file in the master log directory when the size of that directory reaches 8Mbytes to prevent excessive accumulation of log files on the system. Similarly, it will delete the oldest log file in the alternate log directory when the size of that directory reaches 4Mbytes. Users can overwrite these default settings by using the ***-m*** and/or ***-a*** command line options at start-up. For example, to accommodate 32MB of storage space for master log files and 16MB of storage space for alternate log files, one must invoke ***mlogd*** as follows:

mlogd -m 32768 -a 16384

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.3.1, **apmconfig** on page 6-30

6.2.12 mml

NAME

mml Starts man-machine language interface

SYNOPSIS

mml [-d] [-f filename] sp

DESCRIPTION

mml Allows node/configuration management tasks on a user-specified signalling point (SP) using a customized version of the Man-Machine Language (MML) interface language. instead of the graphical user interface). When an ***mml*** session is started, the following prompt will be displayed by default: `MML_TH>`. The terminal handler prompts may be customized by setting the MML_PROMPT environment variable.

While in operation, ***mml*** will be registered exclusively with the NewNet Distributed7 environment on the local host machine as a daemon object, under the name MML_MNGR(sp) (a macro defined in the <***api_macro.h***> header file).

OPTIONS

-d	Disable ***ksh*** technique that allows use of special keys (e.g., arrow keys) to recall previously entered MML commands at MML prompt line.
-f filename	The batch process MML commands included in filename prior to returning control to user at the prompt line.
sp	The logical signalling point number used with *upmd*.

FILES

$EBSHOME/access/RUN/config/MML/params.txt

$EBSHOME/access/RUN/config/MML/help.text

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.2, **AccessMOB (mob)** on page 6-5
- Section 6.2.13, **mmi** on page 6-23

6.2.13 mmi

NAME

mmi Starts man-machine language interface

SYNOPSIS

mmi [-d] [-f filename]

DESCRIPTION

mmi Allows node/configuration management tasks on a user-specified signalling point (SP) using a customized version of the Man-Machine Language (MML) interface language. instead of the graphical user interface). ***mmi*** is a restricted version of *mml* that provides the same set of capabilities for managed objects that are not associated with a specific signalling point. Thus, one cannot use the ***mmi*** utility to perform SS7-related node/configuration management tasks. When an ***mmi*** session is started, the following prompt will be displayed by default: **`MMI_TH>.`** The terminal handler prompts may be customized by setting the MMI_PROMPT environment variable.

While in operation, ***mmi*** will be registered exclusively with the NewNet Distributed7 environment on the local host machine as a daemon object, under the name MMI_MNGR (a macro defined in the <***api_macro.h***> header file).

OPTIONS

-d Disable ***ksh*** technique that allows use of special keys (e.g., arrow keys) to recall previously entered MMI commands at MMI prompt line.

-f filename The batch process MMI commands included in filename prior to returning control to user at the prompt line.

FILES

$EBSHOME/access/RUN/config/MMI/params.txt

$EBSHOME/access/RUN/config/MMI/help.text

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.2, **AccessMOB (mob)** on page 6-5

• Section 6.2.12, **mml** on page 6-22

6.2.14 netd

NAME

netd Starts TCP/IP network daemon.

SYNOPSIS

$EBSHOME/access/bin/netd [-d|n] [-i] [-s]

DESCRIPTION

netd

Starts the ***netd*** daemon process, which sets up and maintains the STREAMS-based kernel-level interfaces from the local host's SPM to the SPMs of remote hosts in a distributed NewNet Distributed7 environment. The ***netd*** daemon process registers exclusively with the NewNet Distributed7 environment on the local host as the daemon object, NET_MNGR. This daemon process is required only for distributed configurations.

The kernel-level interface is built upon revision 1.5 of the Transport Provider Interface (TPI) specifications, and utilizes the connection-oriented TCP/IP protocol suite. This interface allows the processes that operate in the distributed network to exchange inter-machine messages in a reliable and high-performance manner.

This daemon is responsible for controlling the operations of distribution related NTWK, HOST, and TCPCON managed objects. The distributed processing environment, such as remote hosts and the TCP/IP connections between hosts, is defined by these managed objects. A distributed configuration can be a single TCP/IP connection between two hosts, or a maximum of eight host machines and/or redundant LAN configurations in which dual TCP/IP connections are set up between each and every host included in the network.

The ***netd*** daemon can be started automatically by the *apmd* daemon at system start-up time, provided that there is an entry in the ***$EBSHOME/access/RUN/config/PMGR/apmconfig*** file. Alternatively, it can be started manually from the command line.

The EBSHOME environment variable must be set before invoking this daemon as it makes use of this variable to locate various configuration files.

-d

This option is used to disable the built-in LAN interface testing feature of the ***netd*** daemon. When disabled, NewNet Distributed7 software cannot effectively detect and/or act upon failures, such as those resulting from disconnected or cut LAN cables. By default, this feature is enabled.

i	This function indicates that the Solaris IP Network Multipathing (IPMP) feature is in use; therefore, operations that involve LAN interfaces will need to be closely coordinated between Distributed7 and IPMP software (e.g., detection of failure or repair of a failed LAN interface). The IPMP feature was introduced in Solaris 8 OE update 2 (10/00) release, and it enables a host machine to have multiple network ports connected to the same subnet. This capability coupled with multiple network connections per subnet provide a host with one or both of the following advantages: [1] resilience from network adapter failure; [2] increased data throughput for outbound traffic. Since IPMP feature is intended to provide fully transparent recovery at TCP layer, when this option is specified, Distributed7 will not make any attempts to recover from single LAN interface failures (i.e., since IPMP software is expected to be in charge of "failover" and "failback" operations). Note that one may need to fine tune the duration of "failure detection time" used by IPMP software and/or TCP/IP heartbeat interval in use by Distributed7 software across individual TCP/IP connections to ensure no heartbeat failure would occur during so-called "failover blackout" periods. This can be achieved by taking one of the following approaches: [1] reducing "failure detection time" value in use by IPMP software to a value less than five times the length of TCP/IP heartbeat interval in use; [2] increasing the length of TCP/IP heartbeat interval in use to be more than one fifth of the "failure detection time" value in use by IPMP software.
-n	This option is used to enable optional ***ndd*** checks in an effort to detect dual-LAN cable disconnects more reliably and quickly. By default, NewNet Distributed7 relies on “cable disconnected/problem” messages reported through the STREAMS log to detect LAN cable disconnects. There are times, however, when dual-LAN cable disconnects are neither reliably nor quickly reflected in the log messages. This option instructs the ***netd*** daemon to conduct an additional set of checks using the ***ndd*** utility to detect dual-LAN cable disconnects more reliably and quickly. Note that this option can be used only for newer type LAN interfaces, such as /dev/hme or /dev/qfe, and does not support older interface types, such as /dev/1e.
-s	This option allows ***netd*** to execute in stand-alone mode. In this mode of operation, ***netd*** bars TCP/IP connections to remote hosts.

FILES

$EBSHOME/access/RUN/DBfiles/net_ntwk.DB

$EBSHOME/access/RUN/DBfiles/net_host.DB

$EBSHOME/access/RUN/DBfiles/net_tcpcon.DB

Related Information

- ebs_sync
- Section 7.2.1, **ebs_audit** on page 7-3

6.2.15 scmd

NAME

scmd Initializes the Signalling Connection Control Part (SCCP) multiplexer.

SYNOPSIS

scmd sp#

DESCRIPTION

scmd Starts the SS7 daemon process, ***scmd***, which initializes and administers the SCCP for a signalling point on the local system.

sp# Identifies the logical signalling point number of the node on the system to be configured; should be the same one used with the related *upmd* command.

The ***scmd*** daemon registers exclusively with the NewNet Distributed7 environment on the local host machine as the daemon object, SCM_MNGR(*sp#*). An ***scmd*** instance for each signalling point can exist on a host.

The ***scmd*** daemon initializes the SCCP multiplexer, links it to the UPM, and downloads it with the configuration information from the configuration database files located in the ***$EBSHOME/access/RUN\fI***<sp#>***\fP/DBfiles\fR*** directory. If no pre-configured data exists in the database, the following warning message occurs:

There is no spc in the database

The SCCP multiplexer provides routing control, connectionless control, connection-oriented control, and management functions required by the SCCP protocol layer.

The ***scmd*** daemon for a particular ***sp#*** can be started automatically by the *apmd* by making an entry in the associated configuration file (such as *apmconfig*). It can also be started from the command line.

Important*: The upmd process must be running before executing this command. Use ebs_ps to confirm that they exist.*

NOTES

The ***scmd*** daemon requires the ***upmd*** daemon for the corresponding signalling point to be up and running. It also requires the MTP protocol for that signalling point to be configured using the NewNet Distributed7 Managed Object interface, i.e., by manipulating the MTP managed object parameters. If the MTP protocol is not configured when the ***scmd*** daemon is started, ***scmd*** will suspend its execution indefinitely until this operation is completed before it proceeds with its normal start-up procedures. Note that under such circumstances, it will not be possible to see the ***scmd*** entry in the ***ebs_ps*** listing until the MTP protocol is configured.

FILES

$EBSHOME/access/RUN<sp#>***/DBfiles/snsp.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/subsys.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/cpc.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mate.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/gtentry.DB***

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.17, **upmd** on page 6-28

6.2.16 spmd

NAME

spmd Starts the NewNet Distributed7 infrastructure.

SYNOPSIS

spmd

DESCRIPTION

spmd Initializes and sets up the foundation of the multi-layered STREAMS architecture required by the NewNet Distributed7 infrastructure.

The ***spmd*** daemon configures the device drivers, TRMOD module, and Service Provider Module (SPM, also known as the signalling point multiplexer). The ***spmd*** daemon registers exclusively with the NewNet Distributed7 environment on the local host as a daemon object called SPM_MNGR.

On start-up, the ***spmd*** daemon first opens and initializes the SPM multiplexer to establish soft links to the SS7 signalling board device drivers (sbs334, pci334, pci3xpq, pci3xapq, cpc3xpq, pmc8260 and artic8260) as they are configured using the MMI/NMI interfaces. The ***spmd*** also assumes the responsibility of removing these links during system software shutdown.

The ***spmd*** daemon can be started automatically by the *apmd* process at system start-up time provided that there is an entry in the ***$EBSHOME/access/RUN/config/PMGR/apmconfig*** file. It can also be started from the command line.

Note: *The apmd process is now responsible for process management.*

FILES

$EBSHOME/access/RUN/DBfiles/spm_ss7board.DB
$EBSHOME/access/RUN/DBfiles/spm_class.DB
$EBSHOME/access/RUN/DBfiles/spm_port.DB
$EBSHOME/access/drv/sal.*.rel
$EBSHOME/access/drv/mtpl2.*.rel

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.17, **upmd** on page 6-28
- Section 6.2.14, **netd** on page 6-24
- Section 7.3.8, **apm_start** on page 7-49
- Section 6.2.15, **scmd** on page 6-26

6.2.17 upmd

NAME

upmd Sets up the User Part Multiplexer.

SYNOPSIS

upmd sp#

DESCRIPTION

upmd Sets up and configures the User Part Multiplexer (UPM) for a particular Signalling Point (sp) under the local NewNet Distributed7 environment. The primary responsibility of the UPM is to perform various Signalling Message Handling (SMH) functions and Signalling Network Management (SNM) functions required by the Message Transfer Part (MTP) Layer 3 protocol.

sp# Signalling point number of the logical node. It is usually 0 for a single or first node, but can be 1, 2, 3, 4, 5, 6, or 7. When the INE feature is being used, up to eight logical nodes can be started and configured, but all must have different signalling point numbers.

On start-up, the ***upmd*** daemon opens and initializes the appropriate UPM multiplexer and links it to the bottom of the previously initialized SPM.

After start-up, ***upmd*** interacts with the other ***upmd*** instances across the network, if there are any, in an effort to synchronize its local database files so that the signalling points on individual machines start from the same copy of the database.

The ***upmd*** subsequently continues its life by monitoring various events associated with the corresponding UPM, handling MMI/NMI requests that may be initiated by the users, and taking appropriate actions.

While in operation, the ***upmd*** daemon on each host registers exclusively with the NewNet Distributed7 environment on that host as the daemon object, UPM_MNGR(*sp#*) (a macro defined in <***api_macro.h***>). A ***upmd*** instance for each signalling point can exist on a host.

The ***upmd*** daemon for a particular ***sp#*** can be started automatically by the *apmd* by making an entry in the associated configuration file (such as *apmconfig*). It can also be started from the command line.

Important*: The* ***$EBSHOME*** *environment variable must be set before invoking this daemon because the variable is used to locate the MTP database files.*

FILES

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_l3timer.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_link.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_linkstat.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_lset.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_lsetstat.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_route.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_rsidx.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_rtset.DB***

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_sltimer.DB***

$EBSHOME/access/RUN<sp#>/Dbfiles/mtp_alias.DB

$EBSHOME/access/RUN<sp#>***/DBfiles/mtp_sp.DB***

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 6.2.15, **scmd** on page 6-26

6.3 ConfigurationFiles

6.3.1 apmconfig

NAME

apm_config apmd configuration file

SYNOPSIS

$EBSHOME/access/RUN/config/PMGR/apmconfig

DESCRIPTION

tag:estate:action:sstate:fstate:hstate:astate:nstate:acktime:retrycnt:retryint: retrydel:hbeatint:progid:groupid:priclass:clparams:dirpath:process

Configuration file which controls the operations of the *apmd* daemon process. It is composed of individual entries with the position-dependent syntax defined in the synopsis. The fields of the entry must be separated by a **:** character, with no white space in the line. Each entry is delimited by a *newline*. A maximum of 512 characters are permitted for each entry. Comments may be inserted as separate lines using the convention for comments described in *sh(1)*. A sample of the default file is listed on page 6-38.

Important*: The first statement in the file must be* ***version=1.0.0*** *or version 1.2.0. The apmd process must know that the entries in the file follow Release 1.x.y conventions.*

When creating and/or modifying the file, the following guidelines should be observed:

- Each line must contain the correct number of fields and must be formatted correctly. If a line is incorrect, then *apmd* ignores the entries in the file. However, it logs the line number at which an error or inconsistency was encountered.
- The UNIX path names for the executables are valid and correct.
- The start-up and steady-state information provided for individual entries must not conflict with other entries, which could cause undesired loops in processing.
  The apmd process is as intelligent as the logic provided in the apmconfig file!
- Shell scripts which are invoked from an entry must have a statement in the first line that specifies which shell version to be invoked (e.g., #!/bin/sh to invoke plain UNIX shell).
- After making any changes to the contents of the file while the system is running, you must issue the *apm_update* command to notify the *apmd* daemon to re-read and execute the file.

There are no limits on the number of entries. The entry fields are defined as follows:

tag An alphanumeric string used to uniquely identify a process in the **id@host** format when operating in a network of hosts. The **id** is an alphanumeric string that uniquely identifies the process on the host machine identified by **host**. The **@host** portion may be omitted for processes on the local host because the system will automatically substitute this information. The total size of the **id@host** string cannot exceed 24 characters.

estate The state(s) that the *apmd* must be in for this entry to be executed. If the *apmd*'s run state while executing this file is among the states specified in this field, the entry is executed. Otherwise, it is ignored. A maximum of 16 execution states may be specified for an individual entry. Multiple execution states must be separated from each other with only the **|** character - no *white space* should exist. If no states are defined for **estate**, then the entry will be executed every time *apmd* executes the file. Execution states are defined by the developer and may have names that are up to 4 alphanumeric characters long.

action The action(s) that *apmd* should take on the process identified in the process field. Several key words exist that are recognized by *apmd*. Actions are only taken if *apmd*'s run state matches a state in the **estate** field. Valid actions are:

- **initdefault**: An entry with this action is scanned only once when *apmd* is initially invoked. This entry is used to determine the initial run state for *apmd*. This initial run state is set to the first execution state specified in the **estate** field of this entry. If the ***apmconfig*** file does not contain an entry with the *initdefault* action type, *apmd* determines its initial run state as follows:
  for the AccessSERVICES version, state **A**
  for the AccessCRP version, state **D**
  for the basic NewNet Distributed7 version, state **init**
- **once**: If the process named in the **process** field is not running, *apmd* should start it and proceed to the state specified in appropriate state field. The *apmd* should not wait for the process's termination and it should not restart the process if the process terminates. After starting the process, *apmd* will go to the state specified in one of the following *state* fields. The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.
  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or empty)
  nstate - 1) negative acknowledgment was received from process
  2) no acknowledgment was received within the time specified in **acktime**
  sstate - 1) process terminates with an exit code of 0, at any time
  fstate - 1) process terminates with a non-zero exit code, at any time
  2) process is killed by a signal it could not handle

- **auto**: If the process named in the **process** field is not running, *apmd* should start it and proceed to the state specified in appropriate state field. The *apmd* should not wait for the process's termination. From then on, provided the process terminates its execution within **retrydel** seconds, restart the process at periodic intervals as specified via the **retrydel** setting and repeat this pattern forever. A missing or zero value of **retrydel** makes this action type equivalent to "once" and disables the periodic start-up capability. Similarly, at any point in time, if the process fails to terminate its execution within **retrydel** seconds, the periodic start-up capability is automatically disabled. After starting the process, *apmd* will go to the state specified in one of the following *state* fields. The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.

  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or empty)

  nstate - 1) negative acknowledgment was received from process
  2) no acknowledgment was received within the time specified in **acktime**

  sstate - 1) process terminates with an exit code of 0, at any time

  fstate - 1) process terminates with a non-zero exit code, at any time
  2) process is killed by a signal it could not handle
  The exit code of the process at termination time has no impact on the periodic start-up capability. As long as the process somehow manages to terminate its execution within **retrydel** seconds, an attempt will be made by *apmd* to re-start it at the end of **retrydel** seconds.

- **failsafe**: If the process named in the **process** field is not running, *apmd* should start it and proceed to the state specified in appropriate state field. The *apmd* should not wait for the process's termination but it should restart the process if start-up fails (fstate). The process should not be restarted if it exits with a zero exit code. After initially starting the process, *apmd* will go to the state specified in one of the following *state* fields. The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.

  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or

empty)
nstate - 1) negative acknowledgment was received from process
2) no acknowledgment was received within the time
specified in **acktime**
sstate - 1) process terminates with an exit code of 0, at any time
(do not restart process)
fstate - 1) process terminates with a non-zero exit code (at any
time)
2) process is killed by a signal it could not handle
The *apmd* should keep attempting to restart the process
until
it starts or until the number of attempts that occur in
retryint seconds exceeds **retrycnt**. A delay of **retrydel**
seconds should occur between attempts.
hstate - 1) number of attempts to restart process in the last
retrydel
seconds has exceeded the value in **retrycnt**
The cycle of restart attempts described in *fstate* should not
begin for 60 seconds.

• **respawn**: If the process named in the **process** field is not running, *apmd* should start it and proceed to the state specified in the appropriate state field. The *apmd* should not wait for the process's termination but it should restart the process if start-up fails (fstate). The process should not be restarted if it exits with a zero exit code. After initially starting the process, *apmd* will go to the state specified in one of the following *state* fields. The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.
astate - 1) **acktime** field holds a non-zero value and a positive
acknowledgment was received from the process
2) no acknowledgment is expected (**acktime** is 0 or
empty)
nstate - 1) negative acknowledgment was received from process
2) no acknowledgment was received within the time
specified in **acktime**
sstate - 1) process terminates with an exit code of 0, at any time
(do not restart process)
fstate - 1) process terminates with a non-zero exit code (at any
time)
2) process is killed by a signal it could not handle
The *apmd* should keep attempting to restart the process
until
it starts or until the number of attempts that occur in
retryint seconds exceeds **retrycnt**. A delay of **retrydel**

seconds should occur between attempts.

hstate - retrydel 1) number of attempts to restart process in the last seconds has exceeded the value in **retrycnt**
No further attempts to restart the process will be made.

- **wait**: If the process named in the **process** field is not running, *apmd* should start it and ***WAIT FOR IT TO TERMINATE***. After the process terminates, it can proceed to the state specified in the appropriate state field. (No other activities will occur until the process terminates.) The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.
  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or empty)
  nstate - 1) negative acknowledgment was received from process
  2) no acknowledgment was received within the time specified in **acktime**
  sstate - 1) process terminates with an exit code of 0, at any time
  fstate - 1) process terminates with a non-zero exit code, at any time
  2) process is killed by a signal it could not handle
- **off**: If the process named in the **process** field is currently running, generate a SIGTERM signal to terminate it. If the process does not terminate within the next 3 seconds, send a SIGKILL signal to terminate it. Then, switch to state specified in **sstate** field. If the process is not running, ignore this entry.
- **setstate**: Change the current run state of *apmd* to the state specified in the **sstate** field.

sstate The success state that *apmd* will be set to when the process terminates with a zero exit code. Two separate states, start-up and steady-state, may be defined in this field. The start-up success state is only used when *apmd* starts up, i.e., the ***apmconfig*** file is being executed for the first time. From then on, only the steady-state success state is used.
Each state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. The first state in the field is the start-up state, followed by the steady-state. The two success states must be separated from each other by the **-** character, with no white space. A single state in the field implies both the start-up state and steady-state are the same. Either or both fields may be set to *don't care*. To set the start-up state to *don't care*, this field must contain the **-** character followed by the steady-state success state. To set the steady-state success state to *don't care*, this field must contain the start-up state followed by the **-** character.

If this field is empty, then both start-up and steady-state success states are *don't care.*

fstate The failure state that *apmd* will be set to when the process terminates with a non-zero exit code or the process terminates because of a signal it could not handle. Two separate states, start-up and steady-state, may be defined in this field. The start-up failure state is only used for failures when *apmd* initially starts up, i.e., first time ***apmconfig*** file is executed. From then on, only the steady-state failure state is used.

Each state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. The first state in the field is the start-up state, followed by the steady-state. The two failure states must be separated from each other by the **-** character, with no white space. A single state in the field implies both the start-up state and steady-state are the same. Either or both fields may be set to *don't care*. To set the start-up state to *don't care*, this field must contain the **-** character followed by the steady-state failure state. To set the steady-state failure state to *don't care,* this field must contain the start-up state followed by the **-** character. If this field is empty, then both start-up and steady-state failure states are *don't care.*

hstate The hopeless state that *apmd* will be set to when **retrycnt** successive attempts to restart the process fail within the **retryint** interval. Two separate states, start-up and steady-state, may be defined in this field. The start-up hopeless state is only used when *apmd* initially starts up, i.e., first time ***apmconfig*** file is executed. From then on, only the steady-state hopeless state is used.

Each state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. The first state in the field is the start-up state, followed by the steady-state. The two hopeless states must be separated from each other by the **-** character, with no white space. A single state in the field implies both the start-up state and steady-state are the same. Either or both fields may be set to *don't care*. To set the start-up state to *don't care*, this field must contain the **-** character followed by the steady-state hopeless state. To set the steady-state hopeless state to *don't care*, this field must contain the start-up state followed by the **-** character. If this field is empty, then both start-up and steady-state hopeless states are *don't care.*

astate The positive acknowledgment state that *apmd* will be set to when a positive acknowledgment is received from the process during the acknowledgment interval or when no acknowledgment message is expected. Two separate states, start-up and steady-state, may be defined in this field. The start-up state is only used when *apmd* initially starts up, i.e., first time ***apmconfig*** file is executed. From then on, only the steady-state acknowledgment state is used.

Each state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. The first state in the field is the start-up state, followed by the steady-state. The two states must be separated from each other by the **-** character, with no white space. A single state in the field implies both the start-up state and steady-state are the same. Either or both fields may be set to *don't care*. To set the start-up state to *don't care*, this field must contain the **-** character followed by the steady-state acknowledgment state. To set the steady-state acknowledgment state to *don't care*, this field must contain the start-up state followed by the **-** character. If this field is empty, then both start-up and steady-state acknowledgment states are *don't care*.

nstate The negative acknowledgment state that *apmd* will be set to when a negative acknowledgment is received from the process or when the acknowledgment interval expires without an acknowledgment being received. Two separate states, start-up and steady-state, may be defined in this field. The start-up state is only used when *apmd* initially starts up, i.e., first time ***apmconfig*** file is executed. From then on, only the steady-state negative acknowledgment state is used.

Each state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. The first state in the field is the start-up state, followed by the steady-state. The two states must be separated from each other by the **-** character, with no white space. A single state in the field implies both the start-up state and steady-state are the same. Either or both fields may be set to *don't care*. To set the start-up state to *don't care*, this field must contain the **-** character followed by the steady-state negative acknowledgment state. To set the steady-state negative acknowledgment state to *don't care*, this field must contain the start-up state followed by the **-** character. If this field is empty, then both the start-up state and steady-state are *don't care*.

acktime The time interval, in seconds, during which *apmd* should wait for a positive or negative acknowledgment from the process before moving to a run state. A value of 0 or an empty field means that *apmd* will automatically move to the run state specified in the **astate** field and will not wait for an acknowledgment from the process. A value of -1 means that *apmd* must wait indefinitely for an acknowledgment.

retrycnt The number of times *apmd* should try to restart the process within **retryint** seconds. Restarts are attempted only if the **action** field is *respawn* or *failsafe* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 3 is used.

retryint The time interval, in seconds, during which *apmd* should try to restart the process, up to **retrycnt** number of times. Restarts are attempted only if the **action** field is *respawn* or *failsafe* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 60 seconds is used.

retrydel The time delay, in seconds, that should occur between attempts to restart the process. Restarts are attempted only if the **action** field is *respawn* or *failsafe* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 1 second is used.

hbeatint The time interval, in seconds, that should exist between heartbeat request messages sent to the process by *apmd*. If no value or a value of 0 is specified in this field, the heartbeat feature is disabled for the process.
The heartbeat feature sends periodic heartbeat messages to the process if it is enabled. If the process fails to respond to 3 consecutive heartbeat request messages, apmd terminates the process by sending it a SIGKILL signal. Then, when the process terminates, the *apmd* will go to the state specified in the **fstate** field.

progid The program ID associated with the process. It can be an integer value between 0 and 255. If this field is empty, the default value of 0 will be assigned to the process. The program ID is used to identify program output in trace and log records. It should be the same as the one specified in the program using the *apm_init()* function call because the value in this field will overwrite the one specified by the function.

groupid The group ID associated with the process. Group IDs are non-negative integer values that define *functional* process groups. If this field is empty, *apmd*'s UNIX group ID is assigned to the process by default.
The group ID allows a group of processes to be targeted by the *apm_kill* function and *apm_kill()* utility. They can send UNIX signals to all the processes of a group at once instead of having to send to each process separately.

priclass The priority class associated with the process. When either of the **real-time** or **time-sharing** classes is selected, *apmd* daemon will invoke the ***priocntl*** system call to adjust the scheduling class and class-specific scheduling parameters of the process spawned as specified via ***priclass*** and ***clparams*** fields.
This ***priclass*** optional field allows one of the following scheduling classes to be selected by the process:

- **rt**
  The real-time scheduling class.
- **ts**
  The time-sharing scheduling class.

clparams The individual parameters associated with the ***priclass*** field. This optional field allows the individual parameters associated with a specific priority class (i.e., **real-time** or **time-sharing**) to be initialized. Multiple parameters must be separated from each other using the '|' character, with no white space left in between.

- When ***priclass*** is set to **real-time**, the ***clparams*** field must contain, in specified order, the rt_pri|rt_tqsecs|rt_tqnsecs parameter values.

• When ***priclass*** is set to **time-sharing**, the ***clparams*** field must contain, in specified order, the ts_uprilim|ts_upri parameter values.

dirpath The UNIX path name of the executable program. This field is used with the **process** field to fully identify the executable. Either a keyword (home or run), a full UNIX path name (starting with the **/** character) or a UNIX environment variable that is part of the current execution environment must be in this field. The keywords have the following meanings:

- **home**
  For AccessSERVICES version: ***$EBSHOME/access***
  For CRP version: ***$CRPDIR***
  For basic NewNet Distributed7 version: ***$EBSHOME/access***
- **run**
  For AccessSERVICES version: ***$EBSHOME/access/RUN***
  For CRP version: ***$APPHOME***
  For basic NewNet Distributed7 version: ***$EBSHOME/access/RUN***

process The relative path name of the executable program and its command-line arguments, if any. This field is used with the **dirpath** field to locate the UNIX path name for the executable program, the executable name, and its arguments. The maximum number of command-line arguments that can be specified is 32.
Input/output redirection is not currently supported.

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.16, **apm_update** on page 7-65
- Section 7.3.2, **apm_getstate** on page 7-37
- Section 7.3.7, **apm_setstate** on page 7-47
- Section 6.3.2, **apmconfig.old** on page 6-42
- priocntl

SAMPLE FILE

```
#
# apmconfig(4A) - configuration file for apmd(1A) daemon
#
# entries in this file must comply with the following format:
#
# tag:estate:action:sstate:fstate:hstate:astate:nstate: \
# acktime:retrycnt:retryint:retrydel:hbeatint:progid:groupid:dirpath:process
#
# where the individual fields are defined as follows:
#
#     tag       - process identifier tag [must be unique]
#     estate    - execution state(s)
#     action    - action type
#     sstate    - success state(s)
```



```
#     fstate     - failure state(s)
#     hstate     - hopeless state(s)
#     astate     - ack received state(s)
#     nstate     - nak received state(s)
#     acktime    - time to wait for ack/nak [in seconds]
#     retrycnt   - # times to respawn within `retryint` interval
#     retryint   - respawn interval [in seconds]
#     retrydel   - delay before a respawn attempt [in seconds]
#     hbeatint   - heartbeat interval [in seconds]
#     progid     - process program id
#     groupid    - process group id
#     priclass   - priority class
#     clparams   - parameters of priority class
#     dirpath    - directory at which the process is located
#     process    - executable name & arguments
#
# for more info, pls refer to the apmconfig(4A) man page.
#

# specify apmcfgfile(4A) version
version=1.0

# specify apmd(1A) start-up run state
is:init:initdefault:

# specify rules for daemons that must exist at all times
mlogd::failsafe::::::-1::::60:1::home:./bin/mlogd -x
spmd::failsafe::::::-1::::60:2::home:./bin/spmd
netd::failsafe::::::-1::::60:3::home:./bin/netd
alarmd::failsafe::::::-1::::60:4::home:./bin/alarmd
dsmd::failsafe::::::-1::::60:5::home:./bin/dsmd
dkmd::failsafe::::::-1::::60:6::home:./bin/dkmd -m dramod
# change apmd(1A) run state from "init" to "safe"
sc:init:setstate:safe-:

#
# start daemon processes associated with signalling point 0
# when a run state of "sp0u" is explicitly specified by the user
#
upmd0:sp0u:respawn::sp0d:sp0d::sp0d:-1::::60:7:100:home:./bin/upmd 0
scmd0:sp0u:respawn::sp0d:sp0d::sp0d:-1::::60:8:100:home:./bin/scmd 0

#
# terminate daemon processes associated with signalling point 0
# when a run state of "sp0d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd0:sp0d:off:::::::::::::home:./bin/scmd 0
upmd0:sp0d:off:::::::::::::home:./bin/upmd 0

#
# start daemon processes associated with signalling point 1
# when a run state of "sp1u" is explicitly specified by the user
```

```
#
upmd1:sp1u:respawn::sp1d:sp1d::sp1d:-1::::60:7:101:home:./bin/upmd 1
scmd1:sp1u:respawn::sp1d:sp1d::sp1d:-1::::60:8:101:home:./bin/scmd 1

#
# terminate daemon processes associated with signalling point 1
# when a run state of "sp1d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd1:sp1d:off:::::::::::::home:./bin/scmd 1
upmd1:sp1d:off:::::::::::::home:./bin/upmd 1

#
# start daemon processes associated with signalling point 2
# when a run state of "sp2u" is explicitly specified by the user
#
upmd2:sp2u:respawn::sp2d:sp2d::sp2d:-1::::60:7:102:home:./bin/upmd 2
scmd2:sp2u:respawn::sp2d:sp2d::sp2d:-1::::60:8:102:home:./bin/scmd 2

#
# terminate daemon processes associated with signalling point 2
# when a run state of "sp2d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd2:sp2d:off:::::::::::::home:./bin/scmd 2
upmd2:sp2d:off:::::::::::::home:./bin/upmd 2

#
# start daemon processes associated with signalling point 3
# when a run state of "sp3u" is explicitly specified by the user
#
upmd3:sp3u:respawn::sp3d:sp3d::sp3d:-1::::60:7:103:home:./bin/upmd 3
scmd3:sp3u:respawn::sp3d:sp3d::sp3d:-1::::60:8:103:home:./bin/scmd 3

#
# terminate daemon processes associated with signalling point 3
# when a run state of "sp3d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd3:sp3d:off:::::::::::::home:./bin/scmd 3
upmd3:sp3d:off:::::::::::::home:./bin/upmd 3

#
# start daemon processes associated with signalling point 4
# when a run state of "sp4u" is explicitly specified by the user
#
upmd4:sp4u:respawn::sp4d:sp4d::sp4d:-1::::60:7:104:home:./bin/upmd 4
scmd4:sp4u:respawn::sp4d:sp4d::sp4d:-1::::60:8:104:home:./bin/scmd 4

#
# terminate daemon processes associated with signalling point 4
# when a run state of "sp4d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
```



```
#
scmd4:sp4d:off:::::::::::::home:./bin/scmd 4
upmd4:sp4d:off:::::::::::::home:./bin/upmd 4

#
# start daemon processes associated with signalling point 5
# when a run state of "sp5u" is explicitly specified by the user
#
upmd5:sp5u:respawn::sp5d:sp5d::sp5d:-1::::60:7:105:home:./bin/upmd 5
scmd5:sp5u:respawn::sp5d:sp5d::sp5d:-1::::60:8:105:home:./bin/scmd 5

#
# terminate daemon processes associated with signalling point 5
# when a run state of "sp5d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd5:sp5d:off:::::::::::::home:./bin/scmd 5
upmd5:sp5d:off:::::::::::::home:./bin/upmd 5

#
# start daemon processes associated with signalling point 6
# when a run state of "sp6u" is explicitly specified by the user
#
upmd6:sp6u:respawn::sp6d:sp6d::sp6d:-1::::60:7:106:home:./bin/upmd 6
scmd6:sp6u:respawn::sp6d:sp6d::sp6d:-1::::60:8:106:home:./bin/scmd 6

#
# terminate daemon processes associated with signalling point 6
# when a run state of "sp6d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd6:sp6d:off:::::::::::::home:./bin/scmd 6
upmd6:sp6d:off:::::::::::::home:./bin/upmd 6

#
# start daemon processes associated with signalling point 7
# when a run state of "sp7u" is explicitly specified by the user
#
upmd7:sp7u:respawn::sp7d:sp7d::sp7d:-1::::60:6:107:home:./bin/upmd 7
scmd7:sp7u:respawn::sp7d:sp7d::sp7d:-1::::60:8:107:home:./bin/scmd 7

#
# terminate daemon processes associated with signalling point 7
# when a run state of "sp7d" is explicitly specified by the user
# and/or a failure is encountered during the start-up phase
#
scmd7:sp7d:off:::::::::::::home:./bin/scmd 7
upmd7:sp7d:off:::::::::::::home:./bin/upmd 7

# add application specific entries below this line
```

6.3.2 apmconfig.old

NAME

apmconfig.old *apmd* configuration file for previous versions

SYNOPSIS

$EBSHOME/access/RUN/config/PMGR/apmconfig.old

DESCRIPTION

**id:dirpath:execname:action:estate:hbeatind:sstate:fstate:hstate:astate:nstate:
retrycnt:retryint:retrydel:acktime:args**

Previous version of the configuration file which controls the operations of the *apmd* daemon process. It is composed of individual entries with the position-dependent syntax defined in the synopsis. The fields of the entry must be separated by a **:** character, with no white space in the line. Each entry is delimited by a *newline*. A maximum of 512 characters are permitted for each entry. Comments may be inserted as separate lines using the convention for comments described in *sh(1)*.

Important*: Beginning with NewNet Distributed7, the apmconfig file should be used. The format in the* ***apmconfig.old*** *file has only been preserved to provide backward compatibility in this release. There is no guarantee that it will be supported in future releases of the product.*

When creating and/or modifying the file, the following guidelines should be observed:

- Each line must contain the correct number of fields and must be formatted correctly. If a line is incorrect, apmd will ignore the entries in the file. However, it will log the line number at which an error or inconsistency was encountered.
- The UNIX path names for the executables are valid and correct.
- The start-up and steady-state information provided for individual entries must not conflict with other entries, which could cause undesired loops in processing.
  The apmd process is as intelligent as the logic provided in the apmconfig file!
- Shell scripts which are invoked from an entry must have a statement in the first line that specifies which shell version to be invoked (e.g., #!/bin/sh to invoke plain UNIX shell).
- After making any changes to the contents of the file while the system is running, you must issue the *apm_update* command to notify the *apmd* daemon to re-read and execute the file.

There are no limits on the number of entries. The entry fields are defined as follows:

id An alphanumeric string used to uniquely identify an entry. The contents of this field are appended to the local host name to form the identity in the **id@host** format. The total size of the **id@host** string cannot exceed 24 characters.

dirpath The UNIX path name of the executable program. This field is used with the **execname** field to fully identify the executable. Either a keyword (home or run) or a full UNIX path name (starting with the / character) must be in this field. The keywords have the following meanings:

- **home**
  For AccessSERVICES version: ***$EBSHOME/access***
  For CRP version: ***$CRPDIR***
  For basic NewNet Distributed7 version: ***$EBSHOME/access***
- **run**
  For AccessSERVICES version: ***$EBSHOME/access/RUN***
  For CRP version: ***$APPHOME***
  For basic NewNet Distributed7 version: ***$EBSHOME/access/RUN***

execname The relative path name and file name of the executable program. This field is used with the **dirpath** field to locate the UNIX path name for the executable program and the executable name

action The action(s) that *apmd* should take on the process identified in the **process** field. Several key words exist that are recognized by *apmd*. Actions are only taken if *apmd*'s run state matches a state in the **estate** field. Valid actions are:

- **START**: If the process named in the **execname** field is not running, *apmd* should start it and not wait for the process's termination
- **KEEPALIVE**: If the process named in the **execname** field is not running, *apmd* should start it and proceed to the state specified in the appropriate state field. The *apmd* should not wait for the process's termination but it should restart the process if start-up fails. If the process terminates for any reason, the *apmd* should keep attempting to restart the process until it starts or until the number of attempts that occur in the last interval of **retryint** seconds exceeds **retrycnt**. A delay of **retrydel** seconds should occur between attempts.
  After initially starting the process, *apmd* will go to the state specified in one of the following *state* fields. The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.
  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or empty)
  nstate - 1) negative acknowledgment was received from process
  2) no acknowledgment was received within the time specified in **acktime**
  sstate - 1) process terminates with an exit code of 0, at any time
  fstate - 1) process terminates with a non-zero exit code (at any time)
  2) process is killed by a signal it could not handle
  hstate - 1) number of attempts to restart process in the last

retrydel

seconds has exceeded the value in **retrycnt**
No further attempts to restart the process will be made.

- **WA I T**: If the process named in the **process** field is not running, ***apmd*** should start it and wait for it to terminate. After the process terminates, it can proceed to the state specified in the appropriate state field. (No other activities will occur until the process terminates.) The *state* field it will access depends on the contents of **acktime** and actions of the process. Based on this combination of results, each state field should hold an appropriate *apmd* run state.
  astate - 1) **acktime** field holds a non-zero value and a positive acknowledgment was received from the process
  2) no acknowledgment is expected (**acktime** is 0 or empty)
  nstate - 1) negative acknowledgment was received from process
  2) no acknowledgment was received within the time specified in **acktime**
  sstate - 1) process terminates with an exit code of 0, at any time
  fstate - 1) process terminates with a non-zero exit code, at any time
  2) process is killed by a signal it could not handle
- **OFF**: If the process named in the **process** field is currently running, generate a SIGTERM signal to terminate it. If the process does not terminate within the next 3 seconds, send a SIGKILL signal to terminate it. Then, switch to state specified in the **sstate** field. If the process is not running, ignore this entry.
- **CHNGSTATE**: Change the current run state of *apmd* to the state specified in the **sstate** field.

estate The state(s) that the *apmd* must be in for this entry to be executed. If the *apmd*'s run state while executing this file is among the states specified in this field, the entry is executed. Otherwise, it is ignored. A maximum of 16 execution states may be specified for an individual entry. Multiple execution states must be separated from each other with a comman (**,**); no white space should exist. If no states are defined for **estate**, then the entry is executed every time *apmd* executes the file. Execution states are defined by the developer and may have names that are up to four alphanumeric characters long.

hbeatind The heartbeat indicator. A value in this field enables the heartbeat feature, which causes *apmd* to send heartbeat request messages to the process every 5 seconds. If the **()** characters are in this field, the heartbeat feature is disabled for the process.
The heartbeat feature sends periodic heartbeat messages to the process if it is enabled. If the process fails to respond to 3 consecutive heartbeat request messages, apmd terminates the process by sending it a SIGKILL

signal. Then, when the process terminates, the *apmd* will go to the state specified in the **fstate** field.
In the AccessCRP version, this field can also be used to specify the subsystem ID associated with the process by entering the non-negative integer subsystem ID.

sstate The success state that *apmd* will be set to when the process terminates with a zero exit code. The state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. If the field holds the \\ characters, then the state is set to *don't care*.

fstate The failure state that *apmd* will be set to when the process terminates with a non-zero exit code or the process terminates because of a signal it could not handle. The state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. If the field holds the \\ characters, then the state is set to *don't care*.

hstate The hopeless state that *apmd* will be set to when **retrycnt** successive attempts to restart the process fail within the **retryint** interval. The state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. If the field holds the \\ characters, then the state is set to *don't care*.

astate The positive acknowledgment state that *apmd* will be set to when a positive acknowledgment is received from the process during the acknowledgment interval or when no acknowledgment message is expected. The state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. If the field holds the \\ characters, then the state is set to *don't care*.

nstate The negative acknowledgment state that *apmd* will be set to when a negative acknowledgment is received from the process or when the acknowledgment interval expires without an acknowledgment being received. The state is defined by the developer and may have a name that is up to 4 alphanumeric characters long. If the field holds the \\ characters, then the state is set to *don't care*.

retrycnt The number of times *apmd* should try to restart the process within **retryint** seconds. Restarts are attempted only if the **action** field is *KEEPALIVE* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 3 is used.

retryint The time interval, in seconds, during which *apmd* should try to restart the process, up to **retrycnt** number of times. Restarts are attempted only if the **action** field is *KEEPALIVE* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 60 seconds is used.

retrydel The time delay, in seconds, that should occur between attempts to restart the process. Restarts are attempted only if the **action** field is *KEEPALIVE* and the process has terminated with a non-zero exit code or was killed by an unexpected signal. If this field is left blank, the default value of 1 second is used.

acktime The time interval, in seconds, during which *apmd* should wait for a positive or negative acknowledgment from the process before moving to a run state. A value of 0 or an empty field means that *apmd* will automatically move to the run state specified in the **astate** field and will not wait for an acknowledgment from the process. A value of -1 means that *apmd* must wait indefinitely for an acknowledgment.

args The command-line arguments, if any for the process. A maximum of 32 arguments can be specified.

Important*: Input/output redirection is not currently supported.*

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.16, **apm_update** on page 7-65
- Section 7.3.2, **apm_getstate** on page 7-37
- Section 7.3.7, **apm_setstate** on page 7-47
- Section 6.3.1, **apmconfig** on page 6-30

6.4 Signaling Gateway Processes

6.4.1 sctpd

NAME

sctpd ***SCTP (Stream Control Transmission Protocol) daemon***

SYNOPSIS

sctpd [-h] [-n | -s] [-p ipc port | -a ipc path] [-f cfg file]

DESCRIPTION

The sctp process provides SCTP service to its client processes on Solaris 8/9, such as an SGP or ASP. An SGP uses SCTP over IP to transport M3UA traffic (translated from SS7 messages) to the ASP in the IP domain, and vice versa.

SCTP is a reliable transport protocol operating on top of a connectionless packet network such as IP. It offers the fol lowing services to its users:

- acknowledged error-free non-duplicated transfer of user data
- data fragmentation to conform to discovered path MTP size
- sequenced delivery of user messages within multiple streams
- optional bundling of multiple user messages into a singla SCTP packet
- network-level fault tolerance through multi-homing support

There can be only one instance of SCTP running on each host. An SCTP daemon can be configured to use up to two IP addresses for multi-homing purposes. Its configuration parameters are all specified in the **`$SGHOME/sg/RUN/config/SCTP/sctpd.conf`** file.

-***h*** Provides help information about SCTP daemon.

-f cfg file Start SCTP daemon using the configuration file specified in file. Optionally, SCTP daemon can be started without the **-f** option so that default parameters are used. Please see the default file **`$SGHOME/sg/RUN/config/SCTP/sctpd.conf`** for default parameter values.

-a ipc path The socket path name used for IPC communication between the sctp daemon and SGPs and/or ASPs. **`/var/run/.sctpdSocket`** is the default if one is not specified.

The **-a** and **-p** options cannot be entered in the same command.

-p ipc port Port number used for IPC communication between sctp daemon and sgp/asp. Port 9000 is used by default if the **-p** option is not specified.

The **-a** and **-p** options cannot be entered in the same command.

-n Notify APM This option is required when sctp runs on the Distributed7 platform.

-s Silent mode, i.e. no logging. By default, sctp runs in silent mode.

FILES

SCTP startup configuration parameters are defined in the `$SGHOME/sg/RUN/config/SCTP/sctpd.conf` file. Edit this file to modify any parameters. After modification, send a HUP signal to sctpd so that it re-reads the configuration file, e.g. kill -HUP pid, where pid is the process ID of sctpd. (See Section 4.5.1.3, **SCTP Configuration** on page 4-13 for more information.)

CAUTION*: Never start sctpd manually. To be sure that process management is in effect, it should always be started using the sg_start command. Edit the apmconfig file in the* ***`$EBSHOME/access/RUN/config/PMGR/apmconfig`*** *directory to modify the default command line arguments.*

ENVIRONMENT

The $EBSHOME and $SGHOME environment variables must be set to the base installation directory of Distributed7 and Signaling Gateway software. These variables are set automatically when the user logs in as sgadm.

RELATED INFORMATION

- Section 6.3.1, **apmconfig** on page 6-30
- Section 6.4.2, **sgpd** on page 6-48
- Section 7.4.2, **sg_start** on page 7-67

6.4.2 sgpd

NAME

sgpd	Signaling Gateway Process daemon

SYNOPSIS

sgpd [-h] [-p ipc port | -a ipc path] [-f cfg file]

DESCRIPTION

SGP is a Signaling Gateway Process (SGP) running in a Signaling Gateway (SG). It is an element in which interworking between SS7 and IP networks take place. It processes traffic from the SS7 network, translates them into M3UA (MTP3 User Adaptation) format, and sends to the IP network over SCTP. Similarly, it receives traffic from the IP network via SCTP, translates them into SS7 MTP3 format and sends to the SS7 destination over SS7 links. There can be only one instance of SGP running on each host.

-h	Print help information.
-a ipc path	The socket path name used for IPC communication between the sctp daemon and SGPs and/or ASPs. /var/run/.sctpdSocket is the default if

one is not specified. The -a and -p options cannot be entered in the same command. This is valid on Solaris 8/9.

