Cisco Billing and Measurements Server User's Guide, Release 3.20

Software Release 3.20
April 2010

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number:
Text Part Number: OL-6975-01 Rev. A8

Cisco Billing and Measurements Server User's Guide, Release 3.20

CONTENTS

CHAPTER 4

Preface

This section provides the following information:

Who Should Use This Guide

This guide provides the procedures you need to configure and manage the Cisco Billing and Measurements Server (BAMS) using the Man-Machine Language (MML) command-line interface.

This guide is intended for people responsible for system configuration and day-to-day operations.

Caution Users of this guide should be familiar with Central Office direct and tandem switch operations; SS7 signaling operations; Cisco PGW 2200 PSTN Gateway (formerly known as the Cisco Media Gateway Controller), in particular its Call Control and Signalling modes; network management tools and protocols, such as FTP, TCP/IP; UNIX system administration for Solaris systems with Korn shell; SUN Netra platform management; telco billing operations, including BAF Bellcore requirements; use of MML; and Cisco requirements on which BAMS software is based.

How This Guide Is Organized

The Cisco *Billing and Measurements Server User's Guide* is organized into the following chapters:

- Chapter 4, Using MML Commands
- Chapter 5, Using BAMS Tag IDs
- Chapter 6, Configuring BAMS for BAF Output
- Chapter 7, Configuring BAMS for ASCII Output and Measurements
- Chapter 8, Configuring BAMS for P01 Output
- Chapter 9, Configuring BAMS for NICS Output
- Chapter 10, Configuring BAMS for 1110 Binary Output
- Chapter 11, Obtaining Measurements
- Appendix A, Troubleshooting Cisco BAMS
- Appendix B, Upgrading to BAMS Release 3.20
- Appendix C, Backing Up and Restoring BAMS
- Glossary

Understanding Document Conventions

The following conventions are used in the procedures in this guide:

- All steps are numbered from 1.
- If there is more than one effort to a step, the step is broken down by the letters a, b, c, and so on.
- If there is more than one option in a step, the options are identified by bullet points. You can select only one option.
- Table 1 lists MML conventions.

Table 1 MML Conventions

Convention	Meaning	Examples and Comments
Boldface	Commands and keywords you enter literally as shown	**offset-list**
Italics	Variables for which you supply values	**command** *type interface* You replace the variable with the type of interface. In contexts that do not allow italics, such as online help, arguments are enclosed in angle brackets (< >).
Square brackets ([ ])	Optional elements	**command** [abc] abc is optional (not required), but you can choose it.
Vertical bars (|)	Separated alternative elements	**command** [abc | def] You can choose either abc or def, or neither, but not both.

Table 1 MML Conventions (continued)

Convention	Meaning	Examples and Comments
Braces ({ })	Required choices	**command** { abc | def } You **must** use either abc **or** def, but not both.
Braces and vertical bars within square brackets ([{ | }])	A required choice within an optional element	**command** [abc { def | ghi }] You have three options: • nothing • abc def • abc ghi
A string	A nonquoted set of characters	For example, when you set a Simple Network Management Protocol (SNMP) community string to *public*, do not use quotation marks around the string; otherwise, the string will include the quotation marks.
System prompts	Denotes interactive sessions and indicates that the user enters commands at the prompt	The system prompt indicates the current command mode. For example, the prompt `mml:sys>` indicates an open MML session at the system level, and `mml:`*n*`>` indicates a session for the displayed *n*th node. **Note** *n* can be any whole number from 1 to 8, or a node name; for example, `mml:1>` or `mml:1:boston>`
Screen font	Terminal sessions and information the system displays	For example: `Save`
Angle brackets (< >)	Nonprinting characters such as passwords	For example: <*password*>

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Note Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.

Tip Means *the following information might help you solve a problem.*

Timesaver Means the *described action saves time*. You can save time by performing the action described in the paragraph.

Related Documentation

Cisco Media Gateway Controller Hardware Installation Guide

Cisco Media Gateway Controller Software Release 9 Installation and Configuration Guide

Cisco MML Command Reference (online document)

Cisco PGW 2200 PSTN Gateway installation and user documentation

Sun Solaris UNIX user documentation

Tape backup instructions: Sun Netra j2.0 (or later) *Answerbook (Administrator's Guide)*

SNMP information: http://www.isp-planet.com/equipment/automanage1.html

Sun Microsystems information: http://www.sun.com

General UNIX information: SYBEX: *UNIX Complete*

General networking information: SYBEX: *Networking Complete*

BAF documentation: Bellcore GR-1100-CORE

Obtaining Documentation

Cisco documentation and additional literature are available on Cisco.com. Cisco also provides several ways to obtain technical assistance and other technical resources. These sections explain how to obtain technical information from Cisco Systems.

Cisco.com

You can access the most current Cisco documentation at this URL:

http://www.cisco.com/univercd/home/home.htm

You can access the Cisco website at this URL:

http://www.cisco.com

You can access international Cisco websites at this URL:

http://www.cisco.com/public/countries_languages.shtml

Ordering Documentation

You can find instructions for ordering documentation at this URL:

http://www.cisco.com/univercd/cc/td/doc/es_inpck/pdi.htm

You can order Cisco documentation in these ways:

- Registered Cisco.com users (Cisco direct customers) can order Cisco product documentation from the Ordering tool:

  http://www.cisco.com/en/US/partner/ordering/index.shtml

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco Systems Corporate Headquarters (California, USA) at 408 526-7208 or, elsewhere in North America, by calling 1 800 553-NETS (6387).

Documentation Feedback

You can send comments about technical documentation to bug-doc@cisco.com.

You can submit comments by using the response card (if present) behind the front cover of your document or by writing to the following address:

Cisco Systems
Attn: Customer Document Ordering
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Obtaining Technical Assistance

For all customers, partners, resellers, and distributors who hold valid Cisco service contracts, Cisco Technical Support provides 24-hour-a-day, award-winning technical assistance. The Cisco Technical Support Website on Cisco.com features extensive online support resources. In addition, Cisco Technical Assistance Center (TAC) engineers provide telephone support. If you do not hold a valid Cisco service contract, contact your reseller.

Cisco Technical Support Website

The Cisco Technical Support Website provides online documents and tools for troubleshooting and resolving technical issues with Cisco products and technologies. The website is available 24 hours a day, 365 days a year, at this URL:

http://www.cisco.com/techsupport

Access to all tools on the Cisco Technical Support Website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register at this URL:

http://tools.cisco.com/RPF/register/register.do

Note Use the Cisco Product Identification (CPI) tool to locate your product serial number before submitting a web or phone request for service. You can access the CPI tool from the Cisco Technical Support Website by clicking the **Tools & Resources** link under Documentation & Tools. Choose **Cisco Product Identification Tool** from the Alphabetical Index drop-down list, or click the **Cisco Product Identification Tool** link under Alerts & RMAs. The CPI tool offers three search options: by product ID or model name; by tree view; or for certain products, by copying and pasting **show** command output.

Search results show an illustration of your product with the serial number label location highlighted. Locate the serial number label on your product and record the information before placing a service call.

Submitting a Service Request

Using the online TAC Service Request Tool is the fastest way to open S3 and S4 service requests. (S3 and S4 service requests are those in which your network is minimally impaired or for which you require product information.) After you describe your situation, the TAC Service Request Tool provides recommended solutions. If your issue is not resolved using the recommended resources, your service request is assigned to a Cisco TAC engineer. The TAC Service Request Tool is located at this URL:

http://www.cisco.com/techsupport/servicerequest

For S1 or S2 service requests or if you do not have Internet access, contact the Cisco TAC by telephone. (S1 or S2 service requests are those in which your production network is down or severely degraded.) Cisco TAC engineers are assigned immediately to S1 and S2 service requests to help keep your business operations running smoothly.

To open a service request by telephone, use one of the following numbers:

Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)
EMEA: +32 2 704 55 55
USA: 1 800 553-2447

For a complete list of Cisco TAC contacts, go to this URL:

http://www.cisco.com/techsupport/contacts

Definitions of Service Request Severity

To ensure that all service requests are reported in a standard format, Cisco has established severity definitions.

Severity 1 (S1)—Your network is “down,” or there is a critical impact to your business operations. You and Cisco will commit all necessary resources around the clock to resolve the situation.

Severity 2 (S2)—Operation of an existing network is severely degraded, or significant aspects of your business operation are negatively affected by inadequate performance of Cisco products. You and Cisco will commit full-time resources during normal business hours to resolve the situation.

Severity 3 (S3)—Operational performance of your network is impaired, but most business operations remain functional. You and Cisco will commit resources during normal business hours to restore service to satisfactory levels.

Severity 4 (S4)—You require information or assistance with Cisco product capabilities, installation, or configuration. There is little or no effect on your business operations.

Obtaining Additional Publications and Information

Information about Cisco products, technologies, and network solutions is available from various online and printed sources.

- Cisco Marketplace provides a variety of Cisco books, reference guides, and logo merchandise. Visit Cisco Marketplace, the company store, at this URL:

  http://www.cisco.com/go/marketplace/

- The Cisco *Product Catalog* describes the networking products offered by Cisco Systems, as well as ordering and customer support services. Access the Cisco Product Catalog at this URL:

  http://cisco.com/univercd/cc/td/doc/pcat/

- *Cisco Press* publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press at this URL:

  http://www.ciscopress.com

- *Packet* magazine is the Cisco Systems technical user magazine for maximizing Internet and networking investments. Each quarter, Packet delivers coverage of the latest industry trends, technology breakthroughs, and Cisco products and solutions, as well as network deployment and troubleshooting tips, configuration examples, customer case studies, certification and training information, and links to scores of in-depth online resources. You can access Packet magazine at this URL:

  http://www.cisco.com/packet

- *iQ Magazine* is the quarterly publication from Cisco Systems designed to help growing companies learn how they can use technology to increase revenue, streamline their business, and expand services. The publication identifies the challenges facing these companies and the technologies to help solve them, using real-world case studies and business strategies to help readers make sound technology investment decisions. You can access iQ Magazine at this URL:

  http://www.cisco.com/go/iqmagazine

- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:

  http://www.cisco.com/ipj

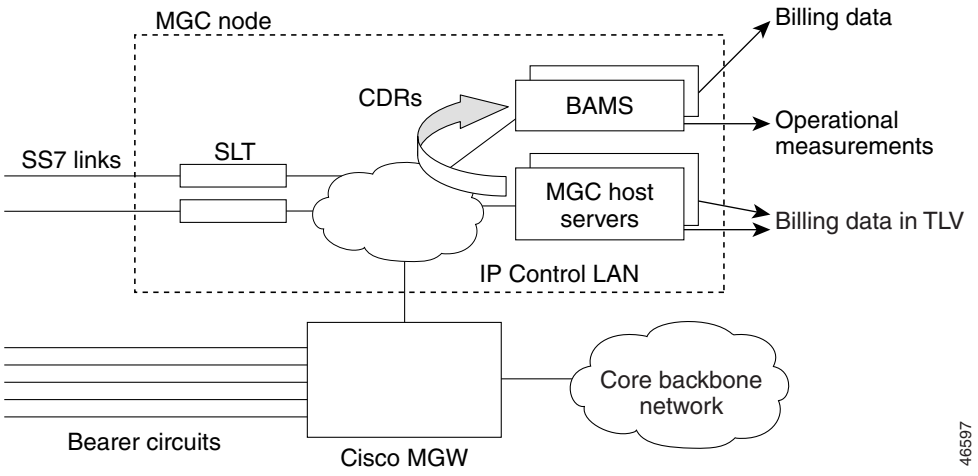
- World-class networking training is available from Cisco. You can view current offerings at this URL:

  http://www.cisco.com/en/US/learning/index.html

Introduction to Cisco BAMS

The Cisco Billing and Measurements Server (BAMS) collects, formats, and stores billing and measurements data from the Cisco Media Gateway Controller (MGC), either a Cisco VSC3000 virtual switch controller or a Cisco SC2200 application (see Figure 1-1). BAMS-formatted data can then be processed by a billing system and other measurement collection and reporting systems. BAMS also reports error information using the *bamstrap* utility. This error information is collected and displayed by the Cisco MGC Node Management System.

Figure 1-1 Cisco PGW 2200 PSTN Gateway Components



Collecting Data

Cisco BAMS collects data from the Cisco MGC through File Transfer Protocol (FTP). BAMS is configured with a login ID and password for the Cisco MGC hosts. BAMS can collect data from the Cisco MGC in simplex or redundant mode operation.

Data Collection

Cisco BAMS can collect data from up to eight MGC nodes with BAMS software Release 3.0 and above. Data collected from the first MGC node is stored in the /opt/CiscoBAMS/data/s01 directory, data collected from the second MGC pair is stored in the /opt/CiscoBAMS/data/s02 directory, and so forth. Cisco BAMS collects data from the Cisco MGC through the File Transfer Protocol (FTP).

Each BAMS node can collect from a single MGC node. The polling function of each BAMS node is configured with a login ID and password for the Cisco MGC hosts. BAMS can collect data from the Cisco MGC in simplex or redundant mode of operation. In a simplex mode of operation, the single BAMS unit performs the data collection across all eight nodes. In redundant mode, for each node, one BAMS unit actively collects from the MGC while the other unit is in standby mode. The active/standby polling status is independent for each BAMS node. On a single BAMS unit in a redundant system, some nodes may be in the active collection state, while other nodes are in the standby state. The corresponding nodes on the other BAMS unit will have the opposite polling status.

For each node, the data collection can be manually rotated from the active to the standby BAMS unit. Such a rotation is referred to as a data switchover and can be carried out routinely for preventive maintenance purposes. The fail-safe mode of operation invokes an automatic rotation from the active BAMS unit to the standby unit in the event of a system failure on the active unit.

Cisco BAMS Terminology

Cisco has changed some of the names for its products. Not all of these name changes have yet been made within the Cisco BAMS software. Note the following terms used in the Cisco BAMS software and configurations and their more recent Cisco MGC software equivalent.

Cisco BAMS Software Term	Cisco MGC Software Term
Nailed	Signaling mode
Switched	Call Control mode
Cisco MGC node	Cisco PGW node
Cisco MGC software	Cisco PGW software
Cisco SC2200 Signalling Controller	Cisco PGW 2200 PSTN Gateway
Cisco VSC3000 Virtual Switch Controller	Cisco PGW 2200 PSTN Gateway

Formatting Data

The raw data collected from the Cisco MGC units is in the form of binary files. This data must be converted into a format that the billing system can recognize. In addition to formatting the data files, BAMS also augments the billing data, adding data required by the billing system but not automatically provided by the Cisco MGC. BAMS validates the records and flags exceptions.

Finally, BAMS provides the data in a uniform format to the billing system, which permits record comparisons. Because the records are normalized, they are in a format suitable for statistical analysis. In addition to standardized billing data, BAMS generates measurements data. Measurements can be taken in a specified time interval and for a variety of data objects. This data is suitable for traffic studies on network usage.

Note Depending on the error flagged during validation, the record can be written to an error file and dropped, or it can flow through the system with default values. In either case, alarms are generated and written to the system log file (syslog). In general, structure or record-format errors result in the record being dropped. Lookup errors result in the record acquiring default values.

Storing Data

BAMS stores the formatted data on disk in a data directory where it can be collected or polled by the billing system. In addition to billing data, BAMS stores exception and measurements data, and writes system messages to a log file. The data collected by the billing system is renamed and is deleted from BAMS after archival, if desired, by the Mass Storage Control (MSC) task.

Files collected by BAMS from the Cisco MGC are time-stamped and sequence-numbered. Successfully formatted files are renamed with special prefix and suffix symbols. The MSC task must be set up with this file pattern information. The MSC can also be configured to generate alarms when specified data thresholds are passed. Alarm notification can also be set up for aged files. Alarms flag polling problems and prevent available disk space from filling up completely and halting or otherwise impeding system operation.

Note Billing files contain Bellcore AMA (Automatic Message Accounting) Format (BAF) records. BAF records do not contain an indexing field, such as a record count or sequence number. The billing file name, however, can be traced back to the original input file from the Cisco MGC.

Trapping Errors

Each BAMS software task generates its own set of error messages. A Simple Network Management Protocol (SNMP) task, *bamstrap*, is used to manage the trap messages. All messages are written to the system log by the Alarm task (ALM), converted into traps by bamstrap, and forwarded to the Cisco MGC Node Manager for response. The operator can then determine appropriate action based on the reported alarm or event.

System Backup and Recovery

Backup of data on BAMS can be accomplished in one of the following ways:

- **For BAMS Release 3.20 and later:** By using the BAMS utility, bamsbackup, to back up the configuration and data files to a file or device. For more information, see the "Backing Up and Restoring BAMS" section on page C-1.
- **For BAMS Release 3.14 and earlier:** By backing up the data from the Sun Netra to a mass storage device, in which case the storage device must be supported by the Sun Netra platform.
- **For BAMS Release 3.14 and earlier:** By collecting the data from BAMS over a TCP/IP network connection when backup is required and storing it on a backup server.

The procedures for backing up and restoring data on Sun Solaris client/server systems are well established and documented, and several related tools are available to the UNIX system administrator. In addition to Sun Solaris documentation, third-party vendor documentation (for example, by Veritas) and general reference information are provided by UNIX software publishers, including O'Reilly and Wiley.

The best backup strategy depends on the user's network configuration and available resources. In general, users should collect files in an FTP session as often as necessary and store them as required. Each file recommended for backup is indicated in Table 1-1 by a superscript number following the description.

See the Sun Netra j20 (or later) *Answerbook (Administrator's Guide)* for system backup instructions.

BAMS features built-in recovery processes for system interrupt or crash. If the unit crashes in the middle of a provisioning session, restart the session using the dstver that the system crashed with as your srcver, and use a new dstver (see "prov-sta—Provision Start" section on page 4-15 for details).

Note Contact the Cisco Technical Assistance Center for assistance with performing a BAMS system backup or restoring BAMS from a previous backup.

Installing the Sun Solaris 8 Operating System

Cisco BAMS 3.x requires the Sun Solaris 8 operating system. In addition, Cisco BAMS requires that you follow specific guidelines in the partitioning of the BAMS hard disks.

Before you install Cisco BAMS Release 3.x software, you must make sure that Sun Solaris 8 is installed and that the Cisco BAMS server disks are properly partitioned. For information on installing Solaris 8 installation and disk partitioning, see the *Cisco Media Gateway Controller Software Release 9 Installation and Configuration Guide* located at:

http://www.cisco.com/univercd/cc/td/doc/product/access/sc/rel9/swinstl/index.htm

Creating Directory Structures

Data directories are created for you during BAMS installation. Table 1-1 describes the Cisco BAMS directories.

Table 1-1 Cisco BAMS Directories

Directory	Description
opt/CiscoBAMS/bin	Contains executable processes and shell scripts.
opt/CiscoBAMS/CDR	Contains the call record data.[1]
opt/CiscoBAMS/CDR/archive	Contains archived call record data.
opt/CiscoBAMS/contrib	Contains system utilities.
opt/CiscoBAMS/data	Contains data subdirectories.[1]
opt/CiscoBAMS/data/error	Contains error files.
opt/CiscoBAMS/data/logs	Contains daily MML session logs.
opt/CiscoBAMS/data/s0x	Contains node data files.
opt/CiscoBAMS/data/*nodename0x*	When a node name is assigned to node0x, a directory link with that name is created for s0x.
opt/CiscoBAMS/data/s0x/ASCII	Contains generated Cisco MGC End of Call 1110 files.
opt/CiscoBAMS/data/s0x/ASCIIBAF	Contains generated ASCII BAF files.
opt/CiscoBAMS/data/s0x/BAF	Contains generated BAF files.
opt/CiscoBAMS/data/s0x/BIN1110	Contains generated 1110 Binary files.
opt/CiscoBAMS/data/s0x/error	Contains node-specific error files.

Table 1-1 Cisco BAMS Directories (continued)

Directory	Description
opt/CiscoBAMS/data/s0x/EXTASCII	Contains generated extendable ASCII files.
opt/CiscoBAMS/data/s0x/input	Contains input files.
opt/CiscoBAMS/data/s0x/Measurements	Contains measurements data.
opt/CiscoBAMS/data/s0x/NICS	Contains generated NICS files.
opt/CiscoBAMS/data/s0x/output	Contains output files. This directory contains files that are currently being processed. **Note** Any files in this directory that are older than a few weeks can be manually removed as a preventive maintenance procedure.
opt/CiscoBAMS/data/s0x/p001	Contains generated P01 records.
opt/CiscoBAMS/data/s0x/TLV	Contains raw billing files from the Cisco MGC that are in TLV format.
opt/CiscoBAMS/data/s0x/xfr	This directory contains temporary, raw CDR files to be transferred to the redundant BAMS unit for redundant processing.
opt/CiscoBAMS/defaults	Contains default configuration files.[1]
opt/CiscoBAMS/files	Contains system configuration files and log files.[1]
opt/CiscoBAMS/files/mml	Contains configurations created by the user in provisioning sessions.
opt/CiscoBAMS/files/s0x	Contains configuration files and log files for node0x.[1]
opt/CiscoBAMS/testdata	Contains test files.
opt/CiscoBAMS/files/snmpmibs	Contains SNMP MIB definitions in ASCII formatted files.

1. Indicates files recommended for backup.

Setup and Installation

Overview

This chapter provides information on how to configure the Cisco Media Gateway Controller (MGC) for using the Billing and Measurements Server (BAMS), how to uninstall previous BAMS software versions, and how to install and configure new versions of BAMS.

Caution Before upgrading to a new BAMS release, ensure that you have backed up the latest configuration and data files. For more information, see the "System Backup and Recovery" section on page 1-3.

Upgrading to BAMS Release 3.20

If this is a first-time BAMS installation, follow the installation procedure in this chapter. If you are currently operating a version of BAMS prior to Release 3.20, you need to follow a migration procedure. Proceed to Appendix B, "Upgrading to BAMS Release 3.20."

Configuring the Cisco MGC for Using BAMS

You must edit your dmprSink.dat file to set the interval to dump the call detail records (CDRs) to the spool directory every 4 minutes. To do so, use this procedure:

Step 1 `cd /opt/CiscoMGC/var`

Step 2 `mkdir bam`

Step 3 `chmod 777 bam`

Step 4 Open a vi session:
`vi /opt/CiscoMGC/etc/dmprSink.dat`

Step 5 Edit it as follows:
`1 "callDetail" bin "cdr" "../var/log" "../var/bam" 8000 0 4`

Caution The last entry in the field in the preceding line (that is, "4") is the MGC CDR dump interval (in minutes). This value must be lower than and not equal to the BAMS measurements file interval (interval-minutes parameter) set in the Node Parameters table (see "Updating the Node Parameters Table" section on

page 5-10 for details). If the Cisco MGC file interval is greater than or equal to the BAMS measurement interval, more than one measurement interval could be crossed in a single input file. If this happens, a COR206 major alarm occurs. The end result is the loss of measurement data. This loss of data causes the ACC task to generate an ACC222 minor alarm.

Caution When upgrading the Cisco MGC software from release 7.4(11) or 7.4(12) to release 9.3(2) or 9.4(1) in a configuration that includes Cisco BAMS, you must edit your dmprSink.dat file as described in the preceding Steps 4 and 5. You must edit the dmprSink.dat file before you restart the PGW.

Step 6 Open another vi session on the XECfgParm.dat:

```
vi /opt/CiscoMGC/etc/XECfgParm.dat
```

Step 7 Verify the correct CDB settings:

a. `engine.CDRtimeStamp = M`

b. `engine.CDRmessageTypes = "1010,1020,1030,1040,1050,1060,1070,1071,1080"`

Note The time stamp must be in milliseconds (M) and the message types must be from 1010 through 1080, except 1110 is not supported when you are using BAMS in its normal configuration. However, BAMS can be configured to output Cisco MGC End of Call records (see Chapter 6, "Configuring BAMS for BAF Output," for details).

Step 8 Edit the diskmonitor parameter in XECfgParm.dat to indicate whether the CDR files are to be kept after collection. You can use the same parameter to indicate how long the files are to be kept. If the value is 0, the .finished files are removed immediately. A value of 1 means that the .finished files are deleted after 1 day, 2 for 2 days, and so on. In the following example, the files are removed after 1 day, as reflected in diskmonitor.CdrRmFinished = 1.

```
#----------------------------------
# Disk Monitor Parameters
#----------------------------------
diskmonitor.Limit = 7# MIGRATED
diskmonitor.Threshold = 80# MIGRATED
diskmonitor.SoftLimit = false# MIGRATED
diskmonitor.CdrRmFinished = 1# (remove "finished" cdrs after X days (0 = immediate))
diskmonitor.OptFileSys =# (list of optional file systems to monitor (no trimming))
```

Step 9 Open another vi session:

```
vi /opt/CiscoMGC/etc/CONFIG_LIB/new/XECfgParm.dat
```

Step 10 Repeat Step 4 through Step 8 in the active_link directory, the prov_link directory, and in the new directory.

Step 11 Stop the Cisco MGC:

```
/etc/init.d/CiscoMGC stop
```

Step 12 Start the Cisco MGC for the file changes to take effect:

```
/etc/init.d/CiscoMGC start
```

Step 13 Synchronize the operation of the Cisco PGW 2200 and Cisco BAMS systems by issuing the following MML command:

```
sta-aud-cic
```

Note The MML command sta-aud-cic starts a process that audits the CIC state. The command generates the SS7 CIC Audit CDB 1071 for every configured ISUP TrunkGroup in Call C.

Uninstalling BAMS

If a previous BAMS release is already installed, you must uninstall it to start a new system. This procedure removes all BAMS software; however, you can save the data and archive directories during the uninstall procedure. The uninstall procedure also asks you if you want to save the current active BAMS configuration to a restorable provisioning batch file.

Note The uninstallation process removes the contents of the /opt/CiscoBAMS directory except the /opt/CiscoBAMS/CDR and /opt/CiscoBAMS/data folders.

Caution Do not remove the /opt/CiscoBAMS/CDR and /opt/CiscoBAMS/data directories manually. They are disk partition mount points.

To uninstall BAMS, use this procedure:

Step 1 Log in as the BAMS user.

Step 2 Stop the application by entering the following commands:

```
$ cd /opt/CiscoBAMS/bin
$ mml
mml:sys> stp-softw::confirm
mml:sys> quit
```

Note The default UNIX prompt ($ or #) might vary with the shell program being used.

Step 3 Log out. Make sure that all BAMS logins have exited. This account is removed during the uninstallation.

Step 4 Log in as the root user by entering the following command:

```
:root
:<password>
```

Step 5 Uninstall the application software:

```
$ cd /opt/install
```

Step 6 Run the uninstall script:

```
$ bams uninstall
```

Step 7 Follow the prompts from the uninstallation script. (The uninstall script automatically copies the BAMS configuration files from /opt/CiscoBAMS/files to /usr/tmp/bamsconfig.)

Step 8 Uninstall the application software:

```
$ cd /opt
$ pkgrm CSCOcaBAM
```

This completes the uninstallation procedure.

Installing Cisco BAMS

Note Cisco BAMS Release 3.x requires the Sun Solaris 8 operating system. In addition, Cisco BAMS requires that you follow specific guidelines in the partitioning of the BAMS hard disks.

Before you install Cisco BAMS Release 3.x software, you must make sure that Sun Solaris 8 is installed and that the Cisco BAMS server disks are properly partitioned. For information on installing Solaris 8 installation and disk partitioning, see the *Cisco Media Gateway Controller Software Release 9 Installation and Configuration Guide* located at:

http://www.cisco.com/univercd/cc/td/doc/product/access/sc/rel9/swinstl/index.htm

Step 1 Insert the CD labeled "Cisco Billing and Measurements Server (BAMS)" (Release 3.20 or higher) in a CD-ROM drive.

Step 2 For each machine on which BAMS is to be installed, obtain the following information from your system administrator:

a. Platform host name.

b. Platform IP address.

c. Platform user name.

d. Platform user password.

e. Home directory for the BAMS account (default: /opt/CiscoBAMS/local).

f. Where to put the /opt/CiscoBAMS/data directory for all data files created by the application. This is usually a separate disk or file system (default: /opt/CiscoBAMS).

Note The system prompts are displayed in the UNIX Korn shell. Refer to user documentation such as *UNIX Complete* by SYBEX for more information on the Korn shell. You must be familiar with the Korn shell and use of the UNIX vi text editor in order to properly set up and configure BAMS.

Step 3 Log in as the root user by entering the following command:

```
:root
:<password>
```

Step 4 Install the CSCOcaBAM package (from the solaris_patches directory) by entering the following commands:

```
$ cd /cdrom/cdrom0
$ pkgadd -d CSCOcaBAM.pkg
```

Step 5 Follow the prompts.

Caution Do not install the following Solaris 8 patch:

CSCOh005: It creates two unnecessary file systems on BAMS.

Step 6 Run the BAMS install script by entering the following commands:

```
$ cd /opt/install
$ ./bams install
```

Step 7 Follow the prompts.

Note **For upgrade installations:** An option to restore the BAMS configuration files is given during the installation:

Do you wish to restore BAMS configuration files from the previous installation? [y,n]

Answering “y” to this prompt restores the provisioning tables from the previous installation, provided that you saved the previous configurations when prompted to during the uninstallation procedure. This restore function also retains the simplex/redundant state configured by the **change_mode** command, the polling **seqno** for each node, and the **setbamunit** information.

Step 8 You are prompted for the following:

Enter the version of BAMS that was uninstalled prior to this installation of BAMS 3.20:

0) New installation
1) BAMS 2.68
2) BAMS 3.10
3) BAMS 3.12
4) BAMS 3.13
5) BAMS 3.14
6) BAMS 3.20

Enter (0 | 1 | 2 | 3 | 4 | 5 | 6)?

If you select any option from 1 to 5, the installation program converts any in-progress call records to the latest data format so that these records can be processed by the latest version of the software.

When the installation is complete, the system displays the following message: `Installation of <CiscoBAMS> was successful.`

Step 9 After the installation has successfully completed, reboot the system.

Step 10 Go to the Cisco website and download the latest patch for BAMS Release 3.x.

Step 11 Install the patch.

Configuring BAMS to Forward SNMP Traps

To configure BAMS to forward SNMP traps to a management application, perform the following steps:

Step 1 Log in as the root user by entering the following command:
```
:root
:password
```

Step 2 Open a vi session:

```
vi /etc/srconf/agt/snmpd.cnf
```

Step 3 Make sure the following lines exist under the snmpNotifyEntry section (if they are not there, add them):
```
snmpNotifyEntry 31 Console trap nonVolatile
snmpNotifyEntry 32 TrapSink trap nonVolatile
```

Note Number 31 must be unique in that section. Each entry number must be unique.

Step 4 Make sure the following lines exist under the snmpTargetAddrEntry section (if they are not there, add them):
```
snmpTargetAddrEntry 33 snmpUDPDomain 127.0.0.1:0 100 3 Console \
v1ExampleParams nonVolatile 255.255.255.255:0 2048
snmpTargetAddrEntry 34 snmpUDPDomain 127.0.0.1:0 100 3 Console \
v2cExampleParams nonVolatile 255.255.255.255:0 2048
```

Note Number 33 must be unique in that section. Each entry number must be unique.

Step 5 To forward a V1 trap to the IP address of the network management platform (A.B.C.D in the lines below), add the following line:
```
snmpTargetAddrEntry 35 snmpUDPDomain A.B.C.D:0 100 3 TrapSink \
v1cExampleParams nonVolatile 255.255.255.255:0 2048
```

Step 6 To forward a V2c trap to destination A.B.C.D, add the following line:
```
snmpTargetAddrEntry 36 snmpUDPDomain A.B.C.D:0 100 3 TrapSink \
v2cExampleParams nonVolatile 255.255.255.255:0 2048
```

Step 7 Restart the snmpdm process:
kill -9 *pid*

Note The value of pid is the process ID for snmpdm, and snmpdm restarts itself after termination.

Configuring BAMS

To configure BAMS, perform the following steps:

Step 1 Log in as the root user by entering the following command:

```
:root
:password
```

Step 2 Define the network node names and IP addresses of the Cisco MGC and the BAMS units in the file /etc/hosts. You must edit and save the host file with the information obtained from your system administrator, using the UNIX vi text editor to enter the following commands:

```
$ cd /etc/inet
$ chmod 744 hosts
$ vi hosts
```

Step 3 Log out as the root user.

Caution All of the dual Ethernet connections and host names should be set up and defined for each BAMS and Cisco MGC unit. The redundant Ethernet connection should have a suffix of _b; for example: MGC1 hostname=ciscomgc1, MGC1 redundant connection hostname=ciscomgc1_b.

Step 4 Log in as the BAMS user.

Step 5 (Skip this step if you are upgrading your BAMS version and you chose to restore the previous configuration files during the installation procedure.)

Set up the basic BAMS configuration: simplex or redundant.

a. To change the system settings for simplex versus redundant mode, run **change_mode** and follow the instructions. The following change_mode example shows user input for setting up BAMS in redundant mode:

```
$ cd /opt/CiscoBAMS/bin
$ change_mode

The change_mode program sets the BAMS configuration to simplex or redundant.
Do you want to continue (y/n)? y

Enter mode of operation:
1. Simplex BAMs mode
2. Redundant BAMs mode

Enter mode [1-2]: 2

To finish configuration for a redundant system:

1. On unit 1, set UNITNO to 1 in file /opt/CiscoBAMS/bin/sym_defs
2. Run /opt/CiscoBAMS/bin/setbamunit and enter all parameters

User should execute the following to set up current environment:
$ ./opt/CiscoBAMS/bin/sym_defs

As an alternative, the user can log out and log in again to re-set the shell environment.
```

To set up the BAMS unit for SC2200 applications and check the measurements reporting interval, start an MML provisioning session, modify the configuration of the Node Parameters table using the NODEPARMS tag ID, and deploy any changes you make.

Refer to the "Configuring Node Parameters" section on page 2-12.

Step 6 (Skip this step if you are upgrading your BAMS version and you chose to restore the previous configuration files during the installation procedure.)

Complete the instructions given at the end of the **change_mode** program:

a. Set the unit designation.

Note For a redundant configuration, Step 6 must be performed on both units.

b. Modify the UNITNO environment variable definitions to define the unit number in the file /opt/CiscoBAMS/bin/sym_defs. (The primary BAMS unit should be 0, and the secondary BAMS unit should be 1.)

Note You can designate the Cisco MGC units 1 and 2. You must confirm the unit designations with your system administrator. Do not modify the BAMS 0 and 1 unit numbers to match the Cisco MGC numbers 1 and 2, or the system will not work as intended.

c. After unit designation is complete, reset the system environment variables by entering the following commands:

```
$ cd /opt/CiscoBAMS/bin
$ . sym_defs
```

Note Only valid users with appropriate permissions can access the Cisco MGC. They must be set up with permissions in the Cisco MGC group for Cisco MGC software Release 7.4 and later releases, and in the TransPath group for releases before software Release 7.4.

d. From within the /opt/CiscoBAMS/bin directory, enter the **setbamunit** command to configure the host name, login ID, and password parameters for both BAMS units.

All parameters should be defined. The parameters bam_name0 and bam_name1 should correspond to the node names of the BAMS units defined in the file /etc/hosts. See the following example:

```
$ cd /opt/CiscoBAMS/bin
$ setbamunit
New value for 'bam-name0' [ ]: bams0
New value for 'login0' [bams]: bams
Enter new password for 'passwod0': xxxxxxxx
Re-enter password: xxxxxxxx
New value for 'bam-name1' [ ] : bams1
New value for 'login1' [bams] : bams
Enter new password for 'passwd1': yyyyyyyy
Re-enter password: yyyyyyyy
$
```

Step 7 Configure the Cisco MGC polling parameters for the first node by using the Man-Machine Language (MML) program to set the first node. This creates the initial polling control file for node 1 (opt/CiscoBAMS/files/s01/poll.CTL).

```
$ cd /opt/CiscoBAMS/bin
$ mml
```

You can also change the MML environment by just entering the node identifier:

```
mml:sys> set-node:1:
```

To start an MML editing session, enter the following command:

```
mml:1> prov-sta::srcver=active,dstver=test1
Billing and Measurements Server - BAMS-00 2004-10-09 15:20:30
B  COMPLD
;
```

To set the node name “boston” for the first node, enter the following command:

```
mml:sys> set-nodename:1:name="boston"
```

To display the polling record for this node, enter the following command:

```
mml:1> prov-rtrv:poll:
Billing and Measurements Server - BAMS-00 2004-10-09 15:30:02
B  RTRV
;
```

Initially, this file is empty.
To display the configuration parameters, enter the following command:

```
mml:1> prov-rtrv:poll:?
Billing and Measurements Server - BAMS-00 2004-10-09 15:21:49
B  COMPLD
host-name1 (string)
rem-dir1 (string)
file-prefix1 (string)
file-suffix1 (string)
host-name2 (string)
rem-dir2 (string)
file-prefix2 (string)
file-suffix2 (string)
action (string)
interval (numeric)
maxtries (numeric)
;
```

When you add a polling record to the Poll table, you must specify the file prefix (for example, “cdr*”) and suffix (for example, “.bin”) for the CDR files on each host. The CDR files produced on the Cisco MGC switch use the following naming convention:

<prefix>_yyyymmddhhMMss_nnnnnn.bin

where:

<*prefix*> is defined in the Cisco MGC switch configuration (for example, “cdr”)

yyyymmddhhMMss is the timestamp. BAMS ignores the timestamp when polling; it uses the file sequence number instead.)

nnnnnn is the file sequence number. You specify this number with the **chgno** command in Step 9.

.*bin* is the suffix

To add a polling record (all required fields and values must be specified), enter the following command:

```
mml:1> prov-add:poll:host-name1="MGCtest1",rem-dir1="/opt/CiscoMGC/var/bam",
file-prefix1="cdr*",file-suffix1=".bin",host-name1="MGCtest2",file-prefix2="cdr*",file-suf
fix2=".bin",action="R",interval=5,timeout=2,maxtries=2
Billing and Measurements Server - BAMS-00 2004-10-09 16:28:02
B  COMPLD
"POLL"
;
```

In the example above, the file-prefix1 and file-suffix1 parameters must correspond to the file generated on the Cisco MGC switch 1 (host-name1 parameter), and the file-prefix2 and file-suffix2 parameters must correspond to the file generated on the Cisco MGC switch 2 (host-name2 parameter).

To edit the polling configuration information for this node, use the following command:

```
mml:1> prov-ed:poll:maxtries=3
```

Refer to the "POLL Tag ID" section on page 5-17 for details on the Poll table's required fields and values.

Step 8 To deploy the changes, use the following command:

```
mml:1> prov-dply::
```

If the remote unit is online, the files are synchronized and messages such as the following display on the screen:

```
/* Synchronizing remote BAMS ... */
/* Transferring file /opt/CiscoBAMS/files/s01/zoneinfo.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/TrunkGroup.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/alm_parms.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/country.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/maptype.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/nodeParm.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/NpaNxx.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/poll.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/RatingType.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/RatingException.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/NailedConn.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/skipcdb.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/skipcde.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/thresholds.CTL ... */
/* Transferring file /opt/CiscoBAMS/files/s01/tollfree.CTL ... */
;
```

Note You must be in an active provisioning session (**prov-sta** command) to edit any parameters; use the **prov-dply** command to implement the changes.

Step 9 (Skip this step if you are upgrading your BAMS version and you chose to restore the previous configuration files during the installation procedure.)

Identify the sequence number of the first file to be polled from each Cisco MGC switch. You modify the polling file sequence number by using the MML **chgno** command. This file sequence number must correspond to the sequence number of the files generated by the Cisco MGC switches. Refer to Step 7 for the filename format. You can change one or both file sequence numbers with this command.

```
mml:1> chgno::[seqno1=#,][seqno2=#]
```

seqno1 is the file sequence number of the first polled unit, and seqno2 is the file sequence number of the second polled unit. For example:

```
chgno::seqno1=0001,seqno2=0002
```

The system should be stopped when you execute the **chgno** command. If the system is running, be sure it is not polling, or else a "Load record failed error" message could be generated. If **chgno** fails, you receive the message "Polling control record #1 in use by POL program — try again later."

Step 10 Quit the MML session by entering the following command:

```
mml:1> quit
```

Step 11 For security reasons, the login information for the Cisco MGCs is not entered through the MML interface. The BAMS user must execute the program /opt/CiscoBAMS/bin/setpollpwd to define the login IDs and passwords for the redundant Cisco MGC units.

```
$ cd /opt/CiscoBAMS/bin
$ setpollpwd -sx (Where x represents the node identifier: 1 to 8)
New value for 'userid1' [ ]: mgcuser1
Enter new password for 'password1': xxxxxxxx
Re-enter password: xxxxxxxx
New value for 'userid2' [ ]: mgcuser2
Enter new password for 'password2': yyyyyyyy
Re-enter password: yyyyyyyy
$
```

Step 12 Repeat Step 7 through Step 11 to set up each additional node (2 through 8).

Step 13 Configure the MSC Thresholds table and the MSC Parameters table to monitor the disk usage levels.

Note The MSC threshold levels have been set at default values, but you can modify them by entering the following commands:

```
$ cd /opt/CiscoBAMS/bin
$ mml
mml:sys> sta-softw
```

To display threshold values, enter the following command:

```
mml:sys> prov-rtrv:msc-thres:
```

To display file parameters, enter the following command:

```
mml:sys> prov-rtrv:msc-parms:
```

Note If BAMS Release 1.x has been uninstalled and is being replaced by BAMS Release 3.x, log in as a user and enter the following commands:

```
$ cd /opt/CiscoBAMS/data/output
$ mv /tmp/incomplete_cdr.cor.out incomplete_cdr.cor
```

Reconfigure the provisioning tables using the MML command.

Step 14 Start the system by entering the following commands:

```
$ cd /opt/CiscoBAMS/bin
$ mml
mml:sys> sta-softw
```

This completes the installation process.

Note You can also start or stop the system from the UNIX prompt by using **start_system** or **stop_system**.

Configuring Node Parameters

Use the NODEPARMS tag ID to set node parameters such as record-type outputs (for example, ASCII, BAF, NICS, P01) and measurements.

To display the node parameters, you need to first set a node. Enter the following commands:

```
$ cd/opt/CiscoBAMS/bin
$ mml
mml:sys> set-node:1:
Billing and Measurements Server - BAMS-00 2004-10-12 10:36:39
mml:1:boston> prov-rtrv:NODEPARMS:
Billing and Measurements Server - BAMS-00 2004-10-12 10:36:58
B  RTRV

"NODEPARMS:activate=1,statoutput=1,bafoutput=0,asciibafoutput=0,asciioutput=2,lookupinfo=1
,bafinfo=0,dynamicaccumes=0,sup-zero-counts=0,interval-minutes=15,nailed-cfg=1,p01output=0
,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,bin1110output=0"
   ;
```

Caution The Cisco MGC dump CDR interval (set in the dmprSink.dat file) must be set to a value less than and not equal to the BAMS measurements interval (interval-minutes parameter). For example, if the measurements interval is set to 5 minutes, the dump CDR interval must be set to 4 minutes or less. If the Cisco MGC dump CDR interval is greater than or equal to the BAMS measurements interval, the possibility exists that more than one measurements interval can be crossed in a single input file. If this happens, a COR206 major alarm occurs. The end result is the loss of measurement data. This loss of data

causes the ACC task to generate an ACC222 minor alarm. For details about modifying the Cisco MGC dump CDR interval value in the dmprSink.dat file, see "Configuring the Cisco MGC for Using BAMS" section on page 2-1.

Note Refer to the "NODEPARMS Tag ID" section on page 5-10 for details on the available node parameters and acceptable values.

Host Configuration Setup for Downstream Polling

You can collect files in the opt/CiscoBAMS/data directories (ASCII, BAF, measurements, NICS, and so forth) through an appropriate FTP session. This is referred to as FTP polling of BAMS by the downstream billing system. In an automated FTP session, data collected from BAMS is stored in appropriate directories on the downstream billing system. After a successful poll of a data file, the data file in the BAMS /data subdirectory should be renamed with a *.finished suffix. In a redundant configuration, the duplicate file on the redundant BAMS unit should also be renamed. This allows the BAMS MSC task to delete these polled files, freeing up disk space. Refer to "Updating the Poll Table" section on page 5-17 for information about specifying a *.finished suffix for polled files.

The billing system host must be set up to support such polling. Data files should be collected through FTP in binary transfer mode so that data file corruption is avoided. For automated storage, report creation and generation, and analysis, a number of solutions are available.

Setting Up Disk Monitoring Thresholds

The Mass Storage Control (MSC) task maintains disk space on the BAMS units according to percentage-based disk-space usage thresholds. Use the MSC-THRES tag ID at the system level to configure the MSC thresholds. Refer to Chapter 4, "Using MML Commands," for the MSC-THRES field names and values.

In the following example, the MSC task is set up to run at 15-minute intervals:

```
prov-ed:msc-thres:interval=15,iut=50000,put=60,plt=40,uut=80,mil=75
```

Note the following in this example:

- The inode upper threshold has been set at 50,000 inodes.
- The polling upper threshold is set at 60 percent of the available disk space, and the lower threshold is set at 40 percent of the disk space.
- The unpolled upper threshold has been set at 80 percent of the available disk space.
- A minor alarm level has been set at 75 percent.

The MSC task uses the directory name and a user-defined file pattern to distinguish polled files from unpolled files. The actual values depend on disk usage and fill factors that vary between BAMS locations. The system administrator must determine the appropriate values for a given location.

Calculating Disk Space Usage

When you establish the MSC task interval, you must make disk space usage estimates. The following examples are provided to help you estimate data processing and archive partitioning based on average calls per second (CPS). You can use the estimates listed to:

- Calculate how long data files will remain in the data directories before deletion by the MSC task
- Determine a backup procedure if the BAMS data is not being forwarded to another device
- Customize the MSC disk usage parameters

The estimates listed in Table 2-1 are based on an 18-GB disk partition and an output configuration for BAF, measurements, and ASCII. The estimates in Table 2-2 are based on an 18-GB disk partition and an output configuration for BAF and measurements. The estimates in Table 2-3 are based on a 12-GB disk partition.

Table 2-1 Data Disk Partitioning (Output Configuration for BAF, Measurements, and ASCII)

CPS	Disk Usage MB per Hour	Disk Usage (Percent) per Day	Disk Usage (Percent) per Week
20	32	4.2	29.4
40	64	8.4	58.8
60	96	12.6	88.2
80	128	16.8	100 (in 0.86 of a week)

Table 2-2 Data Disk Partitioning (Output Configuration for BAF and Measurements)

CPS	Disk Usage (MB) per Hour	Disk Usage (Percent) per Day	Disk Usage (Percent) per Week
20	7	0.9	6.5
40	14	1.9	13.0
60	21	2.8	19.6
80	28	3.7	26.1

Table 2-3 Archive Disk Partitioning

CPS	Disk Usage (MB) per Hour	Disk Usage (Percent) per Day	Disk Usage (Percent) per Week
20	36	7.2	50.4
40	72	14.4	100
60	108	21.6	100 (in 0.71 of a week)
80	144	28.8	100 (in 0.57 of a week)

Defining Conditions for MSC Alarms

Use the MSC-PARMS tag ID to define conditions that generate alarms or events, based on parameters such as file size, file type, and file age. Refer to Chapter 4, “Using MML Commands,” for a detailed description of the tag ID field names and values.

In the following example, the MSC task looks for polled files with a baf* prefix to determine whether or not to set off a minor alarm:

```
prov-ed:msc-parms:type=baf*,polled=1
```

Setting a File-Age Condition

You can set a file-age condition by setting an age value (in days) in the MSC-PARMS tag ID's agealarm field. In the following example, an alarm will occur once BAF files are 40 days old:

```
prov-ed:msc-parms:type="baf*",agealarm=40
```

Setting a File-Type Condition

You can add file-type conditions for files that the MSC task should monitor. To do so, use the **prov-add** command. Configure the MSC task to clean up unneeded files and to conserve disk space. A file already polled by the downstream billing system is an example of an unneeded file. A BAMS file-naming convention is used to indicate the file processing state and whether files have been polled or not. Thus file naming serves to indicate to the control task which files must be saved and which files are unneeded.

In the following example, the MSC task looks for files with a *.polled suffix.

```
prov-add:msc-parms:type="*.polled"
```

Setting Disk Usage Thresholds

You can add disk usage thresholds to be monitored by the MSC task. To do so, use the **prov-add** command. In the example below, the MSC task looks for unpolled files with the baf* prefix and generates a minor alarm if a file is older than 10 days. After 30 days, the MSC task deletes that file and sets a critical alarm.

```
prov-add:msc-parms:type="baf*",polled=2,agealarm=10,agelevel=2,delage=30,delalarm=0
```

Setting the Initial PGW CDR Sequence Number

To set the initial PGW CDR sequence number, perform the following steps:

Step 1 On each PGW host, change directory to /opt/CiscoMGC/var.

This directory contains the file .cdr.seq, which contains a number that identifies the last CDR sequence number written to the file.

Step 2 Issue the MML **chgno** command to set the sequence number you want BAMS to collect initially.

Example

```
mml:3>chgno::seqno1=2,seqno2=2
```

Note Within the **chgno** command expression, **seqno1** represents host1 in the POLL table; **seqno2** represents host2 in the POLL table.

For a description of the **chgno** command, see chgno—Change Number, page 4-7.

Note If you wish to collect older records, you can obtain the sequence numbers by issuing the command **ls -lt |cdr*** from the /opt/CiscoMGC/var/bam directory on each PGW host.

Setting the PGW Dynamic Update Mode

Based on a feature on PGW Release 9.4(1) and later, two call detail elements (CDEs), 4234 and 4235, are produced in the 1071 CDB. These two CDEs take a "snapshot" of the trunk group or sigpath on the PGW for the number of circuits in the trunk group (sigpath in nailed mode) and the number of unavailable circuits for the trunk group (sigpath in nailed mode). The 1071 CDB is triggered whenever a trunk group (sigpath) is added or updated, or when a manual MML **audit** command is issued on the PGW.

For a detailed description of the feature, CIC Availability Measurement CDB, refer to the feature module document of the same name, *CIC Availability Measurement CDB* under Cisco Media Gateway Controller Software Release 9 Feature Modules (Release 9.4(1) Feature Modules).

Note To start an audit of the Cisco PGW 2200, issue the following MML command:
sta-audit-cic

For a description of the **sta-audit-cic** MML command, see the document *CIC Availability Measurement CDB* under Cisco Media Gateway Controller Software Release 9 Feature Modules (Release 9.4(1) Feature Modules).

With these two CDEs produced on PGW 9.4(1), it is now possible for BAMS—when running in PGW Dynamic Update mode—to update the trunk groups and the number of circuits in a trunk group dynamically without end-user intervention. BAMS can:

- Update the number of circuits for each update on the PGW
- Report accurately the number of circuits out of service for a trunk group

The set-pgw-mode utility allows you to configure BAMS to process data from PGWRelease 9.4.1 and later (by default, BAMS does not run in PGW Dynamic Update mode).

Note The PGW Dynamic Update mode status can be determined with the **rtrv-ne** command. Refer to "rtrv-ne—Retrieve Network" section on page 4-21 for information about this command.

Processing Pre-9.4.1 Data

By default, the PGW_DYNAMIC_UPDATE value is FALSE; therefore, BAMS uses the number of available circuits in the Trunk Group table as the total number of circuits. Since the number of unavailable circuits is unreliable in this mode, BAMS does not attempt to track them.

Processing 9.4.1 (and Newer) Data

To process PGW 9.4.1 (and newer) data, you set PGW_DYNAMIC_UPDATE to TRUE. The following changes are enabled on BAMS:

- A backup of all current BAMS configuration tables is made.
- The structure of the Trunk Group table is modified by elimination of the circuits field, because BAMS will know the exact number of configured circuits and the number of unavailable circuits.
- A dynamic table of circuits is maintained.
- The BAMS algorithm for counting the number of circuits per trunk group is changed, thereby affecting measurements (see the "Suppressed Measurements" section on page 2-18).

- The **rtrv-circuits** MML command is enabled.

To configure BAMS to run in PGW Dynamic Update mode, perform the following steps:

Step 1 Log in as the BAMS user.

Step 2 Stop the application by entering the following commands:

```
$ cd /opt/CiscoBAMS/bin
$ mml
mml:sys> stp-softw::confirm
mml:sys> quit
```

Note The default UNIX prompt ($ or #) might vary with the shell program being used.

Step 3 Execute the set-pgw-mode utility. The following example shows user input for changing the default PGW_DYNAMIC_UPDATE value:

```
$ set-pgw-mode
The set-pgw-mode program sets the BAMS PGW_DYNAMIC_UPDATE
configuration to TRUE (1) or FALSE (0).

Do you want to continue (y/n)? y

Current value for PGW-DYNAMIC-FLAG is FALSE (*):

0) FALSE *
1) TRUE
q) Quit

New value (0, 1, or q(uit)): 1
You will be setting the BAMS PGW_DYNAMIC_UPDATE configuration to
TRUE (1).
A BAMS configuration rollback file will be created in case
set-pgw-mode is run once again and the value of PGW_DYNAMIC_UPDATE
is set back to FALSE.

This rollback file will reset the BAMS configuration back to the
date when the configuration was changed from FALSE to TRUE.
The rollback file being created at this time is named
/opt/CiscoBAMS/files/set-pgw-mode.200410071121.tar.
** This file should not be deleted **.

The BAMS PGW_DYNAMIC_UPDATE mode requires that the PGW is
running version 9.4.1 or later.

Is PGW version 9.4.1 or later (y/n)? y
Archiving current configuration tables ...
a /opt/CiscoBAMS/files/s01/DynTrunkGroup.CFDT 1K
a /opt/CiscoBAMS/files/s01/NailedConn.CFDT 1K
.
.
.
converting TrunkGroup.CFDT file
moving TrunkGroup.CTL to TrunkGroup.CTL_20041007112133.old
converting TrunkGroup.CTL file
********************************************************************************

User should execute the following to set up current environment:
$ . /opt/CiscoBAMS/bin/sym_defs
```

```
As an alternative, the user can log out and log in again to re-set
the shell environment.

*******************************************************************************

Conversion of BAMS PGW_DYNAMIC_UPDATE mode complete

PGW_DYNAMIC_UPDATE has been set to TRUE (1)

The rollback file /opt/CiscoBAMS/files/set-pgw-mode.200410071121.tar
has been created in case PGW_DYNAMIC_UPDATE is set back to FALSE
```

Step 4 Restart BAMS.

Note If you run the set-pgw-mode utility to return BAMS to its default setting, a warning is issued because the configuration will be rolled back to the date when the default setting was changed.

Suppressed Measurements

The following table illustrates which measurements are suppressed or not pegged based on the PGW_DYNAMIC_UPDATE value (that is, TRUE or FALSE).

Table 2-4 Suppressed Measurements

	PGW Dynamic Update = False			PGW Dynamic Update = True			
Measurement Name	**Pre-defined**	**Dynamic**	**MGCP**	**Pre-defined without 1071**	**Pre-defined with 1071**	**Dynamic**	**MGCP**
Call Attempts Incoming							
Call Attempts Outgoing			No Peg				No Peg
Outgoing Attempts Blocked			No Peg				No Peg
Failed Calls-Congestion							
Successful Calls Incoming							
Successful Calls Outgoing			No Peg				No Peg
Percent Trunk Group Usage Incoming		Suppress		Suppress		Suppress	
Percent Trunk Group Usage Outgoing		Suppress	No Peg	Suppress		Suppress	No Peg
Maintenance Duration per Trunk Group	Suppress	Suppress		Suppress		Suppress	
Total Traffic in Erlangs							

Table 2-4 Suppressed Measurements (continued)

	PGW Dynamic Update = False			PGW Dynamic Update = True			
Measurement Name	**Pre-defined**	**Dynamic**	**MGCP**	**Pre-defined without 1071**	**Pre-defined with 1071**	**Dynamic**	**MGCP**
Total Calls Terminated Normally							
Calls Terminated Abnormally							
Calls Terminated, Failed MGW or NAS							
Calls Rejected							
Calls Rejected, Unknown Dialed Number							
Calls Rejected, Other Reasons							
Overflow, Outgoing Attempts Blocked			No Peg				No Peg
Total Sum of Usage Pegs per Trunk Group							
Tandem Routing Attempts, Outgoing		Suppress	No Peg			Suppress	No Peg
Tandem Completions, Outgoing		Suppress	No Peg			Suppress	No Peg
Tandem Routing Attempts, Incoming		Suppress				Suppress	
Tandem Completions, Incoming		Suppress				Suppress	
Tandem Duration, Outgoing		Suppress				Suppress	
Tandem Duration, Incoming		Suppress				Suppress	
IC Destined Calls							
IC Destined Calls, No Circuit							
IC Usage							
Conversation Duration Ingress							
Conversation Duration Egress			No Peg				No Peg
Setup Duration Ingress							
Setup Duration Egress			No Peg				No Peg

Table 2-4 Suppressed Measurements (continued)

	PGW Dynamic Update = False			PGW Dynamic Update = True			
Measurement Name	**Pre-defined**	**Dynamic**	**MGCP**	**Pre-defined without 1071**	**Pre-defined with 1071**	**Dynamic**	**MGCP**
Teardown Duration Ingress							
Teardown Duration Egress			No Peg				No Peg
Call Routing I Peg							
Call Routing II Peg							
Call Routing III Peg							
Carrier Select No Indication		Suppress				Suppress	
Carrier Select PreSubscribed Not Input		Suppress				Suppress	
CarrierSelect PreSubscribed and Input		Suppress				Suppress	
CarrierSelect PreSubscribed with No Indication		Suppress				Suppress	
Carrier Id Code Not PreSubscribed but Input by Customer		Suppress				Suppress	
Successful H.323 Terminating Pegs							
Successful H.323 Originating Pegs							
Unsuccessful H.323 Terminating Pegs							
Unsuccessful H.323 Originating Pegs							
Successful ISUP Terminating Pegs							
Successful ISUP Originating Pegs							
Unsuccessful ISUP Terminating Pegs							
Unsuccessful ISUP Originating Pegs							

Table 2-4 Suppressed Measurements (continued)

Measurement Name	PGW Dynamic Update = False			PGW Dynamic Update = True			
	Pre-defined	Dynamic	MGCP	Pre-defined without 1071	Pre-defined with 1071	Dynamic	MGCP
ISDN Terminating Setup Message Delay Pegs							
ISDN Originating Setup Message Delay Pegs							
Number of Defined CICs during the Measurement Period		Suppress		Suppress		Suppress	
Average Number of Available CICs during the Measurement Period	Suppress	Suppress		Suppress		Suppress	

Provisioning BAMS

Overview

This chapter provides information and examples on how to provision the Billing and Measurements Server (BAMS), both at the system level and at the node level.

Note Before following the steps in this chapter, ensure that BAMS is correctly set up and installed. For more information, see Chapter 2, "Setup and Installation."

You must provision BAMS tables to specify such attributes as alarm parameters, polling information, node parameters, billing outputs, and disk-space parameters and thresholds. Before starting a provisioning session, read through the procedures at the beginning of this chapter to understand how to provision BAMS according to your system configuration and output requirements, and consult the provisioning examples in the rest of this chapter to assist you in deciding on the best provisioning strategy for your system.

Provisioning Commands

Use the MML **prov-add** (add), **prov-ed** (edit), and **prov-dlt** (delete) commands to modify BAMS tables (tag IDs). Use the **prov-dply** command to deploy your provisioning changes.Use the **prov-rtrv** (retrieve) command to view existing values in the tables, and to verify your provisioning changes.

Additionally, you can use the **prov-exp** (export) command to export existing values, for one table or for the entire system, to a file. You can use the **prov-cpy** (copy) command to copy one or more tables from one node to the current node.

Note For more information about MML commands, see Chapter 4, "Using MML Commands" or the MML online help. For more information about BAMS tables, see Chapter 5, "Using BAMS Tag IDs."

System-level Provisioning

To provision BAMS three system-level tables, perform the following steps:

Step 1 Log in as the BAMS user, and start an MML session.

Step 2 Execute the **set-node:sys:** command to ensure that you are at the system level.

Step 3 Execute the **prov-sta** command.

Step 4 Provision the Alarm Parameters table (ALM-PARMS tag ID), or use the default values.

Step 5 Provision the MSC Thresholds table (MSC-THRES tag ID), or use the default values.

Step 6 Provision the MSC Parameters (MSC-PARMS tag ID), or use the default values.

Note The default values in the MSC Parameters table are set fairly high, so evaluate your disk-space limitations and make any necessary changes to the values in this table. For more information, see the "Setting Up Disk Monitoring Thresholds" section on page 2-13.

Step 7 Execute the **prov-dply** command at the system level.

Node-level Provisioning

For every Cisco MGC to be polled by BAMS, you must define and provision a BAMS node for it. For example, if there are eight Cisco MGCs, you must provision eight BAMS nodes. You need to perform the following steps for each node. You can also create a multi-node batch file to provision more than one node at the same time. For more information, see the "Provisioning Commands and Examples" section on page 3-5.

To provision BAMS node-level tables for one BAMS node, perform the following steps:

Step 1 Execute the **set-node** command to go to the desired node (for example, Node1).

Step 2 Start a provisioning session with the **prov-sta** command.

Step 3 Execute the **set-nodename** command.

Step 4 Provision the Poll table (POLL tag ID).

Step 5 Provision the Node Parameters table (NODEPARMS tag ID).

Note If you do not specify values when you provision the Node Parameters table, default values are automatically assigned.

Step 6 Provision the Trunk Group table (TRUNKGRP tag ID).

Step 7 For Cisco SC2200 configurations, provision the Nailed Connection table (SIGPATH tag ID).

Billing Outputs Provisioning

You can provision BAMS to output billing data in BAF, 1110 Binary, NICS, or P01 formats.

BAF Output

To provision BAMS to produce BAF output, perform the following steps:

Step 1 Ensure that BAF output is enabled in the Node Parameters table (NODEPARMS tag ID).

Step 2 Provision the Switch Information table (SWITCHINFO tag ID).

Step 3 Provision the Map Type table (MAPTYPE tag ID).

Step 4 Provision the Zone Information table (ZONE-INFO tag ID).

Step 5 Provision the NPANXX table (NPANXX tag ID).

Step 6 Provision the Rating Type table (RATING-TYPE tag ID).

Step 7 Provision the Tollfree table (TOLLFREE tag ID).

Step 8 If applicable, provision the Country table (COUNTRY tag ID).

Note For more information about BAF output, see Chapter 6, “Configuring BAMS for BAF Output.”

1110 Binary Output

To provision BAMS to produce 1110 Binary output, perform the following steps:

Step 1 Ensure that 1110 Binary output is enabled in the Node Parameters table (NODEPARMS tag ID).

Step 2 Provision the 1110 Binary table (BIN1110 tag ID).

Note For more information about 1110 Binary output, see Chapter 10, “Configuring BAMS for 1110 Binary Output.”

NICS Output

To provision BAMS to produce NICS output, perform the following steps:

Step 1 Ensure that NICS output is enabled in the Node Parameters table (NODEPARMS tag ID).

Step 2 Provision the Trunk Group Prefix table (TKGPREFIX tag ID).

