Cisco Broadband Troubleshooter Release 3.4 User Guide

Cisco IOS Release 12.2SC
June 2010

Americas 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: 408 527-0883

Text Part Number: OL-22619-01

Cisco Broadband Troubleshooter Release 3.4 User Guide

CONTENTS

Preface ix

Overview ix

Document Revision History ix

Audience ix

Related Documentation x

Obtaining Documentation and Submitting a Service Request x

CHAPTER 1 **Product Overview** 1-1

System Requirements 1-2

Windows Systems 1-3

Linux Operating System 1-3

UNIX Server Systems with Solaris 1-3

UNIX Server Systems with Linux 1-4

PC Server Systems for Windows 1-4

UNIX Client Machine with Solaris 1-4

UNIX Client Machine with Linux 1-4

PC Client Machine with Windows 1-4

Minimum Cisco Universal Broadband Router and Cisco IOS Requirements 1-5

Enabling SNMP on Routers 1-5

Using SNMP MIBs in Cisco CBT 3.4 1-5

CHAPTER 2 **Installing, Downloading, and Upgrading CBT 3.4** 2-1

Using TCP Ports for CBT 3.4 Upgrade and Operation 2-1

CBT 3.4 Download and Upgrade 2-2

Installing and Starting CiscoView on Solaris 2-4

Installing and Starting CBT on Windows 2-5

Installing and Starting CBT on Linux 2-6

Uninstalling CBT 3.4 2-6

Uninstalling CBT on Solaris 2-6

Uninstalling CBT on Windows 2-7

Uninstalling CBT on Linux 2-7

CHAPTER 3 **Configuring the Cisco Broadband Troubleshooter** 3-1

Configuring the Cisco CMTS to Use CBT Spectrum Management Tools 3-1

Configuring the Java Runtime Environment 3-2
JRE 5 OS-Level Requirements 3-2
Installing JRE 5.0 in Microsoft Internet Explorer on the Windows Platform 3-2
Installing JRE 5.0 in Netscape Navigator on the Windows Platform 3-2
Installing JRE 5.0 in Netscape Navigator on the Solaris Platform 3-2
Installing JRE 5.0 in Netscape Navigator on the Linux Platform 3-3
Configuring CiscoView (Optional) 3-4
Using Subscriber Traffic Management with CBT 3.4 (Optional) 3-4
Reviewing the CBT Task Menu 3-5
User Guide 3-6
Configuration 3-7
Diagnostics 3-7
Hotline Tools 3-7
Spectrum Tools 3-7
STM Tools 3-8
Utilities 3-8
User Log Out 3-8
User Types in CBT 3-9
Workflow of Administrator and RF Technician Tasks 3-10
Administrator Tasks 3-10
RF Tech Tasks 3-11
Setting Up CBT 3-11
Changing the Administrator Password 3-12
Disabling or Enabling Password Text Display 3-12
Adding Users 3-12
Adding Router Information 3-14
Adding Subscriber Information 3-16
Adding CMTS Group 3-17
Setting Parameters in .INI Files 3-18
Parameters in the CONFIGS.INI File 3-19
Phone Number Format 3-19
Modem Status Thresholds 3-19
Southbound Interface Settings 3-20
Modem State Settings 3-20
Query Timeout 3-20
Parameters in the POLLER.INI File 3-21
Parameters in the SPECTRUM.INI File 3-22
Parameters in the GUNSLINGER.INI File 3-24
Retrieving Subscriber or Provisioning Data by Using an External Interface 3-24

Retrieving Data with a Script 3-25
Script Parameters for MAC Address Information 3-26
Script Parameters for Subscriber Information 3-26
Script Parameters for Provisioning Information 3-27
Error Handling for Script 3-27
Retrieving Data with a Script 3-27
Troubleshooting Script Problems 3-27
Retrieving Data with an Application on an HTTP Server 3-28
HTTP Parameters for MAC Address Information 3-28
HTTP Parameters for Subscriber Information 3-28
HTTP Parameters for Provisioning Information 3-29
Error Handling for HTTP 3-29
Retrieving Data with an HTTP Application 3-29
Troubleshooting HTTP Problems 3-30
Retrieving Data with LDAP 3-30
Retrieving Data with BPR 3-30
Retrieving Data with CNR 3-31
Retrieving Subscriber Data from the Local Database 3-32
Retrieving Provisioning Data from the Local Database 3-33
About the Poller 3-33
Scheduling the Poller 3-34
Getting Summary Information and a Detailed Real-Time Status Report for a Modem 3-35
Getting Summary Information on a Modem 3-37
Getting a Detailed Real-Time Status Report for a Modem 3-38
Reviewing the CMTS Dashboard 3-39
Showing and Configuring the Flap List Analysis 3-40
Showing the Flap List Analysis 3-40
Scheduling the Flap List Analysis 3-42
Setting Parameters for the Flap List 3-43
Purging the Flap List 3-44
Scheduling the Spectrum Data 3-46
Scheduling the STM Data 3-47
Generic Data Query 3-47

CHAPTER 4 **Using the Spectrum Management Tools** 4-1
Using the Trace Window 4-2
Working with the Trace Window 4-3
Starting the Trace Window 4-3
Viewing Trace Windows 4-5

Sorting Support in the CBT 3.4 Graphical User Interface 4-6
Using Auto-Select in the Trace Window 4-6
Using the Spectrogram 4-7
Starting the Spectrogram 4-7
Analyzing the Carrier-to-Noise Ratio 4-9
Getting the CNR Analysis 4-10
Retrieving CNR Trending Data 4-11
Playing Back Data 4-12
Playing Back Trace Window 4-13
Playing Back CNR Analysis 4-15
Playing Back CNR Trending Data 4-17
Displaying Polling Status 4-19
Real Time Polling Status 4-19
Scheduled Polling Status 4-19
Enabling Instantaneous CPU Assessment for Spectrum Polling 4-20

CHAPTER 5 **Subscriber Traffic Management in CBT 3.4** 5-1
Using the STM Tools 5-1
Displaying STM Enforce Rules Defined for a Specific Period 5-1
Displaying Active Enforce Rules 5-2
Displaying Violators Detected for a Specific Period 5-2
Displaying Current Violators 5-2
Locating Violators 5-3

CHAPTER 6 **Administrative Tasks for CBT 3.4** 6-1
Mapping Hostnames from Applet to Servlet 6-1
Configuring Administrative Parameters for Spectrum Analysis 6-2

CHAPTER 7 **Troubleshooting CBT 3.4** 7-1
Troubleshooting Continuous Sweep Spectrum Operation in CBT 3.4 7-1
Changing Server Ports in XML Script 7-2
Saving System Message Logs for Troubleshooting 7-2
Verifying Installation Status on the CBT 3.4 Server 7-3
Verifying the CBT 3.4 License 7-4
Troubleshooting the Poller 7-4
Troubleshooting Poller Problems 7-4
Error Message: No Router 7-5
Error Message: Database Login Failed 7-5

Error Message: SNMP Timeout 7-6

CHAPTER 8 **Sample Code for Application Program Interfaces 8-1**

Shell Script for Retrieving Subscriber or Provisioning Information 8-1

Purpose of Shell Script 8-2

Sample Script Requirements 8-2

Script and File Descriptions 8-2

Modifying the Script 8-2

Troubleshooting the Script 8-3

Sample Script Code 8-4

Java Code for Retrieving Subscriber or Provisioning Information 8-4

Purpose of Java Code 8-5

Sample HTTP Application Requirements 8-5

Module and File Descriptions 8-5

Modifiable Components in the Sample HTTP Application 8-6

Building and Running the Sample HTTP Application 8-7

Troubleshooting the Sample HTTP Application 8-8

Sample Java Code 8-8

INDEX

Preface

Overview

This guide describes the Cisco Broadband Troubleshooter Release 3.4 (CBT 3.4) network management tool, supporting Cisco universal broadband routers and Cisco Cable Modem Termination System (CMTS) headend deployments. This user guide provides a description of CBT 3.4 features, enhancements, configurations, and troubleshooting. This document does not describe CBT releases prior to CBT 3.4.This Preface provides an overview of this guide with the following sections:

- Document Revision History, page ix
- Audience, page ix
- Related Documentation, page x
- Obtaining Documentation and Submitting a Service Request, page x

Document Revision History

The Document Revision History table below records technical changes to this document.

Document Revision	Date	Change summary
OL-22619-01	June 21, 2010	New document published to describe CBT 3.4.

Audience

This guide is intended for system and network administrators who must configure the router for operation and monitor its performance in the network.

This guide may also be useful for application developers who are developing management applications for the router.

Related Documentation

In addition to this document, the following documents and resources provide additional information related to CBT 3.4.

- *Release Notes for Cisco Broadband Troubleshooter Release 3.4*

  http://www.cisco.com/en/US/docs/net_mgmt/cisco_broadband_troubleshooter/3.4/release/notes/CBT34ReleaseNotes.html

- Cisco CBT 3.4 Online Help—available with licensed installation of CBT 3.4

The following documents are available to describe the Cisco IOS releases and routers supported by CBT 3.4:

- *Cross-Platform Release Notes for Cisco Universal Broadband Routers in Cisco IOS Release 12.2SC*

  http://www.cisco.com/en/US/docs/cable/cmts/release/notes/12_2sc/122sc_cmts_rn.html

- *Caveats for the Cisco uBR10012 Universal Broadband Router in Cisco IOS Release 12.2SC*

  http://www.cisco.com/en/US/docs/cable/cmts/release/notes/12_2sc/122sc_cmts_ub10k_caveats.html

- Caveats for the Cisco uBR7200 Series Universal Broadband Routers in Cisco IOS Release 12.2SC

  http://www.cisco.com/en/US/docs/cable/cmts/release/notes/12_2sc/122sc_cmts_ub72k_caveats.html

The following document is available to describe Upstream Channel Bonding on the Cisco CMTS:

http://www.cisco.com/en/US/docs/ios/cable/configuration/guide/ubr_uscb.html

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

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

Refer also to *Cisco CMTS Universal Broadband Router MIB Specifications Guide 12.2SC* document:

http://www.cisco.com/en/US/docs/cable/cmts/mib/reference/guide/ubrmibv5.html

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

CHAPTER 1

Product Overview

Multiple service operators (MSOs) and cable companies provide subscribers with a variety of cable services such as TV, video-on-demand, data, and voice telephony.
Cisco Broadband Troubleshooter (CBT) Release 3.4 is a troubleshooting tool designed for network administrators and RF technicians to monitor and resolve RF problems at the MSO cable plant.

This document describes configuration and operation procedures for CBT 3.4 with updates to cover newly supported operating systems, feature functions, MIB interoperability, and GUI enhancements.

For itemized information and features introduced in CBT 3.4, refer to the *Release Notes for Cisco Broadband Troubleshooter Release 3.4*.

CBT 3.4 provides the following general functionality:

- Provides summary statistics for each upstream port attached to a Cisco cable modem termination system (CMTS).
- Analyzes data captured from a Cisco CMTS and sorts problem modems into the following categories:
  - Provisioning problems
  - Reverse path noise problems
  - Reverse path attenuation problems
  - Packet corruption problems, showing the cyclic redundancy check (CRC) error value
- Shows the number of flapping modems and provides summary statistics on each one.
- Provides the following Subscriber Traffic Management (STM) tools:
  - Real-time display of STM 1.0 violators
  - Historical display of STM 1.0 violators along with the enforce-rules violated
  - Search for violators based on MAC address
- Provides the following spectrum management tools so you can analyze data without using a spectrum analyzer:
  - Trace Window
  - Spectrogram
  - Carrier-to-noise ratio (CNR) Analysis
  - CNR Trending
  - Scheduling and threshold based data polling

  - Generic trending that gets MIB variable data from the Cisco CMTS
- Locates the cable modem of a subscriber with wildcard searches for any of the following identifiers:
  - MAC address
  - Phone number
  - Account ID
  - Name
  - Street address
  - Zip code
- Shows detailed information for a cable modem in a real-time status report, with information from the following sources:
  - Cisco CMTS
  - Cable modem itself
  - Subscriber database of the MSO
- Provides several, flexible external interface options to integrate provisioning and subscriber information.
- Introduces support for Upstream Channel Bonding on the Cisco CMTS. For additional information, see the "Related Documentation" section.
- Provides support for Cisco UBR-MC20X20V and Cisco uBR-MC88V cable interface line cards.

CBT 3.4 can be installed on a server running Windows Vista Ultimate, Red Hat Enterprise Linux 5 Server, or Solaris 10 operating systems.

System Requirements

This section lists the system requirements for:

- Each supported operating system (OS):
  - Server
  - Client
- The number of Cisco CMTS headend systems to be supported with CBT 3.4
- Cisco uBR7200 series and Cisco uBR10012 series universal broadband routers

CBT 3.4 generally supports the following OS platforms:

- Sun Fire V440 (for 500 CMTS support) with Solaris 10
- Sun Fire V240, V210 and V100 with Solaris 10
- Windows Box (Windows 2K, XP, or, Windows Vista Ultimate)
- Windows XP Workstation with Linux Enterprise Edition or Red Hat Enterprise Linux Server release 5

This section provides additional system-level information in these topics:

- Windows Systems, page 1-3
- Linux Operating System, page 1-3
- UNIX Server Systems with Solaris, page 1-3

- UNIX Server Systems with Linux, page 1-4
- PC Server Systems for Windows, page 1-4
- UNIX Client Machine with Solaris, page 1-4
- UNIX Client Machine with Linux, page 1-4
- PC Client Machine with Windows, page 1-4
- Minimum Cisco Universal Broadband Router and Cisco IOS Requirements, page 1-5

Windows Systems

Windows systems supported include Windows Vista Ultimate, Windows 2000, Windows XP, and Windows XP Workstation with Linux Enterprise Edition.

Linux Operating System

Linux systems supported include Linux Red Hat Enterprise Linux 5 Server. We recommend upgrading from prior Linux versions. You must use Sybase10 on the Linux operating system.

UNIX Server Systems with Solaris

The following environment supports **10 Cisco CMTS headend systems**:

- UNIX Server Sun Fire Systems with Solaris 10
- Sun Fire V100 server: small
- UltraSPARC-IIi processor: one at 550 MHz
- Memory: 256 MB (one 256-MB DIMM)
- 7200 RPM IDE disk drive: one at 80 GB

The following environment supports **50 or 100 Cisco CMTS headend systems**:

- Sun Fire V210 server: Medium
- UltraSPARC IIIi Cu processor: two at 1.34 GHz
- Layer 2 cache per processor: 1 MB
- Memory: 2 GB (four 512-MB DIMMS)
- 10000 RPM Ultra 3 SCSI LVD disk drive: two at 73 GB

The following environment supports **500 Cisco CMTS headend systems**:

- Sun Fire V440 server: Small
- UltraSPARC IIIi Processor: two at 1.593 GHz
- Internal cache per processor: 1 MB
- Memory: 4 GB (eight 12-MB DIMMS)
- 10000 RPM Ultra320 SCSI disk drive: four at 73 GB

UNIX Server Systems with Linux

The following environment supports **10, 50, or 100 Cisco CMTS headend systems**:

- Linux Red Hat Enterprise Edition 5 workstation or Linux Red Hat Enterprise Edition workstation
- Pentium 4, 3.2 gigahertz (GHz) processor or higher
- 1 GB DRAM
- 20 GB disk space

PC Server Systems for Windows

The following environment supports **10, 50, or 100 Cisco CMTS headend systems**:

- Windows Vista, Windows 2000, or Windows XP workstation
- Pentium 4, 3.2 gigahertz (GHz) or higher processor
- 1 GB RAM
- 20 GB disk space

UNIX Client Machine with Solaris

The following client system supports CBT 3.4 on Solaris:

- Sun Fire V100 server: Small
- UltraSPARC-IIi processor: one at 550 MHz
- Memory: 256 MB (one 256-MB DIMM)

UNIX Client Machine with Linux

The following client system supports CBT 3.4 with Linux:

- Pentium 4, 3.2 gigahertz (GHz) or higher processor
- 1 GB RAM
- Red Hat 10 or 9

PC Client Machine with Windows

The following client system supports CBT 3.4 with Windows:

- Windows Vista, Windows 2000, or Windows XP workstation
- 1 GB RAM
- Pentium 4, at 3.2 GHz or higher

Minimum Cisco Universal Broadband Router and Cisco IOS Requirements

You can use CBT 3.4 with the following universal broadband routers (uBRs):

- Cisco uBR7200 series universal broadband router
- Cisco uBR10012 universal broadband router

On these routers, CBT 3.4 supports the following Cisco IOS releases:

- Cisco IOS Release 12.2(33)SCC
- Cisco IOS Release 12.2(33)SCD
- Cisco IOS Release 12.2(33)SCD2

Enabling SNMP on Routers

CBT retrieves the Cisco IOS release number, router name, and router type using SNMP. You must enable SNMP on the routers using the following commands:

```
Router# configure terminal
Router(config)# snmp-server community public RO
Router(config)# snmp-server community private RW
```

Using SNMP MIBs in Cisco CBT 3.4

CBT3.4 uses the following SNMP MIBs with Cisco IOS Release 12.2(33)SCD2:

- DOCS-IF3-MIB.my
- RFC1213-MIB
- IANAifType-MIB
- IF-MIB
- SNMPv2-MIB
- SNMPv2-TC
- CISCO-SMI
- OLD-CISCO-CHASSIS-MIB
- CISCO-PRODUCTS-MIB
  - DOCS-IF-MIB
  - DOCS-IF-EXT-MIB
  - CISCO-CABLE-SPECTRUM-MIB
  - CISCO-DOCS-EXT-MIB
- CISCO-PING-MIB
- INET-ADDRESS-MIB
- SNMP-FRAMEWORK-MIB
  - DOCS-CABLE-DEVICE-MIB
  - CISCO-PROCESS-MIB
  - DOCS-QOS-MIB

  - ENTITY-MIB
  - CISCO-CABLE-QOS-MONITOR-MIB
  - CISCO-DOCS-REMOTE-QUERY-MIB

When extracting the downloaded CBT 3.4 file, these MIBs are placed by default in the following path:

<CBT Install Directory>/httpServer/webapps/ROOT/WEB-INF/mibs

Note Not all the tables contained in these MIBs are used with CBT 3.4.

Generally, all read and create objects in the ccsSpectrumRequestTable and ccsSNRRequestTable can be set with spectrum management tools in CBT 3.4. These objects require a WRITE community string:

- ccsSpectrumRequestTable
  - ccsSpectrumRequestIfIndex
  - ccsSpectrumRequestMacAddr
  - ccsSpectrumRequestLowFreq
  - ccsSpectrumRequestUpperFreq
  - ccsSpectrumRequestResolution
  - ccsSpectrumRequestOperation
  - ccsSpectrumRequestStatus
- ccsSNRRequestTable
  - ccsSNRRequestMacAddr
  - ccsSNRRequestOperation
  - ccsSNRRequestStatus

For additional MIBs information for the Cisco CMTS, refer to the following resources on Cisco.com:

- *Cisco CMTS Universal Broadband Router MIB Specifications Guide 12.2SC*
  http://www.cisco.com/en/US/docs/cable/cmts/mib/reference/guide/ubrmibv5.html
- *SNMP Object Navigator*
  http://www.cisco.com/pcgi-bin/Support/Mibbrowser/unity.pl

CHAPTER 2

Installing, Downloading, and Upgrading CBT 3.4

Use the following steps to download and to install CBT 3.4 on a system with the Linux, Solaris, or Windows operating systems. This section contains the following topics:

- Using TCP Ports for CBT 3.4 Upgrade and Operation, page 2-1
- CBT 3.4 Download and Upgrade, page 2-2
- Uninstalling CBT 3.4, page 2-6

Using TCP Ports for CBT 3.4 Upgrade and Operation

Note Cisco implements a default TCP port of 8105 on the licensing server for CBT 3.4. This prevents port conflict with other applications.

If you require a TCP port other than the default of 8105, then set the TCP port after installation of CBT 3.4, but prior to licensing. Refer to Step 5 below. We recommend ports other than 8005, 8080, 9080, 9443, and 9082, as these are used for tomcat and other applications.

The following are the default ports used by CBT 3.4:

- Port 8105: Java Virtual Machine (JVM) server port
- Port 9080: Non-SSL HTTP port
- Port 9443: SSL HTTP port
- Port 9082: Apache JServ Protocol (AJP) 1.3 Connector
- Port 2640: For Sybase database connectivity
- Port 8020: For Poller operation

CBT 3.4 uses the following port ranges for spectrum operations. Spectrum events such as Trace Window, Spectrogram, CNR Trending, and Generic Query use a specified range of port numbers for operation.

- For both Trace Window events and Spectrogram events, port numbers range from 2100 to 2500.
- For Spectrum tools, such as CNR Trending, port numbers range from 3100 to 3500.
- For Diagnostic tools, such as Generic Query, port numbers range from 5100 to 5500.

The usage of these spectrum port numbers is in increments of 1, that is, the first trace Window launched obtains port number 2100, the next Trace Window obtains port 2101, the next Spectrogram window is port 2102, and so forth till port 2500.

CBT 3.4 Download and Upgrade

Step 1 Download CBT 3.4 software for the desired platform from Cisco.com to your local directory <*CBT3.4_DOWNLOAD_DIR*>.

- Cisco.com (http://www.cisco.com/cisco/software/type.html?mdfid=268439486&flowid=5016&softwareid=281104848)
- The following platform files and release notes are available:
  - There is a single file for the Solaris platform—**CBT34-Solaris.tar.gz**
  - There is a single file for the Linux platform—**CBT34-Linux.tar.gz**
  - Platform files for the Windows platform—**CBT34-Win.exe**
  - CBT 3.4 Release Notes—**CBT34ReleaseNotes.pdf**

Step 2 Extract the software. For Solaris and Linux, use the commands below. For Windows 2000 and Windows XP, skip to Step 3.

- Solaris:
  a. **cd** <*CBT3.4_DOWNLOAD_DIR*>
  b. **gunzip CBT34-Solaris.tar.gz**
  c. **tar xvf CBT34-Solaris.tar**
- Linux:
  a. **cd** <*CBT3.4_DOWNLOAD_DIR*>
  b. **gunzip CBT34-Linux.tar.gz**
  c. **tar xvf CBT34-Linux.tar**

Step 3 If there is a previous CBT installation (CBT 2.x, CBT 3.0, CBT 3.2, or CBT 3.3), uninstall it using the following procedures; otherwise, skip to Step 4.

- Solaris: **cd** <*CBT3.x_DOWNLOAD_DIR*> **./uninstall**
- Linux: **cd** <*CBT3.x_DOWNLOAD_DIR*> **./uninstall**
- Windows:
  - In Windows Explorer, double-click CBT3x-Win.exe in <CBT3.x_DOWNLOAD_DIR>.
  - Remove the existing CBT installation.

Step 4 Install CBT 3.4.

- Solaris: **cd** <*CBT3.4_DOWNLOAD_DIR*> **./install**
- Linux: **cd** <*CBT3.4_DOWNLOAD_DIR*> **./install**
- Windows:
  - From Windows Explorer, double-click **CBT34-Win.exe** in <CBT3.4_DOWNLOAD_DIR>.
  - Select the default answers to the installation questions.

Note A demonstration version of CBT 3.4 is now running (a license is not included in the Cisco.com download). To make the software fully functional, obtain a license for CBT 3.4 from Cisco Systems. See Step 5.

Step 5 Purchased copies of CBT no longer include a software CD, but include a license certificate with a PAK key. This PAK key generates the necessary license file from the following websites, and the software is downloaded from Cisco.com.

- Use this site if you are a registered user of Cisco.com:

  www.cisco.com/go/license

- Use this site if you are not a registered user of Cisco.com:

  www.cisco.com/go/license/public

Demonstration copies can be obtained either from the Internet, or as an ordered accessory for an universal broadband router product:

- The copy available from the Internet uses a registration page where the PAK key is e-mailed to you.
- The copies supplied with the universal broadband router product include a demo license certificate with a PAK key.

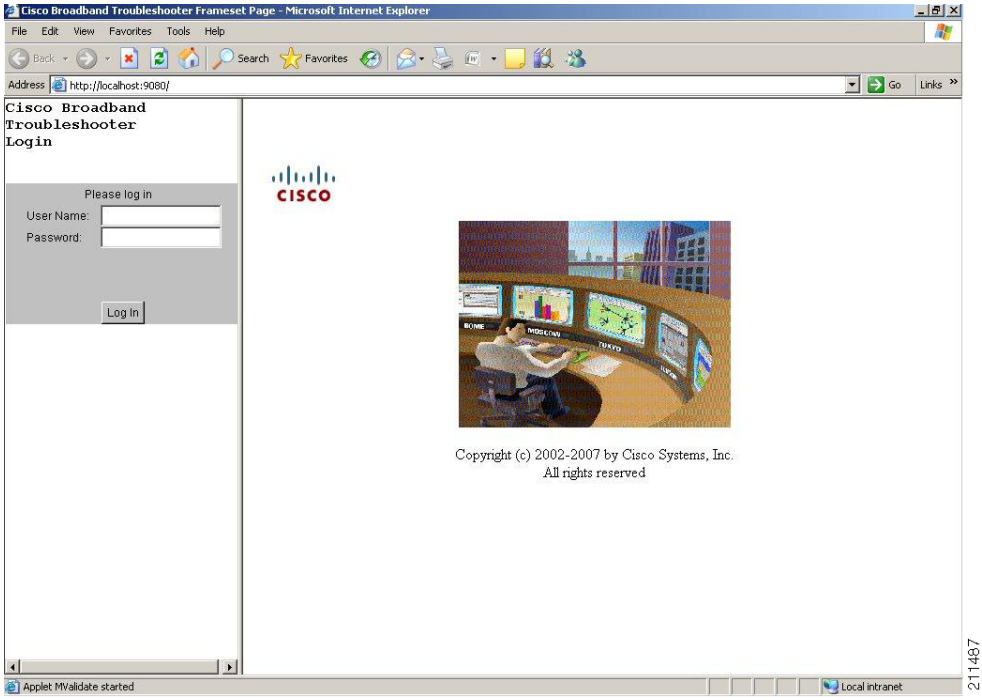
Step 6 Save the license file generated using the PAK key to your system with the name **License**.

Step 7 Use the following commands or click sequences to copy the license to the license directory and restart CBT 3.4.

- Solaris:
  1. **cp License /opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes/lic**
  2. **/opt/CSCOcbt/bin/stop_app**
  3. **/opt/CSCOcbt/bin/start_app**
- Linux:
  1. **cp License /opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes/lic**
  2. **/opt/CSCOcbt/bin/stop_app**
  3. **/opt/CSCOcbt/bin/start_app**
- Windows:
  1. **copy License** *<CBT3.4_INSTALLATION_DIR>\httpServer\webapps\ROOT\WEB-INF\classes\lic>*
  2. Click **Start** > **Programs** > **Cisco Broadband Troubleshooter** > **Stop Troubleshooter**
  3. Click **Start** > **Programs** > **Cisco Broadband Troubleshooter** > **Start Troubleshooter**

Note When the CBT 3.4 license is installed and is active, additional changes to the license are not supported. License changes disable an otherwise valid license.

Figure 2-1 CBT Log In Window



Installing and Starting CiscoView on Solaris

Note The Solaris installation package for CBT 3.4 includes the CiscoView installation package.

Step 1 Obtain the software as per the instructions in the "CBT 3.4 Download and Upgrade" section on page 2-2.

Step 2 Log in as **root**.

Step 3 (Optional) If you are not running Solaris Volume Manager as **root**, you must manually mount the CD:

- If a mount point does not exist, create one by making a new /cdrom directory.
- Enter the following command, where *CD-ROM device* is the name of your CD-ROM:
  mount *CD-ROM device* **/cdrom**

Note If you are running Solaris Volume Manager, the CD-ROM automatically mounts on the /cdrom/cdrom0 directory.

Step 4 To change to the CiscoView Solaris directory and install CiscoView, enter:

```
cd /cdrom/cdrom0/solaris/cv
./setup.sh
```

The CBT server is now running.

Step 5 (Optional) If you are not running Solaris Volume Manager, make sure that no other application is using /cdrom as the current directory so that you can unmount the CD by entering:

```
umount /cdrom
```

Note If you are running Solaris Volume Manager, the CD-ROM automatically unmounts.

Step 6 To eject the CD, enter:

```
eject
```

Step 7 To start CBT by using a command line, enter:

```
~ /opt/CSCOcbt/bin
./start_app
```

or

To access CBT using a browser, open the browser and enter:

`http://`*`CBT server IP address`*`:9080/`

Note In the URL, *CBT server IP address* is the server on which you installed CBT.

Step 8 In the Log In window (see Figure 2-1), enter your username and password.

Step 9 (Optional) To stop CBT manually, enter:

```
~/opt/CSCOcbt/bin
./stop_app
```

Installing and Starting CBT on Windows

Step 1 Obtain the software as per the instructions in the "CBT 3.4 Download and Upgrade" section on page 2-2.

Step 2 From Windows Explorer, double-click **CBT34-Win.exe**.

Step 3 Select the default answers to the installation questions.

The CBT server is now running.

Step 4 To start CBT by using the menus, choose:

Start > Programs > Cisco Broadband Troubleshooter > Start Troubleshooter

Step 5 To access CBT, open the browser and enter:

`http://`*`CBT server IP address`*`:9080/`

Note In the URL, *CBT server IP address* is the server on which you installed CBT.

Step 6 In the Log In window (see Figure 2-1), enter your username and password.

Step 7 (Optional) To stop CBT manually, choose:

Start > Programs > Cisco Broadband Troubleshooter > Stop Troubleshooter

Installing and Starting CBT on Linux

Step 1 Obtain the software as per the instructions in the "CBT 3.4 Download and Upgrade" section on page 2-2.

Step 2 Log in as **root**.

Step 3 To go to the CBT Linux directory, enter:

```
cd /mnt/cdrom/linux
```

Step 4 To start the installation program, enter:

```
./install
```

The CBT server is now running.

Step 5 To start CBT by using a command line, enter:

```
~ /opt/CSCOcbt/bin
./start_app
```

or

To access CBT using a browser, open the browser and enter:

`http://`*`CBT server IP address`*`:9080/`

Note In the URL, *CBT server IP address* is the server on which you installed CBT.