-p ipc port Port number used for IPC communication between sctp daemon and sgp/asp. Port 9000 is used by default if the -p option is not specified. The -a and -p options cannot be entered in the same command. This is valid on Solaris 8/9.

-f cfg file Specify path/file name of the configuration file which contains various timer parameters.

FILES

SGP configuration parameters are defined in the $SGHOME/sg/RUN/config/SGP/sgpd.conf file. Edit this file to modify any parameters. After modification, send a HUP signal to sgpp so that it re-reads the configuration file, e.g. kill -HUP pid where pid is the process ID of the sgpd.

Caution: *Never start sgpd manually. To be sure that process management is in effect, it should always be started using the sg_start command. Edit the apmconfig file in the $EBSHOME/access/RUN/config/PMGR/apmconfig directory to modify the default command line arguments.*

Caution: *The configuration options for sgpd are intended for debugging purposes* ***ONLY****. Modification of this file is not recommended without first getting technical assistance from NewNet Communications Technologies.*

ENVIRONMENT

The $EBSHOME and $SGHOME environment variables must be set to the base installation directory of NewNet Distributed7 and Signaling Gateway software. These variables are set automatically when the user logs in as sgadm.

RELATED INFORMATION

- Section 7.4.2, **sg_start** on page 7-67
- Section 6.3.1, **apmconfig** on page 6-30

Chapter 7: User Commands

7.1 ChapterOverview

This chapter describes the user commands that are used to maintain the Signaling Gateway software, and are summarized in the following table:

Table 7-1: User Command Summary

Command	Description
ebs_audit	Audits dynamic data.
ebs_brdfinfo	Get and clear SS7 board crash dump.
ebs_cfgbrd	Configure and unconfigure SS7 board.
ebs_config	Configure system operation mode.
ebs_dbconfig	Save or restore configuration data
ebs_dnlbrd	Reset and download SS7 board.
ebs_log	Activates and deactivates message logging capabilities for processes.
ebs_mngbrd	Start and stop SS7 board sanity checks.
ebs_modinstall	Installs STREAMS drivers and drivers for boards.
ebs_modremove	Removes drivers.
ebs_modunload	Unload NewNet Distributed7 modules.
ebs_pkgrm	Removes NewNet Distributed7 packages.
ebs_ps	Displays the current status of processes.
ebs_report	Displays alarm report.
ebs_setrelease	Activate specified NewNet Distributed7 release.
ebs_showlink	Displays link status for SS7 hardware and network interface.
ebs_sysinfo	Displays configuration information of the local host.
ebs_tune	Tunes operating system parameters.
getcfg	Retrieves information about SS7 controllers in the system.
apm_audit	Audit ***apmd*** IPC resources.
apm_getstate	Retrieves current ***apmd*** run state.
apm_kill	Sends a signal to a process.
apm_killall	Sends a signal for all processes to terminate.
apm_ps	Reports process status.
apm_report	Generates a log report.
apm_setstate	Manipulates ***apmd*** run state.
apm_start	Starts the ***apmd*** daemon.

Table 7-1: User Command Summary (Continued)

Command	Description
apm_stop	Terminates the ***apmd*** daemon.
apm_trcapture	Captures information used for tracing execution of application programs on the local host.
apm_trclear	Clears the contents of the trace shared memory.
apm_trgetmask	Retrieves trace mask settings.
apm_trinit	Initializes IPC shared memory.
apm_trsetmask	Sets a trace mask.
apm_trshow	Displays the trace output.
apm_update	Informs ***apmd*** of any changes in the configuration.
sg_pkgrm	Removes an installed version of the Signaling Gateway software
sg_start	Starts the Signaling Gateway software
sg_stop	Stops the Signaling Gateway software
sg_setrelease	Updates the Signaling Gateway and NewNet Distributed7 files to the newest release
sg_trace	Activates the tracing function

Online reference manuals on all these commands are also available. These reference manuals are provided as UNIX manual pages that are retrieved by running the standard UNIX **`man`** command. The following example displays the manual pages for the ebs_config command:

```
man ebs_config
```


7.2 BasicPlatformUtilities

7.2.1 ebs_audit

NAME

ebs_audit Audits dynamic data.

SYNOPSIS

ebs_audit [hostname]

DESCRIPTION

ebs_audit Issues a manual request to audit the dynamic data tables maintained on a host machine which is operating under the NewNet Distributed7 environment. The host machine can be the local host or a remote host.

hostname Identifies which host to run audits on. If a host name is not specified, dynamic data on the local host machine will be audited.

The audit is an internal review and possible correction of all appropriate dynamic data on the specified host machine. Examples of dynamic data are the database tables for the objects that are executing under the NewNet Distributed7 environment and for the SS7 signalling link hardware that is available on the individual host machines.

***Important**: Since the NewNet Distributed7 environment has an automatic mechanism to periodically audit and correct the dynamic data tables stored on the individual host machines, execution of this command is not normally required. This command simply provides a means to manually audit the dynamic data if the automatic auditing mechanism fails to operate properly.*

Related Information

- Section 7.2.13, **ebs_ps** on page 7-21
- Section 7.2.16, **ebs_showlink** on page 7-28

7.2.2 ebs_brdfinfo

NAME

ebs_brdfinfo Get and clear SS7 board crash dump.

SYNOPSIS

ebs_brdfinfo [-glc devname]

DESCRIPTION

ebs_brdfinfo This utility gets and clears the crash dump of a user-specified SS7 signalling hardware—the SS7 board—on the local host machine. After an SS7 board crash is detected by the board sanity check mechanism, the SS7 board crash dump is copied from board-shared memory to a buffer area on the host. The ***ebs_brdfinfo*** utility allows the user to view and clear the latest board crash dump.

The ***ebs_brdfinfo*** utility is part of the set of programs called NewNet Distributed7 Configuration Utilities, which includes ***ebs_dnlbrd***, ***ebs_cfgbrd***, and ***ebs_mngbrd***. This set of utilities allows the user to configure an SS7 board without using the MML interface.

devname This argument specifies the board device driver and instance number of the SS7 board for which the user wants to view and clear the board crash dump. The ***devname*** argument can be a value from the following list:

- sbs334 -- The sbs334 device driver that supports SBS334, SBS370, and SBS372 boards.
- pci334 -- The pci334 device driver that supports PCI334, PCI370, and PCI372 boards.
- pci3xpq -- The pci3xpq device driver that supports PCI370PQ and PCI372PQ boards.
- cpc3xpq -- The cpc3xpq device driver that supports CPC370PQ and CPC372PQ boards.
- pmc8260 -- The pmc8260 device driver that supports the PMC8260 board.
- artic8260 -- The artic8260 driver that supports the ARTIC1000 and ARTIC2000 boards.
- vbrd -- The NewNet Distributed7 Virtual BoaRD (VBRD) device driver.

Use the ***getcfg*** command for a list of available SS7 boards and corresponding instance numbers.

-g Get (view) the crash dump of the specified SS7 board.

-c Clear the crash dump of the specified SS7 board, both on the host buffer and on the SS7 board shared memory.

EXAMPLES

To get (view) the crash dump of an sbs334 with instance 0 on local host Host-A:

host-A% ebs_brdfinfo -g sbs3340

To clear the crash dump of an sbs334 with instance 0 on local host Host-B:

host-A% ebs_brdfinfo -c sbs3340

SAMPLE OUTPUT

A sample crash dump output for an sbs334 with instance 0:

```
sbs334[0] crash log begin...
 crash log. . .
 pc = 00425D38
 sw = 2004 (trap)
 sw = 2004 (handler)
 fault vector = 0002 (2)
 fault format = c000
 a0:00FFF000 a1:00002700 a2:007206B2 a3:00720679
 a4:0072067A a5:0072067B a6:00720680 a7:00720634
 d0:00000064 d1:00000001 d2:00000004 d3:0000000F
 d4:00000005 d5:00000064 d6:FFFFFFFF d7:000002A0
code at fault address. . .
 00425D38  2F28 000C 206E FFFC 2F28 0008 206E FFFC
 00425D48  2F28 0004 206E FFFC 2F10 2F2E FFFC 487B
fault stack frame. . .
 00720634  2004 0042 5D38 C008 00FF F00C FFFF F000
 00720644  0042 5D38 0008 0045 0000 0004 0000 0001
 00720654  0000 0005 0000 0080 0072 06B2 0072 06F8
 00720664  0042 89F0 0072 06F0 0000 0001 0000 0000

sbs334[0] crash log end...
```

Related Information

- Section 6.2.16, **spmd** on page 6-31
- Section 7.3.8, **apm_start** on page 7-49
- Section 7.2.6, **ebs_dnlbrd** on page 7-12
- Section 7.2.3, **ebs_cfgbrd** on page 7-6
- Section 7.2.8, **ebs_mngbrd** on page 7-15

7.2.3 ebs_cfgbrd

NAME

ebs_cfgbrd Configures and unconfigures an SS7 board.

SYNOPSIS

ebs_cfgbrd [-c|u|r [hostname:]devname]

DESCRIPTION

ebs_cfgbrd This utility configures and removes the configuration from a user-specified SS7 signaling hardware—the SS7 board—on the local or remote host machine, following the startup of the NewNet Distributed7 system software using the ***apm_start*** utility. The user can exchange SS7 messages through corresponding SS7 board devices only when the connection between the SPM and the board device driver for an SS7 board is in.place, and when the SS7 board is configured. Configuring/removing configuration on the SS7 board with ***ebs_cfgbrd*** is done conceptually the same way as with the MODIFY-SS7BOARD MML.

The ***ebs_cfgbrd*** utility is one of the NewNet Distributed7 Configuration Utilities, which includes ***ebs_dnlbr***d and ***ebs_mngbr***d. This set of utilities allows the user to configure an SS7 board without using the MML interface.

hostname This argument specifies the name of the host machine where the SS7 board is physically located. It is an optional argument. The local host name is assumed when no host name is entered.

devname This argument specifies the board device driver and instance number of the SS7 board for which the user is interested in viewing and clearing the board crash dump. The ***devname*** argument can be a value from the following list:

- **sbs334** The sbs334 device driver which supports SBS334, SBS370, and SBS372 boards.
- **pci334** The pci334 device driver which supports PCI334, PCI370, and PCI372 boards.
- **pci3xpq** The pci3xpq device driver which supports PCI370PQ and PCI372PQ boards.
- **artic8260** The artic8260 driver that supports the ARTIC1000 and ARTIC2000 boards.
- **vbrd** The NewNet Distributed7 Virtual BoaRD (VBRD) device driver.

Use the ***getcfg*** command for a list of available SS7 boards and corresponding instance numbers.

-c	Configures the specified SS7 board. It has the same effect as the MODIFY-SS7BOARD: CONF=ON; MML command.
-u	Unconfigures the specified SS7 board. It has the same effect as the MODIFY-SS7BOARD: CONF=OFF; MML command.
-r	Recovers the specified SS7 board which is in FAILED state. It has the same effect as the MODIFY-SS7BOARD: CONF=ON; MML command when the SS7 board is in FAILED state. This option fails if the board's state is not FAILED.

EXAMPLES

Enter the following to configure an sbs334 with instance 0 on local host Host-A:

host-A% ebs_cfgbrd -c host-A:sbs3340

OR,

host-A% ebs_cfgbrd -c sbs3340

Enter the following to configure an sbs334 with instance 0 on remote host Host-B:

host-A% ebs_cfgbrd -c host-B:sbs3340

Enter the following to unconfigure an sbs334 with instance 0 on local host Host-A:

host-A% ebs_cfgbrd -u host-A:sbs3340

OR,

host-A% ebs_cfgbrd -u sbs3340

Enter the following to unconfigure an sbs334 with instance 0 on remote host Host-B:

host-A% ebs_cfgbrd -u host-B:sbs3340

Enter the following to recover an sbs334 with instance 0 on local host Host-A:

host-A% ebs_cfgbrd -r host-A:sbs3340

OR,

host-A% ebs_cfgbrd -r sbs3340

Enter the following to recover an sbs334 with instance 0 on remote host Host-B:

host-A% ebs_cfgbrd -r host-B:sbs3340

Related Information

- ebs_dnlbrd on page 7-12
- ebs_mngbrd on page 7-15

7.2.4 ebs_config

NAME

ebs_config Configures the system's operation mode.

SYNOPSIS

ebs_config [-u]

DESCRIPTION

ebs_config Use ***ebs_config*** to configure the operation mode of NewNet Distributed7 system software on the local host as stand-alone or distributed. This script is only for modifying the operation mode that was specified during initial system software installation, i.e., from stand-alone to distributed, or vice versa.

ebs_config accesses information stored in the **`/etc/amgrmode`** file—created when NewNet Distributed7 is installed—to determine the current mode of operation on the local machine. It then replaces selected components of the base NewNet Distributed7 system software, such as the executables and configuration files, with their appropriate versions. Finally, it updates the **`/etc/amgrmode`** file to reflect the new mode of operation, and removes all network related managed object database files.

The ***ebs_config*** script can also modify the default hostname setting that NewNet Distributed7 software uses. By default, the software uses the UNIX nodename set on the local host with the **`uname -n`** command as the official hostname. There are times, however, when the user may want to give the software a hostname other than the official UNIX nodename. For example, in a product configuration where a particular host machine is part of multiple networks, such as public and private networks, the user can reserve the official hostname of the machine for one network, and have NewNet Distributed7 software run on another network under a different hostname.

To determine the default hostname, the system software accesses existing hostname information stored in the **`/etc/amgrhost`** file. The user can modify the contents of this file with the **`ebs_config`** script. Note, however, that use of the **`/etc/amgrhost`** file is optional; this file does not get created during initial system software installation. In the absence of this file, the system software default is to the official UNIX hostname. Users interested in operating the software under a hostname that is different from the official hostname must run the ***ebs_config*** script to reset the hostname on that machine after the initial system software installation.

-u Upgrade option for switching LAN configuration from single to dual, or vice versa. Use this option only if you have already configured your system, that is you have specified the operation mode and defined any remote hosts. Be aware that when this option is specified, all managed object (MO) database files associated with the NTWK, HOST, and TCPCON managed objects are removed from the local host, and must be re-entered.

Important*: You must have **root** privileges to run the **ebs_config** script.*

Caution: *Run this script only when NewNet Distributed7 system software on the local host is **NOT** running. Parameters initialized and/or set by this script are used by the NewNet Distributed7 system software during system startup time and thereafter.*

ENVIRONMENT

The EBSHOME environment variable must be set before running this script.

FILES

/etc/amgrhost

/etc/amgrmode

$EBSHOME/access/drv/dramod

$EBSHOME/access/drv/.dramod.sa

$EBSHOME/access/drv/.dramod.dist

$EBSHOME/access/RUN/config/PMGR/apmconfig

$EBSHOME/access/RUN/config/PMGR/.apmconfig.sa

$EBSHOME/access/RUN/config/PMGR/.apmconfig.dist

$EBSHOME/access/RUN/DBfiles/net_tcpcon.DB

$EBSHOME/access/RUN/DBfiles/net_host.DB

$EBSHOME/access/RUN/DBfiles/net_ntwk.DB

Related Information

• ebs_modinstall on page 7-17
• ebs_modunload on page 7-19

7.2.5 ebs_dbconfig

NAME

ebs_dbconfig Save or restore configuration data.

SYNOPSIS

ebs_dbconfig -s*ave -re*store [-d*ir backup-dir] [-f*orce] [-ru*n run-list] [-p*attern pattern-list]

DESCRIPTION

ebs_dbconfig This utility is intended to save or restore Distributed7 configuration data. It can be instructed to use a particular backup directory, select a subset of the configuration directories and backup (or restore) only files matching the specified pattern(s). All options other than **save** and **restore** have default values. One and only one of **save** or **restore** options must be specified for a particular invocation of ***ebs_dbconfig***.

The requesting user must have read and read-write privileges to specified source and destination directories, respectively.

The ***ebs_dbconfig*** utility also checks the usage status of the files to be backed-up (restored). Unless the **force** (-force) option is specified, the backup (restore) action is rejected if files currently in use are being backed-up (restored).

For all options, * sign indicates the end of the mandatory option prefix.

OPTIONS

-s*ave Save existing Distributed7 configuration to the specified backup directory. This option cannot be specified together with the restore option.

-re*store Restore Distributed7 configuration from the specified backup directory. This option cannot be specified together with the save option.

-d*ir Specifies the backup directory. Defaults to ***$EBSHOME/access/BACKUP***.

-ru*n Used to qualify the configuration (RUN) directories to be backed-up (restored). Empty directories are skipped even if specified. Defaults to "RUN RUN0 RUN1 RUN2 RUN3 RUN4 RUN5 RUN6 RUN7"

-p*attern Specifies a list of shell style glob-patterns for selecting files to be backed-up or restored. Defaults to "*".

-f*orce Used to skip file usage check during the backup (restore) operation. When this option is used, backed-up files might contain inconsistent information.

EXAMPLES

The following example illustrates how ***ebs_dbconfig*** can be used to save the complete configuration to the default backup directory:

ebs_dbconfig -s

To restore the core configuration directory as well as SP1 and SP2 configuration directories from backup directory ***/home/config/D7***:

ebs_dbconfig -re -d /home/config/D7 -run "RUN RUN1 RUN2"

Finally, to save all SCCP and MTP configuration data to the default backup directory:

ebs_dbconfig -save -pattern "mtp* sccp*"

FILES

$EBSHOME/access/RUN/config/RUN/DBfiles

$EBSHOME/access/RUN/config/RUN[0-7]/DBfiles

7.2.6 ebs_dnlbrd

NAME

ebs_dnlbrd Resets and downloads an SS7 board.

SYNOPSIS

ebs_dnlbrd [devname]

DESCRIPTION

ebs_dnlbrd This utility resets and downloads SAL/MTPL2 binaries to a user-specified SS7 signaling hardware—SS7 board—on the local host machine.

The ***ebs_cfgbrd*** utility is one of the NewNet Distributed7 Configuration Utilities, which includes ***ebs_cfgbr***d and ***ebs_mngbr***d. This set of utilities allows the user to configure an SS7 board without using the MML interface.

devname This argument specifies the board device driver and instance number of the SS7 board for which the user is interested in viewing and clearing the board crash dump. The ***devname*** argument can be a value from the following list:

- **sbs334** The sbs334 device driver which supports SBS334, SBS370, and SBS372 boards.
- **pci334** The pci334 device driver which supports PCI334, PCI370, and PCI372 boards.
- **pci3xpq** The pci3xpq device driver which supports PCI370PQ and PCI372PQ boards.
- **artic8260** The artic8260 driver that supports the ARTIC1000 and ARTIC2000 boards.
- **vbrd** The NewNet Distributed7 Virtual BoaRD (VBRD) device driver.

Use the ***getcfg*** command for a list of available SS7 boards and corresponding instance numbers.

EXAMPLES

Enter the following to download an sbs334 with instance 0 on local host-A:

```
host-A% ebs_dnlbrd sbs3340
```

Related Information

- ebs_cfgbrd on page 7-6
- ebs_mngbrd on page 7-15

7.2.7 ebs_log

NAME

ebs_log Activates and deactivates message logging and lists logged processes.

SYNOPSIS

ebs_log [-l] [-do]

DESCRIPTION

ebs_log Invokes the utility which controls the NewNet Distributed7 message logging capabilities. The utility prompts for the information needed to activate or deactivate logging for a particular service endpoint, a user process or the link between any two adjacent STREAMS multiplexers. This command can also be used to display a list of all service endpoints that have message logging currently active.

When the message logging capability is activated, a copy of each message received or sent through the service endpoint is forwarded to either the standard NewNet Distributed7 LOG_MNGR daemon or the user process specified with the ***-o*** option. The logging process must be active and running during an entire log session. If the LOG_MNGR daemon terminates, message logging at all appropriate service endpoints will automatically be deactivated. Also, if a process being logged terminates, logging at all service endpoints associated with that process will automatically be deactivated.

-l Prints a list of any processes for which the message logging capability is currently active. (This option only provides the list, it does not prompt for information.) The QUEUE column indicates whether message logging is active at the read-side and/or write-side queue. See ***ebs_ps*** for a description of the columns that appear in the display.

-d Deactivates the message logging capability for a particular service endpoint. Deactivation of message logging at an endpoint where it is not currently active has no effect.

-o Enables the user to redirect the logged messages to a process other than the standard NewNet Distributed7 LOG_MNGR daemon. By default, messages are logged to the LOG_MNGR daemon. If a user-specified process is used, it must be designed to handle the messages it receives. Normally, a logger process will save the message contents to a file and/or display them to the standard output (see ***logd***).

After entering the command to activate or deactivate logging, a prompt will appear for the object type - **named object**, **SS7 object**, **IPC key**, or **MUX object**. After selecting the type, prompts occur to uniquely identify the process of that type.

A named object is a process that does not directly send SS7 messages. An example is a Call Control application that interfaces with the ISUP process. A named object is identified by the name that the process provided in the ***spm_open()*** and ***spm_bind()*** functions at registration, up to 13 characters (the 14th is the null character to terminate the string).

An SS7 object is a process that directly communicates with an SS7 protocol multiplexer, such as a TCAP application.

- A TCAP application is uniquely identified by its user part number (3 for SCCP), logical signalling point number (SP), subsystem number (SSN), and instance number. The SP and SSN were specified by the process. The instance number is assigned by the system when the process registers with ***tcm_open()***. The value is returned by the function. If only one application is registered with a particular SSN, then the instance number is 1.
- Applications associated with Signalling Network Management are uniquely identified by the logical signalling point number and the user part number of 0.
- The ISUP process is identified by the logical signalling point number and the user part number 5.

To select IPC key, the user must know the IPC key that the system assigned to the process when it registered with ***spm_open()*** and ***spm_bind()***. The process would have retrieved the value from the **IPCkey** field of the **SPMreg_t** structure or by calling ***spm_getusrinfo()***, and then would have had to create a way for the operator to access it.

A MUX object is a connection between two STREAMS multiplexers (e.g. UPM, MTP, SCCP, and those listed under ***ebs_ps***). To identify a MUX object, the user is prompted for the multiplexer ID and the signalling point number. This information may be seen in the output of ***ebs_ps***.

Related Information

- Section 6.2.10, **logd** on page 6-21
- Section 7.2.13, **ebs_ps** on page 7-21

7.2.8 ebs_mngbrd

NAME

ebs_mngbrd Starts and stops SS7 board sanity checks.

SYNOPSIS

ebs_mnglbrd [-o|f devname]

DESCRIPTION

ebs_mngbrd This utility starts and stops the sanity check on a user-specified SS7 signaling hardware—SS7 board—on the local host machine. SS7 board sanity check periodically tests the board state to detect software/ hardware problems. Starting the sanity check is normally done during SS7 board configuration. The ***ebs_mngbrd*** utility allows the user to take the SS7 board off line by stopping the sanity check to simulate a board crash.

The ***ebs_mngbrd*** utility is one of the NewNet Distributed7 Configuration Utilities, which includes ***ebs_dnlbr***d and ***ebs_cfgbr***d. This set of utilities allows the user to configure an SS7 board without using the MML interface.

devname This argument specifies the board device driver and instance number of the SS7 board for which the user is interested in viewing and clearing the board crash dump. The ***devname*** argument can be a value from the following list:

- **sbs334** The sbs334 device driver which supports SBS334, SBS370, and SBS372 boards.
- **pci334** The pci334 device driver which supports PCI334, PCI370, and PCI372 boards.
- **pci3xpq** The pci3xpq device driver which supports PCI370PQ and PCI372PQ boards.
- **artic8260** The artic8260 driver that supports the ARTIC1000 and ARTIC2000 boards.
- **vbrd** The NewNet Distributed7 Virtual BoaRD (VBRD) device driver.

Use the ***getcfg*** command for a list of available SS7 boards and corresponding instance numbers.

-o Start the sanity check on the specified SS7 board, in other words, take the SS7 board online.

-f Stop the sanity check on the specified SS7 board, in other words, take the SS7 board off line.

EXAMPLES

Enter the following to start sanity on an sbs334 with instance 0 on local host Host-A:

```
host-A% ebs_mngbrd -o sbs3340
```

Enter the following to stop sanity on an sbs334 with instance 0 on local host Host-B:

host-A% ebs_mngbrd -f sbs3340

Related Information

- ebs_cfgbrd on page 7-6
- ebs_dnlbrd on page 7-12

7.2.9 ebs_modinstall

NAME

ebs_modinstall Installs NewNet Distributed7 modules.

SYNOPSIS

ebs_modinstall [-f]

DESCRIPTION

ebs_modinstall Installs the NewNet Distributed7 STREAMS components, i.e., multiplexers, modules, and device drivers. When run, it copies the executables from an appropriate product directory to the ***/usr/kernel/drv*** and ***/usr/kernel/strmod*** directories. It also updates the various configuration files associated with the newly-introduced device drivers and creates special files associated with each NewNet Distributed7 multiplexer or device driver.

-f Force option. When multiple NewNet Distributed7 releases exist on a machine, the ***ebs_modinstall*** script cannot be used for installation. Rather, the ***ebs_setrelease*** script must be used to activate a particular NewNet Distributed7 release. The ***-f*** option allows the user to bypass checks regarding multiple NewNet Distributed7 releases, and performs the installation in an unconditional manner. Use of this option is restricted to reconfiguration, that is adding, removing, or replacing the signaling hardware on the machine.

Important*: The* ***$EBSHOME*** *environment variable must be set before invoking this script, and you must have root privileges to execute this script.*

Related Information

• ebs_modremove on page 7-18

7.2.10 ebs_modremove

NAME

ebs_modremoveRemoves NewNet Distributed7 modules.

SYNOPSIS

ebs_modremove [-f]

DESCRIPTION

ebs_modremove Removes the NewNet Distributed7 STREAMS components, such as multiplexers, modules, and device drivers. When run, it cleans up the appropriate executables in the ***/usr/kernel/drv*** and ***/usr/kernel/strmod*** directories, updates various configuration files on the removed device drivers, and deletes all appropriate special files associated with the NewNet Distributed7 multiplexers and device drivers.

-f Force option. When multiple NewNet Distributed7 releases exist on a machine, the ***ebs_modremove*** script cannot be used for removal. Rather, the ***ebs_setrelease*** script must be used to deactivate a particular NewNet Distributed7 release. The ***-f*** option allows the user to bypass checks regarding multiple NewNet Distributed7 releases, and performs the removal in an unconditional manner. Use of this option is restricted to reconfiguration, that is adding, removing, or replacing the signaling hardware on the machine.

Important*: The* ***$EBSHOME*** *environment variable must be set before invoking this script, and you must have root privileges to execute this script.*

Related Information

• ebs_modinstall on page 7-17

7.2.11 ebs_modunload

NAME

ebs_modunload Unloads NewNet Distributed7 modules.

SYNOPSIS

ebs_modunload

DESCRIPTION

ebs_modunload This script is for unloading, from a Sun platform, the STREAMS components, such as multiplexers, modules, and device drivers, comprising the NewNet Distributed7 system software.

Important*: You must have root privileges to run this script.*

Related Information

• ebs_modremove on page 7-18

7.2.12 ebs_pkgrm

NAME

ebs_pkgrm Removes the NewNet Distributed7 packages.

SYNOPSIS

ebs_pkgrm\ version

DESCRIPTION

The ***ebs_pkgrm*** script removes all software packages from a Sun platform that are associated with a user-specified ***version*** (e.g., 1.0.0.1) of the NewNet Distributed7 system software. Since the NewNet Distributed7 product has software packages that can be installed, this script is an alternative and a shortcut to the UNIX ***pkgrm()*** utility because it frees the user from knowing the names of the individual software packages and/or the dependencies between them.

When run, ***ebs_pkgrm*** script searches the list of all software packages installed on the local host and compiles a list of all packages associated with the NewNet Distributed7 release entered in the command. Subsequently, ***ebs_pkgrm*** runs the UNIX ***pkgrm()*** utility to remove these software packages in the correct order.

Important*: You must have root privileges to run this script.*

Related Information

• ebs_setrelease on page 7-27

7.2.13 ebs_ps

NAME

ebs_ps Reports process status.

SYNOPSIS

ebs_ps [-a|d|n|m|s] [-lqtx]

DESCRIPTION

ebs_ps Retrieves and displays a snapshot of information about active processes which are running under the NewNet Distributed7 environment. Since the environment is constantly changing, the information is only absolutely true for the instant when it was gathered.

Without options, ***ebs_ps*** displays information about the processes that have the same user ID or group ID as the user who issued the command. Without options, the output contains only the UNIX process ID, process status, operation mode, host machine identifier, STREAMS multiplexer, STREAMS queue identifiers, process type, and process name. Otherwise, the information to be displayed is controlled by the options.

The information displayed by ***ebs_ps*** is based on data stored in a process table on the local host machine. This table contains information about processes running on the local host machine and on all other machines in the distributed network. Individual machines within a network may have differences in the contents of their process tables due to processes that are only registered to the local host machine. The NewNet Distributed7 environment has a built-in mechanism which keeps the appropriate portions of the local process tables on the individual machines synchronized at all times.

-a Prints information about all active processes regardless of the process type, i.e., named object, SS7 object, and ownership, i.e., user ID, group ID. Without this option, only information for processes with the same user ID and/or group ID as the issuer of this command will be printed.

-d Prints information about daemon processes executing on any host within network.

-n Prints information about all active named object processes whose user ID or group ID are the same as that of the issuer of the command.

-m Prints information about all STREAMS multiplexers that are in use. STREAMS multiplexers implement the individual layers of the SS7 protocol stack, i.e., MTP, SCCP, TCAP, for individual signaling points. Process ID, user ID, and group ID fields for multiplexer objects are always set to 0 because they are kernel-level entities, not processes.

-s Prints information about all active SS7 object processes whose user ID or group ID are the same as that of the issuer of the command.

-l Prints additional information about each process including the assigned internal key and the service type, user ID, and group ID associated with it.

-q Prints a complete list of the STREAMS read-side queue addresses associated with each process. This option is only meaningful when used with the ***-l*** option.

-t Print the time of registration for each process. This option is meaningful only when used in conjunction with the -l option.

-x Indicates that the ***ebs_ps*** utility should not bind an address to the service end point associated with it. Unless this option is specified, a named object entry for ***ebs_ps*** will be created in the process table of the local machine. This option allows ***ebs_ps*** to retrieve the contents of the local process table without disturbing it.

OUTPUT VALUES

The output of this command contains several columns of information, depending on the options used in the command. The following table contains the column headings of the output, the meaning of the column, and possible values that may be displayed. The fields that are displayed depend on the command options used.

Table 7-2: ebs_ps Output Description

Column Heading	Valid Values	Description
OBJECT		Object type.
	daemon	Process is a daemon process of the NewNet Distributed7 system software. The name associated with the process is printed within brackets.
	nmdobj	Process is a user process that is addressable as a named object. The name of the process is printed within brackets.
	ss7obj	User process that is addressable as an SS7 object. Brackets contain the signaling point and the MTP user part for the process. For SCCP and TCAP applications, the subsystem and instance number are also included.
	tcpobj	Identifies user processes that are addressable as TCP/IP objects (e.g.,TC applications that utilize the TCP/IP transport services). The TCP/IP port number as well as the instance information associated with the process will be printed within brackets.
	muxobj	A STREAMS multiplexer that is in use. Multiplexers implement the SS7 protocol stack for individual signaling points located on a host machine. All multiplexers other than the SPM may have multiple physical instances on a given host machine. There are 5 types:
		spm: Service Provider Module. (one physical instance per host machine)
		upm: Message Transfer Part (MTP) User Part Multiplexer. Used for implementing the MTP Signalling Message Handling (SMH) protocol.
		snm: MTP Signalling Network Management (SNM) Multiplexer.
		sccp: Signalling Connection Control Part (SCCP) Multiplexer.
		tcap: Transaction Capabilities Application Part (TCAP) Multiplexer.

Table 7-2: ebs_ps Output Description

Column Heading	Valid Values	Description
MODE		Operation mode. Combination of the following:
	A	Active mode. Process can receive and send messages.
	S	Standby mode. Process can send messages but cannot receive messages unless the message originator specifies its destination address in the L_IPCKEY format.
	L	Local. Process is addressable on the local machine only; other machines on the network do not have information about its existence. The lack of this flag indicates that the process is known and addressable across the network.
	X	Exclusive. No other process can bind with the address of this process. If L is shown, this restriction applies to processes on the local machine only. Otherwise, it applies to the network.
STAT		Current process status.
	ok	Process exists and operates in the normal state, i.e., it is neither blocked nor in a wait state.
	blkd	Process is in a blocked state. Its read-side STREAMS queue is full. This status may indicate a hung process and/or an overflow condition on the read-side STREAMS message queue.
	wait	Process is in a transient wait state while a software handshake procedure is underway to confirm its network-wide binding request with all machines in the network.
SRV		Service type of the process. Refer to the <***api.h***> header file for a list of service types available.
HOST		Name of the host machine on which the process is executing.
MUX		The STREAMS multiplexer used for establishing the service endpoint for the process on the host machine. Information displayed in this field identifies the multiplexer type, physical instance number [except for the SPM], and the upper stream number associated with the process. Only the stream number is shown for SPM since it only has one instance on a system.
	sccp/#/#	Signalling Connection Control Part (SCCP) Multiplexer.
	snm/#/#	MTP Signalling Network Management (SNM) Multiplexer.
	spm/#	Signalling Point Multiplexer.
	tcap/#/#	Transaction Capabilities Application Part (TCAP) Multiplexer.
	upm/#/#	Message Transfer Part (MTP) User Part Multiplexer. Used for implementing the MTP Signalling Message Handling (SMH) protocol.
PID		UNIX process ID assigned to the process on the host machine named in HOST.
KEY		Internal key assigned to the process on the local host machine. It identifies the slot allocated for the process in the local process table and is in the range from 0 to NMAXPROC.
GID		Group ID associated with the process, in decimal. For all multiplexer objects, this field is set to 0.
UID		User ID associated with the process, in decimal. For all multiplexer objects, this field is set to 0.
1STQADDR		Address, in hex, of the read-side STREAMS queue for the very first connection between the process and the STREAMS multiplexer.
SPMQADDR		Address, in hex, of the read-side STREAMS queue on the SPM multiplexer that should be used when routing messages to the process via the SPM. If equal to 1STQADDR, the process is readily connected to the SPM multiplexer.
UPMQADDR		Address, in hex, of the read-side STREAMS queue on the corresponding UPM multiplexer that should be used when routing messages to the process via the UPM. If equal to 1STQADDR, the process is readily connected to the UPM multiplexer.

Table 7-2: ebs_ps Output Description

Column Heading	Valid Values	Description
SNMQADDR		Address, in hex, of the read-side STREAMS queue on the corresponding SNM multiplexer that should be used when routing messages to the process via the SNM. If equal to 1STQADDR, the process is readily connected to the SNM multiplexer.
SCMQADDR		Address, in hex, of the read-side STREAMS queue on the corresponding SCCP multiplexer that should be used when routing messages to the process via the SCCP. If equal to 1STQADDR, the process is readily connected to the SCCP multiplexer.
TCMQADDR		Address, in hex, of the read-side STREAMS queue on the corresponding TCAP multiplexer that should be used when routing messages to the process via the TCAP. If equal to 1STQADDR, the process is readily connected to the TCAP multiplexer.
TIME		The time at which the process registered.

7.2.14 ebs_report

NAME

ebs_report Generates an alarm report.

SYNOPSIS

ebs_report [-b mmddyy -e mmddyy -p pri hostname(s)]

DESCRIPTION

ebs_report Collects information from the ***alarmd*** log files stored on the individual host machines in the NewNet Distributed7 environment and creates a report. This utility organizes the records chronologically, searches the records for user-specified information, generates customized alarm reports, and displays the reports on the standard output. Without options, ***ebs_report*** generates a report that contains all alarm conditions existing for the local host up to the current point in time. Otherwise, the contents of the report depend on the options specified.

-b mmddyy Includes all alarm conditions that occurred on or after the specified date. The date is specified in the *mmddyy* format, with the month, day of the month, and year expressed in 2-digit numerals (as in the UNIX date(1) command).

-e mmddyy Includes all alarm conditions that occurred on or before the specified date. The date is specified in the *mmddyy* format, with the month, day of the month, and year expressed in 2-digit numerals (as in the UNIX date(1) command).

-p pri Includes alarm conditions with the specified priority levels only. Without this option, the default includes alarm conditions at all priority levels. The ***pri*** argument may contain any combination of the following values:

- **1** Informational messages.
- **2** Messages at minor priority level.
- **3** Messages at major priority level.
- **4** Messages at critical priority level.
- **5** Messages at fatal priority level.

hostname Identifies the host machine(s) whose alarm log files should be used for generating the report. If multiple hosts are specified, each hostname must be separated from the others by white space. If a hostname is not specified, the report will be generated for the local host only.

***Important**: The **$EBSHOME** environment variable must be set before invoking this utility.*

FILES

$EBSHOME/access/RUN/alarmlog

EXAMPLES

The following are examples of ***ebs_repor***t command:

Enter the following to display all *critical* alarm messages generated on the local host, up to the current time:

```
ebs_report -p 4
```

Enter the following to display all *major* and *critical* alarm messages generated on the host *phantom* since August 14, 1994.

```
ebs_report -b 081494 -p 34 phantom
```

Enter the following to display all *mino*r, *majo*r, and *critical* alarm messages generated on the hosts, *sun* and *mar*s, between the dates September 3, 1994 and December 7, 1994.

```
ebs_report -b 090394 -e 120794 -p 234 sun mars
```

Important*: Refrain from running the* **ebs_report** *utility on a live system (where several user-space application programs are running) as it may consume a large amount of CPU resources to search through and process the event log files accumulated on the system, which is likely to degrade the performance of user-space applications running on the system.*

7.2.15 ebs_setrelease

NAME

ebs_setrelease Activates the specified NewNet Distributed7 release.

Note*: This script replaces the* ***ebs_modinstall*** *script. While the* ***ebs_modinstall*** *script still exists, it cannot be used on machines where multiple versions of the NewNet Distributed7 software are installed.*

SYNOPSIS

ebs_setrelease version | -i

DESCRIPTION

ebs_setrelease Use this utility to install, on a Sun platform, the STREAMS components, such as multiplexors, modules, and device drivers, associated with NewNet Distributed7 version entered in the command. When run, this utility:

- locates the path for the specified NewNet Distributed7 release
- creates and/or updates the **`/etc/amgrhome`** file if necessary
- establishes the $EBSHOME/access symbolic link, that points to the access directory of the specified NewNet Distributed7 release
- copies the kernel components from the ***/usr/kernel/strmod*** directories
- updates various configuration files with the newly introduced device drivers
- creates a number of special device files associated with each NewNet Distributed7 multiplexor or device driver.

After copying the NewNet Distributed7 kernel components, this script converts the NewNet Distributed7 database files from any previous version.

-i Causes ***ebs_setrelease*** to display information about the currently installed release, such as the version and access directory path.

7.2.16 ebs_showlink

NAME

ebs_showlink Reports link status.

SYNOPSIS

ebs_showlink [-lx]

DESCRIPTION

ebs_showlink Retrieves and displays information about all connections to the SS7 boards and the Network Interface (NI) hardware. In a distributed NewNet Distributed7 environment, information on SS7 boards located at remote host machines can also be retrieved. That information includes the link number, link type, host, status, and last change in status. The SS7 and NI connections are established and maintained by the Service Provider Module (SPM) on the local host machine through appropriate STREAMS modules or drivers. STREAMS modules perform the message format translations for all messages flowing through the modules in both upstream and downstream directions.

- TRMOD STREAMS

  The TRMOD STREAMS module connects the SPM and the SS7 board device. SS7 boards provide access to the SS7 signaling link hardware on the local host, and are used to send and receive SS7 messages over the SS7 links. The ***spmd*** daemon establishes and maintains the SS7 device driver connections when NewNet Distributed7 system software is started on the local host machine with ***ebs_start***. Once a connection to the SS7 board driver is made, it remains in that state until NewNet Distributed7 system software on the local host machine is stopped, or until a manual request is placed to reconfigure the corresponding SS7 device connection.

- NIMOD STREAMS

  The NIMOD STREAMS module connects the SPM and the TCP/IP protocol suite. The NI hardware establishes a reliable, connection-oriented, i.e., TCP/IP based, interface between individual machines in a distributed NewNet Distributed7 environment for inter-machine message exchange. The ***netd*** daemon establishes and maintains the TCP/IP connections to remote machines, and must communicate with its peers on the remote host machines to set the TCP/IP connections up. While a TCP/IP connection is in service, optional heartbeat messages can be exchanged periodically over that link to monitor its health. To end a TCP/IP connection, a disconnect request, a disconnect indicator, or an error message from the TCP/IP protocol suite must occur to cause the ***netd*** daemon to tear down the corresponding connection.

-l	Prints additional information about each link, including the internet address of the link and heartbeat-related information (see Table 7-3).
-x	Indicates that ***ebs_showlink*** should not bind an address to the service endpoint associated with it. Unless this option is specified, a named object entry for ***ebs_showlink*** will be created in the process table of the local machine.

OUTPUT VALUES

The output of this command contains several columns of information, depending on the options used. The following table contains column headings of the output, the meaning of the column, and possible values that may be displayed. The fields that are displayed depend on the command options used.

Table 7-3: ebs_showlink Output Description

Column	Valid Values	Description
LINK		Lower stream number on the SPM multiplexer that is connected to an appropriate STREAMS driver. In current implementation, link numbers 0 through 7 are reserved for connections to the SS7 driver module and link numbers 8 through 15 are for connections to the TCP/IP protocol suite.
TYPE		Type of the connection, i.e., the hardware device associated with the link.
	sbs334	Indicates that the link interconnects the SPM multiplexer to the sbs334 device driver which supports SBS334, SBS370, and SBS372 boards.
	pci334	Indicates that the link interconnects the SPM multiplexer to the pci334 device driver which supports PCI334, PCI370, and PCI372 boards.
	pci3xpq	Indicates that the link interconnects the SPM multiplexer to the pci3xpq device driver which supports PCI370PQ and PCI372PQ boards.
	pci3xapq	Indicates that the link interconnects the SPM multiplexer to the pci3xapq device driver, which supports PCI370APQ and PCI372APQ boards.
	cpc37xpq	Indicates that the link interconnects the SPM multiplexer to the cpc37xpq device driver, which supports CPC370APQ and CPC372PQ boards.
	pmc8260	Indicates that the link interconnects the SPM multiplexor to the pmc8260 device driver which supports the PMC8260 board.
	artic8260	Indicates that the link interconnects the SPM multiplexor to the artic8260 device driver which supports the ARTIC1000 and ARTIC2000 boards.
	vbrd	Indicates that the link interconnects the SPM multiplexer to the NewNet Distributed7 Virtual Board (VBRD) device driver.
	ecp	Indicates that the link interconnects the SPM multiplexer to an SS7 board device driver of unknown type, i.e., that does not belong to the aforementioned set of device drivers.
	tcp/ip	Indicates that the link interconnects the SPM multiplexer to the TCP/IP protocol suite; therefore, is used to communicate with a remote host machine on the network.
HOST		Name of the host machine associated with the link. For SS7 board connections, name identifies the host where the SS7 signaling hardware is physically located. For the TCP/IP connections, name identifies the remote host machine accessible via that link. Third-party hosts that are not equipped with the NewNet Distributed7 software are marked with a question mark tag (?) at the end.
RMTHOST		Field that contains the name of the remote host machine that is accessible through the TCP/IP connection. Third-party hosts not equipped with NewNet Distributed7 software are marked with a question mark tag (?) at the end.
INETADDR		IP address of the host (in the dotted decimal notation).

Table 7-3: ebs_showlink Output Description

Column	Valid Values	Description
STAT		Current status of the connection between the SPM and the STREAMS device driver.
	L	Linked. Connection is in place.
	U	Unlinked. Connection is not in place.
	A	Active mode of operation. Messages can be exchanged across the connection.
	S	Standby mode of operation. Connection is not being used for exchanging messages.
	B	Blocked. Connection cannot be used because it is not possible to exchange data across the TCP/ IP STREAMS modules.
	I	Isolated, i.e., disconnected LAN interface
TIME		Last date and time that the link status changed between *linked* and *unlinked*.The time stamp is based on the system clock of the local host machine.
HBEAT		Current heartbeat status for the link.
	-	No heartbeat mechanism on the link.
	ok	Remote host is responding on time to heartbeat requests originated over the link by the local host.
	failed	Remote host machine has failed to respond on time to one or more heartbeat requests originated over the link by the local host.
HBACT		Action to be taken when the heartbeat mechanism over the link fails.
	0	No action. However, the link heartbeat status is changed to *failed*.
	1	Update the process table on the local host machine by removing all entries that belong to processes on the remote host machine that failed to respond to heartbeat messages. The process tables on both host machines will automatically be synchronized when the heartbeat mechanism over the link is restored.
HBINT		Length of the heartbeat interval in milliseconds.
HBSENT		Indicates the total number of heartbeat responses originated by the local host and sent across the link.
HBRCVD		Indicates the total number of heartbeat responses originated by the remote host, i.e., in response to heartbeat requests originated by the local hosts, and received across the link.
SEQNUM		The sequence number of the last message received and processed over the link

7.2.17 ebs_sysinfo

NAME

ebs_sysinfo Shows host machine information.

SYNOPSIS

ebs_sysinfo [-ax] [-l] [-m mode ***]***

DESCRIPTION

ebs_sysinfo Obtains information about the current hardware and software configuration on the local host machine. The basic command, with no options, displays the system name, node name, operating system release, operating system version number, system kernel architecture, machine internet address, and operation mode. Alternate host names and the internet addresses associated with those names are also displayed for multi-homed hosts. All zero's will be displayed [within brackets] for the internet address if the internet address to be used by the kernel-level NewNet Distributed7 components has not been initialized to its proper value.

-a Initializes the internet address to be used by the kernel-level NewNet Distributed7 components on the local machine. This option is normally not needed because either ***spmd*** or ***netd*** usually initializes the internet address at system startup time.

-l This function is used to retrieve and display licensing information on the local host.

-x Indicates that ***ebs_sysinfo*** should not bind an address to the service endpoint associated with it. Unless this option is specified, a named object entry for ***ebs_sysinfo*** will be created in the process table of the local machine.

-m mode Specifies the current operation mode for the local host machine, either *server* or *client*. Currently, mode setting information is not used by the system. However, the ***spmd*** daemon initializes the operation mode to *server* at system startup time.

7.2.18 ebs_tune

NAME

ebs_tune Tunes the operating system parameters.

SYNOPSIS

ebs_tune [-d]

DESCRIPTION

ebs_tune Modifies operating system parameters associated with the STREAMS subsystem and IPC semaphores, shared memory segments, and message queues for the operational needs of the NewNet Distributed7 environment on a Sun hardware platform. The parameters manipulated by this script are contained in the ***/etc/system*** configuration file and are used by the kernel during initialization of the system.

The ***ebs_tune*** script should only be run once, after running ***ebs_modinstall*** when installing the NewNet Distributed7 system software.