Note For more information about NICS output, see Chapter 9, “Configuring BAMS for NICS Output.”

P01 Output

To provision BAMS to produce P01 output, perform the following steps:

Step 1 Ensure that P01 output is enabled in the Node Parameters table (NODEPARMS tag ID).

Step 2 Provision the P01 Filter table (P01FILTER tag ID).

Note For more information about P01 output, see Chapter 8, "Configuring BAMS for P01 Output."

Threshold Crossing Alarms (TCA) Provisioning

To provision BAMS to produce threshold crossing alarms, perform the following steps:

Step 1 Provision the TCA-TBL table (TCA-TBL tag ID).

Note The Skip CDB and Skip CDE tables should only be provisioned as directed by Cisco, or left with the default values.

Deploying Node-level Provisioning

To deploy your configuration and output changes for one node, perform the following steps:

Step 1 Execute the **prov-dply** command at the node level.

Note If you are provisioning a redundant BAMS system, the **prov-dply** command automatically synchronizes the provisioning tables for that node on both units.

Step 2 Provision the next node. Return to the "Node-level Provisioning" section on page 3-2 and repeat the above steps.

Provisioning Commands and Examples

The **prov-add** (add), **prov-ed** (edit), and **prov-dlt** (delete) commands can be used in conjunction with any of the tag IDs shown below to modify BAMS tables. Use the **prov-rtrv** (retrieve) command to view existing values in the tables.

- ALM-PARMS
- BIN1110
- COUNTRY
- MAPTYPE
- MSC-PARMS
- MSC-THRES
- NODEPARMS
- NPANXX
- P01FILTER
- POLL
- RATE-EXC
- RATING-TYPE
- SIGPATH
- NODEPARMS
- SKIPCDB
- SKIPCDE
- SWITCHINFO
- TCA-TBL
- TKGPREFIX
- TOLLFREE
- TRUNKGRP
- ZONE-INFO

For more information about these tag IDs, see the "Tag IDs and Field Names" section on page 5-1.

Starting a Provisioning Session

In order to modify BAMS tables, you must first start a provisioning session with the **prov-sta** command. For example:

```
mml:sys>prov-sta::srcver=active,dstver=test10

Billing and Measurements Server - BAMS-00 2004-10-19 14:15:08
B  COMPLD
   ;
```

Note The value assigned to dstver must be a unique string, unless you use **confirm** to overwrite an existing configuration. When a provisioning session is started, a directory with the specified name is created. The directory is created under the directory name of the destination directory, /opt/CiscoBAMS/files/mml/*dirname*.

In the example above, the /opt/CiscoBAMS/files/mml/test10 directory was created. For more information about using the **prov-sta** command, see the "prov-sta—Provision Start" section on page 4-15.

Provisioning Multiple Records with a Single Command

You can provision multiple records with a single command if you use the **prov-add**, **prov-dlt**, or **prov-ed** command. Specify the parameter values by a range so that multiple records can be provisioned. Here are examples:

Example 1:

```
prov-add:SIGPATH:sigpath=0x140004,trunkgrp=805
```

This command adds 32 records to the SIGPATH table. Each record has sigpath=0x140004, trunkgrp=805.

Example 2:

```
prov-dlt:SIGPATH:sigpath=0x140001-0x140005
```

This command deletes five records from the SIGPATH table.

Example 3:

```
prov-ed:TRUNKGRP:trunkgrp=100-199,connection="T"
```

This command edits 100 records in the TRUNKGRP table. For each record where trunkgrp falls within the range 100–199, the connection parameter is set to "T."

Note To view the current settings in a table, use the **prov-rtrv** command. For details, see the "prov-rtrv—Provision Retrieve" section on page 4-14.

Deploying Changes

Use the **prov-dply** command when changes have been made and you must deploy them to the specified location. In the following example, a provisioning session is started and a change is made and then deployed.

Example:

```
mml:1>prov-sta::srcver=active,dstver=test10901,confirm

Billing and Measurements Server - BAMS-00 2004-10-08 16:30:25
B  COMPLD
   ;
mml:1>prov-add:NPANXX:npanxx=301829,lata=00001,zone=0000001

Billing and Measurements Server - BAMS-00 2004-10-08 16:31:09
B  COMPLD
```

```
   "NPANXX"
   ;
mml:1>prov-dply::srcver=test10901

Billing and Measurements Server - BAMS-00 2004-10-08 16:33:28
B  COMPLD
   ;
```

For more information about using the **prov-dply** command, see the "prov-rtrv—Provision Retrieve" section on page 4-14.

Making Changes Using Batch Files

Consult the examples in this section, which show sample provisioning sessions, if you want to provision BAMS using a batch file.

Note A BAMS provisioning batch file must have a **prov-sta** command at the beginning (right after the **set-node** command) and end with a **prov-dply** command. These commands denote the beginning and the end of the provisioning session.

The command-line instruction to execute a provisioning batch file is as follows:

```
mml -b filename [-o outputfilename]
```

BAMS does not monitor each command's execution status, but the results are placed in the MML session log file. You specify the log filename (that is, *outputfilename*) with the -o parameter, as shown above.

Using a batch file, you can provision the system as follows:

- System-level only
- One node only
- Multiple nodes (with or without the system level)

Note For ease of troubleshooting, it is recommended that you provision only one or two nodes at a time.

System-Level Example

```
set-node:sys:
prov-sta::srcver=new,dstver=bams_system,confirm
prov-add:ALM-PARMS:maxlines=10000,msgdisclvl=5,msgfwdlvl=3
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="baf*bin",
polled=2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="acc_h*",polled=1
,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="acc_d*",polled=1
,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="acc_r*",polled=1
,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="cdr*bin",
polled=2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="*finished",
polled=1,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="elkup*",polled=1
,alarm=6,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="esyn*",polled=1,
alarm=6,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="esem*",polled=1,
alarm=6,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="ebaf*",polled=1,
alarm=6,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="cdr*csv",polled=
2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="baf*csv",polled=
2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/CDR/archive",subdirs="Y",type="cdr*",
polled=1,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="ext*csv",
polled=2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="p01_*bin",
polled=2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="CDR*",polled=2,
alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-PARMS:active="Y",dir="/opt/CiscoBAMS/data",subdirs="Y",type="cdr*BIN",
polled=2,alarm=5,agealarm=30,agelevel=6,delage=30,delalarm=6
prov-add:MSC-THRES:interval=10,put=80,plt=60,uut=95,iut=950000,mil=80,mal=85,crl=90
prov-dply::
```

One-Node Example

```
set-node:1:
prov-sta::srcver=new,dstver=node1,confirm
set-nodename::name="node1"
prov-add:TCA-TBL:ID="global/0",egr-call-blkd=5,ttl-failed-cong=5,ttl-maint-use=10,igr-pct-
trk-use=90,egr-pct-trk-use=90,ttl-term-abnorm=10
prov-add:TRUNKGRP:trunkgrp=3,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=4,connection="D",orgnpa=703,trmnpa=703,circuits=48
prov-add:TRUNKGRP:trunkgrp=5,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=6,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=7,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=8,connection="D",orgnpa=703,trmnpa=703,circuits=48
prov-add:NODEPARMS:activate=1,statoutput=1,bafoutput=0,asciibafoutput=0,asciioutput=1,look
upinfo=1,bafinfo=0,dynamicaccumes=1,sup-zero-counts=1,interval-minutes=15,nailed-cfg=0,p01
output=0,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,bin1110output=0
prov-add:P01FILTER:answered=1,noanswer=1,busy=1,other=1
prov-add:POLL:host-name1="va-okinawa",rem-dir1="/opt/CiscoMGC/var/bam",file-prefix1="cdr*"
,file-suffix1=".bin",action="R",interval=5,timeout=5,maxtries=5
prov-add:SKIPCDB:id=1020
prov-dply::
```

Multiple-Node Example

```
set-node:1:
prov-sta::srcver=new,dstver=node1,confirm
set-nodename::name="node1"
prov-add:TCA-TBL:ID="global/0",egr-call-blkd=5,ttl-failed-cong=5,ttl-maint-use=10,igr-pct-
trk-use=90,egr-pct-trk-use=90,ttl-term-abnorm=10
prov-add:TRUNKGRP:trunkgrp=3,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=4,connection="D",orgnpa=703,trmnpa=703,circuits=48
prov-add:TRUNKGRP:trunkgrp=5,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=6,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=7,connection="D",orgnpa=703,trmnpa=703,circuits=24
prov-add:TRUNKGRP:trunkgrp=8,connection="D",orgnpa=703,trmnpa=703,circuits=48
prov-add:NODEPARMS:activate=1,statoutput=1,bafoutput=0,asciibafoutput=0,asciioutput=1,look
upinfo=1,bafinfo=0,dynamicaccumes=1,sup-zero-counts=1,interval-minutes=15,nailed-cfg=0,p01
output=0,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,bin1110output=0
prov-add:P01FILTER:answered=1,noanswer=1,busy=1,other=1
prov-add:POLL:host-name1="va-okinawa",rem-dir1="/opt/CiscoMGC/var/bam",file-prefix1="cdr*"
,file-suffix1=".bin",action="R",interval=5,timeout=5,maxtries=5
prov-add:SKIPCDB:id=1020
prov-dply::
set-node:2:
prov-sta::srcver=new,dstver=node2,confirm
set-nodename::name="node2_reston"
prov-add:TRUNKGRP:trunkgrp=3,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:TRUNKGRP:trunkgrp=4,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:TRUNKGRP:trunkgrp=5,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:TRUNKGRP:trunkgrp=6,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:TRUNKGRP:trunkgrp=7,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:TRUNKGRP:trunkgrp=8,connection="D",orgnpa=703,trmnpa=703,circuits=96
prov-add:NODEPARMS:activate=1,statoutput=1,bafoutput=0,asciibafoutput=0,asciioutput=1,look
upinfo=1,bafinfo=0,dynamicaccumes=1,sup-zero-counts=1,interval-minutes=15,nailed-cfg=0,p01
output=0,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,bin1110output=0
prov-add:P01FILTER:answered=1,noanswer=1,busy=1,other=1
prov-add:POLL:host-name1="va-reston",rem-dir1="/opt/CiscoMGC/var/bam",file-prefix1="cdr*",
file-suffix1=".bin",host-name2="va-herndon",rem-dir2="/opt/CiscoMGC/var/bam",file-prefix2=
"cdr*",file-suffix2=".bin",action="R",interval=5,timeout=5,maxtries=5
prov-add:SKIPCDB:id=1020
prov-dply::
```

Executing the Batch File

Once the batch file is created, it must be executed from outside of MML:

```
mml -b <filename>
```

Example:

```
$mml -b test_script
Copyright (C) 1998-2004, Cisco Systems, Inc.
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:01
B  COMPLD
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "POLL"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NODEPARMS"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
```

```
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "NPANXX"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
```

```
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:03
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "RATING-TYPE"
    ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "RATING-TYPE"
```

```
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "TRUNKGRP"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   "TRUNKGRP"
   ;
mml:1>Billing and Measurements Server - BAMS-00 2004-10-10 00:47:04
B  COMPLD
   ;
mml:1>/* Synchronizing remote BAMS ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/zoneinfo.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/TrunkGroup.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/alm_parms.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/country.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/maptype.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/nodeParm.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/NpaNxx.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/p01Filter.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/poll.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/RatingType.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/RatingException.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/NailedConn.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/skipcdb.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/skipcde.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/thresholds.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/s01/tollfree.CTL ... */
```

Using MML Commands

Introduction

This chapter provides general reference information to help you understand how to operate the Cisco Billing and Measurements Server (BAMS) using Man-Machine Language (MML) commands. See Chapter 4, "Using MML Commands" for the information about the tag IDs that you use in conjunction with MML commands to modify BAMS tables.

Note MML commands are differentiated from the surrounding text by being set in a Courier typeface. This makes them appear as they would on a computer screen. User input is shown in a bold font, and system responses are shown in a plain font. Text strings that you enter with MML commands are enclosed in quotes. Integers are shown without quotes.

Command Notation

The command notation follows the Backus-Naur conventions.

Command Syntax

The command syntax follows the Bellcore TL1 conventions.

Starting and Stopping BAMS

The following sections describe the system-level MML commands that you use to start and stop BAMS.

Starting the System

To start the system from the MML system prompt, use the sta-softw (start software) command.

Example:

```
sta-softw
or
start_system (from UNIX)
```

Stopping the System or OOS

To stop the system or bring it out-of-service (OOS), use the **stp-softw** (stop software) command.

Example:

stp-softw::confirm
or
stop_system (from UNIX)

Starting an MML Session

To start an MML session, and to get information about MML commands, perform the following steps:

Step 1 Log in to BAMS using your user ID and password.

When MML is started, the system prompt changes to `mml:sys>`.

The system is now ready to accept command-line instructions.

Note This is an alternate method for starting the command-line interface: In the BAMS bin directory, type **mml** in lowercase letters at the UNIX system prompt. Certain environment variables must be defined before MML can run. To define these variables, run **. sym_defs** in the profile of authorized users. The UNIX prompts and scripts follow the conventions for the UNIX Korn shell.

Step 2 Type **help**.

The system displays a list of available MML commands. One screen of information is presented. Press **Enter** to display additional information.

These MML commands are discussed in more detail in the following sections.

Help—Help Command

Purpose Displays a list of available commands or help on a specified command.

Format **help**[:*commandname*]

Description The Help function provides syntax and other general command information. At the node level, the Help function provides command information for the current node. At the system level, the Help function provides command information for the system as a whole, as well as for each node.

Example

```
mml:sys>help
Billing and Measurements Server - BAMS-00 2004-09-22 15:28:46

Available commands (in alphabetical order):
<command name>:?:                    Displays a list of possible tagIDs
<command name>:<tagID>:?             Displays a list of possible field
                                     names
clr-alm:<tagID>:msgnumber="<nnn>"[,msgtext="<message_text>"]
                                     Generates a clear message
dialog:<file name>:                  Redirects output to a file
get-nodenames::                      Lists node names from all nodes
h::[<number>[,<number>]]             Displays commands in the history
                                     buffer
help[:<command name>]                Displays a list of available commands
                                     or help on a specified command
prov-add:<tagID>:<field name1>=["][<value1>]["][range1-range2], ...
                                     <field nameN
>=["][<valueN>]["][range1-range2]
                                     Adds a new record to the ACRF format
                                     file
prov-cpy:[<tagid>|all]:srcnode=<1..8> | <nodename>,srcver=active |
                                     default | new | <dirname>
                                     Copies a table or tables from one node
                                     to the current node
prov-diff::                          Compares active parameters between
                                     units
prov-dlt:<tagID>:<field name1>=["]<value1>["][range1-range2], ...<field
                                     nameN>=["]<valueN>["][range1-range2]
                                     Deletes a record from the ACRF format
                                     file
prov-dply::[srcver=<dirname>]        Copies a provisioning session to the
                                     active parameters
prov-ed:<tagID>:<field name1>=["]<value1>["][range1-range2], ... <field
                                     nameN>=["]<valueN>["][range1-range2]
                                     Modifies one or more records that have
                                     already been inserted
prov-exp:[<tagID>|all]:dirname=<mml subdirectory>
                                     Exports a list of parameters for a
                                     specific table or all tables
prov-rtrv:<tagID>:                   Displays the contents of a record in
                                     ACRF format file
prov-sta::srcver=<dirname>,dstver=<dirname>[,confirm]
                                     Starts a provisioning session.
                                     <dirname>=active | default | new |
                                     dirname
prov-stp::[force]                    Stops the current provisioning session
prov-sync::                          Copies active parameters from the cur-
                                     rent unit to the remote unit
quit                                 Quit the session
r::[n]                               Executes a previously entered command
rtrv-alms                            Displays the active alarms
rtrv-configs                         Displays available configurations
rtrv-file::<file name>               Displays the contents of a Unix text
                                     file
rtrv-ne                              Displays attributes of the Network
                                     Element
rtrv-session                         Displays the current session
rtrv-softw:<process name>|all        Displays the status of system modules
rtrv-syslog::[cont]                  Displays syslog continuously
set-alm:taskname:msgnumber="<nnn>",level=<n>
                                     Generates an alarm message level n can
                                     be 0 | 1 | 2
set-node:<node #>|<node_name>|sys:Sets node number
sta-softw                            Starts the system application
stp-softw::confirm                   Stops the system application
```

Example

```
mml:7:node07>help
```

```
mml:7:node07>help
Billing and Measurements Server - BAMS-00 2004-09-22 12:07:35
Available commands (in alphabetical order):
<command name>:?:               Displays a list of possible tagIDs
<command name>:<tagID>:?        Displays a list of possible field names
chgno::[seqno1=<#>],[seqno2=<#>]Changes polling sequence numbers
clr-alm:<tagID>:msgnumber="<nnn>"[,msgtext="<message_text>"]
                                Generates a clear message
dialog:<file name>:             Redirects output to a file
get-nodenames::                 Lists node names from all nodes
h::[<number>[,<number>]]        Displays commands in the history buffer
help[:<command name>]           Displays a list of available commands or
                                help on a specified command
prov-add:<tagID>:<field name1>=["][<value1>]["][range1-range2], ...
                                <field nameN>=["][<valueN>]["]
                                [range1-range2]
                                Adds a new record to the ACRF format file
prov-cpy:[<tagid>|all]:srcnode=<1..8> | <nodename>,srcver=active |
                                default | new | <dirname>
                                Copies a table or tables from one node to
                                the current node
prov-diff::                     Compares active parameters between units
prov-dlt:<tagID>:<field name1>=["]<value1>["][range1-range2], ...<field
                                   nameN>=["]<valueN>["][range1-range2]
                                Deletes a record from the ACRF format
                                file
prov-dply::[srcver=<dirname>]   Copies a provisioning session to the
                                active parameters
prov-ed:<tagID>:<field name1>=["]<value1>["][range1-range2], ... <field
                                nameN>=["]<valueN>["]
                                [range1-range2]
                                Modifies one or more records that have
                                already been inserted
prov-exp:[<tagID>|all]:dirname=<mml subdirectory>
                                Exports a list of parameters for a
                                specific table or all tables
prov-rtrv:<tagID>:              Displays the contents of a record in ACRF
                                format file
prov-sta::srcver=<dirname>,dstver=<dirname>[,confirm]
                                Starts a provisioning session.
                                <dirname>=active | default | new |
                                dirname
prov-stp::[force]               Stops the current provisioning session
prov-sync::                     Copies active parameters from the current
                                unit to the remote unit
quit                            Quit the session
r::[n]                          Executes a previously entered command
rtrv-alms                       Displays the active alarms
rtrv-circuit:<all>:<trunkgrp>=”nnn”> Displays number of circuits/trunkgrp
rtrv-configs                    Displays available configurations
rtrv-file::<file name>          Displays the contents of a Unix text file
rtrv-ne                         Displays attributes of the Network
                                Element
rtrv-session                    Displays the current session
rtrv-softw:<process name>|all   Displays the status of system modules
rtrv-syslog::[cont]             Displays syslog continuously
set-alm:taskname:msgnumber="<nnn>",level=<n>
                                Generates an alarm message level n can be
                                0 | 1 | 2
set-node:<node #>|<node_name>|sys: Sets node number
set-nodename::name=<node name>  Sets the nodename of a node
sw-ovr                          Forces a manual rotation
```

Note Variables such as tag ID, field name, and so on, are shown in brackets.

Note Strings must be enclosed in quotes, for example "*string*". Anything enclosed in brackets, for example [:*command name*] is an option. Colons are used to separate major command components, for example the command name from the tag ID, and the tag ID from the field name. Fields are separated by commas. When modifying field name values, do not enter spaces before or after the equal sign (=) or commas.

command name:?:—Help on Tag IDs

Purpose Displays a list of tag IDs that can be used with the specified MML command.

Format *command name*`:?:`

Description The value of *command name* is any valid MML command.

Type **help** to display a list of available commands.

Example

```
mml:sys>prov-add:?:

Billing and Measurements Server - BAMS-00 2004-09-22 11:33:04
B  COMPLD
ALM-PARMS
MSC-PARMS
MSC-THRES
   ;
```

Note In the example above, the specified command displays three valid tag IDs at the system level prompt. When a node is set, the tag IDs are displayed, as shown in the example below:

```
mml:1:boston>prov-add:?:
Billing and Measurements Server - BAMS-00 2004-09-22 11:26:35
B  COMPLD
ZONE-INFO
TRUNKGRP
ALM-PARMS
BIN1110
COUNTRY
MAPTYPE
NODEPARMS
NPANXX
P01FILTER
POLL
RATING-TYPE
RATE-EXC
SIGPATH
SKIPCDB
```

```
SKIPCDE
SWITCHINFO
TCA-TBL
TOLLFREE
TKGPREFIX
  ;
```

command name:tagID:?—Help on Field Names

Purpose Obtains a list of field names for a specified table.

Format *command name*:*tagID*:?

Description The value of *command name* is any valid MML command.

The value of *tag ID* is any valid tag ID.

Type **command name:?** to obtain a listing of valid tag IDs.

Example

```
mml:sys>prov-add:MSC-THRES:?
Billing and Measurements Server - BAMS-00 2004-08-06 13:50:13
B  COMPLD
interval (numeric)
put (numeric)
plt (numeric)
uut (numeric)
iut (numeric)
mil (numeric)
mal (numeric)
crl (numeric)
   ;
```

Note In the example above, the specified command and tag ID display the field names for the MSC-THRES table. See the "Tag IDs and Field Names" section on page 5-1 for a description of the fields in each BAMS table.

MML Session Logs

The /opt/CiscoBAMS/data/logs directory contains daily MML session logs.

The log contains the time stamp of all user activity on BAMS and can be used for audits and troubleshooting.

MML Commands

To use the Man-Machine Language (MML) commands, define your user profile by typing **.sym_defs** in the /opt/CiscoBAMS/bin directory. Next, type **mml** to start the MML program. The UNIX system prompt changes to `mml:sys>` and allows you to enter MML commands, which are listed in the following sections.

chgno—Change Number

Purpose Changes the polling sequence numbers

Format `chgno::[seqno1=#],[seqno2=#]`

Description The value of *seqno1* is the value of the first polling sequence number.

The value of *seqno2* is the value of the second polling sequence number.

Note This command can be executed only at the node level.

Example

```
mml:3>chgno::seqno1=1,seqno2=1
Billing and Measurements Server - BAMS-00 2004-09-08 14:11:09
B  COMPLD
   ;
```

clr-alm—Clear Alarm

Purpose Clears a previously set alarm that was written to the alarm history file. These alarms are generally set for test purposes, for example, to test the alarm trapping. This command sends an alarm trap clear to the SNMP agent. (For details about setting alarms, see the "set-alm—Set Alarm" section on page 4-25.)

Format **clr-alm:***taskname*:**msgnumber=***"nnn"*[,msgtext=*"message_text"*]

Description The value of *taskname* is any valid BAMS task name.

The value of msgnumber can be any three-digit number. This number corresponds to the number of the message to be cleared. For ACC227 and POL402 alarms, you need to specify the full msgtext value to clear the alarm, because the message text for these alarms can vary. For ACC227, the msgtext includes the timestamp.

Example

```
mml:3>clr-alm:MSC:msgnumber="030"
mml:3>clr-alm:POL:msgnumber="402",msgtext="Cannot connect to unit
bams0-a"
mml:3>clr-alm:ACC:msgnumber="227",msgtext="TTL CALL ROUTING III, TG1, 0
is equal to 0 @ 20021102220000 UTC"
Billing and Measurements Server - BAMS-00 2004-09-08 14:11:09
B  COMPLD
   ;
```

Note For more information on the field names, refer to the "Updating the Poll Table" section on page 5-17.

dialog—Dialog

Purpose Redirects output to a log file.

Format **dialog**:*filename*:

Description The value of *filename* is a UNIX filename. This command is used to create a special log file that is stored in the /opt/CiscoBAMS/files/mml directory in the format *filename*.dlog when you quit MML. This log is different from the MML daily session log files stored in the /opt/CiscoBAMS/data/log/mml directory, which are prefixed with "mml."

Example

```
mml:sys>dialog:test:
Billing and Measurements Server - BAMS-00 2004-09-08 09:37:30
B  COMPLD
   ;
```

get-nodenames—Get Node Names

Purpose Lists all node names.

Format **get-nodenames::**

Description Displays a list of all defined node names. The system default values for the node names are node01, node02, … node08.

Example

```
mml:sys>get-nodenames::
Billing and Measurements Server - BAMS-00 2004-09-15 17:47:37
   node 1: "newyork"
   node 2: "sanfrancisco"
   node 3: "baltimore"
   node 4: "richmond"
   node 5: "washingtondc"
   node 6: ""
   node 7: "miami"
   node 8: ""
```

h—History

Purpose Displays commands in the history buffer.

Format **h**::[*number*[,*number*]]

Description The value of *number* is a numeric value. **h**:: is equivalent to **h**::**1,15**, which displays the last 15 commands in the history buffer. The first number option represents the *n*th command in the history list. The second number represents the number of commands in the list (a maximum of 15).

Example

```
mml:sys>h::1
Billing and Measurements Server - BAMS-00 2004-09-08 09:40:38
B  COMPLD
 1    h::1
   ;
```

prov-add—Provision Add

Purpose Adds a new record to the table specified by the tag ID.

Format **prov-add**:*tagID*:*field name1*=["]*value1*["], … *field nameN*=["]*valueN*["][range1 - range2]

Description The value of *tagID* is any valid tag ID.

The value of *field name1* is a first field name.

The value of *value1* is a first value.

The value of *field nameN* is an *N*th field name.

The value of *valueN* is an *N*th value.

The value of *range1* is a first value of a desired field range.

The value of *range2* is a second value of a desired field range.

Example

```
mml:2>prov-add:sigpath:sigpath=0x50-0x60,trunkgrp=0-16
Billing and Measurements Server - BAMS-00 2004-09-08 11:06:53
B  COMPLD
   "SIGPATH"
   ;
```

prov-cpy—Provision Copy

Purpose Copies a table or tables from one node to the current node.

Format **prov-cpy:**[<*tagid*>|all]:srcnode=<*1..8*> | <*nodename*>,srcver=active | default | new | <*dirname*>

Description

The value of *tagID* is any valid tag ID.

The value of *srcnode* is a node identifier (1 through 8).

The value of *nodename* must be a valid UNIX filename because this name is used to create a directory link in the /data directory.

The value of *srcver* is one of the following:

- Active—Use this value to copy the table from the source node's active configuration.
- Default—Use this value to copy the table from the source node's default configuration. This could be used if the **prov-sta** command does not use the srcver=default setting. In this case, use **prov-cpy** to copy the default table for one table only.
- New—Use this value to empty the table. This could be used if the **prov-sta** command does not use srcver=new. In this case, use **prov-cpy** to copy an empty table into the current session.

The value of *dirname* specifies that the table is copied from the source node's MML directory. This is the same as the **prov-sta** command's dstver argument.

Example

```
mml:sys>set-node:2:
Billing and Measurements Server - BAMS-00 2004-09-26 18:19:04
B  COMPLD
   ;
mml:2:node02>prov-sta::srcver=active,dstver=test,confirm
Billing and Measurements Server - BAMS-00 2004-09-26 18:19:21
B  COMPLD
   ;
mml:2:node02>prov-cpy:nodeparms:srcnode=1,srcver=active
Billing and Measurements Server - BAMS-00 2004-09-26 18:20:11
B  COMPLD
Do you want to copy control file(s) from /opt/CiscoBAMS/files/s01 to
the current provisioning directory /opt/CiscoBAMS/files/s02/mml/test ?
(y/n)
B  COMPLD
   ;
mml:2:node02>prov-rtrv:nodeparms:
Billing and Measurements Server - BAMS-00 2004-09-26 18:20:22
B  RTRV

"NODEPARMS:activate=1,statoutput=1,bafoutput=1,asciibafoutput=1,asciiou
tput=2,lookupinfo=1,bafinfo=1,dynamicaccumes=0,sup-zero-counts=0,interv
al-minutes=5,nailed-cfg=1,p01output=0,p01prefix="p01_",enable-h323=0,ex
tasciioutput=0,nicsoutput=0,bin1110output=0
   ;
mml:2:node02>prov-dply::
Billing and Measurements Server - BAMS-00 2004-09-26 18:23:06
B  COMPLD

   /* The measurements interval (interval-minutes) has been modified in
the NODEPARMS table. */
   /* This node will be dynamically restarted for the local unit if it
is currently running. */   ;
```

prov-diff—Provision Compare

Purpose Compares active parameters between BAMS units.

Format **prov-diff**::

Description No variables need to be specified.

Example

```
mml:sys>prov-diff::
Billing and Measurements Server - BAMS-00 2004-09-21 17:38:28
B  COMPLD
   /* Table ALM-PARMS matches for both BAMS units */
   /* Table MSC-PARMS matches for both BAMS units */
   /* Table MSC-THRES matches for both BAMS units */
   ;
mml:sys>set-node:1:
Billing and Measurements Server - BAMS-00 2004-09-21 09:45:20
B  COMPLD
   ;
mml:1:node01>prov-diff::
Billing and Measurements Server - BAMS-00 2004-09-21 09:45:24
B  COMPLD

   /* Table ZONE-INFO matches for both BAMS units */
   /* Table TRUNKGRP matches for both BAMS units */
   /* Table ALM-PARMS matches for both BAMS units */
   /* Table BIN1110 matches for both BAMS units */
   /* Table COUNTRY matches for both BAMS units */
   /* Table MAPTYPE matches for both BAMS units */
   /* Table NODEPARMS matches for both BAMS units */
   /* Table NPANXX matches for both BAMS units */
   /* Table P01FILTER matches for both BAMS units */
   /* Table POLL matches for both BAMS units */
   /* Table RATING-TYPE matches for both BAMS units */
   /* Table RATE-EXC matches for both BAMS units */
   /* Table SIGPATH matches for both BAMS units */
   /* Table SKIPCDB matches for both BAMS units */
   /* Table SKIPCDE matches for both BAMS units */
   /* Table SWITCHINFO matches for both BAMS units */
   /* Table TCA-TBL matches for both BAMS units */
   /* Table TOLLFREE matches for both BAMS units */
   /* Table TKGPREFIX matches for both BAMS units */
   ;
```

Note If two (or more) tables are not identical across units, you will receive a message similar to the following:

```
/* Table <table name> DOES NOT MATCH across BAMS units. Examine this table on both BAMS
units. Execute prov-sync from correct unit. */
```

After you determine which BAMS unit has the correctly configured table, execute the prov-sync command for the appropriate node of the correctly configured BAMS unit. For more information, see "prov-sync—Provision Synchronize" section on page 4-18.

prov-dlt—Provision Delete

Purpose Deletes a record from the table specified by the tag ID.

Format **prov-dlt**:*tagID*:*field name1*=["]*value1*["], … *field nameN*=["]*valueN*["][range1-range2]

Description The value of *tagID* is any valid tag ID.

The value of *field name1* is a first field name.

The value of *value1* is a first value.

The value of *fieldnameN* is an *N*th field name.

The value of *valueN* is an *N*th value.

The value of *range1* is a first value of a desired field range.

The value of *range2* is a second value of a desired field range.

Example

```
mml:2>prov-dlt:sigpath:sigpath=0x50-0x60,trunkgrp=0-16,
Billing and Measurements Server - BAMS-00 2004-09-08 11:25:03
B  COMPLD
   "SIGPATH"
   ;
```

prov-dply—Provision Deploy

Purpose Implements changes specified during an active provisioning session.

Format **prov-dply**::[**srcver=***dirname*]

Description The value of *dirname* is the directory name; if you are in a provisioning session, you can deploy another session from within the active directory. **prov-dply**:: copies the changes in the current provisioning session to the active session (directory). If a directory name is specified with the **prov-dply** command, the files in that directory are copied to the active session.

Note When you execute a **prov-dply** command in a redundant BAMS configuration, all the tables at the same level (node or system) are synchronized with the remote unit (i.e., they are transferred to that unit). This is the same behavior as when you execute the **prov-sync** command. The following message indicates that an MML table has been successfully transferred to the remote BAMS unit:

```
/* Transferring file <filename> ... */
```

If an SXF error message is displayed instead, the MML tables may not be synchronized if this table was edited in this session. For more information, refer to the specific message in the "Send File Transfer Task Messages" section on page A-64.

Example

```
mml:2>prov-dply::srcver=test
Billing and Measurements Server - BAMS-00 2004-09-08 11:29:15
B  COMPLD
   ;
```

Note When you execute **prov-dply** at the node level, if the values to be provisioned for the NODEPARMS parameters nailed-cfg and interval-minutes are different from those in the current active configuration, the processing tasks for this node automatically restart so that these parameter changes can take effect. The restart of the node tasks generates MGR801 alarms, which indicate that certain processes are being terminated and restarted. Since changing these parameters has a significant effect on how the data is processed, the measurements data produced might show some incongruities with previously generated measurements data.

prov-ed—Provision Edit

Purpose Modifies one or more provisioning records in the table specified by the tag ID.

Format

```
prov-ed:tagID:field name1=["]value1["], … field
name>=["]valueN["][range1-range2]
```

Description The value of *tagID* is any valid tag ID.

The value of *field name1* is a first field name.

The value of *value1* is a first value.

The value of *fieldnameN* is an *N*th field name.

The value of *valueN* is an *N*th value.

The value of *range1* is a first value of a desired field range.

The value of *range2* is a second value of a desired field range.

Note Execute the **prov-rtrv** command first, to see the existing configuration.

Example

```
mml:2>prov-ed:sigpath:sigpath=0x50-0x61,trunkgrp=0-17
Billing and Measurements Server - BAMS-00 2004-09-08 11:33:47
B  COMPLD
   "SIGPATH"
   ;
```

prov-exp—Provision Export

Purpose Exports a list of parameters for a specific table or all tables. The parameters are the field names and values of the table or tables.

This command can be executed at the system or node level. When executed at the system level, the command generates a file that contains all of the commands to provision the system and all of the nodes. When executed at the node level, the command generates a file that contains all of the commands to provision the indicated node.

Format **prov-exp**:[*tagID*|**all**]:**dirname**=*mml subdirectory*

Description The value of *tagID* is any valid tag ID.

The value of *dirname* is the directory name that is created under the /opt/CiscoBAMS/files/mml directory. The **all** option exports all tag IDs.

Example

```
mml:3>prov-exp:sigpath:dirname=testbackup
Billing and Measurements Server - BAMS-00 2004-09-08 11:51:11
B  COMPLD
   ;
```

prov-rtrv—Provision Retrieve

Purpose Retrieves the configuration for the specified tag ID.

Format **prov-rtrv**:*tagID*:

Description The value of *tagID* is any valid tag ID.

Example

```
mml:3>prov-rtrv:sigpath:
Billing and Measurements Server - BAMS-00 2004-09-08 11:54:15
B  RTRV
   "SIGPATH:sigpath=0x40,trunkgrp=16
   "SIGPATH:sigpath=0x41,trunkgrp=16
   "SIGPATH:sigpath=0x42,trunkgrp=16
   "SIGPATH:sigpath=0x43,trunkgrp=16
   "SIGPATH:sigpath=0x44,trunkgrp=16
   "SIGPATH:sigpath=0x45,trunkgrp=16
   "SIGPATH:sigpath=0x46,trunkgrp=16
   "SIGPATH:sigpath=0x47,trunkgrp=16
   "SIGPATH:sigpath=0x48,trunkgrp=16
   "SIGPATH:sigpath=0x49,trunkgrp=16
   "SIGPATH:sigpath=0x4a,trunkgrp=16
   "SIGPATH:sigpath=0x4b,trunkgrp=16
   "SIGPATH:sigpath=0x4c,trunkgrp=16
   "SIGPATH:sigpath=0x4d,trunkgrp=16
   "SIGPATH:sigpath=0x4e,trunkgrp=16
   "SIGPATH:sigpath=0x4f,trunkgrp=16
   "SIGPATH:sigpath=0x50,trunkgrp=16
   ;
```

prov-sta—Provision Start

Purpose Starts a provisioning session.

Format **prov-sta**::**srcver=***dirname*,**dstver=***dirname*[,**confirm**]

Description In order to modify the contents of BAMS tables, you must execute the **prov-sta** command to start a provisioning session.

You specify the source of the tables to be used in the provisioning session with the srcver parameter. The value of *dirname* is the name of the source or destination directory. Setting srcver=new sets the configuration tables to empty. Setting srcver=default sets the configuration tables to the default values. Setting srcver=active retrieves the currently provisioned tables.

The dstver parameter defines the name of the provisioning session. This parameter creates a separate directory containing the tables for the provisioning session.

Note You must specify the **confirm** command option if the destination directory (dstver) already exists.

Once you start a provisioning session by executing the **prov-sta** command, you can modify the BAMS tables by using the MML editing commands **prov-add, prov-dlt,** and **prov-ed,** in conjunction with tag IDs.

Once all edits have been made, you execute the **prov-dply** command to make the tables of the provisioning session the current active tables at either the BAMS system level or node level (defined by the **set-node** command). If you do not wish to deploy the edited tables, use the **prov-stp** command to quit the provisioning session.

Example

```
mml:sys>prov-sta::srcver=active,dstver=test,confirm
Billing and Measurements Server - BAMS-00 2004-09-08 14:32:49
B  COMPLD
   ;
```

Note When you execute the **prov-sta** command, it creates a working directory with the same name as the dstver parameter value. These directories are saved so that a user can retrieve tables from this directory at a later time. (In this case the user would execute **prov-sta** where the srcver is the name of a previously created dstver value.) Over time, these MML working directories can accumulate if many different values are used for the dstver parameter. A maintenance script, clean_mml.sh, should be executed periodically on the BAMS unit so that the MML working directories do not use up too much disk space. This script prompts you, at the system level and for each BAMS node, to delete an MML directory or not. The newest working directory of each node is automatically saved.

To run this script, log in as the BAMS user and execute the following:

```
$ clean_mml.sh
** clean_mml.sh will clean up the mml working directories.  The latest
** directory for each node will automatically be saved.
** Examining mml directories in BAMS system level:
Remove directory /opt/CiscoBAMS/files/mml/bams_system ?(y/n) y
Newest directory /opt/CiscoBAMS/files/mml/bams_system_vers2
automatically saved.
** Examining mml directories in BAMS node 01:
Remove directory /opt/CiscoBAMS/files/s01/mml/node1_config ?(y/n) n
Remove directory /opt/CiscoBAMS/files/s01/mml/test_config ?(y/n) y
Newest directory /opt/CiscoBAMS/files/s01/mml/jims_config
automatically saved.
** Examining mml directories in BAMS node 02:
Remove directory /opt/CiscoBAMS/files/s02/mml/jims_config ?(y/n) y
Newest directory /opt/CiscoBAMS/files/s02/mml/bobs_config
automatically saved.
** Examining mml directories in BAMS node 03:
Remove directory /opt/CiscoBAMS/files/s03/mml/test ?(y/n) y
Newest directory /opt/CiscoBAMS/files/s03/mml/joes_config
automatically saved.
** Examining mml directories in BAMS node 04:
** Examining mml directories in BAMS node 05:
** Examining mml directories in BAMS node 06:
** Examining mml directories in BAMS node 07:
** Examining mml directories in BAMS node 08:
** clean_mml.sh finished.
```

prov-stp—Provision Stop

Purpose Stops the current provisioning session.

Format **prov-stp::[force]**

Description You use this command if you do not want to deploy the changes made in your provisioning session to the active session. Use the force argument to force a provisioning session to stop (that is, when you must clear a lock on a prov-sta session).

Example

```
mml:3>prov-stp::
Billing and Measurements Server - BAMS-00 2004-09-08 11:55:57
B  COMPLD
   ;
```

prov-sync—Provision Synchronize

Purpose Copies active parameters from the current unit to the remote unit.

Format **prov-sync::**

Description No arguments are required.

Example

```
mml:sys>prov-sync::
Billing and Measurements Server - BAMS-00 2004-09-08 16:56:18

   /* Transferring file /opt/CiscoBAMS/files/alm_parms.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/msc_parm.CTL ... */

   /* Transferring file /opt/CiscoBAMS/files/msc.CTL ... */
B  COMPLD
   ;
```

quit—Quit

Purpose Quits the MML session.

Format **quit**

Description none

Example

```
mml:sys>quit
Billing and Measurements Server - BAMS-00 2004-09-22 14:12:03
```

r—Repeat

Purpose Executes a previously entered command.

Format **r**::[*n*]

Description Here *n* is a numeric value.

Example

```
mml:3>r::2
Billing and Measurements Server - BAMS-00 2004-09-08 11:58:36
B  COMPLD
   ;
mml:3>prov-rtrv:sigpath:
Billing and Measurements Server - BAMS-00 2004-09-08 11:58:36
B  RTRV
   ;
```

rtrv-alms—Retrieve Alarms

Purpose Displays the active alarms or events at the system level (for system-level alarms) or at the node level (for node-level alarms).

Format **rtrv-alms**

Description This command retrieves only clearable alarms currently maintained in the alarm history file and defined with the alarmslist configuration. It does not retrieve event information generated, for example, by table lookup and other configuration errors. No tag IDs are required, just the command verb.

Example

```
mml:3>rtrv-alms
Billing and Measurements Server - BAMS-00 2004-09-08 13:20:09
B  RTRV
08/05/04 13:20:03 *C MSC030: Alarm/Event test message
   ;
B  COMPLD
   ;
```

rtrv-circuit—Retrieve Circuits

This command is available only at node level, and only on systems that have been configured in PGW Dynamic Mode (that is, where PGW_DYNAMIC_UPDATE=TRUE). For more information, see the "Setting the PGW Dynamic Update Mode" section on page 2-16.

Purpose Displays the available circuits (at node level).

Format **rtrv-circuit**

Description No tag IDs are required, just the command verb.

Example

```
mml:1:node01>rtrv-circuit:all:
Billing and Measurements Server - BAMS-00 2004-09-28 15:43:36
B  RTRV
   "CKTS:trunkgrp=1,circuits=124"
   "CKTS:trunkgrp=2,circuits=124"
   "CKTS:trunkgrp=3,circuits=124"
   "CKTS:trunkgrp=4,circuits=96"
   "CKTS:trunkgrp=5,circuits=96"
   "CKTS:trunkgrp=6,circuits=96"
   "CKTS:trunkgrp=7,circuits=96"
   "CKTS:trunkgrp=8,circuits=96"
   ;
mml:1:node01>rtrv-circuit::trunkgrp="2"
Billing and Measurements Server - BAMS-00 2004-09-28 15:43:43
B  RTRV
   "CKTS:trunkgrp=2,circuits=124"
   ;
mml:1:node01>rtrv-circuit::trunkgrp=7
Billing and Measurements Server - BAMS-00 2004-09-28 15:43:48
B  RTRV
   "CKTS:trunkgrp=7,circuits=96"
   ;
```

rtrv-configs—Retrieve Configuration

Purpose Displays the available configurations.

Format **rtrv-configs**

Description The configurations are maintained in the /opt/CiscoBAMS/files/mml subdirectory. When you start a provisioning session, your changes are written to this directory. You can use this command to retrieve the current configuration. No tag IDs are required, just the command verb.

Example

```
mml:3>rtrv-configs
Billing and Measurements Server - BAMS-00 2004-09-12 13:23:08
B  COMPLD

   /* Current session: ACTIVE */

   /* Available Configurations: */

   /* /opt/CiscoBAMS/files/s03/mml/bams_system,  Mon Jul 12 11:45:48 2004
 */
;
```

rtrv-file—Retrieve File

Purpose Displays the contents of a UNIX text file.

Format **rtrv-file**::*filename*

Description The value of *filename* is any existing filename. You need to specify the full UNIX filename that includes the directory path, unless the file exists in the current directory (/opt/CiscoBAMS/bin).

Example

```
mml:sys>rtrv-file::collect_defs
Billing and Measurements Server - BAMS-00 2004-09-08 13:42:53
B  RTRV
#
# Collection variables
#

REDUNDANCY_MODE=1;                              export REDUNDANCY_MODE
   ;
```

rtrv-ne—Retrieve Network

Purpose Displays attributes (for example, polling status) of the network elements. It also displays the current setting of the PGW_DYNAMIC_UPDATE flag at both the system and the node levels.

Note There is an 8-second interval between system start and POL task start. Within this interval, the status shows ACTIVE or STANDBY, even though the POL task has not started. Once the POL task starts, the polling status is accurate.

Format **rtrv-ne**

Description No tag IDs are required, just the command verb.

Example

```
mml:sys>rtrv-ne
Billing and Measurements Server - BAMS-00 2004-10-07 11:15:03
B  RTRV
   "Type: BAMS"
   "Hardware platform: sun4u sparc SUNW,Ultra-250"
   "Vendor: "Cisco Systems, Inc.""
   "Location: Billing and Measurements Server - BAMS-00"
   "Version: "3.20""
   "System Status: OOS"
   "Polling Status[s01 (test01)]: OOS"
   "s01 (test01) Node Status: ACTIVATED"
   "Polling Status[s02 (test02)]: OOS"
   "s02 (test02) Node Status: ACTIVATED"
   "Polling Status[s03 (node03)]: OOS"
   "s03 (node03) Node Status: DEACTIVATED"
   "Polling Status[s04 (node04)]: OOS"
   "s04 (node04) Node Status: DEACTIVATED"
   "Polling Status[s05 (node05)]: OOS"
   "s05 (node05) Node Status: DEACTIVATED"
   "Polling Status[s06 (node06)]: OOS"
   "s06 (node06) Node Status: DEACTIVATED"
   "Polling Status[s07 (node07)]: OOS"
   "s07 (node07) Node Status: DEACTIVATED"
    "Polling Status[s08 (node08)]: OOS"
   "s08 (node08) Node Status: DEACTIVATED"
   "Local hostname: "
   "Unit ID: BAMS-00"
   "BAMS-00 hostname: "
   "BAMS-01 hostname: "
   "Mode: Simplex"
   "PGW Dynamic Update: False"
   ;
```

Example

```
mml:node01>rtrv-ne
Billing and Measurements Server - BAMS-00 2004-09-06 14:43:17
B  RTRV
   "Type: BAMS"
   "Hardware platform: sun4u sparc SUNW,Ultra250"
   "Vendor: "Cisco Systems, Inc.""
   "Location: Billing and Measurements Server - BAMS-00"
   "Version: "3.20""
   "Polling Status[s01 (node01)]: OOS"
   "s02 (node02) Node Status: ACTIVATED"
   "VSC1 seqno: 000001"
   "VSC2 seqno: 000001"
   "Local hostname: "
   "Unit ID: BAMS-00"
   "BAMS-00 hostname: "
   "BAMS-01 hostname: "
   "Measurement Interval: 10 minutes"
   "VSC Configuration: Switched"
   "PGW Dynamic Update: False"
   ;
```

rtrv-session—Retrieve Session

Purpose Displays the current provisioning session.

Format **rtrv-session**

Description No tag IDs are required, just the command verb.

Example

```
mml:3>rtrv-session
Billing and Measurements Server - BAMS-00 2004-09-08 13:56:52
B  COMPLD

   /* Current Session: active */
   ;
```

rtrv-softw—Retrieve Software

Purpose Displays the status of software processes.

Format **rtrv-softw**:[*process name*|**all**]

Description The value of *process name* is any valid task name.

Example

```
mml:3>rtrv-softw:all
Billing and Measurements Server - BAMS-00 2004-08-06 14:45:00
B  RTRV
"MGR-03 RUNNING"
"ALM-03 RUNNING"
"POL-03 RUNNING"
"FMT-03 RUNNING"
"COR-03 RUNNING"
"AGB-03 RUNNING"
"ASC-03 RUNNING"
"EXT-03 RUNNING"
"CTB-03 RUNNING"
"P01-03 RUNNING"
"NIC-03 RUNNING"
"BIN-03 RUNNING"
"ACC-03 RUNNING"
B  COMPLD
   ;
```

Example

```
mml:sys>rtrv-softw:all
Billing and Measurements Server - BAMS-00 2004-09-22 14:25:18

NODE: system
------------
B  RTRV
"MGR-00 RUNNING"
"ALM-00 RUNNING"
"bamsP3MIBagt-00 RUNNING"
"NNL-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MGR-00 RUNNING"
"MSCmain-00 RUNNING"
"mscParmP3MIBagt-00 RUNNING"
"mscP3MIBagt-00 RUNNING"
"pollP3MIBagt-00 RUNNING"
"nodeParmP3MIBagt-00 RUNNING"

NODE: 1
--------
B  RTRV
"MGR-01 RUNNING"
"ALM-01 RUNNING"
"POL-01 RUNNING"
"FMT-01 RUNNING"
"COR-01 RUNNING"
"AGB-01 RUNNING"
"ASC-01 RUNNING"
"EXT-01 RUNNING"
"CTB-01 RUNNING"
"P01-01 RUNNING"
"NIC-01 RUNNING"
"BIN-01 RUNNING"
"ACC-01 RUNNING"
B  COMPLD
```

rtrv-syslog—Retrieve Syslog

Purpose Displays the system log (syslog) continuously.

Format **rtrv-syslog**::[**cont**]

Description No tag IDs are required, just the command verb.

Note The **cont** command option displays the system log continuously. Press **Ctrl-C** to escape out of the display.

Example

```
mml:sys>rtrv-syslog::
Billing and Measurements Server - BAMS-00 2004-09-08 17:22:39
08/06/04 23:35:51    MSC233: Executing system command: '/opt/CiscoBAMS/bin/msc_
08/06/04 23:35:51    MSC233:+pre_del.sh ebaf*'
08/06/04 23:35:51    MSC233: Executing system command: '/opt/CiscoBAMS/bin/msc_
08/06/04 23:35:51    MSC233:+pre_del.sh ext*csv'
08/06/04 23:35:51    MSC233: Executing system command: '/opt/CiscoBAMS/bin/msc_
08/06/04 23:35:51    MSC233:+pre_del.sh p01_*bin'
08/06/04 23:35:51    MSC902: Block: total=493688 free=92923 used=400765 utiliza
08/06/04 23:35:51    MSC902:+tion: 81% partition: /opt
08/06/04 23:35:51    MSC904: Inode: total=256576 free=254624 used=1952 utilizat
08/06/04 23:35:51    MSC904:+ion: 0% partition: /opt
08/06/04 23:35:51    MSC906: Inodes matching path(s) and prefix(es) in partitio
08/06/04 23:35:51    MSC906:+n /opt: 28
08/06/04 23:45:50    MSC001: MSC version W3.02 (CISCO) cleaning disk
```

set-alm—Set Alarm

Purpose Generates an alarm message. Use this command to set a test alarm that is written to the syslog. (For details about clearing alarms, see the "clr-alm—Clear Alarm" section on page 4-7.)

Format **set-alm**:*taskname*:**msgnumber**=*nnn*,**level**=*n*

Description The value of *taskname* is any valid BAMS task name.

The value of *nnn* is a three-character message number.

The value of *n* is a valid single-character threshold level (0, 1, or 2).

Note This command works only for alarms maintained in the alarmslist.CTL file in the /opt/CiscoBAMS/files directory.

Example

```
mml:3>set-alm:MSC:msgnumber="030",level=0
Billing and Measurements Server - BAMS-00 2004-09-08 14:13:13
B  COMPLD
   ;
```

set-node—Set Node

Purpose Sets the node number. Use this command to change from the system level to the node level, or to change from one node to another node.

Format **set-node:*node#*|*nodename*|sys:**

Description The value of *node#* is any integer between 1 and 8.

The value of *nodename* is any valid node name.

sys displays the system level.

Note When 1 is specified as the node number, the MML prompt changes from mml:sys> to mml:1>. Exit out of any provisioning sessions before changing the node number. The prompt displays the node name when it is defined. The prompt always displays the node number, except when system is specified.

Example

```
mml:sys>set-node:1:
Billing and Measurements Server - BAMS-00 2004-09-08 14:45:27
mml:1:node01>
```

set-nodename—Set Node Name

Purpose Sets the node name for a node.

Format **set-nodename:*node#*:name=*nodename***

Description The value of *node#* is any integer between 1 and 8.

The value of *nodename* must be a valid UNIX filename, because this name is used to create a directory link in the data directory.

This command must be executed within an MML editing session at the node level.

Note You can use the **set-nodename** command to modify any existing node names. You cannot use the command to modify the system name.

Example

```
mml:sys>set-nodename:1:name="boston"
Billing and Measurements Server - BAMS-00 2004-09-08 14:45:27
mml:1>
```

sta-softw—Start Software

Purpose Starts the system application.

Format **sta-softw**

Description No tag IDs are required, just the command verb.

Note An alternative way to start the software is to use the UNIX **start_system** command, which is found in the /opt/CiscoBAMS/bin directory.

Example

```
mml:sys>sta-softw
Billing and Measurements Server - BAMS-00 2004-09-08 14:14:41
B  COMPLD
   ;
```

stp-softw—Stop Software

Purpose Stops the system application.

Format **stp-softw::confirm**

Description No tag IDs are required, just the command verb.

Note You can execute this command only at the system level.

Note An alternative way to stop the software is to use the UNIX **stop_system** command, which is found in the /opt/CiscoBAMS/bin directory.

Example

```
mml:sys>stp-softw::confirm
Billing and Measurements Server - BAMS-00 2004-09-08 14:16:34
B  COMPLD

Terminating BAMS

waiting for MGR to shut down...
;
```

sw-ovr—Switch Over

Purpose Forces a manual rotation of the BAMS unit designated to poll the VSC. (You can execute this command only on the active polling unit.)

Format **sw-ovr**

Description You can use the **sw-ovr** command to rotate polling from the active BAMS unit to the standby unit in the event of a switch failover. The command is executed immediately on the local BAMS unit. On the remote BAMS unit, the rotation takes effect at the next poll attempt. Poll attempts are typically set at 5-minute intervals.

Caution Do not execute this command while polling is taking place; it will fail.

No tag IDs are required, just the command verb.

Note Successful execution of this command causes a POL105 message to be written to the syslog (see Appendix A, "Troubleshooting Cisco BAMS").

Example

```
mml:sys>sw-ovr
Billing and Measurements Server - BAMS-00 2004-09-22 13:26:19
B  COMPLD
   ;
```

CHAPTER 5

Using BAMS Tag IDs

Introduction

This chapter provides general reference information to help you understand how to update BAMS tables by using BAMS tag IDs.

For information about the MML commands that you use with the tag IDs, see Chapter 4, "Using MML Commands."

Note MML commands, tag IDs and field names are differentiated from the surrounding text by being set in a Courier typeface. This makes them appear as they would on a computer screen. User input is shown in a bold font, and system responses are shown in a plain font. Text strings that you enter with MML commands are enclosed in quotes. Integers are shown without quotes.

Command Notation

The command notation follows the Backus-Naur conventions.

Command Syntax

The command syntax follows the Bellcore TL1 conventions.

Tag IDs and Field Names

Tag IDs are aliases for BAMS tables. You use them in conjunction with MML commands to modify or delete table records or data. Some tag IDs contain defined parameter sets used to provision the BAMS tables. The field default values are recommended values, unless noted otherwise.

Note Tag IDs and field names are not case sensitive. However, values are case sensitive.

Updating the Alarm Parameters Table

Use the ALM-PARMS tag ID to update the Alarm Parameters table. Use the maxlines field to set the size of the system log (syslog) file. The default value is 10,000 lines. When the maxlines threshold is surpassed, BAMS begins deleting lines from the syslog, starting with the oldest entries.

Note BAMS creates oldlog and oldloga as backup system logs by default. You can configure the number of backup logs for a higher value (a maximum of 26) in the sym_defs file. The number of logs you create affects the available disk space.

Use the msgdisclvl field to set the level (for example, minor, major, critical) of alarms written to the syslog. Use the msgfwdlvl field to set the level (for example, critical) of alarms forwarded to a network management platform.

Example:

```
mml:sys>prov-add:ALM-PARMS:maxlines=50000,msgdisclvl=1,msgfwdlvl=2
Billing and Measurements Server - BAMS-00 2004-10-08 15:11:49
B  COMPLD
   "ALM-PARMS"
   ;
```

In this example, the *maxlines* field was set to 50000, the *msgdisclvl* field was set to 1 (that is, major), and the *msgfwdlvl* field was set to 2 (that is, minor).

ALM-PARMS Tag ID

Table 5-1 lists field names and values that can be specified with the ALM-PARMS tag ID.

Table 5-1 Fields in the Alarm Parameters Table

Field Name	Description	Values	Range
maxlines	Maximum number of lines in the syslog file (numeric).	Any number, for example, 10000	1 through 99999

Table 5-1 Fields in the Alarm Parameters Table (continued)

Field Name	Description	Values	Range
msgdisclvl	Message discrimination level (numeric). Used to specify the messages that are written to the syslog file. **Note** Any alarms at the selected value or lower are displayed (for example, a value of 2 displays all minor, major, and critical alarms).	Critical = 0 Major = 1 Minor = 2 Warning = 3 Informational = 5	0, 1, 2, 3, or 5
msgfwdlvl	Message forward level (numeric). Used to specify the messages that are trapped and forwarded to a network management platform. **Note** Any alarms at the selected value or lower are forwarded (for example, a value of 1 forwards all major and critical alarms).	Critical = 0 Major = 1 Minor = 2 Warning = 3 Informational = 5	0, 1, 2, 3, or 5

Updating the BIN1110 Table

Note This table is required only for systems that are configured to produce 1110 Binary output. For more information, see Chapter 9, "Configuring BAMS for NICS Output" in this guide.

Use the BIN1110 tag ID to update the BIN1110 table. This table allows you to disable or enable the generation of 1060 CDBs in TLV (tag, length, value) format, as well as to specify the suffix of the 1110 Binary output. By default, the system generates 1060, 1090, 1100, and 1110 CDB types, and appends the BIN suffix to the output files.

Example:

```
mml:1>prov-add:BIN1110:enable1060=1,filesuffix=BIN
Billing and Measurements Server - BAMS-00 2004-09-28 15:59:29
B  COMPLD
   "BIN1110"
   ;
```

In this example, the enable1060 field was set to 1, and the filesuffix field was set to BIN.

BIN1110 Tag ID

Table 5-2 lists field names and values that can be specified with the BIN1110 tag ID.

Table 5-2 Fields in the BIN1110 Table

Field Name	Description	Values
enable1070	This numeric parameter enables or disables the generation of 1070 CDBs. **Note** The BIN task no longer generates 1070 output. This field is no longer used.	Enabled = 1 Disabled = 0 (default)
enable1060	This numeric parameter enables or disables the generation of 1060 CDBs.	Enabled = 1 (default) Disabled = 0
filesuffix	This parameter allows you to specify the suffix of the output file. The suffix may be up to 8 characters long. **Note** If you change the default filesuffix value, ensure that you add the file pattern with the new suffix to the MSC Parameters table.	BIN (default)

Updating the Country Table

Use the COUNTRY tag ID to update the Country table. The Country table maintains country codes.

Example:

```
mml:3>prov-add:COUNTRY:countrycode=001,countryname="USA"
Billing and Measurements Server - BAMS-00 2004-10-08 15:20:35
B  COMPLD
   "COUNTRY"
   ;
```

In this example, the countrycode field was set to 001, and the countryname field was set to USA.

COUNTRY Tag ID

Table 5-3 lists field names and values that can be specified with the COUNTRY tag ID.

Table 5-3 Fields in the Country Table

Field Name	Description	Values
countrycode	Country code (numeric)	The 5-digit country code
countryname	Country name (string)	The country name, a maximum of 40 letters

Updating the Map Type Table

Use the MAPTYPE tag ID to update the Map Type table. The Map Type table defines which structure code to generate and which call type to assign for each call category.

Example:

```
mml:3>prov-add:MAPTYPE:maptype="FLAT",calltype="1",answered="502",unanswered="20"
Billing and Measurements Server - BAMS-00 2004-10-08 15:25:50
B  COMPLD
   "MAPTYPE"
   ;
```

In this example, the maptype field was set to flat, the calltype field was set to 1, the answered field was set to 502, and the unanswered field was set to 20.

MAPTYPE Tag ID

Table 5-4 lists field names and values that can be specified with the MAPTYPE tag ID.

Note The Maptype table should contain one record entry for each maptype value. The default system contains the entries. The user should not add or delete any entries, only modify them by specifying different field values with the Rating Type tag ID. The values are case sensitive.

Note A value of –1 in the answered or unanswered field means that a BAF record is not generated for that case.

Table 5-4 Fields in the Map Types Table

Field Name	Description	Values	Range
maptype	Map type (string)	FLAT MESSAGE INTRALATA INTERLATAORIGINATING INTERLATATERMINATING INTERLATATANDEM TOLLFREE	A maximum of 24 characters
calltype	Call type (string)	1, 6, 110, 119, 141	A maximum of 3 digits
answered	Answered (string)	–1, 1, 2, 20, 21, 360, 500, 502, 625, 653 **Note** A value of –1 does not generate a record.	A maximum of 4 digits
unanswered	Unanswered (string)	–1, 1, 2, 20, 21, 360, 500, 502, 625, 653 **Note** A value of –1 does not generate a record.	A maximum of 4 digits

Updating the Mass Storage Control Parameters Table

Use the MSC-PARMS tag ID to update the Mass Storage Control (MSC) Parameters table. This table maintains definitions of conditions that generate alarms or events, based on parameters such as file type, size, and age.

Example:

```
mml:sys>prov-add:MSC-PARMS:type="baf*",polled=0,alarm=5,agealarm=30,agelevel=2,delage=30,
delalarm=6
Billing and Measurements Server - BAMS-00 2004-10-08 15:30:20
B  COMPLD
   "MSC-PARMS"
   ;
```

In this example the type field was set to baf*, the polled field was set to 0, the alarm field was set to 5, the agealarm field was set to 30, the agelevel field was set to 2, the delage field was set to 30, and the delalarm field was set to 6.

MSC-PARMS Tag ID

Table 5-5 lists field names and values that can be specified with the MSC-PARMS tag ID.

Table 5-5 Fields in the Mass Storage Control Parameters Table

Field Name	Description	Values	Range
active	Enables/disables cleanup task (string)	Y = Yes, N = No	Y or N
dir	Directory (string)	/opt/CiscoBAMs/data	

Table 5-5 Fields in the Mass Storage Control Parameters Table (continued)

Field Name	Description	Values	Range
subdirs	Enables/disables cleanup of subdirectories (string)	Y = Yes, N = No	Y or N
type	File prefix or suffix to be monitored (string) **Note** If you change a default file prefix or suffix in another table, ensure that you add the new file pattern here also.	For example: cdr* baf* acc* CDR* cdr*BIN esem* p01_*bin	A maximum of 23 characters
polled	Polled/unpolled flag (numeric)	Polled = 1 Unpolled = 2 Header = 3	1 through 3
alarm	Alarm level to generate when you are deleting files (numeric)	Critical = 0 Major = 1 Minor = 2 Warning = 3 Manual = 4 Informational = 5 Debug = 6	0 through 6
agealarm	Number of days to wait before file is considered aged (numeric)	Any appropriate number	0 through 99
agelevel	Alarm level to generate when the file reaches its age limit (numeric)	Critical = 0 Major = 1 Minor = 2 Warning = 3 Reserved for future use = 4 Informational = 5 Debug = 6	0 through 6
delage	Number of days to wait before the system deletes the file (numeric)	Any appropriate number	0 through 99
delalarm	Alarm level to generate when deleting aged files (numeric)	Critical = 0 Major = 1 Minor = 2 Warning = 3 Reserved for future use = 4 Informational = 5 Debug = 6	0 through 6

Updating the Mass Storage Control Thresholds Table

Use the MSC-THRES tag ID to update the Mass Storage Control Thresholds table. This table maintains the system-level MSC thresholds that determine when the MSC task generates alarms based on disk-capacity thresholds, and when it begins and stops deleting specific types of files to free up disk space.