Step 6 In the Log In window (see Figure 2-1), enter your username and password.

Step 7 (Optional) To stop CBT manually, enter:

```
~/opt/CSCOcbt/bin
./stop_app
```

Uninstalling CBT 3.4

This section provides instructions to uninstall CBT 3.4 on each supported platform.

- Uninstalling CBT on Solaris, page 2-6
- Uninstalling CBT on Windows, page 2-7
- Uninstalling CBT on Linux, page 2-7

Uninstalling CBT on Solaris

Step 1 Log in as **root**.

Step 2 Insert the CBT 3.4 CD-ROM for Solaris into the CD-ROM drive.

Step 3 To change to the CBT Solaris directory, enter:

```
cd/cdrom/cdrom0/solaris/cbt
```

Step 4 To uninstall CBT, enter:

```
./uninstall
```

If there is a previous CBT installation (CBT 2.x, CBT 3.0, CBT 3.2, or CBT 3.3), uninstall it using the following step:

- **cd** <*CBT3.x_DOWNLOAD_DIR*> **./uninstall**

Uninstalling CBT on Windows

Step 1 Choose:

Start > Settings > Control Panel > Add/Remove Programs

Step 2 Select **Cisco Broadband Troubleshooter**.

Step 3 Click **Remove**.

If there is a previous CBT installation (CBT 2.x, CBT 3.0, CBT 3.2, or CBT 3.3), uninstall it using the following steps.

- From Windows Explorer, double-click CBT34-Win.exe in <CBT3.x_DOWNLOAD_DIR>.
- Choose **Remove** to remove all installed components of the existing CBT installation.

Uninstalling CBT on Linux

Step 1 Log in as **root**.

Step 2 Insert the CBT 3.4 CD-ROM into the CD-ROM drive.

Step 3 To mount the CD, enter:

```
/bin/mount /mnt/cdrom
```

Step 4 To change to the CBT Linux directory, enter:

```
cd /mnt/cdrom/linux
```

Step 5 To uninstall CBT, enter:

```
./uninstall
```

If there is a previous CBT installation (CBT 2.x, CBT 3.0, CBT 3.2, or CBT 3.3), uninstall it using the following step:

- **cd** <*CBT3.x_DOWNLOAD_DIR*> **./uninstall**

CHAPTER 3

Configuring the Cisco Broadband Troubleshooter

The following configuration procedures are required for using Cisco Broadband Troubleshooter Release 3.4:

- Configuring the Cisco CMTS to Use CBT Spectrum Management Tools, page 3-1
- Configuring the Java Runtime Environment, page 3-2

The following configuration procedures are optional for CBT 3.4:

- Configuring CiscoView (Optional), page 3-4
- Using Subscriber Traffic Management with CBT 3.4 (Optional), page 3-4

Configuring the Cisco CMTS to Use CBT Spectrum Management Tools

The Cisco Broadband Troubleshooter Release 3.4 provides the spectrum management functionality, which is generated by the digital signal processor (DSP) in the Cisco uBR-MC16S, uBR-MC16U, uBR-MC16X, uBR-MC28U, uBR-MC28X, uBR10-MC5X20S/U/H, UBR-MC20X20V, and uBR-MC88V line cards on the Cisco CMTS.

Before using the spectrum management features in Cisco Broadband Troubleshooter, you must complete spectrum management configuration tasks on the Cisco CMTS. These configurations are outside the scope of Cisco Broadband Troubleshooter.

For more information, refer to the following document on Cisco.com:

- *Spectrum Management and Advanced Spectrum Management for the Cisco CMTS*

  http://www.cisco.com/en/US/docs/ios/cable/configuration/guide/cmts_spctrm_mgt.html

Configuring the Java Runtime Environment

This section describes Java Runtime Environment (JRE) requirements and procedures for JRE 5.0, but includes procedures for earlier JRE platforms also.

JRE 5 OS-Level Requirements

CBT 3.4 supports the latest JRE 5.0 platform. However, the network administrator must ensure that all OS-level patches are installed to avoid system problems with CBT 3.4. This section describes patch requirements according to platform.

The Sun JRE 5.0 is a plug-in that must be installed in the client browser. Cisco Broadband Troubleshooter detects if JRE 5.0 is installed in the browser. This section provides directions to install JRE 5.0 in the browsers supported by each platform.

Installing JRE 5.0 in Microsoft Internet Explorer on the Windows Platform

If JRE 5.0 is not installed in Internet Explorer on the Windows platform, the installation is automatic.

Installing JRE 5.0 in Netscape Navigator on the Windows Platform

If JRE 5.0 is not installed in Netscape Navigator on the Windows platform, refer to the following location:

http://java.sun.com/javase/downloads/index_jdk5.jsp

To install JRE 5.0 in Netscape Navigator on the Windows platform:

Step 1 From https://sdlc5e.sun.com/ECom/EComActionServlet;jsessionid=A603027E34989B56B4909B018FB8F468, save jre-1_5_0_06-windows-i586-p.exe to a local directory.

Note In these steps, *JRE5.0_07InstallationDirectory* represents the local directory you choose.

Step 2 To install JRE 5.0, choose:

Start > Run > *JRE5.0_07InstallationDirectory* \jre-1_5_0_06-windows-i586-p.exe

Step 3 Close Navigator.

Step 4 Reopen Navigator.

Installing JRE 5.0 in Netscape Navigator on the Solaris Platform

If JRE 5.0 is not installed in Netscape Navigator on the Solaris platform, refer to the following location:

http://java.sun.com/javase/downloads/index_jdk5.jsp

To install JRE 5.0 on the Solaris platform:

Step 1 From https://sdlc5e.sun.com/ECom/EComActionServlet;jsessionid=A603027E34989B56B4909B018FB8F468, save jre-1_5_0_08-solaris-sparc.sh to a local directory.

Note In these steps, *JRE5.0InstallationDirectory* represents the local directory you choose and *Netscape7.0InstallationDirectory* represents directory where Netscape 7.0 is installed.

Step 2 To install JRE 5.0, enter:

```
jre-1_5_0_08-solaris-sparc.sh
```

Step 3 Depending on the version of Netscape you are using, point to the Java plug-in by doing the following:

- **Netscape 4.7**—Set the environment variable NPX_PLUGIN_PATH to point to the Java plug-in by entering:

  setenv NPX_PLUGIN_PATH *JRE5.0InstallationDirectory***/plugin/sparc/ns4**

- **Netscape 7.0**—Create a softlink to point to the Java plug-in by entering:

  ln -s *JRE5.0InstallationDirectory*/plugin**/sparc/ns610/libjavaplugin_oji.so** *Netscape7.0InstallationDirectory***/plugins/libjavaplugin_oji.so**

Step 4 Close Navigator.

Step 5 Reopen Navigator.

Installing JRE 5.0 in Netscape Navigator on the Linux Platform

If JRE 5.0 is not installed in Netscape Navigator on the Linux platform, refer to the following location:

http://java.sun.com/javase/downloads/index_jdk5.jsp

To install JRE 5.0 in Netscape Navigator on the Linux platform:

Step 1 From https://sdlc5e.sun.com/ECom/EComActionServlet;jsessionid=A603027E34989B56B4909B018FB8F468, save jre-1_5_0_08-linux-i586.bin to a local directory.

Note In these steps, *JRE5.0_07InstallationDirectory* represents the local directory you choose and *Netscape7.0InstallationDirectory* represents directory where Netscape 7.0 is installed.

Step 2 To install JRE 5.0, enter:

```
jre-1_5_0_08-linux-i586.bin
```

Step 3 Depending on the version of Netscape you are using, point to the Java plug-in by doing the following:

- **Netscape 4.7**—Set the environment variable NPX_PLUGIN_PATH to point to the Java plug-in by entering:

  setenv NPX_PLUGIN_PATH *JRE5.0InstallationDirectory***/plugin/i386/ns4**

- **Netscape 7.0**—Create a softlink to point to the Java plug-in by entering:

```
ln -s JRE5.0_InstallationDirectory/plugin/i386/ns610/libjavaplugin_oji.so
Netscape7.0InstallationDirectory/plugins/libjavaplugin_oji.so
```

Step 4 Close Navigator.

Step 5 Reopen Navigator.

Configuring CiscoView (Optional)

CiscoView is a web-based device management application that provides dynamic status, monitoring, and configuration information for a range of Cisco internetworking products. CiscoView displays a physical view of a device chassis, with color-coding of modules and ports for at-a-glance status, and has two levels of capabilities:

- **Monitoring capabilities**—Displays performance and other statistics
- **Configuration capabilities**—Allows comprehensive changes to devices, if you have security privileges

For product information, refer to the following document on Cisco.com:

http://www.cisco.com/en/US/products/sw/cscowork/ps4565/index.html

You can launch CiscoView 5.*x* from CBT to monitor a device and get real-time information on it.

To configure and launch CiscoView from CBT:

Step 1 To identify the CiscoView server so that CBT can launch the application, from the **Configuration** menu, choose **CiscoView Server**.

Step 2 To identify the device that you want to monitor with CiscoView, from the **Hotline Tools** menu, choose **CiscoView**.

Step 3 For a complete description of each field in the CiscoView dialog boxes, click **Help**.

Note The Solaris installation package for CBT 3.4 includes the CiscoView installation package.

Using Subscriber Traffic Management with CBT 3.4 (Optional)

To use the Subscriber Traffic Management (STM) feature of CBT 3.4, the **cable qos enforce-rule** command on the Cisco CMTS must be used with a QoS profile that is not created by the cable modem. Any in-use QoS profile, such as one created by a cable modem, can be turned into a management profile.

Note STM is supported on the MC16U, MC16X, MC28U, MC28X, MC5X20S/U/H, MC20X20V, and MC88V line cards with CBT 3.4. STM data is enabled with CBT 3.4.

To create a management profile:

	Command	Purpose
Step 1	Router# **show cable modem registered**	Finds the registered profiles on the Cisco CMTS. Finds a current registered profile that has the features you need.
Step 2	Router# **configure terminal**	Enters global configuration mode.
Step 3	Router (config)# **cable qos profile** *n*	Enters the QoS profile number of the registered profile that has the features you need.
Step 4	Router (config)# **exit**	Exits global configuration mode.

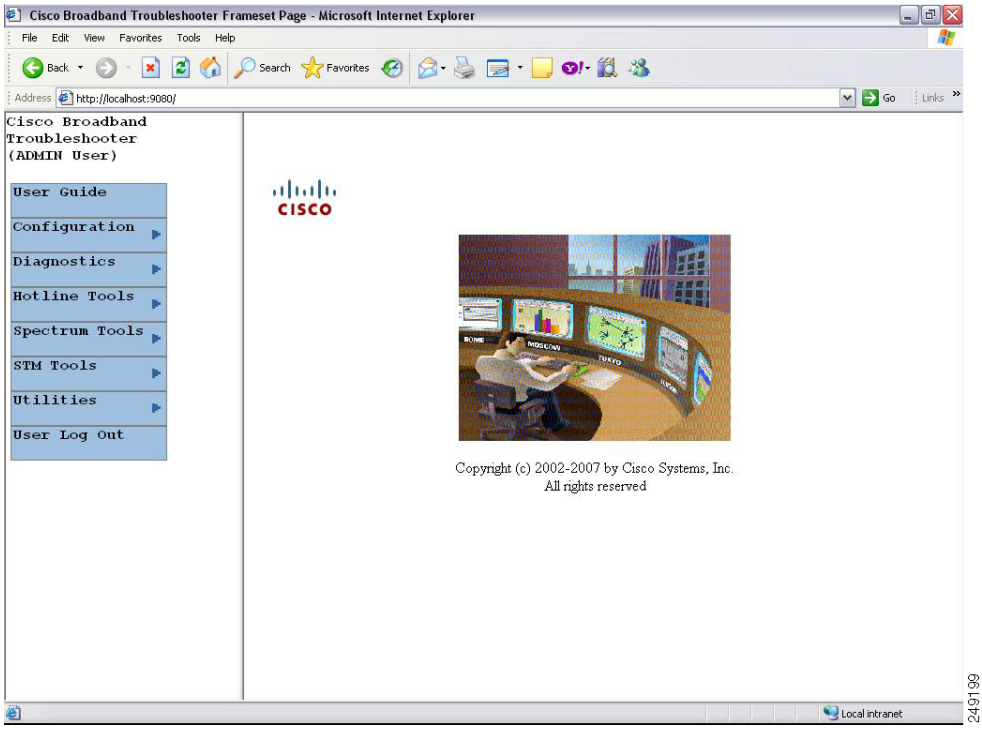
Reviewing the CBT Task Menu

This section describes the following components of the CBT Task Menu:

- User Guide, page 3-6
- Configuration, page 3-7
- Diagnostics, page 3-7
- Hotline Tools, page 3-7
- Spectrum Tools, page 3-7
- STM Tools, page 3-8
- Utilities, page 3-8
- User Log Out, page 3-8

After you log on to CBT, the Task menu appears. See Figure 3-1:

Figure 3-1 CBT Task Menu



Above the menu, your user type is displayed. For more information on user types, see "User Types in CBT" section on page 3-9.

All tasks in CBT are divided into the following menus:

- User Guide
- Configuration
- Diagnostics
- Hotline Tools
- Spectrum Tools
- STM Tools
- Utilities
- User Log Out

The following sections describe the tasks within each menu.

User Guide

The User Guide menu links to the *Cisco Broadband Troubleshooter Release 3.4 User Guide*.

Configuration

The Configuration menu contains the following tasks:

- **User List**—Specify who can access the application and the level of access permissions.
- **CMTS List**—Specify the routers you want to manage by entering the information manually.
- **CMTS Group**—Organize CMTSs into groups, so that you can manage several CMTSs as a single unit.
- **Scheduler**—Schedule a time to capture data from one or more Cisco CMTSs:
  - **Flap List**—Set up the flap list configuration parameters.
  - **Spectrum Data**—Schedule time to capture spectrum data for the selected Cisco CMTS.
  - **STM Data**—Schedule time to capture STM data for the selected Cisco CMTS.
- **Flap List Parameters**—Specify configuration parameters for the flap list.
- **CiscoView Server**—Identify the CiscoView server so that you can launch CiscoView from CBT.
- **Message Log**—Filter errors and informational messages that the server logs.
- **System**—Modify system-level parameters that are saved to the CONFIGS.INI file.
- **External Interface**—Specify how and where to access subscriber and provisioning information.
- **Poller**—Schedule the polling frequency to procure provisioning information from the Cisco CMTSs.

Diagnostics

The Diagnostics menu contains the following tasks:

- **Flap List Analysis**—Analyze the results of data captured for the selected Cisco CMTSs.
- **Cable Monitor Tool**—Summarize information of Cisco cable modems.
- **Generic Query**—Poll the Cisco CMTS for individual MIB variables.

Hotline Tools

The Hotline Tools menu contains the following tasks:

- **Locate Modem**—Summarize information on one or more cable modems.
- **Modem Status**—Generate a real-time status report for a cable modem.
- **CMTS Dashboard**—Procure detailed information on one or more Cisco CMTSs.
- **Ping Test**—View if a device has connectivity.
- **CiscoView**—Identify the device that you want to monitor with CiscoView.

Spectrum Tools

The Spectrum Tools menu contains the following tasks:

- **Trace Window**—Monitor the power and noise levels for a selected cable modem or upstream port.

- **Spectrogram**—Monitor the power and noise levels *over time* for a selected cable modem or upstream port.
- **CNR Analysis**—Retrieve Carrier-to-Noise Ratio (CNR) for selected modems or upstream ports.
- **CNR Trending**—Retrieve CNR trending data for selected modems or upstream ports.
- **Data Playback**
  - **Trace Window**—Play back a saved Trace window.
  - **CNR Analysis**—Play back a saved CNR analysis.
  - **CNR Trending**—View the saved trending data and to monitor the CNR in the CNR trending chart.
- **Polling Status**
  - **Real Time**—Procure a list of each spectrum analysis that is in process on a client machine.
  - **Scheduled**—View the details of the scheduled spectrum polling events.

STM Tools

The Subscriber Traffic Management Tools menu contains the following tasks:

- **Enforce Rules**—View the STM enforce rules on a Cisco CMTS.
- **All Violators**—View the STM violator modems connected to a Cisco CMTS.
- **All Violators Real Time**—View the STM violator modems connected to a Cisco CMTS.
- **Locate Violator**—Find violator modems connected to a Cisco CMTS.

Utilities

The Utilities menu contains the following tasks:

- **Import**
  - **CMTS List**—Import an ASCII text file of subscriber data, such as name and phone number, into CBT.
  - **Subscriber Data**—Import an ASCII text file of all the managed Cisco CMTSs into CBT, so that you do not have to enter each Cisco CMTS manually.
- **Purge Saved Data**
  - **Flap List**—Remove a saved Flap List Analysis from the database.
  - **Spectrum Data**—Remove a saved Trace Window or CNR analysis from the database.
  - **STM Data**—Remove previously saved spectrum data from the database.
- **Message Log**—View error and information messages that the server logs by using filters to show the level of detail you want.

User Log Out

The User Log Out menu logs you out of the Cisco Broadband Troubleshooter.

User Types in CBT

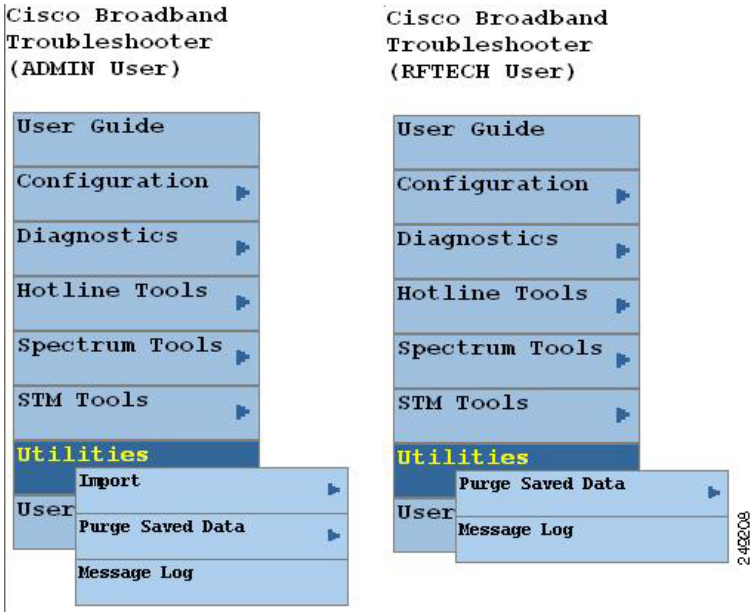
The number of tasks you see in the CBT menu varies according to the type of user you are. There are two user types in CBT, each with access to different tasks:

- **ADMIN**—The administrator has access to all tasks in each menu.
- **RFTECH**—The RF technician has access to a subset of tasks in the following menus:
  - **Configuration menu**—Can access the **Flap List**, **Spectrum Data**, **STM Data** tasks under the **Scheduler** menu.
  - **Utilities menu**—Can access the **Purge Saved Data** and **Message Log** tasks only. Figure 3-3 shows the Utilities menu for each user type.

  In each other menu, the RFTECH user has access to all tasks.

Figure 3-2 Configuration Menus for ADMIN and RFTECH Users

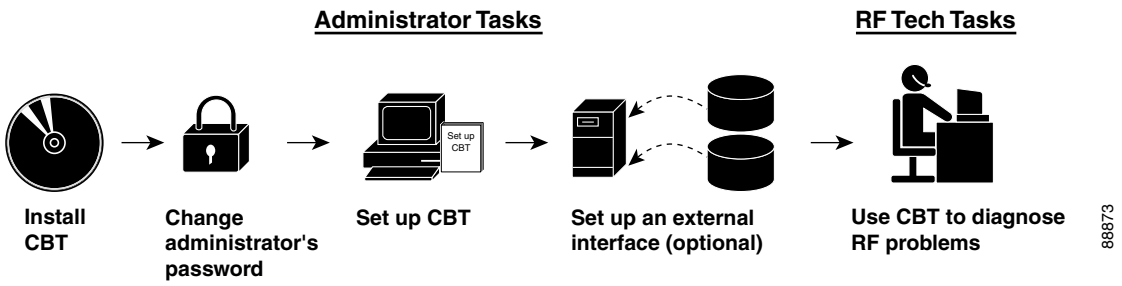
Figure 3-3 Utilities Menus for ADMIN and RFTECH Users



Workflow of Administrator and RF Technician Tasks

Figure 3-4 illustrates a high-level workflow of tasks that each type of user, the administrator and the RF technician, performs in CBT.

Figure 3-4 Administrator and RF Technician Tasks in CBT



Administrator Tasks

The first four tasks shown in Figure 3-4 are performed only by an administrator.

- **Install CBT**—The administrator installs CBT on one of the supported platforms: Linux, Solaris, or Windows.
- **Change administrator's password**—The administrator changes the default ADMIN password to ensure proper security.
- **Set up CBT**—The administrator performs the following setup tasks to configure CBT according to the site's needs:

  - Add users, routers, and subscriber information
  - Configure parameters in .INI files: CONFIGS.INI, GUNSLINGER.INI, POLLER.INI, and SPECTRUM.INI
  - Set message log options to see errors logged by the server
- **Set up an external interface (optional)**—If CBT does not retrieve subscriber or provisioning data from its local database, which is the system default, the administrator sets up an external interface to retrieve the data by using one of the following methods:
  - **Subscriber information**—Can be accessed by script, HTTP, or Lightweight Directory Access Protocol (LDAP)
  - **Provisioning information**—Can be accessed by script, HTTP, LDAP, Cisco Broadband Provisioning Registrar (BPR), or Cisco Network Registrar (CNR)

Note BPR is now called Cisco Broadband Access Center for Cable (BACC) and CNR is now called Cisco CNS Network Registrar (CNS-NR). You can see the new names on Cisco.com and in future releases. The CBT 3.4 GUI and documentation use the latest names and acronyms.

The remainder of this guide explains these tasks in greater detail.

RF Tech Tasks

The last task shown in Figure 3-4 is performed by the RF technician:

- **Use CBT to diagnose RF Problems**—The RF technician can diagnose a frequency problem in the cable plant by using any or all of the CBT troubleshooting capabilities:
  - **CMTS Dashboard**—To view summary information on a Cisco CMTS and the cable modems attached to it
  - **Flap List Analysis**—To analyze data from a Cisco CMTS and categorize problems into four groups: provisioning, reverse path noise, reverse path attenuation, and packet corruption
  - **Spectrum management tools**—To view performance data for an upstream port or cable modem
  - **Cable modem tools**—To ping, monitor, or query a cable modem to assess its status

Setting Up CBT

This section describes the following important setup tasks that only a CBT administrator can perform:

- Changing the Administrator Password, page 3-12
- Disabling or Enabling Password Text Display, page 3-12
- Adding Users, page 3-12
- Adding Router Information, page 3-14
- Adding Subscriber Information, page 3-16
- Adding CMTS Group, page 3-17

Changing the Administrator Password

For security reasons, we recommend that you change the default password for the administrator.

To change the administrator password:

Step 1 Open the browser and enter the URL for your CBT server:

`http://CBT server IP address:9080/`

Note In the URL, *CBT server IP address* is the server on which you installed CBT.

The Login dialog box appears, as shown in Figure 3-5.

Step 2 Log in to CBT as **admin** with the default password **changeme**.

Step 3 To change the password, from the **Configuration** menu, choose the administrator from the **User List**, and proceed through dialogs that prompt you for the new password.

Disabling or Enabling Password Text Display

CBT enables you to display password text, or to disable password text display according to your needs. By default, this parameter is set to YES. If you set the corresponding parameter to NO, CBT does not display password text when the contents of the Password field is displayed.

To disable or enable password text display for one or more User Names:

Step 1 Open the GUNSLINGER.INI file available in the CBT server machine.

- For Solaris and Linux, the GUNSLINGER.INI file is under the directory /opt/CSCOcbt/httpServer/webapps/ROOT or <CBT_Install_Directory>/httpServer/webapps/ROOT.
- For Windows, the GUNSLINGER.INI file is under the directory C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\ or <CBT_Install_Directory>\httpServer\webapps\ROOT\.

Step 2 Change the value of the ShowPasswordWhileEditing parameter to YES or NO as per your need.

- For YES, CBT 3.4 displays the password text when the password field is highlighted.
- For NO, CBT 3.4 does not display the password text when the password field is highlighted.

Step 3 Save the GUNSLINGER.INI file with your changes.

Step 4 Restart CBT 3.4.

Adding Users

You add users in CBT by specifying the following information for a user:

- **User Name**—Specifies the name the user enters in the Login dialog box.

- **Password**—Specifies the password the user enters in the Login dialog box.
- **User Type**—Determines the level of privileges the user has. The two choices are ADMIN and RFTECH.

To add users in CBT:

Step 1 Log in to CBT as **admin**. See Figure 3-5.

Step 2 From the Configuration menu, choose **User List**. The Manage Users dialog box appears.

Step 3 Select **Add User**, and proceed through dialog boxes that prompt you for the new user information. (See Figure 3-6.)

Figure 3-5 CBT Log In Window

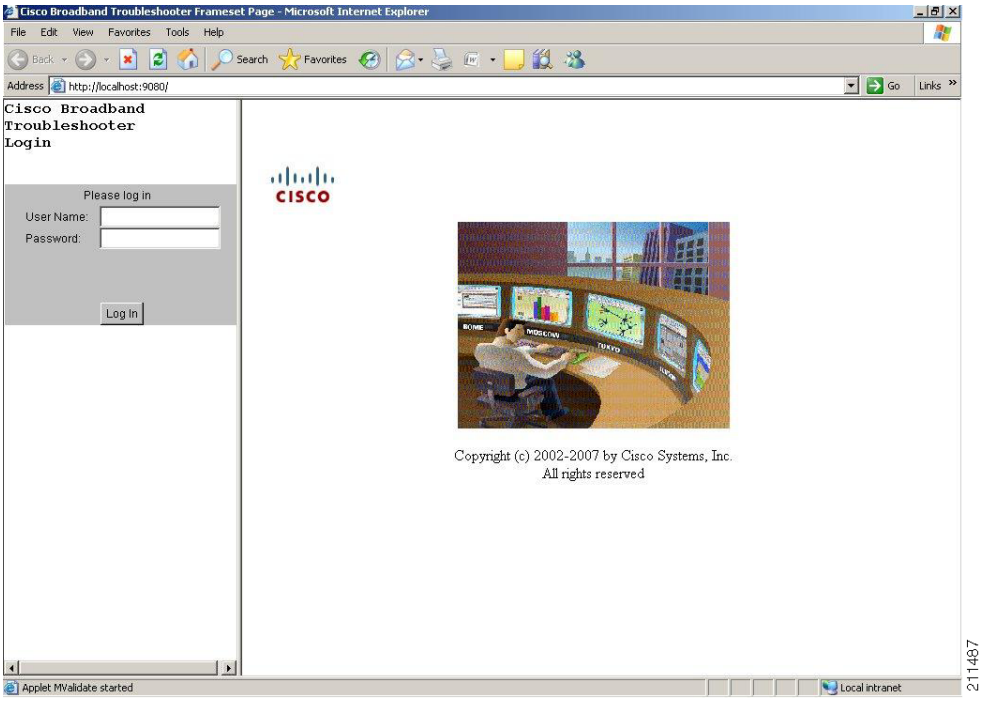
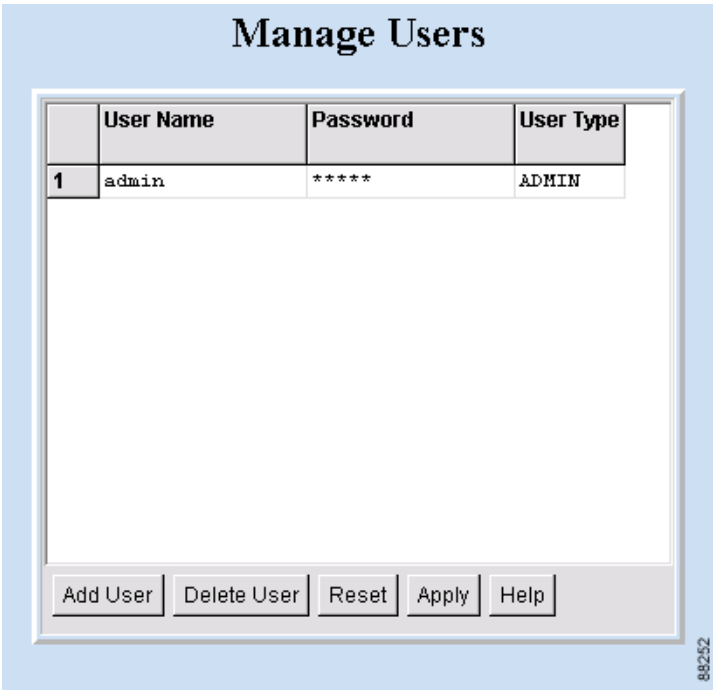


Figure 3-6 Manage Users Dialog Box



Adding Router Information

You can add router information in CBT by using either of the following methods in the CMTS List option of the Configuration menu:

- Enter information for a router by manually filling in fields for each router.
- Import an ASCII file that contains a list of managed Cisco CMTSs. A sample file is in the following locations:
  - **Linux and Solaris**—/opt/CSCOCBT/samples/showmap/cmts.txt
  - **Windows**—%CSCOCBT_ROOT%\samples\showmap\cmts.txt

If you import an ASCII file, the file must contain 13 fields for each CMTS:

- Fields 1 to 4—IP address, Community String (Read), Community String (Write), and CM Community String (Read).
- Fields 5 to 9—User Name, User Password, Line Password, Enable Password, and Telnet Enable.
- Field 10—Telnet port the router uses (the default is 23).
- Fields 11 and 12—AAA Username Prompt and AAA Password Prompt.
- Field 13—Setting to Ping CMTS. The options are YES or NO, and the default is YES.

Each field must be separated by a comma (,). If you leave a field blank, use a comma as a placeholder for it.

The following example shows the ASCII file format and data for a Cisco CMTS in this format:

```
172.21.73.10,public,private,public,jdoe,nAda,nAda,nAda,yes,23,UserName,Password,yes
```

To add information about the Cisco uBR7200 series, and Cisco uBR10012 universal broadband routers in CBT:

Step 1 Log in to CBT as **admin**.