When run, ***ebs_tune*** creates a backup copy of the ***/etc/system*** file and names the backup, ***/etc/system.old***. Then, it reads the contents of the `/etc/system` file and appends to, or deletes from, a set of instructions for the kernel parameters, if those parameters have not already been customized. If some of the parameters have already been customized, a message is displayed on the screen after manipulation of their current values. The system must be re-booted after running this utility for changes to be made to certain kernel parameters.

-d Deletes changes introduced to the `/etc/system` configuration file by executing the ***ebs_tune*** script.

Important*: You must have root privileges to execute this script. The UNIX system must be re-booted after execution of this script for the changes to take effect.*

FILES

/etc/system

/etc/system.old

Related Information

• ebs_modinstall on page 7-17

7.2.19 getcfg

NAME

getcfg Gets information about SS7 controllers in the system.

SYNOPSIS

getcfg

DESCRIPTION

getcfg Used to get information about the SS7 controllers in the system. It displays an output in column format informing the user about the driver, type of board, physical number of the slot carrying the board, and instance number of the boardand driver state (if the operating system is AIX).

Note *Use of the* ***ebs_modremove*** *and the* ***ebs_modinstall*** *commands (located under* ***$EBSHOME/access/install****) is required to get a correct* ***getcfg*** *output when any of the following actions are performed with the boards installed in the host system: removing a board from the system, adding a board to the system, replacing a board with another board of a different type.*

The following are the explanations for each of the columns displayed by the ***getcfg*** utility:

Driver - the name of the driver used to access the board. Driver name is one of the following:

- sbs334 - the sbs334 device driver which supports SBS334, SBS370, and SBS372 boards.
- pci334 - the pci334 device driver which supports PCI334, PCI370, and PCI372 boards.
- pci3xpq - the pci3xpq device driver which supports PCI370PQ and PCI372PQ boards.
- artic8260 -- The artic8260 driver that supports the ARTIC1000 and ARTIC2000 boards.

Board Type - the type of the board. Board type is one of the following:

- sbs334 - the sbus SS7 controller which supports up to four 64 Kbps links.
- sbs370 - the common name for sbus sbs370 (T1) and sbs372 (E1) SS7 controllers which support up to four 64 Kbps links over T1/E1 spans.
- pci334 - the pci bus SS7 controller which supports up to four 64 Kbps links.
- pci370 - the pci bus SS7 controller which supports up to four 64 Kbps links over T1 spans.

- pci372 - the pci bus SS7 controller which supports up to four 64 Kbps links over E1 spans.
- pci370pq - the pci bus SS7 controller which supports up to sixteen 64 Kbps links over T1 spans.
- pci372pq - the pci bus SS7 controller which supports up to sixteen 64 Kbps links over E1 spans.
- artic1000 - the CompactPCI bus SS7 controller with 32 MB on board RAM which supports up to sixtyfour 64 Kpbs links over E1/T1 spans.
- artic2000 - the PCI bus SS7 controller with 32 MB on board RAM which supports up to sixtyfour 64 Kpbs links over E1/T1 spans.

Slot - the physical number of slot carrying the board. Its value depends on the hardware configuration of the host computer.

Instance - the instance number of the board among other boards of its type. its value can be in the range of 0 to 7.

State - the state of the device driver. Applicable only to AIX systems.

SAMPLE OUTPUT

```
Solaris version output sample (CompactPCI bus):

Driver    Board Type Slot   Instance
--------- ---------- ------ --------
cpc3xpq   cpc370pq   1      0
cpc3xpq   cpc372pq   2      1
pmc8260   pmc8260    3-pmc1 0
pmc8260   pmc8260    3-pmc2 1
artic8260 artic1000  4      0

Solaris version output sample (PCI bus):

Driver    Board Type Slot   Instance
--------- ---------- ------ --------
pci334    pci370     1      0
pci334    pci334     3      1
pci3xpq   pci372pq   2      0
pci3xapq  pci372apq  4      0
pmc8260   pmc8260    5      0
artic8260 artic2000  6      0

Solaris version output sample (Sbus):

Driver    Board Type  Slot  Instance
--------  ----------  ----  --------
sbs334    sbs370       1     0
```



```
sbs334    sbs334      3     1
sbs334    sbs370      2     0

AIX version output sample:

Driver   Board Type  Slot  Instance   State
--------  ----------  ----  --------  ---------
pci334    pci370      1     0         Available
pci334    pci334      3     1         Available
pci3xpq   pci372pq    2     0         Available
pci3xapq  pci372apq   4     0         Available
```

AIX output sample:

Driver Board TypeSlotInstanceState
-------- --------------------------------
pci334 pci3701 0 Available
pci334 pci3343 1 Available
pci3xpq pci372pq20 Available

7.3 APMUtilities

7.3.1 apm_audit

NAME

apm_audit Audit ***apmd*** IPC resources.

SYNOPSIS

apm_audit

DESCRIPTION

apm_audit Used to audit the UNIX Inter Process Communication (IPC) resources used by the ***apmd*** daemon. These resources include the message queues, shared memory segments, and semaphores acquired by ***apmd*** at start-up time and used by itself or with other UNIX processes attached to the ***apmd*** domain.

The auditing procedure conducted by the ***apm_audit*** utility simply involves a detailed listing of all the IPC resources allocated by the ***apmd*** daemon and the contents of information displayed is very much the same as that of the UNIX ***ipcs*** command. The ***apm_audit*** utility is intended to detect potential anomalies with IPC resources associated with the APM subsystem (e.g., accumulation in message queues). No corrective action is taken as part of the auditing procedure conducted by ***apm_audit***.

Related Information

• Section 6.2.7, **apmd** on page 6-14
• Section 7.3.8, **apm_start** on page 7-49

7.3.2 apm_getstate

NAME

apm_getstate Retrieves current ***apmd*** run state.

SYNOPSIS

apm_getstate [-c|s|x] [-h host***]***

DESCRIPTION

apm_getstate Retrieves the current run state of the ***apmd*** daemon process on a specified host, operating in the NewNet Distributed7 environment.

-c Retrieves run state of the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Retrieves the run state of the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variable must be set to an appropriate value.

-x Retrieves the run state of the NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-h host Retrieves the run state of ***apmd*** on a remote host identified by ***host***. The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the run state of the ***apmd*** on the local host will be retrieved.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.7, **apm_setstate** on page 7-47

7.3.3 apm_kill

NAME

apm_kill Sends a signal to a process.

SYNOPSIS

apm_kill [-c|s|x] [-l] [-n** signum**] [-h** host**] -p|g|t pid|gid|tag

DESCRIPTION

apm_kill Places a request to send a UNIX signal to a process or a group of processes executing in the NewNet Distributed7 environment.

-c Assumes the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Assumes the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variable must be set to an appropriate value.

-x Assumes the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-l Prints a list of symbolic signal names supported by ***apm_kill***. This list includes only commonly used signals and is only a subset of those supported by the UNIX *kill* command. The list shows the names without the SIG prefix.

-n signum Sends the signal identified in ***signum*** to the specified process(es). The valid entries for ***signum*** can be numeric or the symbolic names that are listed by the ***-l*** option. If no value is provided, the default signal, *SIGTERM*, is sent, which normally kills processes that do not catch or ignore the signal.

-h host Sends a signal to a process executing on a remote host identified by ***host***.The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the local host is the default.

-p pid Sends the signal to the process whose UNIX process ID is ***pid***.

-g gid Sends the signal to the processes whose group ID is ***gid***. This group ID is the one specified in the ***apmd*** configuration file, i.e. ***apmconfig***, and it could be different from the process's UNIX group ID.

-t tag Sends the signal to the process identified by ***tag***. This tag is specified in the ***apmd*** configuration file, i.e., ***apmconfig***. The tag of a process can be obtained by looking at the configuration file of the appropriate host or by executing ***apm_ps***.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.

•f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.

• If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Related Information

• Section 6.2.7, **apmd** on page 6-14
• Section 7.3.9, **apm_stop** on page 7-51
• Section 7.3.4, **apm_killall** on page 7-40
• Section 7.3.3, **apm_kill** on page 7-38

7.3.4 apm_killall

NAME

apm_killall Sends a signal for all processes to terminate.

SYNOPSIS

apm_killall [-c|s|x] [-h** host**]

DESCRIPTION

apm_killall Requests the ***apmd*** on a specific host to terminate all non-failsafe processes and then initialize its run state.

-c Assumes the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Assumes the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Assumes the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-h host Terminates processes executing on a remote host identified by the ***host*** argument. The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the local host is the default.

The ***apmd*** terminates a process by first sending a *SIGTERM* signal to it. If the process does not terminate within 3 seconds, ***apmd*** sends a *SIGKILL* signal. After all processes are terminated, ***apmd*** will change its run state to the default initialization state specified in the *initdefault* entry of the ***apmd*** configuration file. If an *initdefault* entry does not exist, then ***apmd*** enters a run state as follows:

- In AccessCRP environments, it moves to the **D** state.
- In AccessSERVICES environments, it moves to the **A** state.
- In NewNet Distributed7 environments, it moves to the **init** state.

Processes that are defined to operate in the failsafe mode (by their entry in the ***apmd*** configuration file) are not effected by the operations of this utility. Examples of fail-safe processes are ***mlogd***, ***spmd***, and ***netd***.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.9, **apm_stop** on page 7-51
- Section 7.3.3, **apm_kill** on page 7-38
- Section 7.3.7, **apm_setstate** on page 7-47

7.3.5 apm_ps

NAME

apm_ps Reports process status.

SYNOPSIS

apm_ps [-c|s|x] [-l]

DESCRIPTION

apm_ps Retrieves and displays information about active processes that were spawned by the ***apmd*** daemon on the local host machine. Only the processes spawned based on the configuration file are included in the output. Processes spawned dynamically through the ***apm_spawn()*** function are not part of the output. The elements included in the output are described in Table 7-4. This data is maintained by the ***apmd*** daemon and stored in a process table located on the local host machine.

-c Assumes the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Assumes the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Assumes the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-l Prints additional information about each process including the internal keys assigned to the process, its group ID, and various states that the ***apmd*** daemon should switch to based on process behavior.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Note*: The output of* ***apm_ps*** *is a snapshot that is true only for a split-second because of the dynamic nature of the information. Therefore, it may not be completely accurate after it is displayed.*

OUTPUT VALUES

The output of this command contains several columns of information, depending on the options used in the command. The following table contains the column headings of the output, the meaning of the column, and possible values that may be displayed. The fields that are displayed depend on the command options used.

Table 7-4: apm_ps Output Description

Column Heading	Possible Values	Description
KEY		Internal key assigned to the process on the local host. Identifies the slot allocated for the process in the dynamic process table maintained by the ***apmd*** daemon.
RKEY		Registration related key assigned to the process on the local host. Identifies the slot allocated for the process in the NewNet Distributed7 internal registration table. It corresponds to the KEY field in the output of the ***ebs_ps*** command. For processes that are spawned by ***apmd*** but do not register with the NewNet Distributed7 environment, this field is set to 0.
PID		UNIX process ID assigned to the process on its host machine.
GID		Group ID assigned to the process in the ***apmd*** configuration file [if any]. This group ID could be different from the UNIX group ID of the process.
PROG		Program ID assigned to the process in the ***apmd*** configuration file.
TAG		Tag associated with the process. Constructed by combining the process tag information specified in the ***apmd*** configuration file with the node name of the host machine. All processes executing on a particular host must be assigned unique process tags.
ACTION		The action mode defined for the process in the ***apmd*** configuration file. Key words for this field are described in ***apmconfig***.
STATUS		The status of the process. An `*` next to a value in this field indicates the ***apmd*** daemon is no longer executing. Information retrieved/displayed may not be accurate.
	ok	Process executing normally.
	busy	***apmd*** is busy executing a scenario that involves the process.
	exited	Process terminated with a zero exit code.
	failed	Process terminated with a non-zero exit code. Depending on the ACTION field defined for the process in the ***apmd*** configuration file, process may be re-spawned by ***apmd***.
	killed	Process terminated by an unexpected signal. Depending on the ACTION field defined for the process in the ***apmd*** configuration file, process may be re-spawned by ***apmd***.
	stopped	Process has stopped.
HBSTAT		Heartbeat status.
	-	Process will not exchange heartbeat messages with ***apmd***.
	ok	Process is responding to the heartbeat request messages generated by ***apmd*** on a regular basis.
	failed	Process failed to respond to the heartbeat request messages generated by ***apmd*** on a regular basis. Process will be killed by ***apmd***.
	n/a	***apmd*** unable to send heartbeat request messages to the process since process has not yet registered with the NewNet Distributed7 environment.
RETRY		Number of times process has been re-spawned by ***apmd*** following the initial start-up.
SSTATE		Start-up and steady-state success states for the process defined in the ***apmd*** configuration file. The two states are separated from each other by the ***:*** character. The **-** character is used to identify *don't care* states.
FSTATE		Start-up and steady-state failure states for the process defined in the ***apmd*** configuration file. The two states are separated from each other by the ***:*** character. The **-** character is used to identify *don't care* states.
HSTATE		Start-up and steady-state hopeless states for the process defined in the ***apmd*** configuration file. The two states are separated from each other by the ***:*** character. The **-** character is used to identify *don't care* states.

Table 7-4: apm_ps Output Description

Column Heading	Possible Values	Description
ASTATE		Start-up and steady-state positive acknowledgment states for the process defined in the ***apmd*** configuration file. The two states are separated from each other by the ***:*** character. The **-** character is used to identify *don't care* states.
NSTATE		Start-up and steady-state negative acknowledgment states for the process defined in the ***apmd*** configuration file. The two states are separated from each other by the ***:*** character. The **-** character is used to identify *don't care* states.
ESTATE		Execution states for the process defined in the ***apmd*** configuration file. Multiple execution states [if any] are separated from each other by the | character.

Related Information

- Section 7.2.13, **ebs_ps** on page 7-21
- Section 6.2.7, **apmd** on page 6-14
- apm_init() [API call]

7.3.6 apm_report

NAME

apm_report Generates a log report.

SYNOPSIS

apm_report [-b** mmddyy**] [-e** mmddyy**] [-p pri] [-d** dir**] [-f** file**] [-m|a] [**hostname(s)**]

DESCRIPTION

apm_report Collects information from the ***mlogd*** log files stored on the individual host machines in the NewNet Distributed7 environment and creates a report. This utility organizes the records chronologically, searches the records for user-specified information, generates customized log reports, and displays the reports on the standard output. Without options, ***apm_report*** generates a report that contains all log messages existing in the master log files on the local host, up to the current point in time. Otherwise, the contents of the report depend on the options specified.

-b mmddyy Includes all log messages reported to ***mlogd*** on or after the specified date. The date is specified in the *mmddyy* format, with the month, day of the month, and year expressed in 2-digit numerals (as in the UNIX date(1) command).

-e mmddyy Includes all log messages reported to ***mlogd*** on or before the specified date. The date is specified in the *mmddyy* format, with the month, day of the month, and year expressed in 2-digit numerals (as in the UNIX `date(1)` command).

-p pri Includes log messages reported to ***mlogd*** with the specified priority levels only. Without this option, the default includes log messages at all priority levels. The ***pri*** argument may contain any combination of the following values:

- 1:Informational messages.
- 2:Messages at minor priority level.
- 3:Messages at major priority level.
- 4:Messages at critical priority level.

-f file Includes only the log messages that were generated by the executable whose source file is specified in the ***file*** argument. By default, all log messages are included, regardless of the name of the source file.

-d dir Locates master/alternative log files on specified host machines. By default, master/alternative log files are located under ***mlog*** and ***alog*** directories, respectively, in the ***$EBSHOME/access/RUN*** directory. If the ***dir*** is specified, the master/alternative log files are expected to be located under the ***dir/mlog*** and ***dir/alog*** directories, respectively.

-m Includes only the log messages from the master log files (default).

-a	Includes only the log messages from the alternate [secondary] log files.
hostname	Identifies the host machine(s) whose log files should be used for generating the report. If multiple hosts are specified, each hostname must be separated from the others by white space. If a hostname is not specified, the report will be generated for the local host only.

Important*: The $****EBSHOME*** *environment variable must be set before invoking this utility.*

FILES

$EBSHOME/access/RUN/mlog/MLog.*mmddyy*

$EBSHOME/access/RUN/alog/ALog.*mmddyy*

EXAMPLES

The following are example command lines for ***apm_report***.

- To display all *critical* log messages generated on the local host, up to the current time, and stored in the master log files:

  apm_report -p 4

- To display all *major* and *critical* log messages stored in the alternate log files that were generated since August 14, 1994 by the executable, named *sample.c*, which is on the host, *phantom*.

  apm_report -b 081494 -p 34 -f sample.c -a phantom

- To display all *minor*, *major*, and *critical* log messages in the master log files that were generated on the hosts, *sun* and *mars*, between the dates September 3, 1994 and December 7, 1994.

  apm_report -b 090394 -e 120794 -p 234 sun mars

Important: *Refrain from executing the* ***apm_report*** *utility on a live system (where several user-space application programs are running) as it may consume a large amount of CPU resources to search through and process the event log files accumulated on the system, which is likely to degrade the performance of user-space applications running on the system.*

Related Information

- date(1)
- Section 6.2.11, **mlogd** on page 6-22
- Section 7.2.14, **ebs_report** on page 7-25

7.3.7 apm_setstate

NAME

apm_setstate Manipulates ***apmd*** run state.

SYNOPSIS

apm_setstate [-c**|**s**|**x] [-h host***] newstate***

DESCRIPTION

apm_setstate Changes the current run state of the ***apmd*** daemon process on a host operating under NewNet Distributed7 environment. The change in state causes ***apmd*** to execute the ***apmconfig*** configuration file.

-c Changes the run state of the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Changes the run state of the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Changes the run state of the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-h host Changes the run state of ***apmd*** on a remote host identified by ***host***. The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the local host is the default.

newstate Changes the run state to the provided state. The entries in the configuration file (***apmconfig*** or ***apmconfig.old***) which have an execution state matching this state will be executed.

Depending on which entries of the configuration file are executed, a change in state may result in a change in the environment (e.g. new processes started and/or existing processes terminated). Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Important*: The* ***apmd*** *daemon may reject a request to change its current run state if it is busy executing a scenario, possibly at a different run state. If this happens,* ***apm_setstate*** *will fail with an appropriate error code.*

Related Information

- Section 6.2.7, **apmd** on page 6-14

- Section 7.3.2, **apm_getstate** on page 7-37
- Section 7.3.4, **apm_killall** on page 7-40
- Section 7.3.16, **apm_update** on page 7-65

7.3.8 apm_start

NAME

apm_start Starts the ***apmd*** daemon.

SYNOPSIS

apm_start [-c|s|x] [-f cfgfile***]***

DESCRIPTION

apm_start Sets up the trace shared memory segment of the default size using ***apm_trinit*** and then starts the ***apmd*** daemon process on the local host. If the trace shared memory already exists, then only ***apmd*** will be started. Upon start-up, ***apmd*** will create and manage a set of processes as defined in the configuration file.

-c Starts the AccessCRP (Call Routing Point) version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values prior to program execution. This version supports multiple application domains on a single host. The settings of the environment variables are used to invoke the ***apmd*** instance.

-s Starts the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variable must be set to an appropriate value prior to program execution. This version supports multiple application domains on a single host. The setting of the environment variable is used to invoke the ***apmd*** instance.

-x Starts the normal NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value prior to program execution. This version does not support multiple application domains on a single host.The ***apmd*** daemon executes in a local-exclusive mode, i.e., only one instance of this version of ***apmd*** may be executing on a host.

-f cfgfile Use the configuration file specified by ***cfgfile*** instead of the default. By default, ***apmd*** uses the ***apmconfig*** or ***apmconfig.old*** configuration file under an appropriate release directory. (See ***apmd***.)

The ***apmd*** supports different types of application environments, as described by the options (c, s, x). If the user does not explicitly specify one of the options, ***apmd*** will determine the appropriate environment based on environment variable settings and the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes an AccessCRP environment.
- If none of the above environment variables are set but ***$EBSHOME*** is set, it assumes a basic NewNet Distributed7 environment.

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.9, **apm_stop** on page 7-51

7.3.9 apm_stop

NAME

apm_stop Terminates the ***apmd*** daemon.

SYNOPSIS

apm_stop [-c|s|x] [-h host***]***

DESCRIPTION

apm_stop Terminates the ***apmd*** daemon and the processes that it had spawned through the configuration file. The daemon and its processes can be stopped on any host machine operating under the NewNet Distributed7 environment.

-c Assumes the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Assumes the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Assumes the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-h host Stops process and ***apmd*** on a remote host identified by ***host***. The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the local host is the default.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

The ***apmd*** terminates the processes that it has spawned differently for the different ***apmd*** versions.

AccessSERVICES and AccessCRP Versions

1. ***apmd*** warns all of its active processes by sending the *SIGTERM* signal to them.
2. ***apmd*** waits up to 60 seconds for the processes to exit.
3. If a process still has not exited, ***apmd*** will warn this process one more time by sending a second *SIGTERM* signal to it
4. ***apmd*** waits up to 3 seconds for the process to terminate.
5. If the process still has not exited, ***apmd*** terminates the process forcefully by sending a *SIGKILL* signal.

6. After all processes exit and/or are terminated, ***apmd*** terminates itself with an exit code of zero.

NewNet Distributed7 **Version**

The basic version of ***apmd*** terminates processes that are spawned by it and other processes currently using the NewNet Distributed7 platform.

1. ***apmd*** sends the *SIGTERM* signal to processes that are not currently registered with the NewNet Distributed7 environment. (This is the only notification these processes will receive.)
2. ***apmd*** issues a local system software shutdown request to its active processes.
3. ***apmd*** waits up to 60 seconds for its processes to exit.
4. If any of its processes still has not exited, ***apmd*** will warn each process by sending a *SIGTERM* signal to it
5. ***apmd*** waits up to 3 seconds for its remaining process(es) to terminate.
6. If a process still has not exited, ***apmd*** terminates the process forcefully by sending a *SIGKILL* signal.
7. After all processes exit and/or are terminated, ***apmd*** terminates itself with an exit code of zero.

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.8, **apm_start** on page 7-49
- Section 7.3.4, **apm_killall** on page 7-40
- Section 7.3.3, **apm_kill** on page 7-38
- Section 7.3.7, **apm_setstate** on page 7-47

7.3.10 apm_trcapture

NAME

apm_trcapture Captures trace output.

SYNOPSIS

apm_trcapture [-c|s|x] [[-o] | [[-d dir|-f file] [-l limit]]] [{-m|n} mask0,...,mask63]
[{-p|r} progid0,...,progid511***]***

DESCRIPTION

apm_trcapture Displays the current values of the trace mask settings on the local host to the standard output. By default, trace mask settings are displayed for all program IDs, if at least one of the 64 trace masks is currently activated for any of the 512 program IDs.

-c Captures contents of the trace buffer for the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Captures contents of the trace buffer for the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Captures contents of the trace buffer for the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-o Displays trace information captured on ***stdout*** only. When this option is specified, captured trace messages will not be stored in a log file.

Note*: This option cannot be used in combination with any of the following options:*
[-d dir]
[-f file]
[-l limit]

-m mask0,...,mask63 Captures the trace message(s) specified in ***masks***. A maximum of 64 trace masks can be specified (0-63). Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character. All mask settings may be modified by specifying the string **all** as the ***masks*** argument.

-n mask0,...,mask63 Captures all trace messages EXCEPT the one(s) specified in ***masks***. Masks can be specified in the range from 0 to 63. Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character.

-p progid0,...,progid511 Captures the trace message(s) for the processes with the program IDs specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character (see example).

-r progid0,...,progid511 Captures the trace message(s) for all processes EXCEPT the ones whose program IDs are specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character.

-d dir Saves captured trace log files on specified host machines. By default, trace log files are located under ***tracelog*** directory in the ***$EBSHOME/access/RUN*** directory. If the ***dir*** is specified, the trace log files are stored under the ***dir/tracelog*** directory. If ***dir*** directory (or ***tracelog*** sub-directory) does not exist, ***apm_trcapture*** will make an attempt to create all necessary directories.

Note*: This option cannot be used in combination with the [-f file] or [-o] options.*

-f file Saves captured trace log information of filename specified by ***file*** argument. By default, all trace log files will be named with the "TLog" prefix and contain the process ID of the executing program as an extension. This option gives users the flexibility to name trace log files using their own naming conventions and store them in their preferred directories.

Note*: This option cannot be used in combination with the [-d dir] or [-o] options.*

-l limit Limits the number of trace statements to be stored in ***tracelog*** file at any given time to the value specified via ***limit*** argument. Once the specified limit is exceeded, the file is truncated to zero length and writing continues from the beginning of file. This option allows users to control the maximum size of ***tracelog*** files generated by ***apm_trcapture*** utility.

Note*: This option cannot be used in combination with the [-o] option.*

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

EXAMPLES

apm_trcapture -m 1,3,5 -p 0,-,23

Captures trace messages corresponding to trace categories 1, 3, and 5 for program ID's 0 through 23.

apm_trcapture -m 10,20 -p all

Captures trace messages for trace categories 10 and 20 for all program ID's.

apm_trcapture -n 1,-,15 -r 0,-,200

Captures trace messages for all trace categories other than 1 through 15 and for all processes whose program ID's are above 200.

FILES

`$EBSHOME/access/RUN/tracelog/TLog.pid`

Related Information

- Section 7.3.13, **apm_trinit** on page 7-59
- Section 7.3.11, **apm_trclear** on page 7-56
- Section 7.3.12, **apm_trgetmask** on page 7-57

7.3.11 apm_trclear

NAME

apm_trclear Clears the contents of the trace shared memory.

SYNOPSIS

apm_trclear [-c|s|x]

DESCRIPTION

apm_trclear Clears the contents of the local host's IPC shared memory segment which is used for tracing the execution of application programs. This memory segment is also referred to as the *trace buffer*. (Tracing occurs through the ***libapm*** trace macros.)

-***c*** Assumes the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-***s*** Assumes the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-***x*** Assumes the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Important: *The **apm_trclear** utility should be used with care. It will permanently erase all trace messages logged for the specified environment.*

Related Information

- Section 7.3.13, **apm_trinit** on page 7-59
- Section 7.3.12, **apm_trgetmask** on page 7-57
- Section 7.3.14, **apm_trsetmask** on page 7-61
- Section 7.3.15, **apm_trshow** on page 7-63
- Section 7.3.10, **apm_trcapture** on page 7-53
- apm_trace() API CALL

7.3.12 apm_trgetmask

NAME

apm_trgetmask Retrieves trace mask settings.

SYNOPSIS

apm_trgetmask [-c|s|x] [-e] [{-p|r} progid0,...,progid511***]***

DESCRIPTION

apm_trgetmask Displays the current values of the trace mask settings on the local host to the standard output. By default, trace mask settings are displayed for all program IDs, if at least one of the 64 trace masks is currently activated for any of the 512 program IDs.

-c Displays trace mask settings for the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Displays trace mask settings for the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Displays trace mask settings for the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-e Displays any ***pattern*** strings associated with individual trace masks for specified program IDs. The pattern strings comprise regular expressions and are set forth with the *apm_trsetmask* command line utility. If a pattern string exists and there is a match between the actual trace message contents and the regular expression, trace statements associated with the corresponding trace mask will be generated.

-p progid0,...,progid511 Displays the trace mask settings for the processes with the program IDs specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character (see example).

-r progid0,...,progid511 Displays the trace mask settings for all processes EXCEPT the ones whose program IDs are specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.

•f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.

• If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

EXAMPLES

apm_trgetmask -p 1,3,5,10,-,20

Retrieves the current trace mask settings for program IDs 1, 3, 5, and 10 through 20.

Related Information

• Section 7.3.13, **apm_trinit** on page 7-59
• Section 7.3.11, **apm_trclear** on page 7-56
• Section 7.3.14, **apm_trsetmask** on page 7-61
• Section 7.3.15, **apm_trshow** on page 7-63
• Section 7.3.10, **apm_trcapture** on page 7-53
• apm_trace(),

7.3.13 apm_trinit

NAME

apm_trinit Initializes IPC shared memory.

SYNOPSIS

apm_trinit [-f] [-c|s|x] [-m** maxcnt**]

DESCRIPTION

apm_trinit Creates and initializes the IPC shared memory segment used during the tracing of application programs on the local host.

-f Forces the initialization of the trace shared memory if it already exists. Normally, a user is not allowed to re-initialize the trace shared memory segment if it already exists.

-c Initializes trace shared memory for the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Initializes trace shared memory for the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to appropriate values.

-x Initializes trace shared memory for the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-m maxcnt Specifies the size of the IPC shared memory segment to be initialized. The ***maxcnt*** argument specifies the size in number of messages. The size should never be made less than the default size of 1000 messages.

The trace shared memory segment is a circular buffer storing trace messages. The trace messages are generated by applications using the ***libapm*** trace macros. The shared memory segment must be initialized prior to the start-up of the ***apmd*** daemon and any other application program that will use the ***libapm***.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

***Important**: The **apm_trinit** utility cannot be used while the **apmd** daemon on the local host is running. It should only be used before the **apmd** daemon is started.*

Related Information

- Section 7.3.11, **apm_trclear** on page 7-56
- Section 7.3.12, **apm_trgetmask** on page 7-57
- Section 7.3.14, **apm_trsetmask** on page 7-61
- Section 7.3.15, **apm_trshow** on page 7-63
- Section 7.3.10, **apm_trcapture** on page 7-53

- apm_trace() API Call

7.3.14 apm_trsetmask

NAME

apm_trsetmask Sets a trace mask.

SYNOPSIS

apm_trsetmask [-c|s|x] [-a] [{-m|n} mask0,...,mask63] [{-p|r} progid0,...,progid511] [pattern]

DESCRIPTION

apm_trsetmask Sets a trace mask or changes the current values of the trace mask settings available on the local host. Without any options, all 64 trace masks for all 512 program IDs will be set. The ***pattern*** command line argument specifies pattern strings of regular expressions for selective tracing under a particular trace mask setting and program ID.

-c Modifies mask setting for the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Modifies mask setting for the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Modifies mask setting for the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-a Appends the specified trace mask(s) to any existing ones.

-m mask0,...,mask63 Modifies the trace mask(s) specified in ***masks***. A maximum of 64 trace masks can be specified (0-63). Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character. All mask settings may be modified by specifying the string **all** as the ***masks*** argument.

-n mask0,...,mask63 Modifies all trace masks EXCEPT the one(s) specified in ***masks***. Masks can be specified in the range from 0 to 63. Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character.

-p progid0,...,progid511 Manipulates the trace mask settings for the processes with the program IDs specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character (see example).

-r progid0,...,progid511 Manipulates the trace mask settings for all processes EXCEPT the ones whose program IDs are specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

•f ***$DOMID*** is set, it assumes an AccessSERVICES environment.

•f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.

• If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

NewNet Distributed7 system software supports a total of 64 separate trace masks for each program ID. A total of 512 program IDs are available on a given host. Therefore, 512 processes may be uniquely traced and identified. If more than 512 processes co-exist, some processes will be assigned the same program ID. The result of such a situation is that trace messages generated by both programs that share an ID would be displayed on the screen together.

EXAMPLES

apm_trsetmask -m 1,3,5 -p 0,-,23

Sets the masks corresponding to trace categories 1, 3, and 5 for program ID's 0 through 23.

apm_trsetmask -m 10,20 -p all

Sets trace categories 10 and 20 for all program IDs.
Note: The same result could also be achieved by not specifying the -p option.

apm_trsetmask -m 3 -p 5 ’^lset=[A-Z]+ link=[0-7] ’

Sets the mask corresponding to trace category 3 for program ID 5 selectively (if the trace message contents match the regular expression specified by the ***pattern*** argument).

Related Information

• Section 7.3.13, **apm_trinit** on page 7-59
• Section 7.3.11, **apm_trclear** on page 7-56
• Section 7.3.12, **apm_trgetmask** on page 7-57
• Section 7.3.15, **apm_trshow** on page 7-63
• Section 7.3.10, **apm_trcapture** on page 7-53
• apm_trace() API Call

7.3.15 apm_trshow

NAME

apm_trshow Displays the trace output.

SYNOPSIS

apm_trshow [-c|s|x] [{-m|n} mask0,...,mask63***] [{-p|r}*** prodig0,...,progid511***]***

DESCRIPTION

apm_trshow Displays the trace information that is in the IPC shared memory segment of the local host. (Tracing shows the execution of application programs.) Without any options, all trace messages in the shared memory segment will be displayed on the standard output. The options allow selective display of certain messages from memory.

-c Displays contents of the trace buffer for the environment that is controlled by the AccessCRP version of ***apmd***. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Displays contents of the trace buffer for the environment that is controlled by the AccessSERVICES version of ***apmd***. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Displays contents of the trace buffer for the environment that is controlled by the basic NewNet Distributed7 version of ***apmd***. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-m mask0,...,mask63 Displays the trace mask(s) specified in ***masks***. A maximum of 64 trace masks can be specified (0-63). Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character. All mask settings may be modified by specifying the string **all** as the ***masks*** argument.

-n mask0,...,mask63 Displays all trace masks EXCEPT the one(s) specified in ***masks***. Masks can be specified in the range from 0 to 63. Multiple trace masks must be separated from each other with the **,** character and no white space. A range of masks may be specified with the **-** character.

-p progid0,...,progid511 Displays the trace mask settings for the processes with the program IDs specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character (see example).

-r progid0,...,progid511 Displays the trace mask settings for all processes EXCEPT the ones whose program IDs are specified in ***progid***. A maximum of 512 program IDs can be specified (0-511). Multiple program IDs must be separated from each other using the **,** character with no white space. A range of program IDs may be specified using the **-** character.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

•f ***$DOMID*** is set, it assumes an AccessSERVICES environment.

•f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.

• If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

EXAMPLES

apm_trshow -m 1,3,5 -p 0,-,23
: Displays the trace messages for the trace categories 1, 3, and 5 for program ID's 0 through 23.

apm_trshow -m 10,20 -p all
: Displays the trace messages for trace categories 10 and 20 for all program ID's.

apm_trshow -n 1,-,15 -r 0,-,200
: Displays trace messages for all trace categories other than 1 through 15 and for all processes whose program IDs are above 200.

Related Information

- Section 7.3.13, **apm_trinit** on page 7-59
- Section 7.3.11, **apm_trclear** on page 7-56
- Section 7.3.12, **apm_trgetmask** on page 7-57
- Section 7.3.14, **apm_trsetmask** on page 7-61
- Section 7.3.10, **apm_trcapture** on page 7-53

- apm_trace() API Call

7.3.16 apm_update

NAME

apm_update Informs ***apmd*** of any changes in the configuration.

SYNOPSIS

apm_update [-c*|s|x****] [-h*** host***]***

DESCRIPTION

apm_update Notifies the ***apmd*** daemon process on a specified host that changes in the ***apmd*** configuration file have occurred. It causes the ***apmd*** to re-read and re-execute the instructions in the configuration file, based on its current state. It copies the updated file into its internal process table.

-c Assumes the AccessCRP version of ***apmd*** and re-reads the ***apmconfig.old*** configuration file. The ***$PRODID*** and ***$RUNID*** environment variables must be set to appropriate values.

-s Assumes the AccessSERVICES version of ***apmd*** and re-reads the ***apmconfig.old*** configuration file. The ***$DOMID*** environment variables must be set to an appropriate value.

-x Assumes the basic NewNet Distributed7 version of ***apmd*** and re-reads the ***apmconfig*** configuration file. The ***$EBSHOME*** environment variable must be set to an appropriate value.

-h host Causes an update of the ***apmd*** on a remote host identified by ***host***. The ***apmd***s on both the remote and local host must be operational since the request is placed through the local ***apmd***. If the option is not provided, the local host is the default.

Since ***apmd*** supports multiple versions, if the user does not explicitly specify one in the command, then ***apmd*** determines the version by the following logic:

- f ***$DOMID*** is set, it assumes an AccessSERVICES environment.
- f ***$DOMID*** is not set but ***$PRODID*** and ***$RUNID*** are set, it assumes the AccessCRP (Call Routing Point) environment.
- If none of the above environment variables are set, but ***$EBSHOME*** is set, it assumes the basic NewNet Distributed7 environment.

Important*: The* ***apmd*** *daemon may reject a request to re-execute and update the configuration file if it is busy executing a scenario. If this happens,* **apm_update** *will fail with an appropriate error code.*

Related Information

- Section 6.2.7, **apmd** on page 6-14
- Section 7.3.7, **apm_setstate** on page 7-47

7.4 Signaling Gateway Application Utilities

7.4.1 sg_pkgrm

NAME

sg_pkgrm Removes a version of the Signaling Gateway package software.

SYNOPSIS

sg_pkgrm version

DESCRIPTION

The **sg_pkgrm** command removes a version of the Signaling Gateway package (package name SGcore). If there are files left in the software directory after removing the software, (files that are not part of the original package), then the software directory remains on the disk and **sgadm** user account is preserved. Otherwise, the entire software directory and user account are removed.

version Version number of the software to be removed. Use the UNIX **pkginfo** command to find out what version(s) are installed if you are not sure of the software version:

pkginfo | grep SGcore

7.4.2 sg_start

NAME

sg_start Starts the Signaling Gateway software.

SYNOPSIS

sg_start [d7] [sp[0-7]] [sc[0-7]] [om[0-7]] [ip] [sgp]

DESCRIPTION

sg_start The sg_start command starts the Signaling Gateway and NewNet Distributed7 software. As described in Section 4.2.4 on page 4-3, the Signaling Gateway system consists of the *basic platform processes*, *SS7 node processes* and *Signaling Gateway processes*. This command allows the user to start the system in different phases in which one or more subset of processes are running. The following states can be used as command line arguments to start Signaling Gateway:

d7 Starts only the basic platform processes on the local host.

sp[0-7] Starts the MTP3 process for a signaling point on the local host. Valid signaling points are 0 through 7. *All basic platform processes are also started automatically, if they have not already been started, when this state is specified.*

sc[0-7] Starts the SCCP process on the local host. Valid signaling points are 0 through 7. All basic platform processes and MTP3 process are started automatically, if they have not already been started, when this state is specified.

om[0-7] Starts the OMAP process for a signaling point on the local host. Valid signaling points are 0 through 7. All basic platform processes are started automatically, if they have not already been started, when this state is specified.

ip Starts the SCTP process on the local host.

sgp Starts the SGP process for a signaling point on the local host. Basic platform processes and SCTP process are also started automatically, if they have not already been started, when this state is specified.

EXAMPLE

1. To start MTP3 and SCCP on signaling point 0

$ sg_start sc0

This command starts basic platform processes, MTP3 for signaling point 0, and SCCP for signaling point 0. This is equivalent of executing:

$ sg_start d7 sp0 sc0

2. To start MTP3 and SCCP on signaling point 0, MTP3 on signaling point 1 and SGP:

$ sg_start sc0 sp1 sgp

This command starts basic platform processes, MTP3 for signaling point 0, SCCP for signaling point 0, MTP3 for signaling point 1, SCTP and SGP processes. The following command performs the same operation:

$ sg_start d7 sp0 sc0 sp1 ip asp

7.4.3 sg_stop

NAME

sg_stop Stops the Signaling Gateway software.

SYNOPSIS

sg_stop [d7] [sp[0-7]] [sc[0-7]] [om[0-7]] [ip] [sgp]

DESCRIPTION

sg_stop The sg_stop command stops the Signaling Gateway and NewNet Distributed7 software. As described in Section 4.2.4 on page 4-3, the Signaling Gateway system has the *basic platform processes*, *SS7 node processes* and *Signaling Gateway processes*. This command allows the user to stop one or more subset(s) of these processes. The following states can be used as command line arguments to stop Signaling Gateway:

d7 Stops all Signaling Gateway and NewNet Distributed7 processes on the local host.

sp[0-7] Stops the MTP3 process for a signaling point on the local host. Valid signaling points are 0 through 7. SCCP process stops automatically when this state is specified.

om[0-7] Stops the OMAP process for a signaling point on the local host. Valid signaling points are 0 through 7.

ip Stops the SCTP process on the local host. The host name can be appended to the state name to stop SCTP on a remote host. *Signaling Gateway process stops automatically when this state is specified.*

sgp Stops the SGP process on the local host.

EXAMPLE

To stop MTP3, SCCP on signaling point 0 and also the SGP process:

$ sg_stop sp0 sgp

This command is equivalent to the following:

$ sg_stop sc0 sp0 sgp

7.4.4 sg_setrelease

NAME

sg_setrelease Sets the Signaling Gateway software.

SYNOPSIS

sg_setrelease version | -i

DESCRIPTION

sg_setrelease The **sg_setrelease** command activates a specific version of the Signaling Gateway and NewNet Distributed7 software. The following arguments are supported:

version Version of the software to be activated.

-i The -i option prints the information about the currently activated software version, such as the base installation directory, Signaling Gateway version and NewNet Distributed7 version.

7.4.5 sg_trace

NAME

sg_trace Runtime tracing utility.

SYNOPSIS

sg_trace [sgp | ip]

sg_trace [-c]

sg_trace -s | -u | -m sgp | ip | all

DESCRIPTION

sg_trace The sg_trace command is used to collect traces for SCTP and Signaling Gateway processes for debugging purposes. Runtime tracing can be enabled or disabled dynamically, and traces that are collected are stored in a circular memory buffer, with new traces overwriting old ones when the buffer is full. This command supports the following command line options:

-s [ip] [sgp] The -s option enables the tracing for one or more processes.

- Specify *ip* as the argument to trace the SCTP process
- Specify *sgp* as the argument to trace the SGP process

-u [ip] [sgp] | all The -u option disables the tracing for one or more processes.

- Specify *ip* as the argument to disable tracing for SCTP
- Specify *sgp* as the argument to disable tracing for the SGP process
- Specify *all* as the argument to disable tracing for all processes

-m [ip] [sgp] | all The -m option displays the current trace statuses to show whether they are enabled or disabled on the system. The arguments are entered the same way as in the -u option (see the previous bullet).

[ip] [sgp] Specify *ip* or *sgp* to filter traces for SCTP or SGP respectively. All traces in the buffer are displayed when no argument is specified.

-c The -c option clears all existing traces in the buffer.

This page is intentionally blank.

Chapter 8: Maintenance and Troubleshooting

8.1 Overview

This chapter identifies the alarms and troubleshooting tools of the Signaling Gateway.

If necessary, the Technical Assistance Center (TAC) can be reached by phone at *(800) 416-1624 US*, or *(408) 432-2600 International*, or also by email at *nnsupport@ss8.com*.

8.2 SoftwareMaintenance

8.2.1 Configuration Backup

Periodically the configurations on each of the distributed hosts should be backed up to a tape to prevent data loss in the event of a hard disk failure or file corruption. It is recommended that the following configuration directories be backed up:

- ***$EBSHOME/access/RUN*** and ***$EBSHOME/access/RUNx***, where ***x*** is the signaling point number
- $SGHOMEsg/RUN

To back up a directory to tape, insert a tape to the tape drive connected to the host, then execute the following UNIX command:

t**ar cvf *tape_device backup_dir***

where ***tape_device*** is the device name of the tape drive, and ***backup_dir*** is the directory to be copied to tape.

8.2.2 Monitoring Alarms

By default, alarms are displayed on the console. The operator should pay attention to alarms that are displayed, or check for any outstanding alarms using MML, for example:

display-strdalm:;

The operator should also use MML to delete obsolete alarms of SET_ALARM type that are non-self-clearing, for example:

delete-strdalm:hostname=x,group=x,module=x,type=x,last_occur=x;

Periodically it is also recommended to check the alarm log files in **$EBSHOME/access/RUN/alarmlog** to observe for potential problems or any unusual system behavior.

8.2.3 Monitoring Logs

The Master Log (MLog) is used to log system events that are noteworthy, including system start/stop, major configuration changes, and faulty events. Master Log files are created daily and are stored under the **$EBSHOME/access/RUN/mlog** directory. Periodically monitoring these log files may help detect problems or unusual system behavior early.

8.3 TroubleshootingTools

8.3.1 Alarms

When problem occurs, alarms should be checked both in the alarm log file and through MML. The following tables lists the Signaling Gateway alarms and the corresponding actions that can be taken to remedy the problem. If the error condition persists, contact the Technical Assistance Center (TAC) for help. See Appendix C: Alarms for information about the NewNet Distributed7 alarm groups.

Table 8-1: SGP Alarm Group

Alarm No.	Severity	Type	Message	Operation
010101	INFO	EVENT	SGP: SGP terminated	Indicates the termination of SGP. Check mlog file to determine reason for termination. No action is required for normal termination.
010102	CRITICAL	EVENT	SGP: EBSHOME environment variable not set	Make sure Signaling Gateway is started by sgadm. If it was started by sgadm and still getting this alarm, check the sgadm’s .cshrc file and be sure that the environment variable EBSHOME is exported.
010103	CRITICAL	EVENT	SGP: Failed to read license file	Be sure that you have valid license before starting the Signaling Gateway.
010104	CRITICAL	EVENT	SGP: License feature not found in license file	Be sure that you have valid license before starting the Signaling Gateway.
010105	CRITICAL	EVENT	SGP: License key corrupted	Be sure that you have valid license before starting the Signaling Gateway.
010106	CRITICAL	EVENT	SGP: Not licensed for local host	Be sure that you have valid license before starting the Signaling Gateway.
010107	CRITICAL	EVENT	SGP: Licensed expired	Be sure that you have valid license before starting the Signaling Gateway.
010108	MAJOR	EVENT	SGP: Licensed number of AS (%d) exceeds system capacity; default to %d	Check the system capacity in section 2.5. Make sure the licensed number of AS does not exceed the capacity.

Table 8-1: SGP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
010109	MAJOR	EVENT	SGP: Licensed number of peer SG (%d) exceeds system capacity; default to %d	Check the system capacity in section 2.5. Make sure the licensed number of remote peer SG does not exceed the capacity.
01010a	CRITICAL	EVENT	SGP: MTP API initialization error	Stop the entire Signaling Gateway platform, then restart. If problem persists, contact TAC.
01010b	CRITICAL	EVENT	SGP: Gateway registration failed	Make sure the Signaling Point (SP) that you are trying to activate is in service, i.e. "sg_start spX" command has been executed for SP X. Also, run the "ebs_ps" command to verify that the process "upmd" is running. Alternative is to stop the entire Signaling Gateway platform, then restart. If problem persists, contact TAC.
01010c	CRITICAL	EVENT	SGP: SCTP IPC initialization failed	Run the "ebs_ps" command to verify that SCTP process is running. If not, run the following command to start SCTP: "sg_start ip"

Table 8-2: OAM Alarm Group

Alarm No.	Severity	Type	Message	Operation
010201	MAJOR	EVENT	SGP: Database error	SGP failed to perform database operation. Check mlog file for the failure reason. Verify that databases under $EBSHOME/RUNx/DBfiles have write permission for sgadm, where x is the signaling point number at which the SGP is running.
010202	MAJOR	EVENT	SGP: Managed Object creation error	SGP failed to create Managed Object definitions during startup. Check the mlog file for the failure reason. If the error indicates that MO already exists, then no action is required. Otherwise, verify that the MML commands can be run on all Signaling Gateway hosts. If the commands can be run then no action is required. If MML commands cannot be run, then restart the Signaling Gateway or call TAC.