Example:

```
mml:sys>prov-add:MSC-THRES:interval=15,put=80,plt=60,uut=95,iut=950000,mil=80,
mal=85,crl=90
Billing and Measurements Server - BAMS-00 2004-10-08 15:34:02
B  COMPLD
   "MSC-THRES"
   ;
```

In this example the interval field was set to 15, the *put* field was set to 80, the plt field was set to 60, the uut field was set to 95, the *iut f*ield was set to 950000, the mil field was set to 80, the mal field was set to 85, and the crl field was set to 90.

MSC-THRES Tag ID

Table 5-6 lists field names and values that can be specified with the MSC-THRES tag ID.

Table 5-6 Fields in the Mass Storage Control Thresholds Table

Field Name	Description	Value	Range
interval	The run interval (in minutes). Identifies when the Mass Storage Control (MSC) task operates in the background and deletes available eligible files in the specified directories. This field requires a numeric value. A typical value is 30 minutes.	10	1 through 9999
put	The polled upper threshold (put) value. Identifies the disk capacity level (percent) at which the MSC task begins to delete polled files. **Note** Polled files are files that have been polled from BAMS by a polling computer and renamed to identify them as such. Unpolled files are files that have not been polled yet. This field requires a numeric value. A typical value is 60 percent.	80	1 through 100
plt	The polled lower threshold (plt) value. Identifies the disk capacity level (percent) at which the MSC task stops deleting polled files. This field requires a numeric value. A typical value is 40 percent.	60	1 through 100

Table 5-6 Fields in the Mass Storage Control Thresholds Table (continued)

Field Name	Description	Value	Range
uut	The unpolled upper threshold (uut) value. Identifies the disk capacity level (percent) at which the MSC task begins to delete unpolled data files. The MSC continues to delete these files until the disk capacity level drops below the threshold value. **Note** This value should be set high. For example, a value of 100% ensures that no unpolled files are deleted. This field requires a numeric value. A typical value is 80 percent.	95	1 through 100
iut	The inode upper threshold (iut). Defines an upper limit expressed by the maximum number of inodes, which is roughly equivalent to the number of files. When the maximum number is exceeded, the MSC task begins to delete files until the disk capacity level drops below the threshold value. This field requires a numeric value.	950000	1000 through 1000000
mil	The minor alarm level value (a percentage of the disk). Indicates when the MSC task will generate a minor alarm. If the disk is still occupied by files (up to the threshold disk level) after the MSC task background cleanup routine is run, a minor alarm, such as MSC032, is generated. This field requires a numeric value. A typical value is 60 percent.	80	1 through 100
mal	The major alarm level value (a percentage of the disk). Indicates when the MSC task will generate a major alarm. If the disk is still occupied by files (up to the threshold disk level) after the MSC task background cleanup routine is run, a major alarm, such as MSC031, is generated. This field requires a numeric value. A typical value is 70 percent.	85	1 through 100
crl	The critical alarm level value (a percentage of the disk). Indicates when the MSC task will generate a minor alarm. If the disk is still occupied by files (up to the threshold disk level) after the MSC task background cleanup routine is run, a minor alarm, such as MSC030, is generated. This field requires a numeric value. A typical value is 80 percent.	90	1 through 100

Updating the Node Parameters Table

Use the NODEPARMS tag ID to update the Node Parameters table. This table maintains node configuration information such as output types (for example, ASCII, BAF, NICS, P01) and measurements.

Example:

```
mml:1:node1>prov-add:NODEPARMS:activate=1,statoutput=1,bafoutput=1,asciibafoutput=1,
asciioutput=2,lookupinfo=1,bafinfo=1,dynamicaccumes=0,sup-zero-counts=0,interval-minutes=
15,nailed-cfg=0,p01output=0,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,
bin1110output=0
Billing and Measurements Server - BAMS-00 2004-10-19 15:51:44
B  COMPLD
   "NODEPARMS"
   ;
```

In this example, the activate field was set to 1, the statoutput field was set to 1, the bafoutput field was set to 1, the asciibafoutput field was set to 1, the asciioutput field was set to 2, the lookupinfo field was set to 1, the bafinfo field was set to 1, the dynamicaccumes field was set to 0, the sup-zero-counts field was set to 0, the interval-minutes field was set to 15, the nailed-cfg field was set to 0, the p01output field was set to 0, the p01prefix field was set to p01_, the enable-h323 field was set to 0, the extasciioutput field was set to 0, the nicsoutput field was set to 0, and the bin1110 output field was set to 0.

Caution Modifying the interval-minutes or nailed-cfg parameters automatically triggers a restart of the node-level processing tasks, resulting in measurement anomalies and the generating of associated traps.

NODEPARMS Tag ID

Table 5-7 lists field names and values that can be specified with the NODEPARMS tag ID.

Table 5-7 Fields in the Node Parameters Table

Field Name	Description	Value	Range
activate	This numeric parameter activates polling for a node. If the parameter is enabled, BAMS actively attempts to poll data files from the switch. After polling, BAMS processes all data. If this parameter is disabled, the BAMS polling function is turned off, and BAMS does not poll any data from the switch.	Enabled = 1 (default for node 1) Disabled = 0 (default for nodes 2 to 8)	0 or 1
statoutput	Statistics output mode (numeric). This parameter enables or disables the generation of the measurement files.	Enabled = 1 (default) Disabled = 0	0 or 1

Table 5-7 Fields in the Node Parameters Table (continued)

Field Name	Description	Value	Range
bafoutput	BAF output mode (numeric). This parameter enables or disables the generation of the BAF output files. When the parameter is enabled, BAMS converts the CDR records from the switch to BAF format, and places these files in the BAF subdirectory under each node's data directory (for example, opt/CiscoBAMS/data/s0x/BAF).	Enabled = 1 Disabled = 0 (default)	0 or 1
asciibafoutput	This parameter enables or disables the generation of the BAF output files in readable ASCII format. When enabled, BAMS converts the CDR records from the switch to ASCII BAF format, and places these files in the ASCIIBAF subdirectory under each node's data directory (for example, opt/CiscoBAMS/data/s0x/ASCIIBAF).	Disabled = 0 (default) Enabled = 1	0 through 1
asciioutput	ASCII output mode (numeric). This parameter enables or disables the generation of the ASCII output files. When the parameter is enabled, BAMS converts the CDR records from the switch to ASCII format, and places these files in the ASCII subdirectory under each node's data directory (for example, opt/CiscoBAMS/data/s0x/ASCII).	Disabled = 0 Enabled = 1 (ANSI output) Enabled = 2 (ITU output) (default)	0 through 2
lookupinfo	Look up info mode (numeric). This parameter enables or disables messages to the node syslog of errors associated with table lookup failures. Lookup failures are always displayed in the files /opt/CiscoBAMS/files/sxx/AUG_cor_ac.log and /opt/CiscoBAMS/files/sxx/AUG_cor_ag.log. If the lookupinfo parameter is enabled, these messages are also logged to the node syslog so that the operator is readily notified. Normally the node syslog contains information on the total number of lookup errors per file. Enabling this parameter displays detailed information about each lookup error.	Trunk Group Enabled = 1 (default) NPA-NXX Enabled = 2 Both Enabled = 3 Both Disabled = 0	0 through 3

Table 5-7 Fields in the Node Parameters Table (continued)

Field Name	Description	Value	Range
bafinfo	BAF info mode (numeric). This parameter enables or disables messages to the node syslog concerning errors associated with the CDR-to-BAF (CTB) process. The CTB process outputs call records in BAF format. This parameter should be enabled if bafoutput is enabled.	Enabled = 1 Disabled = 0 (default)	0 or 1
dynamicaccumes	Dynamic accumulation mode (numeric). For more information, see the "Configured vs. Dynamic Trunk Group Output" section on page 11-25. **Note** If you change this value during the day, the measurements will be inaccurate for the remainder of the day. Also, dynamic trunk measurements are not kept if the mode is set to "2" (that is, report defined trunk groups only).	Report configured trunks only = 2 Report dynamic trunks only = 1 Report both configured and dynamic trunks = 0 (default)	0 through 2
sup-zero-counts	Zero count suppression (numeric). For more information, see the "Zero Counts" section on page 11-25.	Enabled = 1 Disabled = 0 (default)	0 or 1

Table 5-7 Fields in the Node Parameters Table (continued)

Field Name	Description	Value	Range
interval-minutes	This numeric parameter sets the measurements file interval (in minutes). By default, BAMS always creates hourly and daily measurements files as long as the statoutput parameter is enabled. This parameter specifies the interval if files are to be created at less than 1-hour intervals. The value of this parameter must be greater than and not equal to the value of the MGC CDR dump interval value in the dmprSink.dat file. For more information, see the "Configuring the Cisco MGC for Using BAMS" section on page 2-1. **Note** Modifying this parameter while the system is running triggers an automatic restart of the processing tasks for the current node. Such a restart could affect the accuracy of the current measurements. A restart also creates alarms and traps that state that the processing tasks have been terminated and are being restarted (MGR801).	5, 10, 15, 20, 30 (Default = 15)	5 through 30
nailed-cfg	Enabling this numeric parameter specifies that the switch has been set to a nailed configuration. Disabling this parameter specifies that the switch has been set to a switched configuration. **Note** Modifying the parameter nailed-cfg while the system is running triggers an automatic restart of the processing tasks for the current node. Such a restart could affect the accuracy of the current measurements. A restart also creates alarms and traps that state that the processing tasks have been terminated and are being restarted (MGR801).	Enabled = 1 (default) Disabled = 0	0 or 1

Table 5-7 Fields in the Node Parameters Table (continued)

Field Name	Description	Value	Range
p01output	This parameter enables or disables the generation of the p01 output files by the P01 node task. These files are generated and stored in the directory /opt/CiscoBAMS/data/sxx/p001, where xx=node number. If you enable P01 output, you can filter out certain types of calls (that is, answered, noanswer, busy and other), based on the cause code from the MGC data record. For more information, see the "Updating the P01 Filter Table" section on page 5-16.	Enabled = 1 Disabled = 0 (default)	0 or 1
p01prefix	This string parameter specifies the file prefix of the p01 output files stored in /opt/CiscoBAMS/data/sxx/p001.	p01_ (default) **Note** A value of NULL or empty means that the p01 output file will have the same name as the input file.	23 characters
enable-H323	This numeric parameter enables or disables the H.323 measurements. For a complete listing of H.323 measurements, see the "Noncarrier Measurements" section on page 11-2.	Enabled = 1 Disabled = 0 (default)	0 or 1
extasciioutput	This parameter enables or disables the generation of Extendable ASCII data.	Disabled = 0 (default) Enabled = 1 (ANSI output) Enabled = 2 (ITU output)	0 through 2
nicsoutput	This parameter enables or disables the generation of Non-Intercompany Settlement System (NICS) data.	Disabled = 0 (default) Enabled = 1 (ANSI output) Enabled = 2 (ITU output)	0 through 2
bin1110output	This parameter enables or disables the generation of 1110 Binary data.	Disabled = 0 (default) Enabled = 1 (ANSI output) Enabled = 2 (ITU output)	0 through 2

Updating the NPANXX Table

Use the NPANXX tag ID to update the NPANXX table. This table allows you to assign a zone ID and a local access transport area (LATA) to each NPA-NXX.

Example:

```
mml:1>prov-add:NPANXX:npanxx=301829,lata=00001,zone=0000001
Billing and Measurements Server - BAMS-00 2004-10-08 15:52:29
B  COMPLD
   "NPANXX"
   ;
```

In this example, the npanxx field was set to 301829, the lata field was set to 00001, and the zone field was set to 0000001.

NPANXX Tag ID

Table 5-8 lists field names and values that can be specified with the NPANXX tag ID.

Table 5-8 Fields in the NPANXX Table

Field Name	Description	Values
npanxx	NPA and NXX (numeric)	The 6-digit NPANXX number
lata	LATA (numeric)	The 5-digit LATA number to which this NPANXX belongs
zone	Zone ID (numeric)	The 1- through 8-digit zone ID for a given Cisco MGC

Updating the P01 Filter Table

Use the P01FILTER tag ID to determine the type of calls that should be filtered from the P01 output, based on the cause code of the MGC data record. You enable P01 output with the NODEPARMS tag ID. For more information, see the "Updating the Node Parameters Table" section on page 5-10.

Example:

```
mml:1:node1>prov-rtrv:p01filter:
Billing and Measurements Server - BAMS-00 2004-10-11 09:35:57
B  RTRV
   "P01FILTER:answered=1,noanswer=1,busy=1,other=1"
```

In this example, the answered field was set to 1, the noanswer field was set to 1, the busy field was set to 1, and the other field was set to 1.

For more information about P01 filtering, see Chapter 8, "Configuring BAMS for P01 Output."

P01FILTER Tag ID

Table 5-9 lists field names and values that can be specified with the P01FILTER tag ID.

Table 5-9 Fields in the P01 Filtering Table

Field Name	Description	Values	Range
answered	Answered call filter (numeric). This parameter enables (1) or disables (0) the filtering of answered calls.	Enabled = 1 Disabled = 0 (default)	0 or 1
noanswer	Noanswer call filter (numeric). This parameter enables (1) or disables (0) the filtering of noanswer calls.	Enabled = 1 Disabled = 0 (default)	0 or 1
busy	Busy call filter (numeric). This parameter enables (1) or disables (0) the filtering of busy calls.	Enabled = 1 Disabled = 0 (default)	0 or 1
other	Other call filter (numeric). This parameter enables (1) or disables (0) the filtering of other calls.	Enabled = 1 Disabled = 0 (default)	0 or 1

Updating the Poll Table

Use the POLL tag ID to update the Poll table. This table allows you to configure BAMS polling parameters such as hostname, remote directory and file prefix and suffix information.

Example:

```
mml:1>prov-add:POLL:host-name1="newyork1",rem-dir1="/opt/CiscoMGC/var/bam",
file-prefix1="cdr*",file-suffix1=".bin",action="R",interval=1,timeout=1,maxtries=2
Billing and Measurements Server - BAMS-00 2004-10-08 15:56:05
B  COMPLD
   "POLL"
   ;
```

In this example, the host-name1 field was set to newyork1, the rem-dir1 field was set to /opt/CiscoMGC/var/bam, the file-prefix1 field was set to cdr*, the file-suffix1 field was set to .bin, the action field was set to R, the interval field was set to 1, the timeout field was set to 1, and the maxtries field was set to 2.

POLL Tag ID

Table 5-10 lists field names and values that can be specified with the POLL tag ID.

Table 5-10 Fields in the Poll Table

Field Name	Description	Values	Range
host-name1	Host name for Cisco MGC1 (string)	Name of the host	A maximum of 15 characters
rem-dir1	Remote directory for Cisco MGC1 (string)	Full text path of the directory	A maximum of 42 characters
file-prefix1	Prefix of data files on Cisco MGC1 (string)	cdr*	A maximum of 10 characters
file-suffix1	Suffix of files on Cisco MGC1 (string)	.bin	A maximum of 10 characters
host-name2	Host name for Cisco MGC2 (string) **Note** Required only if a second MGC is operational.	Name of the second host	A maximum of 15 characters
rem-dir2	Directory name on Cisco MGC2 (string) **Note** Required only if a second MGC is operational.	Full text path of the directory, for example, opt/CiscoMGC/var/bam or opt/TransPath/var/bam	A maximum of 42 characters
file-prefix2	Prefix of data files on Cisco MGC2 (string)	cdr	A maximum of 10 characters
file-suffix2	Suffix of files on Cisco MGC2 (string) **Note** Required only if a second MGC is operational.	.bin	A maximum of 10 characters

Table 5-10 Fields in the Poll Table (continued)

Field Name	Description	Values	Range
action	Action after polling (delete/rename) (string)	Delete = D Rename = R Rename to .finished on the MGC.	D or R
interval	Polling interval (in minutes) (numeric)	1 through 99 5 minutes is the recommended setting.	1 through 99
timeout	Timeout for file transfer (in minutes) (numeric)	1 through 9999	1 through 9999
maxtries	Maximum number of tries on each file (numeric)	1 through 10	1 through 10

Note When BAMS collects a CDR file from the Cisco MGC, it renames that file on the Cisco MGC with a .finished suffix. For purposes of polling efficiency, if BAMS is not able to rename the polled file ".finished", it does not retry but instead generates an SNMP trap based on the POL115 message.

When a downstream polling computer collects a file from the BAMS unit, it renames that file on the BAMS unit to .finished. The Mass Storage Control (MSC) task maintains disk space and periodically checks polled and unpolled files on the BAMS unit, that is, files that have or have not been polled by a downstream polling computer.

Updating the Rating Exception Table

Use the RATE-EXC tag ID to update the Rating Exception table. The values in the Rating Exception table override the relationships (in terms of call category) that were defined with the NPANXX tag ID. This table allows you to support special billing arrangements.

Example:

```
mml:1>prov-add:RATE-EXC:orgnpanxx=301829,trmnpanxx=301831,ratetype="FLAT"
Billing and Measurements Server - BAMS-00 2004-10-08 15:59:12
B  COMPLD
   "RATE-EXC"
   ;
```

In this example, the orgnpanxx field was set to 301829, the trmnpanxx field was set to 301831, and the ratetype field was set to FLAT.

RATE-EXC Tag ID

Table 5-11 lists field names and values that can be specified with the Rating Exception tag ID.

Table 5-11 Fields in the Rating Exception Table

Field Name	Description	Values	Range
orgnpanxx	Originating NPA and NXX (numeric)	6-digit NPANXX number	A maximum of 6 digits
trmnpanxx	Terminating NPA and NXX (numeric)	6-digit NPANXX number	A maximum of 6 digits
ratetype	Rating type (string)	FLAT MESSAGE INTRALATA INTERLATA	A maximum of 24 characters

Updating the Rating Type Table

Use the RATING-TYPE tag ID to update the Rating Type table. The Rating Type table defines the relationships (in terms of call category) between different zones. This table is directional. That is, calling from Zone 1 to Zone 2 might be categorized one way, and calling in the reverse direction might be categorized another way.

Example:

```
mml:1>prov-add:RATING-TYPE:orgzone=123,trmzone=123,ratetype="FLAT"
Billing and Measurements Server - BAMS-00 2004-10-08 15:58:19
B  COMPLD
   "RATING-TYPE"
   ;
```

In this example, the orgzone field was set to 123, the trmzone field was set to 123, and the ratetype field was set to FLAT.

RATING-TYPE Tag ID

Table 5-12 lists field names and values that can be specified with the RATING-TYPE tag ID.

Table 5-12 Fields in the Rating Type Table

Field Name	Description	Values	Range
orgzone	Originating zone (numeric)	8-digit originating zone	A maximum of 8 digits
trmzone	Terminating zone (numeric)	8-digit terminating zone	A maximum of 8 digits
ratetype	Rating type (string)	FLAT MESSAGE INTRALATA INTERLATA	A maximum of 24 characters

Updating the Nailed Connection Table

In a nailed environment, use the SIGPATH tag ID to update the Nailed Connection table. This table maintains nailed connections for the Cisco SC2200 configuration.

Trunk groups are populated in the Cisco MGC call detail records based on table lookups of read-in hexadecimal values for a signal path. On Cisco SC2200 applications, sigpath ID is used as a reference for trunk groups. Trunk group numbers are as follows: BAF (1 through 9999), ITU (4096), and ANSI (1684). On a nailed application, you must have a SIGPATH association for every circuit that must be added.

The maximum number of records defined in the Nailed Connection table is 100,000.

Example:

```
mml:1>prov-add:sigpath:sigpath=0x50-0x60,trunkgrp=0-16
Billing and Measurements Server - BAMS-00 2004-10-08 16:02:37
B  COMPLD
   "SIGPATH"
   ;
```

In this example, the sigpath field was set to the range 0x50–0x60, and the trunkgrp field range was set to 0–16.

SIGPATH Tag ID

Table 5-13 lists field names and values that can be specified with the SIGPATH tag ID.

Note The sigpath and trunkgrp fields are independent keys; therefore, they must follow a one-to-one mapping. BAMS does not allow duplicate entries in either column.

Table 5-13 Fields in the Nailed Connection Table

Field Name	Description	Values	Range
sigpath	Signal path (numeric)	8-digit hexadecimal number	A maximum of 8 digits
trunkgrp	Trunk group (numeric) **Note** BAF records store only the 4 least-significant decimal digits of the trunk group value. Cisco MGC End of Call 1110 ASCII records and measurements support the full range.	0 through 65535	0 through 65535

Updating the Skip CDB Table

Use the SKIPCDB tag ID to update the Skip CDB table. This table maintains a list of CDBs produced by the Cisco MGC that BAMS should not process. By default this table contains two values: 1020 and 1070.

Example:

```
mml:1>prov-add:SKIPCDB:id=1020
Billing and Measurements Server - BAMS-00 2004-10-08 16:02:37
B  COMPLD
      ;
mml:1>prov-add:SKIPCDB:id=1070
Billing and Measurements Server - BAMS-00 2004-10-08 16:02:51
B  COMPLD
      ;
```

In this example, the id field was set to 1020, and again to 1070.

SKIPCDB Tag ID

Table 5-14 lists the field name that can be specified with the SKIPCDB tag ID.

Table 5-14 Fields in the Skip CDB Table

Field Name	Description	Values	Range
id	A CDB identifier is used to identify any new CDB that the Cisco MGC is producing and that BAMS should skip in its processing. Please contact Cisco for assistance.	Default: 1020 and 1070	Up to 10-digit numeric value ($2^{31} - 1$)

Updating the Skip CDE Table

Use the SKIPCDE tag ID to update the Skip CDE table. This table maintains a list of CDEs produced by the Cisco MGC that BAMS should not process. By default this table is empty.

Example:

```
mml:1>prov-add:SKIPCDE:id=5000
Billing and Measurements Server - BAMS-00 2004-10-08 16:02:37
B  COMPLD
      ;
```

In this example, the id field was set to 5000.

SKIPCDE Tag ID

Table 5-15 lists field names and values that can be specified with the SKIPCDE tag ID.

Table 5-15 Fields in the Skip CDE Table

Field Name	Description	Values	Range
id	A CDE identifier is used to identify any new CDE that the Cisco MGC is producing and that BAMS should skip in its processing. Please contact Cisco for assistance.	Default: empty	Up to 10-digit numeric value ($2^{31} - 1$)

Updating the Switch Information Table

Use the SWITCHINFO tag ID to update the Switch Information table. This table must be populated if BAF output has been enabled. For more information, see Chapter 6, "Configuring BAMS for BAF Output."

The values maintained in this table are customer-specific and must be obtained from Telcordia. Refer to the Telcordia/Bellcore document GR-478-CORE for more detailed descriptions of these fields.

Example:

```
mml:1:node01>prov-ed:switchinfo:sensortype="118",sensorid="999999",recoffctype="118",
recoffcid="999999"
Billing and Measurements Server - BAMS-00 2004-09-15 10:15:19
B  COMPLD
   "SWITCHINFO"
   ;
```

In this example, the sensortype field was set to 118, the sensorid field was set to 999999, the recoffctype field was set to 118, and the recoffcid field was set to 999999.

SWITCHINFO Tag ID

Table 5-16 lists field names and values that can be specified with the SWITCHINFO tag ID.

Table 5-16 Fields in the Switch Information Table

Field Name	Description	Values	Range
sensortype	Sensor type	Default = 0	See Telcordia/Bellcore document GR-478-CORE
sensorid	Sensor ID	Default = 0	See Telcordia/Bellcore document GR-478-CORE
recoffctype	Recording office type	Default = 0	See Telcordia/Bellcore document GR-478-CORE
recoffcid	Recording office ID	Default = 0	See Telcordia/Bellcore document GR-478-CORE

Updating the Threshold Crossing Alarms Table

Use the TCA-TBL tag ID to update the Threshold Crossing Alarms (TCA) table.

Each TAG/TRK (where TAG is a user-defined value, and TRK is the trunk) can have a maximum of 43 measurements. A global TCA can be set up with a maximum of 43 measurements. Any TAG/TRK that does not have a specific TCA is checked against the global TCA.

For eight measurements, you can specify TAG/TRK/IC, where TAG is a user-defined value, TRK is the trunk, and IC is the interexchange carrier.

You need to know the carrier codes (for example, 0288 for AT&T). Enter three-digit codes as four digits with a 0 as the first digit. Refer to Chapter 11, "Obtaining Measurements," for additional information.

A special measurement group can be specified to apply to all TAG/TRK measurement groups that are not otherwise specified. This measurement group is identified by the name "global/0," where the TAG is "global" and the trunk group is "0."

If there is no global/0 defined, any measurement that does not have a specific threshold set for it is not checked. The measurement is still reported in the acc_x, acc_r, acc_h, or acc_d file, but no alarm is generated, regardless of the value.

If global/0 is defined, it is used when no specific thresholds have been specified for a trunk group. If you set thresholds for a specific TAG/TRK, only the specified values are checked. Any unspecified measurements within the TAG/TRK are ignored.

Note When setting up a global TCA for a trunk group, do not use a global measurement change for trunk groups where measurements are not used.

In Table 5-17, peg counts refer to statistics. Any number can be specified for the threshold values; the only limit is the field length. The condition values are as follows:

0 = Ignore, 1 = Less than, 2 = Equal, 3 = Greater than, and 4 = Not equal.

None of the fields shown in Table 5-17 are required fields. Each threshold is based on user-defined alarms. If you specify a value parameter, you must also specify its corresponding condition parameter, and vice versa. For each defined record, the id record field must be populated.

Note When setting up a TCA for all trunk groups, do not use a global measurement change for trunk groups where measurements are not used.

Once you use the **prov-add** command for all trunk groups, use the **prov-ed** command for all other trunk group changes.

Example (one trunk group):

```
mml:1>prov-add:tca-tbl:id="1/800",igr-call-att-cond=3,igr-call-att=5

Billing and Measurements Server - BAMS-00 2004-10-06 15:45:32
B COMPLD
 "TCA-TBL"
 ;
```

In this example, the id field was set to 1/800, the igr-call-att-cond field was set to 3, and the igr-call-att field was set to 5.

Example (all trunk groups):

```
mml:1>prov-add:tca-tbl:id="global/0",igr-call-att-cond=3,igr-call-att=5

Billing and Measurements Server - BAMS-00 2004-10-06 15:45:32
B COMPLD
 "TCA-TBL"
 ;
```

In this example, the id field was set to global/0, the igr-call-att-cond field was set to 3, and the igr-call-att field was set to 5.

Note To suppress a threshold condition, specify a value of 0 (ignore) with the **prov-ed** command. Do not use the **prov-dlt** command, because it deletes all thresholds for the specified id.

TCA-TBL Tag ID

Table 5-17 lists field names and values that can be specified with the Threshold Crossing Alarm (TCA-TBL) tag ID.

Table 5-17 Fields in the Threshold Crossing Alarms Table

Field Name	Description	Values
id	User-defined tag/trunk group/interconnect carrier (string) **Note** Do not include the trunk group prefix (from the TRUNKGRP table) in this field.	A maximum of 43 characters
igr-call-att-cond	Call attempts incoming threshold condition (numeric)	0 through 4
igr-call-att	Call attempts incoming threshold value (numeric)	Any number
egr-call-att-cond	Call attempts outgoing threshold condition (numeric)	0 through 4
egr-call-att	Call attempts outgoing threshold value (numeric)	Any number
egr-call-blkd-cond	Outgoing calls blocked threshold condition (numeric)	0 through 4
egr-call-blkd	Outgoing calls blocked threshold value	Any number
ttl-failed-cong-cond	Failed calls congestion threshold condition (numeric)	0 through 4
ttl-failed-cong	Failed calls congestion threshold value (numeric)	Any number
igr-term-norm-cond	Successful calls incoming threshold condition (numeric)	0 through 4
igr-term-norm	Successful calls incoming threshold value (numeric)	Any number
egr-term-norm-cond	Successful calls outgoing threshold condition (numeric)	0 through 4
egr-term-norm	Successful calls outgoing threshold value (numeric)	Any number
igr-pct-trk-use-cond	Percent trunk group usage incoming threshold condition (numeric)	0 through 4
igr-pct-trk-use	Percent trunk group usage incoming threshold condition (numeric)	Any number

Table 5-17 Fields in the Threshold Crossing Alarms Table (continued)

Field Name	Description	Values
egr-pct-trk-use-cond	Percent trunk group usage outgoing threshold condition (numeric)	0 through 4
egr-pct-trk-use	Percent trunk group usage outgoing threshold value (numeric)	Any number
ttl-maint-use-cond	Maintenance usage per trunk group threshold condition (numeric)[1]	0 through 4
ttl-maint-use	Maintenance usage per trunk group threshold value (numeric)[1]	Any number
ttl-erlangs-cond	Total traffic erlangs threshold condition (numeric)	0 through 4
ttl-erlangs	Total traffic erlangs threshold value (numeric)	Any number
ttl-term-norm-cond	Total calls terminated normally threshold condition (numeric)	0 through 4
ttl-term-norm	Total calls terminated normally threshold value (numeric)	Any number
ttl-term-abnorm-cond	Calls terminated abnormally threshold condition (numeric)	0 through 4
ttl-term-abnorm	Calls terminated abnormally threshold value (numeric)	Any number
ttl-term-failed-mgw-cond	Calls terminated, failed MGW or NAS threshold condition (numeric)	0 through 4
ttl-term-failed-mgw	Calls terminated, failed MGW or NAS threshold value (numeric)	Any number
ttl-calls-rejected-cond	Calls rejected threshold condition (numeric)	0 through 4
ttl-calls-rejected	Calls rejected threshold value (numeric)	Any number
ttl-rejected-dialnum-cond	Calls rejected, unknown dialed number threshold condition (numeric)	0 through 4
ttl-rejected-dialnum	Calls rejected, unknown dialed number threshold value (numeric)	Any number
ttl-rejected-other-cond	Calls rejected, other reasons threshold condition (numeric)	0 through 4
ttl-rejected-other	Calls rejected, other reasons threshold value (numeric)	Any number
egr-ofl-blkd-cond	Overflow, outgoing attempts blocked threshold condition (numeric)	0 through 4
egr-ofl-blkd	Overflow, outgoing attempts blocked threshold value (numeric)	Any number
ttl-traffic-usage-pegs-cond	Total traffic usage pegs threshold condition (numeric)	0 through 4
ttl-traffic-usage-pegs	Total traffic usage pegs threshold value (numeric)	Any number
egr-tndm-att-cond	Tandem routing attempts, outgoing threshold condition (numeric)	0 through 4

Table 5-17 Fields in the Threshold Crossing Alarms Table (continued)

Field Name	Description	Values
egr-tndm-att	Tandem routing attempts, outgoing threshold value (numeric)	Any number
egr-tndm-cmplt-cond	Tandem completions, outgoing threshold condition (numeric)	0 through 4
egr-tndm-cmplt	Tandem completions, outgoing threshold value (numeric)	Any number
igr-tndm-att-cond	Tandem routing attempts, incoming threshold condition (numeric)	0 through 4
igr-tndm-att	Tandem routing attempts, incoming threshold value (numeric)	Any number
igr-tndm-cmplt-cond	Tandem completions, incoming threshold condition (numeric)	0 through 4
igr-tndm-cmplt	Tandem completions, incoming threshold value (numeric)	Any number
egr-tndm-dur-cond	Tandem duration, outgoing threshold condition (numeric)	0 through 4
egr-tndm-dur	Tandem duration, outgoing threshold value (numeric)	Any number
igr-tndm-dur-cond	Tandem duration, incoming threshold condition (numeric)	0 through 4
igr-tndm-dur	Tandem duration, incoming threshold value (numeric)	Any number
igr-conv-dur-cond	Conversation duration, incoming threshold condition (numeric)	0 through 4
igr-conv-dur	Conversation duration, incoming threshold value (numeric)	Any number
egr-conv-dur-cond	Conversation duration, outgoing threshold condition (numeric)	0 through 4
egr-conv-dur	Conversation duration, outgoing threshold value (numeric)	Any number
igr-setup-dur-cond	Setup duration, incoming threshold condition (numeric)	0 through 4
igr-setup-dur	Setup duration, incoming threshold value (numeric)	Any number
egr-setup-dur-cond	Setup duration, outgoing threshold condition (numeric)	0 through 4
egr-setup-dur	Setup duration, outgoing threshold value (numeric)	Any number
igr-teardown-dur-cond	Teardown duration, incoming threshold condition (numeric)	0 through 4
igr-teardown-dur	Teardown duration, incoming threshold value (numeric)	Any number
egr-teardown-dur-cond	Teardown duration, outgoing threshold condition (numeric)	0 through 4

Table 5-17 Fields in the Threshold Crossing Alarms Table (continued)

Field Name	Description	Values
egr-teardown-dur	Teardown duration, outgoing threshold value (numeric)	Any number
egr-ic-calls-cond	IC destined calls threshold condition (numeric)	0 through 4
egr-ic-calls	IC destined calls threshold value (numeric)	Any number
egr-ic-nockt-cond	IC destined calls, no circuit threshold condition (numeric)	0 through 4
egr-ic-nockt	IC destined calls, no circuit threshold value (numeric)	Any number
ttl-ic-usage-cond	IC usage threshold condition (numeric)	0 through 4
ttl-ic-usage	IC usage threshold value (numeric)	Any number
ttl-route1-cond	Route 1 threshold condition (numeric)	0 through 4
ttl-route1	Route 1 threshold value (numeric)	Any number
ttl-route2-cond	Route 2 threshold condition (numeric)	0 through 4
ttl-route2	Route 2 threshold value (numeric)	Any number
ttl-route3-cond	Route 3 threshold condition (numeric)	0 through 4
ttl-route3	Route 3 threshold value (numeric)	Any number
egr-scfl-h323-cond	Successful H.323 terminating threshold condition (numeric)	0 through 4
egr-scfl-h323	Successful H.323 terminating threshold value (numeric)	Any number
igr-scfl-h323-cond	Successful H.323 originating threshold condition (numeric)	0 through 4
igr-scfl-h323	Successful H.323 originating threshold value (numeric)	Any number
egr-unscfl-h323-cond	Unsuccessful H.323 terminating threshold condition (numeric)	0 through 4
egr-unscfl-h323	Unsuccessful H.323 terminating threshold value (numeric)	Any number
igr-unscfl-h323-cond	Unsuccessful H.323 originating threshold condition (numeric)	0 through 4
igr-unscfl-h323	Unsuccessful H.323 originating threshold value (numeric)	Any number
ttl-ic-sel-noind-cond	Total carrier select no indication threshold condition (numeric)	0 through 4
ttl-ic-sel-noind	Total carrier select no indication threshold value (numeric)	Any number
ttl-ic-sel-pr-nipt-cond	Total carrier select presubscribed not input threshold condition (numeric)	0 through 4
ttl-ic-sel-pr-nipt	Total carrier select presubscribed not input threshold value (numeric)	Any number

Table 5-17 Fields in the Threshold Crossing Alarms Table (continued)

Field Name	Description	Values
ttl-ic-sel-pr-input-cond	Total carrier select presubscribed input threshold condition (numeric)	0 through 4
ttl-ic-sel-pr-input	Total carrier select presubscribed input threshold value (numeric)	Any number
ttl-ic-sel-pr-wni-cond	Total carrier select presubscribed with no indication threshold condition (numeric)	0 through 4
ttl-ic-sel-pr-wni	Total carrier select presubscribed with no indication threshold value (numeric)	Any number
ttl-ic-sel-no-pr-cond	Total carrier select not presubscribed threshold condition (numeric)	0 through 4
ttl-ic-sel-no-pr	Total carrier select not presubscribe threshold value (numeric)	Any number
egr-scfl-isup-cond	Successful ISUP terminating pegs condition (numeric)	0 through 4
egr-scfl-isup	Successful ISUP terminating pegs value (numeric)	Any number
igr-scfl-isup-cond	Successful ISUP originating pegs condition (numeric)	0 through 4
igr-scfl-isup	Successful ISUP originating pegs value (numeric)	Any number
egr-unscfl-isup-cond	Unsuccessful ISUP terminating pegs condition (numeric)	0 through 4
egr-unscfl-isup	Unsuccessful ISUP terminating pegs value (numeric)	Any number
igr-unscfl-isup-cond	Unsuccessful ISUP originating pegs condition (numeric)	0 through 4
igr-unscfl-isup	Unsuccessful ISUP originating pegs value (numeric)	Any number
egr-isdn-su-msg-delay-cond	ISDN terminating setup message delay pegs condition (numeric)	0 through 4
egr-isdn-su-msg-delay	ISDN terminating setup message delay pegs value (numeric)	Any number
igr-isdn-su-msg-delay-cond	ISDN originating setup message delay pegs condition (numeric)	0 through 4
igr-isdn-su-msg-delay	ISDN originating setup message delay pegs value (numeric)	Any number
ttl-avlbl-cic-cond	Total number of available CICs threshold condition (numeric)	0 through 4
ttl-avlbl-cic	Total number of available CICs threshold value (numeric)	Any number

1. This value is disabled and if populated, no alarm is generated.

Updating the Tollfree Table

Use the TOLLFREE tag ID to update the Tollfree table. The Tollfree table maintains the toll-free prefixes.

Example:

```
mml:1>prov-add:TOLLFREE:digits=800
Billing and Measurements Server - BAMS-00 2004-10-08 16:25:17
B  COMPLD
   "TOLLFREE"
   ;
```

In this example, the digits field was set to 800.

TOLLFREE Tag ID

Table 5-18 lists the field name that can be specified with the TOLLFREE tag ID.

Table 5-18 Fields in the Tollfree Table

Field Name	Description	Values	Range
digits	This numeric prefix identifies a toll-free call. Default values are 800, 877, 888, 1800, 1877, and 1888. Additional values can be entered if needed.	800, 877, 888, 1877, 1800, 1888, any valid NPA, three or four characters	A maximum of 12 digits

Updating the Trunk Group Table

Use the TRUNKGRP tag ID to update the Trunk Group table. The Trunk Group table provides information about each trunk group. This information defines whether a call is originating or terminating. In addition, the default NPA is taken from here if only 7 digits are populated in the calling or called phone numbers.

Note When BAMS is configured in PGW Dynamic Update mode (that is, where PGW_DYNAMIC_UPDATE=TRUE), the circuits field does not appear in the Trunk Group table. For more information, see "Setting the PGW Dynamic Update Mode" section on page 2-16.

PGW Dynamic Update Mode = False

In this mode the circuits field is included in the Trunk Group table.

Non-BAF Mode Example (Where ORGNPA and TRMNPA Are Not Required)

```
mml:1>Prov-add:trunkgrp:trunkgrp=1000,connection="T",circuits=24
mml:1>Prov-ed:trunkgrp:trunkgrp=1000,circuits=24,prefix="abc"
```

BAF Mode Example (Where ORGNPA and TRMNPA Are Required)

```
mml:2>PROV-ADD:TRUNKGRP:trunkgrp=1000,connection="T",ORGNPA="123",TRMNPA="456",circuits=24
mml:2>PROV-ADD:TRUNKGRP:trunkgrp=1000,connection="T",ORGNPA="123",TRMNPA="456",circuits=24
,prefix="abc"
```

PGW Dynamic Update Mode = True

In this mode the circuits field is not included in the Trunk Group table. For more information, see "Setting the PGW Dynamic Update Mode" section on page 2-16.

Non-BAF Mode Example (Where ORGNPA and TRMNPA Are Not Required)

```
mml:3>Prov-add:trunkgrp:trunkgrp=1000,connection="T"
mml:3>Prov-ed:trunkgrp:trunkgrp=1000,prefix="abc"
```

BAF Mode Example (Where ORGNPA and TRMNPA Are Required)

```
mml:3>PROV-ADD:TRUNKGRP:trunkgrp=1000,connection="T",ORGNPA="123",TRMNPA="456"
mml:3>PROV-ADD:TRUNKGRP:trunkgrp=1000,connection="T",ORGNPA="123",TRMNPA="456",
prefix="abc"
```

TRUNKGRP—Trunk Group Table

Table 5-19 lists field names and values that can be specified with the TRUNKGRP tag ID.

Table 5-19 Fields in the Trunk Group Table

Field Name	Description	Values	Range
trunkgrp	Trunk group (numeric). This field is mandatory. **Note** BAF records store only the 4 least-significant decimal digits of the trunk group value. Cisco MGC End of Call 1110 ASCII records and measurements support the full range.	0 through 65535	0 through 65535
connection	Connection (string). This field is mandatory. This value is used for BAF output mapping. If BAF output is not enabled, you can use either value. **Note** A Direct (D) connection refers to calls terminated locally (directly) on the VSC. A Tandem (T) connection is a call collected over a long distance or remote tandem switching system and forwarded to the VSC.	Direct = D (for direct or local) Tandem = T	D or T
orgnpa	Originating NPA (numeric). This field is optional. For local 7-digit calls, this value is prepended to create a 10-digit number.	The 3-digit originating NPA	A maximum of 3 digits

Table 5-19 Fields in the Trunk Group Table (continued)

Field Name	Description	Values	Range
trmnpa	Terminating NPA (numeric). This field is optional. For local 7-digit calls, this value is prepended to create a 10-digit number.	The 3-digit terminating NPA	A maximum of 3 digits
circuits **Note** When the PGW_DYNAMIC_UPDATE environment variable is set to TRUE, this field is removed from the Trunk Group table.	Circuits (numeric). This value is used to determine some statistics, such as trunk group usage **Note** To eliminate data-validation errors on the BAMS system, this field should be populated even if a valid number does not exist (for example, there are no circuits in an IP trunk group). For a case in which the number of circuits does not exist for a trunk group, set this field to 1.	The number of total circuits in the trunk group	0 through 65535
prefix	Trunk group prefix (string). This field is optional. If you populate this field, the trunk group prefix is prepended to the trunkgrp field value, and both are displayed in statistics output and NICS output.	The trunk group prefix	A maximum of 3 letters (A–Z, a–z)

Updating the Trunk Group Prefix Table

Note This table is required only for systems that are configured to produce NICS output. For more information, see Chapter 8, "Configuring BAMS for P01 Output."

Use the TKGPREFIX tag ID to update the Trunk Group Prefix table. The Trunk Group Prefix table provides information about each trunk group prefix. This table is used by the NIC task to produce NICS output. For each record to be processed, the NIC task does a TKGPREFIX table lookup based on the trunk group prefix of the ingress and egress trunks. If the prefix is found, the ingress and/or egress trunk type is assigned. By default this table is empty.

Example:

```
mml:1>prov-add:TKGPREFIX:prefix="ABC",type=1
Billing and Measurements Server - BAMS-00 2004-10-08 16:26:06
B  COMPLD
   "TKGPREFIX"
   ;
```

In this example, the prefix field was set to ABC and the type field was set to 1 (that is, PSTN).

Note In order for you to correctly provision the Trunk Group Prefix table, each unique value of the prefix field from the Trunk Group table (TRUNKGRP tag ID) must have an entry in the Trunk Group Prefix table that defines the trunk group type (PSTN, SIP, PGW, etc.).

TKGPREFIX—Trunk Group Prefix Table

Table 5-20 lists field names and values that can be specified with the TKGPREFIX tag ID.

Table 5-20 Fields in the Trunk Group Prefix Table

Field Name	Description	Values	Range
prefix	Trunk group prefix (string)	The trunk group prefix. Each value must be unique.	A maximum of 3 letters (A–Z, a–z)
type	The trunk group prefix type (numeric)	The trunk group prefix type.	1 through 5: 1 = PSTN 2 = PGW 3 = SIP 4 = HIS 5 = HIS-NETCHAT

Updating the Zone Information Table

Use the ZONE-INFO tag ID to update the Zone Information table. This table maintains zone membership information.

Example:

```
mml:1>prov-add:ZONE-INFO:zone=456,info="zone2"
Billing and Measurements Server - BAMS-00 2004-10-08 16:27:08
B  COMPLD
   "ZONE-INFO"
   ;
```

In this example, the *zone* field was set to 456, and the *info* field was set to zone2.

ZONE-INFO Tag ID

Table 5-21 lists field names and values that can be specified with the ZONE-INFO tag ID.

Table 5-21 Fields in the Zone Information Table

Field Name	Description	Values	Range
zone	The zone ID number (numeric) **Note** Not required. Useful for reference purposes.	The zone ID	A maximum of 8 digits
info	The zone description (string) **Note** Not required. Useful for reference purposes.	The zone description	A maximum of 80 characters

CHAPTER 6

Configuring BAMS for BAF Output

Overview

This chapter describes how to configure the Cisco Billing and Measurements Server (BAMS) for Bellcore AMA (Automatic Message Accounting) Format (BAF) records. BAMS provides a mediation interface for Cisco Media Gateway Controller (MGC) systems. The purpose of BAMS is to provide enhanced data collection and processing functions, which correspond to those found in a traditional Class 4 tandem switch. BAMS does not include presentation graphics or database query functions; rather, it is designed to produce output billing and measurement feeds for downstream systems that can provide these functions.

Note If the length of the data in the CDR file obtained from the Cisco PGW 2200 is greater than that which is expected by BAMS, the data value will be truncated.

Note You enable BAF output with the NODEPARMS tag ID. For more information, see the "NODEPARMS Tag ID" section on page 5-10.

BAMS must be correctly configured with the following information:

- Billing design information
  - Billing zones
  - Zone relationships
  - Call types
  - Calling exceptions
  - Call type and structure code mapping for output records
- NPA-NXX LATA information
  - Zone assignment
  - LATA assignment
  - Toll-free prefix assignment
- Trunk group information
  - SigPathID
  - Trunk group connections

- International routing information
  - Country codes

Note Before configuring BAMS to generate billing and measurements, you should familiarize yourself with Chapters 3 to 9 of this guide.

Note BAMS performs only limited data validation on the information provisioned into its tables. It is important that you enter data carefully and understand the algorithms and their implementation.

BAMS values entered through the MML command-line interface are case sensitive; the verbs and keywords are not.

Generating BAF Records

The following section explains how BAF records are generated from Cisco MGC switch data.

Interworking of BAMS Configuration Tables

The first step in using BAMS is understanding the billing area that must be covered. Are there special cases for billing within the area? Are there banded areas with special rating? Do some number prefixes receive different treatment? Are special billing records required for these (or other) circumstances?

The answers to these questions are important in creating zones that are used to define interactions between different origination and termination NPA-NXX combinations. These interactions also determine the call types that are used to decide which BAF record to use for billing output. See the "Call Area Provisioning Example" section on page 6-7 for more details.

Once the billing area information has been gathered, it can be tabulated as shown in Table 6-1.

Table 6-1 Datafill Information Example

From			To			Results in Map Type
LATA	NPANXX	Zone	LATA	NPANXX	Zone	
1	202344	1	1	202345	1	FLAT
1	202344	1	1	202223	2	FLAT
2	202767	5	2	202333	3	MESSAGE
3	703298	8	1	202268	1	INTERLATA

Prerequisites for BAF Records

The following steps are required before Bellcore AMA Format (BAF) and Cisco Media Gateway Controller (MGC) End of Call 1110 ASCII records can be generated:

1. The Billing and Measurements Server (BAMS) collects the individual call data blocks (CDBs) generated by the Cisco MGC. This is performed by the Polling (POL) task.

Note The prefix of the files polled by the POL task is the input prefix for FMT, the initial processing task. Changing the parameter file-prefix1 or file-prefix2 in the Poll table automatically changes the input prefix of the FMT task. This change takes effect on the next startup of the system.

2. BAMS correlates the CDBs into a merged call structure. This is performed by the Correlation (COR) task.
3. Each call type is categorized. This is performed by the Augmentation (AUG) task. The AUG task is involved in the output of BAF records, as well as the BAF-to-CDR conversion task (CTB).
4. The ASCII Conversion (ASC) task outputs Cisco MGC End of Call 1110 ASCII files.

Caution In North America, if the network numbering plan does not conform to the North American Numbering Plan, BAMS will malfunction (the AGS and AGB processes will stop) while processing BAF CDRs.

Identifying Call Types

Identification of call-type information is key to generating BAF records. The call type determines the structure code and the call-type tables of the BAF record. Because the Cisco MGC generates partial information in the CDBs, additional data within the CDBs is required for the unique identification of each call scenario. BAMS provides the logic that determines the call type.

Note the following:

- Each Cisco MGC serves a set of home NPA-NXXs.
- Each output BAF record needs to be put into a call category (for example, Flat rate, Message rate, IntraLATA, InterLATA, Toll-Free, and so forth). The call category is important because it dictates the BAF structure code and call-type fields.

Perform the following steps to identify call-type categories:

Step 1 Use the ZONE-INFO tag ID to define zone IDs. (For details, see the "Updating the Zone Information Table" section on page 5-32.)

Step 2 Use the NPANXX tag ID to assign a zone ID and a LATA to each NPA-NXX. The zone narrowly defines an area within the LATA as a call category. LATAs are centrally managed, and the system assumes that you are using the correct source for the population of the LATA (for example, using Bellcore LERG tables). Note that the LERG tables are not part of BAMS. (For details, see the "Updating the NPANXX Table" section on page 5-15.)

Step 3 Use the RATING-TYPE tag ID to establish the calling relationships between zones. This tag ID determines the call category (for example, flat rate, message rate, and so on). (For details, see the "Updating the Rating Type Table" section on page 5-19.)

Step 4 Use the RATE-EXC tag ID to override certain call-category relationships defined with the NPANXX tag ID. (For details, see the "Updating the Rating Exception Table" section on page 5-18.)

Step 5 Use the TRUNKGRP tag ID to determine if a call is originating or terminating. (For details, see the "Updating the Trunk Group Table" section on page 5-29.)

Step 6 Use the MAPTYPE tag ID to specify which structure code to generate and which call type to assign each call category. (For details, see the "Updating the Map Type Table" section on page 5-5.)

Default BAF Structure Codes and Call Types

Table 6-2 defines the default BAF structure codes and call types. The table lists the call categories, call types, and answered and unanswered structure codes.

Table 6-2 BAF Structure Codes and Call Types

Call Category	Call Type	Answered Structure Code	Unanswered Structure Code
FLAT	1	502	21
MESSAGE	1	20	21
INTRALATA	6	1	2
INTERLATA_ORIG	110	625	625
INTERLATA_TERM	119	625	653
INTERLATA_TANDEM	110	625	625
TOLL_FREE	141	360	–1 (A value of –1 means "do not generate a record.")

Special Default Call Category Values

If a zone is present in the RATING-TYPE table and the originating and terminating zone are equal, the default call-category value is "FLAT". For cases in which the originating and terminating zone are not equal, the default call-category value will be "INTERLATA_ORIG".

Example of BAF Structure Codes and Call Types Created from BAF Provisioning

The following example is a formatted printout of a binary record.

```
============================== BAF RECORD ================================

(000)Record Length      : 97
(00)Hex Identifier      : aa
( 0)Structure Code      : 40625c
( 1)Call Type           : 110c
( 2)Sensor Type         : 118c
( 3)Sensor ID           : 0012345c
( 4)Record Offce Type   : 118c
( 5)Record Offce ID     : 0067890c
( 6)Connect Date        : 00713c
( 7)Timing Indicator    : 00000c
( 8)Study Indicator     : 0000000c
( 9)Off Hook Indicator  : 0c
(10)Traffic sampld Ind  : 0c
(11)Operator Action     : 0c
(12)Service Feature     : 000c
(13)Originating NPA     : 703c
(14)Originating number  : 7654321c
(15)Overseas Indicator  : 0c
```

```
(16)Terminating NPA     : 00804c
(17)Terminating number  : 7654321c
(18)Connect Time        : 1611272c
(19)Elapsed Time        : 000000021c
(57)IC/INC Prefix       : 00000c
( 6)Carrier Conn Date   : 00713c
(18)Carrier Conn Time   : 1611260c
(19)Carrier Elpsd Time  : 000000020c
(58)IC/INC Call Status  : 010c
(83)Trunk Group Number  : 38004c
(59)Route Indicator     : 1c
(85)Dial Indicator      : 0c
(60)ANI/CPN Indicator   : 3c
=============== Modules =================
Module 104 (Trunk ID)
(244)TrunkID            : 380030234c
Module 104 (Trunk ID)
(244)TrunkID            : 680042113c
Module 000 (Final Module)

============================== End of BAF Record ========================
```

Exception and Special Processing

Overriding Call Types

In some cases, a carrier might want to override a call type. Some examples include:

- Calls between a subset of NPA-NXXs in two different zones being treated differently from all other NPA-NXXs in the same zones
- InterLATA calls being treated as intraLATA calls for record-generation purposes

Use the RATE-EXC tag ID to override the default call-type classification. For more information, see the "RATE-EXC Tag ID" section on page 5-18.

Processing 7-Digit Numbers

If only 7 digits are populated in the calling and called numbers, the system acts as if each incoming trunk group visible to BAMS has a default NPA and prefixes this NPA to each 7-digit number. The full NPA-NXX is required for BAMS to perform the zone analysis. Each trunk has a default originating and terminating NPA, defined in the orgnpa and trmnpa fields, in the Trunk Group table. These fields are used only if the incoming number is 7 digits.

Processing Toll-Free Calls

A toll-free call can be identified if its dialed number prefix is contained in the Tollfree table. The default toll-free prefixes are: 800, 877, 888, 1800, 1877, and 1888. Toll-free calls have their own call-type category. A zone analysis is still performed, but the call type is overridden with the toll-free call type.

Processing International Calls

Records for international calls are processed like all other records, using zone analysis. Based on the default case, international numbers are classified as interLATA calls.

Processing LNP Calls

Local number portability (LNP) calls are processed like all other records, using zone analysis. The only special processing involved is the propagation of BAF module 720. This module is generated in response to the presence of the lnp_dip field in the CDB.

Processing Transit/Tandem Calls

InterLATA calls can originate at, terminate at, or pass through the Cisco MGC generating the BAF records. Originating records are distinguished from terminating records by the appropriate use of the BAF calltype field in the Map Type table. BAMS identifies a call as an originating interLATA call if only the egress trunk group is populated. Similarly, BAMS identifies a terminating call as an interLATA call if only the ingress trunk group is populated.

However, it is also possible for a call to transit, or pass through the Cisco MGC. If the ingress and egress fields are populated, this call can be identified. BAMS does not support BAF record types for tandem calls. Rather, BAMS separates interLATA calls into three subcategories: originating, terminating, and tandem. The Map Type table provides user-configurable record types. By default, structure code 625/call type 110 is used for interLATA tandem calls.

BAF Output Provisioning Example

The following example shows a BAF output provisioning script.

```
set-node:1:
prov-sta::srcver=”new”,dstver=”baf”,confirm
set-nodename::name="london"
prov-add:poll:host-name1=”va-lewis”,rem-dir1=”/opt/CiscoMGC/var/bam”,file-prefix1=”cdr*”,f
ile-suffix1=”.bin”,action=”R”,interval=1,timeout=5,maxtries=3,host-name2=”va-clark”,rem-di
r2=”opt/CiscoMGC/var/bam”,file-prefix2=”cdr*”,file-suffix2=”.bin”
prov-add:SWITCHINFO:sensortype="118",sensorid="999999",recoffctype="118",recoffcid="999999
"
prov-add:TRUNKGRP:trunkgrp=2002,connection=”D”,orgnpa=120,trmnpa=120,circuits=300
prov-add:TRUNKGRP:trunkgrp=2017,connection=”T”,orgnpa=121,trmnpa=120,circuits=300
prov-add:ZONE-INFO:zone=1,info=”zones 1-10 same dpc as w/2002”
prov-add:ZONE-INFO:zone=11,info=”zones 11-18 same dpc as w/2017”
prov-add:NPANXX:npanxx=120200,lata=1,zone=1
prov-add:NPANXX:npanxx=120201,lata=1,zone=1
prov-add:NPANXX:npanxx=120202,lata=1,zone=2
prov-add:NPANXX:npanxx=120203,lata=1,zone=3
prov-add:NPANXX:npanxx=120204,lata=1,zone=4
prov-add:NPANXX:npanxx=120205,lata=1,zone=5
prov-add:NPANXX:npanxx=120206,lata=1,zone=6
prov-add:NPANXX:npanxx=120207,lata=1,zone=7
prov-add:NPANXX:npanxx=120208,lata=1,zone=8
prov-add:NPANXX:npanxx=121700,lata=2,zone=11
prov-add:NPANXX:npanxx=121701,lata=2,zone=11
prov-add:NPANXX:npanxx=121702,lata=2,zone=12
prov-add:NPANXX:npanxx=121703,lata=2,zone=13
prov-add:NPANXX:npanxx=121704,lata=2,zone=14
prov-add:NPANXX:npanxx=121705,lata=2,zone=15
```

```
prov-add:NPANXX:npanxx=121706,lata=2,zone=16
prov-add:NPANXX:npanxx=121707,lata=2,zone=17
prov-add:NPANXX:npanxx=121708,lata=2,zone=18
prov-add:MAPTYPE:maptype="FLAT",calltype="1",answered="502",unanswered="20"
prov-add:MAPTYPE:maptype="MESSAGE",calltype="1",answered="20",unanswered="20"
prov-add:MAPTYPE:maptype="INTRALATA",calltype="6",answered="1",unanswered="1"
prov-add:MAPTYPE:maptype="INTERLATAORIGINATING",calltype="110",answered="625",unanswered="
625"
prov-add:MAPTYPE:maptype="INTERLATATERMINATING",calltype="119",answered="625",unanswered="
653"
prov-add:MAPTYPE:maptype="INTERLATATANDEM",calltype="110",answered="625",unanswered="625"
prov-add:MAPTYPE:maptype="TOLLFREE",calltype="141",answered="360",unanswered="-1"
prov-add:RATING-TYPE:orgzone=1,trmzone=11,ratetype="FLAT"
prov-add:RATING-TYPE:orgzone=2,trmzone=12,ratetype="MESSAGE"
prov-add:RATING-TYPE:orgzone=3,trmzone=13,ratetype="INTRALATA"
prov-add:RATING-TYPE:orgzone=4,trmzone=14,ratetype="INTERLATA"
prov-add:RATING-TYPE:orgzone=5,trmzone=15,ratetype="INTERLATA"
prov-add:RATING-TYPE:orgzone=6,trmzone=16,ratetype="FLAT"
prov-add:RATING-TYPE:orgzone=7,trmzone=17,ratetype="FLAT"
prov-add:RATING-TYPE:orgzone=7,trmzone=18,ratetype="FLAT"
prov-add:P01FILTER:answered=1,noanswer=1,busy=1,other=1
prov-add:NODEPARMS:activate=1,statoutput=0,bafoutput=1,asciibafoutput=1,asciioutput=0,
lookupinfo=1,bafinfo=1,dynamicaccumes=0,sup-zero-counts=0,interval-minutes=15,
nailed-cfg=0,p01output=0,p01prefix="p01_",enable-h323=0,extasciioutput=0,nicsoutput=0,
bin1110output=0
prov-add:TOLLFREE:digits=1800
prov-add:TOLLFREE:digits=800
prov-add:TOLLFREE:digits=1888
prov-add:TOLLFREE:digits=888
prov-add:TOLLFREE:digits=1877
prov-add:TOLLFREE:digits=877
prov-add:SKIPCDB:id=1020
prov-dply::
```

Call Area Provisioning Example

Figure 6-1 shows a hypothetical BAMS provisioning scenario. A fictional metropolitan city, River City, has been designed to include multiple call zones, local access and transport areas (LATAs), and numerous central offices.

Designing a Billing Plan

Before any provisioning is done, a plan must be developed that determines how BAMS constructs billing records for the implementation in question. Many different methods can be used to accomplish a desired result. BAMS is designed to allow a customer to design billing applications that are unique to any configuration.

Caution Customer billing-system expert inputs are necessary for anyone wishing to configure BAMS for BAF. The information in the following sections is provided by way of example only.

Assumptions Used in This Example

In Figure 6-1, some origination points can call to any point within the metropolitan area, with no toll charged. A call type is determined by BAMS as a relationship between zones. BAMS can be configured to specify that a certain call type produces, or does not produce, a billing record. Obviously, the downstream billing system can determine which records are billable and which are informational only.

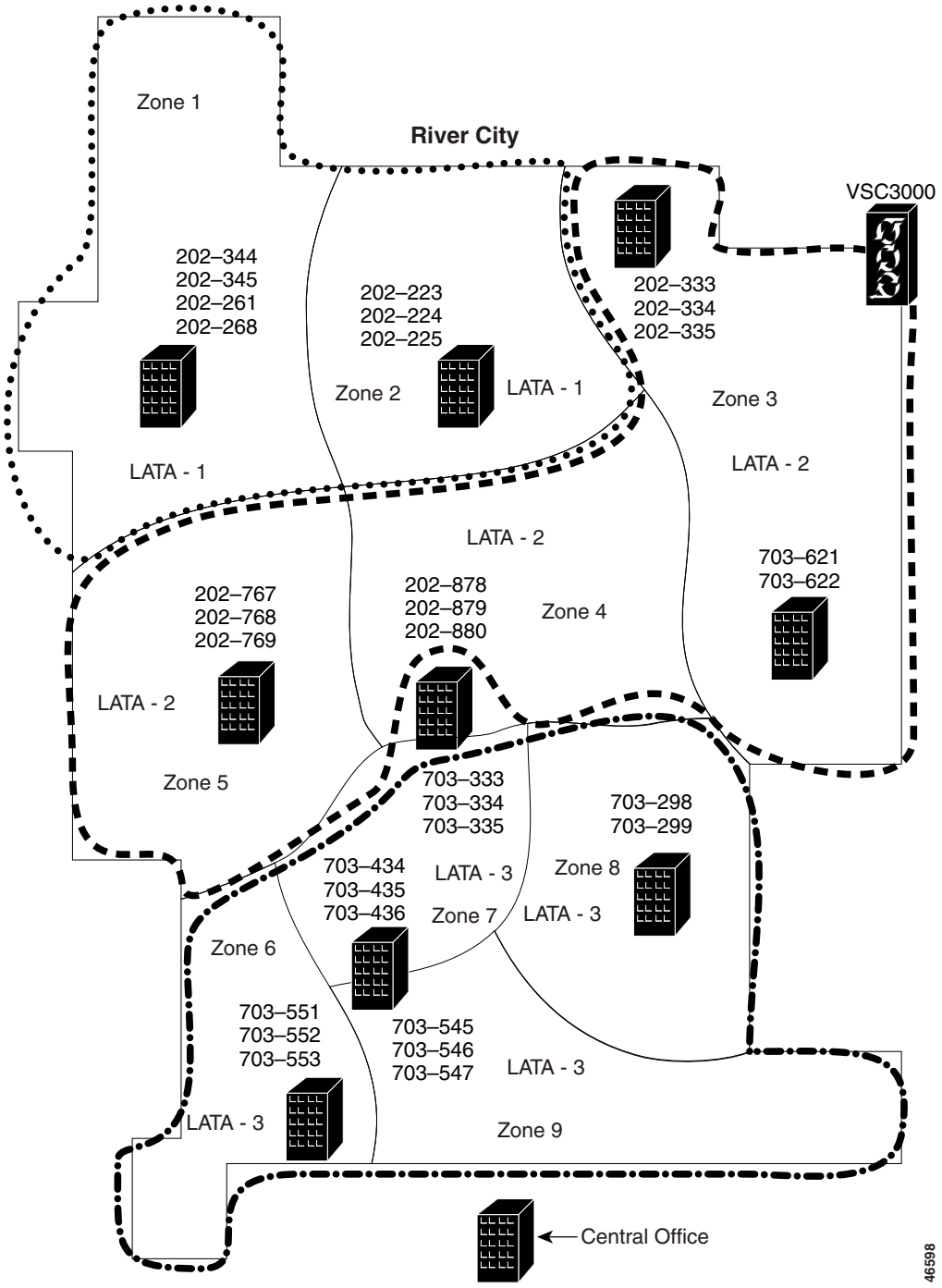
Keep in mind the following general information about the metropolitan area shown in Figure 6-1:

- There are three LATAs, labeled 1, 2, and 3.
- Any zone that is adjacent to another zone within the same LATA is marked as FLAT.
- Calls that cross LATAs are marked as interLATA.
- A call that originates and terminates in zones that are not adjacent, but that are in the same LATA, is designated as MESSAGE to indicate a banded billing rate.