Step 2 Choose a method to import router information:

- To add information for a router manually, from the Configuration menu, choose **CMTS List.**
- To add information by importing an ASCII file of router information, from the Utilities menu, choose **Import > CMTS List**. (See Figure 3-7.)

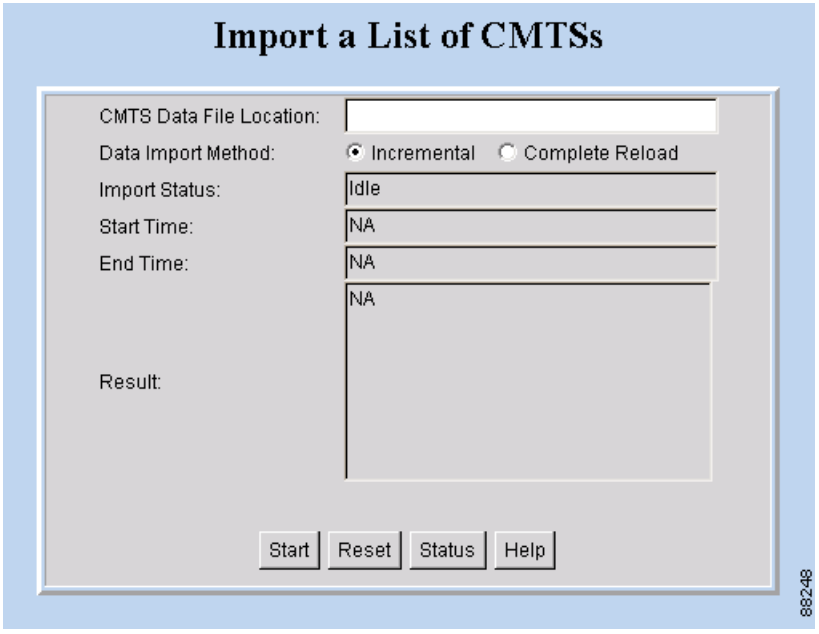
Step 3 Designate the location of the Cisco CMTS data file in the File Location field, and complete the remaining fields as required, click **Start** or **Reset**, or click **Status** to display current status.

Step 4 For a complete description of each field in the CMTS List dialog box, click **Help**.

Tip To successfully manage routers, make sure that you enable SNMP on these routers. Verify that access lists, if configured in the router, allow for SNMP read access from your workstation.

Note The Cisco CMTS name cannot contain a question mark (?).

Figure 3-7 Import a List of CMTSs Dialog Box



Adding Subscriber Information

To add subscriber information in CBT, you import an ASCII text file that contains the subscriber information. A sample file is in the following locations:

- **Linux and Solaris**—/opt/CSCOCBT/samples/showmap/subscriberinfo.txt
- **Windows**—%CSCOCBT_ROOT%\samples\showmap\subscriberinfo.txt

For each subscriber, the ASCII text file must contain 12 fields. The following fields are typical, although only the first one is required:

1. Cable modem MAC address
2. Account ID
3. Customer name
4. Phone number
5. Address
6. City
7. State
8. Zip
9. Country
10. Class of service
11. Fiber node
12. User-defined field

Each field, including blank ones, must be separated with the pipe symbol |.

The following example shows the ASCIII file format data for a subscriber in this format:

```
000216d5a0cf|ID987654|JohnDoe|4085551212|175 West Tasman|San
Jose|CA|95134|USA|Gold|FiberNode_SanJose|Paid
```

To add subscriber information to CBT:

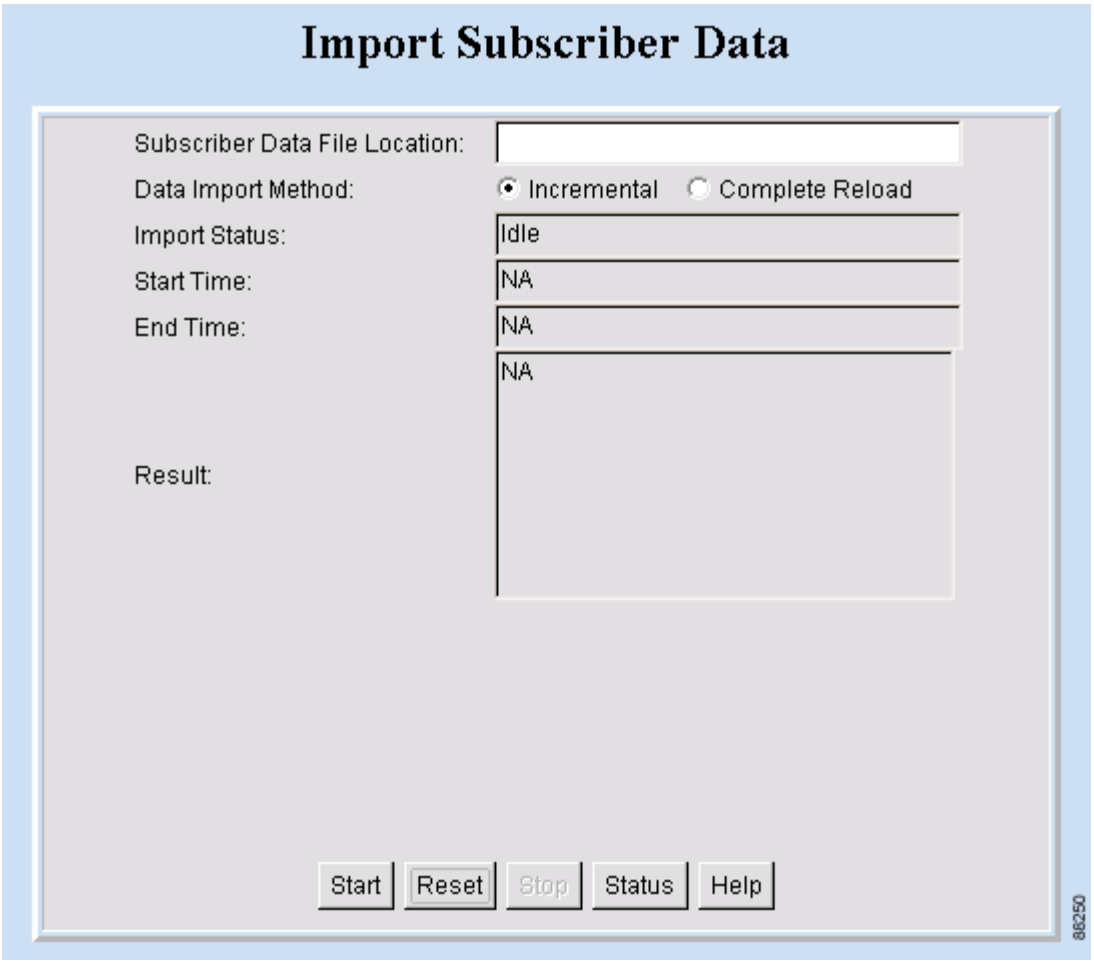
Step 1 Log in to CBT as **admin**.

Step 2 From the Utilities menu, choose **Import > Subscriber Data**. (See Figure 3-8.)

Step 3 Designate a Data File Location, identify the import method, and click **Start** or **Reset**, according to your needs.

Step 4 For a complete description of each field, click **Help**.

Figure 3-8 Import Subscriber Data Dialog Box



Adding CMTS Group

To add a CMTS group:

Step 1 Log in to CBT as **admin**.

Step 2 From the Configuration menu, choose **CMTS Group**. The CMTS Group dialog box appears. (See Figure 3-9.)

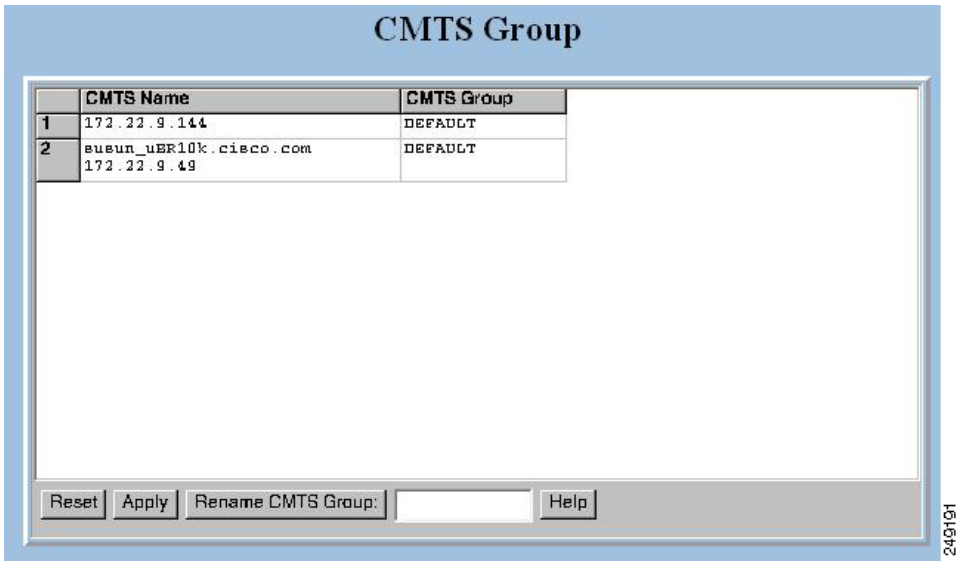
Step 3 Enter a group name in the CMTS Group column for each CMTS in the list (the default group name is DEFAULT, which includes all CMTSs not assigned with a group name).

Step 4 To save the group name assignments, click **Apply**.

Step 5 For a complete description of each field, click **Help**.

Note After you have grouped all the Cisco CMTSs into various groups, you can select the group rather than individual CMTSs.

Figure 3-9 CMTS Group Dialog Box



Setting Parameters in .INI Files

This section describes the parameters you set in three .INI files in CBT:

- CONFIGS.INI—Where you set certain system-level parameters
- POLLER.INI—Where you set parameters for the Poller, the separate Java application that polls Cisco CMTSs for provisioning data
- SPECTRUM.INI—Where you set parameters for the spectrum management tools
- GUNSLINGER.INI—Where you set parameters for password text display and hostnames.

Note Cisco Broadband Troubleshooter and Cisco Cable Diagnostic Manager (CCDM) are companion network management software applications that share some functionality. CCDM is for the MSO customer service representative. In some instances, a parameter in an .INI file applies to one product but not to the other. Those parameters are identified in this section.

This section contains the following topics relating to setting parameters in .INI files for CBT 3.4.

- Parameters in the CONFIGS.INI File, page 3-19
- Parameters in the POLLER.INI File, page 3-21
- Parameters in the SPECTRUM.INI File, page 3-22
- Parameters in the GUNSLINGER.INI File, page 3-24

Parameters in the CONFIGS.INI File

This section describes parameters that you configure in the CONFIGS.INI file, which is in the following locations:

- **Linux**—/opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes
- **Solaris**—/opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes
- **Windows**—C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\WEB-INF\classes

Note When you configure parameters in CONFIGS.INI, you must restart the CBT server for the new settings to take effect.

In addition to specifying system-level parameters in the CONFIGS.INI file, you can also specify some of the parameters in the GUI.

Tip To specify configuration parameters in the GUI, from the Configuration menu, choose **System**.

Phone Number Format

The phoneLength parameter applies to Cisco Cable Diagnostic Manager only.

Modem Status Thresholds

When you view a status report for a cable modem, color-coding indicates the modem's status:

- **Green**—Indicates a normal condition. The modem is online and is within the thresholds that are specified to indicate proper performance.
- **Red**—Indicates that an error condition exists. The modem is under or over thresholds that are specified to indicate proper performance.

The following parameters set the thresholds that determine the boundaries for color-coding and must be configured in the CONFIGS.INI file:

- upstreamXmitPowerFloor
- upstreamXmitPowerCeiling
- downstreamSnrFloor
- downstreamReceivePowerFloor
- downstreamReceivePowerCeiling

upstreamXmitPowerFloor—This parameter determines the minimum upstream transmit power for a cable modem to show a functioning status. This measurement is expressed in decibels millivolt (dBmV). The default setting is 34, as shown below:

```
upstreamXmitPowerFloor=34
```

upstreamXmitPowerCeiling—This parameter determines the maximum upstream transmit power for a cable modem to show a functioning status. This measurement is expressed in decibels millivolt (dBmV). The default setting is 52, as shown below:

```
upstreamXmitPowerCeiling=52
```

downstreamSnrFloor—This parameter determines the minimum signal-to-noise (SNR) for a cable modem to show a functioning status. SNR is a measure of transmission quality, which is expressed by the ratio of good data (signal) to interference (noise) that is heard on a line. This ratio is measured in decibels (dB). A higher ratio indicates a better transmission quality. For example, 30 dB is better than 24 dB. The default setting is 30, as shown below:

```
downstreamSnrFloor=30
```

downstreamReceivePowerFloor—This parameter determines the minimum downstream receive power for a cable modem to show a functioning status. This measurement is expressed in decibels millivolt (dBmV). The default setting is 15, as shown below:

```
downstreamReceivePowerFloor=15
```

downstreamReceivePowerCeiling—This parameter determines the maximum downstream receive power for a cable modem to show a functioning status. This measurement is expressed in decibels millivolt (dBmV). The default setting is 5, as shown below:

```
downstreamReceivePowerCeiling=5
```

Southbound Interface Settings

These parameters apply to Cisco Cable Diagnostic Manager only:

SouthBoundExtIntf

SouthBoundExtIntfTimeout

Modem State Settings

These parameters apply to Cisco Cable Diagnostic Manager only:

ModemStateMaxLife

ModemStateMaxHash

Query Timeout

The administrator specifies external interface parameters to access subscriber and provisioning information. Most of these parameters are in the Set External Interfaces dialog box. However, you must configure the following external interface parameters in the CONFIGS.INI file.

SubscriberExtIntfTimeout—This parameter determines the maximum wait time before CBT times out during a query for subscriber information. This measurement is expressed in milliseconds (msecs). The default setting is 5000, which is equal to 5 seconds, as shown below:

```
SubscriberExtIntfTimeout=5000
```

ProvisionExtIntfTimeout—This parameter determines the maximum wait time before CBT times out during a query for provisioning information. This measurement is expressed in milliseconds (msecs). The default setting is 5000, which is equal to 5 seconds, as shown below:

```
ProvisionExtIntfTimeout=5000
```

Parameters in the POLLER.INI File

This section describes parameters that you configure in the POLLER.INI file, which is in the following locations:

- **Linux and Solaris**—/opt/CSCOcbt/bin
- **Windows**—%CSCOCBT_ROOT%\bin

Note For a detailed description of the Poller, see the "About the Poller" section on page 3-33.

The Poller accesses POLLER.INI to gather the following configurable parameters:

- maximumThread
- maximumDBConnection
- SnmpTimeout
- SnmpRetry
- Debug
- maximumAge
- PollerServerPort

maximumThread—This parameter determines the maximum number of different execution threads that can run simultaneously in the Poller. The valid range is 1 to 50. The default setting is 20, as shown below:

maximumThread=20

maximumDBConnection—This parameter determines the maximum number of CBT Sybase database connections the Poller uses to perform the polling operation. The valid range is 1 to 7. The default setting is 7, as shown below:

maximumDBConnection=7

SnmpTimeout—This parameter determines the number of seconds before the attempt to create the SNMP connection reaches a timeout or fail. The valid range is 1 to 3. The default setting is 2, as shown below:

SnmpTimeout=2

SnmpRetry—This parameter determines the number of times you want to try to create the SNMP connection. The valid range is 1 to 5. The default setting is 3, as shown below:

SnmpRetry=3

Debug—This parameter determines if debug messages are sent to the log file, which is in the following location:

/opt/CSCOcbt/logs/Poller.log

Valid values are true and false. The default setting is false, as shown below:

Debug=false

maximumAge—This parameter tells the Poller which outdated records to purge, based on the number of polling intervals that have occurred during which those records have not been updated. This measurement is expressed in polling intervals. A valid range is 1 to 7. The default setting is 2, as shown below:

maximumAge=2

PollerServerPort—This parameter sets the HTTP server port number where the Poll Manager listens to polling requests. The default setting is 8040:

PollerServerPort=8040

Note If a parameter in POLLER.INI is set out of the valid range, CBT resets the parameter to its default value.

Parameters in the SPECTRUM.INI File

This section describes parameters that you configure in the SPECTRUM.INI file, which is in the following locations:

- **Linux**—/opt/CSCOcbt/jakarta-tomcat-4.0.3/webapps/ROOT/WEB-INF/classes
- **Solaris**—/opt/CSCOcbt/httpServer/servlets
- **Windows**—C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\WEB-INF\classes

Note When you configure parameters in SPECTRUM.INI, you must restart the CBT server for the new settings to take effect.

The parameters you specify in the SPECTRUM.INI file affect the following spectrum management tools:

- Trace Window
- Spectrogram
- CNR Analysis
- CNR Trending
- Data Playback
- Polling Status

The spectrum management tools access SPECTRUM.INI to gather the following configurable parameters:

- CNRFloor
- CNRCeiling
- CNRFloorColor
- CNRCeilingColor
- SpecDataMaxRequests
- SpecDataQueryInterval
- SpecDataMaxCPU
- CableModemCacheAge

CNRFloor—This parameter determines the lower threshold for the carrier-to-noise ratio that you see in the spectrum tools. CNR is a measure of transmission quality, which is expressed by the ratio of good data from the carrier to interference or noise that is heard on a line. A higher ratio indicates a better transmission quality. If the CNR is below this lower threshold, the value is highlighted in the color you

specify in the CNRFloorColor parameter. This measurement is expressed in decibels (dB). Acceptable values range from -20 to 60 dB. If an unacceptable value is entered, the system default is used. The default setting is 20, as shown below:

CNRFloor=20

CNRCeiling—This parameter determines the upper threshold for the carrier-to-noise ratio that you see in the spectrum tools. CNR is a measure of transmission quality, which is expressed by the ratio of good data from the carrier to interference or noise that is heard on a line. A higher ratio indicates a better transmission quality. If the CNR is below this upper threshold, the value is highlighted in the color you specify in the CNRCeilingColor parameter. This measurement is expressed in decibels (dB). Acceptable values range from -20 to 60 dB. If an unacceptable value is entered, the system default is used. The default setting is 30, as shown below:

CNRCeiling=30

CNRFloorColor—This parameter determines the color that is used to highlight a carrier-to-noise ratio that is below the lower threshold, which you specify in the CNRFloor parameter. Accepted colors are red, yellow, blue, cyan, green, magenta, orange, and pink. If an unacceptable color is entered, the system default is used. The default setting is red, as shown below:

CNRFloorColor=Red

CNRCeilingColor—This parameter determines the color that is used to highlight a carrier-to-noise ratio that is below the upper threshold, which you specify in the CNRCeiling parameter. Accepted colors are red, yellow, blue, cyan, green, magenta, orange, and pink. If an unacceptable color is entered, the system default is used. The default setting is yellow, as shown below:

CNRCeilingColor=Yellow

SpecDataMaxRequests—This parameter determines the maximum number of active rows of ccsSpectrumRequestTable in the CMTS SNMP agent. This setting is necessary to avoid overloading the CMTS, and CBT checks to ensure that the number is not exceeded. If the number of active rows in the associated CMTS has reached the specified value, new spectrum requests cannot be launched. The default setting is 10, as shown below:

SpecDataMaxRequests=10

SpecDataQueryInterval—This parameter determines the time required after a spectrum data response and before the next spectrum data query. This measurement is expressed in milliseconds and should not be edited. The default setting is 5000, as shown below:

SpecDataQueryInterval=5000

SpecDataMaxCPU—This parameter determines the CPU utilization in the CMTS SNMP agent. This setting is necessary to avoid overloading the CMTS, and CBT checks to ensure that the percentage is not exceeded. If the CPU utilization in the associated CMTS has reached the specified value, new spectrum requests cannot be launched. This measurement is expressed in percentage. The default setting is 80, as shown below:

SpecDataMaxCPU=80

CableModemCacheAge—This parameter determines the cable modem caching time for the spectrum management tools. If the value is set to 0, the CBT server generates an SNMP request and bypasses the cache, which slows down performance if there are many modems associated with the selected CMTS. This measurement is expressed in seconds. If a negative number is entered, the system default is used. The default setting is 3600, as shown below:

CableModemCacheAge=3600

Note You can update the cache for a selected CMTS manually by using the GUI. From the Spectrum Tools menu, choose **Trace Window**, **Spectrogram**, **CNR Analysis**, **CNR Trending**, or **Polling Status**. Then click **Update CM Info**.

Several additional administrative parameters may be set in the SPECTRUM.INI file to support the latest CBT 3.4 Spectrum Analysis. Refer to the "Configuring Administrative Parameters for Spectrum Analysis" section on page 6-2 for a full list of such parameters, and instructions for changing related settings.

Parameters in the GUNSLINGER.INI File

The GUNSLINGER.INI file can be used to configure CBT 3.4 features for the following parameters:

- The value of the ShowPasswordWhileEditing parameter may be set to YES or NO to hide or to display password text, when the password is highlighted. Refer to the "Disabling or Enabling Password Text Display" section on page 3-12 for additional information.
- The UseHostName parameter may be set to to YES or NO to support hostname-based communications between Applet and Servlet on the network, in addition to sustaining IP-based communications, as with the prior CBT 3.3. Refer to the "Mapping Hostnames from Applet to Servlet" section on page 6-1 for additional information.

Retrieving Subscriber or Provisioning Data by Using an External Interface

With CBT you can set up an external interface to retrieve subscriber or provisioning data:

- **Subscriber information**—Data such as a customer's name and account number
- **Provisioning information**—Data such as the IP address for a customer's cable modem and the IP address for the Cisco CMTS on which the modem is located

You can use the following methods to set up an external interface:

- **Script**—Can be used for subscriber or provisioning information or both
- **Application on an HTTP server**—Can be used for subscriber or provisioning information or both
- **LDAP**—Can be used for subscriber or provisioning information or both
- **BPR**—Can be used for provisioning information only
- **CNR**—Can be used for provisioning information only
- **Local Database**—(Default) Can be used for subscriber or provisioning information or both

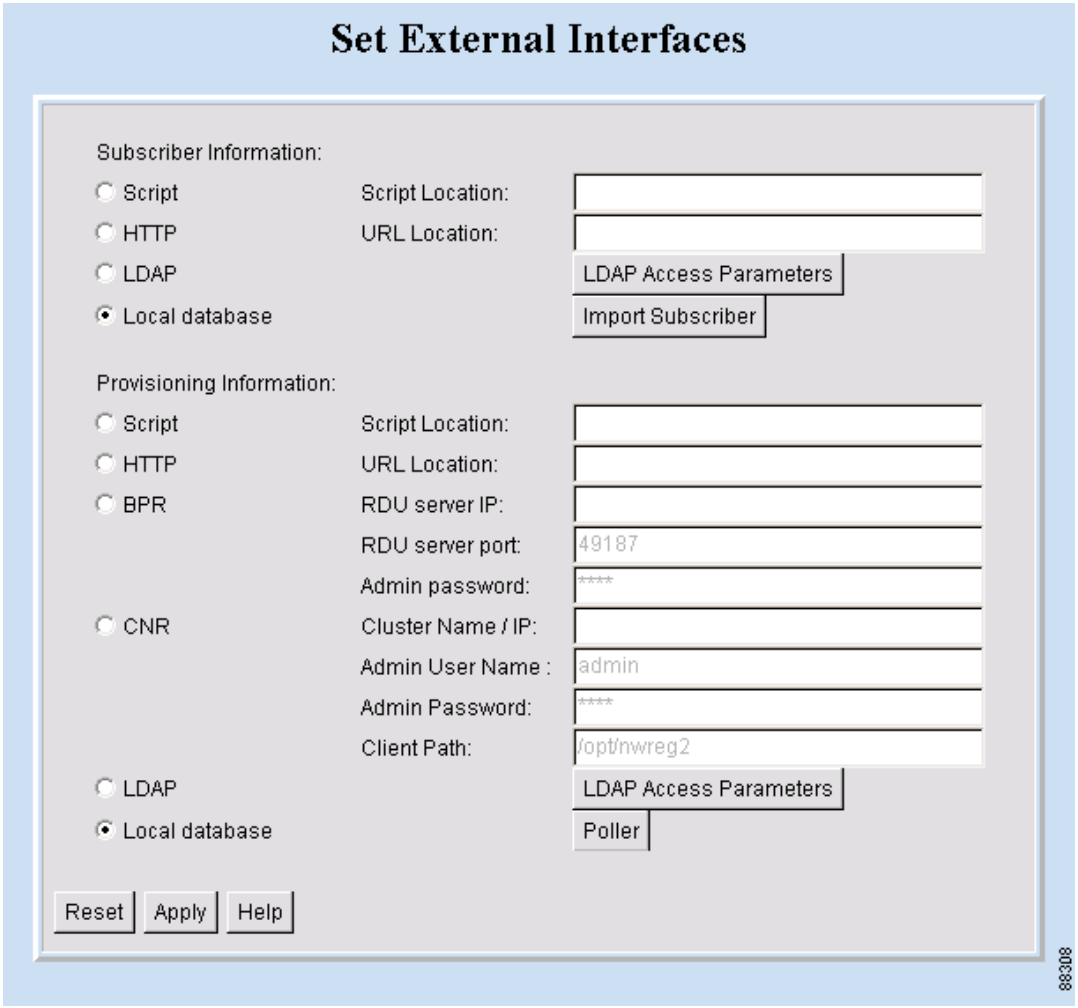
Note If you set an external interface and CBT does not find subscriber or provisioning data there, CBT reverts to the local database.

Figure 3-10 shows the dialog box where you specify the method. The remainder of this section describes how to set up an external interface with each method, referring to Figure 3-10.

This section contains the following topics:

- Retrieving Data with a Script, page 3-25
- Retrieving Data with an Application on an HTTP Server, page 3-28
- Retrieving Data with LDAP, page 3-30
- Retrieving Data with BPR, page 3-30
- Retrieving Data with CNR, page 3-31
- Retrieving Subscriber Data from the Local Database, page 3-32
- Retrieving Provisioning Data from the Local Database, page 3-33

Figure 3-10 Set External Interfaces Dialog Box



Retrieving Data with a Script

You can implement an external data retrieval application by using a scripting language. To see a sample script that you can modify according to your needs, see "Shell Script for Retrieving Subscriber or Provisioning Information" section on page 8-1.

Note You can also implement an external data retrieval application in programming languages such as PL/SQL, C/C++, and Java, as long as it is embedded in a shell script.

Script Parameters for MAC Address Information

The script must support the ability to search for a cable modem's MAC address by entering the customer's phone number or IP address. The input parameters for the script are:

- *subscriber-info-script-name* GET_MAC PHONE *phone-number*
- *provision-info-script-name* GET_MAC IP *ip-address*

For example, the following script invokes a search for the modem's MAC address by using the phone number 408-123-4567:

```
/opt/tools/subscriber-query.sh GET_MAC PHONE 4081234567
```

The next script invokes a search for the modem's MAC address by using the IP address 172.2.3.1:

```
/opt/tools/provision-query.sh GET_MAC IP 172.2.3.1
```

Note A MAC address query by phone number invokes the subscriber information script. A MAC address query by IP address invokes the provisioning information script.

The output of the script must be in the following format:

OUT_DATA=MAC followed by one or more MAC addresses (non-dotted format) separated by ^

The following examples show the output for one and two MAC addresses:

```
OUT_DATA=MAC^001c64ff23ef^
OUT_DATA=MAC^001c64ff23ef^013e45ed1245^
```

Script Parameters for Subscriber Information

If you use a script to query subscriber information, it needs to support the ability to search for subscriber data, such as name and account number, by entering the MAC address of the subscriber's cable modem. The input parameters for the script are:

script-name GET_SUBSCRIBER MAC *non-dotted-modem-mac-address*

For example, the following script invokes a search for subscriber data by using the MAC address 001c.ab23.45fe in a non-dotted format:

```
/opt/tools/subscriber-query.sh GET_ SUBSCRIBER MAC 001cab2345fe
```

The output of the script must be in the following format, separated by ^:

OUT_DATA=SUBSCRIBER^*Field1=value1^Field2=value2^Field3=value3^Field4=value4^Field 5=value5*^

The following example shows the output for a subscriber's data:

```
OUT_DATA=SUBSCRIBER^Account Number=123456^Name=Doe_John^Address=123 Tasman, San Jose,
CA 93443^Phone=4081234567^Class of Service=N/A^FiberNode=FIBER_1^Customer
Since=1999^Account Status=Paid
```

Note In the Real-Time Modem Status Report, CBT displays subscriber information in the order in which the fields are returned from the script.

Script Parameters for Provisioning Information

If you use a script to query provisioning information, it needs to support the ability to search for provisioning data, such as CMTS and cable modem IP addresses, by entering the MAC address of a cable modem. The input parameters for the script are:

script-name GET_PROVISION MAC *non-dotted-modem-mac-address*

For example, the following script invokes a search for provisioning data by using the MAC address 001c.ab23.45fe:

```
/opt/tools/subscriber-query.sh GET_PROVISION MAC 001cab2345fe
```

The output of the script must be in the following format:

```
OUT_DATA=PROVISION^cmts-ip-address^cm-ip-address^
```

The following example shows the output for provisioning data:

```
OUT_DATA=PROVISION^127.23.45.1^127.23.127.5^
```

Error Handling for Script

If an error occurs within the script or the embedded application called by the script, the script should return the output in the following format:

OUT_DATA=ERROR^*error-message*^

The following example shows output for an error:

```
OUT_DATA=ERROR^Unable to query the subscriber database.^
```

In the graphical user interface (GUI), CBT displays the error message to the user in a message box.

Retrieving Data with a Script

To retrieve subscriber or provisioning information by using a script:

Step 1 Review the sample script and modify it according to your needs. See the "Sample Script Code" section on page 8-4.

Step 2 From the Configuration menu, choose **External Interface** to specify one or both of the following:

- In the Subscriber Information section, click **Script** and enter the **Script Location** of the file.
- In the Provisioning Information section, click **Script** and enter the **Script Location** of the file.

Step 3 For a complete description of each field in the External Interface dialog box, click **Help**.

Troubleshooting Script Problems

See the "Troubleshooting the Script" section on page 8-3.