Table 8-3: M3UA Alarm Group

Alarm No.	Severity	Type	Message	Operation
010301	CRITICAL	EVENT	SGP: M3UA stack initialization failed	SGP failed to initialize M3UA stack during startup. Possible reasons may be: SCTP is not running, or an incorrect network appearance. Check the mlog file for the failure reason.
010303	INFO	CLR_ALARM	SGP: SPMC for network appearance %d is up	Indicates that the SPMC for a particular network appearance is up. No action is required.
010304	INFO	SET_ALARM	SGP: SPMC for network appearance %d is down	Indicates that SPMC for a network appearance is down. No action is required.
010305	INFO	EVENT	SGP: Connection with ASP %s is up	Indicates that connection with an ASP is up. No action is required.
010306	INFO	EVENT	SGP: Connection with ASP %s is down	Indicates that connection with ASP is down. Verify that the connection did not go down due to local problem such as cable failure, incorrect IP, or congestion at the SCTP side.
010307	INFO	EVENT	SGP: Connection with ASP %s is congested at level 1	Indicates that the connection between SG and an ASP is congested at level 1.
010308	INFO	EVENT	SGP: Connection with ASP %s is congested at level 2	Indicates that the connection between SG and an ASP is congested at level 2.
010309	INFO	EVENT	SGP: Connection with ASP %s is congested at level 3	Indicates that the connection between SG and an ASP is congested at level 3.
01030a	INFO	SET_ALARM	SGP: ASP %s is down	Indicates that an ASP is down. No action is required.
01030b	INFO	CLR_ALARM	SGP: ASP %s is inactive	Indicates that an ASP is inactive. No action is required.
01030c	INFO	EVENT	SGP: ASP %s for AS %s is inactive	Indicates that an ASP is inactive for an AS. No action is required.
01030d	INFO	EVENT	SGP: ASP %s for AS %s is active	Indicates that an ASP is active for an AS. No action is required.
01030e	INFO	SET_ALARM	SGP: AS %s is down	Indicates that an AS is down. No action is required.
01030f	INFO	CLR_ALARM	SGP: AS %s is inactive	Indicates that an AS is inactive. No action is required.
010310	INFO	EVENT	SGP: AS %s is active	Indicates that an AS is active. No action is required.
010311	INFO	EVENT	SGP: AS %s is pending	Indicates that an AS is pending before going down. No action is required.

Table 8-3: M3UA Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
010312	INFO	EVENT	SGP: No more AS pending buffer, messages may be discarded	Indicates that there is no more pending buffer available to store messages for a particular AS while it is in pending mode, and that messages will be discarded. No action is required.
010313	INFO	SET_ALARM	SGP: ASP %s inhibited	Indicates that an ASP is inhibited, i.e. its traffic is temporarily suspended.
010314	INFO	EVENT	SGP: M3UA Error Indication (error code %d - %s)	Indicates an M3UA error condition. Check the alarm message for the error reason. Call TAC if it is unsolvable.

Table 8-4: M3UA Error Group

Alarm No.	Severity	Type	Message	Operation
010401	INFO	EVENT	SGP: %s M3ua error - invalid version (error code %d)	Contact TAC.
010402	INFO	EVENT	SGP: %s M3ua error - unsupported message class (error code %d)	Contact TAC.
010403	INFO	EVENT	SGP: %s M3ua error - unsupported message type (error code %d)	Contact TAC.
010404	INFO	EVENT	SGP: %s M3ua error - invalid traffic mode (error code %d)	Contact TAC.
010405	INFO	EVENT	SGP: %s M3ua error - unexpected message (error code %d)	Contact TAC.
010406	INFO	EVENT	SGP: %s M3ua error - protocol error (error code %d)	Contact TAC.
010407	INFO	EVENT	SGP: %s M3ua error - invalid stream ID (error code %d)	Contact TAC.
010408	INFO	EVENT	SGP: %s M3ua error - management blocking (error code %d)	Contact TAC.
010409	INFO	EVENT	SGP: %s M3ua error - ASP ID needed (error code %d)	Contact TAC.
01040a	INFO	EVENT	SGP: %s M3ua error - invalid ASP ID (error code %d)	Contact TAC.
01040b	INFO	EVENT	SGP: %s M3ua error - invalid parameter value (error code %d)	Contact TAC.
01040c	INFO	EVENT	SGP: %s M3ua error - field error (error code %d)	Contact TAC.
01040d	INFO	EVENT	SGP: %s M3ua error - unexpected parameter (error code %d)	Contact TAC.
01040e	INFO	EVENT	SGP: %s M3ua error - unknown destination status (error code %d)	Contact TAC.
01040f	INFO	EVENT	SGP: %s M3ua error - invalid network appearance (error code %d)	Contact TAC.
010410	INFO	EVENT	SGP: %s M3ua error - missing parameter (error code %d)	Contact TAC.

8.3.2 Log Files

The Master Log file located under ***$EBSHOME/access/RUN/mlog*** can be checked to help identify a problem when troubleshooting.

8.3.3 Runtime Tracing

The Signaling Gateway system supports runtime tracing in which trace messages are directed to a circular buffer in the shared memory. Tracing can be enabled or disabled dynamically during operation of the system. To utilize the tracing utility, the *program ID* of the process to be traced must be given.

For example, to enable tracing for process 100:

apm_trsetmask -ma -p 100

To display traces collected for process 100:

apm_trshow -p 100

To clear the trace buffer:

apm_trclear

To disable tracing for process 100:

apm_trsetmask -na -p 100

The following table lists the program ID of the application processes:

Table 8-5: Program IDs for Tracing

Process Name	Program ID
mlogd	1
spmd	2
netd	3
alarmd	4
dsmd	5
dkmd	6
upmd	7
scmd	8
sctpd	200
sgpd	100

Caution: *Tracing should not be enabled during normal operation of the system as it may significantly degrade the performance. Be sure to disable the traces when you are done troubleshooting!*

Alternatively, use the sg_trace command to simplify tracing the SCTP and SGP processes, as shown in the following examples:

Enter the following to enable tracing for SGP:

sg_trace -s sgp Ret

Enter the following to disable tracing for SGP:

sg_trace -u sgp Ret

Enter the following to display traces collected for SGP:

sg_trace sgp Ret

Enter the following to clear all traces in the buffer:

sg_trace -c Ret

Enter the following to enable tracing for SCTP:

sg_trace -s ip Ret

Appendix A: Glossary

The following table lists Signaling Gateway abbreviations, common SS7 and telecommunication industry acronyms. Brief definitions are included for frequently used terms found in the Signaling Gateway manuals.

Table A-1:

Acronym or Term	Definition
ADK	Application Development Kit - A set of programming tools and API(s), which are used by the applications developer to speed the development process.
API	Application Programmer Interface - a set of system calls or object library provided with a software product that allow a programmer to execute function provided by a provided by the purchased binary.
AS	Application Server is a logical entity serving a specific routing key, such as a CIC range or a particular Destination Point Code.
	An example of an Application Server is a virtual switch element handling all call processing for a unique range of PSTN trunks, identified by an SS7 SIO/DPC/OPC/CIC_range. Another example is a virtual database element, handling all HLR transactions for a particular SS7 DPC/OPC/SCCP_SSN combination. The AS contains a set of one or more unique Application Server Processes, of which one or more is normally actively processing traffic. Note that there is a 1:1 relationship between an AS and a Routing Key.
ASP	Application Server Process is a process instance of an Application Server. An ASP can be active or standby for an AS.
	An Application Server Process serves as an active or standby process of an Application Server (e.g., part of a distributed virtual switch or database). Examples of ASPs are processes (or process instances) of MGCs, IP SCPs or IP HLRs. An ASP has an SCTP end point and may be configured to process signaling traffic in more than one Application Server.
BHCA	Busy Hour Call Attempts - the capacity of a switching system to make connections, in terms of calls per hour. A call attempt consists of setting up and tearing down one call.
dot notation	Berkeley UNIX notation for Internet addresses. An Internet address in dot notation has one to four numbers in hexadecimal (leading 0x), octal (leading 0) or decimal. It represents a 32-bit address. Each leading number represents eight bits of the address (high byte first) and the last number represents the rest. For example, address 0x25.32.0xab represents 0x252000ab. By far the most common form is four decimal numbers, such as 146.169.22.42. Many commands accept an address in dot notation in place of a hostname.
DPC	Destination Point Code is the address of an SS7 destination node in SS7 signaling messages.
IETF	Internet Engineering Task Force is a large open international community concerned with the evolution of the Internet architecture and its smooth operation.
IP	Internet Protocol is the part of the TCP/IP protocol family dealing with addressing and routing.
IPSP	IP Server Process (IPSP) is process instance of an IP-based application. An IPSP is essentially the same as an ASP, except that it uses M3UA in a point-to-point fashion. Conceptually, an IPSP does not use the services of a Signaling Gateway.

Table A-1:

Acronym or Term	Definition
ISUP	ISDN User Part is an SS7 protocol layer used to set up and tear down calls.
GTT	Global Title Translation functionality of SCCP layer.
GUI	Graphical User Interface - a human-machine interface based on icons, two-dimensional windows/dialog boxes, and a pointing and selection device.
M3UA	MTP3 User Adaptation enables seamless operation of MTP3 user peers in SS7 and IP domains.
MTP	Message Transfer Part provides message handling functions that transfer the signaling messages to the proper signaling link or the user part and network management functions that control the current message routing and configuration of the signaling network.
MML	Man-Machine Language is a human-readable syntax based on text lines for controlling and obtaining status from a network element.
multihoming	SCTP supports multihoming, which allows an SGP and an ASP to communicate over an SCTP association using more than one IP address. Any failure of an IP address causes a fail-over to alternative addresses. Each Signaling Gateway host can have two IP addresses specified for this purpose.
NEBS	Network Equipment-Building System - a standard for product safety.
Network Appearance	The Network Appearance uniquely identifies an SS7 entity (Point Code) into an SS7 network, as presented by the SG. Its purpose is to logically separate the signaling traffic between the SG and the Application Server Processes over a common SCTP association. • This partitioning is necessary when an SG is logically partitioned to appear as end node elements in multiple separate SS7 networks, in which case there is a separate network appearance for each point code in the SS7 networks. • It is also necessary when an SG is configured as an STP hosting multiple point codes, or when configured as multiple end nodes in the same network, in which case each point code is a separate network appearance, between the SG and the Application Server Processes over a common SCTP Association. An example is where an SG is logically partitioned to appear as an element in four separate national SS7 networks. A Network Appearance implicitly defines the SS7 Point Code(s), Network Indicator and MTP3 protocol type/variant/version used in a separate SS7 network.
NEBS	Network Equipment-Building System is a standard for product safety.
NIF	Nodal Interworking Function is a module that interworks between the SS7 and the SIGTRAN stacks in the SG. It translates SS7 MTP3 messages to M3UA messages, and vice versa.
OPC	Origination Point Code is the originating node's address in the SS7 signaling messages.
PSTN	Public Switched Telephone Network is the traditional telephone system, using DS0 format to carry digitized modem or voice traffic.
Routing Key or Route Key	A Routing Key describes a set of SS7 parameters and parameter values that uniquely define the range of signaling traffic to be handled by a particular Application Server. Parameters within the Routing Key cannot extend across more than a single Signaling Point Management Cluster.
SCCP	Signaling Connection Control Part provides additional functions to MTP to provide connectionless and connection-oriented network services on a node-to-node basis.
SCN	Switched Circuit Network is the traditional telephone network that uses DS0 format to carry digitized modem or voice traffic.
SCTP	Stream Control Transmission Protocol is an IP based transmission protocol designed to carry SS7 signaling.

Table A-1:

Acronym or Term	Definition
SEP	A **S**ignaling **E**nd **P**oint is any location in the SS7 network that originates and/or terminates SS7 messages. This generic term is useful only to discuss SS7 architecture without the need to define the specific functions of a node. Therefore, other terms (SSP, SCP, IP, etc.) more fully define what happens at the node, but all are SEPs.
SG	Signaling Gateway is a device that allows SS7 network level services to IP based applications. An SG is a signaling agent that receives/sends SCN native signaling at the edge of the IP network [1]. An SG appears to the SS7 network as an SS7 Signaling Point. An SG contains a set of one or more unique Signaling Gateway Processes, of which one or more is normally actively processing traffic. Where an SG contains more than one SGP, the SG is a logical entity and the contained SGPs might typically be coordinated into a single management view to the SS7 network and to the supported Application Servers.
SGC	Signaling Gateway Client is an AS with one or more ASP that communicate with the SG
SGCI	Signaling Gateway Client Interface is an API provided with the SGC that allow ASPs to send signaling through the SG.
SGP	Signaling Gateway Process is a process instance of a Signaling Gateway. It serves as an active, back-up, load sharing or broadcast process of a Signaling Gateway.
SIGTRAN	Signaling Transport is the name of a work group in IETF that designed the architecture for the transport of SS7 signaling over IP.
SIO	Service Indicator Octet determines the MTP user, such as. ISUP, TUP, and SCCP.
SLS	Signaling Link Selection is used by the MTP users to provide guidance to the MTP layer on signaling link selection.
SNMP	Simple Network Management Protocol is a message-oriented protocol for the control and interrogation of network elements.
SP	Signaling Point is an SS7 network node such as an end office or tandem equipped with signaling link hardware and software.
SPMC	Signaling Point Management Cluster is the complete set of Application Servers represented to the SS7 network under one specific SS7 point code of one specific network appearance.
SSN	Subsystem Number identifies the user of the SCCP service layer.
SS7	Signaling System Number 7 is the protocol used to transport signaling for the PSTN.
STP	**Signaling Transfer Point** is the router in an SS7 network that sends and receives SS7 signals to and from other signaling points. STPs are always paired for redundancy, so that if one of the pair is not functioning then the other one handles the load.
SUA	SCCP User Adaptation enables seamless operation of SCCP user peers in SS7 and IP domains.
TCP	Transmission Control Protocol is the middle level of a protocol commonly used over IP to transport packet traffic.
TCAP	Transaction Capabilities Application Part provides a means to establish non-circuit related communication between two SS7 nodes.
TUP	Telephone User Part is an SS7 protocol layer used to setup and tear down calls.
UDP	User Datagram Protocol is a middle-level protocol used to transport large, fixed-length data blocks in a connectionless manner.

This page is intentionally blank.

Appendix B: Hardware Installation

B.1 Overview

This Appendix describes installing the hardware for NewNet Distributed7 and Signaling Gateway to run on a Sun platform with the SBS334, SB370, SBS372, PCI334, PCI370, PCI372, PCI370Q, PCI372PQ and ARTIC8260 boards.

B.2 HardwareComponents

This section describes the SS7 Controller boards and their configuration. In order to minimize the risk of damage to boards:

- Handle circuit boards very carefully along the edges only.
- Store circuit boards in the anti-static bags provided.
- Ground work-area properly.

Important*: PT-PCIxxx cards are keyed for universal (3.3VDC and 5.0VDC) voltage, but are only 5.0VDC cards. PTI PCI cards should only be installed in 5.0VDC PCI slots.*

B.2.1 Technical Characteristics

The environmental characteristics for the NewNet Communications Technologies software systems are in Table B-1.

Table B-1: Environmental Characteristics

Item	Characteristics[1] for Configuration
AC Power Requirements: (DC Power version)	100 to 240 Vac, 47 to 63Hz 4A
Aggregate Interface[2]	EIA RS-530, RS-449 & V.35
Operating System	UNIX Operating system with advanced development and run-time environment.
Ambient Temperature:	0°C to +50°C Operating (-25°C to +50°C Non-Operating)
Humidity:	5% to 90% relative humidity with no condensation (Operating /Non-Operating)
Altitude:	10,000 feet Operating (40,000 feet Non-Operating)
Enclosure Dimensions:	Fits inside enclosure of Host workstation

1. NewNet Communications Technologies reserves all the rights to change the equipment performance specifications stated herein at any time without notice. For OEM components, NewNet Communications Technologies relies on the specifications furnished by the OEM suppliers.

2. Optional V.35 and EIA RS 530 interface arrangements available on special order.

B.2.1.1 MTTR Maintenance Characteristics

The Mean Time to Repair (MTTR) detected failures for NewNet Distributed7 products should not exceed 0.5 hours.

B.2.1.2 MTBF Ratings

Table B-1 lists the Mean Time Between Failure (MTBF) ratings of the SS7 controller boards. NewNet Communications Technologies recommends use of this information to estimate spare modules to be kept for replacement.

Table B-1: MTBF Ratings

Module	MTBF[1] in Hours
SBS334	506,000
SBS370/372	164,936
PCI334	400,000 POH[2]
PCI370/372	273,063 POH[3]
PCI370PQ/PCI372PQ	519,911 POH
ARTIC1000 ARTIC2000 4-port T1/E1/J1 Line PMC 8-port T1/E1/J1 RTM	188,000 POH 247,000 POH 185,000 POH 367,000 POH

1. Please note that NewNet Communications Technologies makes no commitment in always supplying the components with these MTBF figures.

2. POH - Power On Hours

3. POH: Power On Hours

B.2.2 SBS334 Board

The **SBS334** boards can provide up to four SS7 links each on four DS0 level connections or ports. These boards can be used with the SBS370/372 boards in the same system.

The SBS334 board consists of a main board and two interface mezzanine, or daughter, boards. The main and mezzanine boards are assembled before shipment. The mezzanine board determines the interface type for the associated ports. The Sbus SS7 Link board plugs into the 96-pin connector of an Sbus slot of the workstation. The Sbus SS7 Link board can be installed in any Sbus slot available in the host CPU. Port numbering is described in Section B.2.2.1.

The RS-422 electrical interface board is used for RS-449 and RS-530 physical interfaces (with appropriate cables), and a V.35 version is also available. Connections to the ports are made directly to the port connectors which extend through the rear of the workstation enclosure. DCE/DTE Port configuration is software-controlled, and no jumper repositioning is required. Figure B-1 shows a general diagram of the board.

The SBS334 board is added and configured to the NewNet Distributed7 database through the ADD-SS7BOARD, MODIFY-SS7BOARD commands of MML. Please see Sample Board and Port Configuration on page B-40 for more information about configuring boards.

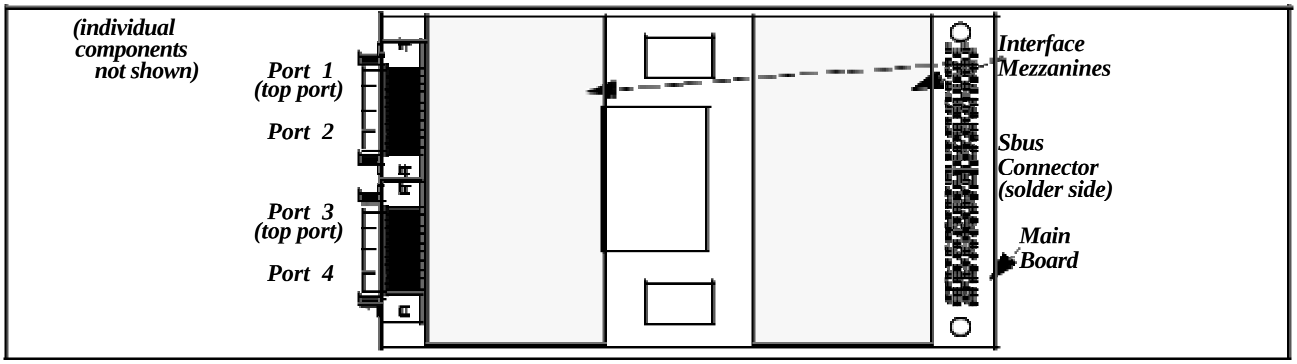


Figure B-1: SBS334 Interface Board

B.2.2.1 Board and Port Numbering

SBS334/370/372 boards use a driver called, sbs334. Device nodes are created under the /dev directory with the name, sbs334#, where the # is the driver instance number. The boards are numbered according to their driver instance number, not their slot position. The instance number is used to determine device driver minor node and thus selecting the physical board hardware.

The driver instance of the board in a particular slot is determined from the output of the *getcfg* command. The output is similar to the following:

```
Driver Board Type  Slot  Instance
------ ----------  ----  --------
sbs334 sbs334      0     0
sbs334 sbs370      1     1
```

Ports are numbered starting from 0 to n where n is the number physical ports on the board. On SBS344/370/372 boards there are 4 physical ports, hence port numbers are between 0 and 3.

B.2.2.2 Port Locations

Slots on a workstation are identified as shown in Figure B-1. Looking at the rear of the unit, the first slot (slot 0) is always the left-most slot of the top row, if multiple rows exist. The next slot (slot 1) is the slot to the right of the first slot. Then, if more than two slots exist, the left-most slot of the row underneath is the next slot, and so on.

The four ports of the board are identified as shown in the figure. The *First Port* refers to port number 1, 5, or 9, etc.

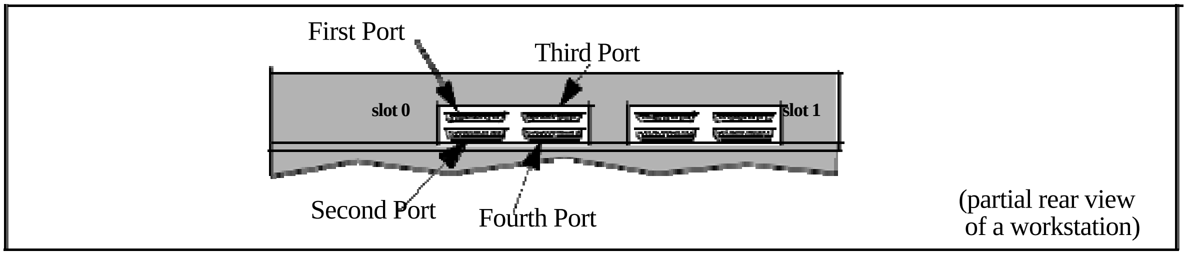


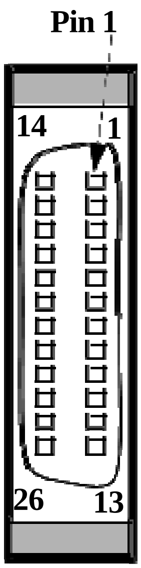
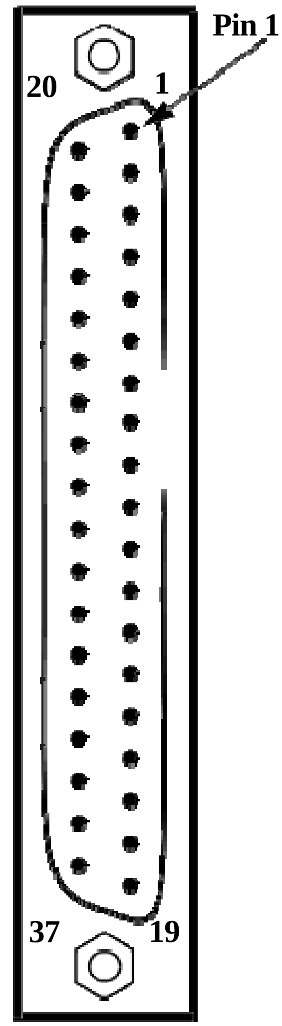
Figure B-1: Slots and Ports of an SBS334

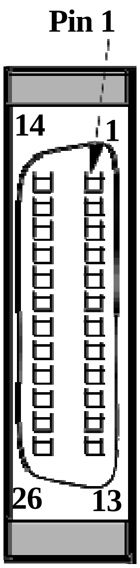
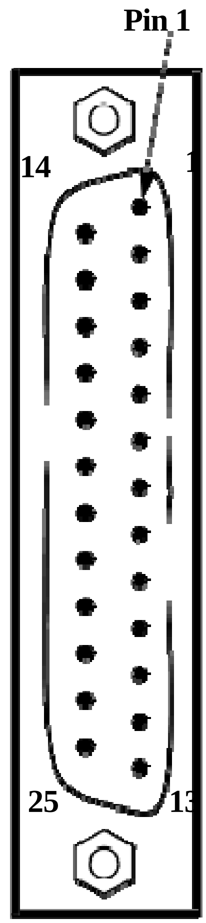
B.2.2.3 SS7 Link Port Connectors and Cables

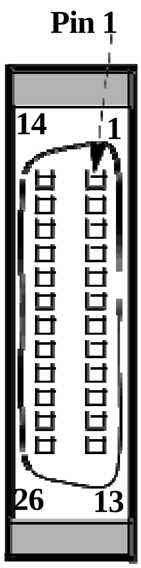
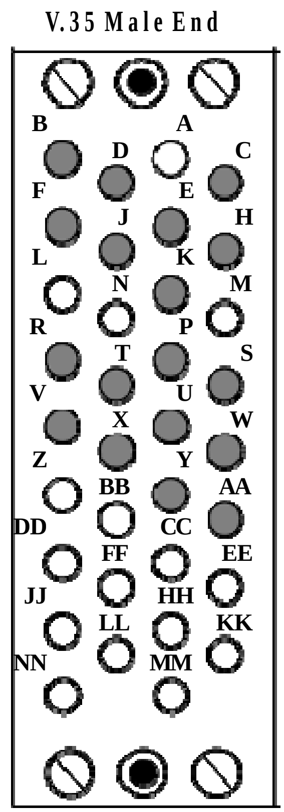
The SS7 Link port connectors extend through the rear panel of the host CPU enclosure. The port connectors contain 26 pins, which are configured by the interface mezzanine installed on the SS7 link board. Three different cables are used to connect this 26-pin connector to the SS7 link, depending on whether the SS7 link is V.35, EIA RS-449 (37-pin), or EIA RS-530 (25-pin). The pin connections for these cables are shown in the following tables:

- EIA RS-449: Table 2-3
- EIA RS-530: Table 2-4
- V.35: Table 2-5

The SS7 link cables plug directly into the port connectors. The cables have side screws on the connectors. The type and number of cables appropriate for the particular board will be provided.

RS-422 26-pin Port SS7 Link End	Pin No.	Signal Designation	Pin No.	RS-449 DB 37-pin SS7 Link End
	1	Ground	19	
	2	No Connection		
	3	No Connection		
	4	No Connection		
	5	Data Mode B	29	
	6	Send Data B	22	
	7	Send Data A	4	
	8	Terminal Ready B	30	
	9	Terminal Ready A	12	
	10	Request to Send B	25	
	11	Request to Send A	7	
	12	Terminal Timing B	35	
	13	Terminal Timing A	17	
	14	Receive Data B	24	
	15	Receive Data A	6	
	16	Clear to Send B	27	
	17	Clear to Send A	9	
	18	Receiver Ready B	31	
	19	Receiver Ready A	13	
	20	Receive Timing/OPTCK B	26	
	21	Receive Timing/OPTCK A	8	
	22	Send Timing B	23	
	23	Send Timing A	5	
	24	Spare Output B	32	
	25	Spare Output A	14	
	26	Data Mode A	11	

RS-422 26-pin Port End	Pin No.	Signal Designation	Pin No.	RS-530 DB 25-pin SS7 Link End
	1	Ground	7	
	2	No Connection		
	3	No Connection		
	4	No Connection		
	5	Data Mode B	22	
	6	Send Data B	14	
	7	Send Data A	2	
	8	Terminal Ready B	23	
	9	Terminal Ready A	20	
	10	Request to Send B	19	
	11	Request to Send A	4	
	12	Terminal Timing B	11	
	13	Terminal Timing A	24	
	14	Receive Data B	16	
	15	Receive Data A	3	
	16	Clear to Send B	13	
	17	Clear to Send A	5	
	18	Receiver Ready B	10	
	19	Receiver Ready A	8	
	20	Receive Timing/OPTCK B	9	
	21	Receive Timing/OPTCK A	17	
	22	Send Timing B	12	
	23	Send Timing A	15	
	24	Ground	7	
	25	Spare Output A	21	
	26	Data Mode A	6	

26-pin Port End	Pin No.	Signal Designation	Pin No.	V.35 SS7 Link End
	1	No Connection		
	2	No Connection		
	3	No Connection		
	4	No Connection		
	5	Data Set Ready	E	
	6	Send Data B	S	
	7	Send Data A	P	
	8	Terminal Ready B	H	
	9	Signal Ground	B	
	10	Request to Send	C	
	11	Signal Ground	B	
	12	Serial Clk Xmit Ext Timing B	W	
	13	Serial Clk Xmit Ext Timing A	U	
	14	Receive Data B	T	
	15	Receive Data A	R	
	16	Clear to Send	D	
	17	Signal Ground	B	
	18	Receiver Ready B	F	
	19	Signal Ground	B	
	20	Serial Clk Receive/OPTCK B	X	
	21	Serial Clk Receive/OPTCK A	V	
	22	Serial Clk Transmit Timing B	AA	
	23	Serial Clk Transmit Timing A	Y	
	24	Line Test	K	
	25	Signal Ground	B	
	26	Ring Indication	J	

B.2.3 SBS370/372 Board

SS7 connectivity over T1 or E1 interfaces is supported through the **SBS370** (T1) and **SBS372** (E1) Sbus boards. The dual port E1/T1 (120 ohms) and the dual/single port E1 (75 ohms) are both supported. The back panels of the board are depicted in Figure B-2. The dual port has modular connectors for two spans and a DB-9 connector for external clock or frame synchronization input. The dual/single port E1 has two DB-9 connectors, one for external synchronization and the other for two E1 spans. Please refer to the manufacturer's manual for information on external clock and frame synchronization connectors.

The **SBS370** and **SBS372** boards can provide up to four SS7 links each. The **SBS370** and **SBS372** boards provide a direct T1 (370) or E1 (372) interface, with the SS7 links on channels of the T1/E1.

The SBS370/372 boards are added and configured to the NewNet Distributed7 database through the ADD-SS7BOARD, MODIFY-SS7BOARD commands of MML. Please refer to the *NewNet Distributed7 User Manual* for the use of these commands and Section B.3.1.2 for a sample configuration script.

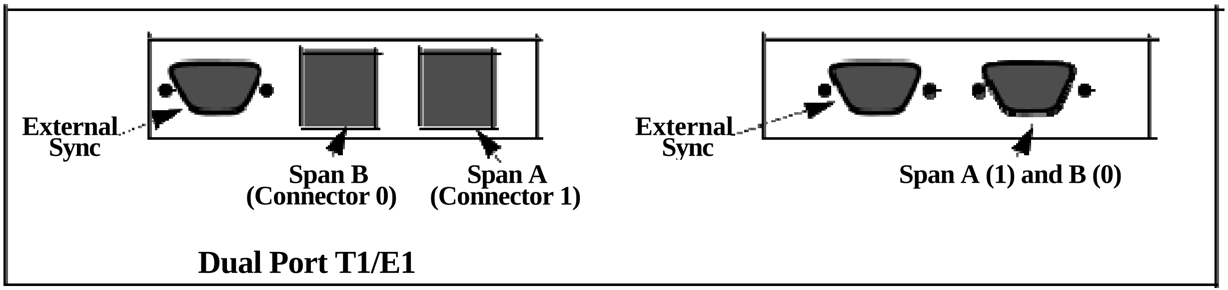


Figure B-2: SBS370/372 Boards - Back Panel View

B.2.3.1 Board and Port Numbering

Please refer to section Section B.2.2.1 for board and port numbering in SBS370/372 boards.

B.2.3.2 Port Locations

Slots on a workstation are identified the same as for the other Sbus boards (slot 0 is upper left slot). The connectors on the SBS370/372 boards are for T1s or E1s. The port numbers associated with the board (through driver instance number) must be mapped to channels of the T1/E1 through configuration of the board (Section B.2.3.4).

B.2.3.3 T1/E1 Configuration Settings

The SBS370/372 board has two T1/E1 ports and four HDLC (High Level Data Link Control) links. The entire board can support a maximum of four SS7 links because each link requires one HDLC link. The links may be allocated to any available time slot on either T1/E1 span by using the MODIFY-LINE command of MML.

MODIFY-LINE mml command initializes or reconfigures the spans of a T1/E1 board. The command has a series of arguments to configure a E1/T1 line (span).

B.2.3.4 Mapping Links to Channels

SS7 links are mapped to T1/E1 channels with Time Slot Interchange (TSI) commands. These commands reconfigure the TSA&PCM switch on the **SBS370/372** boards. T1/E1 boards incorporate on-board, non-blocking, time-space switching capabilities. These capabilities enable the boards to swap time slots among the three interfaces: T1/E1 Span-A, T1/E1 Span-B, and the HDLC controller. This switching operation can be performed independently for the *received* and *transmitted* highways of each interface.

The Span-A, Span-B, and HDLC controller highways operate at 2.048 Mb/s, regardless of the line speed of the T1/E1 interface. Each of these highways has 32 time slots (0-31). However, the time slot numbers of the highway are not the actual T1 or E1 time slot numbers. The correlation between the highway and the T1/E1 time slots is shown in Table B-1.

SS7 links are implemented on the 32-channel HDLC controller. Mapping of SS7 links to HDLC channels is static (non-user accessible). Since only four HDLC links exist on the board, only HDLC channels 0 through 3 can be used for mapping. The channels are associated with the physical port numbers from lowest to highest.

MODIFY-TIMESLOT mml command is used to assign span-A, span-B or HDLC spans (ports) to a time slot on either T1/E1 span (a or b) or a HDLC span. Assignment between two external spans (a or b) is possible and can be used in drop-and-insert applications. Time slot 0 is not used for data in the E1; it is the frame synchronization. However, the time slot mapped to the HDLC port is not the external T1 or E1 channel. coincide with the external T1 or E1 channels as shown in Table B-1.

Table B-1: T1 and E1 Time Slot Mapping

Span[Time Slot]	External T1 Channel	External E1 Channel	Span[Time Slot]	External T1 Channel	External E1 Channel
a[0] or b[0]	1	NC	a[16] or b[16]	17	16
a[1] or b[1]	2	1	a[17] or b[17]	18	17
a[2] or b[2]	3	2	a[18] or b[18]	19	18
a[3] or b[3]	4	3	a[19] or b[19]	20	19
a[4] or b[4]	5	4	a[20] or b[20]	21	20
a[5] or b[5]	6	5	a[21] or b[21]	22	21
a[6] or b[6]	7	6	a[22] or b[22]	23	22
a[7] or b[7]	8	7	a[23] or b[23]	24	23
a[8] or b[8]	9	8	a[24] or b[24]	NC	24
a[9] or b[9]	10	9	a[25] or b[25]	NC	25
a[10] or b[10]	11	10	a[26] or b[26]	NC	26
a[11] or b[11]	12	11	a[27] or b[27]	NC	27
a[12] or b[12]	13	12	a[28] or b[28]	NC	28
a[13] or b[13]	14	13	a[29] or b[29]	NC	29
a[14] or b[14]	15	14	a[30] or b[30]	NC	30
a[15] or b[15]	16	15	a[31] or b[31]	NC	31
NC=not connected					

Default Mapping Configuration

In the default mapping all time slots (span-A, span-B, and HDLCs) are in no-connect state.

Sample Loopback (Back-to-Back) Mapping Configuration

When using an external loop-back cable, a configuration can be set up to loop back from SS7 port 0 (HDLC 0) to SS7 port 2 (HDLC 2), and SS7 port 1 (HDLC 1) to SS7 port 3 (HDLC 3) as follows (assuming sbs372 (E1) board on host-A, instance 0):

```
MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
```

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=HDLC,
PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=LINEB,
PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=LINEB,
PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=HDLC,
PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=HDLC,
PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=LINEA,
PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=sbs334,INST=0,PORTSPAN=LINEA,
PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;

Because the loopback cable carries each time slot from span A to span B and vice versa, this effectively cross-connects SS7 Port 0 to Port 2, and SS7 Port 1 to Port 3.

B.2.3.5 E1/T1 Cables

Both a straight-through cable and a loopback cable can be used with the board. A straight-through cable is for connecting the T1/E1 span from the board to another piece of equipment for normal operation with other SS7 nodes. A loopback cable is for connecting span A to span B for loopback testing.

Straight-through Cable

The modular connector for the dual port E1/T1 board has the configuration defined below:

Table B-2: SBS370 and SBS372 (120 ohm) Connector

RJ48C Pin Function	Span B Connector and Pin	Span A Connector and Pin	At other equipment (e.g. STP or switch), connects to:
Receive RING	J2 1 (lead 1)	J3 1 (lead 1)	Transmit RING
Receive TIP	J2 2 (lead 2)	J3 2 (lead 2)	Transmit TIP
Transmit RING	J2 4 (lead 4)	J3 4 (lead 4)	Receive RING
Transmit TIP	J2 5 (lead 5)	J3 5 (lead 5)	Receive TIP

The cable for the 75-ohm E1 board breaks out from the DB-9 connector to four coaxial ends. The four ends are the transmit and receive for each span. The pinout of the DB-9 connector is shown in the table.

Table B-3: SBS372 (75 ohm) Connector Pinout

DB-9 Pin Function	Span B Pin	Span A Pin	At other equipment (e.g. STP or switch), connects to:
Receive TIP	J1 3	J1 1	Transmit TIP
Transmit TIP	J1 4	J1 2	Receive TIP
Receive RING	J1 8	J1 6	Transmit RING
Transmit RING	J1 9	J1 7	Receive RING

Loopback Cable

The loopback cable connects each time slot of span A to the corresponding time slot of span B. The configuration of the cable for the Sbs370 and Sbs372 (120 ohm) boards is defined:

Span A	Span B
Lead 1 - time slot Ring	Tx Ring, Lead 4
Lead 2 - Rx Tip	Tx Tip, Lead 5
Lead 4 - Tx Ring	Rx Ring, Lead 1
Lead 5 - Tx Tip	Rx Tip, Lead 2

For loopback with the SBS372 (75 ohm) board, the transmit coaxial end of Span B must be connected to the receive end of Span A and the transmit coaxial end of Span A must be connected to the receive end of Span B.

B.2.4 PCI334 Board

The **PCI334** board can provide up to four DS0-level connections or ports for SS7 links. These boards can be used with the PCI370/372 boards in the same system. There are three versions of PCI334 board:

- RS-422 (RS-449) physical interface version
- V.35 physical interface version
- RS-232C physical interface version

Ports are accessed through a port connector which extends through the rear of the workstation enclosure. The boards have a single connector for all four ports. This connector requires a hydra-style breakout cable which provides the four connectors at the desired physical interface (EIA-530 or V.35). DCE/DTE Port configuration is software-controlled, and no jumper repositioning is required.

The PCI334 board is added and configured to the NewNet Distributed7 database through the ADD-SS7BOARD, MODIFY-SS7BOARD commands of MML. Please refer to {section explaining these commands} for the use of these commands and Section B.3.2.1 for a sample configuration script.

B.2.4.1 Board and Port Numbering

PCI334/370/372 boards use a driver called, pci334. Device nodes are created under the /dev directory with the name, pci334#, where the # is the driver instance number. The boards are numbered according to their driver instance number, not their slot position. The instance number is used to determine device driver minor node.

The driver instance of the board in a particular slot is determined from the output of the getcfg command. The output is similar to the following:

```
Driver Board Type  Slot  Instance
------ ----------  ----  --------
pci334   pci334     0     0
pci334   pci370     1     1
```

Ports are numbered starting from 0 to n where n is the number physical ports on the board. In PCI344 there are 4 physical ports, hence port numbers are between 0 and3. Instance number is used to select the actual board to perform the requested operation.

B.2.4.2 SS7 Link Port Connectors and Cables

The PCIbus board port connector has 80 pins, 20 for each port/link. Different breakout cables are used for V.35 and EIA-530. The hydra-style cables break the 80-pin connector out to 4 DB-25 connectors for EIA-530 and 4 M-34 connectors for V.35.

The pin connections for these cables are shown in the following tables:

- PCIbus EIA RS-530: Table B-1 on page B-12
- PCIbus V.35: Table B-2 on page B-14

The type of cable appropriate for the particular board will be provided.

Table B-1: PCIbus EIA-530 Pinout

80-Pin Connector Pin Number				Signal	Description	EIA-530 Mnemonic	EIA530 DB-25 Pin Number
for Port 1	**for Port 2**	**for Port 3**	**for Port 4**				
1	21	41	61	RXD#-	Receive Data	BB(A)	3
2	22	42	62	RXD#+	Receive Data	BB(B)	16
3	23	43	63	DTR#-	Data Terminal Ready	CD(A)	20
4	24	44	64	DTR#+	Data Terminal Ready	CD(B)	23
5	25	45	65	TXD#-	Transmit Data	BA(A)	2
6	26	46	66	TXD#+	Transmit Data	BA(B)	14
7	27	47	67	RTS#-	Request to Send	CA(A)	4
8	28	48	68	RTS#+	Request to Send	CA(B)	19
9	29	49	69	TXC#-	Transmit Clock	DA(A)	24
10	30	50	70	TXC#+	Transmit Clock	DA(B)	11
11	31	51	71	TXCI#-	Transmit Clock In	DB(A)	15
12	32	52	72	TXCI#+	Transmit Clock In	DB(B)	12
13	33	53	73	DCD#-	Data Carrier Detect	CF(A)	8
# is the port number (e.g. RXD1- is receive data for port 1) *Data from PCI Quad Communications Controller User's Manual*							

Table B-1: PCIbus EIA-530 Pinout

80-Pin Connector Pin Number				Signal	Description	EIA-530 Mnemonic	EIA530 DB-25 Pin Number
for Port 1	for Port 2	for Port 3	for Port 4				
14	34	54	74	DCD#+	Data Carrier Detect	CF(B)	10
15	35	55	75	DSR#-	Data Set Ready	CC(A)	6
16	36	56	76	DSR#+	Data Set Ready	CC(B)	22
17	37	57	77	CTS#-	Clear to Send	CB(A)	5
18	38	58	78	CTS#+	Clear to Send	CB(B)	13
19	39	59	79	RXC#-	Receive Clock	DD(A)	17
20	40	60	80	RXC#+	Receive Clock	DD(B)	9

Table B-2: PCIbus V.35 Pinout

80-Pin Connector Pin Number				Signal	Description	V. 35 Mnemonic	V.35 M-34 Pin Number
for Port 1	for Port 2	for Port 3	for Port 4				
1	21	41	61	RXD#-	Receive Data	104	R
2	22	42	62	RXD#+	Receive Data	104	T
3	23	43	63	DTR#	Data Terminal Ready	108	H
4	24	44	64		No Connection		
5	25	45	65	TXD#-	Transmit Data	103	P
6	26	46	66	TXD#+	Transmit Data	103	S
7	27	47	67	RTS#-	Request to Send	105	C
8	28	48	68	GND#	Signal Ground	102	B
9	29	49	69	TXC#-	Transmit Clock	113	U
10	30	50	70	TXC#+	Transmit Clock	113	W
11	31	51	71	TXCI#-	Transmit Clock In	114	Y
12	32	52	72	TXCI#+	Transmit Clock In	114	AA
13	33	53	73	DCD#	Data Carrier Detect	109	F
14	34	54	74	RI#	Ring Indicator	125	J
15	35	55	75	DSR#	Data Set Ready	107	E
16	36	56	76	LT#	Line Test		K
17	37	57	77	CTS#	Clear to Send	106	D
18	38	58	78		No Connection		
19	39	59	79	RXC#-	Receive Clock	115	V
20	40	60	80	RXC#+	Receive Clock	115	X

 # is the port number (e.g. RXD1- is receive data for port 1)
Data from PCI Quad Communications Controller User's Manual

B.2.5 PCI370/372 Board

SS7 connectivity over T1 or E1 interfaces is supported through the PCI370 (T1) and PCI372 (E1) PCIbus boards. Please refer to the manufacturer's manual for information on these boards. The PCI370 and PCI372 boards can provide up to four SS7 links each. The PCI370 and PCI372 boards provide a direct T1 (370) or E1 (372) interface, with the SS7 links on channels of the T1/E1.

The dual port E1 (120 ohms)/T1 and the dual/single port E1 (75 ohms) are both supported. The back panels of the board are depicted in Figure B-1. The dual port has modular connectors for two spans and a DB-9 connector for external clock or frame synchronization input. The dual/single port E1 has two DB-9 connectors, one for external synchronization and the other for two E1 spans. Please refer to the manufacturer's manual for information on external clock and frame synchronization connectors.

The PCI370/372 boards are added and configured to the NewNet Distributed7 database through the ADD-SS7BOARD, and MODIFY-SS7BOARD MML commands. Please refer to the *NewNet Distributed7 User Manual* for the use of these commands and Section B.3.2.2 and Section B.3.2.3 for sample configuration scripts.

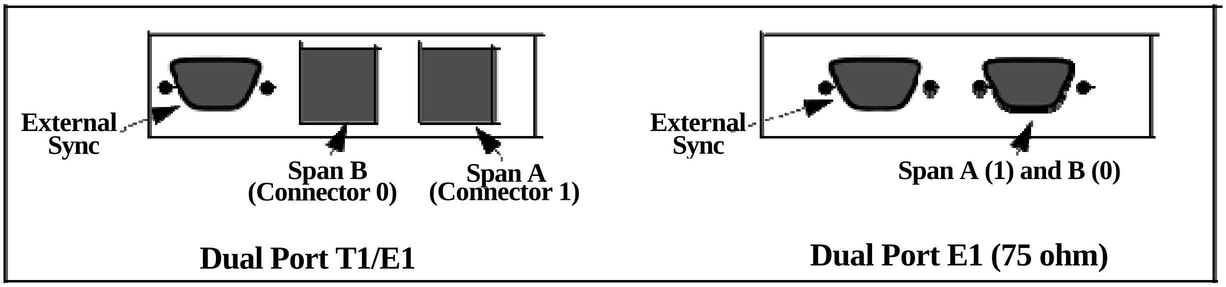


Figure B-1: T1/E1 Boards - Back Panel View

B.2.5.1 Board and Port Numbering

Please see Section B.2.4.1 for board and port numbering in PCI370/372 boards.

B.2.5.2 Port Locations

Slots on a workstation are identified the same as for the other Sbus boards (slot 0 is upper left slot). The connectors on the PCI370/372 boards are for T1s or E1s. The port numbers associated with the board (through driver instance number) must be mapped to channels of the T1/E1 through configuration of the board (Section B.2.3.4).

B.2.5.3 T1/E1 Configuration Settings

The PCI370/372 board has two T1/E1 ports and four HDLC (High Level Data Link Control) links. The entire board can support a maximum of four SS7 links because each link requires one HDLC link. The links may be allocated to any available time slot on either T1/E1 span by using the MODIFY-LINE command of MML.

MODIFY-LINE mml command initializes or reconfigures the spans of a T1/E1 board. The command has a series of arguments to configure a E1/T1 line (span). Please refer to the *NewNet Distributed7 User Manual*.

B.2.5.4 Mapping Links to Channels

SS7 links are mapped to T1/E1 channels with Time Slot Interchange (TSI) commands. These commands reconfigure the TSA&PCM switch on the **PCI370/372** boards. T1/E1 boards incorporate on-board, non-blocking, time-space switching capabilities. These capabilities enable the boards to swap time slots among the three interfaces: T1/E1 Span-A, T1/E1 Span-B, and the HDLC controller. This switching operation can be performed independently for the *received* and *transmitted* highways of each interface.

The Span-A, Span-B, and HDLC controller highways operate at 2.048 Mb/s, regardless of the line speed of the T1/E1 interface. Each of these highways has 32 time slot (0-31). However, the time slot numbers of the highway are not the actual T1 or E1 time slot numbers. The correlation between the highway and the T1/E1 time slots is shown in Table B-1.