From the North American Bell System, the concept of LATAs is a traditional boundary concept that is supported by BAMS. These are geographic regions, defined by NPANXX containment. BAF output sometimes requires LATA assignment for certain record types. Because BAMS is not geared toward local exchange routing guide (LERG) data, a downstream system could use the LATA designation for other purposes.

LERG data is not resident on BAMS.

Figure 6-1 River City Metropolitan Example



Zone Membership

Each NPANXX combination that is supported within the metropolitan area must be a member of a zone. Otherwise, the unknown call origination or termination point is considered to be out of the calling area, and the call is tagged as INTERLATAORIGINATING or INTERLATATERMINATING.

Note Zoning provides a mechanism for differentiating between rating types (for example, flat rate and message calls).

In the example in Figure 6-1, the following commands would be used to assign zone membership in zone 1:

```
prov-add:NPANXX:npanxx=202344,zone=1,lata=1
prov-add:NPANXX:npanxx=202345,zone=1,lata=1
prov-add:NPANXX:npanxx=202261,zone=1,lata=1
prov-add:NPANXX:npanxx=202268,zone=1,lata=1
```

As a further example, the following commands would be used to assign zone membership in zone 9:

```
prov-add:NPANXX:npanxx=703545,zone=9,lata=3
prov-add:NPANXX:npanxx=703546,zone=9,lata=3
prov-add:NPANXX:npanxx=703547,zone=9,lata=3
```

Free Calling Between Zones

Assume that the billing design in this example states that, if a call originates in one zone and is intended for an adjacent zone, the call type should be FLAT. The following commands implement FLAT for adjacent zones. Note that the calling plan must be implemented in both directions, because zone A might be allowed to call zone B for free, but not the other way around.

```
prov-add:RATING-TYPE:ORGZONE=1,TRMZONE=2,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=2,TRMZONE=1,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=1,TRMZONE=5,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=5,TRMZONE=1,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=1,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=1,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=2,TRMZONE=3,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=3,TRMZONE=2,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=2,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=2,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=3,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=3,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=3,TRMZONE=8,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=8,TRMZONE=3,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=8,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=8,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=8,TRMZONE=7,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=7,TRMZONE=8,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=8,TRMZONE=9,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=9,TRMZONE=8,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=7,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=7,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=5,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=4,TRMZONE=5,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=6,TRMZONE=5,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=5,TRMZONE=6,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=6,TRMZONE=7,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=7,TRMZONE=6,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=6,TRMZONE=9,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=9,TRMZONE=6,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=9,TRMZONE=7,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=7,TRMZONE=9,RATETYPE="FLAT"
```

Banded Message Rates

In Figure 6-1, many zones are not adjacent to other zones. The following commands set up MESSAGE rate call types for calls from zone 6 and zone 8, which are not adjacent to each other but are in the same LATA. In the example shown in Figure 6-1, many zone relationships of this type could occur.

```
prov-add:RATING-TYPE:ORGZONE=6,TRMZONE=8,RATETYPE="MESSAGE"
prov-add:RATING-TYPE:ORGZONE=8,TRMZONE=6,RATETYPE="MESSAGE"
```

Configuring Trunk Groups for InterLATA Calls

Calls that cross LATA boundaries are normally considered toll calls, unless the zones are adjacent or an exception case is entered with the RATE-EXC tag ID. Because the default call type is interLATA (when either originating or terminating NPANXX is not found in the Zone Information table), the only provisioning required is on the ingress or egress trunk group.

For BAF billing records, the connectivity of the ingress or egress trunk determines the BAF direction of the calls. When configured properly, BAMS marks interLATA calls as INTERLATAORIGINATING or INTERLATATERMINATING. Because many telephone plan areas require only 7-digit dialing, the Trunk Group table is also used to populate default originating and terminating NPAs when fewer than 10 digits are received for the A number or the B number. Finally, each trunk group has multiple circuits, and the count of the number of circuits is captured in the Trunk Group table for different measurement calculations.

Although there are many trunk groups, in the example shown in Figure 6-1, we configure only a few trunk groups for Direct (to an end office) and a few trunk groups for Tandem (to an access tandem type office). All trunk groups must be provisioned in the table for proper BAMS operation. Only a small sample is shown here:

```
prov-add:TRUNKGRP:trunkgrp=123,connection="D",orgnpa=703,trmnpa=202,circuits=72
prov-add:TRUNKGRP:trunkgrp=124,connection="T",orgnpa=703,trmnpa=201,circuits=24
prov-add:TRUNKGRP:trunkgrp=223,connection="D",orgnpa=703,trmnpa=202,circuits=72
prov-add:TRUNKGRP:trunkgrp=227,connection="T",orgnpa=703,trmnpa=408,circuits=96
```

Configuring Adjacent Area Codes

Adjacent area codes present no problem to BAMS, because the key for the various lookup tables is based on the entire NPA-NXX (6-digit) string. Trunk groups must be provisioned, and, in the case of fewer than 10-digit dialing, they must indicate default originating and terminating NPAs.

Configuring IntraLATA Calls

Cases might occur where a billing design does not use FLAT, or designates calls for adjacent zones to something other than flat rate so that the downstream system can treat those calls differently. The following example shows configuration for intraLATA calls between zones within the same LATA:

```
prov-add:RATING-TYPE:orgzone=5,trmzone=4,ratetype="INTRALATA"
prov-add:RATING-TYPE:orgzone=5,trmzone=6,ratetype="INTRALATA"
prov-add:RATING-TYPE:orgzone=5,trmzone=6,ratetype="INTRALATA"
prov-add:RATING-TYPE:orgzone=5,trmzone=6,ratetype="INTRALATA"
```

Configuring Toll-Free Calls

Different countries have different designations for toll-free dialing. For this reason, BAMS maintains the Tollfree table for the NPANXX combinations that are considered toll free.

Note Any 800, 888, or 877 NPA results in a toll-free BAF record, overriding any zone or rating type designations.

Standard North American entries are provided here:

```
prov-add:TOLLFREE:digits=1800
prov-add:TOLLFREE:digits=1877
```

Configuring a Metropolitan Calling Plan

Suppose that one set of NPANXXs could call FLAT to any zone in Figure 6-1. This could happen if you set up an overlay zone that covers the entire area. The LATA field would be set up for each NPANXX, as appropriate.

To configure a metropolitan calling plan, perform the following steps:

Step 1 Set up zone membership, zone 10 for the overlay:

```
prov-add:NPANXX:npanxx=202388,zone=10,lata=1
prov-add:NPANXX:npanxx=703488,zone=10,lata=3
```

Step 2 Set up zone relationships so that the super-zone can call any other zone as FLAT, and any other zone can call the super-zone as FLAT.

```
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=1,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=2,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=3,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=4,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=5,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=6,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=7,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=8,RATETYPE="FLAT"
prov-add:RATING-TYPE:ORGZONE=10,TRMZONE=9,RATETYPE="FLAT"
```

Configuring for a Rating Exception

Sometimes cases occur in which a particular NPANXX calling another NPANXX needs to be classified as a rating exception. For these cases, you can place an entry in the Rating Exception table, as follows:

```
prov-add:RATE-EXC:orgnpanxx=202344,trmnpanxx=703299,ratetype="FLAT"
prov-add:RATE-EXC:orgnpanxx=703299,trmnpanxx=202344,ratetype="FLAT"
```

BAMS Provisioning Worksheets

Start from a worksheet for provisioning BAF by providing the following information:

Step 1 Determine if the network is switched or is nailed up.

Note For a nailed up (Cisco SC2200) connection, define the field values in the Nailed Connection table using the SIGPATH tag ID. The fields are SIGPATH and TRUNKGRP. The SIGPATH parameter is a hex entry and should be prefixed by 0x.

Step 2 Enter switch data in the Switch Information table. (Refer to the "SWITCHINFO Tag ID" section on page 5-22.)

Step 3 Define the Trunk Group table. You need to determine if each trunk is tandem or direct, default NPA for originating, or default NPA for terminating.

Note Define the Zone Information table by entering a number and a description for each zone.

Step 4 Assign an NPANXX to each zone. (Refer to the "NPANXX Tag ID" section on page 5-15.)

Step 5 Assign a rate type (zone to zone). (Refer to the"RATING-TYPE Tag ID" section on page 5-19.)

Step 6 Define the structure code/call type for every rate type. (Refer to the"MAPTYPE Tag ID" section on page 5-5.)

Step 7 Define the toll-free NPAs. (Refer to the"TOLLFREE Tag ID" section on page 5-29.)

Step 8 Define rating exception cases. (Refer to the"RATE-EXC Tag ID" section on page 5-18.)

Step 9 Define the country codes. (Refer to the"COUNTRY Tag ID" section on page 5-5.)

BAF Configuration Parameters

Updating BAF Structure Codes

Use the MAPTYPE tag ID to update selected BAF structure codes, such as category codes and call types. Refer to the "MAPTYPE Tag ID" section on page 5-5 for field name descriptions and values.

Example:

```
prov-rtrv:MAPTYPE:
prov-ed:MAPTYPE:maptype=map type mnemonic,answered=new structure code,unanswered=new
structure code,calltype=new BAF call type number
```

Adding a New Zone ID

Use the ZONE-INFO tag ID to add new zone IDs. Refer to the "ZONE-INFO Tag ID" section on page 5-33 for field name descriptions and values.

Example:

```
prov-rtrv:ZONE-INFO:
prov-add:ZONE-INFO:zone=<zone number>,desc="<description>"
```

Adding a New NPANXX (Zone Membership)

Use the NPANXX tag ID to add new NPANXX information. Refer to the "NPANXX Tag ID" section on page 5-15 for field name descriptions and values.

Example:

```
prov-rtrv:NPANXX:npanxx=<areacode and exchange 6 digits>
prov-add:NPANXX:npanxx=<6digits>,lata=<latanumber>,zone=<zonenumber>
```

Configuring BAMS for ASCII Output and Measurements

Overview

This chapter describes how to configure the Cisco Billing and Measurements Server (BAMS) for ASCII output and measurements.

Note If the length of the data in the CDR file obtained from the Cisco PGW 2200 is greater than that which is expected by BAMS, the data value will be truncated.

Note You enable ASCII and/or measurements output with the NODEPARMS tag ID. For more information, see the "NODEPARMS Tag ID" section on page 5-10.

ASCII Output

Many mediation systems depend on input data that is preformatted as ASCII. You can configure BAMS to create ASCII records. In addition to requiring ASCII records for billing purposes, the same mediation systems might require bearer-level measurements that are produced on BAMS. In order to satisfy both requirements, BAMS can produce ASCII output that mirrors what is currently produced on the Cisco MGC.

To enable ASCII record generation on BAMS, set the parameter asciioutput in the NODEPARMS tag ID to a value of 1 for ANSI output or 2 for ITU output. See Chapter 2, "Setup and Installation," for further details.

BAMS creates ASCII 1110 records when a call has terminated. Filenames are identical to the names that are currently used on the Cisco MGC and BAMS, with the suffix changed to "csv" (that is, comma-separated values). For example, if a Tag-Length Value (TLV) billing file on the Cisco MGC is named cdr_20000714195130_000777.bin on the Cisco MGC, any file that is produced as a result of processing this file is named cdr_20000714195130_000777.csv. Some files are created with zero records, when no calls terminate during the processing of a TLV file from the Cisco MGC.

Note The Cisco MGC (PGW 2200) uses local time to generate CDR file names. The Cisco BAMS uses UTC to generate both measurement and billing output file names. For example, a file generated by the Cisco MGC in the Eastern Standard Timezone named cdr_200404201300210412_123456.bin, the ASCII output file generated by the Cisco BAMS would be named cdr_200404201800210412_123456.csv.

All fields for records 1110 and 1060 are written in comma-separated format. Tag 6000 (Virtual Switch Controller [VSC] ID) is collected from the 1090 Call Detail Block (CDB). The fields are written in the exact order specified in Table 7-1. When a field was not created by the Cisco MGC, because the field did not apply to the type of call recorded, a blank field or zero field is inserted as a place holder.

A normalized set of fields, or CDEs, is written for every record type, even if that record type does not contain a particular field. In this way, the end user can quickly import the CDR ASCII file into a database with standard data-importing tools. If more fields are added at a later time, they are appended to the end of the record, so you can ignore them, or modify their database import structure by simply adding fields to the end of the layout.

The accuracy selection for time points is configurable in seconds or milliseconds. When you are using ASCII output from BAMS, either choice is possible, because a place holder for each type has been provided in the ASCII formatting layout. For each type of time point, two entries are contained in the output format, one for seconds and one for milliseconds of granularity.

A downstream mediation or billing system can parse these ASCII records easily. Each record is prefixed in the ASCII file with a record identifier field. For example, an 1110 record would begin this way: 1110,1234,5678,2222

Table 7-1 ASCII Output Field Ordering: ANSI and ITU

Field	Tag	Description	Format
1	1060, 1090, or 1110	CDB (record) identifier	Decimal
2	4000	CDB Version	Decimal
3	4001	CDB Timepoint	Seconds (decimal)
4	4002	Call Reference ID	Hexadecimal
5	4003	IAM Timepoint	Seconds (decimal)
6	4004	ACM Timepoint	Seconds (decimal)
7	4005	ANM Timepoint	Seconds (decimal)
8	4008	Originating Trunk Group	Decimal
9	4009	Originating Member	Decimal
10	4010	Calling Number	Text
11	4011	Charged Number	Text
12	4012	Dialed Number	Text
13	4014	Called Number	Text
14	4015	Terminating Trunk Group	Decimal
15	4016	Terminating Member	Decimal
16	4028	First Release Source	Decimal
17	4031	VSC Info Field	Hexadecimal

Table 7-1 ASCII Output Field Ordering: ANSI and ITU (continued)

Field	Tag	Description	Format
18	4100	IAM Timepoint rcvd ms	Seconds, milliseconds
19	4101	IAM Timepoint sent ms	Seconds, milliseconds
20	4102	ACM Timepoint rcvd ms	Seconds, milliseconds
21	4103	ACM Timepoint sent ms	Seconds, milliseconds
22	4104	ANM Timepoint rcvd ms	Seconds, milliseconds
23	4105	ANM Timepoint sent ms	Seconds, milliseconds
24	4106	First REL Timepoint ms	Seconds, milliseconds
25	4107	Second REL Timepoint ms	Seconds, milliseconds
26	4108	RLC Timepoint rcvd ms	Seconds, milliseconds
27	4109	RLC Timepoint sent ms	Seconds, milliseconds
28	2000	ANSI Calling Party Category	Decimal
29	2001	ANSI User Service Information	Hex
30	2003	ANSI Calling Number Nature of Address	Decimal
31	2004	ANSI Charged Number Nature of Address	Decimal
32	2005	ANSI Dialed Number Nature of Address	Decimal
33	2007	ANSI Called Number Nature of Address	Decimal
34	2008	ANSI Reason Code	Hexadecimal
35	2013	ANSI Transit Network Selection	Hexadecimal
36	2015	ANSI Carrier Selection Parameter	Decimal
37	3000	ITU Calling Party Category	Decimal
38	3001	ITU User Service Information	Hex
39	3003	ITU Calling Number Nature of Address	Decimal
40	N.A.	Blank	N/A
41	3005	ITU Dialed Number Nature of Address	Decimal
42	3007	ITU Called Number Nature of Address	Decimal
43	3008	ITU Reason Code	Hexadecimal
44	6000	VSC ID	Text
45	Derived	Subscriber Duration (4106 minus (whichever is greater: 4104 or 4105))	Seconds, milliseconds
46	Derived	Network Usage Duration (whichever is greater: 4108 or 4109) minus (whichever is less: 4100 or 4101))	Seconds, milliseconds
47	4060	Redirecting Number	Decimal
48	5000	Global Call Id	Text

Correlation of Billing Information

BAMS collects information about calls with the various Primary Interexchange Carrier (PIC) CDBs; therefore, at any point in time, some calls are terminating and some calls are just being set up. BAMS must collect the information from the various records, just as with a BAF output, and then output records only upon receipt of either a long-call duration record or a call termination event.

Files and Records

The ASCII file contains the following records:

090—File Header

110—End of Call Record

060—Long Call Duration Event

These files and records are written as if they had been produced from the Cisco MGC. The most significant information from the 1090 records is the VSC ID, and the time stamp and other information on that record should match corresponding entries in the 1090 record from the file that triggered the current file processing.

Extendable ASCII Output

Extendable ASCII is based on ASCII output, but it includes four supplementary fields. These fields are appended to the end of a standard comma-separated value ASCII record, to create a new extendable ASCII record.

To enable extendable ASCII record generation on BAMS, use the NODEPARMS tag ID to set the extasciioutput parameter to a value of 1 for ANSI output, or 2 for ITU output. See "Configuring Node Parameters" section on page 2-12 for further details. For file-naming conventions, see the "Extendable ASCII Filename Elements" section on page A-6.

Extendable ASCII Output Format

Table 7-2 provides the complete list of the Extendable ASCII output fields. The additional fields (beginning with Field 49) are appended to a standard ASCII record. Each record is prefixed in the extendable ASCII file with a record identifier field (for example, an 1110 record would begin like this: 1110,1234,5678,2222). For details about the first 48 fields in an extendable ASCII record, see the "ASCII Output" section on page 7-1.

Table 7-2 Extendable ASCII Output Field Ordering: ANSI and ITU

Field	Tag	Description	Format
1	1060, 1090 or 1110	CDB (record) Identifier	Decimal
2	4000	CDB Version	Decimal
3	4001	CDB Timepoint	Seconds (Decimal)
4	4002	Call Reference ID	Hexadecimal
5	4003	IAM Timepoint	Seconds (Decimal)
6	4004	ACM Timepoint	Seconds (Decimal)
7	4005	ANM Timepoint	Seconds (Decimal)
8	4008	Originating Trunk Group	Decimal
9	4009	Originating Member	Decimal
10	4010	Calling Number	Text
11	4011	Charged Number	Text
12	4012	Dialed Number	Text
13	4014	Called Number	Text
14	4015	Terminating Trunk Group	Decimal
15	4016	Terminating Member	Decimal
16	4028	First Release Source	Decimal
17	4031	VSC Info Field	Hex
18	4100	IAM Timepoint rcvd ms	Seconds, Milliseconds
19	4101	IAM Timepoint sent ms	Seconds, Milliseconds
20	4102	ACM Timepoint rcvd ms	Seconds, Milliseconds
21	4103	ACM Timepoint sent ms	Seconds, Milliseconds
22	4104	ANM Timepoint rcvd ms	Seconds, Milliseconds
23	4105	ANM Timepoint sent ms	Seconds, Milliseconds
24	4106	First REL Timepoint ms	Seconds, Milliseconds
25	4107	Second REL Timepoint ms	Seconds, Milliseconds
26	4108	RLC Timepoint rcvd ms	Seconds, Milliseconds
27	4109	RLC Timepoint sent ms	Seconds, Milliseconds
28	2000	ANSI Calling Party Category	Decimal
29	2001	ANSI User Service Information	Hexadecimal

Table 7-2 Extendable ASCII Output Field Ordering: ANSI and ITU (continued)

Field	Tag	Description	Format
30	2003	ANSI Calling Number Nature of Address	Decimal
31	2004	ANSI Charged Number Nature of Address	Decimal
32	2005	ANSI Dialed Number Nature of Address	Decimal
33	2007	ANSI Called Number Nature of Address	Decimal
34	2008	ANSI Reason Code	Hexadecimal
35	2013	ANSI Transit Network Selection	Hexadecimal
36	2015	ANSI Carrier Selection Parameter	Decimal
37	3000	ITU Calling Party Category	Decimal
38	3001	ITU User Service Information	Hexadecimal
39	3003	ITU Calling Number Nature of Address	Decimal
40	3004	ITU Charged Number Nature of Address	Decimal
41	3005	ITU Dialed Number Nature of Address	Decimal
42	3007	ITU Called Number Nature of Address	Decimal
43	3008	ITU Reason Code	Hexadecimal
44	6000	VSC ID	Text
45	Derived	Subscriber Duration (4106 minus (whichever is greater: 4104 or 4105))	Seconds, Milliseconds
46	Derived	Network Usage Duration ((whichever is greater: 4108 or 4109) minus (whichever is less: 4100 or 4101))	Seconds, Milliseconds
47	4060	Redirecting Number	Text
48	5000	Global Call Id	Hexadecimal
49	2002	ANSI Originating Line Information	Hexadecimal
50	4201	Ingress SIP URL	1 to 256 characters
51	4202	Egress SIP URL	1 to 256 characters
52	4203	SIP Callid	1 to 256 characters
53	4204	Source IP Address	String
54	4205	Ingress Media Device Address	String
55	4206	Egress Media Device Address	String
56	4207	Initial Codec	String
57	4208	Final Codec	String
58	4209	Ingress Media Device Port Number	String
59	4210	Egress Media Device Port Number	String
60	4052	Originating Gateway Primary Select	Decimal integer
61	4053	Terminating Gateway Primary Select	Decimal integer
62	4061	Tariff Rate	Decimal integer
63	4062	Scale Factor	Decimal integer

Table 7-2 Extendable ASCII Output Field Ordering: ANSI and ITU (continued)

Field	Tag	Description	Format
64	4063	Test Line Indicator	Decimal integer
65	4078	Charge Band Number	Decimal integer
66	4079	Furnish Charging Number	String
67	4080	Original Called Number	String
68	4081	T.38 Fax Call	String
69	4082	Charge Unit Number	Decimal integer
70	4034	Ingress Originating Point Code	Decimal
71	4035	Ingress Destination Point Code	Decimal
72	4036	Egress Originating Point Code	Decimal
73	4037	Egress Destination Point Code	Decimal
74	4046	Ingress Packet Info	Hexadecimal
75	4047	Egress Packet Info	Hexadecimal
76	4068	Ingress BearChanId	Decimal
77	4072	Egress BearChanId	Decimal
78	4083	Charge Indicator	Decimal
79	4084	Outgoing Calling Party Number	Text
80	4085	MCID Request Indicator	Decimal
81	4086	MCID Response Indicator	Decimal
82	4087	Ingress MGCP DLCX (Delete Connection) return code value	Decimal
83	4088	Egress MGCP DLCX (Delete Connection) return code value	Decimal
84	4089	Network Translated Address Indicator	Decimal
85	4090	Reservation Request Accepted	Decimal
86	4091	Reservation Request Error Count	Decimal
87	4092	ATM Ingress Configured Profile	Text
88	4093	ATM Egress Configured Profile	Text
89	4094	ATM Negotiated Profile	Text
90	4095	Route List Name	Text
91	4096	Route Name	Text
92	4097	MGCP Script Response String	Text
93	4211	Originating VPN ID	Text
94	4212	Terminating VPN ID	Text
95	4213	Meter Pulses Received	Decimal
96	4214	Meter Pulses Sent	Decimal
97	4215	Charge Tariff Info	Hexadecimal
98	4216	Advice of Charge Indicator	Decimal

Table 7-2 Extendable ASCII Output Field Ordering: ANSI and ITU (continued)

Field	Tag	Description	Format
99	4217	Short Call Indicator	Decimal
100	4218	Charge Limit Exceeded	Decimal
101	4219	CALL Recovered Indication	Decimal
102	4220	Partial CLI	Hexadecimal
103	4221	Service Activation	Decimal
104	4222	PRI AOC Invoke Type	Decimal
105	4223	PRI AOC – S Charge Information	Hexadecimal
106	4224	PRI AOC – D Charge Information	Hexadecimal
107	4225	PRI AOC – E Charge Information	Hexadecimal
108	4226	PRI AOC Invoke Failure	Hexadecimal

Operational Measurements

Defining Bucket Size

The bucket size refers to the measurements interval that is set up during installation through use of the NODEPARMS tag ID.

To change the system settings for the measurement intervals, for example, to take measurements every 5 minutes, you can update the Node Parameters table.

Refer to the "Updating the Node Parameters Table" section on page 5-10 for further details.

Setting Up the TCA Table

Use the TCA-TBL tag ID to set up the Threshold Crossing Alarms (TCA) table. The table is used to define the types of measurements that BAMS traps and makes available to downstream processes. Refer to the "TCA-TBL Tag ID" section on page 5-24 for TCA-TBL tag ID field names, and to Chapter 11, "Obtaining Measurements," for required values and a detailed description of the measuring process.

When a measurement threshold is crossed, an alarm is generated. This alarm is automatically cleared when the measurement statistics fall below the boundary. You set up the conditions to be monitored using the TCA-TBL tag ID. The following example shows which parameters you would set to achieve a specific result:

Objective Generate a threshold alarm for trunk group 008 that exceeds 10 egress call attempts during a specified interval.

Parameters

- id/trkgrp = 0
- Egress call attempt condition = 3 (greater than)
- Egress call attempt value = 10

Result A threshold alarm is generated every time the call attempts exceed 10 in the interval. This alarm is set only once until it is cleared. It is cleared when the number of egress call attempts in the measurement interval drops below 10.

Note Set up only those measurement conditions in the TCA table that you are interested in monitoring (that is, not all table values need to be provisioned).

Configuring BAMS for P01 Output

Overview

This chapter describes how to configure the Cisco Billing and Measurements Server (BAMS) for P01 billing records. You enable P01 output with the NODEPARMS tag ID. For more information, see the "NODEPARMS Tag ID" section on page 5-10.

Note If the length of the data in the CDR file obtained from the Cisco PGW 2200 is greater than that which is expected by BAMS, the data value will be truncated.

P01 Output

P01 output files are created by the P01 task. These files are stored in the p001 subdirectory of the data directory. The file layout of the P01 output format includes a header record, call detail record(s), and a trailer record. This data format is binary and is not viewable by a standard text viewer. The P01 file-naming conventions are explained in "File-Naming Conventions" section on page A-4.

Customized MML Parameters

By default, BAMS generates ASCII output records when processing MGC input data. To turn on the generation of P01 data files, set the p01 output parameter in the Node Parameters table to 1.

The P01 filename format is based on the filename of the raw data files from the MGC:

```
<p01prefix><raw MGC data filename>
```

The default P01 prefix is p01_. You can change this prefix by modifying the p01prefix parameter in the Node Parameters table. For example, suppose that an input MGC file is named cdr_200012252359_012345.bin. The output P01 file is named p01_cdr_200012252359_012345.bin. Setting this parameter to NULL creates an output filename that is the same as the input filename.

Examples:

Change the filename prefix of P01 files to "info_":

```
$ mml

Copyright (C) 1998-2004, Cisco Systems, Inc.
mml:1>prov-sta::srcver=active,dstver=example,confirm
Billing and Measurements Server - BAMS-00 2004-07-27 13:59:06
```

```
B  COMPLD
   ;
mml:1>prov-ed:nodeparms:p01prefix="info_"
Billing and Measurements Server - BAMS-00 2004-07-27 13:59:37
B  COMPLD
   "NODEPARMS"
   ;
mml:1>prov-rtrv:nodeparms:
Billing and Measurements Server - BAMS-00 2004-07-27 14:00:03
B  RTRV
NODEPARMS:statoutput=1,bafoutput=0,asciioutput=0,lookupinfo=1,bafinfo=0,p01output=1,

p01prefix="info_""
   ;
mml:1>prov-dply::
Billing and Measurements Server - BAMS-00 2004-07-27 14:00:23
B  COMPLD
   ;
mml:1>quit
```

Turn off the generation of P01 files:

```
$ mml

Copyright (C) 1998-2004, Cisco Systems, Inc.
mml:1>prov-sta::srcver=active,dstver=example,confirm
Billing and Measurements Server - BAMS-00 2004-07-27 13:59:06
B  COMPLD
   ;
mml:1>prov-ed:nodeparms:p01output=0
Billing and Measurements Server - BAMS-00 2004-07-27 13:59:57
B  COMPLD
   "NODEPARMS"
   ;
mml:1>prov-rtrv:nodeparms:
Billing and Measurements Server - BAMS-00 2004-07-27 14:00:03
B  RTRV
NODEPARMS:statoutput=1,bafoutput=0,asciioutput=0,lookupinfo=1,bafinfo=0,p01output=0,

p01prefix="info_""
   ;
mml:1>prov-dply::
Billing and Measurements Server - BAMS-00 2004-07-27 14:00:23
B  COMPLD
   ;
mml:1>quit
```

Filtering Calls Based on Cause Codes

You can configure the P01 program to filter out certain types of calls based on the cause code of the original MGC data record. The P01FILTER tag ID has four parameters that define the call type: ANSWERED, NOANSWER, BUSY, and OTHER. You can set these parameters to 1 to output CDR records of the specified call type, or to 0 to filter the specified records from the output. The call types are defined by the cause code values (see Table 8-1).

Table 8-1 P01 Call Types and Cause Codes

Cause Code	P01 Call Type
16, 31	ANSWERED
19	NOANSWER
17	BUSY
ALL OTHER CODES	OTHER

You can use the **prov-ed** command in conjunction with the P01FILTER tag ID to edit the default filtering values. For more information about configuring P01 filtering, see the "Updating the P01 Filter Table" section on page 5-16.

The following example shows how to turn off all output except for answered calls:

```
$ mml
mml:1> prov-sta::srcver=active,dstver=example,confirm
Billing and Measurements Server - BAMS-00 2004-07-27 14:02:02
B  COMPLD
   ;
mml:1>prov-rtrv:p01filter:
Billing and Measurements Server - BAMS-00 2004-07-27 14:02:12
B  RTRV
   "P01FILTER:answered=1,noanswer=1,busy=1,other=1"
   ;
mml:1>prov-ed:p01filter:noanswer=0,busy=0,other=0
Billing and Measurements Server - BAMS-00 2004-07-27 14:02:42
B  COMPLD
   "P01FILTER"
   ;
mml:1>prov-rtrv:p01filter:
Billing and Measurements Server - BAMS-00 2004-07-27 14:02:51
B  RTRV
   "P01FILTER:answered=1,noanswer=0,busy=0,other=0"
   ;
mml:1>prov-dply::
Billing and Measurements Server - BAMS-00 2004-07-27 14:02:56
B  COMPLD
   ;
```

P01 Data Format

Table 8-2, Table 8-3, and Table 8-4 define the header, call detail, and trailer data records of the P01 output format. The field name, length, description, format/contents, and MGC mapping for the P01 header record are described in Table 8-2.

Table 8-2 PO1 Header Record

Field	Field Name	Length (Bytes)	Description	Format/Contents	MGC Mapping
1	Length	2	Record length	BCD /(= 60)	
2	Record identifier	1	Record identifier	BCD/(= 00)	
3	Switch ID	30	Switch name	ASCII, left aligned filled with spaces	Tag 6000

Table 8-2 PO1 Header Record (continued)

4	Creation date	4	Logical block creation date	BCD	Tag 4001 is converted from UTC to Local time.
5	Creation time	3	Logical block creation date	BCD	Tag 4001 is converted from UTC to Local time.
6	Spare	20	Spare	(filled with HEX 0xFF)	

The field name, length, description, format/contents, and MGC mapping for the P01 call detail record are described in Table 8-3.

Table 8-3 P01 Call Detail Record

Field	Field Name	Length (Bytes)	Description	Format/Contents	MGC Mapping
1	Length	2	Record length	BCD (= 110)	Fixed
2	Record type	1	Record type	BCD (= 11)	Fixed
3	Called number	12	Called number	BCD (left aligned, filler F)	Tag 4012
4	Called number type	2	Called number type	BCD	Derived from tag 3005 (Dialed Number Nature of Address) and populated as follows: • NoA = international: “0” • NoA = national: “1” • NoA = subscriber: “2”
5	Calling category	2	Calling category	BCD	Derived from tag 3000 (Calling Party Category) populated as follows: • CPC unknown: “0” • Operator, language French: “1” • Operator language English: “2” • Operator language German: “3” • Operator language Russian: “4” • Operator language Spanish: “5” • … • Payphone: “15”
6	Conversation start date	4	Conversation start date	BCD, format yyyymmdd	Tag 4104 or 4105 is converted from UTC to Local time.
7	Conversation start time	4	Conversation start time	BCD, format hhmmsst	Tag 4104 or 4105 is converted from UTC to Local time.
8	Conversation duration	5	Conversation duration (tenths of seconds)	BCD	Derived, 4106 minus 4104 or 4105

Table 8-3 P01 Call Detail Record (continued)

Field	Field Name	Length (Bytes)	Description	Format/Contents	MGC Mapping
9	Call type	2	Call typology	BCD	If (Tag 4069 = 1) then callType = 7; else callType = 4
10	Destination	12	Real number used for call	BCD (left-aligned, filler “F”)	Tag 4014
11	Destination type	2	Destination number type	BCD	Derived from tag 3007 (Called Number Nature of Address) and populated as follows: • NoA = international: “0” • NoA = national: “1” • NoA = subscriber: “2” • NoA = other values: “1”
12	Incoming trunk	13	Incoming trunk ID	ASCII (left-aligned, filled with spaces)	BAMS augmented
13	Outgoing trunk	13	Outgoing trunk ID	ASCII	This field is populated according to the following conditions: • For switched mode; that is, if NODEPARMS nailed-cfg = 0, this field is populated with the Egress Trunk Group ID, which is derived from tag 4015, but only when the trunk group is present in the trunk group table. The field is left blank if the trunk group is not present in the trunk group table. • For nailed mode; that is, if NODEPARMS nailed-cfg = 1, this field is populated with FF.
14	Calling number	12	Calling number	BCD (left-aligned, filler “F”)	Tag 4010
15	Calling number type	2	Calling number type	BCD	Derived from tag 3003 (Calling Number Nature of Address) as follows: • NoA = international: “0” • NoA = national: “1” • NoA = subscriber: “2” • NoA = other values: “1”

Table 8-3 P01 Call Detail Record (continued)

Field	Field Name	Length (Bytes)	Description	Format/Contents	MGC Mapping
16	Bearer capability	2	Bearer capability	BCD	Derived from tag 3001 (User Service Information) as follows: • Speech 3.1 kHz: “0” • Audio 3.1 kHz: “1” • Digital 64 kbit/s: “2” • Audio 7 kHz: “6”
17	Call result	2	Call result	BCD	Derived from the last seven bits of tag 3008 as follows: • Call successful with answer: “1” • Call successful without answer (no reply): “4” • Call successful, user busy: “5”
18	Teleservice	2	Teleservice	BCD	This field hard coded to the value “4”
19	Call connection type	2	Call connection type	BCD	Logic based on Field 16
20	DST indicator	1	DST indicator	BCD	BAMS augmented
21	Partial record	1	Partial record	BCD (= “0”)	Default 0
22	LCE-id	3	LCE-id	BCD (filled with HEX “FF”)	Hex FF
23	Call identity number	3	Call identity number	BCD (filled with HEX “FF”)	Hex FF
24	Restart/reload indicator	1	Restart/reload indicator (future use)	BCD (filled with HEX “FF”)	Hex FF
25	Spare	5	Spare	Filled with HEX “FF”	Hex FF

The field name, length, description, format/contents, and MGC mapping for the P01 tail record are described in Table 8-4.

Table 8-4 P01 Tail Record

Field	Field Name	Length (Bytes)	Description	Format/Contents	MGC Mapping
1	Length	2	Record length	BCD (= 18)	18
2	Record identifier	1	Record identifier	BCD (= 90)	90
3	CDR number	5	Number of CDR contained in logical block	BCD	Counted and inserted in BAMS
4	Spare	10	Spare	Filled with HEX “FF”	FF

Configuring BAMS for NICS Output

Overview

This chapter describes the NICS billing record output by the Cisco Billing and Measurements Server (BAMS). You enable NICS output with the NODEPARMS tag ID. For more information, see the "NODEPARMS Tag ID" section on page 5-10.

Once you have configured your system for NICS output you also need to populate the prefix field in the Trunk Group table (see the "TRUNKGRP—Trunk Group Table" section on page 5-30), and the fields in the Trunk Group Prefix table (see the "TKGPREFIX—Trunk Group Prefix Table" section on page 5-32).

Note If the length of the data in the CDR file obtained from the Cisco PGW 2200 is greater than that which is expected by BAMS, the data value will be truncated.

NICS Output

NICS output files are created by the NIC task. These files are stored in the NICS subdirectory of the /data/*nodename* directory (for example, /data/s0x/NICS). The fields generated in the NICS output are explained in Table 9-1. The NICS file-naming conventions are explained in "File-Naming Conventions" section on page A-4.

NICS File Sequence Numbering

The following rules apply to the generation of NICS file-sequence numbering:

- BAMS generates a NICS file sequence number (FSN) for each NICS output file.
- The NICS FSN increments by one each time a new NICS output file is produced.
- The NICS FSN is not affected by PGW failover, and is independent of the PGW file sequence number.
- The NICS FSN is in the range 0001 through 9999. When BAMS first starts up without a' previous FSN, the first FSN is 0001. When the FSN reaches 9999 the next FSN is 0001.
- BAMS stores the NICS FSN on disk so that the FSN is not affected by system shutdown.

NICS Billing Output File Names

NICS files are named in the following format:

CDR.<switch name>.<NICS FSN>.<date time>

- The switch name is the same as the switch ID (field 3).
- The NICS FSN is generated by BAMS as described in "NICS File Sequence Numbering" section on page 9-1.
- The date time is the date and time when the file is created. BAMS takes the file creation time, CDE 6001 from the 1090 CDB and converts it to the format YYYYMMDDhhmmss in UTC.

Synchronizing NICS FSNs between Redundant BAMS Nodes

Note This section applies only to redundant BAMS nodes.

BAMS synchronizes the NICS file sequence number every day at midnight, using the same logic as that of NICS record sequence-number synchronizing.

In some cases, the active and the standby BAMS nodes may not generate the same FSN for the same CDR file. This can occur if one BAMS node is running before the other BAMS node is ready, if the redundancy is set up after BAMS starts processing CDRs, or if one BAMS node misses some CDR files.

To synchronize the FSNs in a redundant configuration, BAMS synchronizes the FSN on a daily basis. BAMS exchanges the FSN before processing the first NICS output file for the day. The date-time from the NICSOUTPUT filename is used to determine whether a file is the first file of the day. All Date-Time values are in UTC.

Delay in Processing

BAMS waits no longer than one hour for the FSN from the remote node before proceeding without synchronization. If the timer expires, BAMS raises a minor alarm.

Larger FSN Prevalence

In a redundant configuration, both BAMS nodes use the greater FSN from the two BAMS nodes as the new FSN for the first file of the day.

BAMS sends the date-time being synchronized and the sequence number for each synchronization.

For example, "10-3-2002, 00:01:30, 1234" informs the other node that the next file being processed has time stamp 10/03/2002 00:01:30 (from the 1090 CDB time stamp), and the next FSN is 1234.

BAMS raises a minor alarm if a gap is detected between the current FSN and the previous FSN after synchronization.

NICS Data Format

Table 9-1 defines the field name, format, length, PGW 2200 CDB, CDE tag numbers and derivation mapping, and padding and alignment for NICS output.

Table 9-1 NICS Output Field Ordering

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
1	Record Sequence Number (RSN)	(1-25 digits), number in decimal format	No direct mapping. BAMS generates the RSN on its own as a 64-bit integer, starting from 1. First record, RSN = 1. Second record, RSN=2, … The Nth record, RSN = N, … The RSN is not reset when BAMS restarts or when BAMS generates a new output file.	None
2	Call Event ID	char (26)	Char (1-6), softswitch group ID, first 6 characters of tag 6000 from 1090 CDB. Char (7-14), date in YYYYMMDD format, using value from tag 4100/4101 in CDB 1010 or 1030, uses 4001 if neither is available. Char (15-23), time in HHMMSSmmm format. Char (24-26), the sequence number, decimal format left padded by 0. Uses the last 3 digits of the following number: (tag4002) & 0xffffffff	
3	NICS Identifier (a.k.a. Switch id)	char (10)	CDE tag 6000 from CDB 1090. Tag 6000 may contain up to 32 characters, only up to the first 10 characters are used	None
4	Call Direction	number 1	No direct mapping available, not required, always populated with the digit “0”	None
5	Link ID	number 1	All 1030 CDBs: 0 1040 CDBs without any previous 1060s 0 the first 1060, 1 the second and later 1060, 2 1040 with any previous 1060s, 3	None
6	Calling Party Cat	char(3)	Same as tag 3000 or tag 2000 in 1010 or 1030 CDB, whichever is present. In decimal format, such as 000, 013, etc. Defaults to “000” if no tag 2000 or 3000 is found.	Right aligned, zero padded on left.
7	Origin Line Info		Same as tag 2002 or 3002 in 1010 or 1030 CDB. In decimal format, such as 000, 013, etc., defaults to “000” if no 3002 or 2002 is found.	Right aligned, zero padded on left.
8	Jurisdiction	char(6)	Populated with NUL.	None

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
9	Called NAI	number, 3 digits, decimal format	Tag 3007 or 2007 from 1010 or 1030 CDB.	Right aligned, pad by 0
10	Calling NAI	number, 3 digits, decimal format	Tag 3003 or 2003 from 1010 or 1030 CDB.	Right aligned, pad by 0
11	Charge NAI	number, 3 digits, decimal format	Tag 2004 from 1010 or 1030 CDB, defaults to 000 if tag is not available.	Right aligned, pad by 0
12	Caller Presentation	char (2)	Always 00.	
13	Media Type	char (2)	Always 00.	
14	Connection Date	char (8)	Tag 4100 from 1010 or 1030 CDB. Format: YYYYMMDD	
15	Connection Time	char (9)	Tag 4100 from 1010 or 1030 CDB. Format: HHMMSSmmm	
16	Answer Indicator	number (1)	If both tags 4104 and 4105 are present in 1030 or 1010 CDB, the digit is 1. Otherwise the digit is 0.	None
17	Call Termination Code	number (3), decimal	Same as in tag 2008 or 3008 CDE. For example, “016”, “031”, defaults to NUL for long-duration records.	Right aligned, padded by “0”.
18	Originating Country Code	number (5)	Always NUL, not required.	None
19	Originating number	char (40)	CDE 4010 from 1010 or 1030 CDB, with special rules.	None
20	Dialed Country Code	number (5)	Always NUL, not required.	None
21	Dialed Number	char (40)	CDE 4012 from 1010 or 1030 CDB, with special rules.	None
22	Terminating Country Code	number (5)	Always NUL, not required.	None
23	Terminating Number	char (40)	CDE 4014 from 1010 or 1030 CDB.	None
24	Charge Number	char (40)	CDE 4011.	None

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
25	Elapsed Time	char (10)	The earlier of 4106 or 4107 from CDB 1040/1030 minus the later of 4104 or 4105 from 1010/1030 CDB or the time stamp (4001) of the last 1060 CDB. For long durations, this field uses the time stamp of the current 1060 (4001) minus the later of 4104/4105 from the 1010 CDB and the 4001 from the previous 1060 CDB (if any). Format: HHHMMSSmmm	Right aligned for each subfield, 0 padded.
26	Carrier ID Code	char (4)	CDE 2014 from 1010/1030, in decimal format, default NUL.	None
27	Ingress carrier connection date	char (8)	4103 from 1010/1030 CDB. Format: YYYYMMDD	None
28	Ingress carrier connection time	char (9)	4103 from 1010/1030 CDB. Format: HHMMSSmmm	None
29	Ingress carrier disconnection date	char (8)	If (CDE 4028 != 1), it uses the earlier of CDE 4107 and 4106; otherwise, it uses the later of the two. Format: YYYYMMDD NUL for long-duration records.	None
30	Ingress carrier disconnection time	char (9)	If (CDE 4028 != 1), it uses the earlier of CDE 4107 and 4106; otherwise, it uses the later of the two. Format: HHMMSSmmm NUL for long-duration records.	None
31	Ingress Remote Point Code	char (9)	CDE 4034 from 1010 or 1030.	
32	Ingress CIC ID	number (4)	CDE 4068 from 1010 or 1030, default 0000	Right aligned, zero padded
33	Ingress – Access Device ID	number (10)	Always 0000000000	Right aligned, zero padded
34	Ingress Module	number (4)	Always 0000	Right aligned, zero padded
35	Ingress Line	number (4)	Always 0000	Right aligned, zero padded
36	Ingress Channel	number (4)	Always 0000	Right aligned, zero padded

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
37	Ingress trunk group name	char (8)	CDE 4008 from 1010 or 1030 and the prefix value from the Trunk Group table. For example, if the trunk group is 1001 and the prefix is “SIP”, this field is “SIP1001”	None
38	Ingress trunk name	number (8)	CDE 4009 from 1010 or 1030	None
39	Ingress Protocol	char (10)	CDE 4069 from 1010/1030. Values: 0 = ISDN_PRI 1 = SS7 2 = DPNSS 3 = CAS 4 = ASN 5 = Unknown 6 = EISUP 7 = H323 8 = SIP 9 = MGCP Default = Unknown	None
40	Ingress Coding	char (16)	Use value from CDE 4207. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	None
41	Ingress Audio Cap	char (16)	Always NUL.	
42	Ingress - IP Address	char (15)	Use value from CDE 4205. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	Three chars for each dot-separated field of the IP address. Right aligned, zero padded.
43	Ingress RTP Port	number (6)	Use value from CDE 4209. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	Right aligned, zero padded
44	Ingress Pkts sent	number (9)	Always 000000000	Right aligned, zero padded

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
45	Ingress Pkts recv	number (9)	Always 000000000	Right aligned, zero padded
46	Ingress Pkts dropped	number (9)	Always 000000000	Right aligned, zero padded
47	End Date	char (8)	Derived from the earlier of CDE 4106 or 4107 in 1010 or 1030 CDB. For long distance, this field uses the time stamp of the current CDB (CDE 4001). Format: YYYYMMDD	
48	End Time	char (9)	Derived from the earlier of CDE 4106 or 4107 in 1010 or 1030 CDB. For long distance, this field uses the time stamp of the current CDB (CDE 4001). Format: HHMMSSmmm	
49	Answer Date	char (8)	For the first record (Link_ID = 0 or 1), the value is derived from 4104 in 1010 or 1030; if 4104 is not present, it uses 4100 in 1010 or 1030. For the second and later records of the same call, it uses the time stamp (CDE 4001) of the previous 1060 CDB. Format: YYYYMMDD	
50	Answer Time	char (9)	For the first record (Link_ID = 0 or 1), the value is derived from 4104 in 1010 or 1030; if 4104 is not present, it uses 4100 in 1010 or 1030. For the second and later records of the same call, it uses the time stamp (CDE 4001) of the previous 1060 CDB. Format: HHMMSSmmm	
51	Carrier Select Info	number (2)	CDE 2015 from 1010 or 1030, default 00	
52	Routing select	number (2)	Route index, first two octets of CDE 4045 from 1010, default 00	Right aligned, zero padded
53	Egress Remote PC	char (9)	CDE 4037 from 1010 or 1030.	
54	Egress CIC	number (4)	CDE 4072 from 1010 or 1030, default 0000	Right aligned, zero padded
55	Egress—Access Device ID	char (10)	Always NUL.	None

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
56	Egress Module	char (4)	Always NUL.	
57	Egress Line	char (4)	Always NUL.	
58	Egress Channel	char (4)	Always NUL.	
59	Egress trunk group name	char (8)	CDE 4015 from 1010 or 1030, and the prefix value from the Trunk Group table. For example, if the trunk group is 2001 and the prefix is “P”, this field is “P2001”.	None
60	Egress trunk name	number (8)	CDE 4016 from 1010 or 1030.	
61	Egress Protocol	char (10)	CDE 4073 from 1010 or 1030. Values: 0 = ISDN_PRI 1 = SS7 2 = DPNSS 3 = CAS 4 = ASN 5 = Unknown 6 = EISUP 7 = H323 8 = SIP 9 = MGCP Default = Unknown	
62	Egress Coding	char (16)	Use value from CDE 4208. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	
63	Egress Audio Cap	char (16)	Always NUL.	
64	Egress—IP Address	char (15)	Use value from CDE 4206. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	Three characters for each dot-separated field of the IP address. Right aligned, zero padded.
65	Egress RTP Port	number (6)	Use value from CDE 4210. This CDE is available from 1010, 1030 or 1040. The latest value is always used.	Right aligned, zero padded

Table 9-1 NICS Output Field Ordering (continued)

Field	Field Name	Format	PGW 2200 CDB, CDE Tag Numbers and Derivation	Padding and Alignment
66	Egress Pkts sent	number (9)	Always 000000000	Right aligned, zero padded
67	Egress Pkts recv	number (9)	Always 000000000	Right aligned, zero padded
68	Egress Pkts dropped	number (9)	Always 000000000	Right aligned, zero padded
69	Egress carrier connection date	char (8)	4103 from 1010 or 1030. Format: YYYYMMDD	None
70	Egress carrier connection time	char (9)	4103 from 1010 or 1030. Format: HHMMSSmmm	None
71	Egress carrier disconnection date	char (8)	If (CDE 4028 == 1), it uses the earlier of CDE 4106 and 4107; otherwise, it uses the later of the two. Format: YYYYMMDD	None
72	Egress carrier disconnection time	char (9)	If (CDE 4028 == 1), it uses the earlier of CDE 4106 and 4107; otherwise, it uses the later of the two. Format: HHMMSSmmm	None
73	Services		Not used, always NUL.	None

CHAPTER 10

Configuring BAMS for 1110 Binary Output

Overview

This chapter describes the 1110 Binary billing record output by the Cisco Billing and Measurements Server (BAMS). You enable 1110 Binary output with the NODEPARMS tag ID. For more information, see the "Updating the Node Parameters Table" section on page 5-10.

Once you have configured your system for 1110 Binary output you may also need to adjust the default output values in the BIN1110 table (see the "Updating the BIN1110 Table" section on page 5-4). By default, a system configured to generate 1110 Binary output produces 1060 CDBs, and the output files have a BIN suffix.

Note If the length of the data in the CDR file obtained from the Cisco PGW 2200 is greater than that which is expected by BAMS, the data value will be truncated.

Note Default 1110 Binary output also includes 1090 (file header), 1100 (file footer), and 1110 (maintenance) CDB types.

1110 Binary Output

1110 Binary output files are created by the BIN task. These files are stored in the BIN1110 subdirectory of the /data/*nodename* directory (for example, /data/s0x/BIN1110). The fields contained in the BIN1110 output are explained in Table 10-1, Table 10-2, and Table 10-3. The 1110 Binary file-naming conventions are explained in the "File-Naming Conventions" section on page A-4.

File Header (CDB 1090) Data Format

Table 10-1 defines the file header's tag, description, and information source.

Table 10-1 File Header (CDB 1090) Field Ordering

Tag	Description	Information Source
4000	CDB version	1
4001	CDB timepoint	Time written

Table 10-1 File Header (CDB 1090) Field Ordering (continued)

Tag	Description	Information Source
4002	Universal Call reference ID	Always 0, to be compatible with all other CDB layouts
6001	File start time	—
6000	Host Cisco MGC ID	—
6004	Cisco MGC software version	—

File Footer (CDB 1100) Data Format

Table 10-2 defines the file footer's tag, description, and information source.

Table 10-2 File Footer (CDB 1100) Field Ordering

Tag	Description	Information Source
4000	CDB version	1
4001	CDB timepoint	—
4002	Universal reference ID	Always 0, to be compatible with all other CDB layouts
6002	File end time	—
6003	Total number of CDB records	BAMS modifies this tag to match the number of records in the BAMS output. This tag counts both the header CDB and the footer CDB.
6000	Host Cisco MGC ID	—
6004	Cisco MGC software version	—

1110 and 1060 CDB Data Format

Table 10-3 defines the possible CDE fields; it does not refer to the actual offset of the CDE in the CDB because some CDE tags might not be present in the output.

Table 10-3 1110 and 1060 Field Index

	ANSI Tags		ITU Tags			
Index	Tag	Description	Tag	Description	Source CDB	In 1060?
1	5000	Unique Call Id (Mistral and Later)	5000	Unique Call Id (Mistral and later)	1030, 1040 for 1110 output. 1060 for 1060 output.	Y
2	4000	CDB Version	4000	CDB Version	1030, 1040 for 1110 output. 1060 for 1060 output.	Y
3	4001	CDB Timepoint	4001	CDB Timepoint	1030, 1040 for 1110 output. 1060 for 1060 output.	Y
4	4002	Call Reference ID	4002	Call Reference ID	1030, 1040 for 1110 output. 1060 for 1060 output.	Y

Table 10-3 1110 and 1060 Field Index (continued)

	ANSI Tags		ITU Tags			
5	4100	IAM/ Setup timepoint received - milliseconds	4100	IAM/ Setup timepoint received - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
6	4101	IAM/ Setup timepoint sent - milliseconds	4101	IAM/ Setup timepoint sent - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
7	4102	ACM/alert timepoint received - milliseconds	4102	ACM/alert timepoint received - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
8	4103	ACM/alert timepoint sent - milliseconds	4103	ACM/alert timepoint sent - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
9	4104	ANM/Ans. timepoint - received - milliseconds	4104	ANM/Ans. timepoint - received - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
10	4105	ANM/Ans. timepoint - sent - milliseconds	4105	ANM/Ans. timepoint - sent - milliseconds	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
11	4106	First REL/Release timepoint - milliseconds	4106	First REL/Release timepoint - milliseconds	1030 or 1040	N
12	4107	2nd REL/Release timepoint - milliseconds	4107	2nd REL/Release timepoint - milliseconds	1030 or 1040	N
13	4108	RLC/Rel complete timepoint sent - milliseconds	4108	RLC/Rel complete timepoint sent - milliseconds	1030 or 1040	N
14	4109	RLC/Rel complete timepoint rcvd - milliseconds	4109	RLC/Rel complete timepoint received - milliseconds	1030 or 1040	N
15	4008	Originating Trunk Group	4008	Originating Trunk Group	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
16	4009	Originating member	4009	Originating member	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
17	4010	Calling number	4010	Calling number	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
18	4011	Charged number	4011	Charged number	From 1010 or 1030, and from 1060 only if no 1010 found.	Y

Table 10-3 1110 and 1060 Field Index (continued)

	ANSI Tags		ITU Tags			
19	4012	Dialed number	4012	Dialed number	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
20	4014	Called number	4014	Called number	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
21	4015	Terminating Trunk Group	4015	Terminating Trunk Group	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
22	4016	Terminating member	4016	Terminating member	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
23	4028	First release source	4028	First release source	1030 or 1040	N
24	4052	Originating Gateway Primary Select	4052	Originating Gateway Primary Select	1010	N
25	4053	Terminating Gateway Primary Select	4053	Terminating Gateway Primary Select	1010	N
26	4060	Redirecting Number	4060	Redirecting Number	From 1010 or 1030, and from 1060 only if no 1010 found.	Y
27	4061	Tariff Rate	4061	Tariff Rate	1030 or 1040	N
28	4062	Scale Factor	4062	Scale Factor	1030 or 1040	N
29	4063	Test Line Indicator	4063	Test Line Indicator	1010 or 1030	N
30	4201	Ingress SIP URL (Mistral and Later)	4201	Ingress SIP URL (Mistral and Later)	1010 or 1030	N
31	4202	Egress SIP URL (Mistral and Later)	4202	Egress SIP URL (Mistral and Later)	1010 or 1030	N
32	4203	SIP Callid	4203	SIP Callid	1010 or 1030	N
33	4204	Source IP Address	4204	Source IP Address	1010 or 1030	N
34	4205	Ingress Media Device Address	4205	Ingress Media Device Address	1010 or 1030	N
35	4206	Egress Media Device Address	4206	Ingress Media Device Address	1010 or 1030	N
36	4207	Initial Codec	4207	Initial Codec	1010 or 1030	N
37	4208	Final Codec	4208	Final Codec	1010, 1040 or 1030	N

Table 10-3 1110 and 1060 Field Index (continued)

	ANSI Tags		ITU Tags			
38	4209	Ingress Media Device Port Number	4209	Ingress Media Device Port Number	1010 or 1030	N
39	4210	Egress Media Device Port Number	4210	Egress Media Device Port Number	1010 or 1030	N
40	2000	CPC	3000	CPC	1010 or 1030	N
41	2001	User Service Information	3001	User Service Information	1010 or 1030	N
42	2002	Originating Line Information			1010 or 1030	N
43	2003	Calling number NOA	3003	Calling number NOA	1010 or 1030	N
44	2004	Charged number NOA		Does not exist for ITU	1010 or 1030	N
45	2005	Dialed number NOA	3005	Dialed number NOA	1010 or 1030	N
46	2007	Called Number NOA	3007	Called Number NOA	1010 or 1030	N
47	2008	Reason code	3008	Reason code	1030 or 1040	N
48	2013	Transit Network Selection	3013	Transit Network Selection	1010 or 1030	N
49	2015	Carrier Selection Parm			1010 or 1030	N
50	2017	Redirecting Number NOA)	3017	Redirecting Number NOA	1010 or 1030	N
51	4078	Charge Band Number	4078	Charge Band Number	1030 or 1040	N
52	4079	Furnish Charging Number	4079	Furnish Charging Number	1030 or 1040	N
53	4080	Original Called Number	4080	Original Called Number	1010 or 1030	N
54	4081	T.38 Fax Call	4081	T.38 Fax Call	1030 or 1040	N
55	4082	Charge Unit Number	4082	Charge Unit Number	1030 or 1040	N
56	4034	Ingress Originating Point Code	4034	Ingress Originating Point Code	1010 or 1030	N
57	4035	Ingress Destination Point Code	4035	Ingress Destination Point Code	1010 or 1030	N

Table 10-3 1110 and 1060 Field Index (continued)

	ANSI Tags		ITU Tags			
58	4036	Egress Originating Point Code	4036	Egress Originating Point Code	1010 or 1030	N
59	4037	Egress Destination Point Code	4037	Egress Destination Point Code	1010 or 1030	N
60	4046	Ingress Packet Info	4046	Ingress Packet Info	1040 or 1030	N
61	4047	Egress Packet Info	4047	Egress Packet Info	1040 or 1030	N
62	4068	Ingress BearChanId	4068	Ingress BearChanId	1010 or 1030	Y
63	4072	Egress BearChanId	4072	Egress BearChanId	1010 or 1030	Y
64	4083	Charge Indicator	4083	Charge Indicator	1010 or 1030	N
65	4084	Outgoing Calling Party Number	4084	Outgoing Calling Party Number	1010 or 1030	N
66	4085	MCID Request Indicator	4085	MCID Request Indicator	1010 or 1030	N
67	4086	MCID Response Indicator	4086	MCID Response Indicator	1010 or 1030	N
68	4087	Ingress MGCP DLCX (Delete Connection) return code value	4087	Ingress MGCP DLCX (Delete Connection) return code value	1040 or 1030	N
69	4088	Egress MGCP DLCX (Delete Connection) return code value	4088	Egress MGCP DLCX (Delete Connection) return code value	1040 or 1030	N
70	4089	Network Translated Address Indicator	4089	Network Translated Address Indicator	1010 or 1030	N
71	4090	Reservation Request Accepted	4090	Reservation Request Accepted	1010, 1040, or 1030	N
72	4091	Reservation Request Error Count	4091	Reservation Request Error Count	1040 or 1030	N
73	4092	ATM Ingress Configured Profile	4092	ATM Ingress Configured Profile	1010 or 1030	N
74	4093	ATM Egress Configured Profile	4093	ATM Egress Configured Profile	1010 or 1030	N
75	4094	ATM Negotiated Profile	4094	ATM Negotiated Profile	1010 or 1030	N
76	4095	Route List Name	4095	Route List Name	1010 or 1030	N

Table 10-3 1110 and 1060 Field Index (continued)

	ANSI Tags		ITU Tags			
77	4096	Route Name	4096	Route Name	1010 or 1030	N
78	4097	MGCP Script Response String	4097	MGCP Script Response String	1040	N
79	4098	Originating Leg DSP Statistics	4098	Originating Leg DSP Statistics	1030 or 1040	N
80	4099	Terminating Leg DSP Statistics	4099	Terminating Leg DSP Statistics	1030 or 1040	N
81	4211	Originating VPN ID	4211	Originating VPN ID	1010 or 1030	N
82	4212	Terminating VPN ID	4212	Terminating VPN ID	1010 or 1030	N
83	4213	Meter Pulses Received	4213	Meter Pulses Received	1040	N
84	4214	Meter Pulses Sent	4214	Meter Pulses Sent	1040	N
85	4215	Charge Tariff Info	4215	Charge Tariff Info	1040	N
86	4216	Advice of Charge Indicator	4216	Advice of Charge Indicator	1040	N
87	4217	Short Call Indicator	4217	Short Call Indicator	1040	N
88	4218	Charge Limit Exceeded	4218	Charge Limit Exceeded	1040	N
89	4219	CALL Recovered Indication	4219	CALL Recovered Indication	1040	N
90	4220	Partial CLI	4220	Partial CLI	1010 or 1030	Y
91	4221	Service Activation	4221	Service Activation	1030 or 1040	N
92	4222	PRI AOC Invoke Type	4222	PRI AOC Invoke Type	1030 or 1040	N
93	4223	PRI AOC - S Charge Information	4223	PRI AOC—S Charge Information	1030 or 1040	N
94	4224	PRI AOC - D Charge Information	4224	PRI AOC—D Charge Information	1030 or 1040	N
95	4225	PRI AOC - E Charge Information	4225	PRI AOC—E Charge Information	1030 or 1040	N
96	4226	PRI AOC Invoke Failure	4226	PRI AOC Invoke Failure	1030 or 1040	N

CHAPTER 11

Obtaining Measurements

Introduction

The Statistics module on the Cisco Billing and Measurements Server (BAMS) computes, augments, generates, and maintains performance indicators. Performance indicators amount to a history of traffic statistics on a telephone or data network. Counters are calculated for various events (for example, number of call attempts, call duration) for a particular time period. Each counter is associated with a time stamp and a key formed by the concatenation of several fields copied out of the Call Detail Record (CDR) being processed.