Retrieving Data with an Application on an HTTP Server

You can implement an external data retrieval application by using an application running on an HTTP server. CBT sends the request to the HTTP server by using the POST method. To see a sample application file that you can modify according to your needs, see "Java Code for Retrieving Subscriber or Provisioning Information" section on page 8-4.

HTTP Parameters for MAC Address Information

The application running on an HTTP server must support the ability to search for a cable modem's MAC address by entering the customer's phone number or IP address. CBT sends the following parameters to the appropriate server:

- REQUEST=GET_MAC
- SEARCH_TYPE=*PHONE or IP*
- IN_PARAM=*phone* or *ip*

For example, the following parameters search for the modem's MAC address by using the phone number 408-123-4567:

```
REQUEST=GET_MAC
SEARCH_TYPE=PHONE
IN_PARAM=4081234567
```

The response from the server must be in the following format:

OUT_DATA=MAC followed by one or more MAC addresses separated by ^

The following examples show the output for one and two MAC addresses:

```
OUT_DATA=MAC^001c64ff23ef^
OUT_DATA=MAC^001c64ff23ef^013e45ed1245^
```

Note When the SEARCH_TYPE is PHONE, the subscriber information URL is used. When the SEARCH_TYPE is IP, the provisioning information URL is used.

HTTP Parameters for Subscriber Information

If you use an application running on an HTTP server to query subscriber information, it needs to support the ability to search for subscriber data, such as name and account number, by entering the MAC address of the subscriber's cable modem. CBT sends the following parameters to the server:

- REQUEST=GET_SUBSCRIBER
- SEARCH_TYPE=MAC
- IN_PARAM=*non-dotted-modem-mac-address*

For example, the following parameters search for subscriber data by using the MAC address 001cab2345fe:

```
REQUEST=GET_SUBSCRIBER
SEARCH_TYPE=MAC
IN_PARAM=001cab2345fe
```

The output of the script must be in the following format, separated by ^:

OUT_DATA=SUBSCRIBER^*Field1=value1^Field2=value2^Field3=value3^Field4=value4^Field5=value5^*

The following example shows the output for a subscriber's data:

```
OUT_DATA=SUBSCRIBER^Account Number=123456^Name=Doe_John^Address=123 Tasman, San Jose,
CA 93443^Phone=4081234567^Class of Service=N/A^FiberNode=FIBER_1^Customer
Since=1999^Account Status=Paid
```

Note In the Real-Time Modem Status Report, CBT displays subscriber information in the order in which the fields are returned from the script.

HTTP Parameters for Provisioning Information

If you use an application running on an HTTP server to query provisioning information, it needs to support the ability to search for provisioning data, such as CMTS and cable modem IP addresses, by entering the MAC address of the cable modem. CBT sends the following parameters to the server:

- REQUEST=GET_PROVISION
- SEARCH_TYPE=MAC
- IN_PARAM=*non-dotted-modem-mac-address*

For example, the following parameters search for provisioning data by using the MAC address 001cab2345fe:

```
REQUEST=GET_PROVISION
SEARCH_TYPE=MAC
IN_PARAM=001cab2345fe
```

The response from the server must be in the following format:

OUT_DATA=PROVISION^*cmts-ip-address*^*cm-ip-address*^

The following example shows the output for provisioning data:

```
OUT_DATA=PROVISION^127.23.45.1^127.23.127.5^
```

Error Handling for HTTP

If an error occurs within the application running on an HTTP server, the script should return the output in the following format:

OUT_DATA=ERROR^*error-message*^

The following example shows output for an error:

```
OUT_DATA=ERROR^Unable to query the subscriber database.^
```

In the GUI, CBT displays the error message to the user in a message box.

Retrieving Data with an HTTP Application

To retrieve subscriber or provisioning information with an application on an HTTP server:

Step 1 Review the sample application file and modify it according to your needs. See the "Sample Java Code" section on page 8-8.

Step 2 From the Configuration menu, choose **External Interface** to specify one or both of the following:

- In the Subscriber Information section, click **HTTP** and enter the **URL Location** of the file.
- In the Provisioning Information section, click **HTTP** and enter the **URL Location** of the file.

Step 3 For a complete description of each field in the External Interface dialog box, click **Help**.

Troubleshooting HTTP Problems

See "Troubleshooting the Sample HTTP Application" section on page 8-8.

Retrieving Data with LDAP

You can use Lightweight Directory Access Protocol (LDAP) to retrieve external subscriber or provisioning information. LDAP is a nonproprietary, standards-based protocol. If you use the LDAP method, see your LDAP server administrator to obtain the information about the LDAP server that CBT requires.

CBT 3.4 supports LDAP Version 2.0, which was tested with Netscape LDAP Server Version 4.0 and iPlanet Version 5.1.

To retrieve external subscriber or provisioning information by using LDAP:

Step 1 From the Configuration menu, choose **External Interface**.

Step 2 Depending on the type of information you want to retrieve, do one or both of the following:

- To retrieve subscriber information, in the Subscriber Information section select **LDAP** and click **LDAP Access Parameters** to open the LDAP Access Parameters dialog box.
- To retrieve provisioning information, in the Provisioning Information section select **LDAP** and click **LDAP Access Parameters** to open the LDAP Access Parameters dialog box.

Step 3 To fill in the LDAP Access Parameters dialog box, see your LDAP server administrator to obtain the required information and click **Help** for directions on how to fill in each of the LDAP fields that CBT requires.

Retrieving Data with BPR

You can retrieve provisioning information only by using Cisco Broadband Provisioning Registrar (BPR). BPR automates provisioning and configuration tasks. CBT supports BPR 2.0. For product information, refer to:

http://www.cisco.com/en/US/products/sw/netmgtsw/ps529/index.html

Note BPR has recently been renamed as Cisco Broadband Access Center for Cable (BACC). You will see the new name on Cisco.com and in future releases. The CBT 3.4 GUI and documentation use the current name and acronym.

To retrieve provisioning information by using BPR:

Step 1 From the Configuration menu, choose **External Interface**.

Step 2 In the Provisioning Information section, click **BPR**. To interact with BPR, CBT requires the following information:

- RDU Server IP Address
- RDU Server Port Number
- Admin Password

Step 3 For information on each field, click **Help**.

Retrieving Data with CNR

You can retrieve provisioning information only by using Cisco Network Registrar (CNR). CNR is an application that provides scalable Domain Name System (DNS), Trivial File Transfer Protocol (TFTP), and Dynamic Host Configuration Protocol (DHCP) services. For product information, refer to:

http://www.cisco.com/en/US/products/sw/netmgtsw/ps1982/index.html

If you use CNR, the following caveats apply to the setup:

- Either the CNR GUI or server component must be installed on the CBT server.
- The CNR server component can be installed on the same server as CBT or on a different server. If it is installed on a separate server, the CBT server must know the location of that server.
- CBT currently works with one CNR at a time.
- CBT supports CNR 5.0.3 and 5.5.4.

Note CNR has recently been renamed to Cisco CNS Network Registrar (CNS-NR). You will see the new name on Cisco.com and in future releases. The CBT 3.4 GUI and documentation use the current name and acronym.

To retrieve provisioning information by using CNR:

Step 1 From the Configuration menu, choose **External Interface**.

Step 2 In the Provisioning Information section, click **CNR**. To interact with CNR, CBT requires the following information:

- Cluster Name/IP Address
- Admin User Name
- Admin Password
- Client Path

Step 3 For information on each field, click **Help**.

Retrieving Subscriber Data from the Local Database

CBT can retrieve subscriber data from its own local Sybase database after the data has been added in CBT. After installation, the local database is the default method for retrieving subscriber data.

To retrieve subscriber information from the local database:

Step 1 Add subscriber information by following the directions in the "Adding Subscriber Information" section on page 3-16.

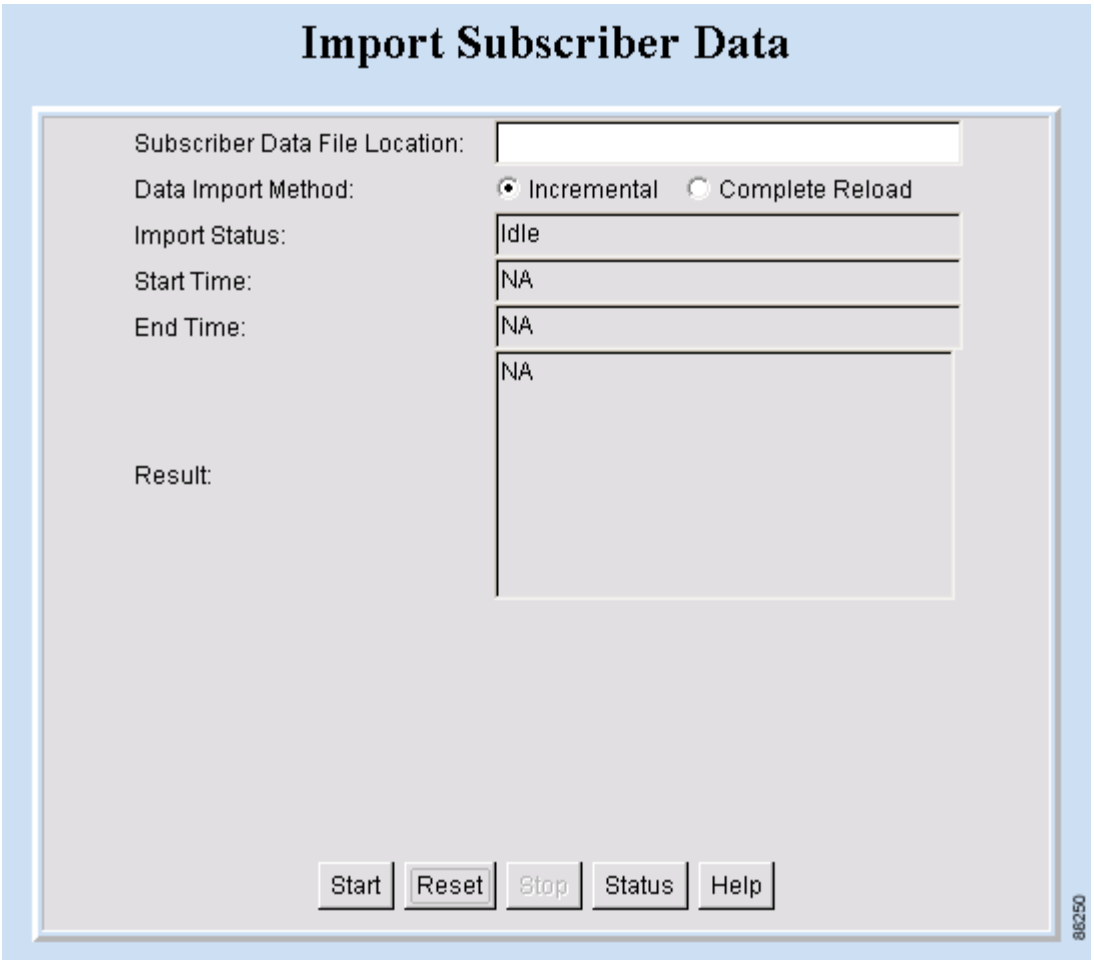
Step 2 From the Configuration menu, choose **External Interface**.

Step 3 In the Subscriber Information section, click **Local Database**.

Step 4 (Optional) To update or replace the existing subscriber information, click **Import Subscriber** to open the Import Subscriber Data dialog box, as shown in Figure 3-11.

Step 5 In the dialog box, click **Help** for directions on how to import subscriber data.

Figure 3-11 Import Subscriber Data Dialog Box



Retrieving Provisioning Data from the Local Database

CBT can retrieve provisioning data from its own local Sybase database after the data has been added to it. After installation, the local database is the default method for retrieving provisioning data. To store provisioning information in its local database, CBT polls CMTSs using SNMP. CBT provides two ways to do this:

- **The Poller Scheduler dialog box**—Use this method if you prefer to work in the CBT GUI.
- **A separate Java poller application**—Use this method, referred to as the Poller, if you prefer to work with a command-line interface.

Tip If you are using CBT on the Windows operating system, we recommend that you use the CBT GUI method.

About the Poller

When you install CBT, the default method for retrieving provisioning information is the local database. Because the Poller is part of the process for this method, it is on by default after installation. The default schedule for the Poller is to run at or about midnight and to repeat every 24 hours. You can change these settings according to your own needs.

To query Cisco CMTSs using the Poller, you specify two parameters:

- ***delta-start-time***—When the Poller should start to poll the CMTSs. Expressed in hours, the delta is the difference between when you schedule the polling and when the polling begins.
- ***poll-interval***—The interval, expressed in hours, between polling sessions.

The syntax to start the Poller on Linux or Solaris is:

/opt/CSCOcbt/bin/start_poller *delta-start-time poll-interval*

The following example shows a polling session that starts 8 hours after you schedule it and repeats every 24 hours:

```
/opt/CSCOcbt/bin/start_poller 8 24
```

The following example shows how to stop the Java Poller application on Linux or Solaris:

```
/opt/CSCOcbt/bin/stop_poller
```

Note There is no impact to CBT if you turn the Poller off.

At each polling interval, the Poller accesses the CBT routers list file and performs SNMP queries to each CMTS using its SNMP read-only community string. The CBT routers list file is in the following locations:

- **Linux**—/opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes/config
- **Solaris**—/opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes/config
- **Windows**—C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\WEB-INF\classes\config

In addition to the on-demand polling that you schedule, CBT submits a polling request when it detects that a change has been made to the list of routers that CBT manages. For example, CBT submits a polling request when it detects that a new chassis has been added to a router.

Note The more frequent the polling interval, the more SNMP traffic that is generated to the CMTSs. This means that CPU utilization on the CMTS increases when polling is in progress. The impact on the CBT server side is minimal.

To retrieve provisioning information from the local database:

Step 1 Add provisioning data by following the directions in the "Adding Subscriber Information" section on page 3-16.

Step 2 From the Configuration menu, choose **External Interface.**

Step 3 In the Provisioning Information section, click **Local Database.**

Scheduling the Poller

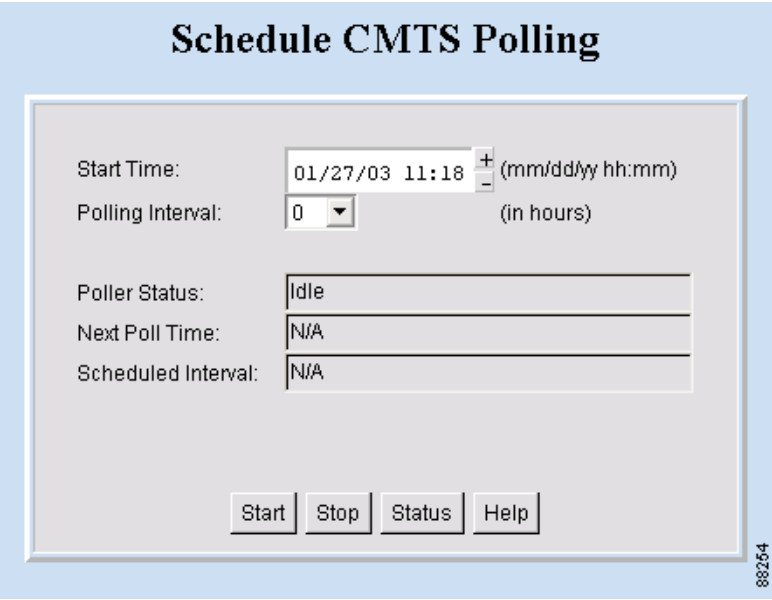
To schedule when and how often the Poller populates the local database with provisioning information that it gathers from CMTSs, you can use the GUI or type a command.

To use the GUI, from the Configuration menu, choose **Poller**. The Schedule CMTS Polling dialog box appears, as shown in Figure 3-12. In this dialog box, click **Help** for directions on how to schedule the Poller.

To use a command, type the following command and specify the parameters in hours, as explained in the previous section:

```
/opt/CSCOcbt/bin/start_poller delta-start-time poll-interval
```

Figure 3-12 Schedule CMTS Polling Dialog Box



Getting Summary Information and a Detailed Real-Time Status Report for a Modem

To get summary information on a cable modem, you can locate the modem by entering the MAC address. If you retrieve subscriber or provisioning data from the local database, you can also locate the modem by using any of the following advanced search criteria:

- Phone number
- Account ID
- Name
- Street address
- Zip code

The CBT advanced search feature also allows you to use the asterisk (*) for wildcard searches. For example, if you enter 1234* for the account ID, you get a list of all cable modems for subscribers who have account numbers that begin with 1234.

Note The CBT advanced search criteria applies to the local database only. If you set up an external interface for subscriber or provisioning data, you must locate a modem by entering its MAC address.

To get detailed information on a cable modem, you can get a real-time modem status report, as shown in Figure 3-13. The information for a cable modem comes from three sources:

- The CMTS, which is the router that directs network traffic.
- The cable modem itself.
- The provider's subscriber database, where each provider can specify the information that appears. Typical information would be the customer's name, phone number, and account number. However, the information is not limited to that.

Within the status report, the following colors are used in certain fields to indicate performance:

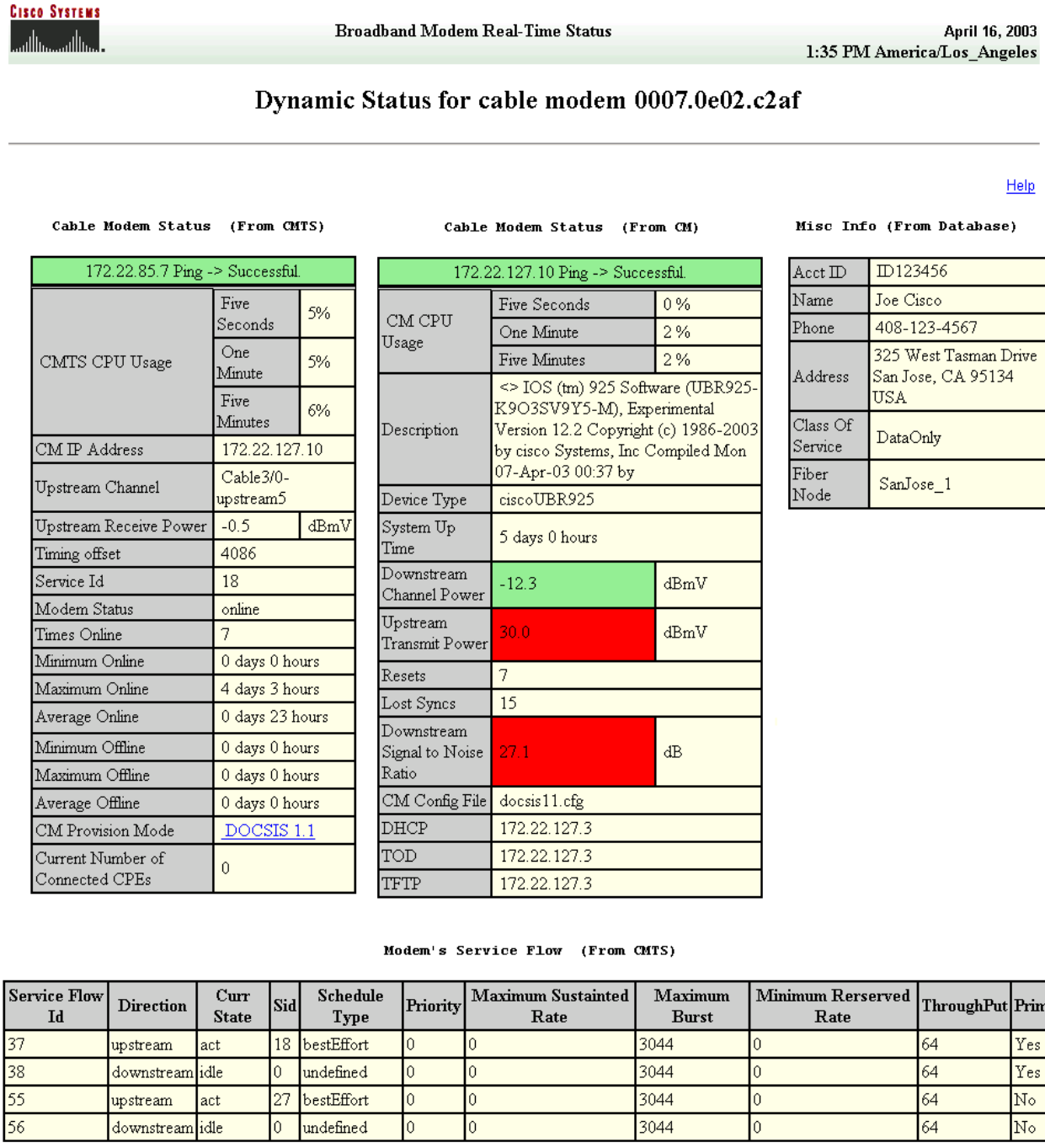
- **Green**—Indicates a value is within the allowed threshold.
- **Red**—Indicates a value is above or below the allowed threshold.

For more information on setting thresholds, see the "Modem Status Thresholds" section on page 3-19.

This section contains the following topics:

- Real-Time Modem Status Report Data Display
- Getting Summary Information on a Modem, page 3-37
- Getting a Detailed Real-Time Status Report for a Modem, page 3-38
- Showing the Flap List Analysis, page 3-40
- Scheduling the Flap List Analysis, page 3-42
- Setting Parameters for the Flap List, page 3-43
- Purging the Flap List, page 3-44

Figure 3-13 Real-Time Modem Status Report Data Display



Broadband Modem Real-Time Status

April 16, 2003
1:35 PM America/Los_Angeles

Dynamic Status for cable modem 0007.0e02.c2af

Help

Cable Modem Status (From CMTS)

172.22.85.7 Ping -> Successful.		
CMTS CPU Usage	Five Seconds	5%
	One Minute	5%
	Five Minutes	6%
CM IP Address	172.22.127.10	
Upstream Channel	Cable3/0-upstream5	
Upstream Receive Power	-0.5	dBmV
Timing offset	4086	
Service Id	18	
Modem Status	online	
Times Online	7	
Minimum Online	0 days 0 hours	
Maximum Online	4 days 3 hours	
Average Online	0 days 23 hours	
Minimum Offline	0 days 0 hours	
Maximum Offline	0 days 0 hours	
Average Offline	0 days 0 hours	
CM Provision Mode	DOCSIS 1.1	
Current Number of Connected CPEs	0	

Cable Modem Status (From CM)

172.22.127.10 Ping -> Successful.		
CM CPU Usage	Five Seconds	0 %
	One Minute	2 %
	Five Minutes	2 %
Description	<> IOS (tm) 925 Software (UBR925-K9O3SV9Y5-M), Experimental Version 12.2 Copyright (c) 1986-2003 by cisco Systems, Inc Compiled Mon 07-Apr-03 00:37 by	
Device Type	ciscoUBR925	
System Up Time	5 days 0 hours	
Downstream Channel Power	-12.3	dBmV
Upstream Transmit Power	30.0	dBmV
Resets	7	
Lost Syncs	15	
Downstream Signal to Noise Ratio	27.1	dB
CM Config File	docsis11.cfg	
DHCP	172.22.127.3	
TOD	172.22.127.3	
TFTP	172.22.127.3	

Misc Info (From Database)

Acct ID	ID123456
Name	Joe Cisco
Phone	408-123-4567
Address	325 West Tasman Drive San Jose, CA 95134 USA
Class Of Service	DataOnly
Fiber Node	SanJose_1

Modem's Service Flow (From CMTS)

Service Flow Id	Direction	Curr State	Sid	Schedule Type	Priority	Maximum Sustainted Rate	Maximum Burst	Minimum Rerserved Rate	ThroughPut	Primary
37	upstream	act	18	bestEffort	0	0	3044	0	64	Yes
38	downstream	idle	0	undefined	0	0	3044	0	64	Yes
55	upstream	act	27	bestEffort	0	0	3044	0	64	No
56	downstream	idle	0	undefined	0	0	3044	0	64	No

88917

Getting Summary Information on a Modem

To get summary information on a modem:

Step 1 From the Hotline Tools menu, choose **Locate Modem**. The Locate Broadband Modems dialog box appears, as shown in Figure 3-14.

Step 2 Use one of the following methods to locate a modem:

- If you know the complete MAC address, enter one or more MAC addresses and click **Start**.
- If you are using the local database for subscriber or provisioning data and want to search by using other criteria, such as phone number or wildcard strings, click **Advanced Search**. The Locate A Broadband Modem dialog box appears, as shown in Figure 3-15.

Step 3 For a complete description of each field in these dialog boxes, click **Help**.

Figure 3-14 Locate Broadband Modems Dialog Box

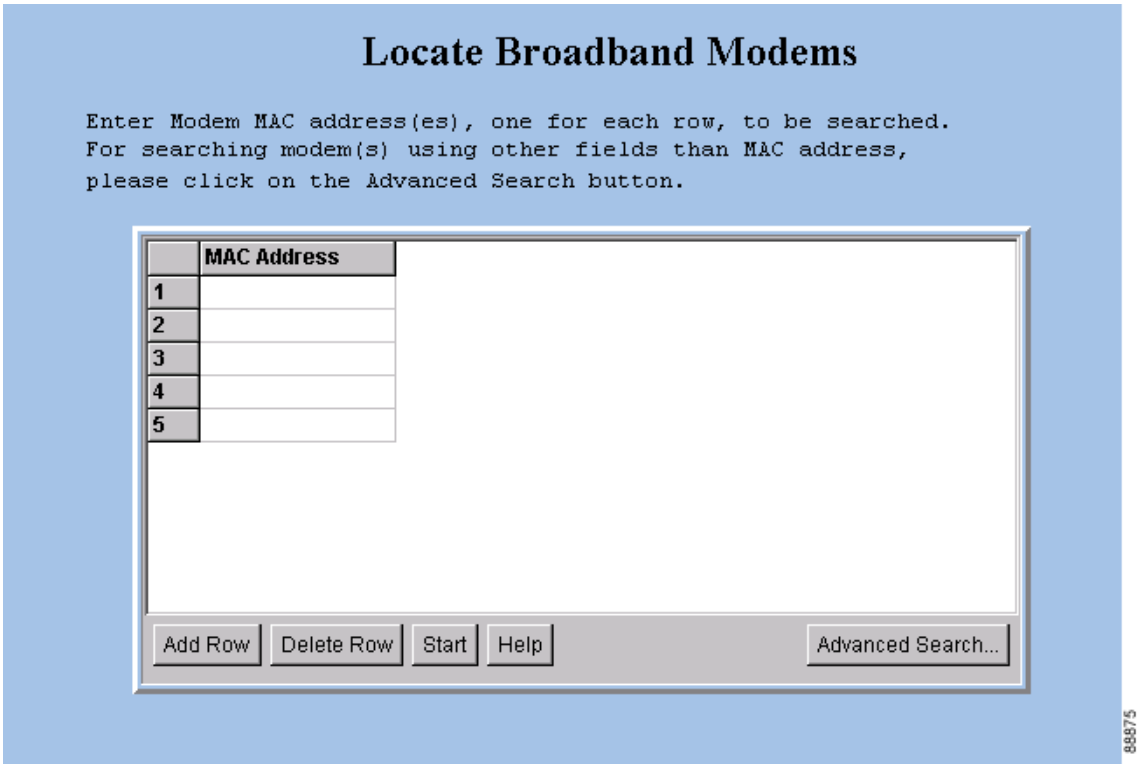
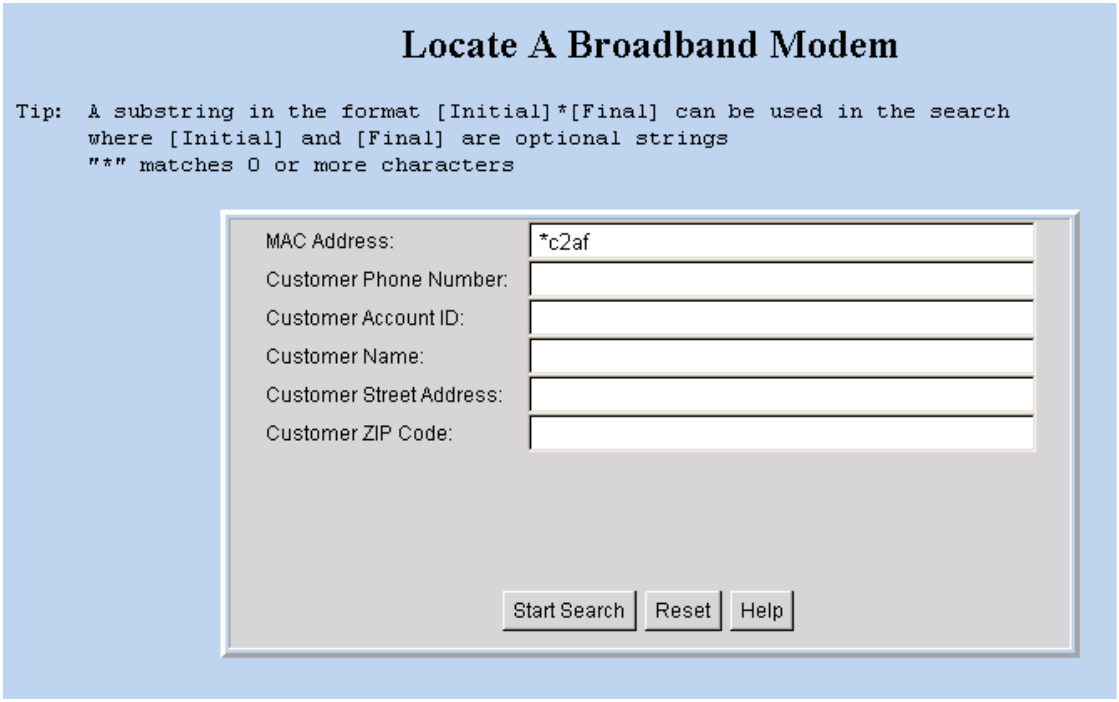


Figure 3-15 Locate A Broadband Modem Dialog Box



Getting a Detailed Real-Time Status Report for a Modem

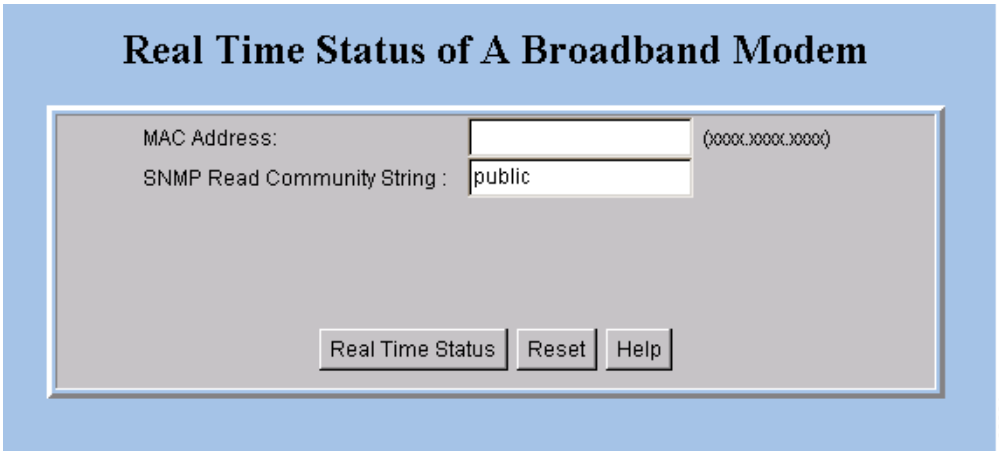
To get a detailed status report for a cable modem:

Step 1 From the Hotline Tools menu, choose **Modem Status**. The Real Time Status of A Broadband Modem dialog box appears, as shown in Figure 3-16.