SS7 links are implemented on the 32-channel HDLC controller. Mapping of SS7 links to HDLC channels is static (non-user accessible). Since only four HDLC links exist on the board, only HDLC channels 0 through 3 can be used for mapping. The channels are associated with the physical port numbers from lowest to highest.

MODIFY-TIMESLOT mml command is used to assign span-A, span-B or HDLC spans (ports) to a time slot on either T1/E1 span (a or b) or a HDLC span. Assignment between two external spans (a or b) is possible and can be used in drop-and-insert applications. Time slot 0 is not used for data in the E1; it is the frame synchronization. However, the time slot mapped to the HDLC port is not the external T1 or E1 channel. coincide with the external T1 or E1 channels as shown in Table B-1.

Table B-1: T1 and E1 Time Slot Mapping

Span[Time Slot]	External T1 Channel	External E1 Channel	Span [Time Slot]	External T1 Channel	External E1 Channel
a[0] or b[0]	1	NC	a[16] or b[16]	17	16
a[1] or b[1]	2	1	a[17] or b[17]	18	17
a[2] or b[2]	3	2	a[18] or b[18]	19	18
a[3] or b[3]	4	3	a[19] or b[19]	20	19
a[4] or b[4]	5	4	a[20] or b[20]	21	20
a[5] or b[5]	6	5	a[21] or b[21]	22	21
a[6] or b[6]	7	6	a[22] or b[22]	23	22
a[7] or b[7]	8	7	a[23] or b[23]	24	23
a[8] or b[8]	9	8	a[24] or b[24]	NC	24
a[9] or b[9]	10	9	a[25] or b[25]	NC	25
a[10] or b[10]	11	10	a[26] or b[26]	NC	26
a[11] or b[11]	12	11	a[27] or b[27]	NC	27
a[12] or b[12]	13	12	a[28] or b[28]	NC	28
a[13] or b[13]	14	13	a[29] or b[29]	NC	29
a[14] or b[14]	15	14	a[30] or b[30]	NC	30
a[15] or b[15]	16	15	a[31] or b[31]	NC	31
NC=not connected					

Default Mapping Configuration

In the default mapping all time slots (span-A, span-B, and HDLCs) are in no-connect state.

Sample Loopback (Back-to-Back) Mapping Configuration

When using an external loop-back cable, a configuration can be set up to loop back from SS7 port 0 (HDLC 0) to SS7 port 2 (HDLC 2), and SS7 port 1 (HDLC 1) to SS7 port 3 (HDLC 3) as follows (assuming pci372 (E1) board on host-A, instance 0):

```
MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
```



```
MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=LINEB,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci334,INST=0,PORTSPAN=LINEA,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;
```

Because the loop-back cable carries each timeslot from span A to span B and vice versa, this effectively cross-connects SS7 Port 0 to Port 2, and SS7 Port 1 to Port 3.

B.2.5.5 E1/T1 Cables

Both a straight-through cable and a loopback cable can be used with the board. A straight-through cable is for connecting the T1/E1 span from the board to another piece of equipment for normal operation with other SS7 nodes. A loopback cable is for connecting span A to span B for loopback testing.

Straight-through Cable

The modular connector for the dual port E1/T1 board has the configuration defined below:

Table B-2: T1 and E1 (120 ohm) Connector Pinout

RJ48C Pin Function	Span B Connector and Pin	Span A Connector and Pin	At other equipment (e.g. STP or switch), connects to:
Receive RING	J2 1 (lead 1)	J3 1 (lead 1)	Transmit RING
Receive TIP	J2 2 (lead 2)	J3 2 (lead 2)	Transmit TIP
Transmit RING	J2 4 (lead 4)	J3 4 (lead 4)	Receive RING
Transmit TIP	J2 5 (lead 5)	J3 5 (lead 5)	Receive TIP

The cable for the 75-ohm E1 board breaks out from the DB-9 connector to four coaxial ends. The four ends are the transmit and receive for each span. The pinout of the DB-9 connector is shown in the table.

Table B-3: PCI370 and PCI372 (120 ohm) Connector Pinout

DB-9 Pin Function	Span B Pin	Span A Pin	At other equipment (e.g. STP or switch), connects to:
Receive TIP	J1 3	J1 1	Transmit TIP
Transmit TIP	J1 4	J1 2	Receive TIP
Receive RING	J1 8	J1 6	Transmit RING
Transmit RING	J1 9	J1 7	Receive RING

The cable for the 75-ohm E1 board breaks out from the DB-9 connector to four coaxial ends. The four ends are the transmit and receive for each span. The pinout of the DB-9 connector is shown in the table.

Table B-4: PCI372 (75 ohm) Connector Pinout

DB-9 Pin Function	Span B Pin	Span A Pin	At other equipment (e.g. STP or switch), connects to:
Receive TIP	J1 3	J1 1	Transmit TIP
Transmit TIP	J1 4	J1 2	Receive TIP
Receive RING	J1 8	J1 6	Transmit RING
Transmit RING	J1 9	J1 7	Receive RING

Loopback Cable

The loopback cable connects each time slot of span A to the corresponding time slot of span B. The configuration of the cable for the T1 and 120-ohm E1 boards is defined as follows:

Span A	Span B
Lead 1 - Rx Ring	Tx Ring, Lead 4
Lead 2 - Rx Tip	Tx Tip, Lead 5
Lead 4 - Tx Ring	Rx Ring, Lead 1
Lead 5 - Tx Tip	Rx Tip, Lead 2

For loopback with the 75-ohm E1 board, the transmit coaxial end of Span B must be connected to the receive end of Span A and the transmit coaxial end of Span A must be connected to the receive end of Span B.

B.2.6 PCI3700PQ/372PQ Board

SS7 connectivity over T1 or E1 interfaces is supported through the PCI370PQ (T1) and PCI372PQ (E1) PCIbus boards. Please refer to the manufacturer's manual for information on these boards. The PCI370PQ and PCI372PQ boards are based on a newer and much more powerful processor (MPC860) than the processor (MC68360) used in PCI370 and PCI372 boards. These boards can provide up to 16 SS7 links each and they provide a direct T1 (370PQ) or E1 (372PQ) interface, with the SS7 links on channels of the T1/E1.

The dual port E1 (120 ohms)/T1 and the dual/single port E1 (75 ohms) are both supported. The back panels of the board are depicted in Figure B-1. The dual port has modular connectors for two spans and a DB-9 connector for external clock or frame synchronization input. The dual/single port E1 has two DB-9 connectors, one for external synchronization and the other for two E1 spans. Please refer to the manufacturer's manual for information on external clock and frame synchronization connectors.

The PCI370PQ/372PQ boards are added and configured to the NewNet Distributed7 database through the ADD-SS7BOARD, and MODIFY-SS7BOARD MML commands. Please refer to the *NewNet Distributed7 User Manual* for the use of these commands and Section B.3.2.2 and Section B.3.2.3 for sample configuration scripts.

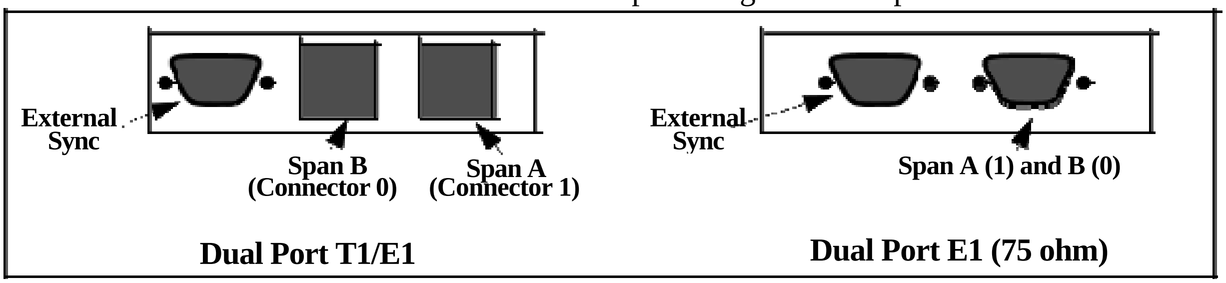


Figure B-1: T1/E1 Boards - Back Panel View

B.2.6.1 Board and Port Numbering

Please refer to Section B.2.4.1 for board and port numbering in PCI370PQ/372PQ boards.

B.2.6.2 Port Locations

Slots on a workstation are identified the same as for the other Sbus boards (slot 0 is upper left slot). The connectors on the PCI370PQ/372PQ boards are for T1s or E1s. The port numbers associated with the board (through driver instance number) must be mapped to channels of the T1/E1 through configuration of the board (Section B.2.3.4).

B.2.6.3 T1/E1 Configuration Settings

The PCI370PQ/372PQ board has two T1/E1 ports and 24 HDLC (High Level Data Link Control) links. The entire board can support a maximum of 24 SS7 links because each link requires one HDLC link. However, NewNet Communications Technologies recommends to configure a maximum of 16 links on each board for full bandwidth performance. With more than 16 links the performance decreases due to the CPU processing power limits. The links may be allocated to any available time slot on either T1/E1 span by using the MODIFY-LINE MML command.

MODIFY-LINE MML command initializes or reconfigures the spans of a T1/E1 board. The command has a series of arguments to configure a E1/T1 line (span). Please refer to the *NewNet Distributed7 User Manual*.

B.2.6.4 Mapping Links to Channels

SS7 links are mapped to T1/E1 channels with Time Slot Interchange (TSI) commands. These commands reconfigure the TSA&PCM switch on the **PCI370PQ/372PQ** boards.

T1/E1 boards incorporate on-board, non-blocking, time-space switching capabilities. These capabilities enable the boards to swap time slots among the three interfaces: T1/E1 Span-A, T1/E1 Span-B, and the HDLC controller. This switching operation can be performed independently for the *received* and *transmitted* highways of each interface.

The Span-A, Span-B, and HDLC controller highways operate at 2.048 Mb/s, regardless of the line speed of the T1/E1 interface. Each of these highways has 32 time slots (0-31). However, the time slot numbers of the highway are not the actual T1 or E1 time slot numbers. The correlation between the highway and the T1/E1 time slots is shown in Table B-1.

SS7 links are implemented on the 32-channel HDLC controller. Mapping of SS7 links to HDLC channels is static (non-user accessible). Since only four HDLC links exist on the board, only HDLC channels 0 through 23 can be used for mapping. The channels are associated with the physical port numbers from lowest to highest.

MODIFY-TIMESLOT mml command is used to assign span-A, span-B or HDLC spans (ports) to a time slot on either T1/E1 span (a or b) or a HDLC span. Assignment between two external spans (a or b) is possible and can be used in drop-and-insert applications. Time Slot 0 is not used for data in the E1; it is the frame synchronization. However, the time slot mapped to the HDLC port is not the external T1 or E1 channel. coincide with the external T1 or E1 channels as shown in Table B-1.

Table B-1: T1 and E1 Time Slot Mapping

Span[Time Slot]	External T1 Channel	External E1 Channel	Span[Time Slot]	External T1 Channel	External E1 Channel
a[0] or b[0]	1	NC	a[16] or b[16]	17	16
a[1] or b[1]	2	1	a[17] or b[17]	18	17
a[2] or b[2]	3	2	a[18] or b[18]	19	18
a[3] or b[3]	4	3	a[19] or b[19]	20	19
a[4] or b[4]	5	4	a[20] or b[20]	21	20
a[5] or b[5]	6	5	a[21] or b[21]	22	21
a[6] or b[6]	7	6	a[22] or b[22]	23	22
a[7] or b[7]	8	7	a[23] or b[23]	24	23
a[8] or b[8]	9	8	a[24] or b[24]	NC	24
a[9] or b[9]	10	9	a[25] or b[25]	NC	25
a[10] or b[10]	11	10	a[26] or b[26]	NC	26
a[11] or b[11]	12	11	a[27] or b[27]	NC	27
a[12] or b[12]	13	12	a[28] or b[28]	NC	28
a[13] or b[13]	14	13	a[29] or b[29]	NC	29
a[14] or b[14]	15	14	a[30] or b[30]	NC	30
a[15] or b[15]	16	15	a[31] or b[31]	NC	31
NC=not connected					

Default Mapping Configuration

In the default mapping all time slots (span-A, span-B, and HDLCs) are in no-connect state.

Sample Loopback (Back-to-Back) Mapping Configuration

When using an external loop-back cable, a configuration can be set up to loop back from SS7 port 0 (HDLC 0) to SS7 port 2 (HDLC 2), and SS7 port 1 (HDLC 1) to SS7 port 3 (HDLC 3) as follows (assuming pci372pq (E1) board on host-A, instance 0):

```
MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;

MODIFY-TIMESLOT:HOSTNAME=host-A,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;
```

Because the loopback cable carries each time slot from span A to span B and vice versa, this effectively cross-connects SS7 Port 0 to Port 2, and SS7 Port 1 to Port 3.

B.2.6.5 E1/T1 Cables

Both a straight-through cable and a loopback cable can be used with the board. A straight-through cable is for connecting the T1/E1 span from the board to another piece of equipment for normal operation with other SS7 nodes. A loopback cable is for connecting span A to span B for loopback testing.

Straight-through Cable

The modular connector for the dual port E1/T1 board has the configuration defined in the following table:

Table B-2: T1 and E1 (120 ohm) Connector Pinout

RJ48C Pin Function	Span B Connector and Pin	Span A Connector and Pin	At other equipment (e.g. STP or switch), connects to:
Receive RING	J2 1 (lead 1)	J3 1 (lead 1)	Transmit RING
Receive TIP	J2 2 (lead 2)	J3 2 (lead 2)	Transmit TIP
Transmit RING	J2 4 (lead 4)	J3 4 (lead 4)	Receive RING
Transmit TIP	J2 5 (lead 5)	J3 5 (lead 5)	Receive TIP

The cable for the 75-ohm E1 board breaks out from the DB-9 connector to four coaxial ends. The four ends are the transmit and receive for each span. The pinout of the DB-9 connector is shown in the following table.

Table B-3: PCI370PQ and PCI372PQ (120 ohm) Connector Pinout

DB-9 Pin Function	Span B Pin	Span A Pin	At other equipment (e.g. STP or switch), connects to:
Receive TIP	J1 3	J1 1	Transmit TIP
Transmit TIP	J1 4	J1 2	Receive TIP
Receive RING	J1 8	J1 6	Transmit RING
Transmit RING	J1 9	J1 7	Receive RING

The cable for the 75-ohm E1 board breaks out from the DB-9 connector to four coaxial ends. The four ends are the transmit and receive for each span. The pinout of the DB-9 connector is shown in the table.

Table B-4: PCI372PQ (75 ohm) Connector Pinout

DB-9 Pin Function	Span B Pin	Span A Pin	At other equipment (e.g. STP or switch), connects to:
Receive TIP	J1 3	J1 1	Transmit TIP
Transmit TIP	J1 4	J1 2	Receive TIP
Receive RING	J1 8	J1 6	Transmit RING
Transmit RING	J1 9	J1 7	Receive RING

Loopback Cable

The loopback cable connects each time slot of span A to the corresponding time slot of span B. The configuration of the cable for the T1 and 120-ohm E1 boards is defined as follows:

Span A	Span B
Lead 1 - Rx Ring	Tx Ring, Lead 4
Lead 2 - Rx Tip	Tx Tip, Lead 5
Lead 4 - Tx Ring	Rx Ring, Lead 1
Lead 5 - Tx Tip	Rx Tip, Lead 2

For loopback with the 75-ohm E1 board, the transmit coaxial end of Span B must be connected to the receive end of Span A and the transmit coaxial end of Span A must be connected to the receive end of Span B.

B.2.7 ARTIC8260 Boards (ARTIC1000 and ARTIC2000)

B.2.7.1 ARTIC1000

The ARTIC1000 is a 64-bit, 33-66MHz CompactPCI single slot, 6U-form factor peripheral board based on the PowerQUICC II (MPC8260), a versatile microprocessor and peripheral controller for communication and networking applications. Installed in a CompactPCI chassis the ARTIC1000 provides Signaling Gateway the SS7 connectivity over T1, E1 or J1 interfaces. The ARTIC1000 is compliant with the PCI Local Bus Specification, Revision 2.2 and the CompactPCI Specification 2.0, Revision 2.1. The board's J1 and J2 connectors have an array of pins for connection to the system's 64-bit CompactPCI bus signals, voltages and grounds. The board's J1 connector has no key, which allows Universal Board plugging.

The ARTIC1000 can be used as the baseboard for up to two line interface PMC modules, allowing support of up to eight T1, E1 or J1 lines. It supports H.110 and SCSA backplane computer telephony (CT) bus standards with its H.110 bus connector and has two Ethernet ports that are not utilized in Signaling Gateway.

Figure B-1 provides an explanation of the important parts of the board:

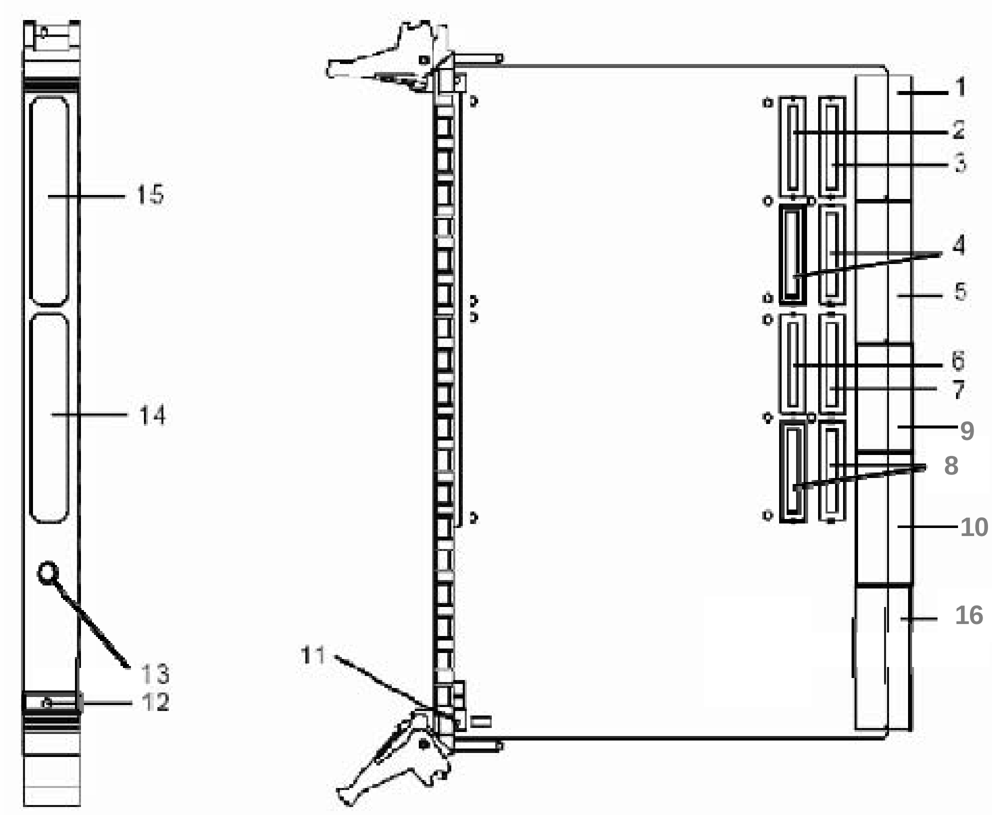


Figure B-1: ARTIC1000 Carrier Board Layout and Key Components

Callout	Description
1	J5 CompactPCI connector for rear I/O
2-3-4	PMC slot 2 J21, J22, J23, J24 connectors
5	J4 CompactPCI connector for H.110 Computer Telephony (CT) bus signals
6-7-8	PMC slot 1 J11, J12, J13, J14 connectors
9	J3 CompactPCI connector for rear I/O
10	J2 CompactPCI connector for 64 bit PCI bus signals
11	Hot-swap switch
12	Hot-swap LED - blue - when lit, board is ready to be hot-swapped
13	Board status LED - yellow or green
14	PMC slot 2 cutout and cap
15	PMC slot 1 cutout and cap
16	J1 CompactPCI connector for 32-bit PCI bus signals

B.2.7.2 ARTIC2000

The ARTIC2000 is a 64-bit, 33-66MHz PCI single slot, full-size PCI card form factor peripheral board based on the PowerQUICC II (MPC8260), a versatile microprocessor and peripheral controller for communication and networking applications. Installed in a PCI chassis ARTIC2000 provides Signaling Gateway the SS7 connectivity over T1, E1 or J1 interfaces. The board has an edge connector to connect to the system's 64-bit PCI bus signals, voltages and grounds. The board's edge connector has a 3.3V key, which requires the board to be installed into a 3.3V PCI slot.

In terms of network functionality the ARTIC2000 is a subset of ARTIC1000. The ARTIC2000 has only one PMC slot, hence it can carry only one line interface PMC, supporting up to four T1, E1 or J1 lines. The ARTIC2000 supports H.100, SCSA, MVIP and H-MVIP bus standards via an edge connector at the back of the card.

Figure B-2 provides an explanation of the important parts of the board:

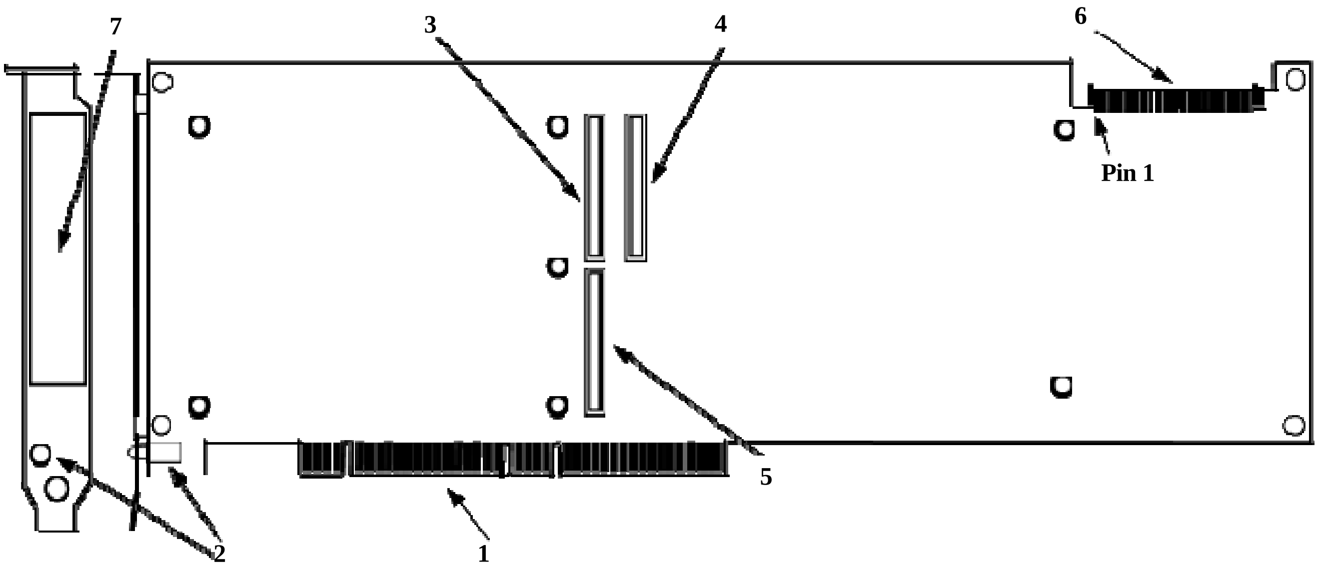


Figure B-2: ARTIC2000 Carrier Board Layout and Key Components

Callout	Description
1	64-bit PCI Bus edge connector (keyed for 3.3V PCI slot)
2	Board status LED - yellow or green
3-4-5	PMC slot J11, J12, J13 connectors
6	Computer Telephony (CT Bus) Bus edge connector complying ECTF H.100 specification
7	PMC slot cutout and cap

CT Bus (H.100) Connector

The ARTIC2000 board has a standard CT Bus (H.100) edge connector on the top back side of the PCB. This connector can be used to connect ARTIC2000 to other CT Bus (H.100, SCSA, MVIP and H-MVIP) compliant hardware in the same chassis.

For more information about H.100 bus, please refer to the *H.100 Hardware Compatibility Specification: CT Bus, revision 1.0, 1997, Enterprise Computer Telephony Forum (ECTF)*. This document can be downloaded from the ECTF web site at http://www.ectf.org/.

Table B-1: H.100 Connector Pinout

Pin Name	Pin#	Pin#	Pin Name
Reserved	1	2	CT_+5Vdc
CT_D31	3	4	CT_D30
CT_D29	5	6	CT_D28
GND	7	8	CT_D27
CT_D26	9	10	CT_D25
CT_D24	11	12	GND
CT_D23	13	14	CT_D22
CT_D21	15	16	CT_D20
GND	17	18	CT_D19
CT_D18	19	20	CT_D17
CT_D16	21	22	GND
CT_D15	23	24	CT_D14
CT_D13	25	26	CT_D12
GND	27	28	CT_D11
CT_D10	29	30	CT_D9
CT_D8	31	32	GND
CT_D7	33	34	CT_D6
CT_D5	35	36	CT_D4
GND	37	38	CT_D3
CT_D2	39	40	CT_D1
CT_D0	41	42	GND
/CT_FRAME_A	43	44	GND
CT_C8_A	45	46	GND
CT_NETREF	47	48	GND
/CT_FRAME_B	49	50	GND
CT_C8_B	51	52	GND
CT_MC	53	54	GND
/FR_COMP	55	56	GND
SCLK	57	58	GND
SCLKx2	59	60	GND
C2	61	62	GND
/C4	63	64	GND

Table B-1: H.100 Connector Pinout (Continued)

Pin Name	Pin#	Pin#	Pin Name
/C16+	65	66	/C16-
GND	67	68	Reserved

B.2.7.3 4-Port T1/E1/J1 Line Interface PMC

The 4-Port T1/E1/J1 Line Interface PMC is a single non-extended PCI Mezzanine Card (PMC) sized daughter card, designed to be mounted onto an ARTIC1000 or ARTIC2000. The line interface PMC will provide four ports of 1.544 Mb/s T1.403 (U.S. and Canadian T1 and Japanese J1) or 2.048 Mb/s G.703 (E1) compatible interfaces. The official name for the 4-Port T1/E1/J1 Line PMC is IOP-PMC-02000 (Spitfire).

The 4-Port T1/E1/J1 Line PMC provides four T1/E1/J1 line interfaces through a Rear Transition Module in CompactPCI systems, or through four on-card RJ-48 connectors, H.100 bus TDM interface and switch, 32-bit 33 MHz PCI interface. The H.100 interface on the line PMC is not used; instead, the baseboard H.110 (ARTIC1000) or H.100 (ARTIC2000) interface is used.

The 4-Port T1/E1/J1 Line PMC supports connections to T1, E1 or J1 lines. Four RJ-48 connectors provide connections for external lines to the four T1/E1/J1 interfaces.

Each T1/E1/J1 line interface can be configured for one of the following:

- T1 or J1 interface - 100 ohm, 1.544 Mb/s, 24 channels
- E1 balanced interface - 120 ohm, 2.048 Mb/s, 32 channels

There is no switching of impedances between T1/J1 and E1 balanced. E1 unbalanced is not supported by hardware on the PMC. An E1 unbalanced interface (75 ohm, 2.048 Mb/s, 32 channels) can be achieved by using a readily available 120 ohm to 75 ohm balun transformer.

Software configuration handles the selection of the clocking rate and the number of channels per line. Each T1/E1/J1 line interface can be configured for either T1 mode, J1 mode or E1 mode. There can be any mixture of T1, E1 and J1 modes on the same card.

Each T1/E1/J1 line interface consists of four wires. Two of the wires are for transmit and two are for receive. The two transmit wires form a differential pair, and the two receive wires form a differential pair. The signal of the T1/E1/J1 line interface is a three voltage level signal (0 volts, +voltage, -voltage) and is such that the data and clock are combined into one waveform (self-clocking).

The four RJ-48 connectors provide the connection for the four T1/E1/J1 line interfaces. The RJ-48 connectors for the 4-Port T1/E1/J1 Line PMC are mounted on the card such that when the card is installed in a system unit, the connector is accessible through the standard I/O area of the system unit.

Each RJ-48 connector has 8 pins. Each T1/E1/J1 line interface has two pins for the transmit signals, two pins for the receive signals, and the other four pins are connected to frame ground. The pinout of each RJ-48 connector is shown in Figure B-3.

The T1/E1/J1 line interface is transformer isolated. The transformer on the PMC provides 1.5 kV of isolation. Lightning protection for the line interfaces can be provided with devices connected externally.

The 4-Port T1/E1/J1 Line PMCs provide connections for the T1/E1/J1 line interface through PMC connector **PN4** to the base card for use with a Rear Transition Module (RTM) in a CompactPCI system. The RTM must supply the transformer isolation and protection.

The 8-Port T1/E1/J1/ 2 Ethernet RTM is an example of an RTM that can be used with the 4-Port T1/E1/J1 Line PMCs in a CompactPCI system. This RTM provides 3 kV of isolation.

Analog switches are used to control which interface (on-card connectors or RTM) is used by the application on the 4-Port T1/E1/J1 Line PMC. The T1/E1/J1 line interfaces can either be used in the front or in the rear with an RTB. A mixture of front and rear use of line interfaces is not possible. If the basecard detects an RTB connected to it, it will automatically configure the PMC to use the connectors on the RTB; otherwise, connectors on the PMC are used. A 4-Port T1/E1/J1 Line PMC on an ARTIC1000 baseboard can be configured to use front or rear connectors, whereas on an ARTIC2000 only front connectors are allowed since there is no RTB available for the ARTIC2000 baseboard.

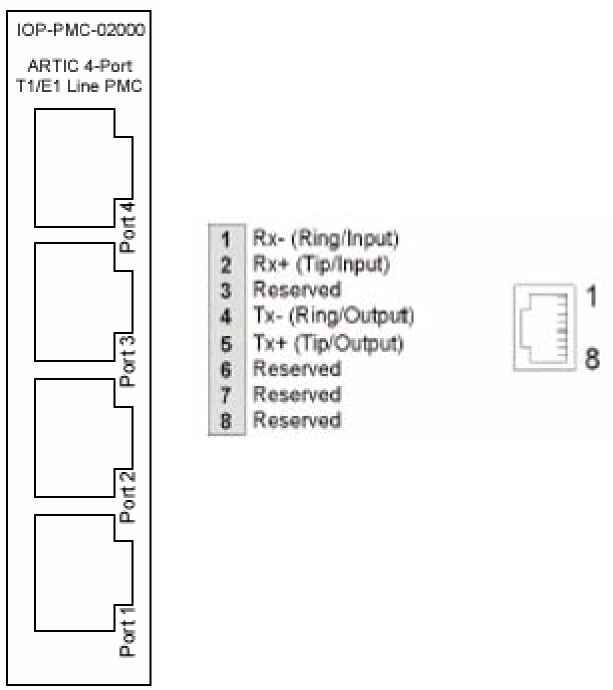


Figure B-3: Line PMC Front Panel (left) and RJ-48 Connector Pinouts (right)

B.2.7.4 Rear Transition Board (RTB)

A Rear Transition Board must be used in order to access the line interfaces in the rear. The necessary I/O signals from the ARTIC1000 are already available via the CompactPCI backplane at the RJ3 connector. The RTB supports T1/E1/J1 line interfaces and provides 3 kV of isolation with the protection circuitry on it that is necessary for long-haul T1/J1 line interface support. The official name for the 8-Port T1/E1/J1/ 2 Ethernet RTM is IOP-PMC-00100.

The RTB front panel contains connectors for eight T1/E1/J1 line interfaces, trunk LEDs for each of T1/E1/J1 line interfaces and two Ethernet ports. The Ethernet ports are not used in Distributed7.

The trunk LEDs provide a visual way of reporting the T1/E1/J1 line status. A green LED for a line (trunk) means that there are no critical framing and/or synchronization errors in the line. A yellow LED for a line means there is at least one critical framing and/or synchronization error in the line.

Figure B-4 shows the diagram of the RTB front panel connector layout.

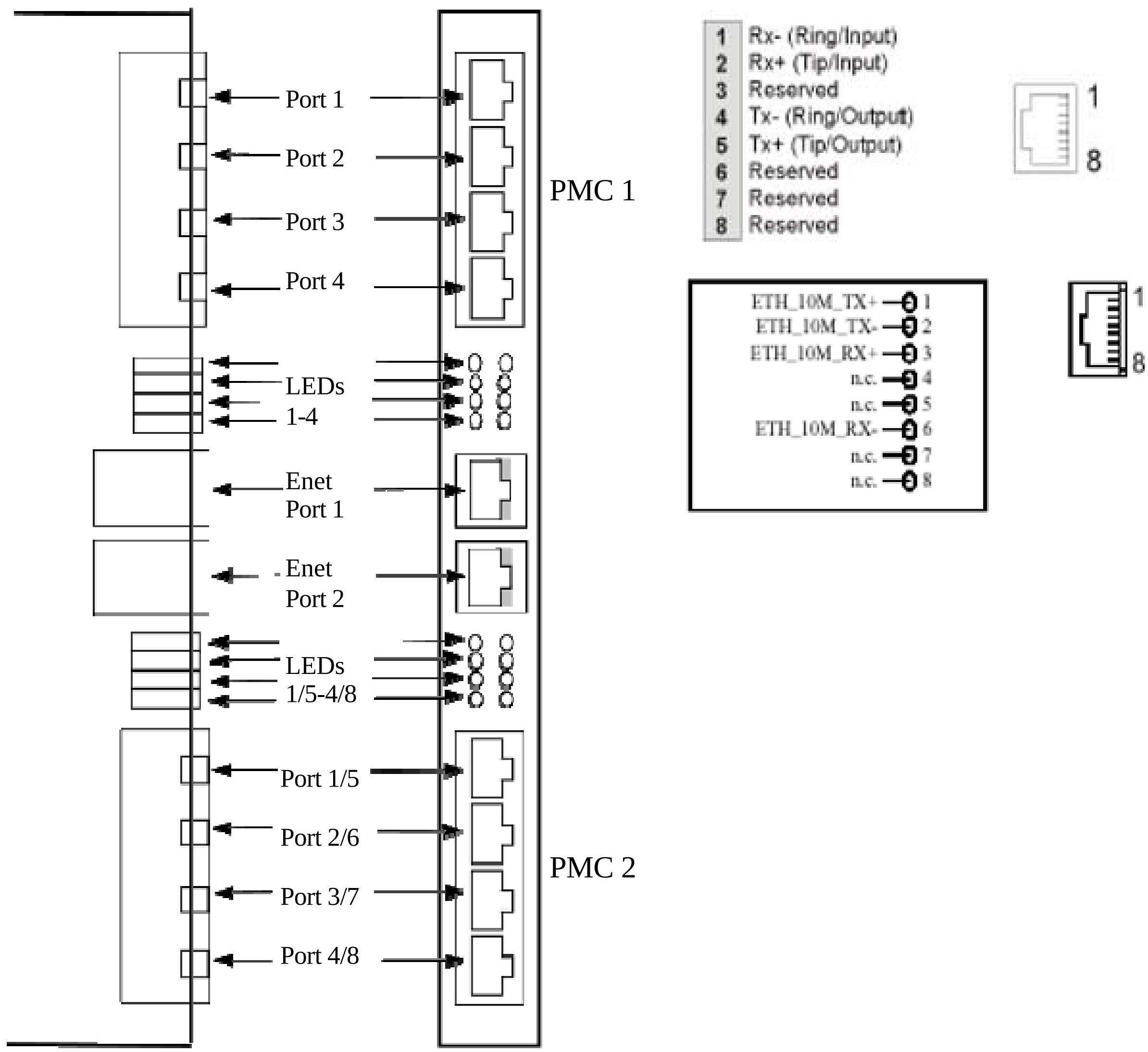


Figure B-4: RTB Front Panel (left) and RJ-48 (top) and Ethernet Connector Pinouts (bottom)

B.2.7.5 Line PMC Installation on ARTIC1000

Installation of the line PMC module onto the ARTIC1000 board:

c. Identify the PMC slot into which the PMC will be installed, and locate the corresponding PMC connectors

Note: *PMC slot 1 cannot be left blank on the ARTIC1000, or the board device deriver will not attach to that board instance.*

d. Remove the corresponding PMC slot cover from the front panel of the ARTIC1000 *e.* Remove the PMC, mounting hardware, screws and standoff from the protective bag

f. Remove the screws from the standoffs on the PMC and set them aside for step g

g. Insert the bezel of the PMC into the cutout on the front panel of the ARTIC1000 (1) as shown in Figure B-5

h. Engage the connectors (2), making sure they are seated completely

i. Secure the four screws through the clearance holes (3) on the back side of the ARTIC1000

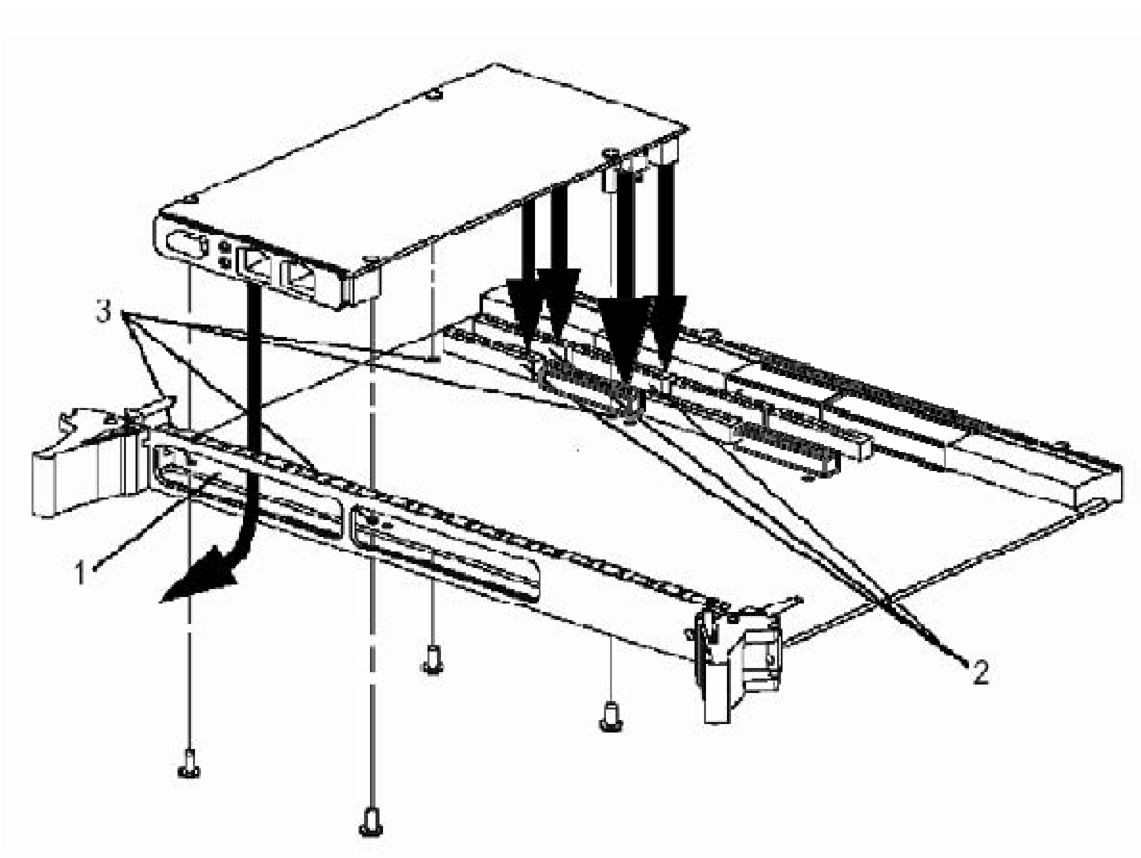


Figure B-5: Line PMC Module Installation

B.2.7.6 Line PMC Installation on ARTIC2000

Installation of the line PMC module onto the ARTIC2000 board is similar to the procedure for the ARTIC1000:

a. Locate the PMC connectors

***Note**: The PMC slot can't be left blank on the ARTIC2000 or the board device deriver will not attach to that board instance.*

b. Remove the corresponding PMC slot cover from the front panel of the ARTIC2000

c. Remove the PMC, mounting hardware, screws and standoff from the protective bag

d. Remove the screws from the standoffs on the PMC and set them aside for step g

e. Insert the bezel of the PMC into the cutout on the front panel of the ARTIC2000 (1) as shown in Figure B-5

f. Engage the connectors (2), making sure they are seated completely

g. Secure the four screws through the clearance holes (3) on the back side of the ARTIC2000

B.2.7.7 ARTIC1000 Installation

The ARTIC1000 board provides hot-swap support. The board can be installed in or removed from a powered system that supports hot swapping of plug-in boards.

Important: *In the Distributed7 Release 1.5.0.1 there is no hot-swap support for the ARTIC1000 board, therefore the board should not be installed or removed while the system is powered.*

Refer to Figure B-6 for the following steps to install the ARTIC1000:

a. Stop the Distributed7 software, if running, with ***apm_stop*** command

b. Become ***su*** enter ***cd $EBSHOME/access/install/*** and do ***./ebs_modremove***

c. Power off external devices connected to the system

d. Shutdown the machine (***init 0***)

e. Power off the system

f. Unplug the power cord from the system's AC power inlet

Notice: *Appropriate anti-static protection measures must be followed when handling static sensitive devices.*

g. Place the top and bottom edges of the ARTIC1000 board in the card guides (1) of the chassis

h. Check that the injector/ejector levers (2) of the two handles are in the outward position

i. Slide the board into chassis until you feel resistance (approximately 1/4 inch short of full insertion)

j. Simultaneously move the injector/ejector levers of the two handles to the inward position

k. Verify that the board is seated properly

l. Tighten the captive screws (3) that secure the board to the chassis

m. Power up the system

n. Become ***su*** enter ***cd $EBSHOME/access/install/*** and do ***./ebs_modinstall***

o. Go to ***$EBSHOME/access/bin*** and run ***getcfg*** command

p. Verify that the new board is identified

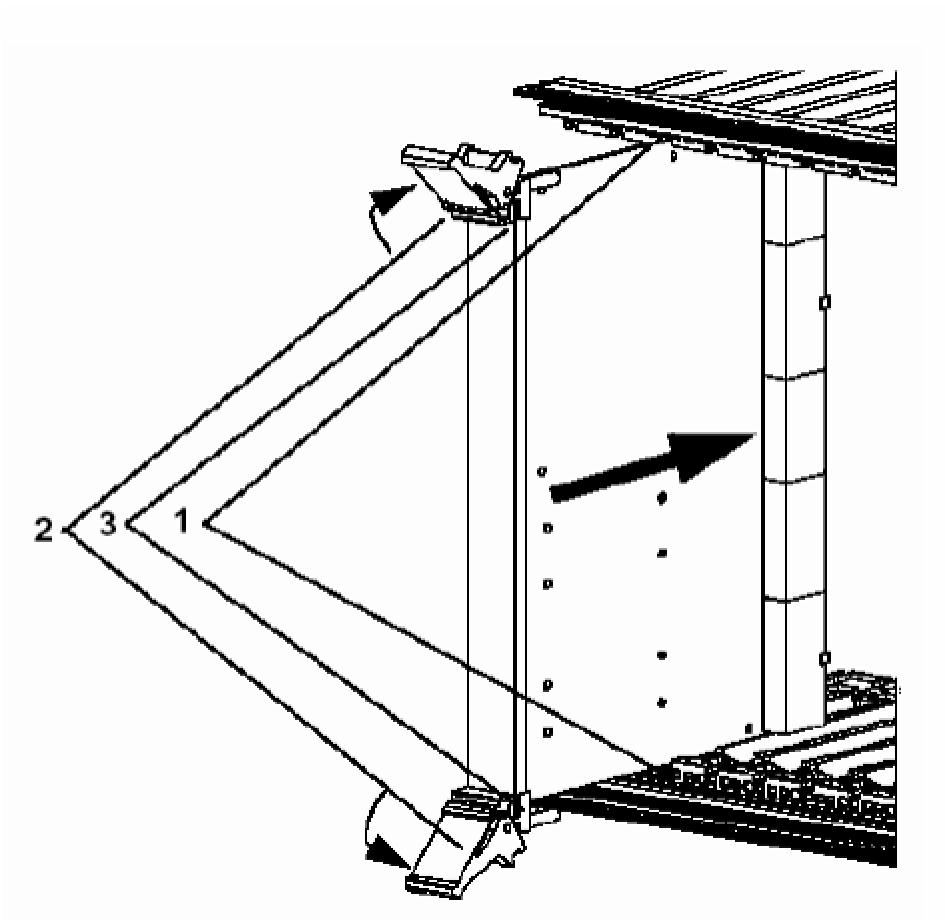


Figure B-6: Installing the ARTIC1000 in a Non-powered System

B.2.7.8 ARTIC2000 Installation

a. Stop the Distributed7 software, if running, with ***apm_stop*** command

b. Become ***su*** enter ***cd $EBSHOME/access/install/*** and do ***./ebs_modremove***

c. Power off external devices connected to the system

d. Shutdown the machine (***init 0***)

e. Power off the system

f. Unplug the power cord from the system's AC power inlet

Notice*: Appropriate anti-static protection measures must be followed when handling static sensitive devices.*

g. Open the machine housing and install the ARTIC2000 board into the appropriate slot

h. Close up the machine housing and power up the machine

i. Become ***su*** enter ***cd $EBSHOME/access/install/*** and do ***./ebs_modinstall***

j. Go to ***$EBSHOME/access/bin*** and run ***getcfg command***

k. Verify that the new board is identified

B.2.7.9 Rear Transition Board (RTB) Installation and Removal

Before using the RTB IOP-PMC-00100 in a system the following items must be checked:

- Only install and use the RTB IOP-PMC-00100 with ARTIC1000 boards; otherwise the RTB and the other board may be destroyed
- The RTB may only be connected to TNV-1 circuits (Telecommunication Network Voltage-1 circuits) as defined in EN60950
- Make sure to use the rear of the correct slot position

Installation of the RTB (Figure B-7):

a. Locate the correct slot position(s) to install the RTB(s)

b. Insert RTB into same slot as the carrier board. When inserting it, use slot rails to align it correctly and do not bend any backplane connector pins

c. Press handles so that board is completely plugged into the CompactPCI connectors

d. Fasten two-front panel screws to fix the RTB at rack frame

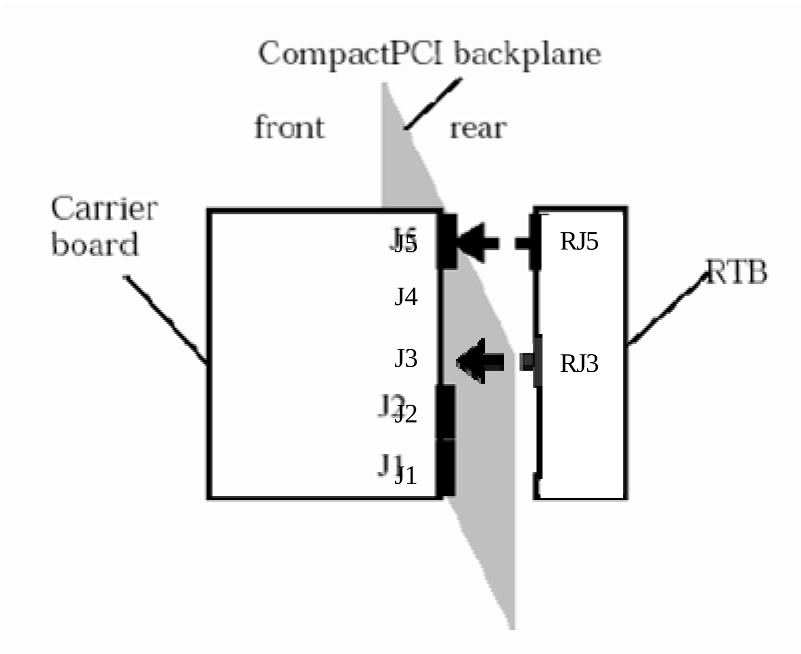


Figure B-7: Rear Transition Board Installation

Removal of RTB:

a. Turn off power

b. Unfasten two front-panel screws until the RTB is detached from rack frame

c. Disconnect the RTB from the backplane by opening handles of the front panel

d. Remove the RTB from slot rails

B.2.7.10 Using ARTIC8260 Boards in Distributed7

The ARTIC8260 board is added and configured to the Distributed7 database through the ***ADD_SS7BOARD*** and ***MODIFY_SS7BOARD*** MML commands. Please refer to the *Distributed7 User Manual* for the use of these commands and Figure B.3.3 for sample configuration scripts.