Counters that correspond to the same key within the same time period are added together, producing an accumulated count. For this reason, performance indicators are also known as accumulators. That is, "accumulators" and "counters" are used interchangeably to refer to performance indicators.

BAMS maintains counters for three different interval categories (real time, hourly, and daily intervals).

BAMS also maintains a flat file for each collection interval. In order for information to be timely, as soon as an interval boundary is reached, the buckets for that interval are written to disk. As the measurements are written, each measurement is checked against a user-defined threshold value and test condition.

Types of Measurements

Each measurement value represents an accumulation of activity that took place during the measurement interval. At any point in time, three intervals are being collected in parallel, in real-time, hourly, and daily. Measurement values are organized into measurement groups. There are two types of measurement groups: noncarrier and carrier-based. For each noncarrier group, 45 different measurements are accumulated. For each carrier-based group, eight different measurements are accumulated.

Types of Measurement Intervals

The Accumulation (ACC) task generates measurements for one variable, real-time interval, or period and two fixed-time intervals. At any moment in time, two collection windows are open for updating, the current window called "N," and the most recent window called "N–1." Each N and N–1 collection window consists of real-time, hourly, and daily counters. The two open windows are necessary because the Cisco Media Gateway Controller (MGC) does not produce a CDR at the first Initial Address Message (IAM) or seizure. Instead, it produces the CDR at the time of answer or abandonment of the call.

Because of the particular time points that are recorded by the Cisco MGC, an event might not be reported until after the collection interval has been closed, even though the event should have been credited to that interval. The one exception to the two-window rule is at startup, where only the current window is open. That remains the case until after the first interval boundary is crossed.

Real-Time Intervals

You can configure the real-time interval to any of the following durations: 5 minutes, 10 minutes, 15 minutes, 20 minutes, or 30 minutes. The default real-time interval is 15 minutes. All real-time measurements are stored in files whose names have the prefix acc_r.

Hourly Intervals

The hourly interval contains the sum of all of the real-time intervals that took place during the hour. For this reason, 60 minutes must be evenly divisible by the real-time interval length. All hourly measurements are stored in files whose names have the prefix acc_h.

Daily Intervals

The daily interval contains the sum of all of the hourly intervals that took place during the day. All daily measurements are stored in files whose names have the prefix acc_d.

Noncarrier Measurements

Noncarrier measurements are organized by trunk group. Table 11-1 lists these measurements and their mnemonics. It also describes each measurement's trigger time point and tag, derivation, and mapping.

Note For a list of which measurements are suppressed or not pegged based on the PGW_DYNAMIC_UPDATE value, see "Setting the PGW Dynamic Update Mode" section on page 2-16.

Table 11-1 Noncarrier Measurements

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Call Attempts Incoming	1010 or 1030 received, Tag 4100 or 4101.	BAM:IGR CALL ATT	Pegged when a 1010 CDB is recorded with 4008 or when 1030 recorded with 4008
Call Attempts Outgoing	1010 or 1030 received, Tag 4100 or 4101.	BAM:EGR CALL ATT	Pegged when a 1010 CDB is recorded with 4015 or when 1030 recorded with 4015. Suppressed in MGCP Dial or MGCP Scripting calls.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Outgoing Attempts Blocked	1030 or 1040 received, Tag 4100 or 4101.	BAM:EGR CALL BLKD	4015 populated, 1030 or 1040 with (Cause Code) Tag {2008 or 3008}== {21, 25, 27, 29, 34, 38, 41, 42, 44, 46, 47, 53, 63}. Suppressed in MGCP Dial or MGCP Scripting calls.
Failed Calls-Congestion	1030 or 1040 received, Tag 4100 or 4101.	BAM:TTL FAILED CONGEST	Peg for all 1030 or 1040 where {2008 or 3008} == {42, 44, 47}
Successful Calls Incoming	1040 or 1030 received, later of Tag 4106 or 4107.	BAM:IGR TERM NORM	Peg for all 1040 CDB or 1030 CDB where 4008 is populated and {2008 or 3008} == {16, 17, 18, 19, 31}
Successful Calls Outgoing	1040 or 1030 received, later of Tag 4106 or 4107.	BAM:EGR TERM NORM	Pegged when 1040 or 1030 CDB recorded with 4015 populated and {2008 or 3008} == {16, 17, 18, 19, 31}. Suppressed in MGCP Dial or MGCP Scripting calls.
Percent Trunk Group Usage Incoming	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received.	BAM:IGR PCT TRK USE	Measured as a percentage of time that circuits are occupied based on the total number of circuits belonging to a trunk group over the provisioned interval of measurement. Any circuit on Tag 4008 triggers this measurement from CDB Tag 1010. The starting time point is the earlier of 4100 or 4101. The end timepoint is in the 1040 CDB and is the later of tag 4108 or 4109. When the PGW_DYNAMIC_UPDATE flag is set to true, this measurement is suppressed before a 1071 CDB is received on a trunk group. Also, it is suppressed for dynamically added trunk groups.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Percent Trunk Group Usage Outgoing	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received.	BAM:EGR PCT TRK USE	Measured as a percentage of time that circuits are occupied based on the total number of circuits belonging to a trunk group over the provisioned interval of measurement. Any circuit on Tag 4015 triggers this measurement from CDB Tag 1010. The starting time point is the earlier of 4100 or 4101. The end timepoint is in the 1040 CDB and is the later of tag 4108 or 4109. When the PGW_DYNAMIC_UPDATE flag is set to true, this measurement is suppressed before a 1071 CDB is received on trunk group. Also, it is suppressed in MGCP Dial or MGCP Scripting mode. This measurements is always suppressed for dynamically added trunk groups.

Note When the measurement Percent Trunk Group Usage (PCT TRK USE) is specified, it is possible for the measurement to be recorded in the real-time acc_r file but not recorded in the hourly acc_h or daily acc_d files. For example, trunk group usage that is as low as 1% for a real-time duration that is set for 10, 15, or 30 minutes, will be recorded in the acc_r file. However, such low usage will fall below 1% for hourly and daily time periods and, therefore, will not be recorded in the acc_h or acc_d file. Similarly, a measurement can meet the minimum usage percentage to be recorded in the real-time and hourly files but not the daily file.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Maintenance Duration per Trunk Group	Starts when 1071 received (Tag 4100 or 4101). Closes when interval is closed or another 1071 received.	BAM:TTL MAINT USE	Only available with PGW release 9.4(1) or later and if the PGW_DYNAMIC_UPDATE flag is set to true. Measured as a percentage of time that circuits are occupied based on the total number of circuits belonging to a trunk group over the provisioned interval of measurement. When the 1071 CDB contains the number of circuits unavailable for a trunk group, BAMS is able to track the number of circuits out of service. This measurement is available only if the PGW_DYNAMIC_UPDATE flag is set to true and the trunk group is configured in the Trunk Group table. The measurement is suppressed before a 1071 is received. It is always suppressed for dynamically added trunk groups.
Total Traffic in Erlangs	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received.	BAM:TTL ERLANGS	Measured as Erlangs for both Ingress and Egress for a trunk group. Use total seconds duration, from 1010 CDB, use timepoint in earlier of 4100 or 4101. For the end of the duration, use the later of 4108 or 4109. Erlangs = (total seconds) / (seconds in measured interval). Example: For a one-hour measurement, with 99,000 secs measured, the formula would be (99,000)/(3600 secs) = 27.5 Erlangs. If the same measurement occurred over a 15-minute interval, the formula would be (99,000)/ (900 secs) = 110 Erlangs.
Total Calls Terminated Normally	1040 received (Tag 4106 or 4107)	BAM:TTL TERM NORM	Pegged when 1040 CDB recorded and release code in the set {16, 17, 18, 19, 31}
Calls Terminated Abnormally	1030 or 1040 received (Tag 4106 or 4107)	BAM:TTL TERM ABNORM	Pegged for any 1040 where {2008 or 3008} != {16, 17, 18, 19, 31} or for 1030 CDB with any release code.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Calls Terminated, Failed MGW or NAS	1030 or 1040 received (Tag 4106 or 4107)	BAM:TTL TERM FAILED MGW	Pegged for any 1030 or 1040 CDB where {2008 or 3008} == {29}
Calls Rejected	1030 CDB received (Tag 4100 or 4101)	BAM:TTL CALLS REJECTED	Pegged for any 1030 CDB where {2008 or 3008} == {21}
Calls Rejected, Unknown Dialed Number	1030 CDB received (Tag 4100 or 4101)	BAM:TTL REJECTED DIALNUM	Pegged for any 1030 CDB where {2008 or 3008} == {1, 5, 22, 28}
Calls Rejected, Other Reasons	1030 CDB received (Tag 4100 or 4101)	BAM:TTL REJECTED OTHER	Pegged for any 1030 CDB where {2008 or 3008} != {1,5,16,17,18,19,21,22,28,29}
Overflow, Outgoing Attempts Blocked	1030 CDB received (Tag 4100 or 4101)	BAM:EGR OFL BLKD	Pegged for 1030 CDB where 4015 is populated and {2008 or 3008} == {27, 34, 41, 42, 44, 47, 53, 63}. Suppressed in MGCP Dial or MGCP Scripting calls.
Total Sum of Usage Pegs per Trunk Group	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:TTL TRAFFIC USAGE PEGS	Pegged for any 1010 or 1030 CDB.
Tandem Routing Attempts, Outgoing	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:EGR TANDEM ATT	Pegged when Tag 4015 (trunk group) is marked T (tandem connection) for 1010 or 1030 CDB. Always suppressed for dynamically added trunk groups. Also suppressed in MGCP Dial or MGCP Scripting calls.
Tandem Completions, Outgoing	1010 CDB received (Tag 4100 or 4101)	BAM:EGR TANDEM COMPLT	Pegged when Tag 4015 (trunk group) is marked T (tandem connection) for 1010 CDB. Always suppressed for dynamically added trunk groups. Also suppressed in MGCP Dial or MGCP Scripting calls.
Tandem Attempts, Incoming	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:IGR TANDEM ATT	Pegged when Tag 4008 (trunk group) is marked T (tandem connection) for 1010 or 1030 CDB. Always suppressed for dynamically added trunk groups.
Tandem Completions, Incoming	1010 CDB received (Tag 4100 or 4101)	BAM:IGR TANDEM COMPLT	Pegged when Tag 4008 (trunk group) is marked T (tandem connection) for 1010 CDB. Always suppressed for dynamically added trunk groups.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Tandem Duration, Outgoing	1010 CDB received (Tag 4100 or 4101)	BAM:EGR TANDEM DUR	Duration measured when Tag 4015 (trunk group) is marked T (tandem connection) for 1010 CDB. Always suppressed for dynamically added trunk groups.
Tandem Duration, Incoming	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received (tag 4108 or 4109).	BAM:IGR TANDEM DUR	Duration measured when Tag 4008 (trunk group) is marked T (tandem connection) for 1010 CDB. Start with earlier of timepoint in 4100 or 4101 of 1010 CDB, end with later of 4108 or 4109 in 1040 CDB. Always suppressed for dynamically added trunk groups.
Conversation Duration Ingress	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received (tag 4108 or 4109).	BAM:IGR CONV DURATION	Duration measured from the later of tag 4104 or 4105 in the 1010 CDB, until the earlier of tag 4106 or 4107, when tag 4008 is populated with a valid trunk group number.
Conversation Duration Egress	Starts when 1010 received (Tag 4100 or 4101). Closes when interval is closed or 1040 received (tag 4108 or 4109).	BAM:EGR CONV DURATION	Duration measured from the later of tag 4104 or 4105 in the 1010 CDB, until the earlier of tag 4106 or 4107, when tag 4015 is populated with a valid trunk group number. Suppressed in MGCP Dial or MGCP Scripting calls.
Setup Duration Ingress	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:IGR SETUP DURATION	Duration measured from timepoint in earlier of tag 4100 or 4101 of 1010 CDB end with later of 4102 or 4103 in 1010 CDB. For 1030 CDB, start with earlier of 4100 or 4101, and end with earlier of 4106 or 4107, when tag 4008 is populated with a valid trunk group number.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Setup Duration Egress	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:EGR SETUP DURATION	Duration measured from timepoint in earlier of tag 4100 or 4101 of 1010 CDB end with later of 4102 or 4103 in 1010 CDB. For 1030 CDB, start with earlier of 4100 or 4101, and end with earlier of 4106 or 4107, when tag 4015 is populated with a valid trunk group number. Suppressed in MGCP Dial or MGCP Scripting calls.
Teardown Duration Ingress	1030 or 1040 CDB received (Tag 4106 or 4107)	BAM:IGR TEARDOWN DURATION	Duration measured from timepoint in earlier of 4106 or 4107 end with later of 4108 or 4109, when tag 4008 is populated with a valid trunk group number.
Teardown Duration Egress	1030 or 1040 CDB received (Tag 4106 or 4107)	BAM:EGR TEARDOWN DURATION	Duration measured from timepoint in earlier of 4106 or 4107 end with later of 4108 or 4109, when tag 4015 is populated with valid trunk group number. Suppressed in MGCP Dial or MGCP Scripting calls.
Call Routing I Peg	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CALL ROUTING I	Pegged when ingress and egress traffic terminations are maintained by the same gateway. This measurement is pegged when tag 4038 and tag 4039 are equal and neither tag 4069 nor 4073 equals 6 (EISUP).
Call Routing II Peg	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CALL ROUTING II	Pegged when ingress and egress traffic terminations are maintained by the different gateways, but under control of the same MGC. This measurement is pegged when tag 4038 and tag 4039 are not equal and neither tag 4069 nor 4073 equals 6.
Call Routing III Peg	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CALL ROUTING III	Pegged when one side of a call originates or terminates under the control of a gateway connected to the MGC, but the other side of the call terminates in another network not under the control of the MGC. This measurement is pegged when either tag 4069 or 4073 equals 6.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Successful H.323 Terminating Pegs	1010 CDB received (Tag 4100 or 4101)	BAM:EGR SUCCESSFUL H.323	Pegged when a 1010 CDB is received with a tag 4073 and a value of 7. **Note** The H.323 measurements are output only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Successful H.323 Originating Pegs	1010 CDB received (Tag 4100 or 4101)	BAM:IGR SUCCESSFUL H.323	Pegged when a 1010 CDB is received with a tag 4069 and a value of 7. **Note** The H.323 measurements are output only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Unsuccessful H.323 Terminating Pegs	1030 CDB received (Tag 4100 or 4101)	BAM:EGR UNSUCCESSFUL H.323	Pegged when a 1030 CDB is received with a tag 4073 with a value of 7. **Note** The H.323 measurements are output only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Unsuccessful H.323 Originating Pegs	1030 CDB received (Tag 4100 or 4101)	BAM:IGR UNSUCCESSFUL H.323	Pegged when a 1030 CDB is received with a tag 4069 of value 7. **Note** The H.323 measurements are output only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Successful ISUP Terminating Pegs	1010 CDB received (Tag 4100 or 4101)	BAM:EGR SUCCESSFUL ISUP	Pegged when a 1010 CDB is received with a tag 4073 of value 0.
Successful ISUP Originating Pegs	1010 CDB received (Tag 4100 or 4101)	BAM:IGR SUCCESSFUL ISUP	Pegged when a 1010 CDB is received with a tag 4069 of value 0.
Unsuccessful ISUP Terminating Pegs	1030 CDB received (Tag 4100 or 4101)	BAM:EGR UNSUCCESSFUL ISUP	Pegged when a 1030 CDB is received with a tag 4073 of value 0.
Unsuccessful ISUP Originating Pegs	1030 CDB received (Tag 4100 or 4101)	BAM:IGR UNSUCCESSFUL ISUP	Pegged when a 1030 CDB is received with a tag 4069 of value 0.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
ISDN Terminating Setup Message Delay Pegs	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:EGR ISDN SETUP MSG DELAY	Pegged when a 1010 or 1030 CDB is received with a tag 4073 with a value of 0, when the setup duration > 3000 ms. The setup duration is measured from timepoint in the earlier of tag 4100 or 4101 of a 1010 CDB. Setup ends with the later of 4102 or 4103.
ISDN Originating Setup Message Delay Pegs	1010 or 1030 CDB received (Tag 4100 or 4101)	BAM:IGR ISDN SETUP MSG DELAY	Pegged when a 1010 or 1030 CDB is received with a tag 4069 having a value of 0, when the setup duration > 3000 ms. The setup duration is measured from timepoint in the earlier of tag 4100 or 4101 of a 1010 CDB. Setup ends with the later of 4102 or 4103.

Table 11-1 Noncarrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
Number of Defined CICs during the Measurement Period	Start of measurement interval	BAM:TTL CIC DEFINED	Number of circuits provisioned in the Trunk Group table when the PGW_DYNAMIC_UPDATE flag is set to false. Or the value for the number of circuits in the trunk group received from the latest 1071 CDB when the PGW_DYNAMIC_UPDATE flag is set to true. Suppressed before 1071 CDB is received on trunk group when the PGW_DYNAMIC_UPDATE flag is set to true. Always suppressed for dynamically added trunk groups. **Note** No corresponding threshold crossing alert exists for this measurement.
Average Number of Available CICs during the Measurement Period	1071 received	BAM:TTL AVLBL CIC	Total – (maintDuration ÷ intervalLength) • Total—complete number of circuits • maintDuration—total maintenance duration (see "TTL MAINT USE" in Table 11-1 for details) • intervalLength—total number of seconds for the measurement period. Only available if the PGW_DYNAMIC_UPDATE flag is set to true and trunk group is configured in the Trunk Group table. Suppressed before 1071 CDB is received on trunk group. Always suppressed for dynamically added trunk groups.

Carrier-Based Measurements

Carrier-based measurements are grouped by Trunk Group/Interexchange Carrier (IC). Table 11-2 lists these measurements with their mnemonics. It also describes each measurement's trigger time point and tag, derivation, and mapping.

Table 11-2 Carrier Measurements

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
IC Destined Calls	1010 CDB received (Tag 4100 or 4101)	BAM:IC:XXXX IC EGR CALLS	Pegged per IC for CDB 1010 where Tag 2014 is populated with a valid carrier ID.
IC Destined Calls, No Circuit	1030 CDB received (Tag 4100 or 4101)	BAM: IC:XXXX IC EGR NO CKT	Pegged per IC for 1030 CDB where Tag 2008 = {42, 44, 47}.
IC Usage	1010 CDB received (Tag 4100 or 4101)	BAM:IC:XXXX TTL DURATION	Duration measured per IC when CDB 1010 tag 2014 is populated with a valid carrier ID. Start with the earlier of the timepoints in 4100 or 4101 of a 1010 CDB. Ends with the later of the timepoints in 4108 or 4109 of a 1040 CDB.
Carrier Select No Indication	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CARRIERSELECT NO INDICATION	Pegged when Tag 2015 != {1, 2, 3, 4} and marked "T" for tandem connected in the Trunk Group table. Output by trunk group and carrier. Always suppressed for dynamically added trunk groups.
Carrier Select PreSubscribed Not Input	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CARRIERSELECT PRESUBSCRIBED NIPT	Pegged when Tag 2015 = { 1 } and marked "T" for tandem connected in the Trunk Group table. Output by trunk group and carrier. Always suppressed for dynamically added trunk groups.
CarrierSelect PreSubscribed and Input	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CARRIERSELECT PRESUBSCRIBED INPT	Pegged when Tag 2015 = { 2 } and marked "T" for tandem connected in the Trunk Group table. Output by trunk group and carrier. Always suppressed for dynamically added trunk groups.

Table 11-2 Carrier Measurements (continued)

Measurement	Trigger Time Point and Tag	Mnemonic	Derivation and Mapping
CarrierSelect PreSubscribed with No Indication	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CARRIERSELECT PRESUBSCRIBED WNI	Pegged when Tag 2015 = { 3 } and marked "T" for tandem connected in the Trunk Group table. Output by trunk group and carrier. Always suppressed for dynamically added trunk groups.
Carrier Id Code Not PreSubscribed but Input by Customer	1030 or 1010 CDB received (Tag 4100 or 4101)	BAM:TTL CARRIERSELECT NOTPRESUBSCRIBED	Pegged when Tag 2015 = { 4 } and marked "T" for tandem connected in the Trunk Group table. Output by trunk group and carrier. Always suppressed for dynamically added trunk groups.

Storage of Measurements

Both carrier-based and noncarrier measurements are stored internally in groups. Each group consists of all the measurements that belong to a particular key. These measurements are then put in subgroups according to interval. Each measurement group contains real-time, hourly, and daily measurements. There are two types of keys or measurement groups. These are noncarrier measurements and carrier-based measurements. Regardless of group type, measurements are held in memory for performance reasons. Up to two time periods for each key can reside in memory simultaneously. These are the current time period and the one preceding the current time period. Because memory is somewhat volatile, the counters must be written to disk to prevent loss of data. At the end of each real-time time period, the contents of memory are written to disk. This disk file is then available to be read at the next startup.

Noncarrier Measurement Production

Noncarrier measurements consist of 45 measurements or accumulators for each of the three intervals kept in memory. This results in 135 measurements for the current time period, plus a possible additional 135 measurements for the preceding time period.

Carrier-Based Measurement Production

Carrier measurements consist of eight measurements or accumulators for each of the three intervals kept in memory. This results in 24 measurements for the current time period, plus a possible additional 24 measurements for the preceding time period.

Memory Allocation

Depending on the operating mode, the system either preallocates counters for all of the configured measurement groups or it allocates counters as activity is detected in each measurement group.

Threshold Crossing Alarms

TCA Table

The Threshold Crossing Alarms (TCA) table contains values and conditions for each measurement that you wish to link to an alarm. These values and conditions are organized by trunk group or Trunk Group/IC (measurement group). Enter the measurement groups that are of concern. You need not populate every value and condition for a specified measurement group. A global measurement group can be specified to be used for all measurement groups that are not specifically entered.

Threshold String Values

Table 11-3 lists the condition value strings and the threshold value strings that you use to identify the condition and threshold values you set in an MML provisioning session with the TCA-TBL tag ID. For more information, see the "Updating the Threshold Crossing Alarms Table" section on page 5-23.

Table 11-3 Threshold String Values

Threshold	Condition Value String	Threshold Value String	Entered By
Call Attempts Incoming	igr-call-att-cond	igr-call-att	TAG/TRK
Call Attempts Outgoing	egr-call-att-cond	egr-call-att	TAG/TRK
Outgoing Attempts Blocked	egr-call-blkd-cond	egr-call-blkd	TAG/TRK
Failed Calls Congestion	ttl-failed-cong-cond	ttl-failed-cong	TAG/TRK
Successful Calls Incoming	igr-term-norm-cond	igr-term-norm	TAG/TRK
Successful Calls Outgoing	egr-term-norm-cond	egr-term-norm	TAG/TRK
Percent Trunk Group Usage Incoming	igr-pct-trk-use-cond	igr-pct-trk-use	TAG/TRK
Percent Trunk Group Usage Outgoing	egr-pct-trk-use-cond	egr-pct-trk-use	TAG/TRK
Maintenance Duration per Trunk Group	ttl-maint-use-cond	ttl-maint-use	TAG/TRK
Total Traffic Erlangs	ttl-erlangs-cond	ttl-erlangs	TAG/TRK
Total Calls Terminated Normally	ttl-term-norm-cond	ttl-term-norm	TAG/TRK

Table 11-3 Threshold String Values (continued)

Threshold	Condition Value String	Threshold Value String	Entered By
Calls Terminated Abnormally	ttl-term-abnorm-cond	ttl-term-abnorm	TAG/TRK
Calls Terminated, Failed MGW or NAS	ttl-term-failed-mgw-cond	ttl-term-failed-mgw	TAG/TRK
Calls Rejected	ttl-calls-rejected-cond	ttl-calls-rejected	TAG/TRK
Calls Rejected, Unknown Dialed Number	ttl-rejected-dialnum-cond	ttl-rejected-dialnum	TAG/TRK
Calls Rejected, Other Reasons	ttl-rejected-other-cond	ttl-rejected-other	TAG/TRK
Overflow, Outgoing Attempts Blocked	egr-ofl-blkd-cond	egr-ofl-blkd	TAG/TRK
Total Sum of Usage Pegs per Trunk Group (not including maintenance pegs for Release 2.xx)	ttl-traffic-usage-pegs-cond	ttl-traffic-usage-pegs	TAG/TRK
Tandem Routing Attempts, Outgoing	egr-tndm-att-cond	egr-tndm-att	TAG/TRK
Tandem Completions, Outgoing	egr-tndm-cmplt-cond	egr-tndm-cmplt	TAG/TRK
Tandem Routing Attempts, Incoming	igr-tndm-att-cond	igr-tndm-att	TAG/TRK
Tandem Completions, Incoming	igr-tndm-cmplt-cond	igr-tndm-cmplt	TAG/TRK
Tandem Duration, Outgoing	egr-tndm-dur-cond	egr-tndm-dur	TAG/TRK
Tandem Duration, Incoming	igr-tndm-dur-cond	igr-tndm-dur	TAG/TRK
IC Destined Calls, Outgoing	egr-ic-calls-cond	egr-ic-calls	TAG/TRK/IC
IC Destined Calls, No Circuit	egr-ic-nockt-cond	egr-ic-nockt	TAG/TRK/IC
IC Usage	ttl-ic-usage-cond	ttl-ic-usage	TAG/TRK/IC
Conversation Duration, Incoming	igr-conv-dur-cond	igr-conv-dur	TAG/TRK

Table 11-3 Threshold String Values (continued)

Threshold	Condition Value String	Threshold Value String	Entered By
Conversation Duration, Outgoing	egr-conv-dur-cond	egr-conv-dur	TAG/TRK
Setup Duration, Incoming	igr-setup-dur-cond	igr-setup-dur	TAG/TRK
Setup Duration, Outgoing	egr-setup-dur-cond	egr-setup-dur	TAG/TRK
Tear Down Duration, Incoming	igr-teardown-dur-cond	igr-teardown-dur	TAG/TRK
Tear Down Duration, Outgoing	egr-teardown-dur-cond	igr-teardown-dur	TAG/TRK
Call Routing I Peg Total	ttl-route-1-cond	ttl-route-1	TAG/TRK
Call Routing II Peg Total	ttl-route-2-cond	ttl-route-2	TAG/TRK
Call Routing III Peg Total	ttl-route-3-cond	ttl-route-3	TAG/TRK
Total Carrier Select No Indication	ttl-ic-sel-noind-cond	ttl-ic-sel-noind	TAG/TRK/IC
Total Carrier Select Presubscribed Not Input	ttl-ic-sel-pr-nipt-cond	ttl-ic-sel-pr-nipt	TAG/TRK/IC
Total Carrier Select Presubscribed and Input	ttl-ic-sel-pr-inpt-cond	ttl-ic-sel-pr-inpt	TAG/TRK/IC
Total Carrier Select Presubscribed with No Indication	ttl-ic-sel-pr-wni-cond	ttl-ic-sel-pr-wni	TAG/TRK/IC
Total Carrier Id Code Not Presubscribed but Input by Customer	ttl-ic-sel-no-pr-cond	ttl-ic-sel-no-pr	TAG/TRK/IC

Table 11-3 Threshold String Values (continued)

Threshold	Condition Value String	Threshold Value String	Entered By
Successful H.323 Terminating Peg	egr-scfl-h323-cond	egr-scfl-h323	TAG/TRK **Note** The H.323 measurements are checked only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Successful H.323 Originating Peg	igr-scfl-h323-cond	igr-scfl-h323	TAG/TRK **Note** The H.323 measurements are checked only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Unsuccessful H.323 Terminating Peg	egr-unscfl-h323-cond	egr-unscfl-h323	TAG/TRK **Note** The H.323 measurements are checked only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Unsuccessful H.323 Originating Peg	igr-unscfl-h323-cond	igr-unscfl-h323	TAG/TRK **Note** The H.323 measurements are checked only when the enable-H323 parameter is set to 1 in the Node Parameters table.
Total Successful Terminating ISUP Requests	egr-scfl-isup-cond	egr-scfl-isup	TAG/TRK
Unsuccessful Terminating ISUP Requests	egr-unscfl-isup-cond	egr-unscfl-isup	TAG/TRK
Successful ISUP Originating Requests	igr-scfl-isup-cond	igr-scfl-isup	TAG/TRK

Table 11-3 Threshold String Values (continued)

Threshold	Condition Value String	Threshold Value String	Entered By
Unsuccessful ISUP Originating Requests	igr-unscfl-isup-cond	igr-unscfl-isup	TAG/TRK
Terminating ISUP Setup Message Response Delay	egr-isdn-su-msg-delay-cond	egr-isdn-su-msg-delay	TAG/TRK
Originating ISUP Setup Message Response Delay	igr-isdn-su-msg-delay-cond	igr-isdn-su-msg-delay	TAG/TRK
Total Number of Available CICs	ttl-avlbl-cic-cond	ttl-avlbl-cic	TAG/TRK

Threshold Crossing Conditions

Each threshold crossing condition is a code that checks the difference (if any) between the user-specified value and the current real-time measurement value. The condition is specified as a number between 0 and 4. Any other value is invalid. Table 11-4 defines the meaning of each condition value.

Condition Value Relationship

Table 11-4 lists condition values used for measurements.

Table 11-4 Condition Values

Value	Condition Description
0	Ignore
1	Less Than (<)
2	Equal To (=)
3	Greater Than (>)
4	Not Equal To (!=)

Threshold Values

With the TCA-TBL tag ID, you specify the threshold value and the condition value to so that they generate an alarm if a specific measurement condition is reached. For example, for a given measurement, if the condition is set to 4 and the threshold value is set to 10, an alarm is generated if the measurement value is greater than 10. Threshold values are specified as positive integers.

Trunk Group Identification (Threshold Key)

Each threshold specification (threshold value and condition value) must be associated with a measurement group. If the Entered By tag specifies TAG/TRK, the measurement is organized by the trunk group number. If the Entered By tag specifies TAG/TRK/IC, the measurement is organized by trunk group number and an interexchange-carrier number. A special measurement group can be specified to apply to all TAG/TRK measurement groups that are not otherwise specified. This measurement group is identified by the name “global/0,” where the TAG is “global” and the trunk group is “0.”

Processing Logic

The same logic is used for processing all accumulation periods: computation is based on the time stamps from call detail records generated by the switch. The distinguishing factor among the different accumulation periods is the time period in which two events are considered to occur for the same counter. Counts for any given event are added to the accumulators for the time period that matches the time stamp of the event. More specifically, Table 11-2 identifies the time point for each event that is used to match the accumulator time period.

Three different levels at which statistics can be generated are as follows:

- Using the CDR details
- Using the aggregate CDRs
- Using the correlated CDRs

There are advantages and disadvantages to each of the above approaches. Statistics computed from CDR details result in more frequent updates, and thus a finer granularity of reporting. However, more records must also be processed. Thus, the volume of connections and the length of the switch-reporting interval can dramatically drive up the amount of processing required to make the statistics available. Conversely, computing statistics from aggregated or correlated CDRs provides a more efficient computation, but less timely statistics.

The following section applies to all accumulation types, periods, and levels.

Statistics Subsystem Functions

The Statistics subsystem provides the following functionality:

- Obtains the chain of aggregated CDR details.
- Receives the CDR details in time order from the Augmentation (AUG) task. The CDRs arrive in two types of files: aug_acei and aug_acbc. The aug_acei files contain complete CDRs taken from fmt files. For each fmt file, at least one aug_acei file exists. An aug_acbc file exists for each threshold crossing. The aug_acbc files contain all partial CDRs (CDRs that did not complete during the interval).
- Assigns the usage in real-time, hourly, and daily intervals. For cumulative count fields, a call that began before the start of the interval and has not ended adds the full length of the real-time interval to the count. Any CDR that begins in the interval (but has not ended) adds the time from the start of CDR to the end of the interval. Any CDR that ends in the interval (but did not start in the interval) adds the time from the start of the interval to the end time of the CDR. Any CDR that both begins and ends in the same interval adds the delta between the start and end time of the CDR.
- Calculates hourly counters.
- Monitors check points at the end of every file interval (complete reading of all aug_acei files and the aug_acbc file for the given interval).
- Summarizes the hourly counters and produces daily counters. These tables should be stored in table sets.
- Manages the daily counters so that counters older than a specified retention period are purged regularly.

At any moment in time, two collection intervals are open for updating, the current interval, called “N,” and the most recent interval, called “N–1.” The two open intervals are necessary because the Cisco MGC does not produce a CDR at the first IAM or seizure; rather, it creates the first indication of a call at

answer or abandonment. Because of the particular time points that are recorded by the Cisco MGC, there are some cases where an event is not reported until after the collection interval has been closed, yet the event should be credited to that interval. A bucket or interval shall never be credited for more than the total duration that is available during that interval, regardless of when the indication of the call was received.

A flat file is maintained for each collection interval. In order to provide timely information, buckets for an interval (the current interval, or N) are written as soon as an interval boundary is reached. At the same time, the previous interval (N–1), which may have been updated because of late reports for call abandonment, is rewritten to disk and is not updated again. Very late reports are written to the oldest collection period that is still open, which is always the N–1 interval. The one exception to this rule is at startup, when only the current period is open, until after the first interval boundary is crossed.

Keys and Counters

Keys and counters are stored in memory and written to a checkpoint file on a regular basis.

The key is a unique sequence number used to identify the specific collection of counters. The key fields are the trunk group number and the IC. Table 11-5 lists the key field names and their descriptions.

Table 11-5 Key Field Name Descriptions

Field	Description
Trunk Group	Taken from the sigpath to trunkgroup configuration. • For Ingress, use the Terminating Trunk Group. • For Egress, use the Originating Trunk Group. The values are 1–9999 for BAF, 4096 for ITU, and 1684 for ANSI. The default is 0 if no information is available.
IC	Carrier ID: 2014.

Counter Sets

Each counter set is made up of three groups of counters, one group for real time, one for hourly, and one for daily. The counters in each group represent running tallies of the specified statistics. Each group of counters represents only the current interval of the counter type (current real time interval, the current hour, the current day). Each counter statistic is credited to the time period in which it occurred. Note that there are different time periods. Hourly counters keep track of the statistics on hourly boundaries. If an event spans multiple hours, one counter for each hour spanned is created. For example, if a call is established at 11:50 and is disconnected at 12:15, one counter for the 11:00 hour is created with 10 minutes of conversation time credited to it, and a second one is created for the 12:00 hour with 15 minutes of conversation time.

Similarly, daily counters credit statistics on daily boundaries.

Frequency of Statistics

In addition to the rollup hourly and 24-hour statistics, which are tabulated with any of the previous options, the system also supports 5-, 10-, 15-, 20-, and 30-minute (real-time), hourly, and daily statistics.

Note You can configure the measurements interval by editing the interval-minutes field in the Node Parameters table. For more information, see the "Updating the Node Parameters Table" section on page 5-10.

Statistics Output

After statistics have been collected, they are output to a flat file. For each real-time interval, an acc_rYYYYMMDDHHMM00 file is created. For each hourly interval, an acc_hYYYYMMDDHH0000 file is created. For each daily interval, an acc_dYYYYMMDD000000 file is created. These files are stored in the opt/CiscoBAMS/data/s0x/Measurements directory.

Note All times are in Universal Coordinated Time, which is taken from the CDR record.

The output files are generated as soon as the ACC task has finished processing the aug_acbc file (last file) for the given interval. This means that the ACC task generates a flat file for the real-time interval at the end of each set of files for the real-time interval (5, 10, 15, 20, or 30 minutes). The hourly output file is generated when the last interval file is processed for that hour. The daily output file is generated as soon as the last interval file for the day is processed. Each file is created on a real-time, 1-hour, and 24-hour basis. Each file contains all of the statistics gathered in the previous period.

In the following section, the term "trunk groups" is used to represent both TAG/Trunk Group and TAG/Trunk Group/IC.

Statistics are generated from CDBs produced by the Cisco MGC. Since the output is reported by TAG/Trunk Group or by TAG/Trunk Group/IC, measurements are produced only for trunk groups that have call activity starts (unless the system is running in configured mode and trunk groups are specified). Therefore, when the system is started, the statistics output files are empty until call activity begins. Regardless of call activity, the appropriate acc_x files are generated. These files can, however, be empty.

Note If CDB files produced by the Cisco MGC software on a Cisco PGW are not available for processing, the acc_x files will not be written for that interval.

Over the course of the day, the system continues to add to the trunk groups that are reported on, as call activity is received. Once added, a trunk group is reported on in every interval that follows, until the end of the day. At midnight, the system generates the acc_d (daily) file. This file contains all of the activity for the day for any trunk group that had call activity during the 24-hour period. Once the daily counts have been reported, the system attempts to clear out as many trunk groups from memory as possible. This step eliminates the need to report on trunk groups that are no longer active. The system purges any trunk group that does not currently have a threshold alarm asserted. These trunks groups must be retained so that the system does not assert additional alarms before the current alarm clears. If the system is running in configured mode, trunks specified in the Trunk Group table are not purged either.

Acc_x files produced after midnight contain only trunk groups that have had call activity after midnight and trunk groups that have threshold crossing alarms asserted. If the system is running in configured mode, trunks specified in the Trunk Group table are also reported.

Since the data is stored in flat files, you can configure MSC to purge outdated statistics.

Example from a MGC acc_h file:

```
0,972477302,3600,203,"occurrences","BAM:EGR CALL ATT","TG8004"
```

Statistics Output Format

The format for the statistics output mirrors the SS7-type statistics format created on the Cisco MGC. The format is comma-delimited, and appears in the order shown in Table 11-6.

Table 11-6 Output Format Order

Column	Description		Comments
Field 1	Record release level (version, 0 (zero) initially).		Constant: set to 0
Field 2	Time when the measurement interval started. The time is based on UNIX time format (seconds elapsed since January 1, 1970).		
Field 3	Elapsed time of collection interval, in seconds.		
Field 4	The value of the measurement at the end of the interval.		
Field 5	Measurement unit.		
Field 6	The measurement category. The direction is identified by EGR for egress, IGR for ingress, or TTL for total. The measurements are either carrier-based or noncarrier-based. The measurement name is one of the following:		
	Carrier-Based Measurements		
	Field Name	**Full Name**	
	IC EGR CALLS	IC Destined Calls	
	IC EGR NO CKT	IC Destined Calls, No Circuit	
	TTL DURATION	IC Duration	
	TTL CARRIERSELECT NO INDICATION	Total Carrier Select, No Indication	
	TTL CARRIERSELECT PRESUBSCRIBED NO NIPT	Total Carrier Select Presubscribed, Not Input	
	TTL CARRIERSELECT PRESUBSCRIBED INPT	Total Carrier Select, Presubscribed and Input	
	TTL CARRIERSELECT PRESUBSCRIBED WNI	Total Carrier Select, Presubscribed with No Indication	
	TTL CARRIERSELECT NOTPRESUBSCRIBED	Total Carrier ID Code, Not PreSubscribed but Input by Customer	
	Noncarrier-Bsed Measurements		
	Field Name	**Full Name**	
	IGR CALL ATT	Call Attempts Incoming	
	EGR CALL ATT	Call Attempts Outgoing	
	EGR CALL BLKD	Outgoing Attempts Blocked	
	TTL FAILED CONGEST	Failed Calls Congestion	
	IGR TERM NORM	Successful Calls Incoming	
	EGR TERM NORM	Successful Calls Outgoing	
	IGR PCT TRK USE	Percent Trunk Group Usage Incoming	
	EGR PCT TRK USE	Percent Trunk Group Usage Outgoing	

Table 11-6 Output Format Order (continued)

Column	Description		Comments
	TTL MAINT USE	Percent Trunk Group Maintenance Usage	
	TTL ERLANGS	Total Traffic in Erlangs	
	TTL TERM NORM	Total Calls Terminated Normally	
	TTL TERM ABNORM	Total Calls Terminated Abnormally	
	TTL TERM FAILED MGW	Total Calls Terminated, Failed MGW or NAS	
	TTL CALLS REJECTED	Total Calls Rejected	
	TTL REJECTED DIALNUM	Total Calls Rejected, Unknown Dialed Number	
	TTL REJECTED OTHER	Total Calls Rejected, Other Reasons	
	EGR OFL BLKD	Overflow, Outgoing Attempts Blocked	
	TTL TRAFFIC USAGE PEGS	Total Sum of Usage Pegs per Trunk Group	
	EGR TANDEM ATT	Tandem Routing Attempts, Outgoing	
	EGR TANDEM COMPLT	Tandem Completions, Outgoing	
	IGR TANDEM ATT	Tandem Attempts, Incoming	
	IGR TANDEM COMPLT	Tandem Completions, Incoming	
	EGR TANDEM DUR	Tandem Duration, Outgoing	
	IGR TANDEM DUR	Tandem Duration, Incoming	
	IGR CONV DURATION	Conversation Duration, Ingress	
	EGR CONV DURATION	Conversation Duration, Egress	
	IGR SETUP DURATION	Setup Duration, Ingress	
	EGR SETUP DURATION	Setup Duration, Egress	
	IGR TEARDOWN DURATION	Teardown Duration, Ingress	
	EGR TEARDOWN DURATION	Teardown Duration Egress	
	TTL CALL ROUTING I	Call Routing I Peg Total	
	TTL CALL ROUTING II	Call Routing II Peg Total	
	TTL CALL ROUTING III	Call Routing III Peg Total	
	EGR SUCCESSFUL H.323	Successful H.323 Terminating Peg	
	IGR SUCCESSFUL H.323	Successful H.323 Originating Peg	
	EGR UNSUCCESSFUL H.323	Unsuccessful H.323 Terminating Peg	
	IGR UNSUCCESSFUL H.323	Unsuccessful H.323 Originating Peg	
	EGR SUCCESSFUL ISUP	Successful Terminating ISDN User Part (ISUP) Requests	
	EGR UNSUCCESSFUL ISUP	Unsuccessful Terminating ISUP Requests	
	IGR SUCCESSFUL ISUP	Successful ISUP Originating Requests	
	IGR UNSUCCESSFUL ISUP	Unsuccessful ISUP Originating Requests	

Table 11-6 Output Format Order (continued)

Column	Description		Comments
	EGR ISDN SETUP MSG DELAY	Terminating ISUP Setup Message Response Delay	
	IGR ISDN SETUP MSG DELAY	Originating ISUP Setup Message Response Delay	
	TTL CIC DEFINED	Total Number of Defined CICs	
	TTL AVLBL CIC	Total Number of Available CICs	
	Note For IC calls, the string "IC:xxxx" where xxxx is the carrier number, precedes the EGR, IGR, or TTL.		
Field 7	Trunk Group "TG," followed by the zero-padded trunk group number.		
Field 8	TCA flag "*" if a threshold crossing alarm is asserted for this measurement, otherwise blank.		

Threshold Crossing Alarms

Each measurement instance can be monitored with a threshold crossing alarm. Threshold values that are permitted are Ignore, Less Than, Equal To, and Greater Than. The system identifies threshold value sets by the TAG and the trunk group number. Each threshold value set consists of a value and a check or test for each measurement category. Threshold value sets can be partially populated to check only one or any number of categories for a trunk group. Any unpopulated category is treated as an Ignore condition.

If no threshold value set has been specified for a given TAG/trunk group, the measurements are checked against a global threshold value set. Like trunk group-specific threshold value sets, the global threshold value set can be partially populated. There is no requirement to specify a global threshold value set. If none is specified, and no specific threshold value set has been entered, then no threshold checks are performed. If a threshold value set has been specified for a given TAG/trunk group, no global test is performed on any categories in that TAG/trunk group.

As measurements are tested against the threshold value set, each time a measurement crosses the threshold value, a minor alarm is generated (ACC227). The text of the alarm contains the strings defined in Appendix A, "Troubleshooting Cisco BAMS," the measured value, the test condition, and the threshold value.

When the threshold is crossed in the opposite direction, a clear alarm is generated containing the same text as the ACC227 alarm. For example, if the test is greater than 5 and the measurement is 8, the minor alarm is generated. If on the next check, the measurement is 10, no new minor alarm is generated. If the measurement drops to 3, the clear alarm is generated. When the system is started, the memory of all alarms is cleared. For example, suppose the measurement is 8 and the system is stopped. When the system first tests the measurement (after restarting), if the value is 8, a minor alarm is generated.

The following special conditions apply to threshold crossing alarms:

- No error is detected if a carrier is applied to a noncarrier-based measurement.
- No error is detected if no carrier is applied to a carrier-based measurement.
- A global threshold exists for TAG/trunk group measurements. The global threshold is specified by "global/0" (as the TAG/trunk group).
- Only those specific thresholds that are entered are checked; all other thresholds are set to ignore.
- If a trunk group-specific threshold is specified, the global thresholds are not checked for that TAG/trunk group.

- A carrier ID of 0 indicates that the carrier should be ignored. Entering abc/8003/0 is the same as entering abc/8003, thus making it a TAG/trunk group specification.
- All thresholds must be entered as integers.
- Conditions must be entered as a value from 0 through 4 (0 = Ignore, 1 = Less Than, 2 = Equal, 3 = Greater Than, 4 = Not Equal)

Note If there is no global/0 defined, any measurement that does not have a specific threshold set for it simply is not checked. The measurement is still reported in the acc_x file, but no alarm is generated, regardless of the value. If global/0 is defined, it is used when no specific thresholds have been specified for a trunk group. If the user sets thresholds for a specific trunk group, only those values specified are checked. Any unspecified measurements within the TAG/TRK are treated as an ignore condition. A trunk group can have a maximum of 43 measurements. A global TCA can be set up with a maximum of 43 measurements, which are listed in Table 11-3. Any trunk group that does not have a specific threshold crossing alarm (TCA) is checked against the global TCA. For some measurements, users can specify TAG/TRK/IC, where TAG is a user identifier, TRK is the trunk, and IC is the interexchange carrier. The user needs to know the carrier codes, such as 0288 for AT&T. Three-digit codes must be entered as four digits with a leading 0.

Zero Counts

The ACC task can operate in several different configurations with respect to zero counts. One configuration parameter outputs or suppresses all measurements that are equal to zero. The other configuration parameter selects all dynamic measurement group output or configured measurement group output regardless of activity.

Zero Count Suppression

Within each trunk group or trunk group/IC (measurement group), some measurements might not accumulate. For instance, if a trunk group is configured as an outgoing trunk, the ingress measurements are never pegged and the ingress durations are never anything other than zero. The ACC task provides a command-line switch to suppress these values. By default, if a measurement group has one measurement that is greater than zero, all measurements for the group are included in the output file. A command-line switch can override this feature and only non-zero values within each measurement group are output. If rounding or truncation causes an output measurement value to be zero, the ACC task treats the measurement as a zero and suppresses it if that feature is active.

Configured vs. Dynamic Trunk Group Output

Dynamic measurement groups are output only if they contain at least one non-zero measurement since midnight or have an alarm asserted. This is known as dynamic output. In BAMS, an MML option enables you to output all configured measurement groups only, regardless of measurement values (configured mode), to output dynamic measurement groups only (dynamic mode), or to output both configured and dynamic measurement groups. (For more information, see the dynamicaccumes field in the "Updating the Node Parameters Table" section on page 5-10.) This is a dynamic parameter that is reread at the start of each measurement interval. The trunk groups are also dynamic and are reloaded at the start of each

measurement interval. If BAMS is not set for configured mode (dynamic mode, or both) any activity detected on a nonconfigured trunk causes the trunk group to be added dynamically (as if in dynamic mode) and measurements are output.

If a trunk group is removed from the configuration, it no longer generates output if it has no counts accumulated for the day. The trunk group continues to be output if any counts for the day have accumulated. Likewise, if a trunk group is not configured and counts accumulate for that unconfigured trunk group (dynamic addition), the measurements for that trunk group are output for the remainder of the day.

The only distinction between a configured trunk with counts removed and a dynamic trunk with counts is that at the end of the day, the dynamic trunk has its pending alarms cleared if there are any. If a dynamically added trunk has an alarm pending at the end of the day, it continues to be reported into the next day and the alarm clears only when the threshold is crossed in the reverse direction.

Changing the overall mode to dynamic from configured causes any trunk groups with no counts accumulated for the day and no alarms to be removed from the output list. All other trunks are changed to dynamic. At midnight, all trunks are then treated as dynamic in the manner described above.

If the system is changed from dynamic to configured, all of the configured trunks are marked as configured, and any other trunks being reported prior to the mode change remain dynamic. All carrier-based measurements are dynamic. These cannot be preconfigured.

Rounding of Measurements

All measurements that are output as a percentage are rounded up or down to the nearest percent. This causes any percentage measurement that is less than 0.5 to round down to zero. The displayed value is zero, internally, but the ACC task maintains the decimal portion of the percentage. Under this condition, the ACC task considers the group to have at least one non-zero measurement. If the system is configured to suppress zero counts (with the NODEPARMS tag ID), the measurement is not displayed.

Truncation of Measurements

All measurements that are output as a duration are truncated to seconds. The ACC task performs all calculations to the millisecond. The truncation is applied only to the output measurement value. Any real-time duration that contains milliseconds is added to the hourly and daily totals with the milliseconds intact.

Last Interval Update

Introduction

Due to the manner in which the VSC produces data, BAMS must sometimes update the measurement data that was output in the previous interval. The VSC does not generate an event when a line is seized. The first event produced is an answer or an abort. Because of this, it is possible for a seizure to take place in one interval, and the answer or abort to take place in the next interval. When this happens, the ACC task determines what pegs or setup durations should be credited to an interval that has already been processed. Then the ACC task applies the measurements to the previously closed interval.

Preliminary vs. Final Measurements

The measurements for each interval are written twice. The first time the measurement file is written, the values are as accurate as possible, given the data provided by the VSC to that point. A measurement file that receives data at this time is located in a Measurements.tmp directory. This write takes place as quickly as the system can process the data following the detection of data that belongs to the next interval. Because some events might not have been signaled by the VSC (seizure), the counts might not be 100 percent accurate.

When the system detects data from the following interval, the system again processes the measurements. At this time, if events are present for calls that began in an interval prior to the current one, the prior interval measurement data is updated. This is the last time that the ACC task writes to the previous interval. Since during any interval, the ACC task will make the final write to the previous interval before making the preliminary write to the current interval, the data in any output file is final when a measurement file exists for a later interval. Final measurements files are located in the Measurements directory.

Interval-Update Rules

BAMS follows these rules when performing last-interval updates:

- Only the interval prior to the current interval can be updated.
- If pegs are detected that apply to an interval older than the previous interval, those pegs are applied to the previous interval. This ensures that the pegs are included in the hourly and daily totals. This also ensures that the sum of the intervals equals the daily and hourly totals.
- If durations are detected that apply to an interval older than the previous interval, those durations are dropped. This prevents any interval from possibly exceeding 100 percent utilization. The duration is not applied to the hourly or daily totals in order to ensure that the sum of the intervals equals the hourly and daily totals.
- On startup, there is no previous interval; therefore, the current interval is treated as the previous interval.
- The previous interval is updated before the preliminary measurements are written for the current interval.
- When a previous interval ends an hour, the hourly measurement file is also updated.
- When a previous interval ends a day, the daily measurement file is also updated.

Nonprovisioned Trunk Group Measurements

Measurements data is written for calls in which the trunk group is not provisioned in the Trunk Group table on BAMS. However, the following special rules apply to nonprovisioned trunk group measurements:

- When BAMS encounters the nonprovisioned trunk group, pegs are written for that trunk group, including 0 counts until midnight, when the memory is cleared.
- Any peg that requires the number of circuits to calculate will be suppressed. The number of circuits are maintained only for trunks defined in the Trunk Group table; therefore, BAMS has no knowledge of the number of circuits when the trunk group is not provisioned.
- We recommend checking the /opt/CiscoBAMS/files/s0x/FMT_cdr.log in which nonprovisioned trunk groups are reported. When you detect a nonprovisioned trunk group, configure the trunk group as soon as possible.

MGCP Dial and MGCP Scripting Handoff Measurements

In MGCP Dial and MGCP Script Handoff calls, the Egress trunk group (4015), Egress SigPath (4070), and Egress BearChan (4072) fields are not populated. Thus special treatment is required for these calls. A MGCP Dial or MGCP Scripting call is defined as a call where Egress Protocol (CDE Tag 4073 from 1010 or 1030 CDE) equals 9 or 10.

When processing MGCP Dial and MGCP Script calls, BAMS does not peg the following Egress measurements:

- BAM:EGR CALL ATT
- BAM:EGR CALL BLKD
- BAM:EGR TERM NORM
- BAM:EGR PCT TRK USE
- BAM:EGR OFL BLKD
- BAM:EGR TANDEM ATT
- BAM:EGR TANDEM COMPLT
- BAM:EGR CONV DURATION
- BAM:EGR SETUP DURATION
- BAM:EGR TEARDOWN DURATION

Troubleshooting Cisco BAMS

Introduction

This appendix provides a discussion of system tasks and messages generated by the Cisco Billing and Measurements Server (BAMS) and their use in troubleshooting. These messages, which can be categorized as alarms or events, are generated by the different software tasks running on BAMS. This appendix also provides information on the file-naming conventions used for files collected or processed by BAMS.

BAMS traps alarms and minor, major, and critical events, and forwards them to the operations management system (OMS).

BAMS writes alarms only to the alarm history file. When an alarm is cleared, its entry is deleted from that file.

Alarms and minor, major, or critical event messages to be forwarded to a message queue and trapped are defined with the Alarm Parameters table (ALM-PARMS tag ID), with the msgfwdlevel field (see the "ALM-PARMS Tag ID" section on page 5-2 for details).

Note All task messages are written to the system log (syslog) file. We recommend that the operations support personnel monitor the syslog file on a regular basis and note alarms and events that require troubleshooting. The bootstrap function can facilitate this task.

Note The following alarms are cleared when the alarm condition is removed: CRD502, MGR801, MSC030, MSC031, MSC032, MSC040, MSC041, MSC042, MSC130, MSC131, MSC132, MSC140, MSC141, MSC142, POL327, POL361, POL401, POL402, and POL115.

System Tasks

BAMS functions are governed by system tasks. Table A-1 displays BAMS non-SNMP tasks, and Table A-2 displays BAMS SNMP tasks. For information about the messages produced by these system tasks, see the "Generic Messages" section on page A-12.

Table A-1 BAMS Non-SNMP Tasks

Task	Explanation
ACC	The Accumulation task. This task accumulates numerical information on collected usage records.
ALM	The Alarm task. This task collects and distributes alarm and event information.
AUG (AGB)	The Augmentation task. This task uses lookup information to add value to the collected usage records. Messages related to BAF output are written to the system log with an Augmented BAF (AGB) task identifier.
ASC	The ASCII conversion task. This task converts the Cisco Media Gateway Controller (MGC) call detail record (CDR) files to ASCII files.
BIN	The BIN1110 output task. This task generates data in 1110 Binary format.
BTP	The bamstrap task. This task converts alarm messages into traps.
COR	The Correlation task. This task correlates records received from each Cisco MGC in a redundant pair configuration and produces a single-record format.
CRD	The Check Remote Down task. This task monitors communications between the two BAMS units and checks the status of the remote data collector.
CTB	The CDR to Bellcore AMA Format (BAF) conversion task. This task generates CDRs in BAF format.
DRX	The Message Receive task. This task is used by one BAMS unit to receive messages from another BAMS unit.
DTX	The Message Transmit task. This task transmits communications from one data collector to the next.
EXT	The Extendable ASCII task. This task generates data in extendable ASCII format.
FMT	The Formatting task. This task controls the initial formatting of data collected from the Cisco MGCs.
MGR	The Manager task. This task manages and spawns all other BAMS tasks.
MML (mml)	The Man-Machine Language (MML) task. This task displays the MML interface.
MSC	The Mass Storage Control task. This task manages the user-defined attributes governing the maintenance and removal of files from BAMS.
NIC	The NICS output task. This task generates data in NICS format.
NNL	The Node Name Link task. This task synchronizes MML commands to the remote BAMS unit.
POL	The Redundant Unit Polling task. This task collects data from (redundant) Cisco MGC units.

Table A-1 BAMS Non-SNMP Tasks (continued)

Task	Explanation
RCV	The Control File Receive task. This task is used by the BAMS unit to receive a control file. This task is executed by the POL task and is not a full-time running task.
SXF	The Send File Transfer task. This task sends a control file to a remote BAMS unit. This task is executed by the POL task and is not a full-time running task.

Table A-2 BAMS SNMP Tasks

Task	Explanation
bamsP3MIBagt	Sends out BAMS alarm messages as SNMP traps.
mscP3MIBagt	Returns parameter values of the MSC Thresholds table.
mscparmP3MIBagt	Returns parameter values of the MSC Parameters table.
nodeparmP3MIBagt	Returns parameter values of the Node Parameters table. This task also returns node name and poll status information for each node.
pollP3MIBagt	Returns parameter values of the Poll table.

Procedure to Recover from Active/Active Polling State

The normal state of the redundant BAMS is active/standby, where one unit is actively polling the Cisco MGC and the other unit is in a standby mode. Under certain error conditions, the BAMS unit can enter an active/active polling state. The following procedure explains how to recover from the undesired active/active polling state.

The polling software was designed so the worst-case scenario would be a standby/standby state, because in this state neither BAMS unit is polling and data is accumulating on the VSC. In safeguarding the system so a standby/standby state would not occur, certain race conditions (where a possible standby/standby state might occur) have been taken care of by defaulting to an active/active polling state. There are few cases in which this would happen, but defaulting to an active/active polling state is a safe method of avoiding any critical problems.

The active/active polling state is not a critical state because in this mode, the BAMS unit that has the correct sequence numbers can continue polling, and the other BAMS unit, although in an active state, cannot poll the files that the first BAMS has already polled. Also, the processing would not be interrupted because the BAMS unit that can poll forwards the data to the other BAMS unit for redundant processing.

To reset the system to a normal active/standby state, you must update the system that is not polling and has the incorrect sequence numbers. Updating can be accomplished by executing a rotation (sw-ovr) from active to standby on the BAMS unit that has the correct sequence numbers; that is, the unit that is actually polling files. The rotation from active to standby causes this BAMS unit to go to standby, and the other BAMS unit that was active but was not able to poll is updated with the correct polling sequence numbers. Once the sw-ovr is executed, the active unit can poll the VSC successfully, and a normal active/standby state results.

File-Naming Conventions

For consistency, the application requires that the data sources follow certain file-naming conventions.

The convention is to use a unique prefix per file format, followed by a time stamp in UTC that indicates when the file was created or generated on the data source, and a file sequence number.

Input and Output File-Naming Conventions

The following file-naming conventions are used for BAMS input and output files.

ASCII BAF Output

The file-naming convention for data collected from the Cisco MGC and converted into ASCII BAF record format is as follows:

baf_<raw file prefix>_YYYYMMDDHHMMSS_SeqNo.csv

Table A-3 explains each element of the filename. Note the use of the underscore.

Table A-3 ASCII BAF Filename Elements

Element	Description
baf	Represents the fixed prefix for ASCII BAF output files
raw file prefix	Represents the raw filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.csv	Represents the suffix

BAF Output

The file-naming convention for BAF billing record files is as follows:

baf_<raw file prefix>_YYYYMMDDHHMMSS_SeqNo.bin

Table A-4 explains each element of the filename. Note the use of the underscore.

Table A-4 BAF Billing Record Filename Elements

Element	Description
baf	Represents the fixed prefix for BAF output files
raw file prefix	Represents the raw filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004

Table A-4 BAF Billing Record Filename Elements (continued)

Element	Description
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.bin	Represents the suffix

Investigating BAMS CDR Output Records

To investigate BAMS billing output records, start by retrieving the archived PGW CDR records, which are located in the node TLV directory. The TLV directory contains most of the information that populates the BAMS output. It is important to notice that the sequence number of the file in the TLV directory may not directly correspond to the sequence number of the BAMS output file. Furthermore, the sequence of individual records recorded in the BAMS file may not be identical to the sequence of records recorded in the TLV file.

BIN1110 Output

The file-naming convention for data collected from the Cisco MGC and converted into 1110 Binary record format is as follows:

cdr_YYYYMMDDHHMMSS_SeqNo.BIN

Table A-5 explains each element of the filename. Note the use of the underscore.

Table A-5 BIN1110 Filename Elements

Element	Description
cdr_	Represents the prefix of the files generated by the Cisco MGC.
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.BIN	Represents the suffix (specified in the BIN1110 table)

Extendable ASCII Output

The file-naming convention for data collected from the Cisco MGC and converted into extendable ASCII record format is as follows:

extasc_<raw file prefix>_YYYYMMDDHHMMSS_SeqNo.csv

Table A-6 explains each element of the filename. Note the use of the underscore.

Table A-6 Extendable ASCII Filename Elements

Element	Description
extasc	Represents the fixed prefix for EXTASCII output files
raw file prefix	Represents the raw filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.csv	Represents the suffix

Input CDR Files

The file-naming convention for data collected from the Cisco MGC is as follows:

prefix_YYYYMMDDHHMMSS_SeqNo.bin

Table A-7 explains each element of the filename. Note the use of the underscore.

Note After the file is polled from the Cisco MGC, FMT (the first BAMS processing task) renames it according to the UTC timestamp inside the file (CDB tag 400 of the file header record). For this reason there may be a slight discrepancy between the original timestamp in the filename and the timestamp in the BAMS filename. All BAMS tasks that process the file after this point use the UTC-based filename.

Table A-7 CDR Filename Elements

Element	Description
prefix	Represents the filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23

Table A-7 CDR Filename Elements (continued)

Element	Description
SeqNo	Represents the sequence number; for example, _000051
.bin	Represents the suffix

MGC End of Call 1110 ASCII Output

The file-naming convention for data collected from the Cisco MGC and converted into ASCII 1110 record format is as follows:

prefix_YYYYMMDDHHMMSS_SeqNo.csv

Table A-8 explains each element of the filename. Note the use of the underscore.

Table A-8 ASCII 1110 Filename Elements

Element	Description
prefix	Represents the filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.csv	Represents the suffix

NICS Output

The file-naming convention for data collected from the Cisco MGC and converted into NICS record format is as follows:

CDR.<switch_name>.<SeqNo>.YYYYMMDDHHMMSS

Table A-9 explains each element of the filename. Note the use of the underscore.

Table A-9 NICS Filename Elements

Element	Description
CDR.	Represents the fixed prefix for NICS output files
switch_name	Represents the switch name. It is taken from TAG6000 of the data produced by the PGW (that is, it denotes the configured switch name for the PGW)

Table A-9 NICS Filename Elements (continued)

Element	Description
SeqNo	Represents the sequence number; for example, 1234 **Note** For Cisco BAMS Release 3.20, the sequence number is no longer derived from the Cisco PGW 2200 raw data filename. Instead, the sequence number is synchronized between BAMS units per node.
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23

P01 Output

The file-naming convention for data collected from the Cisco MGC and converted into the P01 record format is as follows:

p01_<raw file prefix>_YYYYMMDDHHMMSS_SeqNo.bin

Table A-10 explains each element of the filename. Note the use of the underscore.