Step 2 Enter the MAC address for the cable modem for which you want to see a status report and click **Real Time Status**. The Real-Time Modem Status Report for that modem appears, as shown in Figure 3-13.

Step 3 For a complete description of each field in the dialog box and the Real-Time Modem Status Report, click **Help**.

Figure 3-16 Real Time Status of a Broadband Modem Dialog Box



Reviewing the CMTS Dashboard

To get summary information on a Cisco CMTS, review the CMTS Dashboard dialog box.

For the CMTS that you select, the CMTS Dashboard provides the following information:

- Hostname
- IP address
- Interface/Upstream
- Counter reset time
- Signal-to-noise ratio (SNR)
- Minimum receive power level of all modems associated with that CMTS
- Maximum receive power level of all modems associated with that CMTS
- Percentage of time that the modem stays online
- Number of modems that are online
- Total number of modems that connect to the interface
- Router name

For printing or collaborative troubleshooting, you can generate an HTML version of the information.

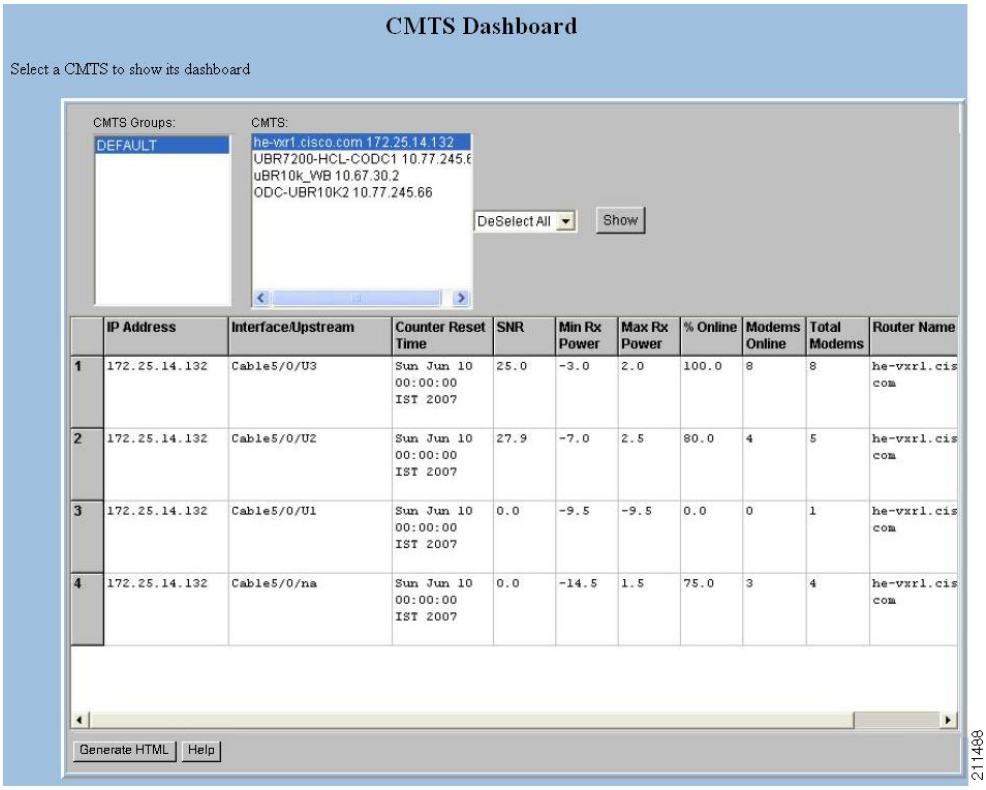
To access the CMTS Dashboard:

Step 1 From the Hotline Tools menu, choose **CMTS Dashboard**.

Step 2 For a complete description of each field in the CMTS Dashboard dialog box, click **Help**.

Figure 3-17 shows the CMTS Dashboard.

Figure 3-17 CMTS Dashboard Dialog Box



Showing and Configuring the Flap List Analysis

To analyze the results of data captured from one or more Cisco CMTSs, review the Flap List Analysis Details display. The analysis shows the number of modems with the following problems:

- Provisioning—The issue could be at one of the provisioning servers.
- Reverse path noise—The issue could be from trouble in the cable plant.
- Reverse path attenuation—The issue could be from a power adjustment that the subscriber makes.
- Packet corruption—The issue could be from degradation on the network.

Note You can not get a flap list analysis for a CMTS that is not from Cisco.

Showing the Flap List Analysis

To show a flap list for one or more CMTSs:

Step 1 From the Diagnostics menu, choose **Flap List Analysis**.

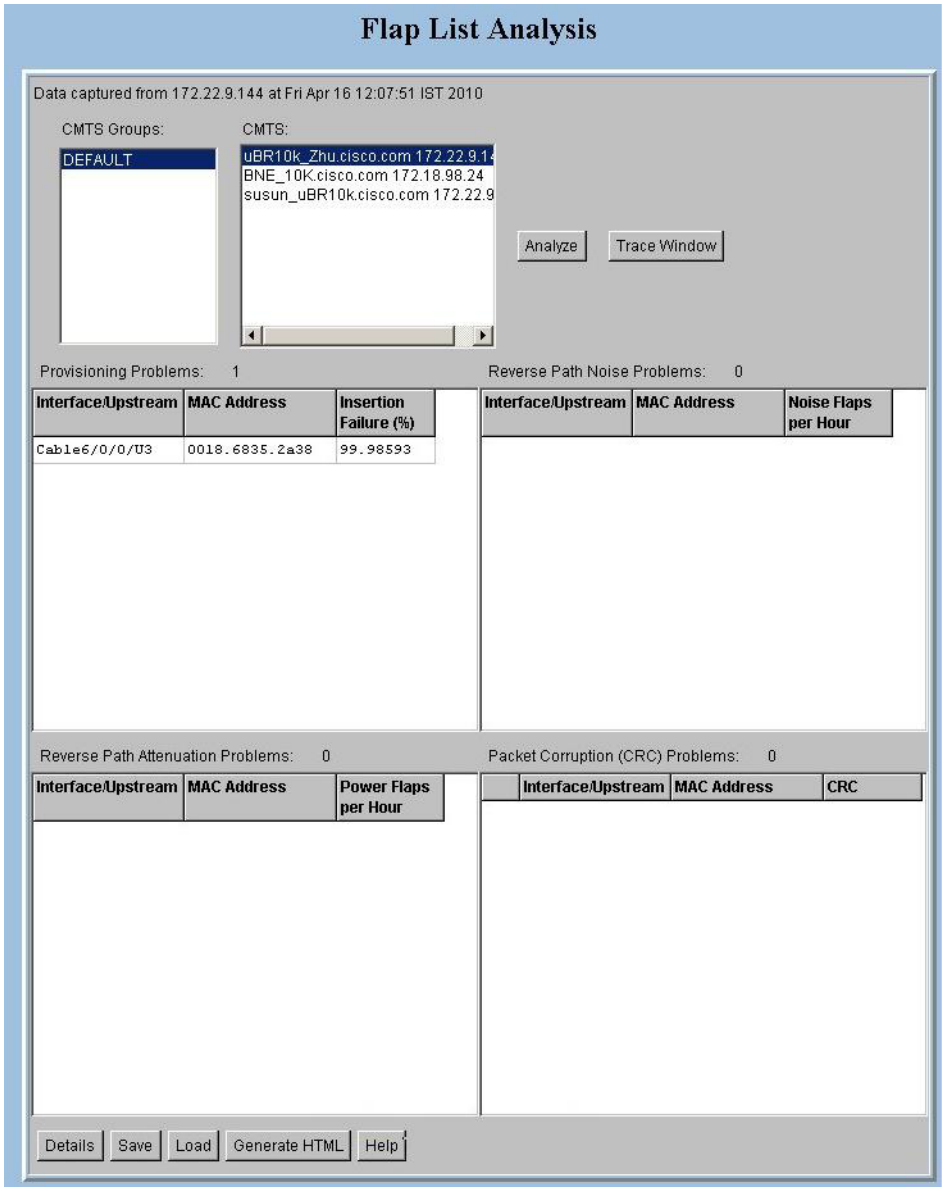
Step 2 Select one or more Cisco CMTSs to analyze and click **Analyze**.

Step 3 (Optional) To see a lower level of detail, click **Details**.

Step 4 For a complete description of each field in the Flap List Analysis dialog boxes, click **Help**.

Figure 3-18 shows the Flap List Analysis information display, which you can sort according to your needs by clicking a column heading. For example, to sort the provisioning problems by MAC address, click the MAC Address column heading.

Figure 3-18 Flap List Analysis Details Display

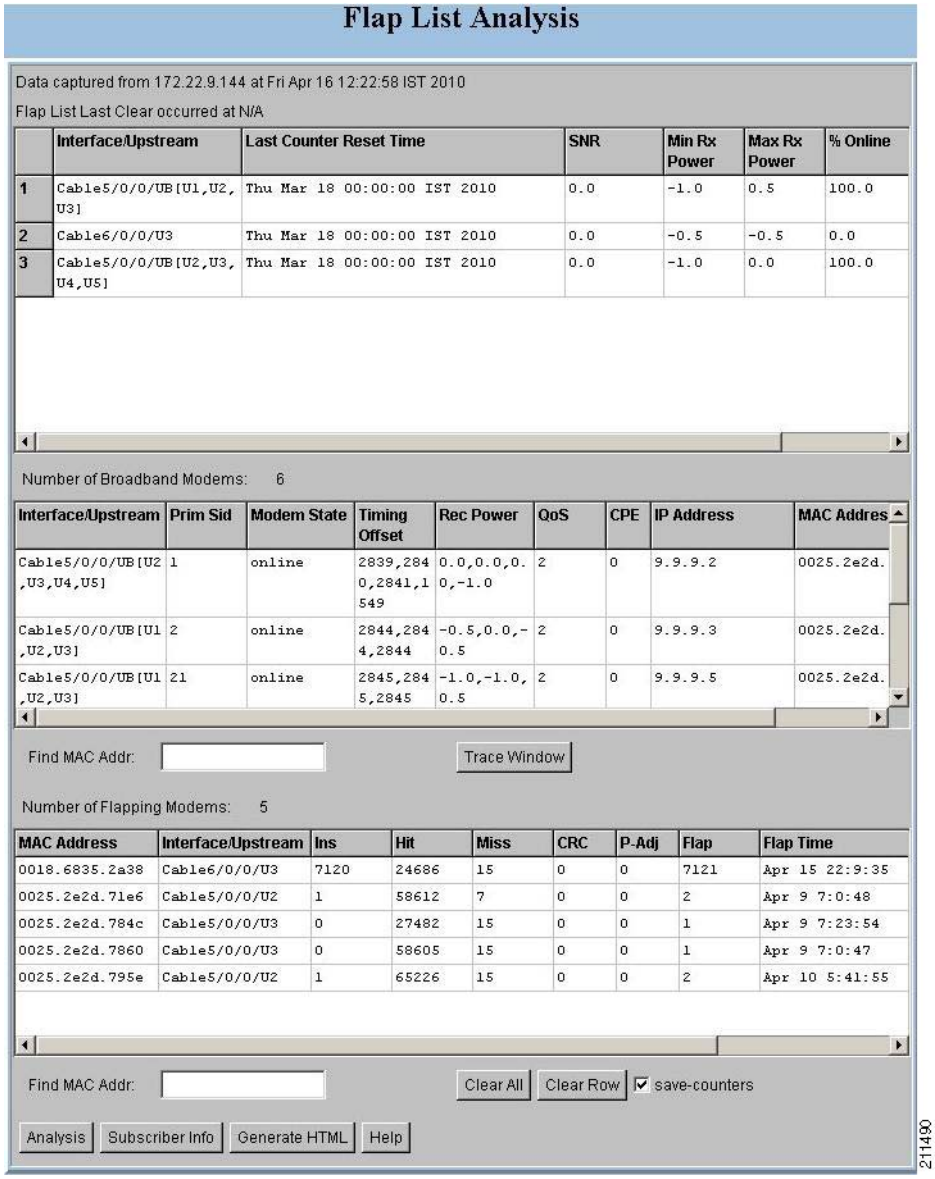


From the Flap List Analysis Details information display, you can see a lower level of detail that contains the following information:

- General status of each interface
- Total number of modems reported by the router and output for each one
- Number of flapping modems that are reported and output for each one

The detailed level of the Flap List Analysis detail display is shown in Figure 3-19.

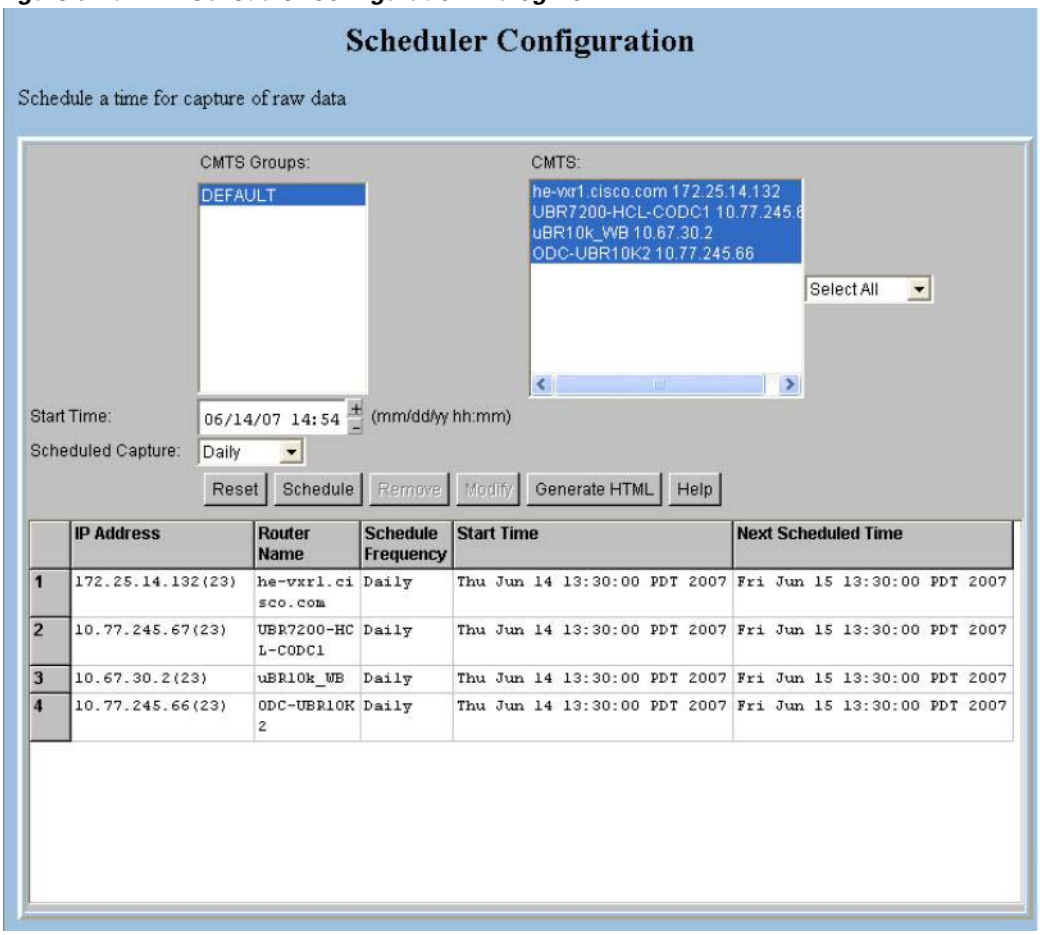
Figure 3-19 Flap List Analysis Details Display



Scheduling the Flap List Analysis

To schedule a time to capture raw data from one or more Cisco CMTSs, use the Scheduler Configuration dialog box, which is shown in Figure 3-20.

Figure 3-20 Scheduler Configuration Dialog Box



To schedule a time to capture raw data from one or more Cisco CMTSs:

Step 1 From the Configuration menu, choose **Scheduler > Flap List**.

Step 2 Select one or more Cisco CMTSs from which you want to capture data and click **Schedule**.

Step 3 Click **OK** in the message box once Cisco CMTSs are added.

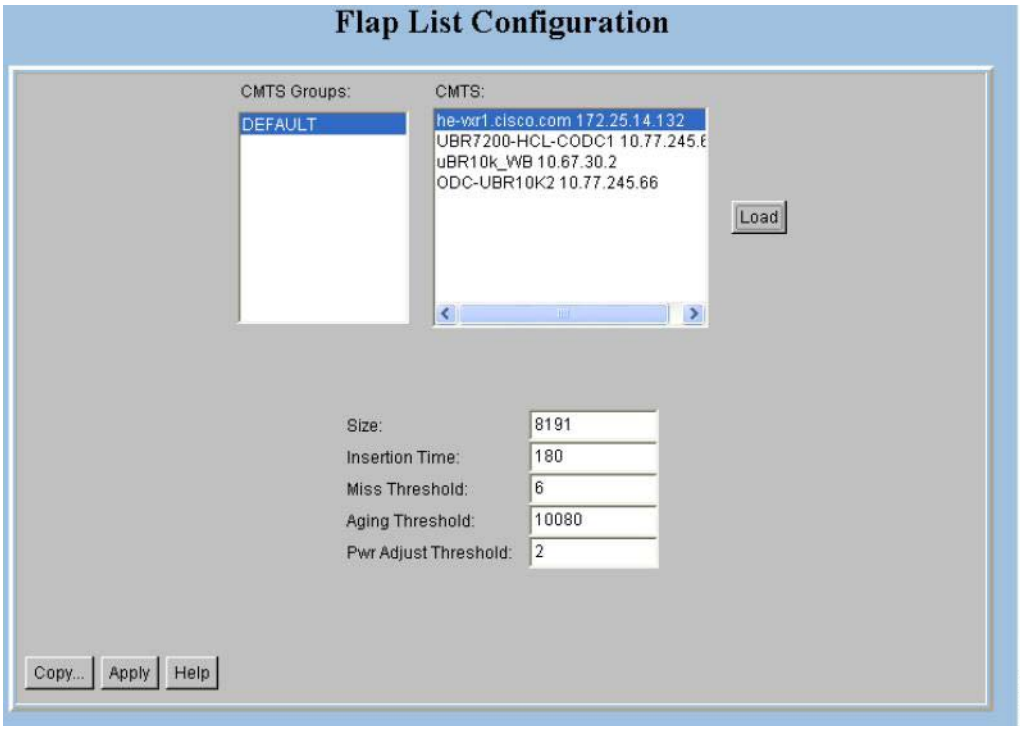
Step 4 For a complete description of each field in the Scheduler Configuration dialog box, click **Help**.

Setting Parameters for the Flap List

The administrator can set parameters for the flap list in the Flap List Configuration dialog box shown in Figure 3-21.

Note The RFTECH user type does not have access to the Flap List Configuration dialog box.

Figure 3-21 Flap List Configuration Dialog Box



For a Cisco CMTS, the administrator can specify the following criteria for a flap list:

- Maximum number of rows that the flap list can have. Increase the default of 100 if you think you have more than 100 flapping modems.
- Insertion time, in seconds, at which a cable modem should be included in the flap list because it has not completed registration within that time.
- Miss threshold, which is the number of consecutive times a cable modem can fail to respond to a keepalive poll from the Cisco CMTS before it should be included in the flap list.
- Aging threshold, which is the number of minutes a cable modem is allowed between flaps before it should be included in the flap list.
- Power adjust threshold, in decibels, that a cable modem cannot exceed without being included in the flap list.

To set parameters for the flap list:

Step 1 From the Configuration menu, choose **Flap List Parameters**.

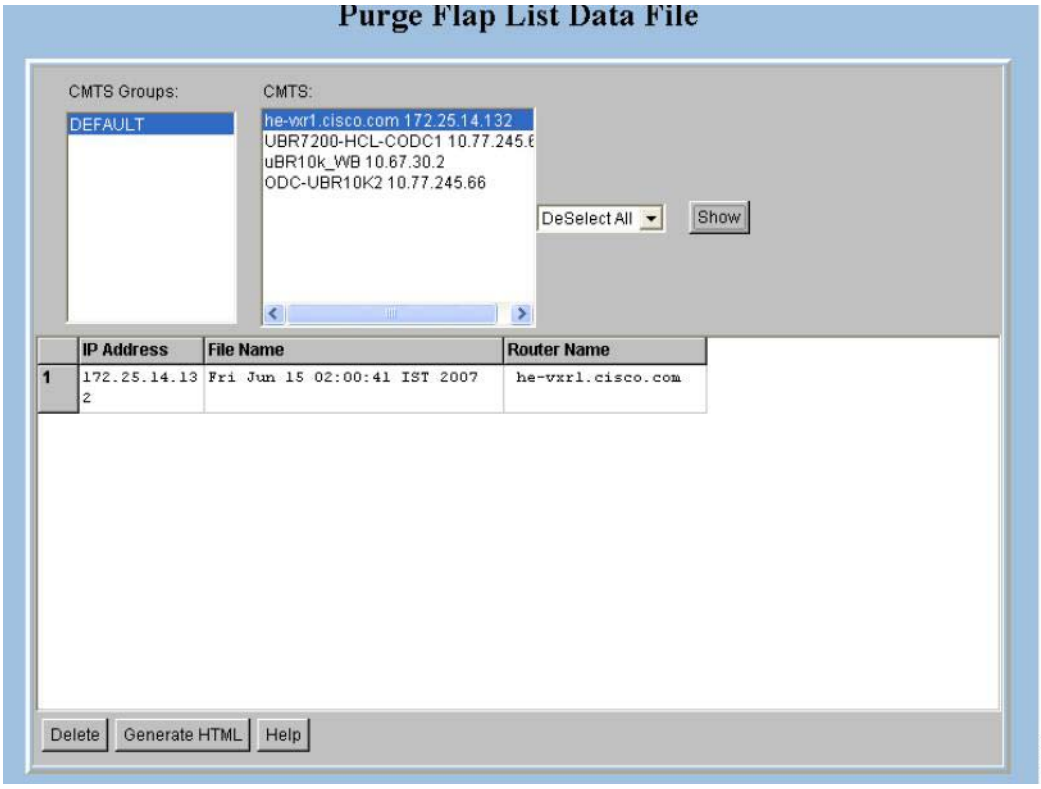
Step 2 Select the Cisco CMTS for which you want to set parameters and click **Load**.

Step 3 For a complete description of each field in the Flap List Configuration dialog box, click **Help**.

Purging the Flap List

When you want to delete a saved flap list file, use the Purge Flap List Data File dialog box, which is shown in Figure 3-22. Before you purge a flap list file, you can generate an HTML version of it.

Figure 3-22 Purge Flap List Data File Dialog Box



Perform these steps to purge a flap list file:

Step 1 From the Utilities menu, choose **Purge Saved Data > Flap List**.

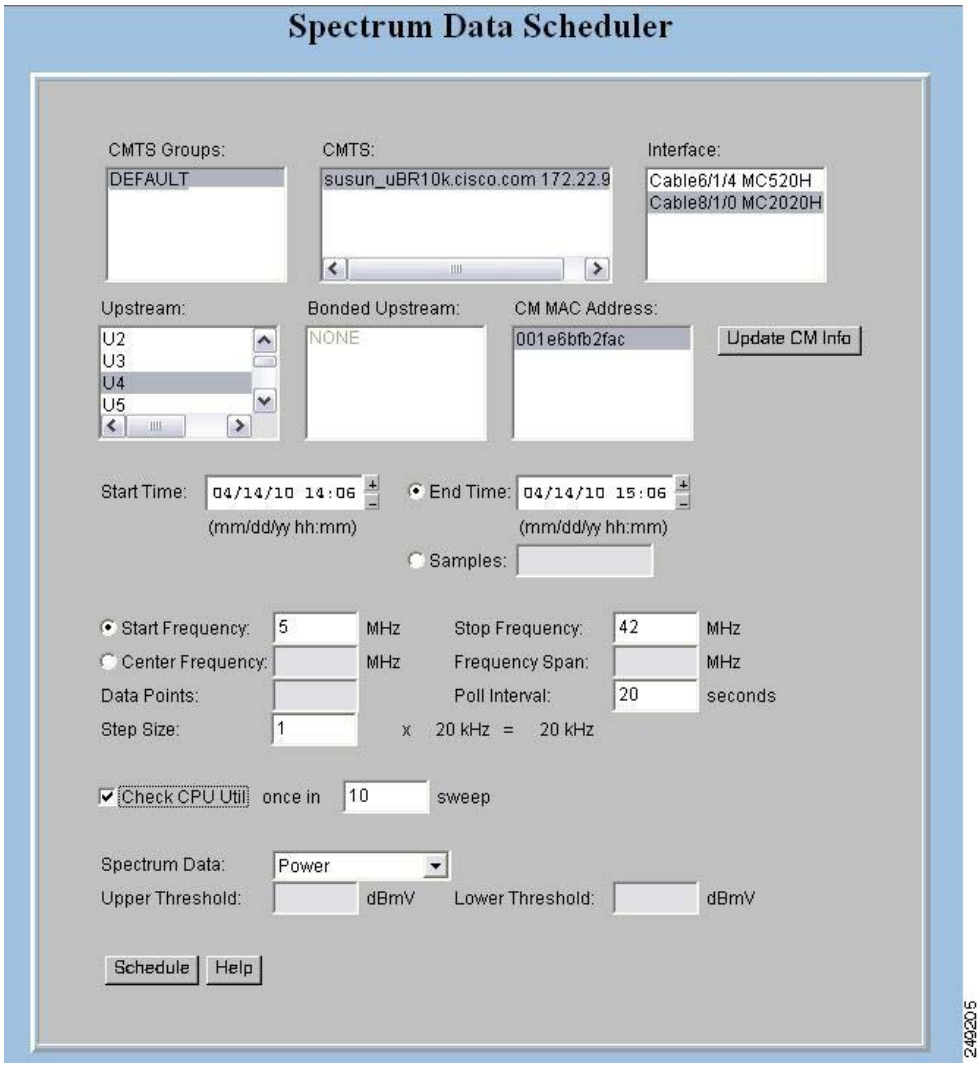
Step 2 Select one or more Cisco CMTSs for which you want to show saved flap list files and click **Show**.

Step 3 For a complete description of each field in the Purge Flap List Data File dialog box, click **Help**.

Scheduling the Spectrum Data

To schedule a time to capture spectrum data from one or more Cisco CMTSs, use the Spectrum Data Scheduler dialog box, which is shown in Figure 3-23.

Figure 3-23 Spectrum Data Scheduler Dialog Box



To schedule a time to capture spectrum data from one or more Cisco CMTSs:

Step 1 From the Configuration menu, choose **Scheduler > Spectrum Data**. The Spectrum Data Schedule dialog box appears.

Step 2 Select one or more Cisco CMTSs from where you want to capture data and click **Schedule**.

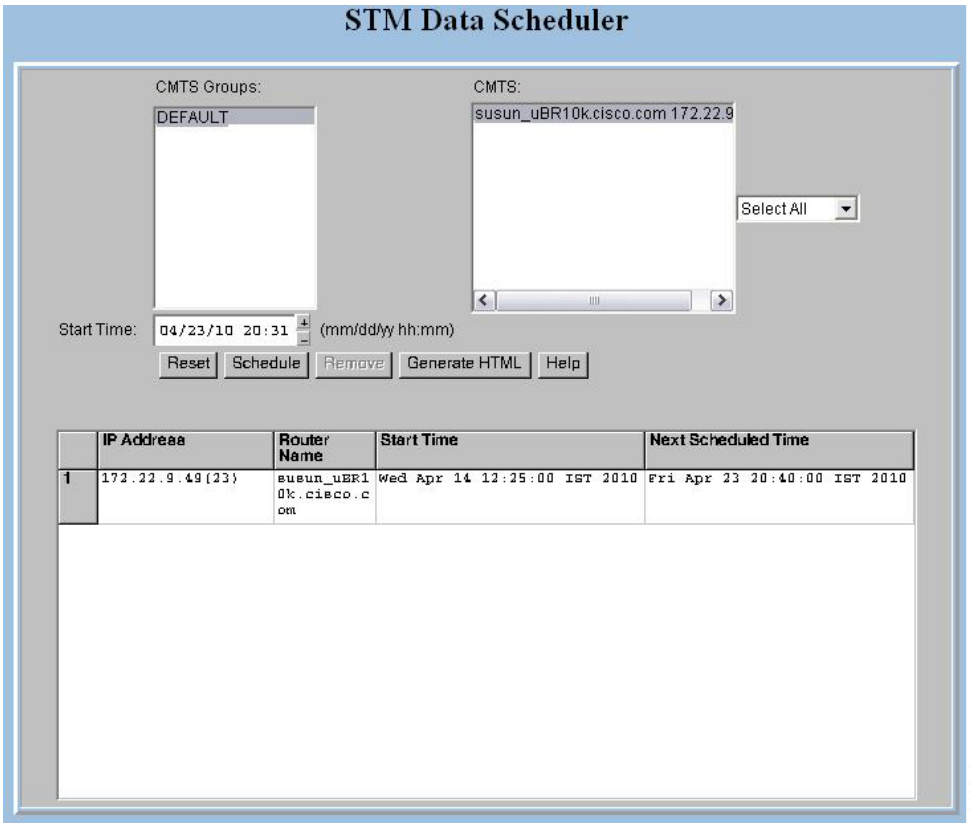
Step 3 Click **OK** in the message box once Cisco CMTSs are added.