B.2.7.11 Board and Port Numbering

The ARTIC1000/2000 boards use a driver called ***artic8260***. Device nodes are created under the ***/dev*** directory with the name ***artic8260#***, where # is the driver instance number. The boards are numbered according to their driver instance number, not their slot position. The instance number is used to determine the device driver minor node.

The driver instance of the board in a particular slot is determined from the output of the ***getcfg*** command. The output is similar to the following:

In a CompactPCI system:

```
Driver    Board Type Slot Instance
--------- ---------- ---- --------
artic8260 artic1000  0    0
artic8260 artic1000  1    1
```

In a PCI system:

```
Driver    Board Type Slot Instance
--------- ---------- ---- --------
artic8260 artic2000  0    0
artic8260 artic2000  1    1
```

As shown in the output above, the ***getcfg*** command detects the bus type of the system and prints the correct designation for the board type.

Ports are numbered starting from 0 to n, where n is the number physical ports on the board. In the ARTIC1000/2000 there are 64 physical ports, hence port numbers are between 0 and 63. Instance number is used to select the actual board to perform the requested operation.

B.2.7.12 Port Locations

Slots on a workstation are identified the same as other CompactPCI boards (slot 0 is left most slot). The connectors on the ARTIC1000/2000 boards are for T1, E1 or J1 connectivity. The port numbers associated with the board (through driver instance number) must be mapped to channels of the T1/E1/J1 through configuration of the board (Section B.2.7.14).

B.2.7.13 T1/E1/J1 Configuration Settings

The ARTIC1000 board can have up to eight and ARTIC2000 board has four T1/E1/J1 ports and 64 HDLC (High-Level Data Link Control) links. The entire board can support a maximum of 64 SS7 links because each link requires one HDLC link. The links may be allocated to any available timeslot on either T1/E1/J1 span by using the ***MODIFY_TIMESLOT*** MML command. Please refer to the *Distributed7 User Manual* for more information about the ***MODIFY_TIMESLOT*** MML command.

The ***MODIFY_LINE*** MML command initializes or reconfigures the spans of a ARTIC1000/2000 board. The command has a series of arguments to configure an E1/T1/J1 line (span). Please refer to the *Distributed7 User Manual* for more information about ***MODIFY_LINE*** MML command.

B.2.7.14 Mapping Links to Channels

SS7 links are mapped to T1/E1/J1 or CT (H.110) bus channels with Time-Space Interchange (TSI) commands. These commands reconfigure the Time-Space Interchange switch on the ARTIC1000/2000 boards. ARTIC1000/2000 boards incorporate on-board, non-blocking, time-space switching capabilities. These capabilities enable the boards to swap timeslots among the 42 interfaces: T1/E1/J1 Span-1 to Span-8, CTbus-0 to CTbus-31 and the HDLC controller. This switching operation can be performed independently for the received and transmitted highways of each interface.

The Span-1 to Span-8 data highways on the board operate at 2.048 Mb/s, regardless of the line speed of the T1/E1/J1 interface. Thus, Span-1 to Span-8 data highways have 32 timeslots (0-31). HDLC controller data highways operate at 4.096 Mb/s, and have 64 timeslots (0-63). CT bus spans can operate at the 2.048 Mb/s, 4.096 Mb/s or 8.192 Mb/s. Therefore, a CT bus data highway can have 32 (0-31), 64 (0-63) or 128 (0-127) timeslots depending on the data rate selected for that span with the ***MODIFY-CTBUS*** MML command. Please refer to the *Distributed7 User Manual* for more information about the ***MODIFY_CTBUS*** MML command.

However, the timeslot numbers on the HDLC data highway are not the actual T1, E1 or J1 timeslot numbers. The correlation between the HDLC data highway and the other data highway (T1/E1/J1 or CT Bus) timeslots is transparent to the user. The basic mapping of external channels to T1/E1/J1 span channels is shown in Table B-2.

SS7 links are implemented on the 64-channel HDLC controller. Mapping of SS7 links to HDLC channels is static (not user accessible). Since 64 HDLC links exist on the board, HDLC channels 0 through 63 can be used for mapping. The channels are associated with physical port numbers from lowest to highest.

The ***MODIFY_TIMESLOT*** MML command is used to assign span (1 to 8), CT bus (0 to 31), or HDLC spans (ports) to a timeslot on either T1/E1/J1 span (1 to 8), CT bus (0 to 31) or an HDLC span. Assignment between two external spans (1 to 8) is possible and can be used in drop-and-insert applications. Timeslot 0 is not used for data in the E1; it is for the E1 frame synchronization. However, assignment between CT Bus timeslots is not allowed.

Table B-2: T1/E1/J1 Channel Mapping

Span[Timeslot]	External T1 Channel	External E1 Channel	Span[Timeslot]	External T1 Channel	External E1 Channel
1[0] to 8[0]	1	NC	1[16] to 8[16]	17	16
1[1] to 8[1]	2	1	1[17] to 8[17]	18	17
1[2] to 8[2]	3	2	1[18] to 8[18]	19	18
1[3] to 8[3]	4	3	1[19] to 8[19]	20	19
1[4] to 8[4]	5	4	1[20] to 8[20]	21	20
1[5] to 8[5]	6	5	1[21] to 8[21]	22	21
1[6] to 8[6]	7	6	1[22] to 8[22]	23	22
1[7] to 8[7]	8	7	1[23] to 8[23]	24	23
1[8] to 8[8]	9	8	1[24] to 8[24]	NC	24
1[9] to 8[9]	10	9	1[25] to 8[25]	NC	25
1[10] to 8[10]	11	10	1[26] to 8[26]	NC	26
1[11] to 8[11]	12	11	1[27] to 8[27]	NC	27
1[12] to 8[12]	13	12	1[28] to 8[28]	NC	28

Table B-2: T1/E1/J1 Channel Mapping (Continued)

Span[Timeslot]	External T1 Channel	External E1 Channel	Span[Timeslot]	External T1 Channel	External E1 Channel
1[13] to 8[13]	14	13	1[29] to 8[29]	NC	29
1[14] to 8[14]	15	14	1[30] to 8[30]	NC	30
1[15] to 8[15]	16	15	1[31] to 8[31]	NC	31
NC					

Default Mapping Configuration

In the default mapping, all timeslots (span-1 to span-8, CTbus-0 to CTbus-31 and HDLCs) are in no-connect state.

Sample Loopback (Back-to-Back) Mapping Configuration

When using an external loop-back cable, a configuration can be set up to loop back from SS7 port 0 (HDLC 0) to SS7 port 2 (HDLC 2), and from SS7 port 1 (HDLC 1) to SS7 port 3 (HDLC 3) as follows (assuming an ARTIC8260 [T1] board on host-A, instance 0):

```
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM=artic8260, INST=0, DESTTYPE=HDLC, DESTSLOT=0,
ORIGTYPE=LINE,ORIGSPAN=1, ORIGSLOT=1;
MODIFY-TIMESLOT:HOSNAME=host-A, BOARDNM= artic8260, INST=0, DESTTYPE=HDLC, DESTSLOT=1,
ORIGTYPE=LINE,ORIGSPAN=1, ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0,
DESTTYPE=LINE,DESTSPAN=1, DESTSLOT=1, ORIGTYPE=HDLC, ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0, DEST-
TYPE=LINE,DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0, DESTTYPE=HDLC, DESTSLOT=2,
ORIGTYPE=LINE,ORIGSPAN=2, ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0, DESTTYPE=HDLC, DESTSLOT=3,
ORIGTYPE=LINE,ORIGSPAN=2, ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0, DESTTYPE=LINE,DESTSPAN=2,
DESTSLOT=1, ORIGTYPE=HDLC, ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=host-A, BOARDNM= artic8260, INST=0, DESTTYPE=LINE,DESTSPAN=2,
DESTSLOT=2, ORIGTYPE=HDLC, ORIGSLOT=3;
```

Because the loop-back cable carries each timeslot from span 1 to span 2 and vice versa, this effectively cross-connects SS7 Port 0 to Port 2, and SS7 Port 1 to Port 3.

B.2.7.15 T1/E1/J1 Cables

Both a straight-through cable and a loop-back cable can be used with the board. A straight-through cable is for connecting the T1/E1/J1 span from the board to another piece of equipment for normal operation with other SS7 nodes. A loop-back cable is for connecting span 1 to span 2 for loop-back testing.

Straight-through Cable

The modular connector for the dual port E1/T1 board has the configuration defined below:

Table B-3: T1/E1/J1 RJ48 Connector Pinouts

RJ48 Pin Function	Connector Pin	At other equipment (e.g. STP or switch) connects to
Receive RING	Lead 1	Transmit RING
Receive TIP	Lead 2	Transmit TIP
Transmit RING	Lead 4	Receive RING
Transmit TIP	Lead 5	Receive TIP

Loopback Cable

The loopback cable connects each timeslot of span X (1-8) to the corresponding timeslot of span Y (1-8). The configuration of the cable for the T1 and 120-ohm E1 spans is defined as follows:

Table B-4: Loopback Cable Configuration

Span-X	Span-Y
Lead 1 - Rx Ring	Tx Ring - Lead 4
Lead 2 - Rx Tip	Tx Tip - Lead 5
Lead 4 - Tx Ring	Rx Ring - Lead 1
Lead 5 - Tx Tip	Rx Tip - Lead 2

B.2.8 Installing a New PCIbus Card

The following procedure should be followed to install a new PCIbus card in a machine running Distributed7 software.

1. Stop the Distributed7 software
2. Become *su* and *cd **$EBSHOME/access/install/*** and do *./ebs_modremove -f*

***Note**: If multiple software releases are installed on a system, a plain **ebs_modremove** attempt will not be permitted; under these circumstances, you must specify the -f [force] command-line option.*

3. Shutdown the machine (init 0)
4. Power down the machine
5. Open the machine housing and install PCIbus SS7 board into the appropriate slot

***Notice**: Appropriate anti static protection measures must be followed when handling static sensitive devices.*

6. Close up the machine housing and power up the machine
7. Become *su* and *cd **$EBSHOME/access/install/*** and do *./ebs_modinstall -f*

***Note**: If multiple software releases are installed on a system, a plain **ebs_modinstall** attempt will not be permitted; under these circumstances, you must specify the -f [force] command-line option.*

8. Go to ***$EBSHOME/access/bin*** and run *getcfg* command.

9. Verify that now the new board is identified.

B.3 Sample Board and Port Configuration

In the following sections configuration scripts for SBS334, SBS370, PCI334, PCI370/372 boards are given. For E1/T1 boards the time slot mapping is done for back-to-back connection of spans A and B.

B.3.1 SBus Configuration

Assuming that the configuration is made on an sbus host "zeus" and the output of *getcfg* command is as follows:

```
Driver Board Type  Slot  Instance
------ ----------  ----  --------
sbs334 sbs334      0     0
sbs334 sbs370      1     1
```

Then the following mml scripts can be used to configure the boards.

B.3.1.1 SBS334

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,PORTS=4;

MOD-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,PORTNUM=0,TYPE=DCE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,PORTNUM=1,TYPE=DCE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,PORTNUM=2,TYPE=DTE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,PORTNUM=3,TYPE=DTE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MOD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=0,CONF=ON;
```

B.3.1.2 SBS370 (T1/E1)

SBS370 is the common designator for SBS370 (T1) and SBS372 (E1) boards. If SBS370 is a T1 board:

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1;

MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=B,
    LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE;
MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=A,
    LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE;

MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
```



```
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=2,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=3,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=0,LISTENSPAN=HDLC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=0;
MODIFY-TIMSLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMSLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=0,LISTENSPAN=HDLC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=3;

MOD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,CONF=ON;
```

If SBS370 is a E1 (120 ohm) board:

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1;

MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=B,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE;
MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=A,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE;

MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=0,
    OUT_TMSLTTYP=LINEB,OUT_TMSLT=1,IN_TMSLTTYP=LINEB,IN_TMSLT=1,
    BAUD=64000,LPBKMODE=NONE;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=1,
    OUT_TMSLTTYP=LINEB,OUT_TMSLT=2,IN_TMSLTTYP=LINEB,IN_TMSLT=2,
    BAUD=64000,LPBKMODE=NONE;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=2,
    OUT_TMSLTTYP=LINEA,OUT_TMSLT=1,IN_TMSLTTYP=LINEA,IN_TMSLT=1,
    BAUD=64000,LPBKMODE=NONE;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=3,
    OUT_TMSLTTYP=LINEA,OUT_TMSLT=2,IN_TMSLTTYP=LINEA,IN_TMSLT=2,
    BAUD=64000,LPBKMODE=NONE;

MOD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,CONF=ON;
```

If SBS370 is a E1 (75 ohm) board:

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PROTOCOL=ansi,CLASS=II,
    PORTS=4,CLOCKMODE=LINEB;

MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=B,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE;
MODIFY-LINE:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,SPAN=A,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE;

MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=2,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTNUM=3,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMSLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LITENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;

MOD-SS7BOARD:HOSTNAME=zeus,BOARDNM=sbs334,INST=1,CONF=ON;
```

B.3.2 PCI Bus Configuration

Assuming that the configuration is made on a PCI bus host “cosmos” and the output of getcfg command is as follows:

```
DriverBoard TypeSlotInstance
----------------------------
pci334pci3340  0
pci334pci3702 1
pci334pci3723 2
pci3xpqpci370pq4 0
pci3xpqpci372pq5 1
```

Use the following MML scripts to configure the boards.

B.3.2.1 PCI334

```
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,PORTS=4;
```



```
MOD-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,PORTNUM=0,TYPE=DCE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,PORTNUM=1,TYPE=DCE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,PORTNUM=2,TYPE=DTE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MOD-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,PORTNUM=3,TYPE=DTE,
    BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=0,CONF=ON;
```

B.3.2.2 PCI370 (T1)

```
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=sbs334,INST=1;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,SPAN=B,
    LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,SPAN=A,
    LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,LINE_NTFY=OFF;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTNUM=2,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTNUM=3,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=0,LISTENSPAN=HDLC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=0,LISTENSPAN=HDLC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=3;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=1,CONF=ON;
```

B.3.2.3 PCI372 (E1)

If PCI372 is a E1 (120 ohm) board:

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=pci334,INST=2;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,SPAN=B,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE,
    LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,SPAN=A,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE,
    LINE_NIFTY=OFF;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=2,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=3,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEB,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
    PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
    PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEA,
    PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEA,
    PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,CONF=ON;
```

If PCI372 is a E1 (75 ohm) board:

```
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=2;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,SPAN=B,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE,
    LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,SPAN=A,
    LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE,
    LINE_NTFY=OFF;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
```

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=2,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTNUM=3,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEB,
PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEB,
PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
PORTSLOT=2,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=HDLC,
PORTSLOT=3,LISTENSPAN=LINEA,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEA,
PORTSLOT=1,LISTENSPAN=HDLC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,PORTSPAN=LINEA,
PORTSLOT=2,LISTENSPAN=HDLC,LISTENSLOT=3;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci334,INST=2,CONF=ON;

B.3.2.4 PCI370PQ (T1)

ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,SPAN=16;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,SPAN=B,L
INE_FRMMOD=T1ESF, LINE_COD=T1B8ZS,LINE_LPBK=NONE,LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,SPAN=A,
LINE_FRMMOD=T1ESF, LINE_COD=T1B8ZS,LINE_LPBK=NONE,LINE_NTFY=OFF;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=0,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=1,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=2,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=3,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=4,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=5,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=6,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=7,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=8,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=9,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;

```
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=10,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=11,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=12,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=13,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=14,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTNUM=15,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
    PORTSLOT=2,LISTENSPAN=LINEB,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
    PORTSLOT=3,LISTENSPAN=LINEB,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=4,LISTENSPAN=LINEB,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=5,LISTENSPAN=LINEB,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
    PORTSLOT=6,LISTENSPAN=LINEB,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
    PORTSLOT=7,LISTENSPAN=LINEB,LISTENSLOT=7;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=0,LISTENSPAN=HLDC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEB,
    PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=7;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=8,LISTENSPAN=LINEA,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
    PORTSLOT=9,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
    PORTSLOT=10,LISTENSPAN=LINEA,LISTENSLOT=2;
```



```
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
      PORTSLOT=11,LISTENSPAN=LINEA,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
      PORTSLOT=12,LISTENSPAN=LINEA,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HDLC,
      PORTSLOT=13,LISTENSPAN=LINEA,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
      PORTSLOT=14,LISTENSPAN=LINEA,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=HLDC,
      PORTSLOT=15,LISTENSPAN=LINEA,LISTENSLOT=7;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=0,LISTENSPAN=HLDC,LISTENSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,PORTSPAN=LINEA,
      PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=15;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=0,CONF=ON;
```

B.3.2.5 PCI372PQ (E1)

If PCI372PQ is a E1 (120 ohm) board:

```
ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=pci3xpq,INST=1;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,SPAN=B,
      LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE,
      LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,SPAN=A,
      LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I120,LINE_LPBK=NONE,
      LINE_NTFY=OFF;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=0,BAUD=64000,
      LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=1,BAUD=64000,
      LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=2,BAUD=64000,
      LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=3,BAUD=64000,
      LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=4,BAUD=64000,
      LPBKMODE=NONE,IDLEDETECT=ON;
```

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=5,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=6,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=7,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=8,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=9,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=10,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=11,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=12,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=13,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=14,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=15,BAUD=64000,
LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=2,LISTENSPAN=LINEB,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=3,LISTENSPAN=LINEB,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=4,LISTENSPAN=LINEB,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=5,LISTENSPAN=LINEB,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=6,LISTENSPAN=LINEB,LISTENSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=7,LISTENSPAN=LINEB,LISTENSLOT=8;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=5;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=8,LISTENSPAN=HLDC,LISTENSLOT=7;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=8,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=9,LISTENSPAN=LINEA,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=10,LISTENSPAN=LINEA,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=11,LISTENSPAN=LINEA,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=12,LISTENSPAN=LINEA,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=13,LISTENSPAN=LINEA,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=14,LISTENSPAN=LINEA,LISTENSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=15,LISTENSPAN=LINEA,LISTENSLOT=8;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=8,LISTENSPAN=HLDC,LISTENSLOT=15;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,CONF=ON;

If PCI372PQ is a E1 (75 ohm) board:

ADD-SS7BOARD:HOSTNAME=zeus,BOARDNM=pci3xpq,INST=1;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,SPAN=B,
LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE,
LINE_NTFY=OFF;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,SPAN=A,
LINE_FRMMOD=E1CR4,LINE_COD=E1HDB3,LINE_IMP=I75,LINE_LPBK=NONE,
LINE_NTFY=OFF;

```
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=0,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=1,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=2,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=3,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=4,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=5,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=6,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=7,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=8,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=9,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=10,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=11,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=12,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=13,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=14,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTNUM=15,BAUD=64000,
    LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
    PORTSLOT=0,LISTENSPAN=LINEB,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
    PORTSLOT=1,LISTENSPAN=LINEB,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
    PORTSLOT=2,LISTENSPAN=LINEB,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
    PORTSLOT=3,LISTENSPAN=LINEB,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
    PORTSLOT=4,LISTENSPAN=LINEB,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
    PORTSLOT=5,LISTENSPAN=LINEB,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
    PORTSLOT=6,LISTENSPAN=LINEB,LISTENSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
    PORTSLOT=7,LISTENSPAN=LINEB,LISTENSLOT=8;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
    PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=0;
```

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEB,
PORTSLOT=8,LISTENSPAN=HLDC,LISTENSLOT=7;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=8,LISTENSPAN=LINEA,LISTENSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=9,LISTENSPAN=LINEA,LISTENSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=10,LISTENSPAN=LINEA,LISTENSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=11,LISTENSPAN=LINEA,LISTENSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=12,LISTENSPAN=LINEA,LISTENSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HDLC,
PORTSLOT=13,LISTENSPAN=LINEA,LISTENSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=14,LISTENSPAN=LINEA,LISTENSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=HLDC,
PORTSLOT=15,LISTENSPAN=LINEA,LISTENSLOT=8;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=1,LISTENSPAN=HLDC,LISTENSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=2,LISTENSPAN=HLDC,LISTENSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=3,LISTENSPAN=HLDC,LISTENSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=4,LISTENSPAN=HLDC,LISTENSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=5,LISTENSPAN=HLDC,LISTENSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=6,LISTENSPAN=HLDC,LISTENSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=7,LISTENSPAN=HLDC,LISTENSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,PORTSPAN=LINEA,
PORTSLOT=8,LISTENSPAN=HLDC,LISTENSLOT=15;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=pci3xpq,INST=1,CONF=ON;

B.3.3 ARTIC1000 Compact PCI Bus Configuration

Assuming that the configuration is made on a CompactPCI bus host "cosmos" and the output of getcfg command is as follows:

```
Driver    Board Type Slot Instance
-------- ---------- ---- --------
artic8260 artic1000  0    0
artic8260 artic1000  1    1
```

Then the following MML scripts can be used to configure the boards.

B.3.3.1 ARTIC1000 with RTB, Span-1, Span-2, Span-3, Span-4 are T1, rear-access, 16 ports configured

```
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTS=16,CLOCKMODE=LINE,
CLOCKSPAN=1;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,SPAN=1,
LINE_TYP=T1,LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,
LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,SPAN=2,
LINE_TYP=T1,LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,
LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,SPAN=3,
LINE_TYP=T1,LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,
LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,SPAN=4,
LINE_TYP=T1,LINE_FRMMOD=T1ESF,LINE_COD=T1B8ZS,LINE_LPBK=NONE,
LINE_ACCS=REAR;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=0,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=1,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=2,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=3,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=4,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=5,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=6,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=7,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=8,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=9,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=10,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
```


MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=11,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=12,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=13,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=14, BAUD=64000,LPBK-
MODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=15,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=0,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=1,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=2,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=3,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=4,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=5,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=6,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=7,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=3;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=8,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=9,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=10,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=11,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,

```
DESTSLOT=12,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=13,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=14,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=15,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=15;
MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,CONF=ON;
```

B.3.3.2 ARTIC1000 with RTB, Span-1, Span-2, Span-3, Span-4 are E1, rear-access, 16 ports configured

```
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTS=16,
CLOCKMODE=LINE,CLOCKSPAN=1;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=1,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=2,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=3,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=REAR;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=4,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=REAR;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=0,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=1,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=2,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=3,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=4,
```

```
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=5,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=6,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=7,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=8,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=9,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=10,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=11,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=12,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=13,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=14, BAUD=64000,LPBK-
MODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=15,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=0,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=1,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=2,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=3,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=4,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=5,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=6,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=7,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=4;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=5;
```

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=8,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=9,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=10,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=11,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=12,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=13,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=14,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=15,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=15;
MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,CONF=ON;

B.3.3.3 ARTIC2000 or ARTIC1000 (with or without RTB), Span-1, Span-2, Span-3, Span-4 are E1, front-access, 24 ports configured, CT Bus A Master,

CTBUS spans 0 and 1 mapped to ports 16 to 23
ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTS=24,
CLOCKMODE=LINE,CLOCKSPAN=1;

MODIFY-CTBUS: HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,COMP=ECTF,
C8A=ON,GRP_A=2048;

MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=1,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=FRONT;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=2,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,


```
LINE_LPBK=NONE,LINE_ACCS=FRONT;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=3,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=FRONT;
MODIFY-LINE:HOSTNAME=cosmos,BOARDNM=artic8260,INST=1,SPAN=4,
LINE_TYP=E1,LINE_FRMMOD=E1CRC4,LINE_COD=E1HDB3,LINE_IMP=I120,
LINE_LPBK=NONE,LINE_ACCS=FRONT;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=0,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=1,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=2,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=3,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=4,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=5,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=6,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=7,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=8,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=9,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=10,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=11,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=12,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=13,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=14, BAUD=64000,LPBK-
MODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=15,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=16,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=17,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=18,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=19,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=20,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=21, BAUD=64000,LPBK-
MODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=22,
```

```
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=23,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=0,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=1,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=2,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=3,ORIGTYPE=LINE,ORIGSPAN=1,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=4,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=5,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=6,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=7,ORIGTYPE=LINE,ORIGSPAN=2,ORIGSLOT=4;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=1,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=2,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=8,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=9,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=10,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=11,ORIGTYPE=LINE,ORIGSPAN=3,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=12,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=13,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=14,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=15,ORIGTYPE=LINE,ORIGSPAN=4,ORIGSLOT=4;
```

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=8;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=9;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=10;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=3,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=11;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=12;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=13;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=14;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=LINE,
DESTSPAN=4,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=15;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=16,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=17,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=18,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=19,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=20,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=21,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=22,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=23,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=16;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=17;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=18;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=19;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=20;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=5,ORIGTYPE=HDLC,ORIGSLOT=21;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=6,ORIGTYPE=HDLC,ORIGSLOT=22;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=7,ORIGTYPE=HDLC,ORIGSLOT=23;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,CONF=ON;

B.3.3.4 ARTIC2000 or ARTIC1000 (with or without RTB), 8 ports configured, CT Bus A Clock Slave, Fallback to CT Bus B Clock, CTBUS spans 0 and 1 mapped to ports 0 to 7

ADD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTS=8;

MODIFY-CTBUS: HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,REFCLK=C8A,
REFINV=OFF,FBMODE=C8B,COMP=ECTF,GRP_A=2048;

MODIFY-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0, CLOCKMODE=REMOTE;

MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=0,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=1,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=2,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=3,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=4,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=5,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=6,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;
MODIFY-PORT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,PORTNUM=7,
BAUD=64000,LPBKMODE=NONE,IDLEDETECT=ON;

MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=0,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=1,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=2,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=3,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=4,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=5,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=6,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,DESTTYPE=HDLC,
DESTSLOT=7,ORIGTYPE=CTBUS,ORIGSPAN=0,ORIGSLOT=7;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=0,ORIGTYPE=HDLC,ORIGSLOT=0;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=1,ORIGTYPE=HDLC,ORIGSLOT=1;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=2,ORIGTYPE=HDLC,ORIGSLOT=2;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=3,ORIGTYPE=HDLC,ORIGSLOT=3;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,


```
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=4,ORIGTYPE=HDLC,ORIGSLOT=4;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=5,ORIGTYPE=HDLC,ORIGSLOT=5;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=6,ORIGTYPE=HDLC,ORIGSLOT=6;
MODIFY-TIMESLOT:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,
DESTTYPE=CTBUS,DESTSPAN=1,DESTSLOT=7,ORIGTYPE=HDLC,ORIGSLOT=7;