Table A-10 P01 Filename Elements

Element	Description
p01	Represents the fixed prefix for P01 output files
raw file prefix	Represents the raw filename prefix; for example, cdr_
YYYY	Represents the year; for example, 2004
MM	Represents the month; for example, 05
DD	Represents the day; for example, 15
HH	Represents the hour; for example, 19
MM	Represents the minute; for example, 46
SS	Represents the second; for example, 23
SeqNo	Represents the sequence number; for example, _000051
.bin	Represents the suffix

Task Log Files

Log files are used to save debug or status information for a particular task. These files are saved to the /opt/CiscoBAMS/files/s01-s08 individual node directories. Here is the filename format:

task-identifier.log

Table A-11 explains each element of the filename. Note the use of the underscore.

Table A-11 Task Log Filename Elements

Element	Description
task_identifier	Tag to identify the task to which this log file refers; for example, the FMT task. Sometimes followed by a qualifier; for example, _cdr.
.log	Fixed suffix to indicate that the file is a log file.

Task Configuration Files

Task configuration files pass all the necessary command-line switches that are needed by the MGR task to properly execute a particular task. These files are stored in the /opt/CiscoBAMS/files/s01-s08 individual node directories. Here is the filename format:

*task_identifier{task_instance}*_cfg

Table A-12 explains each element of the filename. Note the use of the underscore.

Table A-12 Task Configuration Filename Elements

Element	Description
task_identifier	Tag to identify the task to which this configuration file refers.
task_instance	Optional instance sequence. For example, if there are two instances of the COR task running, the filenames might be named cor0_cfg and cor1_cfg.
_cfg	Fixed suffix to indicate that the file is a task configuration file.

Task Parameter Files

Task parameter files contain parameter definitions for a particular task. These files are stored in the /opt/CiscoBAMS/files directory. These files are in ASCII format. The filenames used are shown in Table A-13. Here is the filename format:

task_identifier{*task_instance*}.cfg

Table A-13 Task Parameter Filename Elements

Element	Description
task_identifier	Tag to identify the task to which this configuration file refers.
task_instance	Optional instance sequence. For example, if there are two instances of the COR task running, the filenames might be named cor0.cfg and cor1.cfg.
.cfg	Fixed suffix to indicate that the file is a task configuration file.

System Log File (syslog)

BAMS stores system activity in the system log file (syslog) in the format explained in Table A-14. Located in the /CiscoBAMS/files directory, the syslog is composed of individual log entries that follow this format example: *MM/DD/YY hh:mm:ss severity message ID: +message text*.

Table A-14 Syslog Format

Item	Description
MM/DD/YY	The date that BAMS wrote the entry to the file.
hh:mm:ss	The time that BAMS wrote the entry to the file.
severity	If indicated, represents the importance level of the message: *C indicates a critical event or alarm. ** indicates a major event or alarm. * indicates a minor event or alarm. *R indicates a resolved event or alarm. W indicates a warning event. M indicates a manual problem (operator intervention required). Two blank spaces indicate an informational event. DB indicates a debug event.
message ID	A three-character task name that generated the message, followed by a three-digit code that identifies the particular message.
message text	The brief message or description of the alarm or event. If a plus sign (+) is included it indicates that the line is a continuation of the previous line.

Log File History

The syslog file holds a user-specified number of messages. When the syslog file's maximum file size is reached, BAMS changes the name of the syslog file to oldlog and creates a new syslog file.

While only one syslog file can exist at a time, BAMS can retain multiple oldlog files. BAMS creates additional oldlog files in the same manner that it creates syslog files. If an oldlog file exists when it is time to create a syslog, BAMS renames the oldlog file to oldloga. If more than two oldlog files exist, BAMS uses the letters b, c, d, and so on to identify the oldest oldlog file. (For example, oldlogd is older than oldlogc.)

Message Description

The following categories of information are provided for each system message:

- Message ID
- Text
- Arguments
- Description
- Action

Message ID

The message ID is a six-character label (three characters followed by three digits) that uniquely identifies each message. The first three characters are the task ID, which identifies the task that generated the message (for example, MGR denotes the Manager task and MSC denotes the Mass Storage Control task). The three digits represent the message number (for example, 013 or 123).

Note In the task message section below, minor, major, critical, and warning alarms are preceded by a one- or two-character signifier (* = minor, ** = major, *C = critical, and W = Warning) before the message ID. In some cases, such as for MSC alarms, a message may be generated as minor, major, or critical, depending on the system state. In such cases, the range of levels is represented between parentheses: (*, **, *C). While informational messages are preceded by two blank spaces in the syslog, they are not preceded by any characters in the task message section.

Text

The text is the verbal part of the message as it appears on your screen or in the system log file.

Arguments

Arguments are variable parts of the message (message arguments) and are enclosed in angle brackets (< >). Explanations of arguments are provided.

Description

The message description explains the event(s) that generated the message.

Action

The action describes what you should do as a result of the event described in the message. In some cases (for example, informational messages), no action might be required. Actions for error messages (manual, warning, minor, major, and critical) might include steps that should be followed so that you can identify and correct problems. Error actions can also describe how BAMS responds to the specified error condition.

Alarm Task Messages

The Alarm (ALM) task handles all alarm and event messages that are written to the system log. This includes clearable alarms that are based on user-defined thresholds and are written to an alarm history file.

`ALM100 ALM Version <version> has started.`
The <version> argument represents the task version. This message indicates that the ALM task has started. This is an informational message. No action is necessary.

`* ALM102 <routine> failed on error <UNIX error number>`
The <routine> argument represents the routine name. The <UNIX error number> argument represents the UNIX error set by the system function. This message indicates that a system failure occurred. Check the UNIX error number for an explanation of this error.

** ALM103 <string>() failed on error <number>
The <number> represents the UNIX error number. This message indicates that the ALM task encountered a serious error condition. This is a major event. Examine the error number and contact the Cisco Technical Assistance Center.

ALM104 get_qid (<name>) returned <error> (errno <UNIX error number>)
The <name> argument represents the queue ID (QID). The <error> argument represents the returned error. The <UNIX error number> argument represents the UNIX error set by the system function. This message indicates that BAMS could not get the QID from the acecqid file. Verify that the $FILES environment variable is located in the acecqid file. Make sure that the read permissions are correct.

* ALM105 msgget <function> failed (errno <UNIX error number>)
The <function> argument represents the function name. The <UNIX error number> argument represents the UNIX error set by the system function. This message indicates that the ALM task experienced an error. Check the UNIX error number for an explanation of this error.

* ALM106 Can not load <name>
The <name> argument represents the parameter name. This message indicates that the ALM task experienced an error.

* ALM108 Cannot access <string> (errno <UNIX error number>)
The <UNIX error number> argument represents the UNIX error set by the system function. This message indicates that the ALM task experienced an error. Check the UNIX error number for an explanation of this error.

ALM120 Alarm parameters have been updated
This message indicates that alarm parameters have been updated. This is an informational message. No action is necessary.

Generic Messages

The MGR task controls BAMS operations and uses what is called generic messaging. This means that message numbers can be used by more than one software task (ACC001, BTP001, and MGR001, for example). The text of the message is always the same, although the arguments, if any, might be different. The system messages are numbered from 000 through 399 (not all numbers are used).

Note All system messages start with a date and time stamp (not shown below). The standard three-character task identifier is denoted by XXX in the generic messages that follow.

XXX000 <task> Version <number> has started, pid <ID>.
The <task> argument represents the task mnemonic, <version> represents the task version number, and <ID> represents the UNIX process ID. The task logs this message when it is started.

XXX001 <task> Initialization complete.
The <task> argument represents the task mnemonic. The task logs this message when initialization is complete.

XXX002 Task completed, pid <process id>
The <process id> argument represents the UNIX process ID. The task has completed its job successfully.

(*,**, or *C) XXX003 Exiting with error code <error code>, pid <process id>
The <error code> argument represents the error code, and <process id> represents the UNIX process ID. The task is ending due to an error. Refer to a previous message from the same task in the syslog to get more detailed information.

(*,**,or *C) XXX004 Terminating on signal <signal code>, pid <process id>

The <signal code> argument represents the signal code, and <process id> represents the UNIX process ID. The task is ending abnormally due to the receipt of an unexpected signal. Refer to a previous message by the same task in the syslog to get more detailed information.

`XXX005 Sending <message> to <task or task component>`
The <message> argument represents the information or message sent, and <task> represents the task or task component recipient. The task sent information or a message to another task or task component.

`XXX006 Waiting for <message> message`
The <message> argument represents the message being waited for. The task is attempting to receive some information from another task or task component.

`XXX007 Received <item>`
The <item> argument represents the item or message received. The task has received some information from another task or task component.

`XXX008 Processing <item to process>`
The <item to process> argument represents the item to process, such as filename or message type name. The task is processing an item.

`XXX009 Successfully processed <processed item>`
The <processed item> argument represents the processed item. The task has successfully processed the indicated item.

`XXX010 System state is <system state>`
The <system state> argument represents the system state (for example, Running, Booting, Active, or Standby). For a redundant system, this message is displayed by the task, if its actions are dependent on the system state.

`XXX011 SIG setting signal <signal number> handler to <signal handler name>`
The <signal number> argument represents the signal number, and <signal handler name> represents the signal handler name. The task is setting the handler for the given signal.

`XXX012 SHM Creating shared memory key= <shared memory key> size= <size of shared memory requested>`
The <shared memory key> argument represents the shared memory key, and <size of shared memory requested> represents the size of shared memory requested. The task allocates shared memory as a mechanism for interprocess communication.

`XXX013 SHM Attaching to shared memory key= <shared memory key> size= <size of shared memory>`
The <shared memory key> argument represents the shared memory key, and <size of shared memory> represents the size of shared memory. The task attaches to a region in shared memory.

`XXX014 MSGQ Creating message queue key= <message queue key>`
The <message queue key> argument represents the message queue key. The task allocates a message queue as a mechanism for interprocess communication.

`XXX015 SEMA Creating semaphore key= <semaphore key>`
The <semaphore key> argument represents the semaphore key. The task allocates a semaphore as a mechanism for interprocess communication.

`XXX016 DIR Changing to <directory name>`
The <directory name> argument represents the directory name. The task is setting its current work directory.

`XXX017 Renaming <original filename> to <new filename>`
The <original filename> argument represents the original filename, and <new filename> represents the new filename. The task renames (or move) a file, typically to mark it as processed.

`XXX018 Connecting to <remote host address or name> / <port number or service name>`

The <remote host address or name> argument represents the remote host address or name, and <port number or service name> represents the port number or service name. The task attempts to connect to the remote host (or server). If the task uses a TCP/IP socket and the second argument is supplied as a service name, the service must be registered in the /etc/services file. In addition, when the host name is used instead of an IP address, the host name must be registered in the /etc/hosts file.

`XXX019 Connected to <remote host address or name> / <port number or service name>`
The <remote host address or name> argument represents the remote host address or name, and <port number or service name> represents the port number or service name. The task has successfully connected to the named host. At this point, application-layer communication can continue.

`XXX020 Disconnecting from <remote host address or name> / <port number or service name>`
The <remote host address or name> argument represents the remote host address or name, and <port number or service name> represents the port number or service name. The task closes the connection to the remote server. To find out why the task is disconnecting, check previous messages in the syslog file.

`XXX100 Timed out waiting for <description>.`
The <description> argument represents the description of what the task was waiting for. The task logs a message when the attempt to receive some information from another task or task component has exceeded a certain time threshold.

`(*, **, or *C) XXX101 Invalid command-line option, <unsupported character>.`
The <unsupported character> argument represents the unsupported character command-line switch. The task was passed an unknown or unsupported command-line argument. The configuration file for the task contains an invalid option. To check all available options, invoke the task using the **-h** option, and check the contents of the configuration file.

`(*, **, or C) XXX102 Unrecoverable failure.`
The task cannot recover from an error. Analyze previous or following entries in the syslog file. These messages provide more information about the possible cause of the failure. Contact the Cisco Technical Assistance Center.

`(*, **, *C) XXX103 SIG errno <error number> setting handler for signal <signal number>`
The <error number> argument represents the UNIX error number, and <signal number> represents the signal number. The task failed to set the handler for the given signal. Contact the Cisco Technical Assistance Center if necessary.

`(*, **, *C) XXX104 SYS errno <error number> on system() for <argument>`
The <error number> argument represents the UNIX error number, and <argument> represents the description of the argument passed to the system() call. The task detected an error when passing a command to the UNIX shell. The shell might be unable to find and execute the specified command. Check if the command or script passed to the system call is valid. If it is a script, verify that the execute permissions are set. Another possible reason for a system call to fail is an excessive number of UNIX processes already executing. To see if this is the case, issue the **ps -eaf | pg** command.

`(*, **, *C)XXX105 SYS errno <error number> on fork() for <description>`
The <error number> argument represents the UNIX error number, and <description> indicates why the process is cloning itself. The task failed to clone itself. This might happen if the maximum number of processes in the system has been reached. To see if this is the case, issue the **ps -eaf | pg** command.

`(*, **, *C)XXX106 ENV variable <name> undefined`
The <name> argument represents the name of the undefined environment variable. A variable is undefined in the run-time environment of the task. Check the sym_defs file if there is no entry for the given environment variable. If there is an entry, see if the variable is exported.

`(*, **, *C)XXX107 ENV variable <name> is undefined, using default value <default string>`
The <name> argument represents the name of the undefined environment variable, and <default string> represents the default string to be used. A variable is undefined in the run-time environment of the task. The task is using a default string value.

`(*, **, *C)XXX108 ENV variable <name> is undefined, using default value <default string>`

The <name> argument represents the name of the undefined environment variable, and <default string> represents the default string to be used. A variable is undefined in the run-time environment of the task. The task is using a default numeric value.

`(*, **, *C)XXX109 SHM errno <error number> code <return code> getting key for <name>`

The <error number> argument represents the return code of the function that provides the mapping between shared memory names and the shared memory keys, <return code> represents the UNIX error number, and <name> represents the shared memory name.

The task cannot get the shared memory key value for the given shared memory name. Either the acecmid file is not in the files directory or the named shared memory is not defined in the acecmid file. Contact the Cisco Technical Assistance Center to update this file.

`(*, **, *C)XXX110 SHM errno <error number> creating key=$ <memory key>`

The <error number> argument represents the UNIX error number, and <memory key> represents the shared memory key. The task failed to create the shared memory with the given key. A shared memory with the same key might already exist.

`(*, **, *C)XXX111 SHM errno <error number> attaching to key=$ <memory key>`

The <error number> argument represents the UNIX error number, and <memory key> represents the shared memory key. The task failed to attach to the shared memory with the given key. The shared memory with the given key might not yet have been created or might have been removed.

`(*, **, *C)XXX112 SHM chunk not defined for <memory chunk>`

The <memory chunk> argument represents the name of the memory chunk that caused the lookup failure. The task failed to look up a memory chunk in the shared memory region. Check the configuration file in the files directory.

`(*, **, *C)XXX113 SHM chunk is <assigned memory> bytes for <expected memory> Expecting <chunk size>`

The <assigned memory> argument represents the memory chunk size, <expected memory> represents the expected memory chunk size, and <chunk size> represents the memory chunk size in bytes. The expected memory chunk size of the task does not match the allocated chunk. Check the configuration file in the files directory.

`(*, **, *C)XXX114 MSGQ errno <error number> code <return code> getting key for queue name`

The <error number> argument represents the UNIX error number, <return code> represents the return code of the function that provides the mapping between the message queue names and message queue keys, and <queue name> represents the message queue name.

The task cannot get the message queue key value for the given message queue name. Either the acecqid file is not in the files directory or the named message queue is not defined in the acecqid file.

`(*, **, *C)XXX115 MSGQ errno <error number> creating key=$ <queue key>`

The <error number> argument represents the UNIX error number, and <queue key> represents the message queue key. The task cannot create a message queue with the given key. A message queue with the same key might already exist. Look in /CiscoBAMS/files/acecqid for duplicate key definitions. Contact the Cisco Technical Assistance Center to correct the configuration if duplicates are found.

`(*, **, *C)XXX116 MSGQ errno <error number> code <return code> on sendlocal() key=$ <queue key>`

The <error number> argument represents the UNIX error number, <return code> represents the return code of sendlocal(), and <queue key> represents the message queue key.

The task failed to send a message to a task on the same machine. The message queue with the given key might have been removed, or the system-imposed maximum number of pending messages might have been reached. From the UNIX prompt, issue the command **ipcs -qob**. Check if the queue with the given key still exists and if the total QNUM is less than 40.

(*, **, *C)XXX117 MSGQ errno <error number> code <return code> on sendremote() key=$<queue key>

The <error number> argument represents the UNIX error number, <return code> represents the return code of sendremote(), and <queue key> represents the message queue key.

The task failed to send a message to a task on a remote machine. The message queue for the task dealing with intermachine communication might have been removed, or the system-imposed maximum number of pending messages might have been reached. From the UNIX prompt, issue the command **ipcs -qob**. Check if the queue with the given key still exists and if the total QNUM is less than 40.

(*, **, *C)XXX118 MSGQ errno <error number> code <error number> on recvmsg() key=$<queue key>

The <error number> argument represents the UNIX error number, <return code> represents the return code of recvmsg(), and <queue key> represents the message queue key. The task failed to receive a message from a message queue with the given key. The message queue with the given key might have been removed, or the task might have been interrupted.

(*, **, *C)XXX119 SEMA errno <error number> code <return code> getting key for <semaphore name>

The <error number> argument represents the UNIX error number, <return code> represents the return code of the function that provides the mapping between semaphore names and semaphore keys, and <semaphore name> represents the semaphore name. The task cannot get the semaphore key value for the given semaphore name. Either the acecsid file is not in the files directory, or the named semaphore is not defined in the acecsid file.

(*, **, *C)XXX120 SEMA errno <error number> creating key=$<semaphore key>

The <error number> argument represents the UNIX error number, and <semaphore key> represents the semaphore key. The task cannot create the semaphore with the given key. A semaphore with the same key might already exist.

(*, **, *C)XXX121 FILE errno <error number> opening <name> for <file open>

The <error number> argument represents the UNIX error number, <name> represents the name of the file, and <file open> represents the file open modifier (for example, create, read, or append). The task failed to open the named file using the indicated file open operation.

(*, **, *C)XXX122 FILE errno <error number> reading <filename> size <total byte size>

The <error number> argument represents the UNIX error number, <filename> represents the filename or item being read, and <total byte size> represents the total byte size available to read. The task failed to read the requested number of bytes.

(*, **, *C)XXX123 FILE errno <error number> writing <filename> size <total byte size>

The <error number> argument represents the UNIX error number, <filename> represents the filename or item being written, and <total byte size> represents the total byte size available to write. The task failed to write the requested number of bytes. Check if the disk is full and if the file exists.

(*, **, *C)XXX124 FILE errno <error number> seeking <filename> offset <offset> whence <flag>.

The <error number> argument represents the UNIX error number, <filename> represents the name or description of the input file, <offset> represents the requested offset, and <flag> represents the seek flag (0, 1, or 2). The task failed to move the pointer on the named file.

(*, **, *C)XXX125 FILE fprintf() returned <return code>. Expecting <number>.

The <return code> argument represents the return code of fprintf(), and <number> represents the number of items requested. The task failed to write the requested number of items.

(*, **, *C)XXX126 FILE fscanf() returned <return code>. Expecting <number>

The <return code> argument represents the return code of fscanf(), and <number> represents the number of items requested. The task failed to read the requested number of items.

(*, **, *C)XXX127 FILE errno <error number> linking <file> to <name>
The <error number> argument represents the UNIX error number, <file> represents the file to link from, and <name> represents the name of the new file link. The task failed to create a link to a file. This might happen when the first file does not exist.

(*, **, *C)XXX128 FILE errno <error number> unlinking <unlink>.
The <error number> argument represents the UNIX error number, and <unlink> represents the name of the file to unlink. The task failed to remove a file link. Check if the named file exists. If it does, check the permissions in the file.

(*, **, *C)XXX129 FILE errno <error number> renaming <rename> to <new name>
The <error number> argument represents the UNIX error number, <name> represents the original filename, and <new name> represents the new name. The task failed to rename a file.

(*, **, *C)XXX130 DIR errno <error number> changing to <directory>.
The <error number> argument represents the UNIX error number, and <directory> represents the directory name. The task failed to set its working directory. Check if the directory exists and if the execute permissions are set.

(*, **, *C)XXX131 DIR errno <error number> opening <directory>.
The <error number> argument represents the UNIX error number, and <directory> represents the directory name. The task failed to open the named directory. Check if the directory exists and if the read permissions are set.

(*, **, *C)XXX132 DIR errno <error number> reading <error number>
The <error number> argument represents the UNIX error number, and <directory name> represents the directory name. The task failed to read an entry from the named directory. Check if the directory exists and if the read permissions are set.

(*, **, *C)XXX133 SCKT errno <error number> creating <protocol> socket for <socket>
The <error number> argument represents the UNIX error number, <protocol> represents the protocol (TCP or UDP), and <socket> represents the use of the socket. The task failed to create an end point for communication (a socket). This can happen if the task has opened too many sockets.

(*, **, *C)XXX134 SCKT errno <error number> connecting to <address>/<port number>
The <error number> argument represents the UNIX error number, <address> represents the server name or address, and <port number> represents the service name or port number (as a string).

The task failed to initiate a connection to a server process. This can be due to one of two things: the server name does not have an entry in the /etc/hosts file, or the service name is not in the /etc/services file. If an IP address and a port number are used instead of names and the message still shows up, contact the Cisco Technical Assistance Center.

(*, **, *C)XXX135 SCKT errno <error number> binding socket <socket>
The <error number> argument represents the UNIX error number, and <socket> represents the socket descriptor. The task failed to link a name to a socket.

(*, **, *C)XXX136 SCKT errno <error number> listening on socket <socket>, qlength=<queue length>
The <error number> argument represents the UNIX error number, <socket> represents the socket descriptor, and <queue length> represents the requested queue length. The task failed to listen for connections on a socket.

(*, **, *C)XXX137 SCKT errno <error number> accepting from socket=<socket>
The <error number> argument represents the UNIX error number, and <socket> represents the socket ID. The task failed to accept a connection from a socket.

(*, **, *C)XXX138 SCKT errno <error number> reading <description> size <number>
The <error number> argument represents the UNIX error number, <description> represents the description of data to read, and <number> represents the number of bytes to read. The task encountered an error while reading data from a socket.

(*, **, *C)XXX139 SCKT errno <error number> writing <description> size <number>
The <error number> argument represents the UNIX error number, <description> represents the description of data to write, and <number> represents the number of bytes to write. The task encountered an error while writing data to a socket.

(*, **, *C)XXX140 FILE errno <error number> getting file stat for <name>
The <error number> argument represents the UNIX error number, and <name> represents the name or description of a file. The task failed to get file statistics. Check if the file exists.

(*, **, *C)XXX141 MSGQ errno <error number> code <return code> deleting key=$<queue key>
The <error number> argument represents the UNIX error number, <return code> represents the return code of the routine for deleting the message queue, and <queue key> represents the message queue key. The task failed to delete the message queue with the given key. The message queue might have been deleted manually or by another task.

(*, **, *C)XXX142 SHM errno <error number> code <return code> deleting key=$<queue key>
The <error number> argument represents the UNIX error number, <return code> represents the return code of the routine to delete the shared memory, and <queue key> represents the message queue key. The task failed to delete the shared memory with the given key. The shared memory might have been deleted manually or by another task.

(*, **, *C)XXX143 SHM errno <error number> attaching to <name> shared memory
The <error number> argument represents the UNIX error code, and <name> represents the task name of shared memory owner. This alarm message appears only if the task does not explicitly acquire the shared memory ID through get_mid().

The task failed to attach to the shared memory owned by the given task. The shared memory might not have been created, or it might have been deleted by the owning process. Check if the shared memory owner is executing by issuing the UNIX command **ps -eaf | grep** <task>, where <task> is the owner of the shared memory.

(*, **, *C)XXX200 Timed out waiting for <task>
The <task> argument represents the description of what the task was waiting for. The task logs the message when the attempt to receive some information from another task or task component has exceeded a certain time threshold.

(*, **, *C)XXX201 Invalid command-line option, <switch>
The <switch> argument represents the unsupported character command-line switch. The task was passed an unknown or unsupported command-line argument. The configuration file for the task contains an invalid option. To check all available options, invoke the task using the **-h** option and check the contents of the configuration file.

(*, **, *C)XXX202 Unrecoverable failure
The task cannot recover from an error. Previous or following entries in the log file provide more information on the possible cause of the failure.

(*, **, *C)XXX203 SIG errno <error number> setting handler for signal <signal>
The <error number> argument represents the UNIX error number, and <signal> represents the signal number. The task failed to set the handler for the given signal. Contact the Cisco Technical Assistance Center if necessary.

(*, **, *C)XXX204 SYS errno <error number> on system() for <argument>
The <error number> argument represents the UNIX error number, and <argument> represents the description of the argument passed to the system() call.

The task detected an error when passing a command to the UNIX shell. The shell may be unable to find and execute the specified command. Check if the command or script passed to the system call is valid. If it is a script, verify that the execute permissions are set. A system() call might also fail if an excessive number of UNIX processes are already executing. To see if this is the case, issue the command **ps -eaf | pg**.

`(*, **, *C)XXX205 SYS errno <error number> on fork() for <description>`
The <error number> argument represents the UNIX error number, and <description> represents the description of why the process is cloning itself. The task failed to clone itself. This might happen when the maximum number of processes in the system has been reached. To see if this is the case, issue the command **ps -eaf | pg**.

`(*, **, *C)XXX206 ENV variable <name> undefined`
The <name> argument represents the name of the undefined environment variable. A variable is undefined in the run-time environment of the task. Check the sym_defs file to verify there is an entry for the given environment variable. If there is an entry, verify that the variable is exported.

`(*, **, *C)XXX207 ENV variable <name> is undefined, using default value <default string>`
The <name> argument represents the name of the undefined environment variable, and <default string> represents the default string to be used. A variable is undefined in the run-time environment of the task. The task uses a default string value.

`(*, **, *C)XXX208 ENV variable <variable> is undefined, using default value <default string>`
The <variable> argument represents the name of the undefined environment variable, and <default string> represents the default string to be used. A variable is undefined in the run-time environment of the task. The task is using a default numeric value.

`(*, **, *C)XXX209 SHM errno <error number> code <return code> getting key for <memory name>`
The <error number> represents the UNIX error number, <return code> represents the return code of the function that provides the mapping between shared memory names and shared memory keys, and <memory name> represents the shared memory name.

The task cannot get the shared memory key value for the given shared memory name. Either the acecmid file is not in the files directory or the named shared memory is not defined in the acecmid file. Contact the Cisco Technical Assistance Center if necessary.

`(*, **, *C)XXX210 SHM errno <error number> creating key=$<memory key>`
The <error number> argument represents the UNIX error number, and <memory key> represents the shared memory key. The task failed to create the shared memory with the given key. A shared memory with the same key might already exist.

`(*, **, *C)XXX211 SHM errno <error number> attaching to key=$<memory key>`
The <error number> argument represents the UNIX error number, and <memory key> represents the shared memory key. The task failed to attach to the shared memory with the given key. The shared memory with the given key might not yet have been created or it might have been removed.

`(*, **, *C)XXX212 SHM chunk not defined for <lookup failure>`
The <lookup failure> argument represents the name of the memory chunk that caused the lookup failure. The task failed to look up a memory chunk in the shared memory region. Check the configuration file in the files directory.

`(*, **, *C)XXX213 SHM chunk is <assigned> bytes for <name>. Expecting <size> bytes`
The <assigned> argument represents the assigned memory chunk size, <name> represents the memory chunk name, and <size> represents the expected memory chunk size. The expected memory chunk size of the task does not match the allocated chunk. Check the configuration file in the files directory.

`(*, **, *C)XXX214 MSGQ errno <error number> code <return code> getting key for <queue name>`
The <error number> argument represents the UNIX error number, <return code> represents the return code of the function that provides the mapping between message queue names and message queue keys, and <queue name> represents the message queue name.

The task cannot get the message queue key value for the given message queue name. Either the acecqid file is not in the files directory or the named message queue is not defined in the acecqid file.

(*, **, *C)XXX215 MSGQ errno <error number> creating key=$<queue key>
The <error number> argument represents the UNIX error number, and <queue key> represents the message queue key. The task cannot create a message queue with the given key. A message queue with the same key might already exist. Look in the acecqid file found in the /CiscoBAMS/files directory for duplicate key definitions. Contact the Cisco Technical Assistance Center to correct the configuration if you find duplicates.

(*, **, *C)XXX216 MSGQ errno <error number> code <return code> on sendlocal() key=$<queue key>
The <error number> argument represents the UNIX error number, <return code> represents the return code of sendlocal(), and <queue key> represents the message queue key.

The task failed to send a message to a task on the same unit. The message queue with the given key might have been removed, or the system-imposed maximum number of pending messages might have been reached. From the UNIX prompt, issue the command **ipcs -qob**. Check if the queue with the given key still exists and if the total QNUM is less than 40.

(*, **, *C)XXX217 MSGQ errno <error number> code <return code> on sendremote() key=$<queue key>
The <error number> argument represents the UNIX error number, <return code> represents the return code of sendremote(), and <queue key> represents the message queue key.

The task failed to send a message to a task on a remote unit. The message queue for the task dealing with interunit communication might have been removed, or the system-imposed maximum number of pending messages might have been reached. From the UNIX prompt, issue the command **ipcs -qob** and check that the queue with the given key still exists and that the total QNUM is less than 40.

(*, **, *C)XXX218 MSGQ errno <error number> code <return code> on recvmsg() key=$<queue key>
The <error number> argument represents the UNIX error number, <return code> represents the return code of recvmsg(), and <queue key> represents the message queue key. The task failed to receive a message from a message queue with the given key. The message queue with the given key might have been removed, or the task might have been interrupted.

(*, **, *C)XXX219 SEMA errno <error number> code <return code> getting key for <semaphore name>
The <error number> argument represents the UNIX error number, <return code> represents the return code of the function that provides the mapping between semaphore names and semaphore keys, and <semaphore name> represents the semaphore name.

The task cannot get the semaphore key value for the given semaphore name. Either the acecsid file is not in the files directory, or the named semaphore is not defined in the acecsid file.

(*, **, *C)XXX220 SEMA errno <error number> creating key=$<semaphore>
The <error number> argument represents the UNIX error number, and <semaphore> represents the semaphore key. The task cannot create the semaphore with the given key. A semaphore with the same key might already exist.

(*, **, *C)XXX221 FILE errno <error number> opening <name> for <file>
The <error number> argument represents the UNIX error number, <name> represents the name of the file, and <file> represents the file open modifier (for example, create, read, or append). The task failed to open the named file using the indicated file open operation.

(*, **, *C)XXX222 FILE errno <error number> reading <filename> size <byte size>
The <error number> argument represents the UNIX error number, <filename> represents the filename or item being read, and <byte size> represents the total byte size to read. The task failed to read the requested number of bytes.

(*, **, *C)XXX223 FILE errno <error number> writing <byte size> size <filename>
The <error number> argument represents the UNIX error number, <byte size> represents the total byte size to write, and <filename> represents the filename or item being written. The task failed to write the requested number of bytes. Check if the disk is full and if the file exists.

`(*, **, *C)XXX224 FILE errno <error number> seeking <filename> offset <offset> whence <flag>`

The <error number> argument represents the UNIX error number, <filename> represents the name or the description of the input file, <offset> represents the requested offset, and <flag> represents the seek flag (0, 1, or 2). The task failed to move the pointer on the named file.

`(*, **, *C)XXX225 FILE fprintf() returned <return code>. Expecting <number>`

The <return code> argument represents the return code of fprintf(), and <number> represents the number of items requested. The task failed to write the requested number of items.

`(*, **, *C)XXX226 FILE fscanf() returned <return code>. Expecting <number>`

The <return code> argument represents the return code of fscanf(), and <number> represents the number of items requested. The task failed to read the requested number of items.

`(*, **, *C)XXX227 FILE errno <error number> linking <file> to <new file>`

The <error number> argument represents the UNIX error number, <file> represents the file to link from, and <new file> represents the name of the new file link. The task failed to create a link to a file. This might happen when the first file does not exist.

`(*, **, *C)XXX228 FILE errno <error number> unlinking <name>`

The <error number> argument represents the UNIX error number, and <name> represents the name of file to unlink. The task failed to remove a file link. Check if the named file exists. If it does, check the permissions in the file.

`(*, **, *C)XXX229 FILE errno <error number> renaming <name> to <new name>`

The <error number> argument represents the UNIX error number, <name> represents the original filename, and <new name> represents the new name. The task failed to rename a file.

`(*, **, *C)XXX230 DIR errno <error number> changing to <directory name>`

The <error number> argument represents the UNIX error number, and <directory name> represents the directory name. The task failed to set its working directory. Check if the directory exists and if the execute permissions are set.

`(*, **, *C)XXX231 DIR errno <error number> opening <directory name>`

The <error number> argument represents the UNIX error number, and <directory name> represents the directory name. The task failed to open the named directory. Check if the directory exists and if the read permissions are set.

`(*, **, *C)XXX232 DIR errno <error number> reading <directory name>`

The <error number> argument represents the UNIX error number, and <directory name> represents the directory name. The task failed to read an entry from the named directory. Check if the directory exists and if the read permissions are set.

`(*, **, *C)XXX233 SCKT errno <error number> creating <protocol> socket for <use>`

The <error number> argument represents the UNIX error number, <protocol> represents the protocol (TCP or UDP), and <use> represents the use of the socket. The task failed to create an end point for communication (a socket). This can happen if the task has opened too many sockets.

`(*, **, *C)XXX234 SCKT errno <error number> connecting to <name>/<string>`

The <error number> argument represents the UNIX error number, <name> represents the server name or address, and <string> represents the service name or port number (as a string).

The task failed to initiate a connection to a server process. Either the server name does not have an entry in the /etc/hosts file, or the service name is not in the /etc/services file. If an IP address and a port number are used instead of names and the message still shows up, contact the Cisco Technical Assistance Center.

`(*, **, *C)XXX235 SCKT errno <error number> binding socket <descriptor>`

The <error number> argument represents the UNIX error number, and <descriptor> represents the socket descriptor. The task failed to bind a name to a socket.

`(*, **, *C)XXX236 SCKT errno <error number> listening on socket <descriptor>, qlength=<queue length>`
The <error number> argument represents the UNIX error number, <descriptor> represents the socket descriptor, and <queue length> represents the requested queue length. The task failed to listen for connections on a socket.

`(*, **, *C)XXX237 SCKT errno <error number> accepting from socket=<id>`
The <error number> argument represents the UNIX error number, and <id> represents the socket ID. The task failed to accept a connection from a socket.

`(*, **, *C)XXX238 SCKT errno <error number> reading <data> size <bytes>`
The <error number> argument represents the UNIX error number, <data> represents the description of data to read, and <bytes> represents the number of bytes to read. The task encountered an error while reading data from a socket.

`(*, **, *C)XXX239 SCKT errno <error number> writing <data> size <bytes>`
The <error number> argument represents the UNIX error number, <data> represents the description of data to write, and <bytes> represents the number of bytes to write. The task encountered an error while writing data to a socket.

`(*, **, *C)XXX240 FILE errno <error number> getting file stat for <description>`
The <error number> argument represents the UNIX error number, and <description> represents the name or description of a file. The task failed to get file statistics. Check if the file exists.

`(*, **, *C)XXX241 MSGQ errno <error number> code <return code> deleting key=$<queue key>`
The <error number> argument represents the UNIX error number, <return code> represents the return code of the routine for deleting the message queue, and <queue key> represents the message queue key. The task failed to delete the message queue with the given key. The message queue might have been deleted manually or by another task.

`(*, **, *C)XXX242 (SHM errno <error number> code <return code> deleting key=$<queue key>`
The <error number> argument represents the UNIX error number, <return code> represents the return code of the routine for deleting the shared memory, and <queue key> represents the message queue key. The task failed to delete the shared memory with the given key. The message queue might have been deleted manually or by another task.

`(*, **, *C)XXX243 SHM errno <error code> attaching to <task name> shared memory`
The <error code> argument represents the UNIX error code, and <task name> represents the task name of the shared memory owner.

The task failed to attach to the shared memory owned by the given task. The shared memory might not have been created or might have been deleted by the owning process. Check if the shared memory owner is executing by issuing the UNIX command **ps -eaf | grep** <task>, where <task> is the owner of the shared memory.

`(*, **, *C)XXX244 SYS errno <error code> on UNIX system call <call>`
The <error code> represents the UNIX error code, and <call> represents the system call name. The task encountered an error during a UNIX system call.

Manager Task-Specific Messages

The Manager (MGR) task-specific error messages are numbered from 400 through 899 (not all numbers are used). The specific tasks use the same range of numbers, but the text of the message is different for each task.

`MGR500 Spawning <task>`
The <task> argument represents the task name. The MGR task manages the BAMS software. The MGR task starts and stops the other BAMS tasks. This informational message indicates that a task is being started.

`MGR501 Sending <signal number> to <task>`
The <signal number> argument represents the signal number, and <task> represents the task name. Some communication between UNIX tasks is by signal number. This message indicates that a signal has been sent to a UNIX task.

`MGR502 Attempting a restart`
This message indicates that the MGR task is attempting to restart the system. The startup messages that follow this message are displayed and are logged in the syslog file.

`MGR503 Keepalive internal passed`
Keep alive is a security feature that monitors system activity when you log on to the system as an administrator or operator. If control session activity is idle for more than a desired amount of time, the application logs the user out. This message indicates that the keep alive interval has passed.

`MGR504 Task <name> will be restarted`
The <name> argument represents the task name. The MGR is restarting the displayed task.

`MGR505 Invoking task <name>`
The <name> argument represents the task name. The MGR is starting the displayed task.

`* MGR600 Function <name> failed, code <rc>, errno:<UNIX error number>`
The <name> argument represents the function name, <rc> represents the return code, and <UNIX error number> represents the UNIX error number. The displayed function returned an unexpected return code.

`* MGR601 Too many options for <process>`
The <process> argument represents the process name. This message indicates that too many optional parameters are passed to a process listed in the proglist file. The proglist file contains the names of configuration files specific to each task.

Verify the parameters in the configuration file of the offending task. If the configuration file has been modified, remove some of the options within the configuration file or restore the original configuration file.

`** MGR700 Too few parameters <total>/<expected> on line <number> of proglist`
The <total> argument represents the number of parameters, <expected> represents the number of expected parameters, and <number> represents the line number in the file. The proglist file has an incorrect number of parameters. Contact the Cisco Technical Assistance Center.

`** MGR800 Failed spawning <name>`
The <name> argument represents the task name. This message indicates that the MGR task is unable to start the displayed task. This message can be caused by a combination of hardware and software problems.

Restart the application software and monitor the system log for the same occurrence. If the problem persists, restart the operating system. If the problem is still present, reinstall the application software.

`*C MGR801 Task <name> has died.`
The <name> argument represents the task name. This message indicates that the specified task has stopped responding to the system. The MGR task automatically attempts a restart of the indicated task. Verify the permissions and ownerships on the task that has died. Restart the application and monitor the system log to see if the error persists. If MGR is unable to restart the task, you might need to reinstall the software. To clear this alarm, you must provide a matching text message because multiple MGR801 alarms may be active (each with a different message).

Note This alarm is cleared after the error alarm condition is removed.

*C MGR802 Manager shutting down the application
This message indicates that a task has exited due to the receipt of the specified UNIX signal. The application has been terminated with a software interrupt. A signal 15 indicates that the system received the **stop** command. A signal 11 indicates that a segmentation violation occurred. The MGR task attempts a restart of the application for signals other than signal 15. If the MGR task is unable to restart the application, a UNIX shutdown might be necessary. This error can be associated with a hardware or a software failure.

*C MGR803 Received unexpected signal number <number>. Exiting
The <number> argument represents the signal number. This error message indicates that the task has exited due to the receipt of the specified UNIX signal. The MGR task should reinitialize the system automatically. Monitor the syslog file during the next hour. If the error message reoccurs, reinstall the software.

*C MGR804 Disk full. Must be cleaned up manually
The system has a full data directory and the Mass Storage Control task is unable to effect an automatic system cleanup. This type of error can occur if the disk thresholds are set too high.

Change to the /CiscoBAMS/data directory and remove older data files. The Mass Storage Control parameters should be verified and changed as necessary. For more information about setting up or editing MSC parameters, see the "Setting Up Disk Monitoring Thresholds" section on page 2-13.

Accumulation Task Messages

The Accumulation (ACC) task accumulates peg counts for measurements. During processing, this task writes messages to the syslog with an ACC task identifier (for example, ACC001).

ACC000 <name> version <number1> has started, pid <number2>
The <name> argument represents the task name. The <number1> argument represents the task version. The <number2> argument represents the process ID number. This message indicates that the ACC task has started. This is an informational message. No action is necessary.

ACC001 <name> terminating with code <errno>
The <name> argument represents the task name. The <errno> argument represents the UNIX error number set by the system function. This message indicates that the ACC task has been ended. See the error number to determine the reason for the error.

ACC002 <name> terminating with signal number <number>
The <name> argument represents the task name. The <number> argument represents the signal number. This message indicates that the ACC task has been ended. See the signal number to determine the reason for the error.

ACC003 <name> terminating normally
The <name> argument represents the task name. This message indicates that the ACC task is ending normally. This is an informational message. No action is necessary.

ACC004 Processing file <name>
The <name> argument represents the filename. This message indicates that the ACC task is processing the displayed file. This is an informational message. No action is necessary.

ACC005 File <name> processed
The <name> argument represents the filename. This message indicates that the ACC task processed the displayed file. This is an informational message. No action is necessary.

ACC010 <number> calls processed
The <number> argument represents the number of processed calls. This message indicates the number of call records that the ACC task has processed. This is an informational message. No action is necessary.

`* ACC019 No flat files to process. Retry in <number> seconds`

The <number> argument represents the number of seconds. This message indicates that the ACC task has found no data files to process and is pausing for a period of time before attempting to process additional files. If no *, **, or *C is displayed, this message is displayed as an informational message. No action is necessary.

`ACC022 Processed call set in <number> seconds`

The <number> argument represents the number of seconds. This message indicates the elapsed time to process a single call set (data file). This is an informational message. No action is necessary.

`ACC023 <name>: User time=<time1> sec, CPU time=<time2> sec`

The <name> argument represents the process name. The <time1> argument represents the CPU time used for executing instructions in user space. The <time2> argument represents the CPU time used for executing instructions in system space. This alarm message indicates the user and CPU time used for processing enclosure files. This is an informational message. No action is necessary.

`* ACC101 File error: <name> on <filename> (<errno>)`

The <name> argument represents the name of the failed system call. The <filename> argument represents the filename. The <errno> argument represents the UNIX error number set by the system function. This message indicates that the ACC task encountered an error when performing a system call operation on the specified file. See the UNIX error number for more information.

`*C ACC102 File I/O error: <name> on <filename>, requested: <number1>, got: <number2> (<errno>)`

The <name> argument represents the name of the failed system call. The <filename> argument represents the filename. The <number1> argument represents the number of requested bytes. The <number2> argument represents the number of received bytes. The <errno> argument represents the UNIX error number set by the system function. This message indicates that the ACC task encountered an error when performing a file I/O operation. Contact the Cisco Technical Assistance Center.

`W ACC204 Duplicate Trunk Group Key <trunk group> IC:<carrier id>`

The <trunk group> argument represents the trunk group. The <carrier id> argument represents the carrier identifier. An error occurred when the trunk group key was added. Contact the Cisco Technical Assistance Center.

`** ACC206 Undefined environment variable <varname>`

The <varname> argument represents the variable name. This message indicates that an expected environment variable has not been defined. Try logging out and logging back in. If this problem persists, verify that the specified environment variable is defined in /CiscoBAMS/bin/sym_defs. If it is missing, contact the Cisco Technical Assistance Center for directions on how to proceed.

`** ACC207 Function <name> failed <errno>`

The <name> argument represents the name of the failed function. The <errno> argument represents the UNIX error number set by the system function. This message indicates that an unexpected function failed. Contact the Cisco Technical Assistance Center.

`*C ACC210 The EOF marker missing for the file <filename>`

The <filename> argument represents the filename. This message indicates a missing end-of-file marker in the displayed file. Contact the Cisco Technical Assistance Center.

`* ACC221 Old call set received <id> current <interval string>`

The <id> argument represents a new call set, and <interval string> represents the current call set. This message appears when a call set older than the current one is received. Contact the Cisco Technical Assistance Center.

`* ACC222 Missing call set expected <next interval string> received <call set id>`

The <next interval string> argument represents a new call set, and <call set id> represents the current call set. This message appears when a time gap has been detected in the ACC input stream. Contact the Cisco Technical Assistance Center.

`* ACC223 Can't load interval value from check point file`
This message appears when an error is detected during the reading of the checkpoint file. Contact the Cisco Technical Assistance Center.

`* ACC224 Memory allocation failed requested by <function name>`
The <function name> argument represents the name of the function requesting memory. Contact the Cisco Technical Assistance Center. This message appears when memory is allocated and none is returned.

`ACC225 No Checkpoint file to load`
This message appears when the ACC task starts and no checkpoint file is found. This is an informational message. No action is necessary.

`ACC226 Check point loaded, last interval was <current interval string>`
The <current interval string> argument represents the current interval string loaded from the checkpoint. This message appears when the ACC task starts and the checkpoint file is loaded. This is an informational message. No action is necessary.

`* ACC227 <count direction> <description>, TG <trunk group prefix> <trunk group number> <counter value> <condition> <threshold value> <timestamp>`
The <count direction> represents the direction of the count, <description> is a description of the count, <trunk group prefix> represents the trunk group prefix, <trunk group> represents the trunk group, <counter value> represents the counter value, <condition> represents the threshold crossing condition, <threshold value> represents the value of the threshold, and <timestamp> represents the timestamp (that is, `@ YYYYMMDDHHMMSS UTC`). This message appears when a counter passes a user-determined threshold test or clears. To clear this alarm, you must provide a matching text message, because multiple ACC227 alarms might be active (each with a different message).

Note This alarm is cleared after the alarm condition is removed.

`ACC228 <count direction> <description>, TG <trunk group number> contains an invalid condition`
The <count direction> represents the direction of the count, <description> is a description of the count, and <trunk group number> represents the trunk group number. This message appears when a threshold is tested and the condition (<=>) is an unknown test. This is an informational message. No further action is necessary.

`* ACC230 CTL file error: <string>`
The <string> argument represents a string from ai_strerror(). This message displays errors from calls to the Web Application Program Interface (API). Contact the Cisco Technical Assistance Center.

`* ACC231 Threshold file error <key> <threshold name> <value> <text>`
The <key> argument represents the id/tag, <threshold name> represents the name of the threshold (for example, “ANSWERED”), <value> represents the value in error, and <text> represents a description of the error condition. This message appears when the threshold file contains an incorrect value. Contact the Cisco Technical Assistance Center.

`* ACC232 Multiple Global thresholds found`
On each new call set, ACC reloads the Threshold Crossing Alarms table if the file contains more than one global threshold record that this alarm displays. Contact the Cisco Technical Assistance Center.

`* ACC233 Out of table Space`
If ACC attempts to insert a new node in a tree and no space is available, this alarm is displayed. Contact the Cisco Technical Assistance Center.

`* ACC234 Known node can't be found deleting`
If ACC attempts to delete a node in a tree and the delete returns null (node not found), this alarm is displayed. Contact the Cisco Technical Assistance Center.

`* ACC235 CTL file <name> value error: <value>`
The <name> argument represents the name of the CTL file and the value represents the measurement value. This message displays when the value of the measurement field in the control file is not 0 or 1. Check the configuration file using MML and fix the value.

`* ACC236 Interval minutes must be 5, 10, 15, 20 or 30`
This message displays when the ACC task reads an invalid value from the NODEPARMS configuration file for the interval minutes. Check the NODEPARMS configuration file using MML and fix the value.

`* ACC237 Number of Ckts not populated for TD <trunk group>`
The <trunk group> number represents the trunk group number. This message displays if the number of circuits for a trunk group has not been defined or has been set to zero. An invalid zero value leads to invalid statistics. If this alarm is generated when the number of circuits is zero, the percentage measurements are reported as 100 percent. At the node level, modify the Trunk Group table (TRUNKGRP tag ID) for the specified trunk group and then assign a non-zero value to the circuits parameter.

Augmentation BAF Task Messages

The Augmentation BAF (AGB) task controls the augmentation of data through the use of lookup tables. During processing, this task writes messages to the syslog with an AGB (Augmented BAF) task identifier (for example, AGB001).

`AGB000 <task> version <version> has started <pid>`
The <task> argument represent the task name, <version> represents the task version, and <pid> represents the process identifier. This message indicates that the AUG task has started. This is an informational message. No action is necessary.

`AGB001 <task> terminating with <code> <pid>`
The <task> argument represents the task name, <code> represents UNIX error code, and <pid> represents the process identifier. This message indicates that the AUG task has ended abnormally. See the UNIX error code for an explanation of the problem.

`AGB002 <name> terminating with signal <number1>, pid <number2>`
The <name> argument represents the task name, <number1> represents the UNIX signal number, and <number2> represents the UNIX process identifier. This message indicates that the displayed signal ended the task. See the UNIX signal number for an explanation of the problem.

`AGB003 <name> terminating normally, pid <number>`
The <name> argument identifies the task name, and <number> identifies the process ID number of the task. This message indicates that the task terminated normally. This is an informational message. No action is necessary.

`AGB004 Processing file <name>`
The <name> argument represents the filename. This message indicates that the AUG task is processing the displayed file. This is an informational message. No action is necessary.

`AGB009 Received system signal <signal name>.`
The <signal name> argument represents the name of the UNIX signal. The AUG task received a system signal. This is an informational message. No action is necessary.

`AGB010 Processed <filename>, total: <number> good: <number> sunerr: <number> semerr <number> lk uperr <number>`
The <filename> argument represents the name of the processed file, and <number> represents the total number of call records, the number of good call records, the number of call records with syntax errors, the number of call records with semantic errors, and the number of call records with error lookups, respectively. This message provides summary information of call records being processed. This is an informational message. No action is necessary.

AGB019 No flat files to process. Retry in <number> seconds.
The <number> argument represents the time delay in seconds. There are no data files to process. The task waits and retries in the specified number of seconds. This is an informational message. No action is necessary.

AGB024 Creating <name> table
The <name> argument represents the lookup table name (that is, NPANXX table, Rating Exception table, Rating Type table, or Trunk Group table). The named lookup table is created and loaded into memory. This is an informational message. No action is necessary.

AGB025 Refreshing <name> table.
The <name> argument represents the lookup table name. The lookup table is reloaded. This is an informational message. No action is necessary.

* AGB100 System call failure: <name> <errno>
The <name> argument represents the name of a failed system call, and <errno> represents the UNIX error number. This message indicates that the AUG task encountered an error when requesting the displayed system call. Look up the UNIX error number. Contact the Cisco Technical Assistance Center.

* AGB101 File error: <call> on <file> (<code>)
The <call> argument represents the name of the failed system call, <file> represents the name of the file being operated on, and <code> is the UNIX error number. This message indicates an error during the performing of the displayed system call operation. Look up the UNIX error number. A software-related problem is indicated. Contact the Cisco Technical Assistance Center.

*C AGB102 File I/O error: <call> on <file>, requested <number1>, got <number2>
The <call> argument represents the failed system call, <file> represents the name of the file being operated on, <number1> represents the number of bytes requested, and <number2> represents the number of bytes received. This message indicates an error during performance of the displayed I/O file operation. Check if the disk is full or not. If the disk is not full, a software-related problem is indicated. Contact the Cisco Technical Assistance Center.

* AGB104 Cannot find file <name> to process
The <name> argument represents the filename. This message indicates that the task is expected to reprocess the named file but did not find it. Contact the Cisco Technical Assistance Center.

* AGB201 Undefined environment variable <name>
The <name> argument represents the name of the undefined environment variable. This message indicates that an environment variable is not defined. Check the sym_defs file. Contact the Cisco Technical Assistance Center.

* AGB202 <index> - Invalid <field>: <value>
The <index> argument represents the call record index, <field> represents the field name in the call record, and <value> indicates the value of the field. This message indicates that the AUG task detected an invalid field value in the call record. The input data has an invalid call reference ID. Check the Cisco MGC raw data, and make sure the call reference ID has the correct format. Contact the Cisco Technical Assistance Center.

* AGB203 <index> - fail <function> lookup of <field>: <value>
The <index> argument represents the call record index, <function> represents the function name of the table lookup, <field> represents the field name in the call record, and <value> represents the value of the field being looked up. This message indicates a failed table lookup. Make sure the specific value exists in the table. Contact the Cisco Technical Assistance Center.

* AGB210 CTL file: <name> error: <ai string error>
The <name> argument represents the function name, and <ai string error> represents the error string. This message indicates an error returned by a control file function. Make sure the CTL file exists in the specified directory. Check the existence of the CTL file (*.CTL) and verify the structure of the control file (*.CFDT). If the CFDT file is corrupt, copy the default CFDT file. Contact the Cisco Technical Assistance Center.

`* AGB211 CTL file BAF value error: <value>`
The <value> argument represents a bad value. This message indicates that the content of the BAF field in the control file is not 0 or 1. Verify sysparm configurations. Make sure the BAF output is set to 0 (disabled) or 1 (enabled). Contact the Cisco Technical Assistance Center.

`* AGB232 #<index> Fail <function> lookup of<field1>:<value1> <field2>:<value2> <field3>:<value3>`
The <index> argument represents the call record index, <function> represents the function name of the table lookup, <field1> represents the first field name in the call record, <value1> represents the value of the first field, <field 2> represents the second field name in the call record, <value2> represents the value of the second field, <field3> represents the third field name in the call record, and <value3> represents the value of the third field. This message indicates a failed table lookup at the named locations. The system tried to match three field values to the lookup table. Verify the table configuration file, and make sure the field values exist in that configuration file. Contact the Cisco Technical Assistance Center.

`* AGB233 #<index> Fail <function> lookup of <field>:<value>`
The <index> argument represents the call record index, <function> represents the function name, <field> represents the field name of the call record, and <value> represents the value of the field being looked up. This message indicates a failed table lookup of this field. Verify that the input raw file used only one field value to match to the lookup table. Contact the Cisco Technical Assistance Center.

`* AGB234 #<index> - Invalid block: <field name>`
The <index> argument represents the record index, and <field name> represents field names and values. This message indicates that an invalid block format was encountered in the reading of blocks from the table file. For example, the system expected a six-digit value for NPANXX, but it received only five digits. Verify the field value in the table. Make sure you enter the required value. For field names and descriptions, see the "Tag IDs and Field Names" section on page 5-1.

`* AGB235 #<number> records failed trunk group lookup for file cor_agcdr_<timestamp>_<sequence number>.bin`
The <number> argument represents the number of records that failed the trunk group lookup, the <timestamp> argument represents the date and time, and the <sequence number> argument represents the file sequence number. This minor alarm indicates that the Trunk Group table is not correctly or fully populated. This error precludes these records from being processed by the measurements function. Fix the Trunk Group table. Refer to the AUG*log files in the opt/CiscoBAMS/files directory. This file specifies which trunk group should be modified so that these configuration failures do not occur.

`AGB236 <number> record(s) default to interlata due to absent NPANXX for file`
The <number> argument represents the record number that defaulted to the INTERLATA rating type due to this NPANXX combination not being in the NPANXX tables. This message indicates that the AUG task has started. This is an informational message. No action is necessary.

`AGB237 <number> record(s) did not have called and or calling number for files <name>.`
The <number> argument represents the number of records and the <name argument> represents the filenames of no-data files encountered during processing. This is an informational message. No action is required.

ASCII Conversion Task Messages

The ASCII Conversion (ASC) task outputs Cisco MGC End of Call 1110 ASCII records. During processing, this task writes messages to the syslog with an ASC task identifier (for example, ASC001).

`ASC000 <task name> version <version> has started pid <process identifier>`
The <task name> argument represents the task name, <version> represents the version, and <process identifier> is the UNIX process identifier. This alarm defines what version of the task is running. This is an informational message. No action is required.

`ASC001 <task name> terminating with code <return error code>, pid <process id>`
The <task name> argument represents the task name, <return error code> represents the error code returned, and <process id> represents the UNIX process identifier. The named task has been terminated abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

`ASC002 <task name> terminating with signal <signal number>, pid <process id>`
The <task name> argument represents the task name, <signal number> represents the UNIX signal number, and <process id> is the UNIX process identifier. The named task has been ended by a UNIX signal. This is an informational message. No action is necessary.

`ASC003 <task name> terminating normally, pid <process id>`
The <task name> argument represents the task name, and <process id> is the UNIX process identifier. The ASC task has terminated normally. This is an informational alarm that appears only in test mode.

`ASC004 No file to be processed. Retries in <number> seconds`
The <number> argument represents the number of seconds the system waits before reattempting to process files. The task sleeps for the number of seconds displayed before accessing more files. This is an informational message. No action is necessary.

`* ASC101 File error: <data format type> on <system call> <errno>`
The <data format type> represents the type of data format; <system call> represents the name of the failed system call or the name of the file being operated on; and <errno> represents the UNIX error number. An error was encountered during the performance of a system call operation. Look up the UNIX error number in the listing at the end of this appendix and contact the Cisco Technical Assistance Center.

`** ASC200 Environment variable <name> not found`
The <name> argument represents the name of the missing environment variable, for example CONFIG_DIR, FILES_DIR, dir_in or dir_out. The displayed environment variable was not found. The symbol definition file (sym_defs) is missing a variable, or the variable must be set before the task is run. Contact the Cisco Technical Assistance Center.

`(*,**) ASC201 Function <name> failed, <errno>`
The <name> argument represents the calling function name, and <errno> represents the UNIX error number. This message indicates failure of the displayed function call, that is, Major Events for open_file, stat (sysParm.CTL), ai_open ctlfile, minor events for ai_get_max_num_of_recs(), ai_load_rec(), or ai_close_file. Examine the error number. Contact the Cisco Technical Assistance Center.

`(*,**) ASC201 Function <calling function> failed, (<error number>)`
The <calling function> represents the calling function, and the <error number> argument represents the UNIX error number. The system encountered a failure trying to make a function call. Examine the system error. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`** ASC202 Invalid AMADEF format, read <error string>, expected <expected string>`
The <error string> argument represents the error string found, and the <expected string> argument represents the string that was expected. The system encountered a format problem while trying to read the AMADEF file. The AMADEF file needs to be fixed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`* ASC205 Unsupported field <field name>, conversion not defined`
The <field name> argument represents the name of the field. The system encountered an AMA field that cannot be processed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`ASC207 Processed <filename>, converted <num recs>, output = <suffix>`
The <filename> argument represents the input filename, <num recs> represents the number of records converted, and <suffix> represents the output file suffix. This message displays processing status information. This is an informational message. No action is necessary.

(*,**) ASC208 <data type> code <structure code> not found, cannot process record
The <data type> argument represents the type of data and <structure code> represents the structure code or call type. The system is unable to process a record due to missing code and structure code information. Check the AMADEF file. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

** ASC209 Error <return value> reading file: CALLSTRUCT_MAP
The <return value> represents the value returned. The system encountered a format error reading the callstruct file. Check the CALLSTRUCT_MAP file. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

1110 Binary Task Messages

The 1110 Binary (BIN) task correlates 1010, 1030, 1040, and 1060 CDBs from the Cisco MGC, and generates 1110, and 1060 CDBs. During processing, this task writes messages to the syslog with a BIN task identifier (for example, BIN001).

BIN000 <task> version <version number> has started, pid <process ID>.
The <task> argument represents the task name, <version number> represents the task version number, and <process ID> represents the UNIX process identifier. This message identifies the version of the task that is running. This is an informational message. No action is necessary.

BIN001 <task> terminating with code <return code>, pid <process ID>.
The <task> argument represents the task name, <return code> identifies the terminating signal number encountered by the system, and <process ID> represents the UNIX process identifier. The named task ended abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

BIN002 <task> terminating with signal <number>, pid <process ID>.
The <task> argument represents the task name, <number> represents the UNIX signal number, and <process ID> represents the UNIX process identifier. The named task was ended by a UNIX signal.

BIN003 <task> terminating normally, pid <process ID>.
The <task> argument represents the task name, and <process ID> represents the UNIX process identifier. The named task ended normally. This is an informational message. No action is necessary.

BIN004 Processing file <filename>
The <filename> represents the name of the file being processed. The task has started processing the named file. This is an informational message. No action is necessary.

BIN009 Received system signal <signal>
The <signal> argument represents the received signal number. The BIN task received a system signal. This is an informational message. No action is necessary.

BIN010 Processed <filename>, IN(total: <number1>) OUT(1060: <number2> 1110: <number3>)
The <filename> argument represents the processed filename, <number1> represents the total number of processed call records, <number2> represents the number of 1060 call records, and <number3> represents the number of 1110 call records. This message displays call process summary information. This is an informational message. No action is necessary.

BIN011 Processed <filename>, Skipped CDEs: <number1> Unknown CDEs: <number2>
The <filename> argument represents the processed filename, <number1> represents the number of skipped CDEs, and <number2> represents the number of unknown CDEs. This message indicates the number of CDEs that were skipped or unknown in a processed file. This is an informational message. No action is necessary.

`BIN012 Output file <filename>`
The <filename> argument represents the processed filename. This message indicates the name of the output file. This is an informational message. No action is necessary.

`BIN019 No flat files to process. Retry in <number> seconds.`
The <number> argument represents the number of seconds. This message indicates that no data exists for processing. The BIN task retries in the displayed number of seconds. This is an informational message. No action is necessary.

`BIN020 CDB tag <number>, CDB is skipped.`
The <number> argument represents the CDB number. This CDB is skipped when the CDB tag is found in the Skip CDB table. This is an informational message. No action is necessary.

`BIN021 CDE tag <number>, CDB is skipped`
The <number> argument represents the CDE number. This CDE is skipped when the CDE tag is found in the Skip CDE table. This is an informational message. No action is necessary.

`BIN024 Creating <name> table`
The <name> argument represents the lookup table name (that is, NPANXX table, Rating Exception table, Rating Type table, or Trunk Group table). The named lookup table is created and loaded into memory. This is an informational message. No action is necessary.

`BIN025 Refreshing <name> table`
The <name> argument represents the lookup table name. The lookup table is reloaded. This is an informational message. No action is necessary.

`BIN026 Generated CDB <type> length is <length>.`
The <type> argument represents the CDB type. The <length> argument represents the length of the CDB. This is an informational message. No action is necessary.

`* BIN100 System call failure: <call name> (<error number>)`
The <call name> represents the name of the failed system call, and <error number> represents the UNIX error number. An error was encountered when this system call was requested. Look up the UNIX error number for more information. Contact the Cisco Technical Assistance Center.

`* BIN101 File error: <call> on <file> (<code>)`
The <call> argument represents the name of the failed system call, <file> represents the filename, and <code> represents the UNIX error number. This message indicates that the BIN task encountered an error when performing the displayed system call operation. Look up the error number. Contact the Cisco Technical Assistance Center.