Step 4 For a complete description of each field in the Spectrum Data Scheduler dialog box, click **Help**.

Scheduling the STM Data

To schedule a time to capture STM data from one or more Cisco CMTSs, use the STM Data Scheduler dialog box, which is shown in Figure 3-24.

Figure 3-24 STM Data Scheduler Dialog Box



To schedule time to capture spectrum data from one or more Cisco CMTSs:

Step 1 From the Configuration menu, choose **Scheduler > STM Data**. The STM Data Scheduler dialog box appears.

Step 2 Select one or more Cisco CMTSs from which you want to capture data and click **Schedule**.

Step 3 Click **OK** in the message box once Cisco CMTSs are added.

Step 4 For a complete description of each field in the Scheduler Configuration dialog box, click **Help**.

Generic Data Query

To poll the CMTS for individual MIB variables, use the Generic Query diagnostic tool, which is shown in Figure 3-25. You can choose MIB variables from the MIBs that are preloaded in the tool, or if the variable you need is not available, you can load a new MIB into the tool and then access the variable through the tool.

To query the MIB variables:

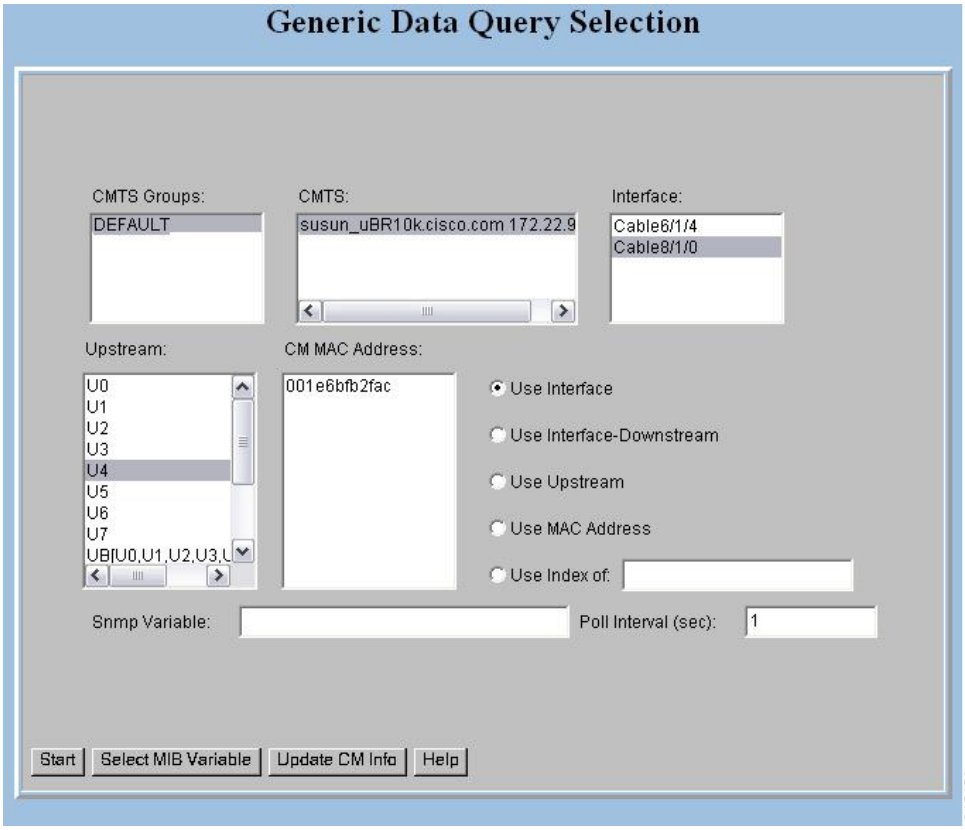
Step 1 Click **Update CM Info** to update the cable modem information on the selected CMTS instead of getting data from the cache.

Step 2 Select a CMTS.

Step 3 Enter a MIB variable in the SNMP Variable field and click **Start**.

Step 4 For a complete description of each field, click **Help**.

Figure 3-25 Generic Data Query Dialog Box



CHAPTER 4

Using the Spectrum Management Tools

CBT 3.4 interfaces with the Cisco CMTS to provide return path spectrum analysis. The following tasks allow you to use the spectrum management tool effectively:

- **Trace Window**—Monitors power and noise levels for a selected modem or upstream port.
- **Spectrogram**—Monitors power and noise levels over time for a selected modem or upstream port.
- **CNR Analysis**—Retrieves the carrier-to-noise ratio (CNR) for selected modems or upstream ports.
- **CNR Trending**—Retrieves CNR trending data for selected modems or upstream ports.
- **Data Playback**—Reviews a saved Trace Window or CNR Analysis.
- **Polling Status**—Displays the spectrum analysis in progress on a client machine and to view the scheduled spectrum polling events.

Note CBT 3.4 supports upstream channel bonding. The bonded upstream channel set of the DOCSIS 3.0 modems is listed in the Upstream field.

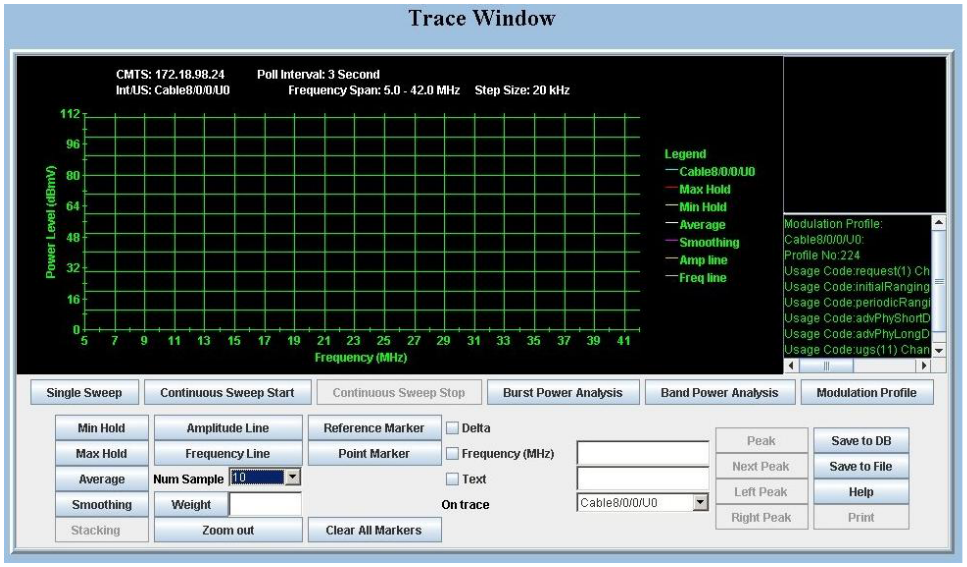
Each task is covered in the following sections:

- Using the Trace Window, page 4-2
- Trace Window Display
- Viewing Trace Windows, page 4-5
- Sorting Support in the CBT 3.4 Graphical User Interface, page 4-6
- Using Auto-Select in the Trace Window, page 4-6
- Using the Spectrogram, page 4-7
- Analyzing the Carrier-to-Noise Ratio, page 4-9
- Retrieving CNR Trending Data, page 4-11
- Playing Back Data, page 4-12
- Playing Back CNR Analysis, page 4-15
- Playing Back CNR Trending Data, page 4-17
- Displaying Polling Status, page 4-19
- Enabling Instantaneous CPU Assessment for Spectrum Polling, page 4-20

Using the Trace Window

To monitor power and noise levels for a selected modem or upstream port, use the Trace Window. As shown in Figure 4-1, the Trace Window shows the output that you would see in a spectrum analyzer. This output displays in the plot line of the Trace Window.

Figure 4-1 Trace Window Display



In the Trace Window, you can:

- Monitor the power and noise levels two ways:
  - **Single sweep**—Provides a snapshot view of levels at one point in time, as shown in Figure 4-1
  - **Continuous sweep**—Provides a real-time view of data as it changes
- Check two types of power levels and display the data in the results block, which is in the lower right corner as shown in Figure 4-1:
  - **Burst power analysis**—Acquires the upstream channel power
  - **Band power analysis**—Acquires the band power measurement for a start and stop frequency that you select
- View the Modulation profile information as shown in Figure 4-1:
  - **Modulation profile**—Displays the modulation profile number, usage code, and the channel type of an upstream.
- Indicate a variety of lines and points in the plot line and display the data in the annotations block, which is in the upper right corner, as shown in Figure 4-1
  - **Reference marker**—Is a unique marker in the plot line and is indicated by r, as shown in Figure 4-1
  - **Point markers**—Is a way to indicate one or more points in the plot line; each point marker is numbered consecutively beginning with p1, as shown in Figure 4-1
  - **Amplitude and frequency lines**—Display a line marker in the plot line, as shown by the horizontal amplitude line at 0 dBmV and the vertical frequency line just before 14000 kilohertz (kHz) in Figure 4-1

- Show additional plot lines:
  - **Minimum hold**—Display the plot line that shows the minimum power level
  - **Maximum hold**—Display the plot line that shows the maximum power level
- Move a marker that you created to peaks in the plot line:
  - **Peak**
  - **Next peak**
  - **Left peak**
  - **Right peak**
- Customize the placement and position of point markers:
  - **Text**—You can customize a point marker's label by entering a name in the Text box. For example, you could change p1 to Test Point 1 and p2 to Test Point 2, as shown in Figure 4-1.
  - **Frequency (kHz)**—You can specify a frequency in kilohertz for the point marker by entering a number in the Frequency text box. For example, you could enter 20200 instead of clicking on the plot line near 20000.
- Show the delta between the reference marker's power level and that of each point marker.
- Zoom in on a selected frequency range to enlarge the view.
- Save a single sweep of the plot line and play it back.

Working with the Trace Window

Table 4-1 shows several ways to work with data in the Trace Window.

Table 4-1 Keyboard Actions for Working with the Trace Window

Task	Keyboard Sequence
To move a line marker	Press **Shift,** click and drag the line to the desired location.
To zoom in on a selected frequency	Move your cursor to the area you want to enlarge and press **Ctrl,** left-click. or Press **Ctrl,** left-click and drag your cursor to create a box that indicates the area you want to enlarge.
To return to a full view from a zoom view	Press **r**.

Starting the Trace Window

To use the Trace Window to monitor power and noise levels for a selected modem or upstream port:

Step 1 From the Spectrum Tools menu, choose **Trace Window**. The Trace Window Criteria dialog box appears, as shown in Figure 4-2.

Step 2 Select an upstream or cable modem for which you want to monitor power and noise levels.

Note To update the cable modem information on the selected CMTS instead of getting data from the cache, click **Update CM Info**. This real-time update varies according to how many modems are attached to the CMTS. The update depends on the number of modems attached to the Cisco CMTS; the update takes longer if more number of modems are attached to the Cisco CMTS.

Tip When you select the resolution bandwidth (RBW), keep in mind that the smaller the RBW, the less variance and the more accurate the frequency is in the Trace Window is. Specifically, when a single-tone signal that resembles a noise burst appears in the Trace Window, the frequency of the signal gets shifted. This causes the frequency in the Trace Window to be different from the spectrum analyzer output. The variance in frequency increases as the resolution bandwidth increases.

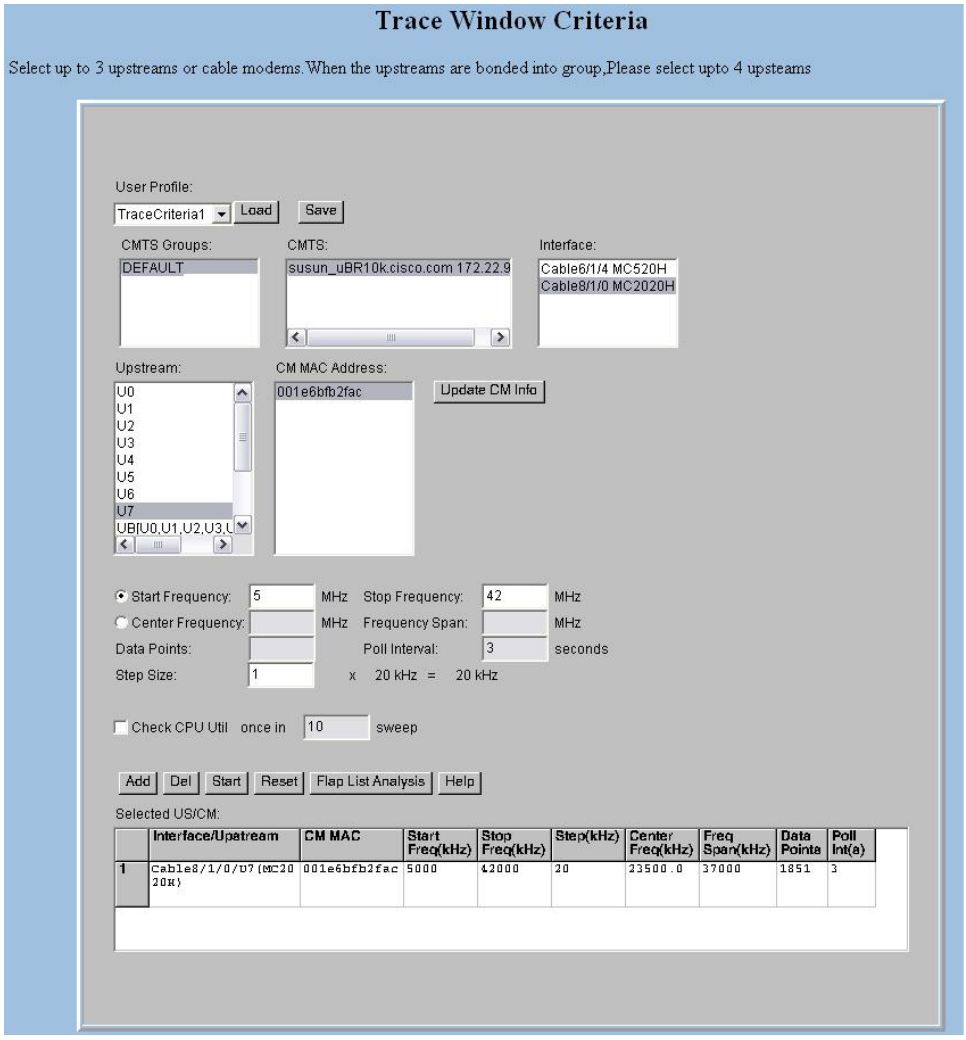
Step 3 Click **Start**. The Trace Window appears, as shown in Figure 4-1.

Step 4 To view the return path data and activate all options in the Trace Window, click one of the following buttons:

- **Single Sweep**—Provides a single snapshot of power and noise levels
- **Continuous Sweep Start**— Starts to monitor for real-time measurements of power and noise levels

Step 5 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-2 Trace Window Criteria Dialog Box



Viewing Trace Windows

Some browsers give the user the ability to stop windows from being launched by the browser. CBT 3.4 normally launches a new window to display the trace pop-up spectrum data charts. If the data chart window does not appear and the browser is configured to stop new windows from launching, you can configure CBT 3.4 to display data charts within the browser window.

To display data charts within the browser window:

Step 1 From the Configuration menu, choose **System**.

Step 2 In the System Configuration screen, uncheck the **Use Popup** check box.

Sorting Support in the CBT 3.4 Graphical User Interface

CBT 3.4 supports sorting for the List fields in the following GUI pages:

- Spectrum Data Scheduler
- Trace Window Criteria
- Spectrogram Criteria
- CNR Analysis Criteria
- CNR Trending Criteria

To change the order of sorting:

Step 1 Select the desired desired page:

- **Configuration > Scheduler > Spectrum Data > Spectrum Data Scheduler**
- **Spectrum Tools > Trace Window > Trace Window Criteria**
- **Spectrum Tools > Spectrogram > Spectrogram Criteria**
- **Spectrum Tools > CNR Analysis > CNR Analysis Criteria**
- **Spectrum Tools > CNR Trending > CNR Trending Criteria**

Step 2 Click on the label of the desired field, and data is sorted in ascending or descending order. Each click of the label reverses the current order displayed.

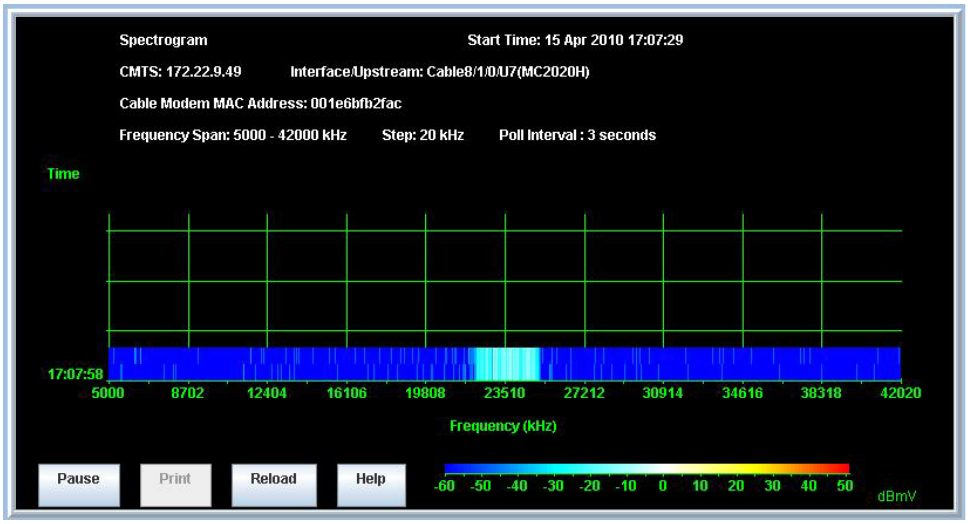
Using Auto-Select in the Trace Window

CBT 3.3 and later do not support auto-select of any given field in the Trace Window by default. To retrieve and display information for any given field in the Trace Window display, select the desired field.

Using the Spectrogram

To monitor power and noise levels, as you do in the Trace Window, while viewing the added dimension of time, use the Spectrogram. As shown in Figure 4-3, the Spectrogram shows the output that you would see in a spectrum analyzer.

Figure 4-3 Spectrogram



In the Spectrogram, you see the following variables in one easy, 3-D view:

- **Power level**—Measured in decibels millivolt (dBmV) and depicted according to the color key at the bottom. As shown in Figure 4-3, the power level is cyan, which the color key indicates is in the -30 to -10 dBmV range.
- **Frequency**—Measured in kilohertz and shown on the X axis. In Figure 4-3, the power level is between 30867 and 34563 kHz.
- **Time**—Measured in hours, minutes, and seconds (HH:MM:SS) and shown on the Y axis. Figure 4-3 shows the latest time that data was captured is 15:18:39.

When you move your cursor over the Spectrogram, the status bar at the bottom of the window displays the time, frequency, and power level for the current location. In the Spectrogram, you can click **Pause** or **Print** at anytime.

Starting the Spectrogram

To use the Spectrogram to monitor power and noise levels for a selected modem or upstream port over time:

Step 1 From the Spectrum Tools menu, choose **Spectrogram**. The Spectrogram Criteria dialog box appears, as shown in Figure 4-4.

Step 2 Select the Cisco CMTS, the interface, and an upstream or cable modem for which you want to monitor power and noise levels over time.

Note To update the cable modem information on the selected CMTS instead of getting data from the cache, click **Update CM Info**. This real-time update varies according to how many modems are attached to the CMTS. The update depends on the number of modems attached to the Cisco CMTS; the update takes longer if more number of modems are attached to the Cisco CMTS.

Step 3 Click **Start**. The Spectrogram appears, as shown in Figure 4-3.

Step 4 After completion of criteria selections, click **Start**, and the Spectrogram Criteria Confirmation dialog box appears. Click **Yes** to apply the changes.

Step 5 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-4 Spectrogram Criteria Dialog Box

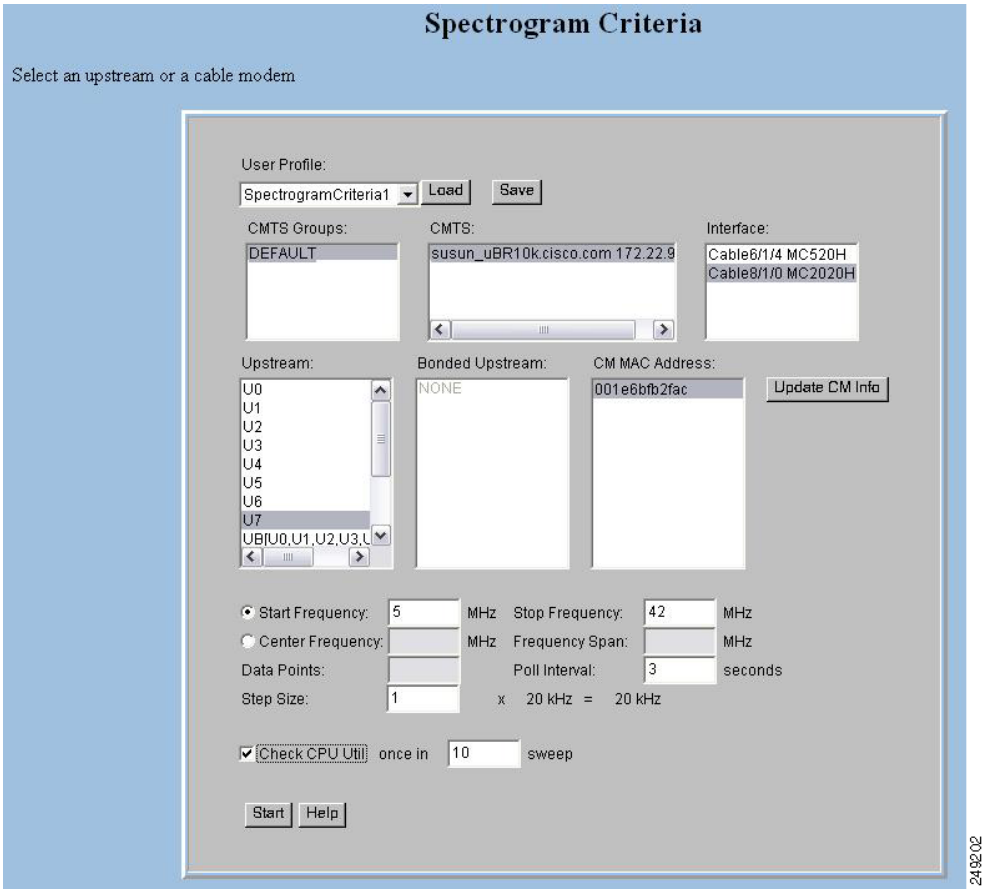
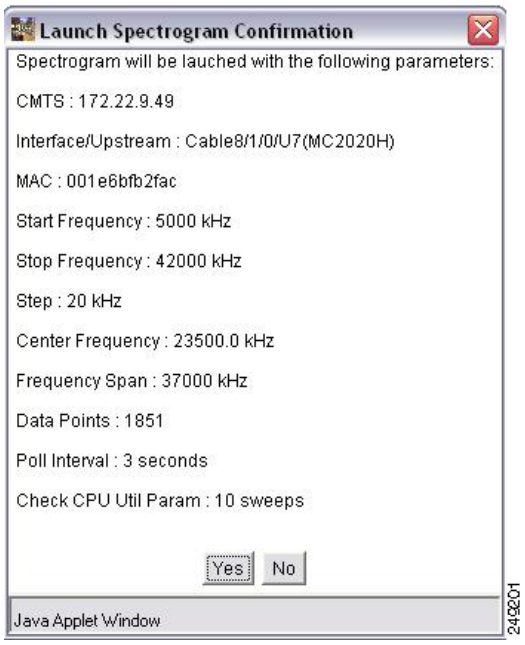


Figure 4-5 Spectrogram Criteria Confirmation Dialog Box

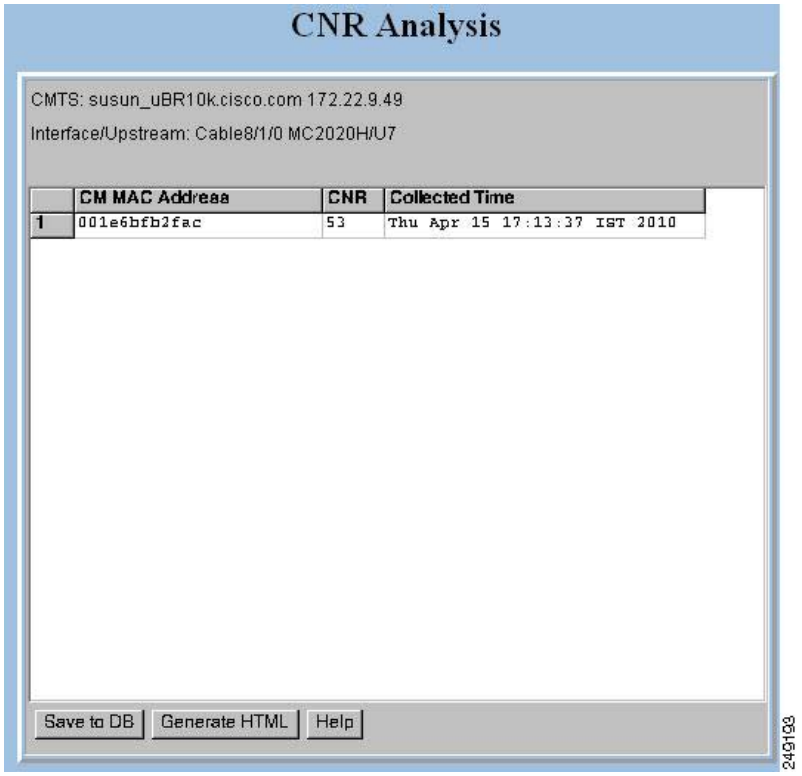


Analyzing the Carrier-to-Noise Ratio

To show the carrier-to-noise ratio for selected cable modems or upstream ports, use the CNR Analysis task. As shown in Figure 4-6, this information displays in the CNR Analysis dialog box, which lets you:

- View the following information:
  - CMTS and interface/upstream
  - CM MAC address
  - Carrier-to-noise ratio
  - Collected time
- Save the results to play back later.

Figure 4-6 CNR Analysis Dialog Box



You can set the following SPECTRUM.INI file parameters:

- Set thresholds for acceptable carrier-to-noise ratios
- Specify colors for CNRs that are below two thresholds, floor and ceiling

In Figure 4-6, the CNRs for the selected modems fall into three categories:

- **Yellow**—The CNR is below the ceiling threshold.
- **Red**—The CNR is below the floor threshold.
- **White**—The CNR is above the ceiling threshold.

For more information on the SPECTRUM.INI file, see the "Parameters in the SPECTRUM.INI File" section on page 3-22.

Getting the CNR Analysis

To show the carrier-to-noise ratio for selected cable modems or upstream ports:

Step 1 From the Spectrum Tools menu, choose **CNR Analysis**. The CNR Analysis Criteria dialog box appears, as shown in Figure 4-7.

Step 2 Select one or more upstreams or cable modems for which you want to see the carrier-to-noise ratio.

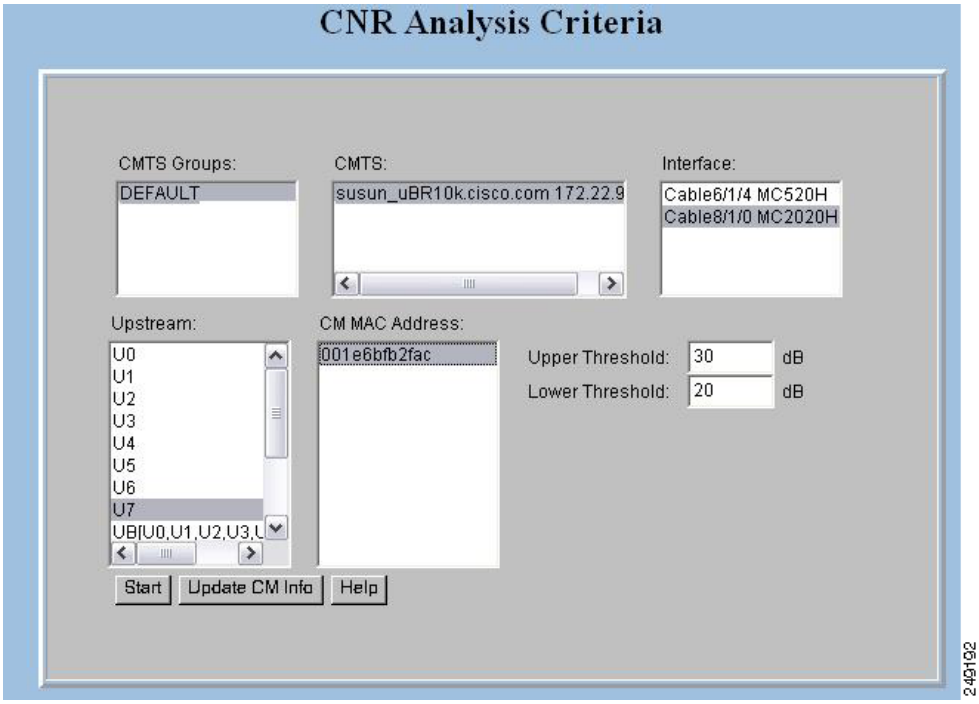
Note To update the cable modem information on the selected CMTS instead of getting data from the cache, click **Update CM Info**. This real-time update varies according to how many modems are attached to the CMTS. The update depends on the number of modems attached to the Cisco CMTS; the update takes longer if more number of modems are attached to the Cisco CMTS.

Tip You specify an upper and lower threshold for carrier-to-noise ratios in the CNRCeiling and CNRFloor parameters of the SPECTRUM.INI file. If you want to specify different thresholds on an analysis-by-analysis basis, you can do that in the Upper Threshold and Lower Threshold fields. For more information on thresholds, see the "Parameters in the SPECTRUM.INI File" section on page 3-22.