MOD-SS7BOARD:HOSTNAME=cosmos,BOARDNM=artic8260,INST=0,CONF=ON;
```

This page is intentionally blank.

Appendix C: Alarms

C.1 Introduction

This chapter lists the NewNet Communications Technologies, LLC alarms that apply to any Distributed7 configuration. When required, the Technical Assistance Center (TAC) can be reached by phone at *(800) 416-1624 US* or by email at *nnsupport@ss8.com*.

This chapter includes a listing of NewNet Communications Technologies, LLC system alarms, by group then alarm number.

C.1.1 Alarm Severities

The alarm severities and their descriptions are listed below:

Information Alarms

Information alarms generally report system software events that occur when the system is on run. In most cases, information alarms do not affect the functional state of the system software, and they can be ignored. These alarms are also helpful for system diagnosis or verification of any faults occurring in the system.

Minor Alarms

Minor alarms are system diagnostics alarms useful for maintenance purposes. They generally indicate a malfunctioning module or a module failure. Resolve minor alarms as soon as possible, as ignoring them can result in minor system problems.

Major Alarms

Major alarms indicate that an event has occurred that affects the normal operations of the system. For example, a signaling link that was in service goes out of service. Major alarms should not be ignored and immediate corrective action is required in most cases.

Critical Alarms

Critical alarms signal failures that may result in total failure of system devices or peripherals.

Important*: Critical alarms must not be ignored.*

Fatal Alarms

Fatal alarms signal an extremely hazardous condition in the system. The system + be shut down immediately.

C.1.2 Alarm Groups and Trouble Tables

The system alarms have been categorized into various groups. Each group is responsible for different types of alarms. All alarm tables are listed in order of alarm number. The tables list the severity, type of alarm, text message, and the action to take when the alarm occurs. The type of alarm is either EVENT or SET_ALARM. Alarms that are SET_ALARM type are tracked in the system and can be displayed with the DISPLAY-STRDALM command. The alarms are cleared from the system with the DELETE-STRDALM command.

The alarm groups are:

- TRMOD - translation module (820101 - 820104)
- SPM - signaling point (SP) management (830101 - 830708)
- UPM - user part management, such as MTP Level 3 (840101 - 840401)
- SCCP - service connection control part management (850101 - 85150a)
- MTPL1 - MTP Level 1 (860101 - 860136)
- MTPL2 - MTP Level 2 (870101 - 870152)
- ETMOD - Ethernet test module (880101 - 880602)
- ISUP - ISUP management (890101 - 8901d3)
- APM - Application Process Management (8b0101 - 8b030a)
- OMAP - operation, maintenance, and administration part (8d0101 - 8d0114)
- NIMOD - connection management (920101 - 920401)
- TCAP - TCAP driver (950101 - 950806)
- TCMOD - TCAP over TCP/IP connection management (960101 - 960206)
- DKM - Distributed kernel memory (970101 - 970805)
- ISUPMOD - ISUP management - distributed (990201 - 990406)
- GSM-A - GSM-A Interface (9a0101 - 9a0401)
- PMMOD - passive monitor kernel module (9c0101 - 9c0103)
- PMON - passive monitor service layer (9b0101 - 9b0109)

Table C-1: TRMOD Alarm Group

Alarm No.	Severity	Type	Message	Operation
820101	Major	EVENT	TRMOD: Invalid incoming SP number [sp=%d]	MTP/L2 sent a message having an invalid SP number. Detach the board and attach it again.
820102	Major	EVENT	TRMOD: Invalid incoming message [stype=%d mtype=%d]	MTP/L2 sent a invalid message type. Detach the board and attach it again.

Table C-1: TRMOD Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
820103	Major	EVENT	TRMOD: Invalid incoming alarm [grp=0x%x mod=0x%x num=0x%x]	MTP/L2 sent an alarm which is not defined in the MTP scope. Record in an event log.
820104	Major	EVENT	TRMOD: UPM STREAMS multiplexor not found [sp=%d]	UPM multiplexer for the given SP could not be found. If the system is about to be stopped, no action is necessary. Otherwise check if the ***upmd*** process is running a nd if not, start ***upmd***.

Table C-2: SPM Alarm Group

Alarm No.	Severity	Type	Message	Operation
830101	Major	EVENT	SPM: Unexpected STREAMS message [msg=0x%x]	SPM received an unexpected STREAMS message from a lower module (e.g. NIMOD, TCMOD). Message is discarded. No action necessary.
830102	Critical	EVENT	SPM: Error/Hangup condition on STREAMS queue [q=0x%x]	M_ERROR or M_HANGUP STREAMS message has been received. To restore the system to operational state, stop and restart the ServiceController 1510 software.
830201	Critical	EVENT	SPM: Invalid SP number [sp=%d] A	message destined to an SS7 object on the local host has an invalid SP# (out of 0-7 range). The message is discarded. Could be a problem with the device driver or TRMOD. Report problem to TAC
830202	Critical	EVENT	SPM: SP not connected [sp=%d]	A message destined to an SS7 object on the local host can not be delivered because the SS7 object is not running. The message is discarded. No action necessary
830203	Minor	EVENT	SPM: Invalid IPC key and/or process ID [key=%d pid=%d]	A received message has been encoded with incorrect L_IPCKEY addressing information. The message is discarded. No action necessary
830204	Minor	EVENT	SPM: Null upper STREAMS queue	The STREAMS queue associated with a message receiver is invalid. The receiver process may have exited. The message is discarded. No action necessary
830205	Critical	EVENT	SPM: STREAMS queue full [q=0x%x]	The STREAMS queue associated with the upstream message receiver has exceeded its limit. The message is discarded and a count of all discards is kept. Use ebs_qstat to retrieve the counts. Use spm_qparams API calls to reset the queue management limits.

Table C-2: SPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
830206	Critical	EVENT	SPM: Exclusiveness violation [key=%d pid=%d]	Conflicting registration information has been received from a remote host for an exclusively registered process that already has a single instance. The local host will not allow the remote registration. No action necessary if the process is an Signaling Gateway daemon. If the process is an application process, execute ebs_ps on all hosts to obtain current data on the processes. Manual intervention, such as restarting the processes, may be necessary. This alarm may appear after a partitioned LAN gets reconnected.
830301	Minor	EVENT	SPM: Unattended STREAMS queue - Message discarded [q=0x%x]	STREAMS queue in the upstream direction is invalid. The process may have terminated or the system is shutting down. The message is discarded. No action necessary.
830501	Info	EVENT	SPM: TCP/IP connection established [link=%d inetaddr=0x%x]	This alarm indicates that a kernel-level tcp/ip connection has been established between the local host and the specified remote host. Kernel-level tcp/ip connections are essential for inter-host communication in a distributed product configuration. If and when a redundant-LAN configuration is in use, multiple tcp/ip connections exist between each host pair.
830502	Info	EVENT	SPM: TCP/IP connection removed [link=%d inetaddr=0x%x]	This alarm indicates that a kernel-level tcp/ip connection between the local host and the specified remote host has been removed. When all kernel-level tcp/ip connections to a particular remote host are removed, it is no longer possible for the local host to exchange any inter-host information with that remote host.
830503	Info	EVENT	SPM: TCP/IP connection switched from STANDBY to ACTIVE [link=%d]	This alarm indicates a switchover took place in the Local Area Network (LAN) interfaces that are in use. This alarm is generated only when dual-LAN is in use and it becomes necessary to swap the active/hot-standby LAN interfaces. By default, the very first TCP/IP connection established to a specified remote host is marked as active, whereas the subsequent TCP/IP connections set up to that remote host are marked as standby. It is only when the active connection is disconnected that Distributed7 attempts to substitute the standby TCP/IP connection for the same destination as the active connection. No alarm gets generated when a standby TCP/IP connection is removed from or restored to service.
830601	Major	EVENT	SPM: Invalid I/O control command [cmd=%d]	An unexpected M_IOCTL reply has been received from a lower STREAMS module. It is discarded. No action necessary

Table C-2: SPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
830701	Minor	EVENT	SPM: Invalid IPC key and/or process ID [key=%d pid=%d]	A STREAMS module has found the destination address for an IPC message to be invalid or incorrect. The message is discarded. No action required.
830702	Critical	EVENT	SPM: STREAMS queue full [q=0x%x]	The STREAMS queue associated with the upstream message receiver has exceeded its limit. The message is discarded and a count of all discards is kept. Use ebs_qstat to retrieve the counts. Use spm_qparams API calls to reset the queue management limits.
830703	Minor	EVENT	SPM: Cannot route message to its destination [msg=0x%x size=%d]	The STREAMS read-side queue associated with a message receiver is invalid. The message is discarded.
830704	Critical	EVENT	SPM: TCP/IP address not accessible [inetaddr=0x%x]	The destination host address associated with a message is invalid. The message is discarded. The host may not be configured to be in the distributed environment or it is not connected to the local host. Convert the specified IP address to a string with the spm_inet_host() function and verify the corresponding host is configured correctly. If it is, use ebs_showlink to verify a TCP/IP connection exists between the host and the local host.
830705	Critical	EVENT	SPM: Link not connected and/or unavailable for use [link=%d]	The STREAMS connection that SPM attempted to route downstream on is currently unavailable due to an improper link or an error condition. Use configuration management tools to remove and re-establish the connection. Use ebs_showlink to identify the connection.
830706	Critical	EVENT	SPM: Invalid SS7 board ID [id=%d]	The STREAMS connection number specified for routing an L_SS7MSG message downstream is invalid (outside the range). The message is discarded. Report problem to TAC.

Table C-2: SPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
830707	Critical	EVENT	SPM: Heartbeat failed on TCP/IP connection [link=%d]	Indicates a failure of the heartbeat mechanism over the specified TCP/IP connection between the local host and a remote host. No heartbeat response was received from the remote end after sending 2 consecutive heartbeat requests. The link cannot be used for inter-host message exchange. A problem most likely exists with the LAN or remote host. Use ebs_showlink to identify the remote host associated with the link specified in the alarm. If both LAN and remote host are normal, call TAC. If the periodicity of the heartbeat message exchange is too low (default is 500ms), this alarm may occur when a problem does not exist. If a redundant-LAN configuration exists, another TCP/IP connection to the same destination will become active.
830708	Info	EVENT	SPM: Heartbeat restored on TCP/IP connection [link=%d]	Indicates that the local host has recovered from a previous heartbeat loss situation. Synchronization will automatically occur between the hosts. No action required

Table C-3: UPM Alarm Group

Alarm No.	Severity	Type	Message	Operation
840101	Minor	EVENT	UPM: Message discarded (Default case) [func=%d msg=0x%x task=%d]	An invalid case occurred. Record in an event log.
840102	Critical	EVENT	UPM: Message discarded (Invalid SP number) [func=%d sp=%d]	Invalid SP number found. Record in an event log.
840103	Critical	EVENT	UPM: Message discarded (Null SP table) [msg=0x%x line=%d mod=%d]	MTP table is NULL. If it is not the shutdown case, then stop the MTP part and restart it.
840104	Critical	EVENT	UPM: Message discarded (Null UPM bottom queue) [func=%d msg=0x%x/0x%x]	UPM bottom queue is NULL. If it is not the shutdown case, then stop the MTP part and restart it.
840105	Major	EVENT	UPM: Message discarded (Null message header) [func=%d msg=0x%x]	Header portion of a message could not be found. Message is discarded. Record in an event log.
840106	Major	EVENT	UPM: Message discarded (Null message pointer) [func=%d]	Message Pointer of a message is NULL. Message is discarded. Record in an event log.
840107	Critical	EVENT	UPM: Message discarded (Error/Hangup condition) [func=%d q=0x%x]	M_ERROR or M_HANGUP STREAM message has been received. To restore the system to operational state, stop and restart the Distributed7 software.
840108	Critical	EVENT	UPM: Message discarded (Null STREAMS queue) [func=%d]	The STREAMS queue of the receiver is NULL. A message for a de-registered process has been received. Record in an event log.
840109	Minor	EVENT	UPM: Message discarded (STREAMS queue full) [func=%d q=0x%x]	The STREAMS queue associated with the upstream message receiver has exceeded its limit. The message is discarded and the counts of all discards is kept. Use ***ebs_qstat*** to retrieve the counts. Use ***spm_qparams()*** to reset the queue management limits.
84010a	Major	EVENT	UPM: Message discarded (Invalid UP) [func=%d up=0x%x]	Invalid User Part number is received. Check if this user part exists in the system. If it does, then check the status of the user part with ***ebs_ps*** comment.
84010b	Critical	EVENT	UPM: Message discarded (Null next STREAMS queue) [func=%d q=0x%x]	UPM connection to SPM driver has been destroyed. To restore the system to the operational state, stop MTP/L3 software and restart
84010c	Major	EVENT	UPM: Message discarded (Invalid IPC key) [func=%d msg=0x%x - 0x%x]	Destination of the message is not valid anymore. The message is discarded. No action.
84010d	Major	EVENT	UPM: Message discarded (SP record not found) [func=%d rc=0x%x msg=0x%x]	SP record could not be found due to the [rc] error code. Record in an event log.

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
84010e	Major	EVENT	UPM: Message discarded (LSET record not found) [func=%d rc=0x%x dpc=0x%x]	Linkset record could not be found due to the [rc] error code. Record in an event log.
84010f	Major	EVENT	UPM: Message discarded (Null queue pointer) [func=%d q=0x%x msg=0x%x]	Device queue pointer is NULL. Record in an event log.
840110	Major	EVENT	UPM: Message discarded (DKM not operational) [func=%d msg=0x%x id=%d]	DKM is not operational. Record in an event log. Use ***dkm_list -hd*** to get the dkm status and, if it is not running, then restart Distributed7.
840111	Major	EVENT	UPM: Message discarded (UPM closed) [func=%d msg=0x%x]	UPM driver is in the closing process. The message is discarded. Use ***ebs_ps*** to check the existence of UPM driver.
840112	Major	EVENT	UPM: Message discarded (Cannot put message on queue) [func=%d msg=0x%x - 0x%x]	The message could not be passed through the UPM driver due to the flow control conditions. The message is discarded. Record in an event log. Use ***ebs_ps -m*** to check the state of the upm multiplexor.
840113	Major	EVENT	UPM: Message discarded (Get IPC key failed) [func=%d msg=0x%x - 0x%x]	Message destined to a specific process is discarded because that process does not exist anymore. Record in an event log.
840114	Major	EVENT	UPM: Message discarded (Invalid task/ host index) [func=%d idx=%d msg=0x%x]	An invalid task id or host id is detected. Message corresponding to this task/host is discarded. Record in an event log.
840115	Major	EVENT	UPM: Message discarded (Pointer mismatch) [func=%d msg=0x%x - 0x%x]	The back pointer of a message does not point to the original message block. The message is discarded. Record in an event log.
840201	Minor	EVENT	MTP-L3: HMDC discrimination error [ni=%d sio=0x%x]	Network indicator (NI) of incoming message does not match with the current setting of MTP/L3. Check the current NI with **display-sp** MML command and compare with the incoming NI
840202	Minor	EVENT	MTP-L3: HMDC discrimination error [dpc=0x%x]	Destination Point Code (DPC) of incoming message is not the Own Point Code. Check the own point code with **display-sp** mml command and compare with the dpc
840203	Minor	EVENT	MTP-L3: HMDC input event [input=0x%x]	Invalid message type received from MTP/L2 software. The message is discarded. Record in an event log
840204	Minor	EVENT	MTP-L3: HMDT SLC event [h1h0=0x%x slc=%d]	The SLC value of the incoming message either is invalid or not set to zero. Check the h0h1 value and the incoming SLC
840205	Major	EVENT	MTP-L3: HMDT DPC event [h1h0=0x%x dpc=0x%x]	The concerned point code in the incoming message is not defined in the MTP database. Check all defined destinations with the **display-rtset** mml command and compare with the incoming dpc

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840206	Minor	EVENT	MTP-L3: HMDT OPC event [h1h0=0x%x opc=0x%x]	The originating point code of the incoming message is not defined in the MTP database. Check all defined destinations with the **display-rtset** mml command and compare with the incoming opc
840207	Minor	EVENT	MTP-L3: HMDT FSNC event [h1h0=0x%x fsnc=0x%x]	The BSNT/FSNC value in the incoming message is not valid. For ASNI networks the 5th bit of FSNC must be 0. For all networks the value of BSNT must be lower than 128. Check the protocol and the incoming message.
840208	Minor	EVENT	MTP-L3: HMDT MSU event [h1h0=0x%x lset=%d]	Invalid MSU received at HMDT. The message is discarded. No action is necessary
840209	Minor	EVENT	MTP-L3: HMDT spare bits event [h1h0=0x%x spare=0x%x]	Spare bits of incoming message is not set to zero. The spare bit handling is different in different protocols and messages. Check the MTP/L3 protocol and the incoming message.
84020a	Minor	EVENT	MTP-L3: HMDT input event [input=0x%x]	Invalid message type received at HMDT. The message is discarded. No action is necessary
84020b	Info	EVENT	MTP-L3: HMDT TFM event [h1h0=0x%x pc=0x%x]	An MTP Transfer Message is received from Network. This indicates a potential state change in destination or route. No action is necessary
84020c	Info	EVENT	MTP-L3: HMRT SLC event [h1h0=0x%x dpc=0x%x]	No routing link could be found for an outgoing message. Check the link status of MTP with the **display-linkstat** mml command and destination status with the **display-rtset** mml command
84020d	Info	EVENT	MTP-L3: HMRT DPC event [dpc=0x%x]	The concerned destination point code of the outgoing TFC message is not defined in the MTP/L3. Check the destinations with the **display-rtset** command and compare with the given dpc
84020e	Minor	EVENT	MTP-L3: HMRT hold event [h1h0=0x%x dpc=0x%x]	The given (h0h1) SNM message could not be sent to SS7 network because of inaccessibility of the links. No action is necessary
84020f	Minor	EVENT	MTP-L3: HMCG message discarded [dpc=0x%x status=%d]	Outgoing message is discarded due to the congestion situation in MTP/L3 and message priority of the outgoing message is less than the current congestion status. Check the message priority of the outgoing message
840210	Info	EVENT	MTP-L3: HMRT input event [state=%d msg=0x%x]	

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840211	Info	EVENT	MTP-L3: HMRT state event [state=%d msg=0x%x]	A message is received at HMRT task in unexpected case. The message is discarded. If the states of MTP components get confused after this alarm, stop and restart the MTP/L3 software on this host
840212	Info	EVENT	MTP-L3: HMRT MSU event [h1h0=0x%x lset=%d]	Invalid MSU is released from User parts in HMRT task. Check the message id (h0h1)
840213	Minor	EVENT	MTP-L3: HMRT inaccessible SP event [dpc=0x%x opc=0x%x]	A message is received for an inaccessible destination. (Only if MTP is configured as STP). The message is discarded, no action is necessary
840214	Minor	EVENT	MTP-L3: SLTC LSET event [lset=%d lset=%d]	An SLTC message is received from an invalid linkset. Check the destination point code of the message and the link identifier
840215	Minor	EVENT	MTP-L3: HMDT local UP unavailable [dpc=0x%x si=%d]	Local user part where the incoming message is destined to is unavailable. Check the local user part status
840216	Minor	EVENT	MTP-L3: HMRT MSU discarded, RS buffer overflow [dpc=0x%x msg=0x%x]	Destination buffer size exceeded. The outgoing user part message is discarded. Record in an event log
840217	Minor	EVENT	MTP-L3: HMRT MSU discarded, LS buffer overflow [dpc=0x%x msg=0x%x]	Linkset buffer size exceeded. The outgoing user part message is discarded. Record in an event log
840218	Minor	EVENT	MTP-L3: HMRT MSU discarded, no routing [dpc=0x%x msg=0x%x]	Outgoing message is discarded due to the inaccessibility of the destination. Check the status of the destination with the **display-rtset** mml command and make sure that the user part is aware of this status
840219	Minor	EVENT	MTP-L3: HMRT MSU discarded, DPC inaccessible [dpc=0x%x msg=0x%x]	Outgoing message is discarded due to destination inaccessibility. Check the status of the destination with the **display-rtset** mml command, and make sure the user part is aware of this status.
84021a	Minor	EVENT	MTP-L3: HMDT local UP unavailable (UPU) [dpc=0x%x si=%d]	UPU message is received for non-existing user part. Check if the concerning user part exists in the network.
84021b	Minor	EVENT	MTP-L3: SMH MSU discarded, own SP restarting [dpc=0x%x msg=0x%x]	The incoming/outgoing message is discarded due to the Own SP Restart procedure. The message flow is restored after the restart procedure is completed. Do not send messages during the Own SP Restart procedure.

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
84021c	Minor	EVENT	MTP-L3: SMH MSU discarded, adjacent SP restarting [dpc=0x%x]	The incoming/outgoing message has been discarded due to the “Adjacent SP Restart” procedure. The message flow is restored after the restart procedure is completed. No action is necessary
84021d	Minor	EVENT	MTP-L3: HMDT MSU discarded, resource outage [h1h0=0x%x dpc=0x%x]	System resources are not adequate enough in order to complete a SNM activity. Try to decrease the memory and CPU usage of the system while stopping some unnecessary processes, tools, etc.
84021e	Minor	EVENT	MTP-L3: HMDT MSU discarded, STP option not set [h0h1=0x%x msg=0x%x]	An STP message is received from the SS7 network, but current SP is not configured as an STP. Check the type of SP with the **display-sp** mml command, and correct if necessary.
84021f	Major	EVENT	MTP-L3: SMH LSET record not found [id=%d line=%d mod=%d]	Linkset record in distributed storage is not found, resulting in an inconsistent MTP database in the kernel. Stop and restart the MTP system in the local host.
840220	Major	EVENT	MTP-L3: SMH DEST record not found [dpc=0x%x line=%d mod=%d]	Destination record in distributed storage could not be found. This will result in an inconsistent MTP database in the kernel. This, stop and restart the MTP system in the local host
840221	Minor	EVENT	MTP-L3: SLTC SLC mismatch [slc=%d slc=%d]	The SLC number of the incoming SLTA message does not match with the previously sent SLTM message. Check if the SLC mapping of the remote end is correct
840222	Minor	EVENT	MTP-L3: Host inaccessible - Message discarded [inetaddr=0x%x sender=%d msg=0x%x]	Messages destined to a link which does not have host connection are discarded. Distributed7 on a host has terminated. Links on a terminating host must have already been stopped by the operator. Record the alarm.
840223	Minor	EVENT	MTP-L3: Circular route - Message discarded [dpc=0x%x line=%d]	Route to the destination was detected to be circular and hence the message is discarded. Check the configuration.
840224	Minor	EVENT	MTP-L3: HMDT discarded TRA received on non-direct linkset [h1h0=0x%x lset=%d]	A TRA/TRW is received over an indirect linkset. Message discarded. Correct problem at the remote end.
840301	Major	EVENT	MTP-L3: Linkset unavailable [lset=%d]	A linkset becomes unavailable. This means that all links in this linkset become unavailable, or the linkset is deactivated. Check the linkset status the **display-lset-stat** and **display-linkstat** mml commands and activate the linkset if necessary. If this is a physical problem, check the cables.

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840302	Minor	EVENT	MTP-L3: Link unavailable [lset=%d slc=%d]	An SS7 link becomes unavailable. this can be caused by either a physical connection problem or user may want to deactivate the link. Check the link status with the **display-linkstat** mml command and activate the link if necessary. If it is a physical problem, check the corresponding cable
840303	Info	EVENT	MTP-L3: Linkset available [lset=%d]	A linkset becomes available. This means that a link in this linkset becomes available
840304	Info	EVENT	MTP-L3: Link available [lset=%d slc=%d]	A link becomes available. This means that the link can carry traffic
840305	Minor	EVENT	MTP-L3: Link inhibited [lset=%d slc=%d]	The local inhibit request of operator is successful. The link can not carry user traffic until it is uninhibited
840306	Info	EVENT	MTP-L3: Link uninhibited [lset=%d slc=%d]	The link previously inhibited locally by the operator is now uninhibited
840307	Minor	EVENT	MTP-L3: Link uninhibiting not possible [lset=%d slc=%d]	The attempt to uninhibit the link which was already inhibited locally, could not be uninhibited. Check if the SS7 network responds to the outgoing LUN message with LUA and the content of the LUA is correct
840308	Minor	EVENT	MTP-L3: Link uninhibit timeout [lset=%d slc=%d]	During the attempt of uninhibiting a link, the timer which waits for the LUA (Link Uninhibited Ack) has expired. Check if the remote end of the link is operational
840309	Minor	EVENT	MTP-L3: Link activation failure [lset=%d slc=%d]	The link could not be aligned within the period of T19. check the physical connection and the remote end of the terminal
84030a	Minor	EVENT	MTP-L3: Link inhibit denied [lset=%d slc=%d]	The inhibit request is denied. This is because the remote MTP responds to the outgoing
84030b	Minor	EVENT	MTP-L3: Link local inhibit denied [lset=%d slc=%d]	Local inhibit request is denied. This is because there is one link remaining in the linkset that can carry traffic, and the user wants to inhibit that link. Try to activate another link in this linkset and retry to inhibit the requested link.
84030c	Minor	EVENT	MTP-L3: Link inhibit request timeout [lset=%d slc=%d]	During the attempt of inhibiting a link, the timer which waits for LIA (Link Inhibit Ack) has expired. Check if the remote MTP is operational.
84030d	Info	EVENT	MTP-L3: Link remotely inhibited [lset=%d slc=%d]	The link is remotely inhibited. No user messages can be carried on this link
84030e	Info	EVENT	MTP-L3: Link remotely uninhibited [lset=%d slc=%d]	The link, previously remotely inhibited, now is uninhibited by the remote end

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
84030f	Minor	EVENT	MTP-L3: TSRC state event [msg=0x%x]	An invalid message has been received by the TSRC task. This event my result in an inconsistency in MTP kernel data.
840310	Info	EVENT	MTP-L3: TSRC route event [dpc=0x%x task=%d]	A (re)routing operation has been completed in the TSRC task. New routing is selected. No action is necessary.
840311	Minor	EVENT	MTP-L3: LLSC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840312	Minor	EVENT	MTP-L3: LSAC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840313	Minor	EVENT	MTP-L3: RCAT state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840314	Minor	EVENT	MTP-L3: RSRT state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840315	Minor	EVENT	MTP-L3: RTAC state event [input=0x%x]	An internal or network message is received for an SNM task in an unexpected case. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, then try to deactivate and re-activate the corresponding entity. If this does not help, then restart the MTP/L3 software.
840316	Minor	EVENT	MTP-L3: RTCC state event [input=0x%x]	An internal or network message is received for an SNM task in an unexpected case. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, then try to deactivate and re-activate the corresponding entity. If this does not help, then restart the MTP/L3 software.
840317	Minor	EVENT	MTP-L3: RTPC state event [input=0x%x]	An internal or network message is received for an SNM task in an unexpected case. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, then try to deactivate and re-activate the corresponding entity. If this does not help, then restart the MTP/L3 software.

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840318	Minor	EVENT	MTP-L3: RTRC state event [input=0x%x]	An internal or network message is received for an SNM task in an unexpected case. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, then try to deactivate and re-activate the corresponding entity. If this does not help, then restart the MTP/L3 software.
840319	Minor	EVENT	MTP-L3: TCBC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84031a	Minor	EVENT	MTP-L3: TCOC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84031b	Minor	EVENT	MTP-L3: TCRC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
84031c	Minor	EVENT	MTP-L3: TFRC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84031d	Minor	EVENT	MTP-L3: TLAC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84031e	Minor	EVENT	MTP-L3: TRCC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84031f	Minor	EVENT	MTP-L3: TSFC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840320	Minor	EVENT	MTP-L3: TSRC delete route failed [dpc=0x%x lset=%d]	Deleting a route in kernel failed.

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840321	Critical	EVENT	MTP-L3: TSRC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840323	Major	EVENT	MTP-L3: MPRIO option not set [msg=0x%x]	An internal or network message is received for an SNM task in which the multiple priorities option (MPRIO) is needed, but MTP/L3 is not configured to support multiple priorities.
840324	Minor	EVENT	MTP-L3: SLTC test pattern mismatch [offset=%d size=%d]	Incoming SLTM test pattern does not match with the previously sent pattern. Check if network acts correctly.
840325	Minor	EVENT	MTP-L3: SLTC test pattern size mismatch [size=%d size=%d]	Incoming SLTM test pattern size does not match with the previously sent pattern size. Check if the network acts correctly.
840326	Info	EVENT	MTP-L3: SLTC input event [h1h0=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840327	Info	EVENT	MTP-L3: SLTC test timeout [lset=%d slc=%d]	Signaling Link Test Control procedure failed, because the remote end did not send SLTA (Test Ack) or sent incorrect SLTA. Check the remote end
840328	Info	EVENT	MTP-L3: SLTC test successful [lset=%d slc=%d]	Signaling Link Test Control procedure succeeded
840329	Info	EVENT	MTP-L3: TSRC linkset event [state=%d task=%d]	A (re)routing operation has been completed for linksets. New routing is selected. No action is necessary

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
84032a	Minor	EVENT	MTP-L3: RTXC input event [msg=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84032b	Minor	EVENT	MTP-L3: TPRC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
84032c	Info	EVENT	MTP-L3: Own SP restarting [spc=0x%x]	Own signaling point starts the SP RESTART procedure. During this period all messages is discarded
84032d	Info	EVENT	MTP-L3: Own SP restarted [spc=0x%x]	SP RESTART procedure has been completed.
84032e	Info	EVENT	MTP-L3: Adjacent SP restarting [dpc=0x%x]	The adjacent signaling point starts SP RESTART procedure. During this period all messages going to that destination is discarded
84032f	Info	EVENT	MTP-L3: Adjacent SP restarted [dpc=0x%x]	The adjacent signaling point finished the SP RESTART procedure
840330	Major	EVENT	MTP-L3: Destination accessible [dpc=0x%x]	A destination becomes accessible. This means user parts can start sending messages to that destination
840331	Major	EVENT	MTP-L3: Destination inaccessible [dpc=0x%x]	A destination point becomes inaccessible. this means all routes to that destination are unavailable. Check the links for each route going to this destination. If it si the network which prohibits the access of that destination, check the adjacent SP status

Table C-3: UPM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
840332	Minor	EVENT	MTP-L3: SLTC state event [state=%d input=0x%x]	A message is received in an SNM task in an unexpected case. This may be caused by a message coming from the network, or a message coming from the own system. This state mismatch can cause inconsistent MTP database in the kernel space. If this alarm appears frequently and the state of the corresponding entity (link, linkset, route, routeset) is not correct, try to deactivate and then activate the corresponding entity. If this does not help, restart the MTP/L3 software
840333	Major	EVENT	MTP-L3: SNM LSET record not found [%d line=%d mod=%d]	Linkset record in distributed storage not found, resulting in inconsistent MTP database in kernel. Stop and restart the MTP system in the local host.
840334	Major	EVENT	MTP-L3: SNM DEST record not found [dpc=0x%x line=%d mod=%d]	Destination record in distributed storage could not be found. This will result in inconsistent MTP database in kernel. Stop and restart the MTP system in the local host
840335	Major	EVENT	MTP-L3: GateWay process started [sp=%d]	Gateway process started
840336	Info	EVENT	MTP-L3: Own SP restarting aborted [spc=0x%x]	
840337	Info	EVENT	MTP-L3: Adjacent SP restarting aborted [dpc=0x%x]	
840401	Major	EVENT	UPM: GateWay process does not exist [prim=0x%x]	A message is destined to the Gateway process but the process could not be found. Check if the process is operational

Table C-4: SCCP Alarm Group

Alarm No.	Severity	Type	Message	Operation
850101	Minor	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
850102	Minor	EVENT	SCCP: Subsystem not replicated [ssn=%d pc=0x%x]	Check if the concerned subsystem has a mate. Define a mate for affected subsystem.
850103	Minor	EVENT	SCCP: No pending SOR for subsystem [ssn=%d pc=0x%x]	SCCP received and discarded a Subsystem Out of Service grant for a subsystem not marked for a state change. No action required.

Table C-4: SCCP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
850104	Minor	EVENT	SCCP: Subsystem not locally replicated [ssn=%d pc=0x%x]	SCCP received and discarded a Subsystem Out of Service request for a system that has no mate. Start a mate subsystem or ensure existing one is allowed, and then retry coordinated state change.
850105	Minor	EVENT	SCCP: Local mate subsystem prohibited [ssn=%d opc=0x%x]	Check if the local mate SS is allowed. Check if the local SS sends an N_STAT_ request with user in service. Activate local mate subsystem.
850106	Minor	EVENT	SCCP: Unequipped local mate subsystem	Check if the local mate application is running. Check if the local mate registers with spm_reg() API call. Activate local mate subsystem.
850107	Minor	EVENT	SCCP: SP has no SCCP [sp=%d ssn=%d]	Check if SCCP daemon (scmd) is running. Run ebs_start.
850108	Minor	EVENT	SCCP: Invalid SCMG format ID [id=%d ssn=%d]	SCCP received and discarded management message with unrecognized format. No action required.
850109	Minor	EVENT	SCCP: Unexpected message [sp=%d msg=0x%x]	SCCP received and discarded message with unrecognized message type. No action required.
850501	Critical	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
850502	Critical	EVENT	SCCP: Pointcode not adjacent [pc=0x%x]	Call TAC.
850503	Critical	EVENT	SCCP: Subsystem not locally replicated [ssn=%d pc=0x%x]	SCCP received and discarded a Subsystem Out of Service request for a subsystem that has no mate. Start a mate subsystem or ensure existing one is allowed, and then retry coordinated state change.
850504	Critical	EVENT	SCCP: Local mate subsystem prohibited	Call TAC.
850505	Critical	EVENT	SCCP: Invalid SCMG format ID [id=%d ssn=%d]	SCCP received and discarded management message with unrecognized format. No action required.
850601	Critical	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
850602	Critical	EVENT	SCCP: Pointcode not adjacent [pc=0x%x]	Call TAC.
850603	Critical	EVENT	SCCP: Subsystem not locally replicated [ssn=%d pc=0x%x]	SCCP received and discarded a Subsystem Out of Service request for a subsystem that has no mate. Start a mate subsystem or ensure existing one is allowed, and then retry coordinated state change.
850604	Critical	EVENT	SCCP: Local mate subsystem prohibited	Call TAC.

Table C-4: SCCP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
850605	Critical	EVENT	SCCP: Invalid SCMG format ID [id=%d ssn=%d]	SCCP received and discarded management message with unrecognized format. No action required.
850701	Critical	EVENT	SCCP: Error/Hangup condition on STREAMS queue [q=0x%x]	Record in an event log.
850702	Minor	EVENT	SCCP: Unexpected SS7 message [dev=%d msg=0x%x]	This is a non-SS7 alarm. It pertains to the UNIX/STREAMS. Call TAC.
850703	Major	EVENT	SCCP: Unexpected STREAMS message [dev=%d msg=0x%x]	This is a non-SS7 alarm. It pertains to the UNIX/STREAMS. Call TAC.
850704	Minor	EVENT	SCCP: Null STREAMS queue - Message discarded	Record in an event log.
850705	Critical	EVENT	SCCP: Null SCCP bottom STREAMS queue [dev=%d]	Call TAC.
850801	Minor	EVENT	SCCP: Cannot deliver message [pc=0x%x ssn=%d]	No instances of the target subsystem (local) are unavailable, or the subsystem instance did not request to receive messages.
850802	Minor	EVENT	SCCP: Invalid SCCP protocol	Check if the SCCP protocol is valid—must be connectionless. Re-send MSU with correct protocol.
850803	Minor	EVENT	SCCP: Unequipped subsystem [pc=0x%x ssn=%d]	Check if local SS application is running. Check if LCL registers through spm_reg() API call. Activate LCL SS application.
850804	Minor	EVENT	SCCP: SP has no SCCP [sp=%d]	Check if SCCP daemon is running. Run ebs_start.
850805	Minor	EVENT	SCCP: Unexpected message [sp=%d msg=0x%x]	SCCP received and discarded message with unrecognized message type. No action required.
850806	Minor	EVENT	SCCP: XUDT does not correspond to any reassembly process [sp=%d ssn=%d]	Neither of the ongoing assembly processes can be associated with the XUDT. No action required.
850807	Minor	EVENT	SCCP: Duplicate reassembly process [sp=%d ssn=%d]	Calling party address has already started and not yet terminated an assembly process with this unique ID. No action required.
850808	Minor	EVENT	SCCP: Unexpected error in outgoing MSU [sp=%d]	Call TAC.
850809	Critical	EVENT	SCCP: STREAMS queue full - Message discarded [sp=%d ssn=%d]	
850a01	Minor	EVENT	SCCP: Invalid SCCP protocol	Check if the SCCP protocol is valid—must be connectionless. Re-send MSU with correct protocol.
850a02	Minor	EVENT	SCCP: Invalid MSU [msg=0x%x]	Message received has invalid message type. No action required.

Table C-4: SCCP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
850a03	Minor	EVENT	SCCP: Received connection-oriented MSU	CO procedures are not supported. No action required.
850a04	Info	EVENT	SCCP: Routing failure, no return option	Check if RETURN option is set. Re-send MSU with RET is the desired option.
850a05	Minor	EVENT	SCCP: Unreasonable pointer value	The pointers in the SCCP message are inconsistent. Make sure pointers are correctly set.
850a06	Info	EVENT	SCCP: Unassigned return option	Unable to transfer UDT message to destination, i.e., message was discarded. Set return option of message before sending so messages are returned if a routing failure occurs.
850a07	Minor	EVENT	SCCP: Inconsistent address indicator	The actual length of the SCCP message was longer than indicated. Message discarded. No action required.
850a08	Minor	EVENT	SCCP: UDTS cannot be returned [pc=0x%x ssn=%d]	UDTS message could not be transferred and was discarded. No action required
850a09	Info	EVENT	SCCP: Unexpected message [sp=%d msg=0x%x]	SCCP received and discarded message with unrecognized message type. No action required.
850a0a	Critical	EVENT	SCCP: SCCP bottom STREAMS queue full - Message discarded [sp=%d msg=0x%x]	
850c01	Minor	EVENT	SCCP: Unknown pointcode [pc=0x%x sp=%d]	Check if PC is provisioned. Provision PC.
850d01	Minor	EVENT	SCCP: Unknown pointcode [pc=0x%x sp=%d]	Check if PC is provisioned. Provision PC.
850e01	Minor	EVENT	SCCP: Unknown pointcode [pc=0x%x sp=%d]	Check if PC is provisioned. Provision PC.
851001	Minor	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
851002	Info	EVENT	SCCP: Subsystem already allowed [ssn=%d pc=0x%x]	Information only. No action required.
851003	Minor	EVENT	SCCP: Invalid SCMG format ID [id=%d ssn=%d]	SCCP received and discarded management message with unrecognized format. No action required.
851004	Info	EVENT	SCCP: State change to ALLOWED [ssn=%d pc=0x%x]	A subsystem's state has been changed to allowed. No action required.
851005	Minor	EVENT	SCCP: SSA received from Network for local subsystem [ssn=%d pc=%x]	Unexpected message, i.e., unknown type, SCCP routing layer.
851101	Minor	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
851102	Info	EVENT	SCCP: Subsystem already prohibited [pc=0x%x ssn=%d]	Information only. No action required.

Table C-4: SCCP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
851103	Minor	EVENT	SCCP: Invalid SCMG format ID [id=%d ssn=%d]	SCCP received and discarded management message with unrecognized format. No action required.
851104	Critical	EVENT	SCCP: State change to PROHIBITED [ssn=%d pc=0x%x]	A subsystem's state has been changed to prohibited. No messages can be sent to it. The local SCCP will reject queries. No action required
851105	Minor	EVENT	SCCP: SSP received from Network for local subsystem [ssn=%d pc=%x]	Outgoing message cannot be passed to MTP because the MTP queue is congested.
851201	Minor	EVENT	SCCP: Unknown subsystem [pc=0x%x ssn=%d]	Check if the concerned subsystem is provisioned. Provision the subsystem.
851202	Minor	EVENT	SCCP: Unexpected message [sp=%d msg=0x%x]	Received message has invalid message type. No action required.
851401	Minor	EVENT	SCCP: Null SCCP bottom STREAMS queue [dev=%d]	Call TAC.
851402	Minor	EVENT	SCCP: STREAMS queue full [q=0x%x]	Queue is receiving more messages than it can handle, messages are being discarded No action required
851501	Info	EVENT	SCCP: Invalid SCCP protocol	Class 3 primitive or message was received for Class 2 connection. No action required.
851502	Info	EVENT	SCCP: Invalid connection-oriented message [msg=0x%x]	A connection-oriented message with invalid message type was received. No action required.
851503	Info	EVENT	SCCP: Subsystem not provisioned	Call TAC.
851504	Minor	EVENT	SCCP: No STREAMS queue for subsystem [pc=0x%x ssn=%d]	Subsystem de-registered when a primitive was sent to it. No action required.
851505	Minor	EVENT	SCCP: SP has no SCCP [sp=%d]	Call TAC.
851506	Info	EVENT	SCCP: Unexpected internal message [sp=%d msg=0x%x]	The primitive received is not applicable to current state of the connection. No action required.
851507	Info	EVENT	SCCP: Routing failure for connection-oriented message	Connection-oriented message could not be delivered to destination. Verify that remote SPC is accessible.
851508	Info	EVENT	SCCP: Unexpected connection state [sp=%d state=0x%x]	Call TAC.
851509	Info	EVENT	SCCP: Data size too large [size=%d]	Maximum amount of data allowed in a DT1 message is L_MAX_DT1_DATA_SIZE. Ensure requested data size does not exceed L_MAX_DT1_DATA_SIZE.
85150a	Info	EVENT	SCCP: Connection ID does not belong to subsystem [id=%d ssn=%d]	A subsystem issued a primitive for a connection that belongs to another subsystem. Check connection IDs used in primitives.

Table C-5: MTPL1 Alarm Group

Alarm No.	Severity	Type	Message	Operation
860101	Major	EVENT	MTP-L1: Framer red (RLOS) alarm ON [spmlinkno=%d span=%d]	Synchronization lost on the receive side of the line. 1) Check remote side is up 2) Check physical line connections.
860102	Major	EVENT	MTP-L1: Framer red (RLOS) alarm OFF [spmlinkno=%d span=%d]	Line synchronization established.
860103	Major	EVENT	MTP-L1: Framer receive carrier loss (RCL) alarm ON [spmlinkno=%d span=%d]	Carrier signal lost 1) Check remote side is up 2) Check physical line connections.
860104	Major	EVENT	MTP-L1: Framer receive carrier loss (RCL) alarm OFF [spmlinkno=%d span=%d]	Carrier signal restored.
860105	Major	EVENT	MTP-L1: Framer yellow (RYEL) alarm ON [spmlinkno=%d span=%d]	Yellow alarm (RAI) received. Wait for other party to disable RAI.
860106	Major	EVENT	MTP-L1: Framer yellow (RYEL) alarm OFF [spmlinkno=%d span=%d]	Yellow alarm (RAI) is off.
860107	Major	EVENT	MTP-L1: Framer blue (RBL) alarm ON [spmlinkno=%d span=%d]	Blue alarm (AIS) received. Wait for other party to disable AIS.
860108	Major	EVENT	MTP-L1: Framer blue (RBL) alarm OFF [spmlinkno=%d span=%d]	Blue alarm (AIS) is off.
860109	Minor	EVENT	MTP-L1: Framer loss of transmit clock (LTOC) alarm ON [spmlinkno=%d span=%d]	Transmit clock is out of sync with receive clock. Temporary problem. If persists may point to a hardware problem.
86010a	Minor	EVENT	MTP-L1: Framer loss of transmit clock (LOTC) alarm OFF [spmlinkno=%d span=%d]	Transmit clock is in sync with receive clock.
86010b	Info	EVENT	MTP-L1: Framer receive elastic store slip (RSLIP) alarm ON [spmlinkno=%d span=%d]	Receive side elastic store is slipping (deleting or repeating bits) due to the frequency differences between the board and the network clocks.Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
86010c	Info	EVENT	MTP-L1: Framer receive elastic store slip (RSLIP) alarm OFF [spmlinkno=%d span=%d]	Receive side elastic store is in sync.
86010d	Info	EVENT	MTP-L1: Framer receive elastic store empty (RESE) alarm ON [spmlinkno=%d span=%d]	Network clock is slower than board clock. Elastic store is empty and repeated a frame. Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
86010e	Info	EVENT	MTP-L1: Framer receive elastic store empty (RESE) alarm OFF [spmlinkno=%d span=%d]	Receive side elastic store is in sync.

Table C-5: MTPL1 Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
86010f	Info	EVENT	MTP-L1: Framer receive elastic store full (RESF) alarm ON [spmlinkno=%d span=%d]	Network clock is faster than board clock. Elastic store is full and deleted a frame. Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
860110	Info	EVENT	MTP-L1: Framer receive elastic store full (RESF) alarm OFF [spmlinkno=%d span=%d]	Receive side elastic store is in sync.
860111	Info	EVENT	MTP-L1: Framer jitter attenuator limit trip (JALT) alarm ON [spmlinkno=%d span=%d]	To much jitter in the line. Check network signal quality.
860112	Info	EVENT	MTP-L1: Framer jitter attenuator limit trip (JALT) alarm OFF [spmlinkno=%d span=%d]	Line signal levels are normal.
860113	Info	EVENT	MTP-L1: Framer transmit elastic store slip (TSLIP) alarm ON [spmlinkno=%d span=%d]	Transmit side elastic store is slipping (deleting or repeating bits) due to the frequency differences between the board and the network clocks. Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
860114	Info	EVENT	MTP-L1: Framer transmit elastic store slip (TSLIP) alarm OFF [spmlinkno=%d span=%d]	Transmit side elastic store is in sync.
860115	Info	EVENT	MTP-L1: Framer transmit elastic store empty (TESE) alarm ON [spmlinkno=%d span=%d]	Network clock is faster than board clock. Elastic store is empty and repeated a frame. Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
860116	Info	EVENT	MTP-L1: Framer transmit elastic store empty (TESE) alarm OFF [spmlinkno=%d span=%d]	Transmit side elastic store is in sync.
860117	Info	EVENT	MTP-L1: Framer transmit elastic store full (TESF) alarm ON [spmlinkno=%d span=%d]	Network clock is slower than board clock. Elastic store is full and deleted a frame. Check network clock quality. Board may be set to internal clocking instead of clock recovery mode.
860118	Info	EVENT	MTP-L1: Framer transmit elastic store full (TESF) alarm OFF [spmlinkno=%d span=%d]	Transmit side elastic store is in sync.
860119	Minor	EVENT	MTP-L1: Framer frame bit error (FBE) alarm ON [spmlinkno=%d span=%d]	Framing bit received in error. 1)Check line signal quality. 2)Check framing mode.
86011a	Minor	EVENT	MTP-L1: Framer frame bit error (FBE) alarm OFF [spmlinkno=%d span=%d]	No framing bit errors.
86011b	Minor	EVENT	MTP-L1: Framer severely errored framing event (SEFE) alarm ON [spmlinkno=%d span=%d]	2 out of 6 framing bits received in error. 1)Check line signal quality. 2) Check framing mode.

Table C-5: MTPL1 Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
86011c	Minor	EVENT	MTP-L1: Framer severely errored framing event (SEFE) alarm OFF [spmlinkno=%d span=%d]	No framing bit errors.
86011d	Minor	EVENT	MTP-L1: Framer 16 zeros detected (16ZD) alarm ON [spmlinkno=%d span=%d]	A string of 16 consecutive zeros received. 1)Check line signal quality. 2) Check framing mode.
86011e	Minor	EVENT	MTP-L1: Framer 16 zeros detected (16ZD) alarm OFF [spmlinkno=%d span=%d]	Stream is back to normal.
86011f	Minor	EVENT	MTP-L1: Framer 8 zeros detected (8ZD) alarm ON [spmlinkno=%d span=%d]	A string of 8 consecutive zeros received. 1) Check line signal quality. 2) Check framing mode.
860120	Minor	EVENT	MTP-L1: Framer 8 zeros detected (8ZD) alarm OFF [spmlinkno=%d span=%d]	Stream is back to normal.
860121	Info	EVENT	MTP-L1: Framer loop down code detected (LDN) alarm ON [spmlinkno=%d span=%d]	Loop down (LDN) code received.Board software releases remote loopback.
860122	Info	EVENT	MTP-L1: Framer loop down code detected (LDN) alarm OFF [spmlinkno=%d span=%d]	Loop down (LDN) code not received anymore.
860123	Info	EVENT	MTP-L1: Framer loop up code detected (LUP) alarm ON [spmlinkno=%d span=%d]	Loop up (LUP) code received.Board software sets framer remote loopback mode.
860124	Info	EVENT	MTP-L1: Framer loop up code detected (LUP) alarm OFF [spmlinkno=%d span=%d]	Loop up (LUP) code not received anymore.
860125	Info	EVENT	MTP-L1: Framer change of frame alignment (COFA) alarm ON [spmlinkno=%d span=%d]	Last resync resulted in a change of frame or multiframe alignment.
860126	Info	EVENT	MTP-L1: Framer change of frame alignment (COFA) alarm OFF [spmlinkno=%d span=%d]	Resync completed.
860127	Info	EVENT	MTP-L1: Framer transmit pulse density violation (TPDV) alarm ON [spmlinkno=%d span=%d]	Transmit data stream does not meet the ANSI T1.403 requirements for pulse density: - no more than 15 consecutive zeros - at least N ones in each and every time window of 8X(N+1) bits where N=1 through 23. Information only. Ignore if B8ZS line coding is running.
860128	Info	EVENT	MTP-L1: Framer transmit pulse density violation (TPDV) alarm OFF [spmlinkno=%d span=%d]	Transmit data stream pulse density is normal.

Table C-5: MTPL1 Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
860129	Info	EVENT	MTP-L1: Framer receive pulse density violation (RPDV) alarm ON [spmlinkno=%d span=%d]	Received data stream does not meet the ANSI T1.403 requirements for pulse density: - no more than 15 consecutive zeros - at least N ones in each and every time window of 8X(N+1) bits where N=1 through 23. Information only. Ignore if B8ZS line coding is running.
86012a	Info	EVENT	MTP-L1: Framer receive pulse density violation (RPDV) alarm OFF [spmlinkno=%d span=%d]	Receive data stream pulse density is normal.
86012b	Major	EVENT	MTP-L1: Framer receive signaling all zeros (RSA0) alarm ON [spmlinkno=%d span=%d]	Over a full MF, timeslot 16 contains all zeros. 1) Check other party is up. 2) Check line physical connections.
86012c	Major	EVENT	MTP-L1: Framer receive signaling all zeros (RSA0) alarm OFF [spmlinkno=%d span=%d]	Receive stream is back to normal.
86012d	Major	EVENT	MTP-L1: Framer receive distant MF (RDMA) alarm ON [spmlinkno=%d span=%d]	Bit 6 of timeslot 16 in frame 0 has been set for two consecutive multiframes.Wait until remote side stops sending distant MF alarm.
86012e	Major	EVENT	MTP-L1: Framer receive distant MF (RDMA) alarm OFF [spmlinkno=%d span=%d]	Reception of distant MF alarm stopped.
86012f	Major	EVENT	MTP-L1: Framer receive signaling all ones (RSA1) alarm ON [spmlinkno=%d span=%d]	Contents of timeslot 16 contains less than three zeros over 16 consecutive frames.Wait remote party stops sending all ones.
860130	Major	EVENT	MTP-L1: Framer receive signaling all ones (RSA1) alarm OFF [spmlinkno=%d span=%d]	Receive stream is back to normal.
860131	Minor	EVENT	MTP-L1: Framer CAS resync criteria met (CASRC) alarm ON [spmlinkno=%d span=%d]	Two consecutive CAS MF alignment words are received in error.Resync done by framer.
860132	Minor	EVENT	MTP-L1: Framer CAS resync criteria met (CASRC) alarm OFF [spmlinkno=%d span=%d]	Resync completed.
860133	Minor	EVENT	MTP-L1: Framer FAS resync criteria met (FASRC) alarm ON [spmlinkno=%d span=%d]	Three consecutive FAS words are received in error.Resync done by framer.
860134	Minor	EVENT	MTP-L1: Framer FAS resync criteria met (FASRC) alarm OFF [spmlinkno=%d span=%d]	Resync completed.

Table C-5: MTPL1 Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
860135	Minor	EVENT	MTP-L1: Framer CRC resync criteria met (CRCRC) alarm ON [spmlinkno=%d span=%d]	915/1000 code words are received in error.Resync done by framer.
860136	Minor	EVENT	MTP-L1: Framer CRC resync criteria met (CRCRC) alarm OFF [spmlinkno=%d span=%d]	Resync completed.

Table C-6: MTPL2 Alarm Group

Alarm No.	Severity	Type	Message	Operation
870101	Minor	EVENT	MTP-L2: T2 expired [lset=%d link=%d]	Timer "not aligned" expired.
870102	Minor	EVENT	MTP-L2: SIO received in Aligned state [lset=%d link=%d]	SIO received on an aligned link. Ignore this until it does not cause link outage.
870103	Minor	EVENT	MTP-L2: T3 expired [lset=%d link=%d]	Timer "aligned" expired.
870104	Minor	EVENT	MTP-L2: SIO received in Proving state [lset=%d link=%d]	SIO received on a link in proving period. Ignore this until it does not cause link outage.
870105	Minor	EVENT	MTP-L2: Abort Proving [lset=%d link=%d]	Proving period aborted. Ignore this until it does not cause link outage.
87010a	Minor	EVENT	MTP-L2: Alignment not possible [lset=%d link=%d]	SS7 link alignment not possible 1)Check the link # connection. 2)Reconfigure and connect the link #
87010b	Minor	EVENT	MTP-L2: SIOS received in Aligned Ready state [lset=%d link=%d]	SIOS is received in Aligned Ready State. Ignore this until it does not cause link outage.
87010c	Minor	EVENT	MTP-L2: SIO received in Aligned Ready state [lset=%d link=%d]	SIO received on an aligned link. Ignore this until it does not cause link outage.
87010d	Minor	EVENT	MTP-L2: SIOS received in Aligned Not Ready state [lset=%d link=%d]	SIOS received on a link that is aligned but not ready to carry trafficIgnore this until it does not cause link outage.
87010e	Minor	EVENT	MTP-L2: SIO received in Aligned Not Ready state [lset=%d link=%d]	SIO received on an aligned link that is still initializing. Ignore this until it does not cause link outage.
87010f	Minor	EVENT	MTP-L2: SIOS received in In Service state [lset=%d link=%d]	SIOS received on an aligned linkIgnore this until it does not cause link outage.
870110	Minor	EVENT	MTP-L2: SIO received in In Service state [lset=%d link=%d]	SIO received on an aligned link.Ignore this until it does not cause link outage.
870111	Minor	EVENT	MTP-L2: SIE received in In Service state [lset=%d link=%d]	SIE received on an aligned link. Ignore this until it does not cause link outage.
870112	Minor	EVENT	MTP-L2: SIN received in In Service state [lset=%d link=%d]	SIN received on an aligned link. Ignore this until it does not cause link outage.
870113	Minor	EVENT	MTP-L2: SIOS received in In PO state [lset=%d link=%d]	SIOS received on an alink. Ignore this until it does not cause link outage.
870114	Minor	EVENT	MTP-L2: SIO received in In PO state [lset=%d link=%d]	SIO received on a link in PO state. Ignore this until it does not cause link outage.
870115	Minor	EVENT	MTP-L2: SIE received in In PO state [lset=%d link=%d]	SIE received on a link in PO state. Ignore this until it does not cause link outage.
870116	Minor	EVENT	MTP-L2: SIN received in In PO state [lset=%d link=%d]	SIN received on a link in PO state. Ignore this until it does not cause link outage.
870117	Minor	EVENT	MTP-L2: T1 Expired [lset=%d link=%d]	Timer "alignment ready" expired.
870118	Minor	EVENT	MTP-L2: Protocol mismatch [lset=%d link=%d]	

Table C-6: MTPL2 Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
87011e	Minor	EVENT	MTP-L2: SUERM Failure [lset=%d link=%d]	MTP level2 transmission errorCorrect configuration and check cable connections if necessary.
870128	Minor	EVENT	MTP-L2: T6 Expired [lset=%d link=%d]	Timer "remote congestion" expired.
870129	Minor	EVENT	MTP-L2: T7 Expired [lset=%d link=%d]	Timer "excessive delay of acknowledgement" expired.
870132	Minor	EVENT	MTP-L2: Abnormal FIBR [lset=%d link=%d]	SS7 link alignment not possible. 1)Verify SS7 link states 2)check the local end and the remote end availability
870133	Minor	EVENT	MTP-L2: Abnormal BSNR [lset=%d link=%d]	SS7 link alignment not possible 1)Verify SS7 link states 2)check the local end and the remote end availability
87013c	Minor	EVENT	MTP-L2: Idle link fail [lset=%d link=%d]	
87013d	Minor	EVENT	MTP-L2: Pstatus timer fail [lset=%d link=%d]	
87013e	Minor	EVENT	MTP-L2: MSU discarded - L2 TX buffer overflow [lset=%d link=%d]	SS7 Message is discarded because of link transmission buffer overflow. SS7 link congestion.
87013f	Minor	EVENT	MTP-L2: MSU discarded - MSU Length [lset=%d link=%d]	SS7 message is discarded because of invalid length.
870140	Minor	EVENT	MTP-L2: Invalid MSU Length [lset=%d link=%d]	MSU value does not match length expected.
870141	Minor	EVENT	MTP-L2: L3 RX input event [lset=%d link=%d]	
870142	Minor	EVENT	MTP-L2: PRI clock loss [lset=%d link=%d]	
870143	Minor	EVENT	MTP-L2: PRI clock gain [lset=%d link=%d]	
870144	Minor	EVENT	MTP-L2: Invalid LSSU Length [lset=%d link=%d]	LSSU value does not match length expected.
870150	Critical	EVENT	MTP-L2: Board Online [spmlinkno=%d]	The SS7 board is online. Links can be created on this board, or existing links will try to align.
870151	Critical	EVENT	MTP-L2: Board Offline [spmlinkno=%d]	SS7 Controller failure or board has been taken down intentionally.
870152	Critical	EVENT	MTP-L2: Host Unavailable	

Table C-7: ETMOD Alarm Group

Alarm No.	Severity	Type	Message	Operation
880101	Major	EVENT	ETMOD: Unexpected STREAMS message [msg=0x%x]	The ETMOD module has received a STREAMS message of unexpected type on its read-side STREAMS queue and discarded it. Report problem to TAC.