`*C BIN102 File I/O error: <call> on <file>, requested: <number1>, got: <number2> (code)`
The <call> argument represents the name of the failed I/O system call, <file> represents the name of the file being operated on, <number1> represents the number of bytes requested, and <number2> represents the number of bytes received, and <code> represents the UNIX error number. The BIN task encountered an error when performing the displayed I/O file operation. Contact the Cisco Technical Assistance Center.

`* BIN201 Undefined environment variable <name>`
The <name> represents the name of the environment variable. This message indicates that the displayed environment variable is not defined. Contact the Cisco Technical Assistance Center.

`* BIN202 Input file does not start with header CDB`
This message indicates that the BIN task detected an invalid field value in the call record. The BIN task placed the input file in the error directory. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* BIN203 Error reading TAG/LEN of TLV errno=<error number>`
The <error number> argument represents the UNIX error number. An error occurred during the reading of the TLV (T = Tag field code, L = Length of the field value, V = field Value). The BIN task stops processing this file, renames it as an error file, and begins processing the next input file. Look up the error. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* BIN204 Input file is not ended by footer CDB`
The system encountered the displayed error. The BIN task placed the input file in the error directory. Contact the Cisco Technical Assistance Center.

`* BIN207 #<index> - <name>: <value> too long, <size> expected.`
The <index> argument represents the call record index, the <name> argument represents the field name, the <value> argument represents the field value, and the <size> argument represents the expected field size. This message indicates that the field string exceeded the expected size. Contact the Cisco Technical Assistance Center.

`* BIN210 CTL file: <name>: error: <ai string error>.`
The <name> argument represents the control filename, and the <ai string error> argument represents the displayed error string. This message indicates the error returned by a control file function. Contact the Cisco Technical Assistance Center.

`* BIN211 CTL file lookup value error: <value>`
The <value> argument represents the measurement value. This message displays when the value of the measurement field in the control file is not 0 or 1. Check the configuration file using MML and fix the value.

`* BIN222 Invalid TLV len:<length> at <position>th chars, CDB len:<size>`
The <length> argument represents the incorrect TLV length, <position> represents the TLV position in the CDB, and <size> represents the size of the CDB. The system read an invalid TLV length. The BIN task stops further processing of the input file and renames the input file by prepending “err_” to the input filename. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* BIN223 #<format> - Get <number1> byte record. <number2> bytes record expected.`
The <format> argument represents the data format, <number1> represents the size of the call record read, and <number2> represents the size of the call record expected. This message indicates the processed record size.

`* BIN224 Process all CDEs: Error in parsing record.`
This message indicates that the BIN task encountered an error during the parsing of this record. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* BIN225 Dump_cde: Unrecognized tag <tag number>`
The <tag number> represents the number of the tag. This message indicates that the BIN task encountered an unrecognized tag. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* BIN226 Incorrect field length <field value length>.`
The <field value length> argument represents the length of the field value.This message indicates that the BIN task encountered an incorrect field length. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`BIN232 <index> - Fail <function> lookup of <field1>: <value1> and <field2>: <value2>`
The <index> argument represents the index of the call record, the <function> argument represents the function name of the table lookup, the <field1> argument represents the first field name in the call record, the <value1> argument represents the first value of the field being looked up, the <field2> argument represents the second field name in the call record, and the <value2> argument represents the second value of the field being looked up. This message indicates that the task failed to look up the indicated fields.

`BIN233 <index> - Fail <function> lookup of <field>: <value>`
The <index> argument represents the index of the call record, the <function> argument represents the function name of the table lookup, the <field> argument represents the first field name in the call record, and the <value> argument represents the first value of the field being looked up. This message indicates that the task failed to look up the indicated fields.

BIN234 #<record index> - Invalid record <field names/values>
The <record index> argument represents the record index, and the <field names/values> argument represents the field names and values. The BIN task encountered an invalid block format while reading blocks from a control file. The message shows which fields are invalid. Use MML to correct the values.

BIN235 <number> record(s) failed <trunk/sigpath> lookup for file <name>
The <number> argument represents the number of records, the <trunk/sigpath> argument represents the trunk group or sigpath, and the <name> argument represents the filename. This message shows the number of lookup failures per file. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/BIN_cdr.log and search for message BIN233. Correct the appropriate record in the Trunk Group table or in the Nailed Connection table.

BIN236 Setting BIN processing prefix to <prefix>
The <prefix> argument represents the file prefix. The BIN task processes files whose prefixes match the file-prefix1 parameter in the Poll table. If this parameter is not populated, the file-prefix2 parameter is used. If both parameters are empty, the default prefix is cdr. Whenever the Poll table is modified, this message displays the value of the file prefix that BIN uses.

Correlation Task Messages

The Correlation (COR) task correlates call record data collected from the Cisco MGC. During processing, this task writes messages to the syslog with a COR task identifier (for example, COR001).

COR000 <task> version <version number> has started pid <process ID>.
The <task> argument represents the version task, <version number> represents the task version number, and <process ID> represents the UNIX process identifier. This message defines the version of the task that is running. This is an informational message. No action is necessary.

COR001 <task> terminating with code <return code>, pid <process ID>.
The <task> argument represents the task name, <return code> represents the return code, and <process ID> represents the UNIX process identifier. The named task has ended abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

COR002 <task> terminating with signal <number>, pid <process ID>.
The <task> argument represents the task name, <signal number> represents the signal number received, and <process ID> is the UNIX process identifier. The named task was ended by a UNIX signal. This is an informational message. No action is necessary.

COR003 <task> terminating normally, pid <process ID>.
The <task> argument represents the task, and <process ID> represents the UNIX process identifier. The named task has ended normally. This is an informational message. No action is necessary.

COR004 No file to be processed. Retries in <number> seconds.
The <number> argument represents the interval, in seconds, between retries—that is, the task sleeps for <number> seconds before accessing more files. This is an informational message. No action is necessary.

COR005 Start processing file <name>.
The <name> argument represents the filename. This message indicates that a file is processing. This is an informational message. No action is necessary.

COR006 Processed file <name>: Output-> Complete:<number1> Incomplete:<number2> Ldur:<number3> Hold-> Incomplete:<number4>.
The <name> argument represents the name of file that has finished processing, <number1> represents the number of complete calls output, <number2> represents the number of incomplete calls output, <number3> represents the number of long duration calls output, and <number 4> represents the number of incomplete calls. This message indicates that the displayed file has been processed. This is an informational message. No action is required.

`* COR100 System call failure: <call name> (<error number>)`
The <call name> represents the name of the failed system call, and <error number> represents the UNIX error number. An error was encountered when this system call was requested. Look up the UNIX error number for more information. Contact the Cisco Technical Assistance Center.

`(*,**) COR101 File error: <system call> on <filename> (<error number>)`
The <system call> argument represents the name of the failed system call, <filename> represents the name of the file being operated on, and <error number> represents the UNIX error number. An error was encountered during performance of a system call operation on the specified file. See the UNIX error number for more information. Contact the Cisco Technical Assistance Center.

`** COR200 Environment variable <variable name> not found.`
The <variable name> argument represents the name of the environment variable that was not found. This message indicates that the displayed environment variable was not found. Verify that the variable is defined in the sym_defs file, or make sure that it is set before the task is run. Contact the Cisco Technical Assistance Center.

`(*,**) COR201 Function <function name>() failed, (<error number>)`
The <function name> argument represents the calling function name, and <error number> represents the UNIX error number. This message indicates the failure of a function call. See the UNIX error number for more information. Contact the Cisco Technical Assistance Center.

`** COR202 Number of CDR memory extensions exceeded <max number>`
The <max number> argument represents the maximum number of memory extensions. The maximum number of CDR memory extensions is set too small. Increase the maximum number of CDR memory extensions in the cor_cfg file in the opt/CiscoBAMS/files directory. In that file, the -m value indicates the size of the extents and measures the number of CDRs to save into the memory, and the -e value indicates the number of extents. Contact the Cisco Technical Assistance Center.

`** COR203 Read incomplete block size <filename>, <filename> expected.`
The <filename> argument represents the name of the file found and the name of the file expected, respectively. The input file generated by the previous task has an I/O error (either a bad disk block or the disk space is full). The input file is incomplete. COR expected multiple records for each input file. The previous task wrote incomplete records, for example, because of a disk space problem. Contact the Cisco Technical Assistance Center.

`** COR204 Invalid CDB type: <name>`
The <name> argument represents the filename found. For more information, check the error file called err_<raw_file_name>. This message indicates that an invalid CDB type was encountered. The COR task writes invalid CDBs to that file. Check the Cisco MGC XECFGPARM.dat file and make sure the MGC CDB type is configured as specified. Contact the Cisco Technical Assistance Center.

`** COR205 Exhausted all possible sequence numbers: <filename>`
The <file name> argument represents the name of the file. This message indicates that the number of input files in a single interval exceeded 99,999. Contact the Cisco Technical Assistance Center.

`** COR206 Multiple boundary crossings in a single file: <filename>`
The <filename> argument represents the name of the input file. This message is generated if there are more than two boundary crossings in a data file. Contact the Cisco Technical Assistance Center.

`* COR207 Errors generating boundary crossing file: <filename>`
The <file name> argument represents the name of the file. This message indicates that an error was encountered when the boundary crossing file was generated. Contact the Cisco Technical Assistance Center.

`COR208 <string> stat file: <filename>`
The <string> argument represents an action, such as “Creating” or “Closing,” and <filename> represents the name of the file. This message is generated for the displayed stat file. This is an informational message. No action is necessary.

`* COR210 CTL file: <function name> error: <ai string error>`
The <function name> argument represents the name of the function, and <ai string error> represents the error code. This message indicates an error returned by a control file function. Contact the Cisco Technical Assistance Center.

`* COR211 CTL file measurement value error: <value>`
The <value> argument represents the incorrect value. This message indicates that the content of the measurement field in the control file is not 0 or 1. Contact the Cisco Technical Assistance Center.

`* COR223 Can't load interval value from check point file`
This message indicates an error in the reading of the check point record from the check point file. Delete the check point file, COR_CHECKPT, located in the opt/CiscoBAMS/files directory. Contact the Cisco Technical Assistance Center.

`COR225 No check point file to load`
This informational message displays when the COR task starts and there is no check point file to load.

`COR226 Check point loaded, last interval was <string>, next seq no <number>`
The <string> argument represents the current interval string loaded from the check point, and the <number> argument represents the current interval sequence number. This informational message displays when the COR task starts and the check point file is loaded.

`* COR227 Interval minutes must be 5, 10, 15, 20 or 30`
This message displays when the COR task reads an invalid value from the NODEPARMS configuration file for the interval-minutes. Check the NODEPARMS configuration file using MML and fix the value.

`COR228 MAINT_UNBLOCK received for TG without blocks`
This informational message displays when a MAINT_UNBLOCK message is received for a trunk group that has no blocked trunks. The cdb is ignored.

`COR229 MAINT CDB received with an invalid maint_type`
This informational message displays when a MAINT cdb is received for a trunk group without a valid maint_type. The cdb is ignored.

`COR230: MAINT CDB received with an invalid trk grp`
This informational message indicates that a 1071 CDB was received. This CDB contains a maintenance trunk group ID. However, BAMS cannot find information about this trunk group because it is not provisioned in either the TRUNKGRP or SIGPATH table, depending on the nailed-cfg value of the NODEPARMS table.

Check Remote Down Task Messages

The Check Remote Down (CRD) task monitors communications between redundant BAMS units, and verifies whether or not the remote unit is down. During processing, this task writes messages to the syslog with a CRD task identifier (for example, CRD001).

`* CRD500 Cannot attach to shared memory errno: <error number>`
The <error number> argument represents the UNIX error number. The system is unable to access the shared memory segment and displays an error message. Look up the UNIX error for further explanation. Contact the Cisco Technical Assistance Center.

`* CRD501 Get Message Queue ID for <message queue name> error, returned <return number>`
The <message queue name> represents the name of the message queue, and <return number> represents the number of the returned function. The system is unable to get the message from the sender/receiver message queue and displays the sender/receiver message queue ID. Use the ipcs utility to check the message queue status. Contact the Cisco Technical Assistance Center.

`* CRD502 Redundant polling unit is down.`
The redundant polling unit is down. Check the remote BAMS unit. If it is down, contact the Cisco Technical Assistance Center.

Note This alarm is cleared when the alarm condition is removed.

`* CRD503 <operation> failed, returned <return number>, errno <error number>`
The <operation> argument represents the failed operation, <return number> represents the number of the returned function, and <error number> represents the UNIX error number. The system is unable to send local, send remote, or receive messages and displays an error category and code. Look up the UNIX error. Contact the Cisco Technical Assistance Center.

`* CRD504 Receive message time out`
The system experienced a receive message timeout. Check the network connection to the remote BAMS unit. Contact the Cisco Technical Assistance Center.

`CRD505 Redundant polling unit is up`
The redundant polling unit is back on line. This message is displayed for informational purposes. No action is necessary.

CDR to BAF Conversion Task Messages

The CDR to BAF Conversion (CTB) task generates BAF output. During processing, this task writes messages to the syslog with a CTB task identifier (for example, CTB001).

`CTB000 <version> <version number> has started, pid <process ID>`
The <version> argument represents the task version, the <version number> represents the task version number, and <process ID> represents the UNIX process identifier. The system displays a message that the named task has started. This message is displayed for informational purposes. No action is necessary.

`CTB001 <task name> terminating with code <return error code>, pid <process id>`
The <task name> argument represents the name of the task, <return error code> represents the returned error code, and <process ID> represents the UNIX process identifier. The named task has terminated abnormally. The code identifies the error encountered by the system. Look up the error code. Contact the Cisco Technical Assistance Center.

`CTB002 <task name> terminating with signal <signal number>, <process ID>`
The <task name> argument represents the name of the task, <signal number> represents the UNIX signal number, and <process ID> represents the UNIX process identifier. The named task was ended by a UNIX signal. This is an informational message. No action is necessary. See the "Signal Numbers" section on page A-68.

`CTB003 <name> terminating normally, pid <number>`
The <name> argument represents the task name, and <number> represents the process ID number of the task. This message indicates that the task terminated normally. This is an informational message. No action is necessary.

`CTB004 No file to be processed. Retries in <n> seconds`
The <n> argument represents the number of seconds during which the task sleeps before accessing more files.

`* CTB101 File error <data type> on <system call> (<error number>)`
The <data type> argument represents the type of data format, <system call> represents the failed system call, and <error number> represents the UNIX error number. The system encountered an error when performing a system call operation on the specified file. Look up the error number. Contact the Cisco Technical Assistance Center.

`** CTB200 Environment variable <variable name> not found`
The <variable name> argument represents the name of the environment variable not found. The system encountered a missing or undefined variable. Either the symbol definitions are missing, or the variable must be set before the task is run. Contact the Cisco Technical Assistance Center.

`(*,**) CTB201 Function <calling function> failed, (<error number>)`
The <calling function> argument represents the name of the calling function, and <error number> represents the UNIX error number. The system encountered a failure trying to make a function call. Look up the error number. Contact the Cisco Technical Assistance Center.

`** CTB202 Invalid AMADEF format, read <error string>, expected <expected string>`
The <error string> argument represents the error string found, and <expected string> represents the string expected. The system encountered a format problem while trying to read the AMADEF file. The AMADEF file needs to be fixed. Contact the Cisco Technical Assistance Center.

`* CTB205 Unsupported field <field name>, conversion not defined`
The <field name> argument represents the name of the field that is not supported. The system encountered an AMA field that cannot be processed. Contact the Cisco Technical Assistance Center.

`CTB207 Processed <filename>, converted <record number> records, output <output filename>`
The <filename> argument represents the name of the processed file, <record number> represents the record number, and <output filename> represents the output filename. The system displays the following process information: name of processed file, number of records converted, and output filename. This is an informational message. No action is necessary.

`(*,**) CTB208 <data type> code <structure code> not found, cannot process record`
The <data type> argument represents the data type, and <structure code> represents the structure code. The system is unable to process a record due to missing code and structure code information. Check the AMADEF file. A major alarm is generated if an incomplete, on-going call is rejected by the Correlation Task (COR) because an on-going record was not received within 12 hours (based on the cdb_creation_time of the latest data). This record does not have an associated map type. In Release 2.64, this cdr is written out to an error file ebaf* stored in the error directory, /opt/CiscoBAMS/data/error. Contact the Cisco Technical Assistance Center.

`** CTB209 Error <return value> reading file: CALLSTRUCT_MAP`
The <return value> represents the error value returned. The system encountered a format error reading the callstruct file. Check the CALLSTRUCT_MAP file. Contact the Cisco Technical Assistance Center.

Message Receive Task Messages

The Message Receive (DRX) task is used by one BAMS unit to receive messages from the other unit in a redundant configuration. During processing, this task writes messages to the syslog with a DRX task identifier (for example, DRX001).

`DRX000 Message Receive Task version <version number> has started`
The <version number> argument represents the task version number. The system displays the message at the start of the named task. This message is an informational message. No action is necessary.

`DRX001 Terminating with signal <signal number>`
The <signal number> argument represents the UNIX signal number. The system ends the DRX task and displays a signal number. If an operator stops the system, it terminates with signal 15. Look up the "Signal Numbers" section on page A-68. Contact the Cisco Technical Assistance Center.

`DRX002 Terminating with exit value <signal number>`
The <signal number> argument represents the UNIX signal number. The system terminates the DRX task and displays a signal number. Look up the "Signal Numbers" section on page A-68. Contact the Cisco Technical Assistance Center.

`DRX003 Established connection with remote unit`
The system has successfully established a connection with the remote unit. This is an informational message. No action is necessary.

`DRX004 Closed connection with remote unit`
The system has successfully closed the connection with the remote unit. This is an informational message. No action is necessary.

`** DRX005 Cannot wait for connections <error number>`
The <error number> argument represents the UNIX error number. The system cannot wait for connection and displays a UNIX error. Look up the error number. Contact the Cisco Technical Assistance Center.

`** DRX006 Cannot accept connection <error number>`
The <error number> argument represents the UNIX error. The system is unable to accept the connection request and displays a UNIX error. Look up the error number. Contact the Cisco Technical Assistance Center.

`** DRX007 Cannot read from connection <error number>`
The <error number> argument represents the UNIX error number. The system cannot read from the connection and displays a UNIX error. Look up the error number. Contact the Cisco Technical Assistance Center.

`DRX008 Cannot route local message queue; <message queue ID>, <return value>, <error number>`
The <message queue ID> argument represents the message queue identifier, <return value> represents the returned value, and <error number> represents the UNIX error number. The system is unable to route the local message to the intended destination and displays a UNIX error. Look up the error number. Use the ipcs utility to check the message queue status. Contact the Cisco Technical Assistance Center.

`** DRX009 Connection broken <error number>. Will reconnect`
The <error number> argument represents the UNIX error number. The system displays an error message regarding a broken connection. Look up the error number. Check if the remote BAMS unit system was stopped by the operator. Contact the Cisco Technical Assistance Center.

Message Transmit Task Messages

The Message Transmit (DTX) task oversees communication transmissions between BAMS units. During processing, this task writes messages to the syslog with a DTX task identifier (for example, DTX001).

`DTX000 Message Transmit Task version <version number> has started`
The <version number> argument represents the task version number. The system has started the Message Transmit task. This is an informational message. No action is necessary.

`DTX001 Terminating with signal <signal number>`
The <signal number> argument represents the UNIX signal. The system is terminating this task and displaying a UNIX signal. Look up the signal number. Check if the system was stopped by the operator if signal number = 15. Contact the Cisco Technical Assistance Center.

`DTX002 Terminating with exit value <exit value>`
The <exit value> argument represents the UNIX exit value. The system is terminating this task and displaying a UNIX exit value. Contact the Cisco Technical Assistance Center.

`DTX003 Established connection with remote unit`
The system has established a connection to the remote unit. This is an informational message. No action is necessary.

`DTX004 Closed connection with remote unit`
The system has closed the connection with the remote unit. This is an informational message. No action is necessary.

DTX005 Could not discard queued messages <returned value> (<error number>)
The <returned value> argument represents the return value, and <error number> represents the UNIX error number. The system is unable to discard queued messages and displays a UNIX error code. Look up the error number. Contact the Cisco Technical Assistance Center.

** DTX006 Connect attempt failed <error number>. Will retry later
The <error number> argument represents the UNIX error number. The system is unable to connect and displays a UNIX error. The system tries to establish the connection again later. Look up the error number. Contact the Cisco Technical Assistance Center.

** DTX007 Failed receiving message <return value> (<error number>)
The <return value> argument represents the returned value, and <error number> represents the UNIX error number. The system is unable to receive a message. Look up the error. Check if the remote BAMS unit is down. Contact the Cisco Technical Assistance Center.

** DTX008 Connection broken <error number>. Will re-connect
The <error number> argument represents the UNIX error number. The system reports an interrupted connection to the remote unit and displays a UNIX error. The system attempts to reconnect to the remote unit. Look up the error number. Check the network connection to the remote BAMS unit. See if the operator stopped the remote BAMS unit or if some other error occurred on that unit. Contact the Cisco Technical Assistance Center.

** DTX009 Cannot write to connection.
The system is unable to transmit information. Contact the Cisco Technical Assistance Center.

*C DTX010 Could not create message queue <return value> (<error number>).
The <return value> argument represents the returned value, and <error number> represents the UNIX error number. The system is unable to create a message queue. Look up the error number. Use the **ipcs** utility to check the message queue status. Contact the Cisco Technical Assistance Center.

* DTX011 Connect socket for host <first host name> failed, trying host <second host name>
The <first host name> argument represents the primary host name for the remote BAMS unit, and <second host name> represents the secondary name. The system was unable to connect to the remote BAMS unit using the first host name and is now trying to connect to the unit using the second host name. Check the host names. Check if a network problem exists that prevents connection to the host represented by <first host name>. Contact the Cisco Technical Assistance Center.

Extendable ASCII Task Messages

The Extendable ASCII (EXT) task generates extendable ASCII data. During processing, this task writes messages to the syslog with an EXT task identifier (for example, EXT001).

EXT000 <task name> version <version> has started pid <process identifier>
The <task name> argument represents the task name, <version> represents the version, and <process identifier> is the UNIX process identifier. This alarm defines what version of the task is running. This is an informational message. No action is required.

EXT001 <task name> terminating with code <return error code>, pid <process id>
The <task name> argument represents the task name, <return error code> represents the error code returned, and <process id> represents the UNIX process identifier. The named task has been terminated abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

EXT002 <task name> terminating with signal <signal number>, pid <process id>
The <task name> argument represents the task name, <signal number> represents the UNIX signal number, and <process id> is the UNIX process identifier. The named task has been ended by a UNIX signal. This is an informational message. No action is necessary.

`EXT003 <task name> terminating normally, pid <process id>`
The <task name> argument represents the task name, and <process id> is the UNIX process identifier. The EXT task has terminated normally. This is an informational alarm that appears only in test mode.

`EXT004 No file to be processed. Retries in <number> seconds`
The <number of seconds> argument represents the number of seconds the system waits before reattempting to process files. The task sleeps for the number of seconds displayed before accessing more files. This is an informational message. No action is necessary.

`* EXT101 File error: <data format type> on <system call> <errno>`
The <data format type> argument represents the type of data format, <system call> represents the name of the failed system call or the name of the file being operated on; and <errno> represents the UNIX error number. An error was encountered during a system call operation. Look up the UNIX error number in the listing at the end of this appendix and contact the Cisco Technical Assistance Center.

`** EXT200 Environment variable <name> not found`
The <name> argument represents the name of the missing environment variable, for example CONFIG_DIR, FILES_DIR, dir_in, dir_out. The displayed environment variable was not found. The symbol definition file (sym_defs) is either missing a variable, or the variable must be set before the task is run. Contact the Cisco Technical Assistance Center.

`(*,**) EXT201 Function <name> failed, <errno>`
The <name> argument represents the calling function name, and <errno> represents the UNIX error number. This message indicates failure of the displayed function call, that is, Major Events for open_file, stat (sysParm.CTL), ai_open ctlfile, minor events for ai_get_max_num_of_recs(), ai_load_rec(), or ai_close_file. Examine the error number. Contact the Cisco Technical Assistance Center.

`(*,**) EXT201 Function <calling function> failed, (<error number>)`
The <calling function> represents the calling function, and the <error number> argument represents the UNIX error number. The system encountered a failure trying to make a function call. Examine the system error. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`** EXT202 Invalid AMADEF format, read <error string>, expected <expected string>`
The <error string> argument represents the error string found, and <expected string> represents the string that was expected. The system encountered a format problem while trying to read the AMADEF file. The AMADEF file needs to be fixed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`* EXT205 Unsupported field <field name>, conversion not defined`
The <field name> argument represents the name of the field. The system encountered an AMA field that could not be processed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`EXT207 Processed <filename>, converted <num recs>, output = <suffix>`
The <filename> argument represents the input filename, <num recs> represents the number of records converted, and <suffix> represents the output file suffix. This message displays processing status information. This is an informational message. No action is necessary.

`(*,**) EXT208 <data type> code <structure code> not found, cannot process record`
The <data type> argument represents the type of data, and <structure code> represents the structure code or call type. The system is unable to process a record due to missing code and structure code information. Check the AMADEF file. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

`** EXT209 Error <return value> reading file: CALLSTRUCT_MAP`
The <return value> argument represents the value returned. The system encountered a format error reading the callstruct file. Check the CALLSTRUCT_MAP file. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

Formatter Task Messages

The Formatter (FMT) task formats files. FMT processes the polled raw files from the switch and converts them to an aggregated record format that BAMS uses for all processing tasks. FMT renames the timestamp in the filename to match the timestamp in the raw file header record. During processing, this task writes messages to the syslog with an FMT task identifier (for example, FMT001).

`FMT000 <task> version <version number> has started pid <process ID>.`
The <task> argument represents the task name, <version number> represents the task version number, and <process ID> represents the UNIX process identifier. This message identifies the version of the task that is running. This is an informational message. No action is necessary.

`FMT001 <task> terminating with code <return code>, pid <process ID>.`
The <task> argument represents the task name, <return code> identifies the terminating signal number encountered by the system, and <process ID> represents the UNIX process identifier. The named task ended abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

`FMT002 <task> terminating with signal <number>, pid <process ID>.`
The <task> argument represents the task name, <number> represents the UNIX signal number, and <process ID> represents the UNIX process identifier. The named task was ended by a UNIX signal.

`FMT003 <task> terminating normally, pid <process ID>.`
The <task> argument represents the task name, and <process ID> represents the UNIX process identifier. The named task ended normally. This is an informational message. No action is necessary.

`FMT004 Processing file <filename>`
The <filename> represents the name of the file being processed. The task has started processing the named file. This is an informational message. No action is necessary.

`FMT009 Received system signal <signal>`
The <signal> argument represents the received signal number. The FMT task received a system signal. This is an informational message. No action is necessary.

`FMT010 Processed <filename>, total:<number1> good:<number2> fmterr:<number3> synerr:<number4> semerr:<number5> skipcdb`
The <filename> argument represents the processed filename, <number1> represents the total number of processed call records, <number2> represents the number of good call records, <number3> represents the number of bad format call records, <number4> represents the number of call records with syntax errors, and <number5> represents the number of call records with semantic errors. This message displays call process summary information. This is an informational message. No action is necessary.

`FMT011 Processed <filename>, Skipped CDEs: <number1> Unknown CDEs: <number2>`
The <filename> argument represents the processed filename, <number1> represents the number of skipped CDEs, and <number2> represents the number of unknown CDEs. This message indicates the number of CDEs that were skipped or unknown in a processed file. This is an informational message. No action is necessary.

`FMT012 Output file <filename>`
The <filename> argument represents the processed filename. This message indicates the name of the output file. This is an informational message. No action is necessary.

`FMT019 [CDR] No flat files to process. Retry in <number> seconds.`
The <number> argument represents the number of seconds. This message indicates that no data exists for processing. The FMT task retries in the displayed number of seconds. This is an informational message. No action is necessary.

`FMT020 CDB tag <number>, CDB skipped.`
The <number> argument represents the CDB number. This CDB is skipped when the CDB tag is found in the Skip CDB table. This is an informational message. No action is necessary.

FMT021 CDE tag <number>, CDB skipped.
The <number> argument represents the CDE number. This CDE is skipped when the CDE tag is found in the Skip CDE table. This is an informational message. No action is necessary.

FMT024 Creating <name> table
The <name> argument represents the lookup table name (that is, NPANXX table, Rating Exception table, Rating Type table, or Trunk Group table). The named lookup table is created and loaded into memory. This is an informational message. No action is necessary.

FMT025 Refreshing <name> table.
The <name> argument represents the lookup table name. The lookup table is reloaded. This is an informational message. No action is necessary.

* FMT100 System call failure: <name> (<errno>)
The <name> argument represents the name of a failed system call, and <errno> represents the UNIX error number. This message indicates that the FMT task encountered an error when requesting the displayed system call. Look up the UNIX error number. Contact the Cisco Technical Assistance Center.

* FMT101 File error: <call> on <file> (<code>)
The <call> argument represents the name of the failed system call, <file> represents the filename, and <code> represents the UNIX error number. This message indicates that the FMT task encountered an error when performing the displayed system call operation. Look up the error number. Contact the Cisco Technical Assistance Center.

*C FMT102 File I/O error: <call> on <file>, requested <number1>, got <number2>
The <call> argument represents the name of the failed I/O system call, <file> represents the name of the file being operated on, <number1> represents the number of bytes requested, and <number2> represents the number of bytes received. The FMT task encountered an error when performing the displayed I/O file operation. Contact the Cisco Technical Assistance Center.

* FMT201 Undefined environment variable <name>
The <name> represents the name of the environment variable. This message indicates that the displayed environment variable is not defined. Contact the Cisco Technical Assistance Center.

* FMT202 Input file does not start with header CDB
This message indicates that the FMT task detected an invalid field value in the call record. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

* FMT203 Error reading TAG/LEN of TLV errno=<error number>
The <error number> argument represents the UNIX error number. An error occurred during the reading of the TLV (T = Tag field code, L = Length of the field value, V = field Value). The FMT task stops processing this file, renames it as an error file, and begins processing the next input file. Look up the error. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

* FMT204 Input file is not ended by footer CDB
The system encountered the displayed error. Contact the Cisco Technical Assistance Center.

* FMT207 #<index> - <name>: <value> too long, <size> expected.
The <index> argument represents the call record index, the <name> argument represents the field name, the <value> argument represents the field value, and the <size> argument represents the expected field size. This message indicates that the field string exceeded the expected size. Contact the Cisco Technical Assistance Center.

* FMT210 CTL file: <name>:<error>.
The <name> argument represents the control filename, and the <error> argument represents the displayed error string. This message indicates the error returned by a control file function. Contact the Cisco Technical Assistance Center.

* FMT211 CTL file measurement value error: <value>
The <value> argument represents the incorrect value. This message indicates that the content of the measurement field in the control file is not 0 or 1. Contact the Cisco Technical Assistance Center.

`* FMT212 CTL file lookup value error: <value>`
The <value> argument represents the measurement value. This message displays when the value of the measurement field in the control file is not 0 or 1. Check the configuration file using MML and fix the value.

`(*,**) FMT221 Function <function name>() failed, (<error number>)`
The <function name> argument represents the calling function name, and <error number> represents the UNIX error number. This message indicates the failure of a function call. See the UNIX error number for more information. Contact the Cisco Technical Assistance Center.

`* FMT222 Invalid TLV len:<length> at <position>th chars, CDB len:<size>`
The system read in invalid TLV length. The <length> argument represents the incorrect TLV length, <position> represents the TLV position in the CDB, and <size> represents the size of the CDB. The FMT task stops further processing of the input file and renames the input file by prepending "err_" to the input filename. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* FMT224 Process all CDBs: Error in parsing record.`
This message indicates that the FMT task encountered an error during the parsing of this record. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* FMT225 Dump_cde: Unrecognized tag <tag number>`
The <tag number> represents the number of the tag. This message indicates that the FMT task encountered an unrecognized tag. Verify the Cisco MGC input file. Contact the Cisco Technical Assistance Center.

`* FMT226 Incorrect field length <field value length>.`
The <field value length> argument represents the length of the field value. This message indicates that the FMT task encountered an incorrect field length. Verify that the Cisco MGC input file has valid entries. Contact the Cisco Technical Assistance Center.

`FMT232 <index> - Fail <function> lookup of <field1>: <value1> and <field2>: <value2>`
The <index> argument represents the index of the call record, the <function> argument represents the function name of the table lookup, the <field1> argument represents the first field name in the call record, the <value1> argument represents the first value of the field being looked up, the <field2> argument represents the second field name in the call record, and the <value2> argument represents the second value of the field being looked up. This message indicates that the task failed to look up the indicated fields.

`FMT234 #<record index> - Invalid record <field names/values>`
The <record index> argument represents the record index, and the <field names/values> argument represents the field names and values. This message is generated if bafoutput (Node Parameters table) is enabled and the FMT task encounters an invalid block format while reading blocks from a control file. The message shows which fields have invalid entries. Use MML to correct the values.

`FMT235 <number> record(s) failed <table> lookup for file <name>`
The <number> argument represents the number of records, the <table> argument represents either the Trunk Group table (TRUNKGRP) or the NAILED Connection table (SIGPATH), and the <name> argument represents the filename. This message shows the number of lookup failures per file. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

Correct the appropriate record in the Trunk Group table (TRUNKGRP) or in the Nailed Connection table (SIGPATH).

Example

The following example represents an FMT [23470] message from a /opt/CiscoBAMS/files/sxx/FMT_cdr.log file.

```
FMT [23470] #1 - Fail Trunk Group lookup of Orig_trk_grp: 1002
```

The example of an FMT_cdr.log file message includes the following elements:

- [23470]—the process ID of the process on the BAMS node
- #1—the record number
- Orig_trk_grp—the Trunk Group parameter
- 1002—the value that failed lookup

Note The process ID changes whenever the FMT task restarts.

`FMT236 Setting FMT processing prefix to <prefix>`
The <prefix> argument represents the file prefix. The FMT task processes files whose prefixes match the file-prefix1 parameter in the Poll table. If this parameter is not populated, the file-prefix2 parameter is used. If both parameters are empty, the default prefix is cdr. Whenever the Poll table is modified, this message includes the value of the file prefix that FMT uses.

CDR Binary Error Lookup Files

Cisco BAMS generates error lookup (elkup) files when the FMT process encounters lookup failures for a trunk group, NAILED_CONN (SIGPATH), NPA_NXX, RATE_TYPE, or RATE_EXEC, and cannot perform data conversion. The elkup error files are binary CDR files that are saved in the directory /opt/CiscoBAMS/data/s0x/error on the BAMS server.

For example, the file elkup_cdr_20080822155821_006831.bin in the directory /opt/CiscoBAMS/data/s0x/error indicates that some number of lookups failed for cdr_20080822155821_006831.bin. As in the example, for every occurrence, the elkup error file name is formed by the expression elkup_ as a prefix to the CDR bin file name.

If you login as user "bams**,**" you can display a translation of the elkup bin file by issuing the command **cdrdump** ***<elkup_cdr_bin_filename>***. The **cdrdump** program is located in the directory /opt/CiscoBAMS/contrib, but can be executed from any directory if the login user is "bams."

The data displayed in the translated bin file includes an error code that identifies the table for which the lookup failed. Table A-15 maps the error code values to the table lookup types.

Table A-15 Elkup File Error Code Identification

Elkup File Error Code	Error Type
0x000100	NPA_NXX
0x000200	TRUNKGROUP
0x000400	RATE_TYPE
0x000800	RATE_EXC
0x001000	NAILED_CONN (SIGPATH)
0x000000	**Note** This code indicates that all executed lookups were successful.

For example, the error code 0x0001200 indicates that the lookup failed for both a NAILED_CONN (SIGPATH) table and TRUNKGROUP table.

The error code value that appears in the elkup file is derived by adding (using HEX arithmetic) the individual error code values of the failed lookup functions.

Consider the derivation of the error codes in the following two examples.

Example 1

1. The error code 0x00001300 results from errors in the NAILED_CONN (0x1000), TRUNKGRP (0x200), and NPA_NXX (0x100) lookup tables.

   0x1000 + 0x200 + 0x100 = 0x00001300

Example 2

2. The error code 0x00000d00 results from errors in both the NPA_NXX (0x100), RATE_TYPE (0x400), and RATE_EXC (0x800) lookup tables.

   0x100 + 0x400 + 0x800 = 0x00000d00

To correct lookup failures, examine the relevant fields in the elkup file and the BAMS tables as described in the following sections.

NPANXX Table Errors

The elkup file error code 0x001000 indicates that a NPANXX table lookup failed. To identify the table entries that are missing or that have failed lookup, check the file FMT_cdr.log in the directory /opt/CiscoBAMS/files/s0x. Specifically, in the FMT_cdr.log file, search for FMT235 messages, which identify the table for which a number of lookup errors occurred. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

If the BAF output has been turned on, there is a lookup failure of the npanxx of either the called_number or calling_number from the elkup record. This npanxx value should be populated in the table NPANXX.

See the section "Formatter Task Messages" in this appendix.

To display the provisioned NPA_NXX table entries, issue the MML command **prov-rtrv:NPANXX:**

To reprovision an NPANXX table entry issue the MML command **prov-add**.

See the section "Updating the NPANXX Table" in Chapter 5.

Trunk Group Table Errors

The elkup file error code 0x000200 indicates that a TRUNKGROUP table lookup failed. To identify the table entries that are missing or that have failed lookup, check the file FMT_cdr.log in the directory /opt/CiscoBAMS/files/s0x. Specifically, in the FMT_cdr.log file, search for FMT235 messages, which identify the table for which a number of lookup errors occurred. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

The following sample display was extracted from a FMT_cdr.log file. The data confirms a specific trunk group lookup failure.

```
09/05/08 00:09:47 FMT [ 6890] #9525 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9526 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9527 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9528 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9529 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9530 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9531 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9532 - Fail Trunk Group lookup of Orig_trk_grp: 7000
09/05/08 00:09:47 FMT [ 6890] #9533 - Fail Trunk Group lookup of Orig_trk_grp: 7000
```

For trunk group errors, you should examine the values of orig_trk_grp, term_trk_grp, and maint_trk_grp in the elkup record. If any of these are populated, a corresponding entry should be present in the table TRUNKGRP. A lookup error indicates that at least one of the fields (orig, term, or maint) failed the TRUNKGRP lookup.

See the section "Formatter Task Messages" in this appendix.

To display the provisioned TRUNKGRP table entries, issue the MML command **prov-rtrv:TRUNKGRP:**

To reprovision a TRUNKGRP table entry issue the MML command **prov-add**.

See the section "Updating the Trunk Group Table" in Chapter 5.

Rating Type Table Errors

The elkup file error code 0x000400 indicates that a RATING-TYPE table lookup failed. To identify the table entries that are missing or that have failed lookup, check the file FMT_cdr.log in the directory /opt/CiscoBAMS/files/s0x. Specifically, in the FMT_cdr.log file, search for FMT235 messages, which identify the table for which a number of lookup errors occurred. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

If the BAF output is turned on, the field call_rate_type is populated from either the table RATE_TYPE. If the error was RATE_TYPE_LOOKUP, then lookup of the fields igr_call_zone_id or egr_call_zone_id from the elkup record failed in the table RATE_TYPE.

See the section "Formatter Task Messages" in this appendix.

To display the provisioned RATING-TYPE table entries, issue the MML command **prov-rtrv:RATING-TYPE:**

To reprovision a RATING-TYPE table entry issue the MML command **prov-add**.

See the section "Updating the Rating Type Table" in Chapter 5.

Rating Exception Table Errors

The elkup file error code 0x000800 indicates that a RATE-EXC table lookup failed. To identify the table entries that are missing or that have failed lookup, check the file FMT_cdr.log in the directory /opt/CiscoBAMS/files/s0x. Specifically, in the FMT_cdr.log file, search for FMT235 messages, which identify the table for which a number of lookup errors occurred. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

In this case, if the BAF output has been turned on, the field call_rate_type is populated from either the table RATE_EXC. If the error was RATE_EXC_LOOKUP, then the lookup of the npanxx of either the called_number or calling_number from the elkup record failed in the table RATE_EXC.

See the section "Formatter Task Messages" in this appendix.

To display the provisioned RATE-EXC table entries, issue the MML command **prov-rtrv:RATE-EXC:**

To reprovision a RATE-EXC table entry issue the MML command **prov-add**.

See the section "Updating the Rating Exception Table" in Chapter 5.

Nailed Connection Table Errors

The elkup file error code 0x000800 indicates that a SIGPATH table lookup failed. To identify the table entries that are missing or that have failed lookup, check the file FMT_cdr.log in the directory /opt/CiscoBAMS/files/s0x. Specifically, in the FMT_cdr.log file, search for FMT235 messages, which identify the table for which a number of lookup errors occurred. To find details of a lookup failure, examine the log file /opt/CiscoBAMS/files/sxx/FMT_cdr.log and search for FMT [23470] messages.

If the BAMS system is in the nailed configuration mode, you should examine the values of igr_sigpath_id, egr_sigpath_id, and maint_sigpath_id in the elkup record. If any of these are populated, a corresponding entry should be present in the table NAILED_CONN. A lookup error indicates that at least one of the fields (igr, egr, or maint) failed the NAILED_CONN lookup.

See the section "Formatter Task Messages" in this appendix.

To display the provisioned SIGPATH table entries, issue the MML command **prov-rtrv:SIGPATH:**

To reprovision a SIGPATH table entry issue the MML command **prov-add: sigpath**.

See the section "Updating the Nailed Connection Table" in Chapter 5.

Man-Machine Language Messages

The Man-Machine Language (MML) task supports the command-line interface. This task also displays syntax error messages if the operator makes a mistake entering a command. The syntax error messages in the following paragraphs are generally not written to the syslog file.

`* MML600 Missed ';' in MML_REC_MAP file, line number <#>`
The <#> argument indicates which line has a problem. The system encountered a syntax error in the record map file found in the opt/CiscoBAMS/files directory. Each line in the record map file has a maximum of 3 semicolon-separated fields. Use vi to look at the file and insert the semicolon field separator, if it is missing. Contact the Cisco Technical Assistance Center

`* MML601 Open control file <name> failed`
The <name> argument represents the name of the control file that the MML task could not open. The system was unable to open the displayed control file. Verify that the specified control file exists in the opt/CiscoBAMS/files directory, and that the group permissions (775) and ownership (bams) are properly set up. Contact the Cisco Technical Assistance Center.

`* MML603 Load record <number> failed`
The <number> argument represents the record number. The system was unable to load the displayed record. Contact the Cisco Technical Assistance Center.

`* MML604 Set field <name> failed for record <number>`
The <name> argument represents the field name that the MML task could not set, and <number> represents the call record number. The system was unable to set a field name for the displayed record. Contact the Cisco Technical Assistance Center.

`* MML605 Update record <number> failed`
The <number> argument represents the number of the record that the MML task was unable to update. The system was unable to update the displayed record. Contact the Cisco Technical Assistance Center.

`* MML606 Close control file <name> failed`
The <name> argument represents the name of the control file that the MML task was unable to close. The system was unable to close the displayed control file. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML608 Get number of keys failed`
The system was unable to get the number of keys. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML609 Missing row number or key field`
The system encountered a missing row number or key field. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML610 Get number of fields failed`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML611 Get field names failed`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`* MML612 Get field names error, NULL field name encountered`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`* MML613 Get fields <name> failed for record <number>`
The <name> argument represents the name of the field that the MML task was unable to get, and <number> represents the record number. The system was unable to get the displayed fields for the displayed record. Contact the Cisco Technical Assistance Center.

`** MML614 Error finding type for field <name>. May have problem with CFDT file.`
The <name> argument represents the field name. This message indicates an error encountered in finding the displayed field name type. There might be a problem in the configuration file. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML615 Get key names failed`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`(*,**.*C) MML616 Get max number of records failed`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MML617 Max number of records should be greater than 0`
The system experienced the displayed failure. Contact the Cisco Technical Assistance Center.

`*C MML618 Dir <destination directory> could not be created.`
The <destination directory> variable represents the destination directory name. The system was unable to create a destination directory. Contact the Cisco Technical Assistance Center.

`* MML619 System call <name> failed. errno=<error number>`
The <name> argument represents the name of the failed system call, and <error number> represents the UNIX error number. A system call failed. Look up the error number. Contact the Cisco Technical Assistance Center.

`MML621 Successful prov-dply from <source directory>.`
The <source directory> argument represents the file location specified with the **prov-dply** command. This is an informational message. No action is necessary.

`(*, **, *C) MML622 File <number> could not be opened for <filename>.`
The <number> argument represents the record number, and <filename> represents the control filename. The file might not exist or might have become corrupt. Check the file. Contact the Cisco Technical Assistance Center for further assistance.

`MML623 Read record <number> failed for control <file>.`
The <number> argument represents the record number, and <filename> represents the control filename. The control file might be in use. Try again in a few minutes. Contact the Cisco Technical Assistance Center for further assistance.

Mass Storage Control Task Messages

The Mass Storage Control (MSC) task monitors and conserves disk space on the BAMS units. It generates alarms when disk occupancy passes user-defined thresholds based on file prefixes found in specified data directories, and deletes files to free disk space.

Polled files are files that have been polled by a polling computer. Unpolled files are files that have not been polled yet. The MSC task can be set up to delete polled files when the percentage of disk space occupied by these files exceeds the user-defined thresholds. For example, if the polled upper threshold is set at 80 percent of disk occupancy and the lower polled threshold is set at 60 percent, the MSC task begins deleting polled files from oldest to newest when the total polled file size exceeds the 80 percent upper threshold. It stops deleting these files when it falls below the lower threshold. The unpolled threshold value is usually set high, for example, 90 percent, to avoid losing unpolled data. An alarm is generated to alert the user that the disk is filling up.

During processing, this task writes messages to the syslog with an MSC task identifier (for example, MSC001).

`MSC001 MSC version <version> cleaning disk`

The <version> argument represents the version number of the MSC task. The MSC task periodically deletes user-specified files from the disk at user-defined intervals so that the disk does not fill up and cause system failure. This message indicates that the MSC task is currently working. This is an informational message. No action is necessary.

`* MSC003 <string1> error <UNIX error number> on <string2>`

The <string1> argument represents the path or full filename and statistics. The <UNIX error number> argument represents the UNIX error set by the system function. The <string2> argument represents the path, or full filename, and statistics. This message indicates that an error occurred while the MSC task was searching for files to set up for deletion. An informational message is recorded in the system log file if the MSC task cannot obtain information on the file. A minor event is recorded if an error occurs while the MSC task attempts to retrieve information on the number of inodes, free disk space, and other similar information.

Check the UNIX error number to determine the reason for the error. Log in to BAMS and source the sym_defs file. Use the **make nodes** and **make home** commands, restart BAMS, and monitor the system log file to see if the error persists. If it does, restart BAMS to clear system resources.

`* MSC006 Error <UNIX error number> opening <file>`

The <UNIX error number> argument represents the UNIX error set by the system function. The <file> argument represents the filename. This message indicates that an error occurred while BAMS was attempting to locate a file within the subdirectories of the /CiscoBAMS/data directory.

Check the UNIX error number to determine the reason for the error. Verify that the msct.parm file exists and that the permissions are correct. Contact the Cisco Technical Assistance Center.

`* MSC007 Invalid parameter file`

This message indicates that the parameter file for the MSC task has a problem. The file might be corrupted or it might have an invalid parameter that was saved during MSC maintenance.

Verify that the msct.parm file exists and that the permissions are correct. Verify the information in the file. Contact the Cisco Technical Assistance Center.

`MSC008 Maximum number of files exceeded`

This message indicates that the MSC task attempted to clean up the disk by deleting files specified by the data storage attributes and has found that there are too many files on the disk. This is an informational message. No action is necessary.

Log in to BAMS and check disk usage. Delete data files from BAMS manually. The data storage attributes might not be set correctly; check them to ensure that this situation does not recur.

`MSC009 Error <UNIX error number> opening DISK_STATS file`

The <UNIX error number> argument represents the UNIX error set by the system function. This message indicates that BAMS tried to open the DISK_STATS file and encountered the specified UNIX error. The DISK_STATS file shows the amount of disk space currently being used on BAMS.

Log in to BAMS and use the **ls** command (**ls DISK_STATS**) to check if the DISK_STATS file is on the disk.

`* MSC010 Undefined environment variable: <variable>`
The <variable> argument represents the undefined variable. This message indicates that the MSC task encountered an error while attempting to run the housekeeping utility for deleting older data files. The data storage utility verifies that the path for the data directory is defined and can get the specific attributes from the msct.parm file.

Verify that the /CiscoBAMS/data directory is defined in the sym_defs file (located in the /CiscoBAMS/bin directory). Verify that the filenames set in the deletion list are correct (they are case-sensitive).

`* MSC012 Cannot convert regular expression <reg>`
The <reg> argument represents the regular expression. This message indicates that BAMS could not recognize the displayed regular expression attribute. Verify the expression used on the MSC menu through the user interface. Contact the Cisco Technical Assistance Center.

`* MSC014 regBAMS() returned NULL for pattern <reg>`
The <reg> argument represents the regular expression. This message indicates that BAMS provided a null value in place of the displayed regular expression. Verify the expression used on the MSC menu through the user interface.

`(*,**,*C) MSC015 <file> is <number> days old, exceeded <max number> day limit`
The <file> argument represents the name of the file. The <number> argument represents the number of days that the file has resided on the system, and the <max number> argument represents the user-defined file aging limit. The file aging limit (0 to 99 days) is set with the agealarm parameter in MSC-PARMS. This message indicates that BAMS has identified the displayed file as being too old to remain on the system. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MSC016 <file> is <number> days old, exceeded <max number> day limit DELETED.`
The <file> argument represents the name of the file. The <number> argument represents the number of days that the file has resided on the system and the <max number> argument represents the user-defined deletion age. The file deletion age (0 to 99 days) is set with the delage parameter in MSC-PARMS. This message indicates that BAMS has identified the displayed file as being too old to remain on the system and has deleted the file. Contact the Cisco Technical Assistance Center.

`(*,**,*C) MSC017 Disk block utilization <number> percent.`
The <number> argument represents the number of blocks as a user-defined percentage of the disk block utilization. This minor, major, or critical alarm level is set with the mil, mal, or crl parameter in the MSC Thresholds table. This message identifies the number of blocks used in the file system. The alarm level depends on the percentage of disk block usage. No action is necessary, unless there is a very high percentage of disk block usage. In that case, contact the Cisco Technical Assistance Center.

`(*,**,*C) MSC018 Disk inode utilization <number> percent`
The <number> argument represents the number of inodes as a percentage of the total number of inodes. This inode upper threshold is set with the iut parameter in the MSC Thresholds table. This message identifies the number of blocks used in the file system. The alarm level depends on the percentage of inode usage. No action is necessary, unless there is a very high percentage of inode usage. In that case, contact the Cisco Technical Assistance Center.

`MSC019 MSC waiting <number> minutes before starting.`
The <number> argument represents the number of minutes the task will wait before starting. This is set with the interval parameter in the MSC Thresholds table. The task waits the displayed number of minutes before starting. This is an informational message. No action is necessary.

`* MSC020 Access error trying to access <directory path>`
The <directory path> argument represents the directory path. Search permission has been denied on a component of the path prefix. Verify the directory path and try again. Contact the Cisco Technical Assistance Center.

`* MSC021 Not a valid directory <directory path>`

The <directory path> argument represents the directory path that is not valid. A component of the path prefix of <directory path> is not a directory. Verify the directory path and try again. Contact the Cisco Technical Assistance Center.

`* MSC022 The directory does not exist <directory path>`

The <directory path> argument represents the missing directory or file. Either a component of the path prefix or the file referred to by path does not exist. Verify the directory path and try again. Contact the Cisco Technical Assistance Center.

`*C MSC030 Data Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. This critical alarm level is set with the crl parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`** MSC031 Data Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. This major alarm level is set by the mal parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`* MSC032 Data Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. This minor alarm level is set with the mil parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`*C MSC040 Data Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. This critical alarm level is set with the crl parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`** MSC041 Data Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. This major alarm level is set with the mal parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`* MSC042 Data Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. This minor alarm level is set with the mil parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`*, **, *C MSC044 <filename> is <number of days> days old, exceeded <number> day limit, DELETED`

The <filename> argument represents the full name of the file, <number of days> represents the number of days the file has aged, and <number> represents the aging limit in days. This message indicates that the displayed file age has exceeded the aging limit, as set in the MSC Thresholds table, and the file has been deleted.

`*, **, C MSC045 <filename> is <number of days> old, exceeded <number> day warning age`

The <filename> argument represents the full name of the file, <number of days> represents the number of days the file has aged, and <number> represents the aging limit in days. This message indicates that the displayed file age has exceeded the warning limit defined in the MSC Thresholds table.

`*, **, *C MSC046 Disk block utilization: <number> percent, <filename> DELETED`

The <filename> argument represents the full name of the file, and <number> represents the disk block utilization as a percentage of the total available disk space. This message indicates that the displayed file was deleted because the disk usage exceeded the block threshold defined in the MSC Thresholds table.

`*,**,*C MSC047 Disk inode utilization: <number> percent, <filename> DELETED`

The <filename> argument represents the full name of the file, and <number> represents the inode utilization as a percentage of the total available inodes. This message indicates that the displayed file was deleted because the inode usage exceeded the inode threshold, as set with the iut parameter in the MSC Thresholds table.

`*,**,*C MSC048 Total inodes: <number> exceeded <maxnumber> inode limit, <filename> DELETED`

The <filename> argument represents the full name of the file, and <number> represents the inode utilization as a percentage of the total available inodes, <maxnumber>. This message indicates that the displayed file was deleted because the inode usage exceeded the inode limit, as set with the iut parameter in the MSC Thresholds table.

`*, **, *C MSC049 Disk block utilization: <number> percent in partition, <partition>`

The <partition> argument represents the partition name, and <number> represents the disk block utilization as a percentage of the available partition disk space. This message indicates that the partition thresholds, as defined in the MSC Thresholds table, were exceeded.

`MSC100 Update MSC task parameters`

The parameters in the MSC configuration file have been updated and saved to disk. Someone has logged on as an administrative user and has modified the MSC parameters. Administrative users have access to all of the operator-level features, and they can modify system parameters and disk files. This is an informational message. No action is necessary.

`* MSC101 Error <UNIX error number> running script: <name>`

The <UNIX error number> argument represents the UNIX error set by the system function. The <name> argument represents the script name. This message indicates that the MSC task experienced an error while running the displayed script from the command line. Look up the error number. Contact the Cisco Technical Assistance Center.

`*C MSC130 Archive Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. This critical alarm level is set with the crl parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`** MSC131 Dir Entry 1: Archive Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. “Dir Entry 1:” refers to the archive disk partition. This major alarm level is set with the mal parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`* MSC132 Dir Entry 1: Archive Disk block utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk utilization. “Dir Entry 1:” refers to the archive disk partition. This minor alarm level is set with the mil parameter in the MSC Thresholds table. This message reports on disk usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`*C MSC140 Dir Entry 1: Archive Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. “Dir Entry 1:” refers to the archive disk partition. This critical alarm level is set with the crl parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`** MSC141 Dir Entry 1: Archive Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. “Dir Entry 1:” refers to the archive disk partition. This major alarm level is set with the mal parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

`* MSC142 Dir Entry 1: Archive Disk inode utilization <number> percent`

The <number> argument represents the threshold value as a percentage of disk inode utilization. “Dir Entry 1:” refers to the archive disk partition. This minor alarm level is set with the mil parameter in the MSC Thresholds table. This message reports on disk inode usage when thresholds are exceeded. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the error alarm condition is removed.

NIC Conversion Task Messages

The NIC task outputs NICS format billing records. During processing, this task writes messages to the syslog with a NIC task identifier (for example, NIC001).

`NIC000 <task name> version <version> has started pid <process identifier>`
The <task name> argument represents the task name, <version> represents the version, and <process identifier> is the UNIX process identifier. This alarm defines what version of the task is running. This is an informational message. No action is required.

`NIC001 <task name> terminating with code <return error code>, pid <process id>`
The <task name> argument represents the task name, <return error code> represents the error code returned, and <process id> represents the UNIX process identifier. The named task has been terminated abnormally. The code identifies the error encountered by the system. This is an informational message. No action is necessary.

`NIC002 <task name> terminating with signal <signal number>, pid <process id>`
The <task name> argument represents the task name, <signal number> represents the UNIX signal number, and <process id> is the UNIX process identifier. The named task has been ended by a UNIX signal. This is an informational message. No action is necessary.

`NIC003 <task name> terminating normally, pid <process id>`
The <task name> argument represents the task name, and <process id> is the UNIX process identifier. The NIC task has terminated normally. This is an informational alarm that appears only in test mode.

`NIC004 No file to be processed. Retries in <number> seconds`
The <number> argument represents the number of seconds the system waits before reattempting to process files. The task sleeps for the number of seconds displayed before accessing more files. This is an informational message. No action is necessary.

`* NIC101 File error: <data format type> on <system call> <errno>`
The <data format type> represents the type of data format; <system call> represents the name of the failed system call or the name of the file being operated on; and <errno> represents the UNIX error number. An error was encountered during the performance of a system call operation. Look up the UNIX error number in the listing at the end of this appendix and contact the Cisco Technical Assistance Center.

`* NIC102 Cannot attach to shared memory, errno <errno>`
The <errno> argument represents the UNIX error number. The attach-to-shared-memory operation failed. Look up the UNIX error number in the listing at the end of this appendix and contact the Cisco Technical Assistance Center.

`** NIC200 Environment variable <name> not found`
The <name> argument represents the name of the missing environment variable, for example CONFIG_DIR, FILES_DIR, dir_in, or dir_out. The displayed environment variable was not found. The symbol definition file (sym_defs) is missing a variable, or the variable must be set before the task is run. Contact the Cisco Technical Assistance Center.

`(*,**) NIC201 Function <name> failed, <errno>`
The <name> argument represents the calling function name, and <errno> represents the UNIX error number. This message indicates failure of the displayed function call, that is, Major Events for open_file, stat (sysParm.CTL), ai_open ctlfile, minor events for ai_get_max_num_of_recs(), ai_load_rec(), or ai_close_file. Examine the error number. Contact the Cisco Technical Assistance Center.

`(*,**) NIC201 Function <calling function> failed, (<error number>)`
The <calling function> represents the calling function, and the <error number> argument represents the UNIX error number. The system encountered a failure trying to make a function call. Examine the system error. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

** NIC202 Invalid AMADEF format, read <error string>, expected <expected string>
The <error string> argument represents the error string found, and the <expected string> argument represents the string that was expected. The system encountered a format problem while trying to read the AMADEF file. The AMADEF file needs to be fixed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

* NIC205 Unsupported field <field name>, conversion not defined
The <field name> argument represents the name of the field. The system encountered an AMA field that cannot be processed. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

NIC207 Processed <filename>, converted <num recs>, output = <suffix>
The <filename> argument represents the input filename, <num recs> represents the number of records converted, and <suffix> represents the output file suffix. This message displays processing status information. This is an informational message. No action is necessary.

(*,**) NIC208 <data type> code <structure code> not found, cannot process record
The <data type> argument represents the type of data and <structure code> represents the structure code or call type. The system is unable to process a record due to missing code and structure code information. Check the AMADEF file. Contact the Cisco Technical Assistance Center for help in checking the system configuration and making any necessary modifications.

* NIC209 Switch name not available in data, set switch name to NULL
The switch name is not available in the data records. The switch name is used to create the NIC output file, so this field is missing (set to NULL) in the file name. Check the data configuration on the switch.

* NIC210 Remote RSN: <seqno1>, local RSN: <seqno2>, setting local value to remote value
The <seqno1> argument represents the remote NICS record sequence number (RSN), and <seqno2> represents the local NICS record sequence number. The RSN value of the remote BAMS unit was greater than the local unit. The system is setting the local value to remote. There is a chance that the RSN values for the NICS files from the two BAMS units do not match. If there has been a rotation of the polling of the NICS files from the BAMS units in the past day, there may be a gap or duplication of the RSN values of the NICS records.

* NIC211 Remote RSN file <name> not received from remote BAMS
The <name> argument represents the filename of the remote RSN file. The remote RSN file was not received from the remote BAMS unit. The NIC task will wait 1 hour for this file. The RSN synchronization will be performed again at midnight (UTC) of the next day. There is a chance that the RSN values for the NICS files from the two BAMS units do not match. If there has been a rotation of the polling of the NICS files from the BAMS units in the past day, there may be a gap or duplication of the RSN values of the NICS records.

* NIC212 RSN sync files are different. Cannot sync RSN values. Local:<name1>, remote:<name2>
The <name1> argument represents the filename of the first file of the day of the local machine, and <name2> represents the filename of the first file of the day of the remote machine. The NIC task will synchronize the NICS record sequence numbers for the first file of each day. If these files are not the same, the RSN synchronization will be aborted. The RSN synchronization will be performed again at midnight (UTC) of the next day. There is a chance that the RSN values for the NICS files from the two BAMS units do not match. If there has been a rotation of the polling of the NICS files from the BAMS units in the past day, there may be a gap or duplication of the RSN values of the NICS records.

NIC213 File <name> not found. Resetting RSN to 1
The <name> arguments represents the name of the file (that is, a NICS RSN sequence number file - NICS_RSN). The RSN seqno file was not found. The system reset the RSN to 1.

NIC214 NIC program waiting up to 1 hour for record sequence number (RSN) sync file from remote BAMS: <path>
The <path> argument represents the pathname of the remote RSN file. The NIC task is looking for the remote RSN file to synchronize the RSN values.

NIC215 RSN sync aborted. Remote BAMS is detected down
The RSN synchronization will be attempted again at midnight (UTC) of the next day. Check the remote BAMS unit to see why it is detected down.

NIC216 Remote FSN: 329, local FSN: 328, setting local value to remote value
BAMS synchronizes the local file sequence number (FSN) to the remote file sequence number.

Node Name Link Task Messages

The Node Name Link (NNL) task synchronizes MML commands to the remote BAMS unit. During processing, this task writes messages to the syslog with an NNL task identifier (for example, NNL001).

NNL101 NNL version 1.01 has started pid <pid>
The <pid> argument represents the process identifier. The NNL task has been started. This is an informational message. No action is necessary.

NNL102 NNL terminating with signal <signal number>
The <signal number> argument represents the UNIX signal number. The NNL task is stopping. This is an informational message. No action is necessary.

NNL500 Node <number>: changing nodename <oldname> to <newname>
The <number> argument represents the node number. The <oldname> and <newname> arguments represent the old node name and the new node name, respectively. The node name has been modified through the MML command **set-nodename**. The node name on the local unit has been changed by the MML command. The node name on the remote unit is synchronized by the NNL task. This is an informational message. No action is necessary.

NNL514 Generate alarm <alarm ID> for node <node number>. Level = <alarm level>
An alarm was generated from the user interface (mml) of the remote BAMS units. The NNL task generates the same alarm on the local BAMS unit.

Redundant Unit Polling Task Messages

The Redundant Unit Polling (POL) task collects data from the Cisco MGCs. During processing, this task writes messages to the syslog with a POL task identifier (for example, POL001).

POL000 Entering <name>()
The <name> argument represents the function name. The named function has been entered. This is an informational message. No action is necessary.