Step 3 Click **Start**. The CNR Analysis dialog box appears, as shown in Figure 4-7.

Step 4 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-7 CNR Analysis Criteria Dialog Box



Retrieving CNR Trending Data

To retrieve the CNR trending data for selected cable modems or upstream ports:

Step 1 From the Spectrum Tools menu, choose **CNR Trending**. The CNR Trending Criteria dialog box appears as shown in Figure 4-8.

Step 2 Select one or more upstreams or cable modems for which you want to see the CNR trending data.

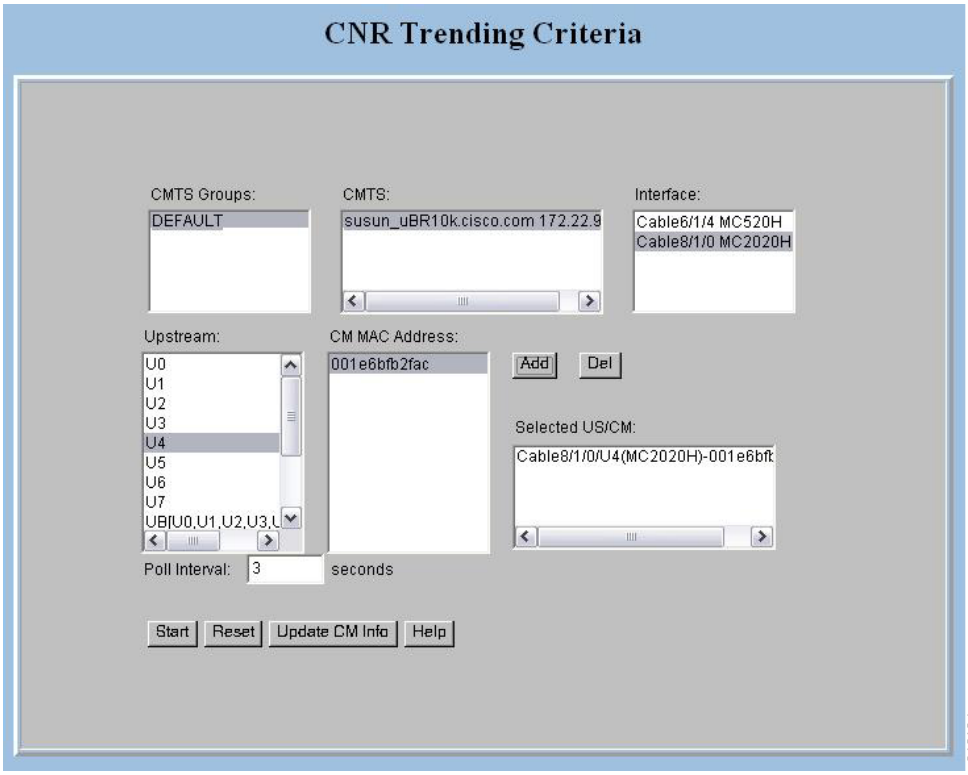
Note To update the cable modem information on the selected CMTS instead of getting data from the cache, click **Update CM Info**. This real-time update varies according to how many modems are attached to the CMTS. The update depends on the number of modems attached to the Cisco CMTS; the update takes longer if more number of modems are attached to the Cisco CMTS.

Step 3 Click **Add** to add the upstream or cable modem to the Selected US/CM list.

Step 4 Click **Start**. The CNR Trending graph is displayed.

Step 5 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-8 CNR Trending Criteria Dialog Box



Playing Back Data

You can play back a saved Trace Window or CNR Analysis. This allows you to capture and preserve troubleshooting information and use it later.

Playing Back Trace Window

To play back Trace window:

Step 1 From the Spectrum Tools menu, choose **Data Playback > Trace Window**. The Trace Window Playback Criteria dialog box appears, as shown in Figure 4-9. Otherwise, Figure 4-10 shows an alternate view of this dialog box, with additional information displayed for illustrative purpose.

Step 2 Select one or more MAC addresses for which you want to see a list of saved traces and click **Start**. The Trace Window Playback List dialog box appears, as shown in Figure 4-11.

Step 3 From the list of saved traces, select the one you want to play back and click **Show**. The Trace Window for the saved trace, as shown in Figure 4-1, appears.

Note To remove a saved Trace Window from the database, from the Utilities menu, choose **Purge Saved Data, Spectrum Data**.

Step 4 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-9 Trace Window Playback Criteria Dialog Box

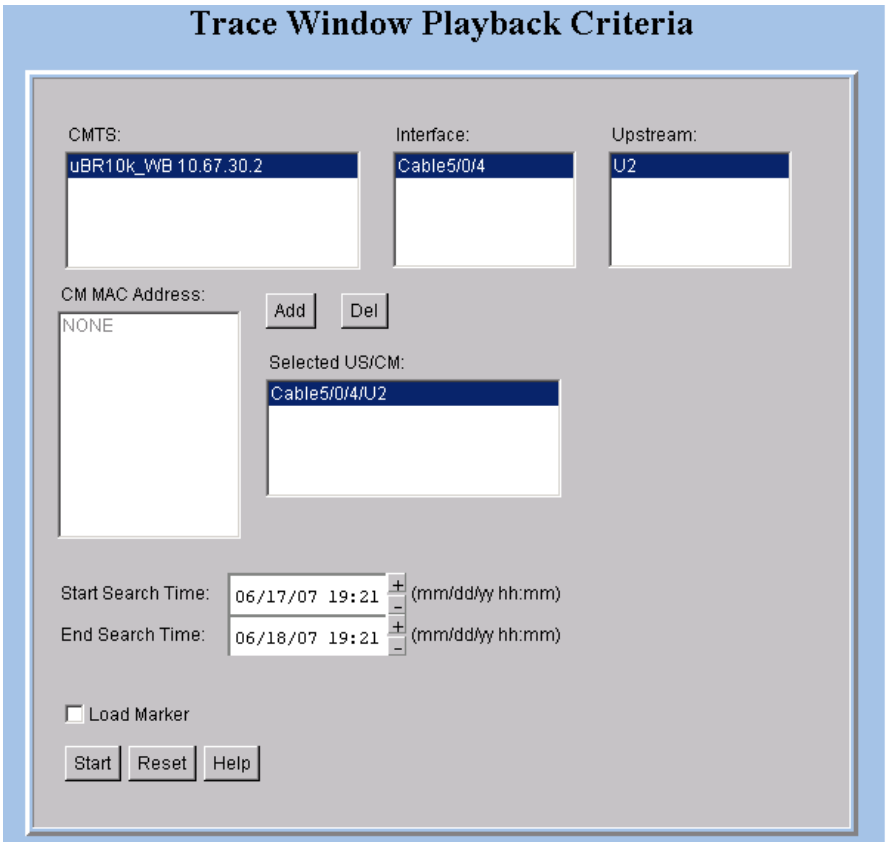


Figure 4-10 Trace Window Playback Criteria Dialog Box, Alternate View

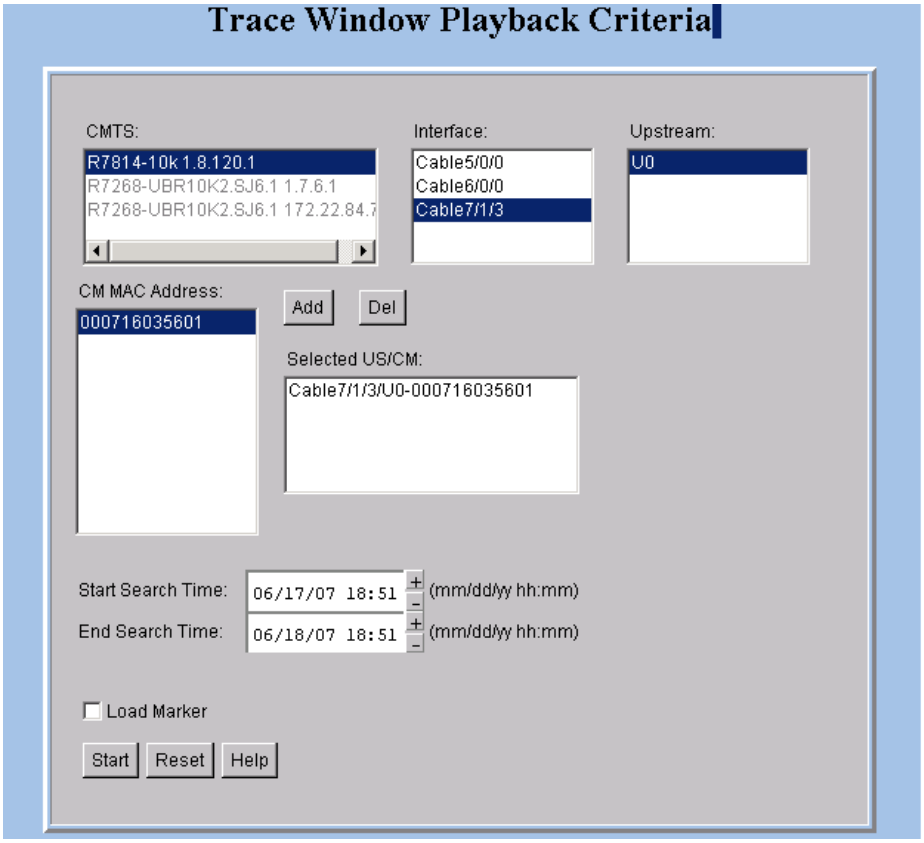
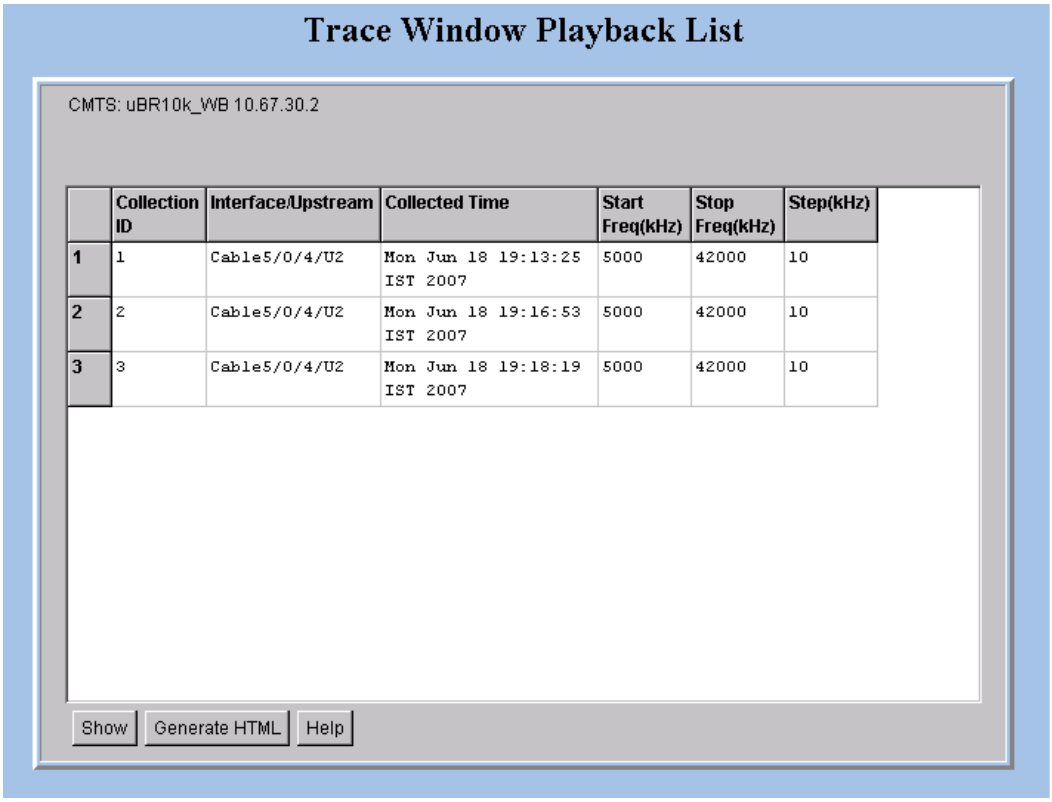


Figure 4-11 Trace Window Playback List Dialog Box



Playing Back CNR Analysis

To play back CNR analysis:

Step 1 From the Spectrum Tools menu, choose **Data Playback > CNR Analysis**. The CNR Analysis Playback Criteria dialog box appears, as shown in Figure 4-12.

Step 2 To select one or more MAC addresses for which you want to see a saved CNR Analysis, click **Start**. The CNR Analysis Playback Result dialog box appears, as shown in Figure 4-13.

Note To remove a saved CNR Analysis from the database, from the Utilities menu, choose **Purge Saved Data, Spectrum Data**.

Step 3 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-12 CNR Analysis Playback Criteria Dialog Box

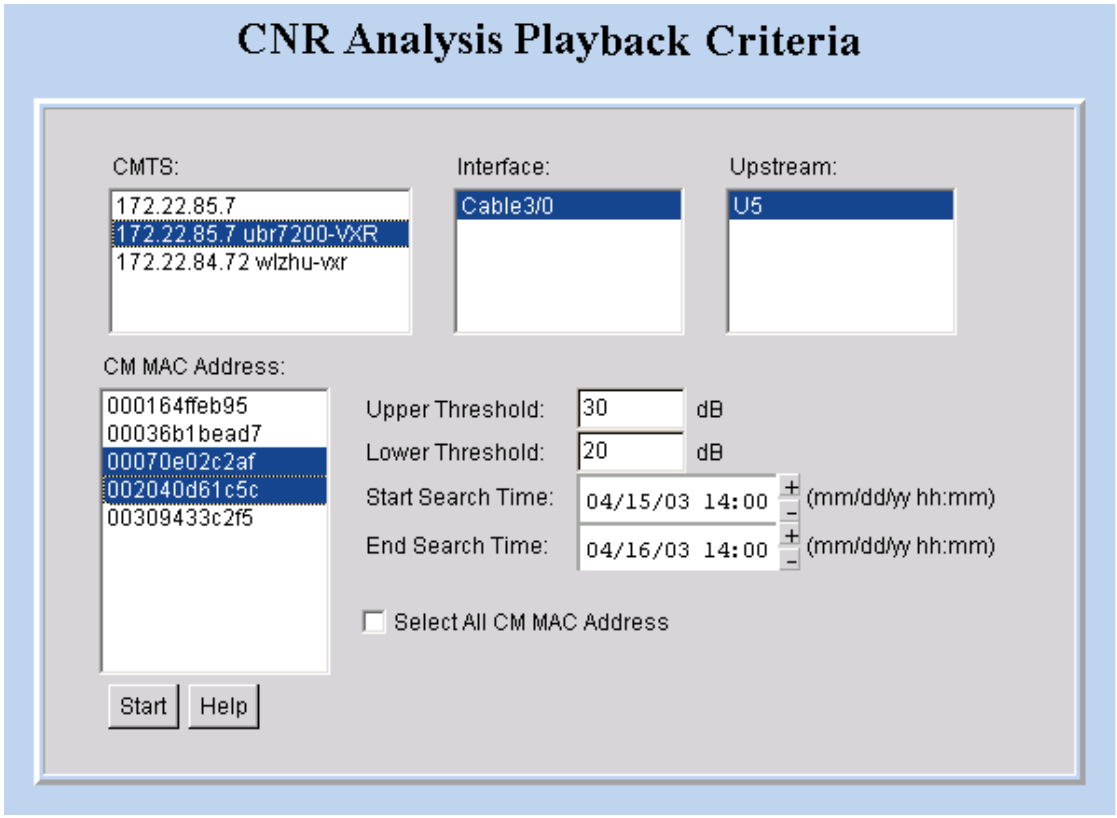
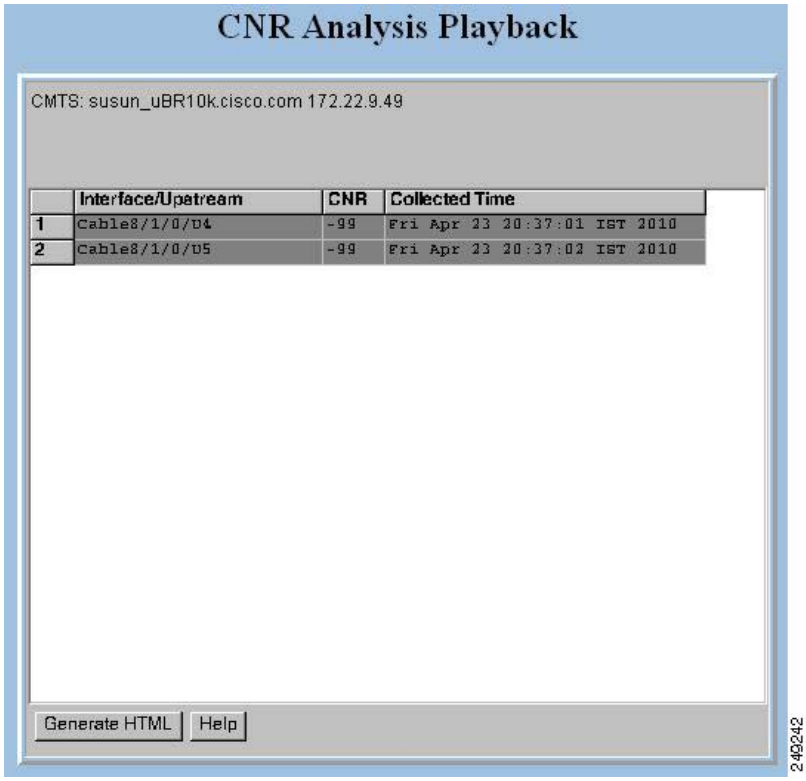


Figure 4-13 CNR Analysis Playback Result Dialog Box



Playing Back CNR Trending Data

To play back CNR trending data:

Step 1 From the Spectrum Tools menu, choose **Data Playback > CNR Trending**. The CNR Trending Playback Criteria dialog box appears, as shown in Figure 4-14.

Step 2 Select the MAC addresses for which you want to see the saved CNR trending data and then click **Add**.

Step 3 Click **Start** to view the CNR Trending Playback result dialog box. (See Figure 4-15.)

Step 4 Click **Print** to create a printer-friendly version of the chart and print it at your local printer.

Step 5 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-14 CNR Trending Playback Criteria Dialog Box

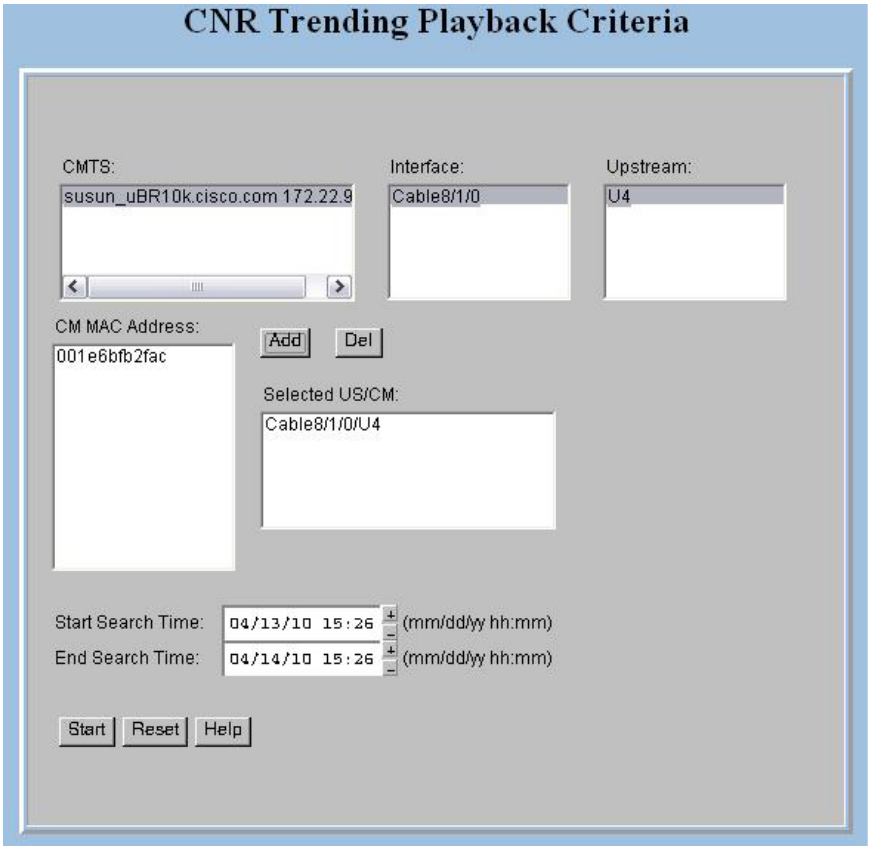
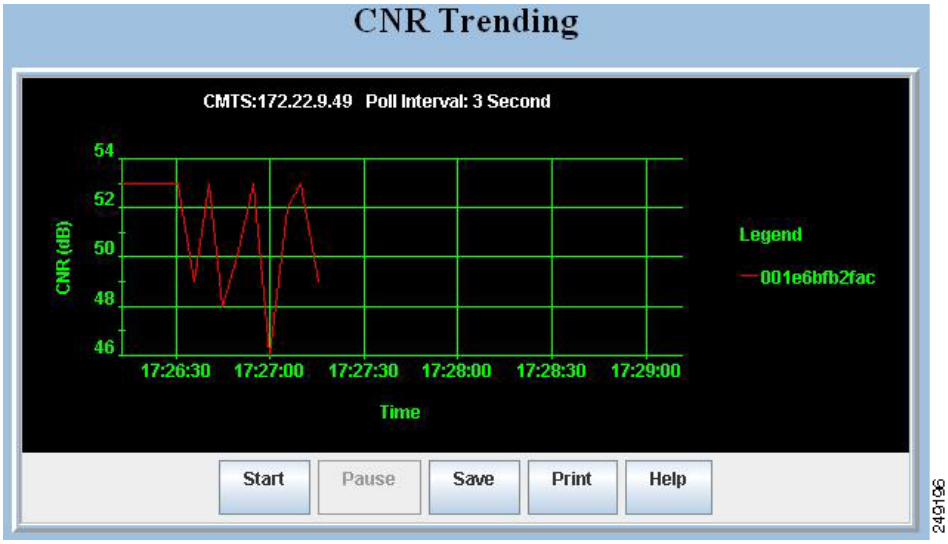


Figure 4-15 CNR Trending Playback Result Dialog Box



Displaying Polling Status

To get a list of each spectrum analysis in process on a client machine and to view the details of the scheduled spectrum polling events, use the Polling Status tool.

Real Time Polling Status

To see a list of clients currently using CBT's spectrum management tools:

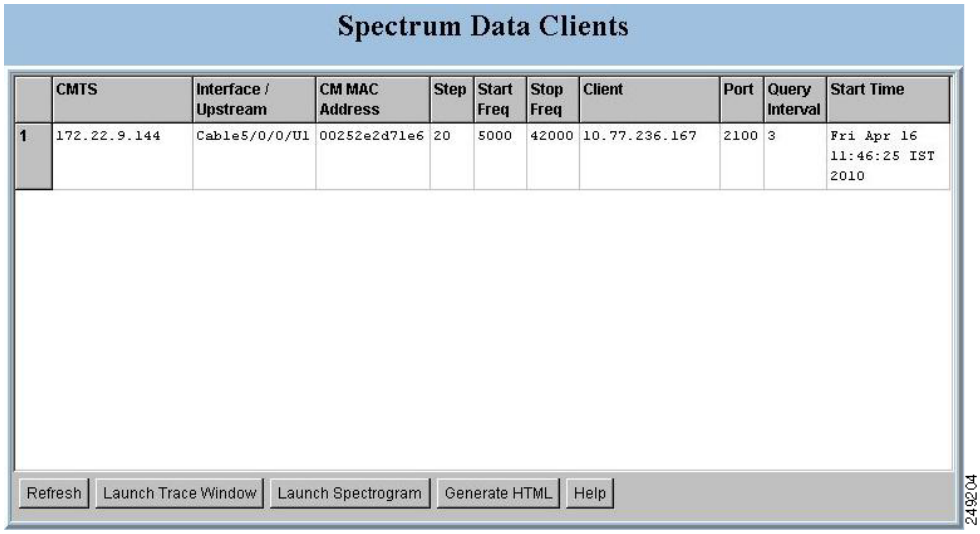
Step 1 From the Spectrum Tools menu, choose **Polling Status > Real Time**. The Spectrum Data Clients dialog box appears, as shown in Figure 4-16.

Step 2 Perform one of the following:

1. **Refresh** - Refreshes the screen.
2. **Launch Trace Window** - Launches the trace window.
3. **Launch Spectrogram** - Launches the spectrogram.
4. **Generate HTML** - Generates the HTML file that can be printed and saved.

Step 3 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-16 Spectrum Data Clients Dialog Box



Scheduled Polling Status

To view the details of the scheduled spectrum polling events:

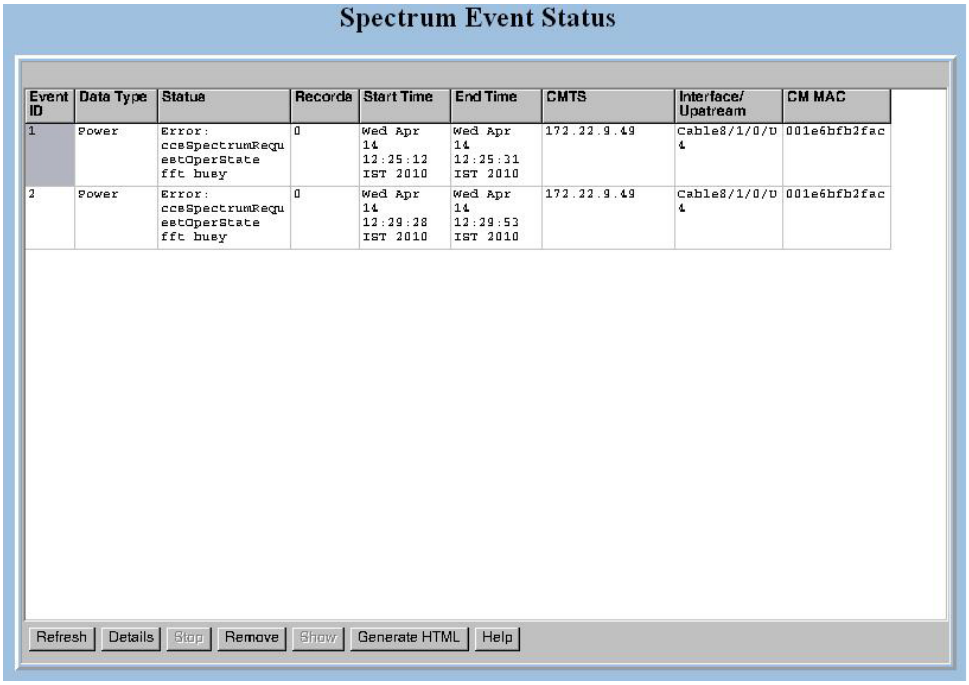
Step 1 From the Spectrum Tools menu, choose **Polling Status > Scheduled**. The Spectrum Event Status dialog box appears, as shown in Figure 4-17.

Step 2 Perform one of the following:

1. **Refresh** - Polls the database for the latest events.
2. **Details** - Displays a pop-up window with the details for the scheduled collection.
3. **Stop** - Cancels the scheduled collection.
4. **Remove** - Cancels the scheduled collection.
5. **Show -** Displays the data collection.
6. **Play** - Shows the actual trace.
7. **Generate HTML** - Generates the HTML file that can be printed and saved.

Step 3 For a complete description of each field in these dialog boxes, click **Help**.

Figure 4-17 Spectrum Event Status Dialog Box



Enabling Instantaneous CPU Assessment for Spectrum Polling

CBT 3.4 enables the option of preventing excessive CPU consumption, in circumstances in which the Trace Window, Spectrogram functions, or Scheduled Polling features might otherwise exceed CPU bandwidth.

CBT has provisions to instantaneously check the CPU utilization and prevents the spectrum operation if the CPU utilization exceeds the CPU thresholds (value of SpecDataMaxCPU parameter).

Perform these steps to enable or disable CPU assessment in the Trace Window Criteria dialog box, the Spectrogram Criteria dialog box, or the Spectrum Data Scheduler dialog box before launching the spectrum event.

Step 1 Launch CBT 3.4.

Step 2 Log in to CBT as **admin**.

Step 3 Launch the desired window in which to configure this change.

- For Trace Window changes, from the Spectrum Tools menu, choose **Trace Window**. The Trace Window Criteria dialog box appears. Proceed to the next step.
- For Spectrogram changes, from the Spectrum Tools menu, choose **Spectrogram**. The Spectrogram Criteria dialog box appears. Proceed to the next step.

Step 4 Select the checkbox **`Check CPU Util`**. This setting defines the number of sweeps for every CPU utilization check. Only when the check box is selected is the text box enabled to enter the number of spectrum sweeps that should be done for every CPU utilization check. If periodic check of CPU utilization is not necessary, then deselect this option and proceed. If not selected, the CPU utilization is checked only once before the launch of the spectrum operation. Refer to Figure 4-18.

Figure 4-18 Check CPU Util Dialog Box

Step 5 When the **Check CPU Util** check box is selected, enter the number of sweeps in which CPU utilization should be checked. The default value is 10.

This setting means that the CPU utilization is checked and compared with the SpecDataMaxCPU parameter. Upon positive results, the spectrum operation gets started. The spectrum data is collected continuously with the specified poll interval.

One sweep corresponds to a single polling of spectrum data. After 10 such polling events, CBT again checks the CPU Utilization and compares it with the SpecDataMaxCPU parameter. If the results are positive, the spectrum operation continues; otherwise, the continuous query is stopped. If the number of sweeps is 1, then the CPU utilization is checked before every polling of spectrum data.

Step 6 Continue with the launch of the spectrum operation.

CHAPTER 5

Subscriber Traffic Management in CBT 3.4

This section describes the subscriber traffic management (STM) tools that are used for viewing the enforce rules and locating violators:

- **Enforce Rules**—Displays the STM enforce rules on a Cisco CMTS.
- **All Violators**—Displays the STM violator modems connected to a Cisco CMTS.
- **All Violators Real Time**—Displays the STM violator modems connected to a Cisco CMTS.
- **Locate Violator**—Finds violator modems connected to a Cisco CMTS.