880201	Major	EVENT	ETMOD: Unexpected STREAMS message [msg=0x%x]	The ETMOD module has received a STREAMS message of unexpected type on its write-side STREAMS queue and discarded it. Report problem to TAC.
880301	Info	EVENT	ETMOD: Network interface up [inetaddr=0x%x ppa=%d]	The specified LAN interface is restored to service as a result of a UNIX network interface configuration/maintenance command, e.g., ifconfig(1M). The IP address of the local host on the affected LAN—as well as its physical point of access (PPA)—is printed in the form of arguments.
880302	Critical	EVENT	ETMOD: Network interface down [inetaddr=0x%x ppa=%d]	The specified LAN interface is out of service as a result of a UNIX network interface configuration/maintenance command, i.e., ifconfig(1M). The IP address of the local host on the affected LAN—as well as its physical point of access (PPA)—is printed in the form of arguments. If dual-LAN is in use and the currently active LAN interface is removed out of service, a LAN switchover is automatically initiated. If all the LAN interfaces configured are removed out of service, a forced local system shutdown is triggered.
880401	Info	EVENT	ETMOD: Network re-connected [inetaddr=0x%x ppa=%d]	Data link layer activity has been re-detected on the specified LAN interface, i.e., after a failure to detect any activity within a three (3) second interval. The IP address of the local host on the affected LAN—as well as its physical point of access (PPA)—is printed in the form of arguments.

Table C-7: ETMOD Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
880402	Critical	EVENT	ETMOD: Network disconnected [inetaddr=0x%x ppa=%d]	No data link layer activity has been detected on the specified LAN interface for three (3) seconds. The IP address of the local host on the affected LAN—as well as its physical point of access (PPA)—is printed in the form of arguments. If dual-LAN is in use and the currently active LAN interface has no data link layer activity, a LAN switchover procedure is automatically initiated. If data link layer activity stops on all the LAN interfaces configured, a forced local system shutdown is triggered.
880501	Fatal	EVENT	ETMOD: Initiated forced shutdown on local host	The local host has been isolated from the remainder of the network, and Distributed7 system software on the local host is shutting down. Isolation may be the resulted of removing all LAN interfaces that are configured on the local host out-of-service, disconnecting them, or unplugging them while Distributed7 software is running. Note that SS7 related recovery work in such cases is performed by the surviving hosts.
880601	Info	EVENT	ETMOD: Network re-connected [inetaddr=0x%x ppa=%d]	
880602	Critical	EVENT	ETMOD: Network disconnected [inetaddr=0x%x ppa=%d]	

Table C-8: ISUP Alarm Group

Alarm No.	Severity	Type	Message	Operation
890101	Info	EVENT	ISUP: Unexpected module ID [mod=%d msg=0x%x state=0x%x]	This 'message type - sender module' combination is not expected in current state. No action required.
890102	Major	EVENT	ISUP: Unexpected message type [mod=%d msg=0x%x state=0x%x]	This message type is not expected at current module state, it may be received from Call Control or network. If message is from Call Control, make sure primitive used is valid in that state of circuit. If message is valid in that state or message is from network, log alarm sequence and call TAC.
890103	Info	EVENT	ISUP: Unexpected module state [mod=%d]	Unexpected state for module, state is not one of valid states. Internal error, log alarm sequence and call TAC
890104	Info	EVENT	ISUP: Unexpected timer expiry [mod=%d timerid=%d state=0x%x]	Module does not expect to handle expiration of this timer in current state. Internal error, log alarm sequence and call TAC.
890105	Major	EVENT	ISUP: Timer start operation failed [mod=%d timerid=%d reason=%d]	Not enough resources to start timer. Internal error, log alarm sequence and call TAC.
890106	Critical	EVENT	ISUP: DPC %x inaccessible [mod=%d msg=0x%x]	DPC in message is inaccessible. Check status of DPC in MTP and ISUP layers. If they are inconsistent, call TAC. If they both indicate that DPC is inaccessible, use MTP diagnosis methods.
890107	Minor	EVENT	ISUP: Invalid suspend/resume indicators [mod=%d msg=0x%x state=0x%x]	SUS or RES message received from network has invalid coding for CPCI. Check indicator at source of message
890108	Major	EVENT	ISUP: Range mismatch in CGBA [mod=%d range=%d/%d]	CGBA range does not match CGB range. Check CGBA range at source of message.
890109	Major	EVENT	ISUP: Range mismatch in CGUA [mod=%d range=%d/%d]	CGUA range does not match CGU range. Check CGUA range at source of message.
89010a	Minor	EVENT	ISUP: Invalid parameter format [mod=%d msg=0x%x state=0x%x]	Parameter within MSU is not legally coded. Log message and verify it is legal. If it is legal, log alarm sequence and call TAC.
89010b	Minor	EVENT	ISUP: Inconsistent range in GRA [mod=%d msg=0x%x state=0x%x]	GRA contains range value in conflict with the protocol or the database. Check GRA range at source of message.
89010c	Info	EVENT	ISUP: Invalid event indicator in CPG [mod=%d]	Event indicator in incoming CPG message contains options not supported. Adjust event indicator at source of message not to include unsupported options.

Table C-8: ISUP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
89010d	Minor	EVENT	ISUP: Dual seizure detected [mod=%d msg=%x state=%x]	A glare condition has occurred on a circuit controlled by the local exchange. No removal operation is required.
89010e	Info	EVENT	ISUP: Invalid circuit ID [mod=%d]	Circuit ID used exceeds range. Check trunk group and circuit ID in message received.
89010f	Critical	EVENT	ISUP: Unprovisioned trunk or CC not present [mod=%d trunk=%d]	Message was received from network or a configuration manager process (like MML) on a trunk to which no Call Control registration has been made. Register Call Control application with trunk group containing affected circuit.
890110	Critical	EVENT	ISUP: DPC not provisioned [mod=%d dpc=0x%x]	Message received from Call Control or network with circuit ID not associated with any DPC. Check ISUP configuration to see if DPC is configured. If configured, log alarm sequence and call TAC.
890111	Info	EVENT	ISUP: Unrecognized MSU [mod=%d msg=0x%x prim=0x%x]	Unrecognized MSU received from network. No removal operation required.
890112	Info	EVENT	ISUP: Circuit group not provisioned [mod=%d]	Circuit group ID in MSU received from network is not provisioned. Check ISUP configuration.
890113	Major	EVENT	ISUP: (De)registration failure [mod=%d trunk=%d]	Registration or de-registration request has failed. Check *errno* in returned message.
890114	Info	EVENT	ISUP: Invalid CIC [mod=%d]	Message from Call Control refers to invalid trunk-circuit combination. Check trunk group and circuit ID in received message.
890115	Critical	EVENT	ISUP: Unequipped CIC [mod=%d cic=0x%x dpc=0x%x]	Message from Call Control or network refers to unequipped circuit. If message is from Call Control, check trunk group and circuit ID in the message. Otherwise check ISUP configuration.
890116	Major	EVENT	ISUP: MSU/Primitive type mismatch [mod=%d msu=0x%x prim=0x%x]	MSU type is not expected in primitive. If message is from Call Control, determine correct primitive to use with MSU type. Otherwise, log alarm sequence and call TAC.
890117	Major	EVENT	ISUP: Unexpected primitive [mod=%d prim=0x%x]	Message translator function does not know how to handle this primitive. Check if mtype in SDU from Call Control is valid. If valid, log alarm sequence and call TAC.
890118	Info	EVENT	ISUP: Message too large [mod=%d]	Message destined to Call Control won't fit buffer. Log alarm sequence and call TAC.

Table C-8: ISUP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
890119	Info	EVENT	ISUP: Invalid MSU (ISUP -> CC) for SP variant [mod=%d]	Primitive type received from Call Control is invalid for current SP variant. Check if primitive sent from Call Control is valid for current SP variant.
89011a	Info	EVENT	ISUP: Invalid PRIM (ISUP -> CC) for SP variant [mod=%d]	Primitive type received from Call Control is invalid for current SP variant. Check if primitive sent from Call Control is valid for current SP variant.
89011b	Info	EVENT	ISUP: Resume type incompatible with previous suspend [mod=%d msg=0x%x prim=0x%x]	Suspend/Resume indicator parameter in Resume message does not match the one in Suspend message. Check Suspend/Resume indicator parameter at source of message. Source should send compatible Suspend/Resume indicator in Suspend and corresponding Resume messages.
89011c	Info	EVENT	ISUP: Invalid resume type [mod=%d msg=0x%x prim=0x%x]	Suspend/Resume indicator parameter in Resume message sent from Call Control is invalid for current suspend/resume status of circuit. Check Suspend/Resume indicator in Resume message sent from Call Control. Suspend/Resume indicator parameter must be compatible with suspend/resume status of circuit.
89011d	Info	EVENT	ISUP: Invalid trunk number [mod=%d cic=0x%x trunk=%d]	Trunk ID is invalid (out-of-range) in message received from Call Control or network. If message is from Call Control, check trunk ID in the message. Otherwise, log alarm sequence and call TAC.
89011e	Critical	EVENT	ISUP: Protocol incompatible with MTP [mod=%d protocol=%d opt=%d]	MTP and ISUP protocols are incompatible - no message processing is possible. Either modify MTP protocol to a compatible one, or delete ISUP configuration and reconfigure ISUP.
89011f	Info	EVENT	ISUP: Message received from MTP for unprovisioned SP [mod=%d]	Message received from network for an unprovisioned SP. Log alarm sequence and call TAC.
890120	Critical	EVENT	ISUP: Protocol became incompatible with MTP [mod=%d protocol=%d opt=%d]	MTP and ISUP protocols became incompatible - message processing suspended. Either modify MTP protocol to a compatible one, or delete ISUP configuration and reconfigure ISUP.
890121	Info	EVENT	ISUP: Protocol became compatible with MTP [mod=%d protocol=%d opt=%d]	MTP and ISUP protocols became compatible - message processing resumed. No action required.
890122	Info	EVENT	ISUP: Unexpected module requesting circuit state change [mod=%d sender=%d msg=0x%x]	Unexpected module requesting circuit state change.Internal error, log alarms and call TAC

Table C-8: ISUP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
890123	Info	EVENT	ISUP: Unexpected internal primitive received [mod=%d msg=0x%x]	An unexpected internal primitive.is received .Internal error, log alarms and call TAC.
890124	Info	EVENT	ISUP: Unrecognized circuit state change request [mod=%d sender=%d msg=0x%x]	Unrecognized primitive received from module requesting circuit state change. Log alarm sequence and call TAC.
890125	Info	EVENT	ISUP: T30 expired when no congestion in destination [mod=%d dpc=0x%x msg=0x%x]	T30 expired in an unexpected state. Log alarm sequence and call TAC.
890126	Info	EVENT	ISUP: Expired timer ID invalid [mod=%d timerid=%d dpc=0x%x]	Timer expired is invalid for ISUP. Log alarm sequence and call TAC.
890127	Info	EVENT	ISUP: Destination status message from MTP for unprovisioned DPC [mod=%d dpc=0x%x msg=0x%x]	Check MTP and ISUP configuration, make sure DPC is configured in both. If configuration is OK, log alarms and call TAC.. Log alarm sequence and call TAC.
890128	Info	EVENT	ISUP: Call attempted on busy circuit [mod=%d cctnum=%d]	Check the circuit's state in ISUP and Call Control, then find the reason for the inconsistency Log alarm sequence and call TAC.
890129	Info	EVENT	ISUP: Switching equipment congestion [mod=%d dpc=0x%x msg=0x%x]	No removal operation. This is normal under a heavy load.
89012a	Info	EVENT	ISUP: ISUP variant incompatible with MTP [mod=%d]	Reconfigure ISUP and MTP using compatible variants.
890140	Info	EVENT	ISUP: Unexpected timer expiration [mod=%d timerid=%d tcic=0x%x]	Log alarm sequence and call TAC.
890150	Info	EVENT	ISUP: Circuit data has inconsistency [mod=%d internal=0x%x]	ISUP circuit data corrupted. Log alarm sequence and call TAC.
890151	Info	EVENT	ISUP: Unrecognized parameter value [mod=%d pmtid=%d cctid=%d]	If originator of the message, then resend the message with the correct parameter contents.
890152	Info	EVENT	ISUP: Status bit mismatch [mod=%d cctid=%d grpid=%d]	The status bits in the received CGUA message does not much with the CGU sent.Check the status bits of the CGU and CGUA messages.
890153	Info	EVENT	ISUP: Invalid first CIC in multirate call [mod=%d cctid=%d msg=0x%x]	If received at the outgoing side, resend the IAM on the correct controlling CIC.Log alarm sequence and call TAC.
890154	Info	EVENT	ISUP: Mismatch between transmission medium requirement and CAM parameter [mod=%d cctid=%d msg=0x%x]	If received at the outgoing side, resend the IAM with the correct values of Transmission Medium Requirement parameter and CAM parameter.Log alarm sequence and call TAC.
890155	Info	EVENT	ISUP: Mismatch between first CIC and first set bit of CAM parameter [mod=%d cctid=%d msg=0x%x]	If received at the outgoing side, resend the IAM on the correct controlling CIC and coding correct CAM parameter.

Table C-8: ISUP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
890156	Info	EVENT	ISUP: GRA received without CAM parameter [mod=%d cctid=%d grpid=%d]	When GRS is sent with CAM parameter, the GRA received in response to GRS has no CAM parameter.No removal operation is required
890157	Info	EVENT	ISUP: GRA received with CAM parameter [mod=%d cctid=%d grpid=%d]	When GRS is sent without CAM parameter, the GRA received in response to GRS has a CAM parameter.No removal operation is required
890158	Info	EVENT	ISUP: Unrecognized parameter [mod=%d pmtid=%d cctid=%d]	If originator of the message, then resend the message with the correct parameter.
890159	Info	EVENT	ISUP: Multirate call spread across two trunk groups [mod=%d fcic=%d cctcnt=%d]	When a single multirate call is attempted on circuits covering two contiguous groups.If received at the outgoing side, re-send the IAM with a valid CIC.
89015a	Info	EVENT	ISUP: Mismatch between CAM in GRS and CAM in GRA [mod=%d cctid=%d grpid=0x%x]	When the GRA received has a different CAM parameter content from that of the GRS sent.No removal operation is required
89015b	Info	EVENT	ISUP: Non optional parameter in unknown message [mod=%d msu=0x%x prim=0x%x]	Send the unknown message with optional parameters only
89015c	Info	EVENT	ISUP: No message compatibility parameter in unknown message [mod=%d msu=0x%x prim=0x%x]	Send the unknown message with message compatibility information parameter.
8901c8	Critical	EVENT	ISUP: Cannot lock CCT record [mod=%d grp=%d cct=%d]	An attempt to lock indicated circuit data failed in distributed mode. Log alarm sequence and call TAC.
8901c9	Critical	EVENT	ISUP: Cannot unlock CCT record [mod=%d grp=%d cct=%d]	An attempt to unlock indicated circuit data failed in distributed mode. Log alarm sequence and call TAC.
8901ca	Critical	EVENT	ISUP: Cannot lock TRUNK record [mod=%d dpc=0x%x trunk=%d]	An attempt to lock indicated trunk data failed in distributed mode. Log alarm sequence and call TAC.
8901cb	Critical	EVENT	ISUP: Cannot unlock TRUNK record [mod=%d dpc=0x%x trunk=%d]	An attempt to unlock indicated trunk data failed in distributed mode. Log alarm sequence and call TAC.
8901cc	Critical	EVENT	ISUP: Cannot lock SP record [mod=%d reason=%d]	An attempt to lock signaling point data failed in distributed mode. Log alarm sequence and call TAC.
8901cd	Critical	EVENT	ISUP: Cannot unlock SP record [mod=%d reason=%d]	An attempt to unlock signaling point data failed in distributed mode. Log alarm sequence and call TAC.
8901ce	Critical	EVENT	ISUP: Cannot lock NODE record [mod=%d dpc=0x%x]	An attempt to lock indicated destination point data failed in distributed mode. Log alarm sequence and call TAC.

Table C-8: ISUP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
8901cf	Critical	EVENT	ISUP: Cannot unlock NODE record [mod=%d dpc=0x%x]	An attempt to unlock indicated destination point data failed in distributed mode. Log alarm sequence and call TAC.
8901d0	Critical	EVENT	ISUP: Cannot lock CCTGRP record [mod=%d dpc=0x%x group=%d]	An attempt to lock indicated circuit group data failed in distributed mode. Log alarm sequence and call TAC.
8901d1	Critical	EVENT	ISUP: Cannot unlock CCTGRP record [mod=%d dpc=0x%x group=%d]	An attempt to unlock indicated circuit group data failed in distributed mode. Log alarm sequence and call TAC.
8901d2	Critical	EVENT	ISUP: SP restart in progress - Messages discarded [mod=%d prim=%x]	MTP signaling point restart is in progress therefore the message sent has been discarded. Stop sending messages to ISUP.
8901d3	Critical	EVENT	ISUP: Abnormal termination via UNIX signal [mod=%d signum=%d]	Isupd terminated with given signal. It must be started again. Log alarm sequence and call TAC.

Table C-9: APM Alarm Group

Alarm No.	Severity	Type	Message	Operation
8b0101	Info	EVENT	APM: Starting at run state %s	Indicates default start-up state for apmd daemon.
8b0102	Info	EVENT	APM: Request via %s to change run state	Indicates that a manual request is placed, using apm_setstate command-line utility, to change the current run state of apmd daemon.
8b0103	Info	EVENT	APM: Request via %s to re-execute process table	Indicates that a manual request is placed, using apm_update command-line utility, to re-read/execute the apmd configuration file.
8b0104	Info	EVENT	APM: Request via %s to stop system software	Indicates that a manual request is placed, using apm_stop command-line utility, to terminate all processes running on the local host.
8b0105	Info	EVENT	APM: Request via %s to kill all non-failsafe processes	Indicates that a manual request is placed, using apm_killall command-line utility, to terminate all processes [that are not marked as "failsafe" in the apmd configuration file] on the local host.
8b0106	Info	EVENT	APM: Request via %s to adopt process %s	Indicates that apmd daemon on a remote host requested adoption of a child process to be spawned/executed on the local host.
8b0107	Info	EVENT	APM: Change in run state [%s ---> %s]	Indicates a change in the run state of apmd daemon. Each such change results in apmd configuration file contents to be re-read/executed by the apmd dameon.
8b0108	Minor	EVENT	APM: Request denied - In critical window	Indicates that a manual request, initiated using an apm command-line utility, is not processed by the apmd daemon and discarded. This occurs when apmd is in start-up and/or transitional stage.
8b0109	Major	EVENT	APM: Cannot allocate process table entry [tag=%s]	Indicates that the apmd daemon is unable to allocate internal resources necessary to store information about a newly spawned child process. Stop and re-start the corresponding child process. If the problem persists, stop/re-start all processes on the local host.
8b0201	Info	EVENT	APM: Request via %s to change run state	Indicates that an application request is placed, using apm_setstate() function call, to change the current run state of apmd daemon.

Table C-9: APM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
8b0301	Major	EVENT	APM: Access to file denied [path=%s]	Indicates that a request is made to spawn/execute a new child process but the path name associated with the process is not accessible to apmd. Check to make sure that the path name corresponds to an existing UNIX file name and the access privileges of the file are set correctly.
8b0302	Major	EVENT	APM: Cannot fork process [path=%s]	Indicates that a request is made to spawn/execute a new child process but the fork() system call invoked by apmd is failed. Consult with master log file on the local host to determine the exact cause of the fork() system call before placing a new request.
8b0303	Major	EVENT	APM: Cannot exec process [path=%s]	Indicates that a request is made to spawn/execute a new child process but the exec() system call invoked by apmd is failed. Consult with master log file on the local host to determine the exact cause of the exec() system call before placing a new request.
8b0304	Minor	EVENT	APM: Unknown process exited [pid=%d]	Indicates that apmd detected termination of a child process spawned by itself but is unable to locate a corresponding entry in its process table. No action is necessary.
8b0305	Major	EVENT	APM: Process respawning too frequently [tag=%s]	Indicates that apmd is making repetitive attempts to spawn/execute the specified process but the process is terminating its execution prematurely upon start-up. For system processes that are part of Distributed product, consult with master log file on the local host to determine the exact cause of premature process termination (e.g., missing configuration information, corrupted database, unexpected signal) and alleviate it.
8b0306	Major	EVENT	APM: Process failed [tag=%s]	Indicates that a child process spawned by the apmd daemon terminated its execution with a non-zero exit code [signifying failure]. Consult with master log file on the local host to determine the potential cause of failure. Under normal circumstances, child processes are expected to terminate their execution with a zero exit code [signifying voluntary termination].
8b0307	Major	EVENT	APM: Process killed via UNIX signal [tag=%s signum=%d]	Indicates that a child process spawned by the apmd daemon terminated its execution due to an unexpected signal (i.e., a UNIX signal that the process is not prepared to handle).

Table C-9: APM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
8b0308	Info	EVENT	APM: Process stopped via UNIX signal [tag=%s signum=%d]	Indicates that the execution of a child process spawned by the apmd daemon is halted via SIGSTOP. To resume process execution, a SIGCONT signal needs to be delivered to the process.
8b0309	Info	EVENT	APM: Process continued execution [tag=%s]	Indicates that a SIGCONT signal is delivered to a child process whose execution has been halted earlier due to delivery of a SIGSTOP signal.
8b030a	Critical	EVENT	APM: No heartbeat response received from process [tag=%s]	Indicates that the apmd daemon is unable to exchange periodic heartbeat messages with a child process that it spawned earlier. This alarm gets generated after apmd fails to receive any replies to three successive "heartbeat request" messages in a row. Under such circumstances, apmd daemon will assume that the child process is stuck in a non-recoverable way and deliver a SIGKILL signal to it in an attempt to forcefully terminate it.

Table C-10: OMAP Alarm Group

Alarm No.	Severity	Type	Message	Operation
8d0101	Critical	EVENT	OMAP: Cannot access MTP-L2 30-minute log file [sp=%d]	Check to see if $EBSHOME/access/RUN[0-7]/omaplog directory exists for the signaling point specified.
8d0102	Critical	EVENT	OMAP: Cannot save MTP-L2 30-minute log record [sp=%d]	Unable to write to the MTP-l2 30 minute log file, possibly due to incorrect permissions. Check to see if the log file specified has write permissions
8d0103	Minor	EVENT	OMAP: Out-of-sequence MTP-L2 30-minute report [sp=%d seqno=%d]	The sequence number contained in a 30 minute report request from MTP-L2 is found to be incorrect. AccessOMAP simply discards the request.
8d0104	Critical	EVENT	OMAP: Cannot access MTP-L3 30-minute log file [sp=%d]	Check to see if $EBSHOME/access/RUN[0-7]/omaplog directory exists for the signaling point specified.
8d0105	Critical	EVENT	OMAP: Cannot save MTP-L3 30-minute log record [sp=%d]	Unable to write to the MTP-L3 30 minute log file, possible due to incorrect permissions. Check to see if the log file specified has write permissions.
8d0106	Minor	EVENT	OMAP: Out-of-sequence MTP-L3 30-minute report [sp=%d seqno=%d]	The sequence number contained in a 30 minute report request from MTP-L3 is found to be incorrect. AccessOMAP simply discards the request.
8d0107	Critical	EVENT	OMAP: Cannot access MTP-L3 5-minute log file [sp=%d]	Check to see if $EBSHOME/access/RUN[0-7]/omaplog directory exists for the signaling point specified.
8d0108	Critical	EVENT	OMAP: Cannot save MTP-L3 5-minute log record [sp=%d]	Unable to write to the MTP-L3 5 minute log file, possibly due to incorrect permissions. Check to see if the log file specified has write permissions.
8d0109	Minor	EVENT	OMAP: Out-of-sequence MTP-L3 5-minute report [sp=%d seqno=%d]	The sequence number contained in a 5 minute report request from MTP-L3 is found to be incorrect. AccessOMAP simply discards the request.
8d010a	Critical	EVENT	OMAP: Cannot access SCCP 30-minute log file [sp=%d]	Check to see if $EBSHOME/access/RUN[0-7]/omaplog directory exists for the signaling point specified.
8d010b	Critical	EVENT	OMAP: Cannot save SCCP 30-minute log record [sp=%d]	Unable to write to the SCCP 30 minute log file, possibly due to incorrect permissions. Check to see if the log file specified has write permissions.
8d010c	Minor	EVENT	OMAP: Out-of-sequence SCCP 30-minute report [sp=%d seqno=%d]	The sequence number contained in a 30 minute report request from SCCP is found to be incorrect. AccessOMAP simply discards the request.

Table C-10: OMAP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
8d010d	Critical	EVENT	OMAP: Cannot save SCCP 30-minute SP spurious response [sp=%d]	There is no room left in the SCCP spurious response table maintained by AccessOMAP to store SP related measurements info. AccessOMAP simply discards the measurement report received.
8d010e	Critical	EVENT	OMAP: Cannot save SCCP 30-minute SSN spurious response [sp=%d]	There is no room left in the SCCP spurious response table maintained by AccessOMAP to store SSN related measurements infol AccessOMAP simply discards the measurement report received.
8d010f	Critical	EVENT	OMAP: Cannot save SCCP 30-minute unallocated SP spurious response [sp=%d]	There is no room left in the SCCP spurious response table maintained by AccessOMAP to store un-allocated SP related measurements info. AccessOMAP simply discards the measurement report received.
8d0110	Critical	EVENT	OMAP: Cannot save SCCP 30-minute unallocated SSN spurious response [sp=%d]	There is no room left in the SCCP spurious response table maintained by AccessOMAP to store un-allocated SSN related measurements info. AccessOMAP simply discards the measurement report received.
8d0111	Critical	EVENT	OMAP: Cannot access TCAP 30-minute log file [sp=%d]	Check to see if $EBSHOME/access/RUN[0-7]/omaplog directory exists for the signaling point specified. If an SP value of 255 is specified, check to see if $EBSHOME/access/RUN/omaplog directory exists instead.
8d0112	Critical	EVENT	OMAP: Cannot save TCAP 30-minute log record [sp=%d]	Unable to write to the TCAP 30 minute log file, possibly due to incorrect permissions. Check to see if the log file specified hs write permissions.
8d0113	Minor	EVENT	OMAP: Out-of-sequence TCAP 30-minute report [sp=%d seqno=%d]	The sequence number contained in a 30 minute report request from TCAP is found to be incorrect. AccessOMAP will simply discard the request.
8d0114	Critical	EVENT	OMAP: Cannot save TCAP 30-minute SSN spurious response [sp=%d]	There is no room left in the TCAP spurious response table maintained by AccessOMAP to store SSN related measurements info. AccessOMAP will simply discard the measurement report received.

Table C-11: NIMOD Alarm Group

Alarm No.	Severity	Type	Message	Operation
920101	Major	EVENT	NIMOD: Unexpected TCP/IP message [msg=0x%x]	NIMOD has received a STREAMS M_PROTO message of unknown type from the TCP/IP module (message type is not T_DATA_IND, T_ORDREL_IND, or T_DISCON_IND). If and when another M_PROTO message is received, it is discarded. Report alarm and message type to TAC.
920102	Major	EVENT	NIMOD: Unexpected STREAMS message [msg=0x%x]	NIMOD module has received a STREAMS message of unexpected type on its read-side STREAMS queue and discarded it. Report to TAC.
920201	Critical	EVENT	NIMOD: Failed to report TCP/IP event [type=0x%x]	NIMOD module was in the process of relaying a T_ORDREL_IND or T_DISCON_IND message from the TCP/IP network to the netd daemon on the local host, but was unable to locate the *netd* daemon process. Look into log file of the same date under the *$EBSHOME/access/RUN/mlog* directory to find out the netd problem. The apmd daemon has been pre-programmed to keep the netd daemon alive at all times. Report findings to TAC.
920301	Critical	EVENT	NIMOD: Invalid message tag encountered in TCP/IP message [tag=0x%x]	NIMOD has received a T_DATA_IND message from the TCP/IP module but the message does not contain the proper message header, most likely because it has been corrupted. The message is discarded by NIMOD. Since each ServiceController 1510 host within a distributed environment has been pre-programmed to encode/decode exchanged T_DATA_IND messages in a particular way, this alarm message is likely to indicate a problem with the physical network.
920302	Critical	EVENT	NIMOD: Failed to copy data from message [nbytes=%d offset=%d]	NIMOD failed to retrieve information contained within a message received across a TCP/IP connection due to a failure in the message copying procedure. As a result, inter-host messages is lost. Check your LAN hardware first. Report problem to TAC if it persists.

Table C-11: NIMOD Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
920303	Critical	EVENT	NIMOD: Failed to adjust message [size=%d nbytes=%d]	NIMOD failed to retrieve information contained within a message received across a TCP/IP connection due to a failure in the message size adjustment procedure. As a result, inter-host messages is lost. Check your LAN hardware first. Report problem to TAC if it persists.
920401	Critical	EVENT	NIMOD: Unexpected STREAMS message [msg=0x%x]	NIMOD module has received a STREAMS message of unexpected type on its write-side STREAMS queue and discarded it. Report problem to TAC.

Table C-12: TCAP Alarm Group

Alarm No.	Severity	Type	Message	Operation
950101	Critical	EVENT	TCAP: Cannot relay close indicator upstream	The tcmd daemon is not running. This process is not started by the apmd daemon in the default case and must be started manually.
950201	Critical	EVENT	TCAP: Error/Hangup condition on STREAMS queue [q=0x%x]	An M_ERROR or M_HANGUP STREAMS message has been received by the TCAP module, indicating a STREAMS error or hang-up condition. Stop and restart the associated TC application. If the problem is experienced by all TC applications on the local host, stop and restart the tcmd daemon.
950202	Critical	EVENT	TCAP: Invalid instance number [dev=%d inst=%d]	The instance number associated with a TCAP message received is outside the range [0, 7]. The message is discarded. Report the condition to TAC.
950203	Major	EVENT	TCAP: Transaction ID not retrievable [dev=%d msg=0x%x]	The transaction portion of a received TCAP message contains incorrect information. Message is discarded. Report problem to TAC.
950204	Major	EVENT	TCAP: Unknown SS7 message [dev=%d msg=0x%x]	TCAP module has received a message of unexpected M_DATA/M_PROTO message and discarded it. Report problem to TAC.
950205	Major	EVENT	TCAP: Unknown STREAMS message [dev=%d msg=0x%x]	TCAP module has received a STREAMS message of unexpected type and discarded it. Report problem to TAC.
950206	Minor	EVENT	TCAP: Transaction ID not in use [trid=0x%x msg=0x%x]	The transaction ID included in the TCAP message received cannot be associated with an active dialogue, either because the dialogue has already been terminated, or because the transaction ID field is incorrectly populated.
950207	Minor	EVENT	TCAP: Received Abort Cause is out-of-range [trid=0x%x msg=0x%x]	The received message contains an Abort Cause that is invalid. Valid values for ITU protocols are 0 - 4; ETSI protocols are 0 - 127.
950301	Critical	EVENT	TCAP: Invalid major device number [dev=%d msg=0x%x]	The major device number associated with an incoming TCAP message of specified type is found to be incorrect, i.e., outside the permissible [0, 15] range, possibly because message received was of unexpected type. The message is discarded by the system. Report condition to TAC.
950302	Critical	EVENT	TCAP: Invalid minor device number [dev=%d msg=0x%x]	The minor device number associated with an incoming TCAP message of specified type is found to be incorrect, i.e., outside the permissible [0, 7] range, possibly because message received was of unexpected type. The message is discarded by the system. Report condition to TAC.

Table C-12: TCAP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
950303	Minor	EVENT	TCAP: Null STREAMS queue - Message discarded	TCAP module was in the process of routing a TCAP message to a particular instance of a TC application; however, the read-side STREAMS queue associated with that application is found to be invalid, most likely because the instance was in the process of termination. The message will simply be discarded.
950304	Critical	EVENT	TCAP: STREAMS queue full [q=0x%x]	The read-side STREAMS queue associated with a particular instance of a TC application is found to contain more messages than its configuration allows. To prevent further accumulation on the queue, the message received is discarded and a measurement count to that effect is pegged. If this is not affordable, the TC application should use the spm_qparams() series of library calls to increase the queue limits.
950401	Minor	EVENT	TCAP: Null STREAMS queue - Message discarded	TCAP module found the next STREAMS queue in the upstream direction is invalid. The application may have terminated or the system is being shut down. The message is discarded. No action necessary.
950501	Critical	EVENT	TCAP: Null TCAP bottom STREAMS queue [dev=%d]	The specified instance of the TCAP module is experiencing trouble (e.g. it is not linked or a STREAMS error condition exists). Messages submitted by the TC application is discarded. Terminate all instances of the TC application and restart them. The module will unlink and then be re-linked.
950502	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ QWP state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received Query With Permission (QWP) message is incorrect. The message is discarded and a measurement count is kept.
950503	Major	SET_ALARM	TCAP: Unexpected QWP state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a QWP message is different from idle. The message is discarded and a measurement count is kept.
950504	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ QWOP state [dev=%d dlgid=%d]	The ownership information associated with a transaction ID specified within a Query Without Permission (QWOP) message is incorrect. The message is discarded and a measurement count is kept.
950505	Major	SET_ALARM	TCAP: Unexpected QWOP state [dev=%d dlgid=%d]	The state associated with a the transaction ID specified in a QWOP message is different from idle. The message is discarded and a measurement count is kept.

Table C-12: TCAP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
950506	Major	SET_ALARM	TCAP: Unexpected UNI state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a unidirectional TCAP message is different from idle. The message is discarded and a measurement count is tracked.
950507	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ CWP state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a Continue With Permission (CWP) message received is incorrect. The message is discarded and a measurement count is tracked.
950508	Major	SET_ALARM	TCAP: Unexpected CWP state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a CWP message is incorrect. The message is discarded and a measurement count is tracked.
950509	Major	SET_ALARM	TCAP: Unexpected CWOP state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a Continue Without Permission (CWOP) message is incorrect. The message is discarded and a measurement count is tracked.
95050a	Major	SET_ALARM	TCAP: Unexpected RESPONSE state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a response message is incorrect. The message is discarded and a measurement count is tracked.
95050b	Major	SET_ALARM	TCAP: Unexpected ABORT state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in an abort message is incorrect. The message is discarded and a measurement count is tracked.
95050c	Major	SET_ALARM	TCAP: Unexpected dialogue type [dev=%d msg=0x%x]	The TCAP message received contains an invalid package type. The message is discarded. Report condition to TAC.
95050d	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ BEGIN state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received BEGIN message is incorrect. The message is discarded and a measurement count is tracked.
95050e	Major	SET_ALARM	TCAP: Unexpected BEGIN state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a BEGIN message is different from idle. The message is discarded and a measurement count is tracked.
95050f	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ CONT state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received CONTINUE message is incorrect. The message is discarded and a measurement count is tracked.
950510	Major	SET_ALARM	TCAP: Unexpected CONT state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a CONTINUE message is incorrect. The message is discarded and a measurement count is tracked.
950511	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ END state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received END message is incorrect. The message is discarded and a measurement count is tracked.

Table C-12: TCAP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
950512	Major	SET_ALARM	TCAP: Unexpected END state [dev=%d dlgid=%d]	The state associated with the transaction ID specified in a CONTINUE message is incorrect. The message is discarded and a measurement count is tracked.
950513	Minor	SET_ALARM	TCAP: Unexpected dialogue ID @ ABORT state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received ABORT message is incorrect. The message is discarded and a measurement count is tracked.
950514	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ RESP state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received RESP message is incorrect. The message is discarded and a measurement count is tracked.
950515	Major	SET_ALARM	TCAP: Unexpected dialogue ID @ CWOP state [dev=%d dlgid=%d]	The ownership information associated with the transaction ID specified within a received CWOP message is incorrect. The message is discarded and a measurement count is tracked.
950601	Critical	EVENT	TCAP: Null TCAP bottom STREAMS queue [dev=%d]	TCAP module was about to send a TCAP message from an application downstream but the module is not linked properly or there is a STREAMS error condition. Terminate all instances of the specified TC application and restart them so the module is unlinked and re-linked.
950701	Critical	EVENT	TCAP: Null TCAP bottom STREAMS queue [dev=%d]	TCAP module was about to send an L_IPCMSG message from an application downstream, but the module is not linked properly or there is a STREAMS error condition. Terminate all instances of the specified TC application and restart them so the module is unlinked and re-linked.
950702	Critical	EVENT	TCAP: STREAMS queue full [q=0x%x]	TCAP module was routing a message upstream but the next STREAMS queue in the upstream direction is invalid. The message is discarded. Most likely the application terminated or the system is being shut down. No action necessary.
950801	Critical	EVENT	TCAP: Cannot signup for DKM event notification [dev=%d reason=%d]	TCAP module failed to register for DKM event notification and will not be able to detect DKM events. Recovery of TCAP transactions cannot be guaranteed if and when there is a need. Terminate all instances of the specified TC application and restart them.
950802	Critical	EVENT	TCAP: Cannot create and/or access DKM segment [key=0x%x reason=%d]	TCAP module failed to create or to access the kernel-resident tables used for storing transaction-related information, most likely due to a failure in the DKM subsystem. If the alarm condition is generated because a TCAP application started on a remote host, this alarm indicates that transaction recovery will not be possible if and when there is a need.

Table C-12: TCAP Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
950803	Critical	EVENT	TCAP: Cannot destroy DKM segment [dkmid=%d reason=%d]	TCAP module failed to destroy the local copy of the kernel-resident table used for storing transaction-related information. The kernel-space memory associated with the table will not be de-allocated. No action is necessary.
950804	Critical	EVENT	TCAP: Cannot acquire DKM lock [dkmid=%d reason=%d]	TCAP module failed to lock the DKM segment used for storing transaction-related information. Recovery of the transactions will not be possible if and when there is a need. The associated message will simply be discarded and the transaction record will not be updated.
950805	Critical	EVENT	TCAP: Cannot release DKM lock [dkmid=%d reason=%d]	TCAP module failed to unlock the DKM segment used for storing transaction-related information. Recovery of the associated transaction will not be possible if and when there is a need.
950806	Critical	EVENT	TCAP: Cannot sync DKM segment contents [dkmid=%d reason=%d]	TCAP module failed to propagate the changes made in the transaction table to remote hosts during a recovery attempt, most likely due to a failure in the DKM subsystem. This may result in a failure in the transaction recovery attempts from other hosts.

Table C-13: TCMOD Alarm Group

Alarm No.	Severity	Type	Message	Operation
960101	Major	EVENT	TCMOD: Unexpected TCP/IP message [msg=0x%x]	A STREAMS M_PROTO message of unknown type has been received from the TCP/IP module (message type is not T_DATA_IND, T_ORDREL_IND, or T_DISCON_IND). If and when another M_PROTO message is received, it is discarded. Report alarm and message type to TAC.
960102	Critical	EVENT	TCMOD: Failed to report TCP/IP event [type=0x%x]	During the process of relaying a T_ORDREL_IND or T_DISCON_IND message from the TCP/IP network to the netd daemon on the local host, the *netd* daemon process could not be located. Look into log file of the same date under the *$EBSHOME/access/RUN/mlog* directory to find out the netd problem. The apmd daemon has been pre-programmed to keep the netd daemon alive at all times. Report findings to TAC.

Table C-13: TCMOD Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
960201	Critical	EVENT	TCMOD: Unexpectedly short message [size=%d]	Message received from the TCP/IP module was unexpectedly short. The message is discarded. TCAP messages over TCP/IP must have a header portion that is 22 bytes. The message was probably smaller than this size. Make sure the sender is encoding according to TCAP over TCP/IP specs.
960202	Critical	EVENT	TCMOD: Invalid address indicator [ind=0x%x]	Address indicator field of the received TCAP message contains an invalid value (something other than 0x01). Message is discarded. Problem at sender.
960203	Critical	EVENT	TCMOD: Invalid calling and/or called address length [size=%d]	Address length field of the received TCAP message contains an invalid value (other than 7). The message is discarded. Problem at sender.
960204	Critical	EVENT	TCMOD: Invalid IP address [inetaddr=0x%x]	Called party IP address of the received TCAP message contains a value different from the local host's IP address. Message is discarded. Problem at sender.
960205	Critical	EVENT	TCMOD: Invalid port number [port=%d]	Called party port number information within the received TCAP message contains a value outside the valid range [0, 255]. Message is discarded. Problem at sender.
960206	Major	EVENT	TCMOD: Unexpected message [msg=0x%x]	TCMOD module received an unexpected STREAMS M_DATA message (message type was not L_TC_DATA_MSG, L_TC_HBREQ_MSG, or L_TC_HBRSP_MSG). Message is discarded. Call TAC.

Table C-14: DKM Alarm Group

Alarm No.	Severity	Type	Message	Operation
970101	Major	EVENT	DKM: Unexpected STREAMS message [msg=0x%x]	DKM module has received a STREAMS message of unexpected type and discarded it. Report problem to TAC.
970201	Major	EVENT	DKM: Unexpected STREAMS message [msg=0x%x]	DKM module has received a STREAMS message of unexpected type and discarded it. Report problem to TAC.
970202	Critical	EVENT	DKM: Error/Hangup condition on STREAMS queue [q=0x%x]	An M_ERROR or M_HANGUP STREAMS message has been received by the DKM module. To restore the DKM module into operational state, stop and restart the *dkmd* daemon on the local host.
970301	Major	EVENT	DKM: Cannot trigger event [type=0x%x]	DKM module failed to trigger the specified event due to a failure in allocating the necessary STREAMS resources.

Table C-14: DKM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
970302	Major	EVENT	DKM: Cannot relay shutdown indicator upstream	DKM module failed to notify the *dkmd* daemon of a software shutdown condition. No action is necessary; the *apmd* daemon will send a SIG-TERM signal to it.
970303	Critical	EVENT	DKM: Cannot issue control command [cmd=0x%x]	DKM module failed to notify the *dkmd* daemon of a service related request. For a host ID related service request, this is likely to cause a discrepancy between the individual hosts comprising a network re: the logical host ID's associated with each host; therefore, it needs to be corrected immediately by stopping and restarting the Distributed7 software on the host at which this alarm condition has occurred.
970304	Critical	EVENT	DKM: Cannot relay request to global instance [prim=%d/%d]	DKM module failed to relay a DKM-related request from a local user to the global instance of the DKM module. The originator will receive an error code. No action is necessary.
970305	Critical	EVENT	DKM: Cannot reply to peer request [prim=%d/%d]	DKM module failed to reply to a DKM-related request from one of its peers. The originator will receive an error code. No action is necessary.
970306	Critical	EVENT	DKM: Cannot broadcast to peers	DKM module failed to convey a DKM-related request to its peers. The originator will receive an error. No action is necessary.
970307	Critical	EVENT	DKM: Cannot enqueue request received from peer	DKM module failed to allocate the necessary STREAMS resources to enqueue a DKM lock-related request initiated by a remote DKM user.
970308	Critical	EVENT	DKM: Cannot propagate data changes [dkmid=%d]	DKM module failed to propagate data changes for a particular DKM segment to its peers. This will cause inconsistencies in copies of the DKM segment. The entire system may need to be shut down and restarted to correct this problem.
970401	Critical	EVENT	DKM: Cannot create user record	DKM module failed to allocate the necessary STREAMS resources to service a DKM event registration request by a DKM user. The originator will receive an error. No action is necessary.
970402	Critical	EVENT	DKM: Cannot create service record	DKM module failed to allocate the necessary STREAMS resources to service a DKM-related request from a DKM user. The originator will receive an error. No action is necessary.
970403	Critical	EVENT	DKM: Cannot create main segment record	DKM module failed to allocate the necessary STREAMS resources to service a request to allocate a DKM segment. The originator will receive an error. No action is necessary.
970404	Critical	EVENT	DKM: Cannot create extension segment record	DKM module failed to allocate the necessary STREAMS resources to service a request to extend an existing DKM segment. The originator will receive an error. No action is necessary.

Table C-14: DKM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
970405	Critical	EVENT	DKM: Cannot create lock record	DKM module failed to allocate the necessary STREAMS resources to service a request to acquire a DKM lock. The originator will receive an error. No action is necessary.
970501	Critical	EVENT	DKM: Cannot allocate STREAMS message block	DKM module failed to allocate the STREAMS resources necessary to issue a request to retrieve information about its peers.
970502	Critical	EVENT	DKM: Cannot retrieve list of peers	DKM module failed to retrieve information about its peers. The identity of the global DKM instance will not be known by the system. DKM requests initiated while this situation exists may not be processed properly.
970601	Critical	EVENT	DKM: Cannot locate main segment record [dkmid=%d prim=%d]	DKM module failed to locate the DKM segment associated with a particular segment identifier. The originator will receive an error. No action is necessary.
970602	Critical	EVENT	DKM: Cannot locate extension segment record [dkmid=%d extid=%d]	DKM module failed to locate the DKM segment associated with a particular segment and extension identifier. The originator will receive an error. No action is necessary.
970603	Critical	EVENT	DKM: Cannot issue maintenance request [type=%d]	DKM module failed to relay a DKM maintenance-related request to one of its peers. This will cause differences in the copies of DKM-related data maintained by the local DKM module. The *dkmd* daemon on the local host may need to be terminated and restarted to correct this situation.
970604	Critical	EVENT	DKM: Cannot broadcast maintenance request [dkmid=%d extid=%d]	DKM module failed to convey a DKM maintenance-related request to its peers. This will cause differences in the copies of DKM-related data maintained by the local DKM module.
970605	Critical	EVENT	DKM: Cannot reply to maintenance request [dkmid=%d type=%d]	DKM module failed to reply to a DKM maintenance initiated by one of its peers. The originator must re-send the request. No action is necessary.
970606	Major	EVENT	DKM: Cannot sync segment contents [dkmid=%d]	DKM module failed to update the contents of the specified DKM segment per sync messages received from one of its peers. This will cause inconsistencies in the contents of copies of the segment.
970607	Major	EVENT	DKM: Cannot sync extension segment contents [dkmid=%d extid=%d]	DKM module failed to update the contents of the specified DKM segment extension per sync messages received from one of its peers. This will cause inconsistencies in the contents of copies of the segment extension.
970701	Critical	EVENT	DKM: Cannot align STREAMS message - Message discarded	DKM module failed to align, i.e., pull-up, the STREAMS message properly. The message is discarded.

Table C-14: DKM Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
970702	Critical	EVENT	DKM: Cannot locate main segment record [dkmid=%d prim=%d]	DKM module failed to locate the DKM segment associated with a particular segment identifier. The segment has probably already been destroyed. The originator will receive an error. No action is necessary.
970801	Major	EVENT	DKM: Cannot service pending sync request	DKM module failed to service a pending sync request from a DKM user due to inconsistencies within the service record associated with the request. The request is ignored.
970802	Major	EVENT	DKM: Cannot service pending lock request	DKM module failed to service a pending lock request from a DKM user due to inconsistencies within the service record associated with the request. The request is ignored.
970803	Major	EVENT	DKM: Cannot service blocked lock request	DKM module failed to service a blocked lock request from a DKM user due to inconsistencies within the service record associated with the request. The request is ignored.
970804	Major	EVENT	DKM: Cannot service sync request [dkmid=%d extid=%d]	
970805	Major	EVENT	DKM: Cannot service lock request [dkmid=%d extid=%d]	

Table C-15: ISUPMOD Alarm Group

Alarm No.	Severity	Type	Message	Operation
990101	Major	EVENT	ISUPMOD: Unexpected STREAMS message [msg=0x%x]	ISUPMOD module has received a STREAMS message of unexpected type and discarded it. Report problem to TAC.
990201	Major	EVENT	ISUPMOD: UPM STREAMS multiplexor not found - Message discarded [msg=0x%x grp=%d]	ISUPMOD attempted to route a message to a Call Control on another host. However, ISUPMOD could not locate the MTP3 on that host. This could be caused by MTP3 software on the remote host going down; no action necessary.
990401	Critical	EVENT	ISUPMOD: Cannot acquire trunk table [key=0x%x reason=%d]	An attempt to get the dkm segment for indicated key failed. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.
990402	Major	EVENT	ISUPMOD: Cannot lock trunk record [trk=%d reason=%d]	An attempt to lock Call Control address table for indicated trunk failed for the given reason. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.
990403	Major	EVENT	ISUPMOD: Cannot unlock trunk record [trk=%d reason=%d]	An attempt to unlock Call Control address table for indicated trunk failed for the given reason. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.
990404	Major	EVENT	ISUPMOD: Cannot lock trunk table [dkmid=%d reason=%d]	An attempt to lock dkm segment failed for the given reason. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.
990405	Major	EVENT	ISUPMOD: Cannot unlock trunk table [dkmid=%d reason=%d]	An attempt to unlock dkm segment failed for the given reason. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.
990406	Critical	EVENT	ISUPMOD: Cannot destroy trunk table [dkmid=%d reason=%d]	The indicated dkm segment cannot be destroyed for the given reason. Log alarm sequence and call NewNet Communications Technologies, LLC TAC.

Table C-16: GSM-A Alarm Group

Alarm No.	Severity	Type	Message	Operation
9a0101	Major	EVENT	GSM-A: Inconsistency in Message and IE Databases [errno=%d]	
9a0102	Critical	EVENT	GSM-A: Registration failure [errno=%d]	
9a0103	Major	EVENT	GSM-A: Cannot send registration request [errno=%d]	
9a0104	Major	EVENT	GSM-A: Invalid registration ACK received [errno=%d]	
9a0105	Major	EVENT	GSM-A: Cannot access timer configuration file [errno=%d]	
9a0201	Major	EVENT	GSM-A: Message receive operation failed [errno=%d]	
9a0202	Minor	EVENT	GSM-A: Unexpected message - Message discarded [msg=0x%x]	
9a0203	Minor	EVENT	GSM-A: N_UnidataReq failure - Message discarded [msg=0x%x]	
9a0204	Minorl	EVENT	GSM-A: N_ConnectReq failure - Message discarded [msg=0x%x id=0x%x]	
9a0205	Minor	EVENT	GSM-A: Connection ID in use, resetting connection [msg=0x%x id=0x%x]	
9a0206	Minor	EVENT	GSM-A: Unexpected N_CONNECT_response [msg=0x%x id=0x%x]	
9a0207	Minor	EVENT	GSM-A: N_ConnectRsp failure [msg=0x%x id=0x%x]	
9a0208	Minor	EVENT	GSM-A: Unexpected N_DATA_request [msg=0x%x id=0x%x]	
9a0209	Minor	EVENT	GSM-A: N_DataReq failure [msg=0x%x id=0x%x]	
9a020a	Minor	EVENT	GSM-A: Unexpected N_DISCONN_request [msg=0x%x id=0x%x]	
9a020b	Minor	EVENT	GSM-A: N_DisconnectReq failure [msg=0x%x id=0x%x]	
9a020c	Minor	EVENT	GSM-A: Unexpected SCCP primitive [msg=0x%x]	
9a020d	Major	EVENT	GSM-A: SCCP registration failure [sp=%d ssn=%d]	
9a020e	Major	EVENT	GSM-A: mtp_spc failure	
9a020f	Minor	EVENT	GSM-A: N_StateReq failure	
9a0210	Critical	EVENT	GSM-A: SCCP re-registration failure [sp=%d ssn=%d]	
9a0211	Minor	EVENT	GSM-A: Connection ID in use, resetting [id=0x%x]	

Table C-16: GSM-A Alarm Group (Continued)

Alarm No.	Severity	Type	Message	Operation
9a0212	Minor	EVENT	GSM-A: Unrecognized N_CONNECT_confirmation [id=0x%x]	
9a0213	Minor	EVENT	GSM-A: Unrecognized N_DATA_indication [id=0x%x]	
9a0214	Minor	EVENT	GSM-A: Unrecognized N_DISCONN_indication [id=0x%x]	
9a020a	Minor	EVENT	GSM-A: Unexpected N_DISCONN_request [msg=0x%x id=0x%x]	
9a0301	Major	EVENT	GSM-A: Message send operation failed [type=0x%x len=%d]	
9a0401	Info	EVENT	GSM-A: GSMA registered successfully [sp=%d]	

Table C-17: PMMOD Alarm Group

Alarm no	Severity	Type	Message	Operation
9c0101	Major	EVENT	PMMOD: Invalid incoming SP number [sp=%]	Passive Monitor Layer sent a message having an invalid destination. Detach the board and attach it again.
9c0102	Major	EVENT	PMMOD: Invalid incoming message [stype=%d mtype=%d]	Passive Monitor Layer sent an invalid message type. Detach the board and attach it again.
9c0103	Minor	EVENT	PMMOD: Invalid incoming alarm [grp=0x%x mod=0x%x num=0x%x]	Passive Monitor Layer sent an invalid type of alarm.

Table C-18: PMON Alarm Group

Alarm no	Severity	Type	Message	Operation
9b0101	Info	EVENT	P-MON: SHORT frame, message discarded [spmlinkno=%d port=%d size=%d]	A frame is received with a smaller size than the size specified for SS7 networks. Frame is discarded. Check if monitoring an SS7 link
9b0102	Info	EVENT	P-MON: LONG frame, message discarded [spmlinkno=%d port=%d p2=internal]	A frame is received with a size greater than the size specified for SS7 networks. Frame is discarded. Check if monitoring an SS7 link
9b0103	Info	EVENT	P-MON: CRC error, message discarded [spmlinkno=%d port=%d p2=internal]	Check cables, and link/line clocking mode.
9b0104	Info	EVENT	P-MON: ABORTED frame, message discarded [spmlinkno=%d port=%d p2=internal]	Check cables, and connectors. Also check link baud rates.
9b0105	Info	EVENT	P-MON: Non octet aligned frame, message discarded [spmlinkno=%d port=%d p2=internal]	N/A

Table C-18: PMON Alarm Group (Continued)

Alarm no	Severity	Type	Message	Operation
9b0106	Info	EVENT	P-MON: Hardware, IRQ table overflow [spmlinkno=%d p1=0, p2=internal]	Information only. This alarm can safely be ignored.
9b0107	Major	EVENT	P-MON: Hardware, RX global overflow [spmlinkno=%d p1=0, p2=internal]	Product capacity limits are exceeded, or very abnormal link events are detected on one or more links. Probably many MSUs are dropped. Check if excessive abnormal alarm messages are being received. If so, try to firm cables. If abnormal link alarms are not removed, ignore this alarm message until the condition is corrected. Otherwise, make sure that product capacity limits are not exceeded. If not, consult to TAC.
9b0108	Major	EVENT	P-MON: Hardware, RX descriptor overflow [spmlinkno=%d p1=0, p2=internal]	Product capacity limits are exceeded, or very abnormal link events are detected on one or more links. Probably many MSUs are dropped. Check if excessive abnormal alarm messages are being received. If so, try to firm cables. If abnormal link alarms are not removed, ignore this alarm message until the condition is corrected. Otherwise, make sure that product capacity limits are not exceeded. If not, consult to TAC.
9b0109	Minor	EVENT	P-MON: MSU received in OOS state [spmlinkno=%d port=%d sio=%d]	Abnormal SS7 event. Received message is discarded. Check if monitoring an SS7 link.

Index

Index

Index

Index

Index

Index