POL001 Leaving <name>()
The <name> argument represents the function name. The named function has been left. This is an informational message. No action is necessary.

POL002 Assertion failed in <name>, line <number>
The <name> argument represents the function name and the <number> argument represents the line number. An assertion failed in the named function. This is an informational message. No action is necessary.

POL100 POL Version <version number> for site <site number> has started
The <version number> argument represents the software or task version, and <site number> represents the polling site number. The POL task has been started. This is an informational message. No action is necessary.

POL104 Polling VSC unit <MGC node name>
The <MGC node name> argument represents the name of the Cisco MGC. The POL task has started to poll the named Cisco MGC unit. This is an informational message. No action is necessary.

*POL105 Polling unit rotates from unit <from unit number> to <to unit number> successfully
The <from unit number> argument represents the number of the unit that was actively polling before the polling unit rotation and the <to unit number> argument represents the number of the unit that was the standby unit before the polling unit rotation. The unit responsible for polling has rotated as displayed.

POL107 File <filename> received from unit <from unit number> successfully
The <filename> argument represents the name of the file that was successfully received from the polling unit and the <from unit number> represents the remote polling unit number. The POL task has received a file as displayed. This is an informational message. No action is necessary.

POL108 Retry #<retry times> will start in <number>
The <retry times> argument represents the number of times the task has retried to poll the site, and the <number> argument represents the number of seconds the task waits before attempting to poll the site again. The POL task retries to poll the data. This is an informational message. No action is necessary.

POL109 Successfully polled <filename> with size <file size>
The <filename> argument represents the name of the file that was successfully polled and the <file size> argument represents the size of the polled file. The POL task has polled the named file. This is an informational message. No action is necessary.

POL110 Cannot update CTLFILE from polling unit to standby unit
The task performed an illegal operation in attempting to update the control file. Check the task permissions and times. The control file is locked if either unit is actively polling, and it can be modified only when both units are at rest.

POL111 This is standby unit
The identified unit is not the current polling unit. This is an informational message. No action is necessary.

POL112 Polling VSC host <host name1> failed, trying host <host name2>
The <host name1> argument represents the name of the first Cisco MGC host line polled and the <host name2> argument represents the name of the Cisco MGC host for the second Ethernet card. The system's attempt to poll the first Cisco MGC host failed, and it is now trying to poll the Cisco MGC host for the second Ethernet card. This is an informational message. No action is necessary.

* POL113 Node is inactive, session terminated
Polling has been disabled. This minor message occurs once when a node is provisioned from the active state to the inactive state with the activate parameter in the Node Parameters table. Once this alarm is set, it is not cleared until the node is once again activated or the user clears it. Investigate to make sure that node de-activation was intentional because switch files will no longer be polled for this node. Clear the alarm through the mml command **clr-alm**. If polling should not have been deactivated, then set the activate parameter in the Node Parameters table to ENABLE (1), and the alarm will be cleared automatically.

POL114 Rename operation failed for file <filename> on polled unit <unit number>
The <filename> argument represents the file that could not be renamed and the <unit number> argument represents the polled unit number. After a successful poll of a file from the VSC, BAMS was not able to rename the file with a *.finished suffix. Check the polling directory on the VSC units to make sure the login user for the VSC has sufficient permissions to delete and rename files within this directory. The POL task does not reattempt to rename or delete files after they have already been successfully polled, so you must manually rename or delete these files on the VSC machines.

`** POL115 Rename operation failed on polled unit <unit number>, this alarm must be manually cleared`

The <unit number> argument represents the polled unit number. This major alarm is generated the first time that a file rename operation fails. It is not generated again until this alarm has been manually cleared by the operator and the rename operation fails again. Check the polling directory on the VSC units to make sure the login user for the VSC has sufficient permissions to delete and rename files within this directory. The POL task does not reattempt to rename or delete files after they have already been successfully polled, so you must manually rename or delete these files on the VSC machines. This alarm must be manually cleared through use of the MML command **clr-alm**.

`POL116 Search unit <unit number> for files matching <file pattern>`

The <unit number> argument represents the BAMS unit number, and the <file pattern> argument represents the polled file pattern. On the *active* polling unit, this message is displayed for each polling session. This message indicates the unit number of the polled unit. Also shown is the file pattern of the polled file, consisting of the file-prefix, start sequence number, and file-suffix parameters. No action is necessary.

`POL201 POL session ended normally`

The poll session ended. This is an informational message. No action is necessary.

`POL202 POL session from BAMS unit <unit number> starts`

The <unit number> argument represents the BAMS unit number. The poll session has started. This is an informational message. No action is necessary.

`* POL203 chgno executed, unit: <unitno> old: <seqno> new: <seqno>`

The <unit no> represents the unit number of the BAMS unit, <seqno> represents the old sequence number, and the second <seqno> represents the new sequence number. The sequence number has changed. This is recorded as a minor event for audit purposes only.

`* POL204 chgno cannot execute, POL record is in use`

The sequence number cannot be changed, because the control file is in use. Try again later.

`* POL300 Environment variable <environment variable name> is undefined`

The <environment variable name> argument represents the name of the environment variable. The named environment variable is not defined. Contact the Cisco Technical Assistance Center.

`* POL302 regcmp returned NULL for pattern <pattern>`

The <pattern> argument represents the polling pattern. This message indicates that an error occurred in a regular expression comparison and that the file pattern does not match. Verify the polling parameters and file prefix and suffix values. Contact the Cisco Technical Assistance Center.

`* POL305 Invalid operation: <operation number>`

The <operation number> argument represents the invalid operation. The system was unable to perform the specified FTP operation; it is not valid. Contact the Cisco Technical Assistance Center.

`* POL306 Cannot open <filename>`

The <filename> argument represents the name of the file that cannot be opened. The system is unable to open the named file. This message indicates that the polling task cannot open the named file on the BAMS unit. Check the directory permissions and file ownership. Contact the Cisco Technical Assistance Center.

`* POL307 Poll failed for <filename>`

The <filename> argument represents the name of the file that could not be polled. The system was unable to poll the named file. Manually test whether FTP is working. Create a test file on the Cisco MGC and manually FTP it to the BAMS unit. This message might indicate a Cisco MGC or network problem. Contact the Cisco Technical Assistance Center.

`* POL308 Cannot get file status of <filename>`

The <filename> argument represents the name of the file. The system was unable to obtain a file status for the named file. Check if the named file exists on the local BAMS unit. Check the file, directory, and ownership permissions. Contact the Cisco Technical Assistance Center.

* POL309 File size incorrect for polling <filename>

The <filename> argument represents the name of the file to be polled. The polling file size is not correct. The polling task compares the size of the named file with what is expected from the Cisco MGC. Check whether a test file can be transferred by FTP to the BAMS unit from the Cisco MGC. If the file size is different, a network problem is indicated. Contact the Cisco Technical Assistance Center.

* POL312 Failed updating control record file

The POL task was unable to update the control record file. This message indicates that the control file is locked. Use the **who** command to check if another user is logged in and might have opened the control file. Check the MML command history in the MML daily log file. Contact the Cisco Technical Assistance Center.

* POL321 Cannot open directory file <directory name>

The <directory name> represents the name of the directory. The POL task cannot open the directory. If the problem persists, contact the Cisco Technical Assistance Center.

* POL322 Parent process: system() call \"<command name>"\ failed, errno=<error number>

The <command name> argument represents the name of the command on which the system call failed, and the <error number> argument represents the UNIX error number. The parent POL task displays an error code for a failed system call. Look up the UNIX error. Contact the Cisco Technical Assistance Center.

* POL323 Child process: system() call \"command name"\ failed, errno=<error number>

The <command name> argument represents the name of the command on which the system call failed, and the <error number> argument represents the UNIX error number. The child POL task displays an error code for a failed system call. Look up the UNIX error. Contact the Cisco Technical Assistance Center.

* POL325 fork() failed

The system reports a fork failure for the named site. This message indicates the task cannot create a child process. Restart the application. If the problem persists, contact the Cisco Technical Assistance Center.

* POL326 Login failed for VSC unit <MGC node name>, please check ID/passwd

The <MGC node name> argument represents the MGC node name. The login has failed for the Cisco MGC unit. Check the user ID and password. Contact the Cisco Technical Assistance Center.

* POL327 Login failed for VSC unit <MGC node name>, please check ID/passwd

The <MGC node name> argument represents the name of the Cisco MGC node. The system was unable to log in to the named Cisco MGC unit. Check the user ID and password. Check **setpollpwd**. Contact the Cisco Technical Assistance Center. To clear this alarm, you must provide a matching text message, because multiple POL327 alarms can be active (each with a different message).

Note This alarm is cleared after the alarm condition is removed.

* POL328 File <filename> send to redundant poll unit failed

The <filename> argument represents the name of the file intended to be sent. The system was unable to send the named file to the redundant unit. Check that FTP is working on the BAMS unit. Create a test file on the Cisco MGC and manually FTP it to the BAMS unit. If the attempt is unsuccessful, there might be a network problem. Contact the Cisco Technical Assistance Center.

POL329 File <filename> sent to redundant poll unit successfully

The <filename> argument represents the name of the file sent. In a redundant system, after a Cisco MGC file is polled by one BAMS unit, it is processed by that unit and is also sent to the redundant BAMS unit for processing. This is an informational message. No action is necessary.

POL330 Cannot open XFR directory, errno <error number>

The <error number> argument represents the error number. The polling task cannot open the XFR directory to transfer files to the redundant BAMS unit. Check the error. If the problem persists, contact the Cisco Technical Assistance Center.

* POL331 Polling unit rotates from unit <from unit number> to <to unit number> failed

The <from unit number> argument represents the polling unit that was actively polling just before the unit rotation and the <to unit number> argument represents the unit that was on standby. The system was unable to perform the requested polling unit rotation. Check the network connection between the BAMS units. Perform an FTP test. Contact the Cisco Technical Assistance Center.

* POL332 Polling unit rotates from unit <from unit number> to <to unit number>, but CTLFILE transfer failed

The <from unit number> argument represents the polling unit that was actively polling just before the unit rotation and <to unit number> represents the unit that was on standby. The POL task was able to perform the requested polling unit rotation, but it was unable to transfer the necessary control file information. Check the network and perform an FTP test. Contact the Cisco Technical Assistance Center.

* POL333 Error in linking to <directory name> directory, errno <error number>

The <directory name> argument represents the directory name. The <error number> argument represents the UNIX error code. The POL task was unable to perform the requested file linking operation. Check the permissions for the specified directory. Contact the Cisco Technical Assistance Center.

* POL334 Cannot attach to shared memory, errno <error number>

The <error number> argument represents the UNIX error code. The POL task was unable to access the shared device memory. Use the ipcs utility to check the shared memory status. Contact the Cisco Technical Assistance Center.

* POL335 Rotate cannot be executed in non-polling unit

The POL task was unable to rotate the requested unit that was on standby at the time. The system only allows polling unit rotation from the current polling unit to the standby unit.

* POL336 File <filename> received from unit <from unit number> failed

The <filename> argument represents the name of the file that the system was unable to receive and the <from unit number> argument represents the unit number of the sending device. The POL task was unable to receive a file from the named unit and site. Perform an FTP test. Contact the Cisco Technical Assistance Center.

* POL338 Site number invalid or not specified

The POL task is unable to validate the site number; it might be undefined. Check polling parameters, and ensure that the site exists. Contact the Cisco Technical Assistance Center.

* POL340 No data in 3 consecutive polling sessions

The POL task was unable to execute the requested polling operation in three consecutive attempts. Check the Cisco MGC and verify the existence of data. Use the **chgno** command to check the sequence numbers on the BAMS units. Contact the Cisco Technical Assistance Center.

* POL341 Cannot access shared memory, MGR may not have started

The polling tasks cannot get the contents of the shared memory. If the problem persists, contact the Cisco Technical Assistance Center.

* POL342 Cannot <access> xfr directory <directory name>, errno <error number>

The <access> argument represents the operation, <directory name> represents the name of the directory, and <error number> represents the error number. The polling task is unable to perform the named operation on the xfr directory. Check the error message. If the problem persists, contact the Cisco Technical Assistance Center.

* POL343 Cannot <operation> site data directory <directory>, errno <error number>

The <operation> argument represents the attempted system operation, <directory> represents the site data directory, and <error number> represents the UNIX error number. The POL task is unable to perform the specified system operation on the site data directory and displays a UNIX error code. Look up the error. Check the permissions of the BAMS user to access or create this directory. Contact the Cisco Technical Assistance Center.

* POL353 Cannot read record in file <control filename>.CTL

The <control filename> argument represents the control filename. The POL task was unable to access the requested record. The record might be locked by another process. Determine what other process is using and locking the record. Contact the Cisco Technical Assistance Center.

* POL356 Record already in use in file <control filename>.CTL

The <control filename> argument represents the control filename. This record is already locked by another process. Try again later. If the problem persists, contact the Cisco Technical Assistance Center.

* POL358 Invalid REDUNDANCY_MODE

The symbol definitions contain an invalid redundancy mode specification. Check the sym_defs configuration file in the opt/CiscoBAMS/bin directory. The Redundancy mode value should be 2 for redundant and 1 for simplex configuration. Run the **change_mode** command in the opt/CiscoBAMS/bin directory if the setting is wrong. Contact the Cisco Technical Assistance Center.

* POL359 Cannot rotate unit in simplex mode

The POL task is unable to rotate polling units, because the system in operating in simplex (single-unit) mode, which does not permit a polling unit rotation. Contact the Cisco Technical Assistance Center.

* POL360 Open control file <filename> failed

The <filename> argument represents the name of the control file. The POL task is unable to open the control file. Check permissions of the control file and ownership. Contact the Cisco Technical Assistance Center.

* POL361 Polling config record is blank in <filename>

The <file name> argument represents the name of the control file. The polling configuration record is blank. Check the polling configuration file and verify that polling configuration file is populated for this site. Contact the Cisco Technical Assistance Center.

Note This alarm is cleared after the alarm condition is removed.

* POL362 Sequence number is blank for config record in <filename>.CTL

The <filename> argument represents the name of the control file. The sequence number is blank. Use the **chgno** command to specify the sequence number to poll. Contact the Cisco Technical Assistance Center.

* POL363 Cannot open <directory name> directory, errno <error number>

The <directory name> argument represents the name of the polled directory, and <error number> represents the UNIX error. The POL task cannot open the named directory and returns a UNIX error number. Check the UNIX error number and permissions on the directory.

* POL364 <function name> error for CTL file <filename>: <error message>

The <function name> argument represents the name of the webapi function, <filename> represents the name of the CTL file, and <error message> represents the error message. The webapi function returns an error for the displayed CTL file. Contact the Cisco Technical Assistance Center.

*C POL401 Max FTP failures for one file reached

The POL task has exceeded the maximum FTP failure threshold specified for a single file polling operation. Contact the Cisco Technical Assistance Center *immediately* and request that personnel there check the network and perform an FTP test.

Note This alarm is cleared after the alarm condition is removed.

*C POL402 Cannot connect to unit <unit name>
The <unit name> argument represents the name of the unit the system is attempting to poll. The system is unable to connect to the specified unit. Contact the Cisco Technical Assistance Center *immediately* and request that personnel there check the network and perform an FTP test. To clear this alarm, you must provide a matching text message, because multiple POL402 alarms can be active (each with a different message).

Note This alarm clears after the alarm condition is removed, or the system connects to either the principal (hostname) or the backup (hostname_b) site.

*C POL403 Standby/Standby state detected. BAMS unit <unit number> will go active, seqno1: <sequence number 1> seqno2: <sequence number 2>
The <unit number> represents the unit number of the BAMS unit, <sequence number 1> represents the first sequence number, and <sequence number 2> represents the second sequence number. If a standby/standby condition is detected where both BAMS units are in the standby state for three consecutive intervals, the POL task of BAMS unit 0 goes active. This message is generated to indicate this condition. No action is required. BAMS has automatically recovered from a problem condition (standby/standby state).

Control File Receive Task Messages

The Control File Receive (RCV) task receives control files. During processing, this task writes messages to the syslog with an RCV task identifier (for example, RCV001).

*RCV501 Error loading record <error number>
The <error number> argument represents the UNIX error code. The RCV task is unable to load a control record. Look up the error number. Contact the Cisco Technical Assistance Center.

* RCV502 Invalid operation <operation number>
The <operation number> argument represents the operation number. The RCV task has performed an invalid operation. Contact the Cisco Technical Assistance Center.

* RCV503 Cannot open <filename>
The <filename> argument represents the name of the file that could not be opened. The RCV task is unable to open the named file. Verify that the file exists and that the file directory can be written to by user bams. Contact the Cisco Technical Assistance Center.

* RCV507 FTP failed for <filename>
The <filename> argument represents the name of the file that the RCV task could not operate on. The RCV task is unable to perform an FTP operation on the named file. Check if the control file name exists on the remote BAMS unit. Check the network connection to the remote BAMS unit and perform an FTP test. Contact the Cisco Technical Assistance Center.

* RCV508 Cannot get file status of <filename>
The <filename> argument represents the name of the file. The RCV task is unable to get a file status for the named file. Check if the control file exists on the local BAMS unit and if the user has read permissions on the file. Contact the Cisco Technical Assistance Center.

* RCV509 File size incorrect for FTP <filename>
The <filename> argument represents the name of the file. The RCV task is unable to match the file size for the named file. Perform an FTP test. If the file sizes match, contact the Cisco Technical Assistance Center.

* RCV520 Cannot connect to unit <unit host name>
The <unit host name> argument represents the name of the unit host. The RCV task is unable to connect to the other unit. Check the network connection to <unit host name>. Contact the Cisco Technical Assistance Center.

* RCV521 Cannot open directory file <directory name>
The <directory name> argument represents the name of the directory. The RCV task is unable to open the named directory file. Check if the user has permission to read directory <directory name>. Contact the Cisco Technical Assistance Center.

* RCV522 system () call failed, errno = <error number>
The <error number> argument represents the UNIX error number. A system call has failed and the system displays a UNIX error. Look up the error number. Contact the Cisco Technical Assistance Center.

* RCV524 Max FTP failures for one file reached
The RCV task has failed to FTP a file after the maximum number of tries. Check the network connection to the remote BAMS unit and perform an FTP test. Contact the Cisco Technical Assistance Center.

* RCV525 fork() failed
The RCV task experienced a fork failure at the named site. Contact the Cisco Technical Assistance Center.

* RCV527 Login failed for unit <unit host name>, please check ID/passwd
The <unit host name> argument represents the name of the unit host. The login has failed for the other BAMS unit. Check the user ID and password using **setbamunit**. Contact the Cisco Technical Assistance Center.

RCV528 Receive control information from redundant poll unit host <unit host 1> failed, trying host <unit host2>
The <unit host 1> argument represents the name of the first unit host, and <unit host 2> represents the name of the unit host for the second Ethernet card. The RCV task was unable to receive the control file from the displayed host. Check the network connection to the remote BAMS <unit host 1> and perform an FTP test.

Send File Transfer Task Messages

The Send File Transfer (SXF) task sends control files to a redundant BAMS unit. During processing, this task writes messages to the syslog with an SXF task identifier (for example, SXF001).

* SXF037 Error loading control record, returned <return value>
The <return value> argument represents the system-returned error value. The SXF task experienced an error loading a control record and returned an error value. Contact the Cisco Technical Assistance Center.

SXF041 Send control information to redundant poll unit host <unit host name> failed, trying host <unit host name of second ethernet card>
The <unit host name> argument represents the first unit host name polled, and <unit host name of second ethernet card> represents the unit host name of the second Ethernet card. The SXF task failed in its attempt to send the control record and is trying another line. Check the network connection to the remote BAMS <unit host name> and perform an FTP test.

* SXF502 stat () on <filename> failed, errno <error number>
The <filename> argument represents the name of the file, and <error number> represents the UNIX error number. Check the UNIX error.

`* SXF503 <filename> size incorrect: remote=<remote size>, local=<standard buffer size>`
The <filename> argument represents the name of the file, <remote size> represents the size on the remote unit, and <standard buffer size> represents the standard buffer size on the local unit. Check the network status. A communication problem exists between the two BAMS units. Check whether the Ethernet connection is good. Verify that the setbamunit utility has the correct parameters.

`* SXF510 Attempt #<number> to rename or transfer <filename> failed`
The <number> argument represents the number of communication attempts, and <filename> represents the name of file to be transferred or renamed. Check the network status. A communication problem exists between the two BAMS units. Check whether the Ethernet connection is good. Verify that the setbamunit utility has the correct parameters.

`* SXF640 <directory filename> No such file or directory.`
The <directory filename> argument represents the directory file name. Verify the path exists on the remote unit.

UNIX Error Numbers

Table A-16 lists the UNIX error numbers that might appear within system messages that BAMS tasks write to the syslog.

Table A-16 UNIX Error Numbers

Number	Reason
1	Not super-user
2	No such file or directory
3	No such process
4	Interrupted system call
5	I/O error
6	No such device or address
7	Arg list too long
8	Exec format error
9	Bad file number
18	Cross-device link
19	No such device
20	Not a directory
21	Is a directory
22	Invalid argument
23	File table overflow
24	Too many open files
25	Inappropriate ioctl for device
26	Text file busy
27	File too large
28	No space left on device
29	Illegal seek

Table A-16 UNIX Error Numbers (continued)

Number	Reason
30	Read only file system
31	Too many links
32	Broken pipe
33	Math argument out of domain of function
34	Math result not representable
35	No message of desired type
36	Identifier removed
37	Channel number out of range
38	Level 2 not synchronized
39	Level 3 halted
40	Level 3 reset
41	Link number out of range
42	Protocol driver not attached
43	No CSI structure available
44	Level 2 halted
45	Deadlock condition
46	No record locks available
47	Operation canceled
48	Operation not supported
49	Disk quota exceeded
50	Invalid exchange
51	Invalid request descriptor
52	Exchange full
53	No anode
54	Invalid request code
55	Invalid slot
56	File locking deadlock error
57	Bad font file format
60	Device not a stream
61	No data (for no delay I/O)
62	Timer expired
63	Out of streams resources
64	Machine is not on the network
65	Package not installed
66	The object is remote
67	The link has been severed

Table A-16 UNIX Error Numbers (continued)

Number	Reason
68	Advertise error
69	Srmount error
70	Communication error on send
71	Protocol error
74	Multihop attempted
77	Trying to read unreadable message
78	Path name is too long
79	Value too large to be stored in data type
80	Given log. name not unique
81	F.D. invalid for this operation
82	Remote address changed
83	Cannot access a needed shared library
84	Accessing a corrupted shared library
85	.lib section in a.out corrupted
86	Attempting to link in too many libraries
87	Attempting to execute a shared library
88	Illegal byte sequence
89	Unsupported file system operation
90	Symbolic link loop
91	Restartable system call
92	If pipe/FIFO, do not sleep in stream head
93	Directory not empty
94	Too many users (for UFS)
95	Socket operation on non-socket
96	Destination address required
97	Message too long
98	Protocol wrong type of socket
99	Protocol not available
120	Protocol not supported
121	Socket type not supported
122	Operation not supported on socket
123	Protocol family not supported
124	Address family not supported by protocol family
125	Address already in use
126	Cannot assign requested address
127	Network is down

Table A-16 UNIX Error Numbers (continued)

Number	Reason
128	Network is unreachable
129	Network dropped connection because of reset
130	Software caused connection abort
131	Connection reset by peer
132	No buffer space available
133	Socket is already connected
134	Socket is not connected
143	Cannot send after socket shutdown
144	Too many references: can't splice
145	Connection timed out
146	Connection refused
147	Host is down
148	No route to host
149	Operation already in progress
150	Operation now in progress
151	Stale NFS file handle

Signal Numbers

Table A-17 displays the signal numbers that might appear within system messages that BAMS tasks write to the syslog.

Table A-17 Signal Numbers

Number	Name	Reason
1	SIGHUP	Hangup
2	SIGINT	Interrupt (rubout)
3	SIGQUIT	Quit (ASCII FS)
4	SIGILL	Illegal instruction (not reset when caught)
5	SIGTRAP	Trace trap (not reset when caught)
6	SIGIOT	IOT instruction
7	SIGEMT	EMT instruction
8	SIGFPE	Floating-point exception
9	SIGKILL	Kill (cannot be caught or ignored)
10	SIGBUS	Bus error
11	SIGSEGV	Segmentation violation
12	SIGSYS	Bad argument to system call

Table A-17 Signal Numbers (continued)

Number	Name	Reason
13	SIGPIPE	Write on a pipe with no one to read it
14	SIGALRM	Alarm clock
15	SIGTERM	Software termination signal from kill
16	SIGUSR1	User-defined signal 1
17	SIGUSR2	User-defined signal 2
18	SIGCLD	Child status change
18	SIGCHLD	Child status change alias (POSIX)
19	SIGPWR	Power-fail restart
20	SIGWINCH	Window size change
21	SIGURG	Urgent socket condition
22	SIGPOLL	Pollable event condition
22	SIGIO	Socket I/O possible (SIGPOLL alias)
23	SIGSTOP	Stop (cannot be caught or ignored)
24	SIGSTP	User stop requested from tty
25	SIGCONT	Stop process has been continued
26	SIGTTIN	Background tty read attempted
27	SIGTTOU	Background tty write attempted
28	SIGVTALRM	Virtual time expired
29	SIGPROF	Profiling timer expired
30	SIGXCPU	Exceeded CPU limit
31	SIGXFSZ	Exceeded file size limit
32	SIGWAITING	Process's lwps are blocked
33	SIGLWP	Special signal used by thread library
34	SIGFREEZE	Special signal used by CPR
35	SIGTHAW	Special signal used by CPR
36	SIGCANCEL	Thread cancellation signal used by libthread
37	SIGTMIN	First (highest-priority) real-time signal
44	SIGRTMAX	Last (lowest-priority) real-time signal

Upgrading to BAMS Release 3.20

Introduction

Caution Before upgrading to a new BAMS release, ensure that you have backed up the latest configuration and data files. For more information, see the "System Backup and Recovery" section on page 1-3.

Note Before beginning this upgrade procedure, read the entire appendix to familiarize yourself with the prerequisites and major steps of the upgrade process, as well as the differences between the two releases.

The upgrade procedure automatically converts configuration and data files from previous releases to Release 3.20 format.

Upgrade Procedure

Use the following procedure (on both units for redundant systems) to upgrade to BAMS Release 3.20.

Caution When you upgrade to Release 3.20, measurement data is zeroed out and measurement data that was in progress at the time of the upgrade is not retained.

Step 1 As the BAMS user, perform a **stop_system** to shut down the BAMS application.

Step 2 Log in as root user.

Step 3 Run the BAMS uninstall script. (See the "Uninstalling BAMS" section on page 2-3.)

Step 4 In response to the following question, enter Yes:

```
Would you like the current active configuration to be saved as the restore configuration?
[y,n] y
```

Step 5 In response to the following question, answer No:

```
Would you like to remove the data directories? [y,n] n
```

Step 6 In response to the following question, answer No:

```
Would you like to remove the archive directories? [y,n] n
```

Step 7 Enter the command **pkgrm CSCOcaBAM**

Step 8 Reboot the system.

Step 9 Login as root user.

Step 10 Enter the command **pkgadd -d** on the base BAMS 3.20 package (CSCOcaBAM).

Step 11 Run the BAMS install script:

```
cd /opt/install
./bams install
```

Step 12 Login as bams and perform a **start_system** to restart the BAMS application.

BAMS 3.20 Configuration Changes

The Nailed Connection (SIGPATH) table has been modified in BAMS 3.20. For nailed configurations (signaling mode), BAMS uses the Nailed Connection table to map the sigpath to logical trunk groups. For Release 3.20, BAMS uses only the sigpath value to map to the trunk group. The bearchan and trunknum columns from previous releases have been removed. With these columns deleted, the Nailed Connection table has been transformed to a one-to-one mapping between the sigpath and trunkgrp columns (that is, for any value of sigpath there is only one trunkgrp value and vice versa).

When BAMS 3.20 is installed and the configuration of the previous version is restored, the Nailed Connection table is automatically converted so that the bearchan and trunknum columns are deleted. In this conversion, any resulting records that violate the one-to-one mapping are deleted from the table. Because of this change in which the Nailed Connection table has been transformed to a one-to-one mapping, the circuits column from the Trunk Group (TRUNKGRP) table may need to be modified to reflect the changes.

When BAMS 3.20 is installed, the default value for PGW_DYNAMIC_UPDATE is FALSE, and the structure of the Trunk Group table is the same as in pre-3.20 releases. In this mode, the circuits column from the Trunk Group table exists and is used in measurements calculations. When the set-pgw-mode utility is executed and PGW_DYNAMIC_UPDATE is set to TRUE (see the "Setting the PGW Dynamic Update Mode" section on page 2-16), the circuits column is deleted from the Trunk Group table. In this mode, the value for the number of circuits per trunk group is taken from the 1071 CDBs as received from the PGW.

The convert_all_nodes_mml_NailedConn.sh utility described below is useful for converting any MML batch scripts to match the new configuration structures in BAMS 3.20.

BAMS 3.20 MML Script Conversion Utility

BAMS 3.20 includes a utility you can use to convert any pre-3.20 MML provisioning scripts to the 3.20 format. The convert_all_nodes_mml_NailedConn.sh utility is stored in the /opt/CiscoBAMS/bin directory. This program converts any SIGPATH and TRUNKGRP provisioning entries so that they are compatible with the 3.20 provisioning structures.

The convert_all_nodes_mml_NailedConn.sh script is executed during the installation of BAMS 3.20 if you choose to restore the configuration of the previous installation. In this case, the utility is executed on the file /usr/tmp/bams_phase3.sav, which is the MML provisioning script created when BAMS is uninstalled. This script can also be executed manually on any user MML scripts.

When the convert_all_nodes_mml_NailedConn.sh utility is executed, the bearchan and trunknum parameters of SIGPATH provisioning entries are deleted. Also deleted are the duplicate records resulting from the deletion of these columns. The convert_all_nodes_mml_NailedConn.sh utility can be executed on MML provisioning scripts that include multi-node entries. For these scripts, the duplicate SIGPATH entries are identified and filtered for each set of provisioning commands delimited by the **set-node** command.

The TRUNKGRP entries are modified if the BAMS system configuration setting of PGW_DYNAMIC_UPDATE is TRUE. If this is the case, the circuits field is deleted from the TRUNKGRP provisioning entries.

Here are the steps for using the convert_all_nodes_mml_NailedConn.sh utility:

Step 1 Log in as the bams user.

Step 2 Go to the directory where the MML provisioning script is stored.

Step 3 Execute the utility with the MML provisioning script as an argument. For example:

```
$ convert_all_nodes_mml_NailedConn.sh <name of mml batch script>
```

After execution, the specified file is transformed to be compatible with the BAMS 3.20 structures. A copy of the original MML provisioning script is created. For example:

```
$ cd /opt/CiscoBAMS/files/s01/mml/bams_system

$ ls -l *mml
-rw-rw-r--   1 bams     bams       29202 Oct 14 18:23 bams_system.mml
$ convert_all_nodes_mml_NailedConn.sh bams_system.mml
moving bams_system.mml to bams_system.mml.20041014182549.old

"PGW_DYNAMIC_UPDATE" is 0.
The "circuits" column will not be removed.

Converting bams_system.mml node:sys
Converting bams_system.mml node:1
WARNING: SIGPATH: sigpath=0x00140003 TRUNK GRP: trunkgrp=4001 lost
WARNING: SIGPATH: sigpath=0x00140004 TRUNK GRP: trunkgrp=4001 lost
WARNING: SIGPATH: sigpath=0x00140008 TRUNK GRP: trunkgrp=4004 lost
Converting bams_system.mml node:2
Converting bams_system.mml node:3
Converting bams_system.mml node:4
Converting bams_system.mml node:5
Converting bams_system.mml node:6
Converting bams_system.mml node:7
WARNING: SIGPATH: sigpath=0x00150002 TRUNK GRP: trunkgrp=4001 lost
WARNING: SIGPATH: sigpath=0x00150003 TRUNK GRP: trunkgrp=4001 lost
WARNING: SIGPATH: sigpath=0x00150004 TRUNK GRP: trunkgrp=4001 lost
WARNING: SIGPATH: sigpath=0x00150008 TRUNK GRP: trunkgrp=4004 lost
Converting bams_system.mml node:8
$ ls -l bams_system*
-rw-rw-r--   1 bams     bams       16620 Oct 14 18:25 bams_system.mml
-rw-rw-r--   1 bams     bams       29202 Oct 14 18:25 bams_system.mml.20041014182549.old
$
```

Backing Up and Restoring BAMS

Backup Utility

You can use the BAMS utility, **bamsbackup**, to back up the current configuration files and all executable files and intermediate files (that is, incompletely processed files in the /output directories and checkpoint files). You can also specify whether you wish to back up the billing files (BAMS output files and the archived CDR files from the PGW). You can specify whether the back up is to a regular file or a tape device.

Note Before you run the **bamsbackup** utility, check to ensure that the file ADMss is present in the directory /opt/CiscoBAMS/bin. The ADMss file must be present before you start this utility.

Format `bamsbackup [ -y ] [ -b | -n ]  [ -f filename | -t device_name ]`

Description The value of **-y** is do not prompt user to confirm backup.

The value of **-b** is back up billing files.

The value of **-n** is do not back up billing files.

The value of **-f filename** is back up to regular file named *filename*.

The value of **-t device_name** is back up to tape device named *device*.

Note Only one of the -b/-n or -f/-t flags can be used in the same command.

Example

```
$ bamsbackup

-b or -n parameter must be specified to indicate whether billing files
are to be backed up:

b)  Back up billing files
n)  Do not back up billing files
q)  Quit

Enter selection (b,n,q): b

-f or -t parameter must be specified to indicate backup to file or tape
device:

f)  Backup to file
t)  Backup to tape device
q)  Quit

Enter selection (f,t,q): f
Enter backup filename: /tmp/bamsbackup.12252004.cpio
Backup filename specified: /tmp/bamsbackup.12252004.cpio

**********************************************************************

The program /opt/CiscoBAMS/bin/bamsbackup will archive the following to
the file /tmp/bamsbackup.12252004.cpio:

1) Current BAMS configuration files
2) All BAMS executables
3) All BAMS intermediate files
4) All BAMS billing output files

Do you want to continue with bamsbackup (y/n)? y

/opt/CiscoBAMS/bin
/opt/CiscoBAMS/bin/ACC
/opt/CiscoBAMS/bin/ADMal
.
.
.
229280 blocks
Backup successful.

Program /opt/CiscoBAMS/bin/bamsbackup complete.  All files archived to
/tmp/bams
backup.12252004.cpio
```

Restore Utility

The bamsrestore utility restores BAMS Release 3.20 and later releases from any previously backed up files.

The restore utility prompts you to back up the current files and proceeds only after you confirm that you want to do so.

You can use the restore utility to restore files from a backup file or a tape device. The utility also provides you with an option to restore the billing output files.

Note Before you run the **bamsrestore** utility, check to ensure that the file ADMss is present in the directory /opt/CiscoBAMS/bin. The ADMss file must be present before you start this utility.

Format `bamsrestore [ -y ] [ -i | -k ] [ -b | -n ] [ -f filename | -t device_name ]`

Description The value of **-y** is confirm (that is, the user is not reminded to restore current files).

The value of **-i** is restore intermediate files (applicable only if you are restoring the same BAMS version).

The value of **-k** is keep intermediate files, do not restore intermediate files (applicable only if you are restoring the same BAMS version).

The value of **-b** is restore billing files.

The value of **-n** is do not restore billing files.

The value of **-f filename** is restore from regular file named *filename*.

The value of **-t device_name** is restore from tape device named *device*.

Note Only one of the -i/-k, -b/-n or -f/-t flags can be used on the command line. If the -y flag is not provided, the utility reminds you about backing up current files and prompts you to confirm that you wish to do so.

Example

```
$ bamsrestore

-b or -n parameter must be specified to determine whether billing files are
restored:

b)  Restore billing files
n)  Do not restore billing files
q)  Quit

Enter selection (b,n,q): b

-i or -k parameter must be specified to determine whether intermediate
files are restored:

i)  Restore intermediate files
k)  Keep current intermediate files.  Do not restore intermediate files q)
Quit

Enter selection (i,k,q): k

-f or -t parameter must be specified to indicate restore from file or tape
device:

f)  Restore from file
t)  Restore from tape device
q)  Quit

Enter selection (f,t,q): f
Enter restore filename: /tmp/bamsbackup.12252004.cpio
Restore filename: /tmp/bamsbackup.12252004.cpio

Current BAMS Version = 3.20
/opt/CiscoBAMS/bin/ADMss
229280 blocks
BAMS restore VERSION = 3.20
Current BAMS Version 3.20 is the same as the restore version of 3.20.

**********************************************************************

The program /opt/CiscoBAMS/bin/bamsrestore will restore the following from
the file /tmp/bamsbackup.12252004.cpio:

1) Current BAMS configuration files
2) All BAMS executables
3) All BAMS billing output files

* Current BAMS intermediate files will be kept.  Intermediate files
* will not be restored from /tmp/bamsbackup.12252004.cpio

Before restoring from /tmp/bamsbackup.12252004.cpio, the current BAMS files
should be backed up using the utility 'bamsbackup'.

Do you want to continue with bamsrestore (y/n)? y

Saving current intermediate files since restore from FILES_DIR will
overwrite these
/opt/CiscoBAMS/files/s01/acc_checkpt
/opt/CiscoBAMS/files/s01/cor_checkpt
/opt/CiscoBAMS/files/s02/acc_checkpt
20 blocks
Restoring files from /tmp/bamsbackup.12252004.cpio ...229280 blocks
Restore successful.
```

Example (cont.)

```
Restoring current intermediate files since these were temporarily
overwritten by the restore from FILES_DIR
/opt/CiscoBAMS/files/s01/acc_checkpt
/opt/CiscoBAMS/files/s01/cor_checkpt
.
.
.
20 blocks

For listing of files restored from /tmp/bamsbackup.12252004.cpio, see:
/tmp/bamsrestore.list.20041006143211

Program /opt/CiscoBAMS/bin/bamsrestore complete
```

Restoring a BAMS Release That Is the Same as the Current BAMS Release

The restore utility asks you whether you want to restore the intermediate files and billing files.

If you want to restore intermediate files, the restore utility removes the current intermediate files in the output directory and the checkpoint files, then restores intermediate files from the backup version.

If you do not want to restore intermediate files, then only executable and configuration files are restored. The current intermediate files are not touched.

Restoring a BAMS Release That Is Different from the Current BAMS Release

The restore utility removes the current intermediate files in the output directory and the checkpoint files, then restores intermediate files from the backup version.

Quality of Service Statistics Output

Overview

This chapter describes the Cisco BAMS ability to generate output files that include voice-gateway, Quality-of-Service (QoS) statistics produced by Cisco voice gateways, collected by the Cisco PGW 2200, and sent to the Cisco BAMS.

Note Not all calls produce the DSP and gateway statistics. If the required CDE tags are present in the CDB and the corresponding gateway statistics are present in the CDE tag, then the corresponding field will be populated in the QoS output.

Software Requirements

This section identifies the software releases required to support the production of Quality of Service (QoS) Statistics.

IOS for Cisco Voice Gateways

To support the generation of QoS statistics output files, a Cisco voice gateway must be running Cisco IOS Release 12.4(4)T or later.

Cisco PGW 2200

To support the generation of QoS statistics output files, a Cisco PGW 2200 must be running PGW 9.6(1) with the patch that includes the featurette: IOS DSP Stats in PGW - K factor and Other...

Cisco BAMS

The functionality to generate QoS statistics output files is included in a patch of Cisco BAMS named CSCOBAMS320QOS.pkg. This patch is added by issuing the following command:

```
pkgadd -d CSCOBAMS320QOS.pkg
```

Note Installing the QoS package adds an entry for file maintenance in the MSC_PARMS table for the QoS files.

By default, the patch enables QoS output on all nodes. You can determine whether QoS output is enabled on a BAMS node by issuing the following Unix command:

```
$ show_bams_qos
```

This command typically returns a response as shown in the following example:

```
Node 1: enabled
Node 2: enabled
Node 3: enabled
Node 4: enabled
Node 5: enabled
Node 6: enabled
Node 7: enabled
Node 8: enabled
```

The following Unix-level commands enable and disable the QoS output on an individual node:

```
$ enable_bams_qos
```

The following command would enable BAMS on a node numbered 2 (the **enable_bams_qos** command can be used to enable the feature on 1 to 8 BAMS nodes):

```
$ enable_bams_qos -s2
```

The following command would disable BAMS on a node numbered 2.

```
$ disable_bams_qos -s2
```

The QoS feed produces its own logs.which are located in respective node directories, such as /opt/CiscoBAMS/files/s0x.

QoS logs are not viewable in CLI as other logs are. QoS logs rollover and are named:

```
QOS.log
QOS.log.1
QOS.log.2
QOS.log.3
QOS.log.4
```

The default log level of LOG_NOTICE should be sufficient; however, the log levels can be set in the qos.cfg file, which is also located in the individual directory of each BAMS node, for example, /opt/CiscoBAMS/files/s0x.

The available log levels for QoS feeds are:

- LOG_NOTICE—For this log level, BAMS writes out only the filename that is being processed and archived, and nothing more. This is the log level for normal processing.
- LOG_DEBUG—This log level is for debug purposes only and should be set only when debugging.

Billing Interface

New gateway statistics output are generated from BAMS for the gateway statistics. Each record in the output corresponds to a finished call processed by BAMS.

Output File Location

The new output files are stored in the /opt/CiscoBAMS/data/s0X/QOS_STAT directory.

Output File Naming

A new gateway statistics output file is generated for each PGW CDR file. The generated file name is in the format QOS_STAT_YYYYMMDDhhmmss_XXXXXX.csv. In this format:

- YYYYMMDDhhmmss is date-time in UTC format contained in CDE tag 6001 of the corresponding CDR file header.
  - YYYY—year
  - MM—month
  - DD—day of the month
  - hh—hour
  - mm—minute
  - ss—second

  For example, 04/16/2005 13:45:27 UTC would generate 20050416134527.
- XXXXXX is the file sequence number, which is the same as the PGW file sequence number. For example, 000123.

File Content

Each file contains 0(zero) or more records. The records are separated by the new-line character (0xA). Cisco BAMS generates a record for each 1030 or 1040 CDB in the corresponding Cisco PGW CDR file. Each record contains fields separated by commas.

Table 0-1 describes the contents of the fields included in records generated for a QoS output file.

Table 0-1 QoS Output File—Record Fields

Field	Tag	Description	Format	PGW CDR CDE Tag
1	5000	Global Call Id	Text	5000
2	4002	Call Reference ID	Hex	4002
3	4106	First REL Timepoint ms	Seconds, milliseconds	4106
4	4107	Second REL Timepoint ms	Seconds, milliseconds	4107
5	4108	RLC Timepoint rcvd ms	Seconds, milliseconds	4108
6	4109	RLC Timepoint sent ms	Seconds, milliseconds	4109
7	P:PS	Ingress Packets sent	Integer	4046
8	P:PR	Ingress Packets received	Integer	4046
9	P:PL	Ingress Packets Lost	Integer	4046
10	P:OS	Ingress Octet Sent	Integer	4046
11	P:OR	Ingress Octet Received	Integer	4046
12	P:JI	Ingress Jitter	Integer	4046

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
13	P:LA	Ingress Latency	Integer	4046
14		Ingress Reservd1		4046
15		Ingress Reserved2		4046
16	P:PS	Egress Packets sent	Integer	4047
17	P:PR	Egress Packets received	Integer	4047
18	P:PL	Egress Packets Lost	Integer	4047
19	P:OS	Egress Octet Sent	Integer	4047
20	P:OR	Egress Octet Received	Integer	4047
21	P:JI	Egress Jitter	Integer	4047
22	P:LA	Egress Latency	Integer	4047
23		Egress Reservd1		4047
24		Egress Reserved2		4047
25	DSP/TX:PK	Ingress tx packets	Integer	4098
26	DSP/TX: SG	Ingress signalling packets	Integer	4098
27	DSP/TX: NS	Ingress noise packets	Integer	4098
28	DSP/TX: DU	Ingress tx duration	Integer	4098
29	DSP/TX: VO	Ingress voice tx duration	Integer	4098
30	DSP/RX: PK	Ingress voice packets	Integer	4098
31	DSP/RX: SG	Ingress signalling packets	Integer	4098
32	DSP/RX: RX	Ingress comfort noise packets		4098
33	DSP/RX: VO	Ingress rx duration		4098
34	DSP/RX: BS	Ingress voice rx duration		4098
35	DSP/RX: BP	Ingress bad sequence		4098
36	DSP/RX: LP	Ingress bad protocol		4098
37	DSP/RX: EP	Ingress late packets		4098
38	DSP/PD: CU	Ingress early packets		4098
39	DSP/PD: MI	Ingress playout delay current		4098
40	DSP/PD: MA	Ingress playout delay min		4098
41	DSP/PD: CO	Ingress playout delay max		4098
42	DSP/PD: IJ	Ingress playout delay clock_offset		4098
43	DSP/PE: PC	Ingress playout delay interarrival jitter		4098
44	DSP/PE: IC	Ingress playout error predictive concealment		4098

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
45	DSP/PE: SC	Ingress playout error interpolative concealment		4098
46	DSP/PE: RM	Ingress playout error silence concealment		4098
47	DSP/PE: BO	Ingress playout error retroactive mem update		4098
48	DSP/PE: EE	Ingress playout error buffer overflow		4098
49	DSP/LE: TP	Ingress playout error talkspurt end point error		4098
50	DSP/LE: TX	Ingress Level tx power in 0.1 dBm		4098
51	DSP/LE: RP	Ingress Level tx mean in 0.1 dBm		4098
52	DSP/LE: RM	Ingress Level rx power in 0.1 dBm		4098
53	DSP/LE: BN	Ingress Level rx mean in 0.1 dBm		4098
54	DSP/LE: ER	Ingress Level background noise		4098
55	DSP/LE: AC	Ingress Level erl level		4098
56	DSP/LE: TA	Ingress Level acom level		4098
57	DSP/LE: RA	Ingress Level curr tx act		4098
58	DSP/ER: RD	Ingress Level curr rx act		4098
59	DSP/ER: TD	Ingress error_stats rx dropped		4098
60	DSP/ER: RC	Ingress error_stats tx dropped		4098
61	DSP/ER: TC	Ingress error_stats rx control		4098
62	DSP/IC: IC	Ingress error_stats tx control		4098
63	DSP/EC:CI	Ingress ICPIF value for measuring voice quality		4098
64	DSP/EC:FM	Ingress Codec ID	Text	4098
65	DSP/EC:FP	Ingress Frame size in ms	Integer	4098
66	DSP/EC:VS	Ingress Frames per packet	Integer	4098
67	DSP/EC:GT	Ingress VAD enabled flag	Integer	4098
68	DSP/EC:GR	Ingress TX Gain (linear)	Integer	4098
69	DSP/EC:JD	Ingress RX Gain (linear)	Integer	4098
70	DSP/EC:JN	Ingress Jitter Buffer Mode	Integer	4098

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
71	DSP/EC:JM	Ingress Jitter buffer nominal playout delay	Integer	4098
72	DSP/EC:JX	Ingress Jitter buffer minimum playout delay	Integer	4098
73	DSP/KF:KF	Ingress Jitter buffer max playout delay	Integer	4098
74	DSP/KF:AV	Ingress K-factor MOS-k (inst)	Integer	4098
75	DSP/KF:MI	Ingress Average MOS-k	Integer	4098
76	DSP/KF:MI	Ingress Minimum MOS-k	Integer	4098
77	DSP/KF:BS	Ingress Baseline MOS-k (Max)	Integer	4098
78	DSP/KF:NB	Ingress Number of bursts	Integer	4098
79	DSP/KF:FL	Ingress Average frame loss rate	Integer	4098
80	DSP/KF:NW	Ingress Number of windows in average MOS	Integer	4098
81	DSP/KF:VR	Ingress MOS K-factor Version ID	Integer	4098
82	DSP/CS:CR	Ingress Conceal Ratio (instantaneous)	Integer	4098
83	DSP/CS:AV	Ingress Average CR	Integer	4098
84	DSP/CS:MX	Ingress Maximum CR	Integer	4098
85	DSP/CS:CT	Ingress Concealment Time	Integer	4098
86	DSP/CS:TT	Ingress Total time (duration)	Integer	4098
87	DSP/CS:OK	Ingress OK seconds	Integer	4098
88	DSP/CS:CS	Ingress Concealed seconds	Integer	4098
89	DSP/CS:SC	Ingress Severely concealed seconds	Integer	4098
90	DSP/CS:TS	Ingress Conceal threshold	Integer	4098
91	DSP/CS:DC	Ingress Dead connection indication	Integer	4098
92	DSP/RF:ML	Ingress R-factor MOS-LQE	Integer	4098
93	DSP/RF:MC	Ingress R-factor MOS-CQE	Integer	4098
94	DSP/RF:R1	Ingress R-factor LQ profile 1	Integer	4098
95	DSP/RF:R2	Ingress R-factor LQ profile 2	Integer	4098
96	DSP/RF:IF	Ingress Ie_eff	Integer	4098
97	DSP/RF:ID	Ingress Idd	Integer	4098

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
98	DSP/RF:IE	Ingress Codec baseline IE score	Integer	4098
99	DSP/RF:BL	Ingress Codec baseline BPL	Integer	4098
100	DSP/RF:R0	Ingress R0 default	Integer	4098
101	DSP/RF:VR	Ingress R-factor Version ID	Integer	4098
102	DSP/UC:U1	Ingress User conceal seconds 1 count (UCS1)	Integer	4098
103	DSP/UC:U2	Ingress User conceal seconds 2 count (UCS2)	Integer	4098
104	DSP/UC:T1	Ingress UCS1 threshold in ms	Integer	4098
105	DSP/UC:T2	Ingress UCS2 threshold in ms	Integer	4098
106	DSP/DL:RT	Ingress Round trip delay	Integer	4098
107	DSP/DL:ED	Ingress End system delay	Integer	4098
108	DSP/TX:PK	Egress tx packets	Integer	4099
109	DSP/TX: SG	Egress signalling packets	Integer	4099
110	DSP/TX: NS	Egress noise packets	Integer	4099
111	DSP/TX: DU	Egress tx duration	Integer	4099
112	DSP/TX: VO	Egress voice tx duration	Integer	4099
113	DSP/RX: PK	Egress voice packets	Integer	4099
114	DSP/RX: SG	Egress signalling packets	Integer	4099
115	DSP/RX: CF	Egress comfort noise packets	Integer	4099
116	DSP/RX: RX	Egress rx duration	Integer	4099
117	DSP/RX: VO	Egress voice rx duration	Integer	4099
118	DSP/RX: BS	Egress bad sequence	Integer	4099
119	DSP/RX: BP	Egress bad protocol	Integer	4099
120	DSP/RX: LP	Egress late packets	Integer	4099
121	DSP/RX: EP	Egress early packets	Integer	4099
122	DSP/PD: CU	Egress playout delay current	Integer	4099
123	DSP/PD: MI	Egress playout delay min	Integer	4099
124	DSP/PD: MA	Egress playout delay max	Integer	4099
125	DSP/PD: CO	Egress playout delay clock_offset	Integer	4099
126	DSP/PD: IJ	Egress playout delay interarrival jitter	Integer	4099
127	DSP/PE: PC	Egress playout error predictive concealment	Integer	4099

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
128	DSP/PE: IC	Egress playout error interpolative concealment	Integer	4099
129	DSP/PE: SC	Egress playout error silence concealment	Integer	4099
130	DSP/PE: RM	Egress playout error retroactive mem update	Integer	4099
131	DSP/PE: BO	Egress playout error buffer overflow	Integer	4099
132	DSP/PE: EE	Egress playout error talkspurt end point error	Integer	4099
133	DSP/LE: TP	Egress Level tx power in units of 0.1 dBm	Integer	4099
134	DSP/LE: TX	Egress Level tx mean in units of 0.1 dBm	Integer	4099
135	DSP/LE: RP	Egress Level rx power in units of 0.1 dBm	Integer	4099
136	DSP/LE: RM	Egress Level rx mean in units of 0.1 dBm	Integer	4099
137	DSP/LE: BN	Egress Level background noise	Integer	4099
138	DSP/LE: ER	Egress Level erl level	Integer	4099
139	DSP/LE: AC	Egress Level acom level	Integer	4099
140	DSP/LE: TA	Egress Level curr tx act	Integer	4099
141	DSP/LE: RA	Egress Level curr rx act	Integer	4099
142	DSP/ER: RD	Egress error_stats rx dropped	Integer	4099
143	DSP/ER: TD	Egress error_stats tx dropped	Integer	4099
144	DSP/ER: RC	Egress error_stats rx control	Integer	4099
145	DSP/ER: TC	Egress error_stats tx control	Integer	4099
146	DSP/IC: IC	Egress ICPIF value for measuring voice quality	Integer	4099
147	DSP/EC:CI	Egress Codec ID	Integer	4099
148	DSP/EC:FM	Egress Frame size in ms	Integer	4099
149	DSP/EC:FP	Egress Frames per packet	Integer	4099
150	DSP/EC:VS	Egress VAD enabled flag	Integer	4099
151	DSP/EC:GT	Egress TX Gain (linear)	Integer	4099
152	DSP/EC:GR	Egress RX Gain (linear)	Integer	4099
153	DSP/EC:JD	Egress Jitter Buffer Mode	Integer	4099

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
154	DSP/EC:JN	Egress Jitter buffer nominal playout delay	Integer	4099
155	DSP/EC:JM	Egress Jitter buffer minimum playout delay	Integer	4099
156	DSP/EC:JX	Egress Jitter buffer max playout delay	Integer	4099
157	DSP/KF:KF	Egress K-factor MOS-k (inst)	Integer	4099
158	DSP/KF:AV	Egress Average MOS-k	Integer	4099
159	DSP/KF:MI	Egress Minimum MOS-k	Integer	4099
160	DSP/KF:BS	Egress Baseline MOS-k (Max)	Integer	4099
161	DSP/KF:NB	Egress Number of bursts	Integer	4099
162	DSP/KF:FL	Egress Average frame loss rate	Integer	4099
163	DSP/KF:NW	Egress Number of windows in average MOS	Integer	4099
164	DSP/KF:VR	Egress MOS K-factor Version ID	Integer	4099
165	DSP/CS:CR	Egress Conceal Ratio (instantaneous)	Integer	4099
166	DSP/CS:AV	Egress Average CR	Integer	4099
167	DSP/CS:MX	Egress Maximum CR	Integer	4099
168	DSP/CS:CT	Egress Concealment Time	Integer	4099
169	DSP/CS:TT	Egress Total time (duration)	Integer	4099
170	DSP/CS:OK	Egress OK seconds	Integer	4099
171	DSP/CS:CS	Egress Concealed seconds	Integer	4099
172	DSP/CS:SC	Egress Severely concealed seconds	Integer	4099
173	DSP/CS:TS	Egress Conceal threshold	Integer	4099
174	DSP/CS:DC	Egress Dead connection indication	Integer	4099
175	DSP/RF:ML	Egress R-factor MOS-LQE	Integer	4099
176	DSP/RF:MC	Egress R-factor MOS-CQE	Integer	4099
177	DSP/RF:R1	Egress R-factor LQ profile 1	Integer	4099
178	DSP/RF:R2	Egress R-factor LQ profile 2	Integer	4099
179	DSP/RF:IF	Egress Ie_eff	Integer	4099
180	DSP/RF:ID	Egress Idd	Integer	4099

Table 0-1 QoS Output File—Record Fields (continued)

Field	Tag	Description	Format	PGW CDR CDE Tag
181	DSP/RF:IE	Egress Codec baseline IE score	Integer	4099
182	DSP/RF:BL	Egress Codec baseline BPL	Integer	4099
183	DSP/RF:R0	Egress R0 default	Integer	4099
184	DSP/RF:VR	Egress R-factor Version ID	Integer	4099
185	DSP/UC:U1	Egress User conceal seconds 1 count (UCS1)	Integer	4099
186	DSP/UC:U2	Egress User conceal seconds 2 count (UCS2)	Integer	4099
187	DSP/UC:T1	Egress UCS1 threshold in ms	Integer	4099
188	DSP/UC:T2	Egress UCS2 threshold in ms	Integer	4099
189	DSP/DL:RT	Egress Round trip delay	Integer	4099
190	DSP/DL:ED	Egress End system delay	Integer	4099
191	4087	Egress MGCP DLCX Return Code	Hex	4087
192	4088	Egress MGCP DLCX Return Code	Hex	4088
193	4205	Egress Media Device Address	String	4205
194	4206	Egress Media Device Address	String	4206
195	4207	Initial Codec	String	4207
196	4208	Final Codec	String	4208
197	4209	Egress Media Device Port	String	4209
198	4210	Egress Media Device Port	String	4210
199	4227	Route Optimization/Path Replacement Action	Hex	4227
200	4228	Route Optimization/Path Replacement Call Reference	Hex	4228
201	4229	Route Optimization/Path Replacement Trunk Group Info	Hex	4229
202	4230	Route Optimization/Path Replacement Channel Info	Hex	4230
203	4231	Route Optimization Switchover Timestamp	Hex	4231

A

AMA Automatic Message Accounting.

APC adjacent point code.

ASCII American Standard Code for Information Interchange.

B

BAF Bellcore AMA Format.

BAM billing and measurements.

BAMS Billing and Measurements Server. Performs CDR mediation and generates operational measurements derived from the call records. Converts the TLV (binary) CDRs produced on the PGW host to industry standard formats. More specifically, a standalone (simplex) or redundant pair of Sun host machines that Gateway TLV records via FTP from Cisco PGW nodes for post-processing into BAF output records using internal flat file tables and appropriate filtering, formatting, and CDR-to-BAF conversion routines.

C

CC call control.

CDB call detail block.

CDR call detail record.

Cisco MGC software Cisco Media Gateway Controller (MGC) software. The generic name given to the Cisco software application that performs signaling and call control as well as the network functions expected from a PSTN switching point (SP). It terminates MTP3 and higher layers of the SS7/C7 protocol stack.

Cisco PGW 2200 PSTN Gateway A collection of PGW hosts that collectively provide distributed call control (CC) and signaling services.

Cisco SLT Cisco Signaling Link Terminal. An SLT provides physical connection to the SS7 network. It terminates MTP Layer 1 and 2 and backhauls MTP 3 and higher layers to the PGW hosts over the signaling (IP) network for call. Two or more SLTs are deployed in load-sharing mode.

CLEC competitive local exchange carrier. Authorized by the Telecommunications Act of 1996, the creation of CLECs effectively constituted Phase II of the deregulation of the telecommunications market in the United States. Phase I was characterized by the divestiture of AT&T with the Judge Green decision of 1984. This decision led to the creation of the Baby Bells and enabled stronger competition in the long-distance market by MCI and Sprint. Phase II targeted the "local loop" and the Local Exchange Carriers (Bells and others) and led to the formation of startup companies with new investment capital. Some CLECs chose to resell excess capacity from the traditional carriers, others to create new plant, such as SONET ring fiber networks for business customers.

competitive local exchange carrier See CLEC.

D

DPC Destination Point Code.

DS0 digital service 0. A 64-kbps digital TDM channel used for carrying a single POTS call.

DS3 digital service 3. A 45-Mbps digital link.

E

EGR (egr) egress: outgoing or terminating.

EMS Element Management System. Provides element management for the Cisco PGW 2200. The Cisco Voice Services Provisioning Tool (Cisco VSPT) enables bulk provisioning and Cisco MGC Node Manager (Cisco MNM) provides fault and performance management.

F

FTP File Transfer Protocol.

G

GW gateway. *See also* MGW.

H

HSI H.323 Signaling Interface. It enables the Cisco PGW 2200 (in Call Control mode) to act as an H.323 end point.

I

IAM Initial Address Message.

IC Interexchange Carrier.

IGR (igr) ingress: incoming or originating.

ILEC incumbent local exchange carrier.

IMT Inter-Machine Trunks.

IP Internet Protocol.

IP mediation A collection of Internet network usage records for billing purposes, such as real-time streaming video and voice/IP.

IP telephony Telephony services provided over the Internet.

ISDN Integrated Services Digital Network. ISDN services are gradually replacing POTS in affluent communities. Competing against ISDN are asynchronous digital subscriber lines and multiservice wireless applications.

ISP Internet service provider.

L

LAN local-area network.

LAN switch Provides IP connectivity between all the elements of the signaling network. This signaling network is also referred to as the Control Signaling Network.

LERG local exchange routing guide.

M

Man Machine Language *See* MML.

media gateway *See* MGW.

Media Gateway Control Protocol *See* MGCP.

MGC software Media Gateway Controller software. A generic term used for the Cisco PGW 2200 PSTN Gateway software when both call control and signaling applications apply.

MGCP Media Gateway Control Protocol. A protocol based on a merging of the IPDC and SGCP protocols.

MGH media gateway host. A device that physically contains the media gateway (that is, MGX is a media gateway host, VISM is the corresponding media gateway). MGH and MG are sometimes used interchangeably, depending on the context and device being referenced.

MGW media gateway. A generic term used for the gateway between a QoS packet network and the PSTN/ISDN. There are three types of gateways: the trunking gateway, the access gateway, and the network access server or nonvoice gateway. Media gateways are termination points for the Media Gateway Control Protocol. Media gateways terminate MGCP links.

MML The Man Machine Language used in a Cisco MGC software.

N

NAS network access server.

NNL Node Name Link.

NICS Non-Intercompany Settlement System.

NPA Numbering Plan Area.

O

OPC Own Point Code.

P

PGW See Cisco PGW 2200 PSTN Gateway.

PGW Host The Sun computing platform on which the MGC software application resides, providing call control or switching functionality. Each host controls a unique subset of the media gateways, that is, trunking gateways, access gateways, network access servers. PGW hosts are deployed in pairs for redundancy. Fault tolerance is achieved by checkpointing call context information between active and standby hosts. All stable calls are preserved in case of a switchover from one host to the other.

PIC Primary Interexchange Carrier.

POTS plain old telephone service. A term used to distinguish traditional human-to-human telephony services from other uses of the telephone circuits, for example, computer-to-computer data transmission over modem.

PRI Primary Rate Interface.

PSTN public switched telephone network.

Q

QoS packet network An IP or ATM bearer network for voice/data and signaling traffic transport through the virtual switch. The Control Signaling Network runs over the QoS Packet Network along with the bearer traffic.

S

SCP Service Control Point.

Signaling Link Terminal *See* Cisco SLT.

SLT Signaling Link Terminal. *See* Cisco SLT.

SNMP Simple Network Management Protocol.

SS7 Signaling System#7. This signaling system is gradually replacing older systems globally.

STP Signal Transfer Point.

T

TAG User identifier.

TDM time-division multiplexing. In PSTN digital circuits, signals are multiplexed through the use of time slices, rather than different frequencies, phase shifts, or codes (FDM, PSK, CDM).

Time-division multiplexing See TDM.

TLV Tag Length Value.

TRK trunk.

TTL (ttl) total.

U

UTC Universal Time Coordinated.

W

WAN wide-area network.

Numerics

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

Z