Using the STM Tools

This section describes how to use the STM tools available in CBT 3.4:

- Displaying STM Enforce Rules Defined for a Specific Period, page 5-1
- Displaying Active Enforce Rules, page 5-2
- Displaying Violators Detected for a Specific Period, page 5-2
- Displaying Current Violators, page 5-2
- Locating Violators, page 5-3

Displaying STM Enforce Rules Defined for a Specific Period

To display enforce rules defined for a specific period:

Step 1 From the STM Tools menu, choose **Enforce Rules**. The Get All Enforce Rules dialog box appears.

Step 2 Choose the CMTS group that includes the CMTS where you want to view the enforce rules.

Step 3 Choose the CMTS in the **CMTS** field for which you want to view the enforce rules.

Step 4 Set Start Time and End Time for the enforce rules that you want to view.

Step 5 Click **Get Rules** to display the enforce rules for the specified time period.

Step 6 For a complete description of each field in the Get All Enforce Rules dialog box, click **Help**.

Displaying Active Enforce Rules

To display active enforce rules:

Step 1 From the STM Tools menu, choose **Enforce Rules**. The Get All Enforce Rules dialog box appears.

Step 2 Choose the CMTS group that includes the CMTS where you want to view the enforce rules.

Step 3 Choose the CMTS in the **CMTS** field for which you want to view the enforce rules.

Step 4 Select **Real Time** for the enforce rules you want to view.

Step 5 Click **Get Rules** to display the active enforce rules.

Step 6 For a complete description of each field in the Get All Enforce Rules dialog box, click **Help**.

Displaying Violators Detected for a Specific Period

To view the STM violator modems connected to a Cisco CMTS, use the **All Violators** tool.

To display violators detected for a specific period:

Step 1 From the STM Tools menu, choose **All Violators**. The Get All Violator Modems dialog box appears.

Step 2 Choose the CMTS group that includes the CMTS where you want to view the enforce rules.

Step 3 Choose the CMTS in the **CMTS** field for which you want to view the enforce rules.

Step 4 Set Start Time and End Time for the enforce rules that you want to view.

Step 5 Click **Get Violators** to display the violators for the specified time period.

For a complete description of each field in the Get All Violator Modems dialog box, click **Help**.

Displaying Current Violators

To view the STM violator modems connected to a Cisco CMTS, use the **All Violators Real Time**.

To display current violators:

Step 1 From the STM Tools menu, choose **All Violators Real Time**. The Get All Violator Modems Real Time dialog box appears.

Step 2 Choose the CMTS group that includes the CMTS where you want to view the enforce rules.

Step 3 Choose the CMTS in the **CMTS** field for which you want to view the enforce rules.

Step 4 Click **Get Violators** to display the current violators.

Step 5 Click **Get Enforce Rules** to display the current enforce rules for the specified time period. A new screen appears showing the rules enforced during the specified period.

Step 6 Click **Display Violators** to return to the violators screen.

Step 7 For a complete description of each field in the Get All Violator Modems Real Time dialog box, click **Help**.

Locating Violators

To find violator modems connected to a Cisco CMTS, use the **Locate Violator** tool.

To locate violators:

Step 1 From the STM Tools menu, choose **Locate Violators**. The Locate A Violator Modem dialog box appears.

Step 2 Enter the MAC address of the modem you want to locate.

Step 3 Set the Start Time and End Time for the period you want to view the violator.

Step 4 Click **Locate Modem** to display the violators for the specified time period.

Step 5 For a complete description of each field in the Locate A Violator Modem dialog box, click **Help**.

Note To remove saved STM data from the database, choose **Purge Saved Data > STM Data** in the **Utilities** menu.

CHAPTER 6

Administrative Tasks for CBT 3.4

This section describes administrative configuration tasks that improve additional functions of CBT 3.4:

- Mapping Hostnames from Applet to Servlet, page 6-1
- Configuring Administrative Parameters for Spectrum Analysis, page 6-2

Mapping Hostnames from Applet to Servlet

CBT 3.4 supports hostname-based communications between applet and servlet on the network, in addition to sustaining IP-based communications.

In CBT 3.2 all the communication between the applet and the servlet was based only on the IP address of the server. Every request and response between the applet and servlet had the IP address in the URL. CBT 3.3 and later remove this limitation, to support both hostnames and IP addresses.

This procedure changes the UseHostName parameter of the GUNSLINGER.INI file. By default, this value is NO in CBT 3.3 and later. Changing this value to YES enables hostname-based communication between applet and servlet, and removes IP addresses from corresponding displays and processing.

Some Web proxy functions only allow hostname-based IP communication, and otherwise drop communications with IP addresses in the URL. This feature is helpful in such cases.

This procedure requires that applet and servlet hostnames be established and otherwise functional on the network prior to mapping hostnames in CBT. Both the client and server should be able to resolve the hostname of the CBT server machine.

Perform these steps to map hostnames from applet to servlet.

Step 1 Open the GUNSLINGER.INI file available in the CBT server machine.

- For Solaris and Linux, the GUNSLINGER.INI file is under the directory /opt/CSCOcbt/httpServer/webapps/ROOT or <CBT_Install_Directory>/httpServer/webapps/ROOT.
- For Windows, the GUNSLINGER.INI file is under the directory C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\ or <CBT_Install_Directory>\httpServer\webapps\ROOT\.

Step 2 Change the UseHostName parameter to YES or NO depending on your need.

- YES enables hostname-based communications.
- NO implements IP-address-based communications.

Step 3 Save the GUNSLINGER.INI file with your changes.

Step 4 Restart CBT 3.4.

Configuring Administrative Parameters for Spectrum Analysis

CBT releases prior to CBT 3.4 support options for spectrum operation parameters, and CBT 3.4 continues support for these parameters. CBT 3.4 enables the adminstrators to configure upper and lower limits for multiple fields in the Spectrum Operations windows.

To define administrative upper and lower limits in Spectrum Operations windows:

Step 1 Open the SPECTRUM.INI file available in the CBT server machine.

- For Solaris and Linux, the SPECTRUM.INI file is under the directory /opt/CSCOcbt/httpServer/webapps/ROOT or <CBT_Install_Directory>/httpServer/webapps/ROOT.
- For Windows, the SPECTRUM.INI file is under the directory C:\Program Files\Cisco Systems\Cisco Broadband Troubleshooter\httpServer\webapps\ROOT\ or <CBT_Install_Directory>\httpServer\webapps\ROOT\.

Step 2 Change the default settings for as few or as many parameters as desired. Listed below are default values, and acceptable ranges for these parameters.

- SpecEventStartFreqMinValue=5
- SpecEventStartFreqMaxValue=65
- SpecEventStopFreqMinValue=5
- SpecEventStopFreqMaxValue=65
- SpecEventCenterFreqMinValue
- SpecEventCenterFreqMaxValue
- SpecEventFreqSpanMinValue
- SpecEventFreqSpanMaxValue
- SpecEventDataPointsMinValue
- SpecEventDataPointsMaxValue
- SpecEventPollIntervalMinValue
- SpecEventPollIntervalMaxValue
- SpecEventStepSizeMinValue
- SpecEventStepSizeMaxValue
- SpecEventSamplesMinValue
- SpecEventSamplesMaxValue

You can configure the upper and lower limit for the start, stop frequency, poll interval, step size and data points. While entering such data spectrum parameters, CBT 3.4 sustains native-level validation, and additionally, CBT 3.4 validates entries with respect to the newly configured limits in the SPECTRUM.INI file. The following list describes various fields and supported ranges for validation in a Spectrum Event:

- Start Frequency—Should be in the range of 5 to 65, with the value representing MHz. A decimal value is supported.
- Stop Frequency—Should be in the range of 5 to 65, with the value representing MHz. A decimal value is supported.
- Center Frequency and Frequency Span—The frequency span cannot be greater than these results:
  - minimum—(center frequency - 5) x 2
  - minimum—(65 - center frequency) x 2

  Validation is performed on Center Frequency and Frequency Span in the following way:
  - Start and stop Frequencies are calculated.
  - startFreq = centerFreq - (freqSpan/2)
  - stopFreq = centerFreq + (freqSpan/2)

  As a result, these values should be in the range 5 to 65. In this way both the fields are validated.
- Data Points—Should be an integer greater than 2, and are subject to the following additional range requirements, according to the cable interface line cards:
  - Should be in the range from 2 to 6001 for the MC5X20S/U/H BPE.
  - Different upper limits of Data Points for different cable interface line cards are as follows:
    - MC16S: 617
    - MC16U: 3001
    - MC16X: 3001
    - MC28U: 3001
    - MC28X: 3001
    - MC520S: 5886
    - MC520U: 6001
    - MC20X20V: 3001
    - MC88V: 3001
  - Data Points with other cable interface line card combinations should result in a start and stop frequency range of 5 to 65.
- Poll Interval—Must be a positive integer.
- Step Size—Should be a positive integer not greater than 6000.

Note For any combination of the above-mentioned field values that you enter, the start and stop frequency should always be in the range 5 to 65.

Step 3 CBT 3.4 checks the administratively configured values in the SPECTRUM.INI file. When changing this file, all the field values should also be in the range specified, and saved to the SPECTRUM.INI file for non-default configurations.

Example

Only when the value of the SpecEventPollIntervalMinValue parameter is set to 1, values above 1 are accepted in the Poll Interval dialog box. This example removes the bandwidth-consuming zero-second polling.

CHAPTER 7

Troubleshooting CBT 3.4

This section contains the following procedures for verifying and troubleshooting CBT 3.4:

- Troubleshooting Continuous Sweep Spectrum Operation in CBT 3.4, page 7-1
- Changing Server Ports in XML Script, page 7-2
- Saving System Message Logs for Troubleshooting, page 7-2
- Verifying Installation Status on the CBT 3.4 Server, page 7-3
- Verifying the CBT 3.4 License, page 7-4
- Troubleshooting the Poller, page 7-4
- Troubleshooting Poller Problems, page 7-4

Troubleshooting Continuous Sweep Spectrum Operation in CBT 3.4

A non-standard behavior has been observed in which the Continuous Spectrum Operation of CBT 3.4 fails, even in circumstances in which the Single Sweep Spectrum Operation remains functioning. One example would be the proper Trace Window single sweep operation, but with failed Continuous Sweep behavior in the Trace Window.

CBT 3.4 requires that the following processes and tasks be used for Continuous Sweep Operation.

Step 1 Verify that the TCP ports used by CBT 3.4 are as follows:

- Port 8105: JVM server port
- Port 9080: Non-SSL HTTP port
- Port 9443: SSL HTTP port
- Port 9082: AJP 1.3 Connector
- Port 2640: For Sybase database connectivity
- Port 8020: For Poller operation

Step 2 Implement the following TCP port ranges for spectrum operations. Spectrum events such as Trace Window, Spectrogram, CNR Trending, and the Generic Query use a specified range of port numbers for operation.

- For both Trace Window events and the Spectrogram events, the port numbers range from 2100 to 2500.
- For the Spectrum tools and CNR Trending, port numbers range from 3100 to 3500.
- For the Diagnostics tools and Generic Query, the port numbers range from 5100 to 5500.
- The usage of these port numbers should in increments of 1. That is, for the first Trace Window launched, the port number is 2100, and the next Trace Window port 2101. For the next Spectrogram window, the port number is 2102 and so forth up to port 2500.

Changing Server Ports in XML Script

The Tomcat server port in server XML script is set to port 8105. This prevents possible port conflict when multiple Tomcat Web servers are running on the same workstation.

If required, to change the Tomcat server port in XML script to port 8105 and check for additional port conflicts:

Step 1 Edit the server.xml script in the following location:

/opt/CSCOcbt/httpServer/conf

Step 2 Check for all ports numbers that have conflicts with CBT and change all port numbers that have such conflicts:

- Port 8105: JVM server port
- Port 9080: Non-SSL HTTP port
- Port 9443: SSL HTTP port
- Port 9082: AJP 1.3 Connector

Saving System Message Logs for Troubleshooting

When troubleshooting CBT 3.4, we recommend that message logs be saved and filtered using the following steps.

Step 1 To view saved message logs, click **Utilities > Message log** in the user interface.

This is a fixed-size log file that continuously removes the oldest entry as it is updated. This log file is not viewable using an editor (such as vi).

Step 2 Filter the message log entries by severity, by user, by module, or by date.

Step 3 To set the logging level, click **Configuration > Message Log** in the user interface. Setting the logging level helps to prevent unnecessary messages occupying the fixed-size log file.

For Solaris and Linux, the log file is located at /opt/CSCOcbt/httpServer/logs/catalina.out. This file is viewable with viewing utilities such as vi, Cat, Tail, or others.

Note This message log file grows in size. For Windows, these messages go to the console and a message log file is not created.

Verifying Installation Status on the CBT 3.4 Server

To verify the status of the installed CBT 3.4 server:

Step 1 Verify that there are no exceptions in the catalina.out file located as follows:

- Solaris and Linux—/opt/CSCOcbt/httpServer/logs/
- Windows—The CBT 3.4 console

Step 2 Verify that the dbeng8 process is running on the Solaris and Windows systems and the dbeng10 process is running on the Linux system:

- Solaris:

```
ps -ef|grep dbe
root 26449 1 0 Aug 18 ? 0:24 dbeng8 -x tcpip{ServerPort=2640} -q -ud -s local0 -m -c
16M -n cbtdbengine /opt
```

- Linux:

```
ps -ef|grep dbe
root 26449 1 0 Aug 18 ? 0:24 dbeng10 -x tcpip{ServerPort=2640} -q -ud -s local0 -m -c
16M -n cbtdbengine /opt
```

- Windows: There is a Sybase icon shown as a running process.

Step 3 Verify that the following Java processes are running:

- Solaris and Linux:

```
ps -ef|grep java
root 26478  1 0 Aug 18 ?  0:39 /opt/CSCOcbt/jre/bin/java -DCBTpoller -cp
/opt/CSCOcbt/httpServer/webapps/ROOT/

root 26489 1 0 Aug 18 ?  148:55 /opt/CSCOcbt/jre/bin/java
-Djava.endorsed.dirs=/var/CSCOcbt/httpServer/common/e
```

Note Linux displays threads; therefore, there are many Java entries.

- Windows:

  Use the command console for Tomcat and the CBT poller to view log messages.

Verifying the CBT 3.4 License

To verify the CBT 3.4 installation license:

Step 1 For Solaris and Linux, perform these steps:

1. Verify that the license is present and the file is correct in the following location:

   /opt/CSCOcbt/httpServer/webapps/ROOT/WEB-INF/classes/lic/License
2. Log in and allow for error prompts to identify any issues.
3. Scan the catalina.out file for error messages in the following location:

   /opt/CSCOcbt/httpServer/logs/

Step 2 For Windows, perform these steps:

1. Verify that the license is present and the file is correct in the following location:

   INSTALLATION_DIR\httpServer\webapps\ROOT\WEB-INF\classes\lic\License
2. Log in and allow for error prompts to identify any issues.
3. Scan the command console for Tomcat and see if there are error messages in the console.

Troubleshooting the Poller

This section provides directions to troubleshoot error messages for the following CBT components:

- **Script**—Retrieves subscriber or provisioning information from an external source.
- **Application on an HTTP server**—Retrieves subscriber or provisioning information from an external source.
- **Poller**—Queries CMTSs by using SNMP and stores provisioning information in the CBT local Sybase database.

Troubleshooting Poller Problems

You can use the Poller, a separate Java application, to store provisioning information in the CBT local Sybase database. This database Poller queries the CMTSs by using SNMP. If the Debug parameter in POLLER.INI is set to true, the Poller sends information to the Poller.log file in the following location:

- **Linux and Solaris**—/opt/CSCOcbt/logs
- **Windows**—%CSCOCBT_ROOT%\logs

Note See the "Parameters in the POLLER.INI File" section on page 3-21 for an explanation of each parameter in POLLER.INI.

This section:

- Shows output from Poller.log that indicates there is a problem
- Describes the cause of the problem

- Provides a resolution for the problem

Error Message: No Router

Problem

The following output from Poller.log indicates that no router was found:

```
Thu Jun 27 15:47:35 EDT 2002: INFO: Start polling...
Thu Jun 27 15:47:37 EDT 2002: INFO: No router found in routers list file:
/opt/CSCOcbt/jakarta-tomcat-4.0.3/webapps/ROOT/WEB-INF/classes/config/myrouters
Thu Jun 27 15:47:37 EDT 2002: INFO: Finished polling... elapsed time: 2371
milliseconds
Thu Jun 27 15:47:37 EDT 2002: INFO: Poller exited.
```

Cause

No router was found in the routers list file, myrouters, because of one of the following conditions:

- The routers list file does not exist. This is the case when you first install CBT.
- The CMTS routers list is empty; it does not contain any routers.

Resolution

To add routers in CBT, see the "Adding Router Information" section on page 3-14.

Error Message: Database Login Failed

Problem

The following output from Poller.log indicates that the login to the database failed:

```
Wed Jun 26 11:14:12 PDT 2002: INFO: Start polling...
Wed Jun 26 11:14:13 PDT 2002: FATAL: SQL Exception in setupDB makeDBConnection: 1
JZ00L: Login failed. Examine the SQLWarnings chained to this exception for the
reason(s).
Software aborted.
```

Cause

The database login failure occurs because the database ran out of connections; that is, it tried to use more connections than it can support.

Resolution

If the database ran out of connections, restart CBT to clear all existing database connections and start the database.

Error Message: SNMP Timeout

Problem

The following output from Poller.log indicates that an SNMP timeout has occurred:

```
Thu Jun 20 13:45:08 EDT 2002: INFO: Start polling...
Thu Jun 20 13:45:18 EDT 2002: ERROR: Thread16 got SNMP exception on Router:
24.216.122.254. Exception: Snmp Timeout... Router 24.216.122.254 is unreachable
```

Cause

The SNMP timeout occurs when the network connection to the router, or the router itself, is so busy that it cannot satisfy the request within the specified time.

Resolution

If the SNMP timeout occurs too frequently, change the SNMP settings in the POLLER.INI file:

Step 1 Open the POLLER.INI file for your operating system:

- **Linux and Solaris**—/opt/CSCOcbt/bin
- **Windows**—%CSCOCBT_ROOT%\bin

Step 2 Increase one or both of the following settings:

- **SnmpTimeout**—You can increase the number of seconds before the attempt to create the SNMP connection reaches a timeout.
- **SnmpRetry**—You can increase the number of times to try to create the SNMP connection.

Step 3 To activate the new settings, restart the CBT server.

Tip Check for incorrect SNMP community strings, because an invalid community string for read/write also results in an SNMP timeout.

CHAPTER 8

Sample Code for Application Program Interfaces

This section contains the README files for the following application program interfaces (APIs):

- Shell script for retrieving subscriber or provisioning information from an external source by using a script
- Java code for retrieving subscriber or provisioning information from an external source by using an HTTP application

The sections that describe scripts and code provide the following information:

- Location
- Purpose
- Requirements
- Description of components
- Directions for modifying the code
- Directions for troubleshooting the code
- Sample code that you can modify to meet your needs

This section contains the following topics:

- Shell Script for Retrieving Subscriber or Provisioning Information, page 8-1
- Java Code for Retrieving Subscriber or Provisioning Information, page 8-4

Shell Script for Retrieving Subscriber or Provisioning Information

You can use a sample script written for various databases to retrieve subscriber or provisioning information from an external source. These scripts, that you can modify according to your needs, are in the README file in the following locations:

- **Linux and Solaris**
  - /opt/CSCOcbt/samples/db_script/oracle/
  - /opt/CSCOcbt/samples/db_script/sybase/
- **Windows**
  - %CSCOCBT_ROOT%\samples\db_script\oracle\

- %CSCOCBT_ROOT%\samples\db_script\sybase\

This section provides a printed copy of that README file, formatted for this user guide.

Purpose of Shell Script

This section describes how to use scripts written against a Sybase database. The sybase directory contains sample scripts that you can modify to meet your needs. These scripts demonstrate how to:

- Receive requests from CBT
- Query data against a Sybase database
- Return data to CBT in the format specified in "Retrieving Data with a Script" section on page 3-25

Sample Script Requirements

The sample scripts have the following criteria for use:

- Sample scripts require Sybase running with CBT's database schema, which is created automatically during the CBT install.
- Sample scripts are written for the Solaris OS. To use the scripts on a Linux machine, see the "Troubleshooting the Script" section on page 8-3.

After trying the sample scripts, you can tailor them for your database or any other external data source.

Script and File Descriptions

The following list of sample scripts describes the purpose of each script or file:

- **get_subscriber**—Queries subscriber information from the database
- **get_provision**—Queries provisioning information from the database
- **insert_cmts**—Invokes insert_cmts.sql
- **insert_cmts.sql**—Inserts CMTSs and cable modems into the database and is invoked by insert_cmts
- **subscriberinfo.txt**—Contains subscriber information in ASCII text file format
- **add_col_name.pl**—Adds column names to the subscriber fields (for example, Customer Last Name) queried from get_subscriber

Modifying the Script

To modify a script:

Step 1 Modify insert_cmts.sql to insert a list of CMTSs and cable modems into the database.

Step 2 Execute insert_cmts from the command line: `./insert_cmts`

Step 3 Modify subscriberinfo.txt to import subscriber information into CBT. For field descriptions, refer to online help for the Import Subscriber Data dialog box.

Step 4 Start CBT. Log in as an Admin user.

Step 5 Click **Import Subscriber Data**, enter the Subscriber Data File Location, and click **Start**.

Step 6 Click **Set External Interfaces**.

Step 7 Under Subscriber Information, select the **Script** radio button and enter the location of the get_subscriber filename (for example, /my_dir/get_subscriber) in the Script Location field.

Step 8 Under Provisioning Information, select the **Script** radio button and enter the location and the get_provision filename (for example, /my_dir/get_provision) in the Script Location field.

Troubleshooting the Script

- The get_subscriber and get_provision scripts can be run from the UNIX command line. The following examples show how to execute a script and the output the script produces:

  ./get_provision GET_PROVISION MAC 000216d5a0cf

  OUT_DATA=PROVISION^172.22.85.10^172.22.127.26^

  ./get_provision GET_MAC IP 172.22.127.26

  OUT_DATA=MAC^000216d5a0cf^

  ./get_subscriber GET_SUBSCRIBER MAC 000216d5a0cf

  OUT_DATA=SUBSCRIBER^AccId=ID000006^Name=Name000006^Phone=6172300006^Address=175 West Tasman^ClassOfService=Policy000006^FiberNode=User A000006^

  ./get_subscriber GET_MAC PHONE 5106663152

  OUT_DATA=MAC^000164ffeb95^000164ffc3c7^

  ./get_subscriber GET_ADDRESS MAC 000164ffc3c7^000164ffeb95^

  OUT_DATA=ADDRESS^000164ffc3c7=170 West Tasman Drive,95134^000164ffeb95=170 West Tasman Drive,95134^

- The get_subscriber script invokes add_col_name.pl. Modify the scriptDir to accurately reflect where add_col_name.pl is located.
- All scripts use /tmp as a temporary directory for creating temp files. Create the /tmp directory with read and write privileges for all users.
- If you are running the scripts on a Linux machine, make the following changes to the get_provision, get_subscriber, and insert_cmts scripts:
  - Modify commLib to /bin
  - Replace dbisql to dbisqlc
- If you get Invalid Data when using echo in Window's batch file:
  - Turn off echo:
    ```
    @echo off
    ```
  - If you are not using double quotes around the message, then the delimiter has to be "^^" instead of "^".

    For example:

    echo OUT_DATA=PROVISION^^172.22.85.10^^172.22.127.26^^

Sample Script Code

The following sample code is extracted from the get_subscriber script:

```
#!/bin/ksh
argc=$#
set -A argv $*
integer i=0
while let "i < $argc"; do
  case ${argv[$i]} in
    "GET_MAC") function=${argv[$i]};;
  esac
  let "i = i + 1"
done

#retrieving MAC based on Customer phone number
if [ "$function" = "GET_MAC" ] ; then

   str="SELECT MACAddress FROM SUBSCRIBERINFO where CusPhone='$in_param';";
   echo $str >> "$tmpDir/script.$filenameExt"
   str="OUTPUT TO $tmpDir/test.$filenameExt FORMAT ASCII;"
   echo $str >> "$tmpDir/script.$filenameExt"

   tmpi=`dbisqlc -q -c $dbAccess read $tmpDir/script.$filenameExt`
   rm $tmpDir/script.$filenameExt

   MAC=`$commLib/cat $tmpDir/test.$filenameExt | $commLib/sed s/\'//g`
   echo $MAC > $tmpDir/test.$filenameExt
   MAC=`$commLib/cat $tmpDir/test.$filenameExt | $commLib/sed 's/ /^/g'`

   tmp="OUT_DATA=MAC^$MAC^";
   if [ $tmp = "OUT_DATA=MAC^^" ] ; then
      echo "OUT_DATA=ERROR^No data found for $in_param^";
      exit;
   fi
   echo $tmp

   rm $tmpDir/test.$filenameExt
   exit;

fi
```

Java Code for Retrieving Subscriber or Provisioning Information

You can use an application that is running on an HTTP server to retrieve subscriber or provisioning information from an external source.

A sample application that you can modify according to your needs is in the README file in the following locations:

- **Linux and Solaris**
  - /opt/CSCOcbt/samples/db_script/oracle/
  - /opt/CSCOcbt/samples/db_script/sybase/
- **Windows**
  - %CSCOCBT_ROOT%\samples\db_script\oracle\

  - %CSCOCBT_ROOT%\samples\db_script\sybase\

This section provides a printed copy of that README file, formatted for this user guide.

Purpose of Java Code

This sample script file describes how to use Data Manager, a sample HTTP application. The HTTP directory contains this sample application that you can modify to meet your needs. Data Manager demonstrates how to:

- Receive requests from CBT
- Query data against an Oracle database
- Return the data to CBT in the format specified in the "Retrieving Data with an HTTP Application" section on page 3-29

Sample HTTP Application Requirements

The sample HTTP application:

- Requires the Oracle database running with the schema described in create_tbls.sql.
- Is written to use Oracle OCI, instead of the pure Java JDBC, to communicate with the database. To use the sample HTTP application on a Solaris machine, see the "Troubleshooting the Sample HTTP Application" section on page 8-8.

Module and File Descriptions

This section lists the modules and files used in Data Manager, the sample HTTP application. For a description, refer to a specific module or file. To modify Data Manager to suit your needs, refer to the next section, "Modifiable Components in the Data Manager Application."

- Java modules:
  - DataManager.java
  - Db.java
  - DbProps.java
  - DbQuery.java
  - GetResponse.java
  - HttpManager.java
  - HttpObj.java
  - HttpRequest.java
  - HttpResponse.java
  - HttpServer.java
  - LoadOracleJDBCDriver.java
  - Logger.java
  - PostResponse.java

  - ShutDown.java
- System and configuration files:
  - Makefile
  - Db-Vital.properties
  - Db-Sigma.properties
  - install
  - start_dm
  - stop_dm
- SQL files:
  - create_tbls.sql
  - insert_cmts.sql
  - insert_subscriber.sql

Modifiable Components in the Sample HTTP Application

- **create_tbls.sql, insert_cmts.sql, insert_subscriber.sql**—Scripts that create database tables and insert data into the database.
- **DbQuery.java**—Module that queries the data from the database. The following functions were written based on the schema described in create_tbls.sql:
  - getMacByIP
  - getMacByPhone
  - getProvInfo
  - getProvInfoExt
  - getFiberNode
  - getSubscriberInfo

  If create_tbls.sql is changed, the code for these functions must be changed accordingly.
- **Db-Sigma.properties**—Contains attributes of the subscriber database.
  Db-Vital.properties—Contains attributes of the provisioning database.

  If the provisioning and subscriber data tables reside in the same database, these two files must have identical information. In Data Manager, the provisioning and subscriber data is in two separate databases.

  For both files, the following settings must be changed accordingly:
  - db_sid
  - db_port
  - db_userid
  - db_password
- **Makefile**—Compiles the Java files.
- **start_dm**—Starts the Data Manager HTTP application.

  The current variable settings in start_dm are:

```
debug_flag=1
(Send debug messages to a log file.)
port=8012
(Change to the port of your choice.)
logDir= /opt/CSCOcbt/jakarta-tomcat-4.0.3/logs
(Specify the location where the log file should be created and changes should be
written.)
```

- **stop_dm**—Stops the Data Manager HTTP application.

  The following variable settings are in stop_dm:

```
ip=171.71.50.52
(IP address where the HTTP application is running.)
port=8012
(Port in which the HTTP application is running. This should be the same port specified
in the start_dm script.)
```

  For these three scripts (Makefile, start_dm, and stop_dm), the following properties must be updated according to the system environment:

  - ORACLE_HOME
  - JAVA_HOME
  - LIBRARY_PATH

Building and Running the Sample HTTP Application

1. Build class files and copy files to destination.

   From the command line, type:

   ./make

   ./install

2. Start the application.

   From the command line, type:

   /opt/CSCOcbt/bin/start_dm

3. Stop the application.

   From the command line, type:

   /opt/CSCOcbt/bin/stop_dm

4. Link Data Manager to CBT.
   a. Start CBT. Log in as an **Admin** user.
   b. Click **Set External Interfaces**.
   c. Under Subscriber Information, select the **HTTP** radio button and enter the URL for the application in the **URL Location** text box:

      http://*your_machine_name:port_num*/**DataManager**

5. Under Provision Information, select the **HTTP** radio button and enter the URL for the application in the **URL Location** text box:

   http://*your_machine_name:port_num*/**DataManager**

Note In the previous steps *port_num* is the port number that is configured in start_dm.

Troubleshooting the Sample HTTP Application

- Failed to create database connection.

  Make sure there is connectivity between the machine and the Oracle database. A quick way to test this is to invoke the Oracle SQL*Plus application. For example, assume that the following settings are in Db-Vital.properties:

  - *db_sid*=CBT
  - *db_userid*=user1
  - *db_password*=mypasswd

  From the command line, type:

  sqlplus user1/mypasswd@CBT

  If there is no response or an error occurs, check the tnsnames.ora setting.

- Makefile failed - javac not found.

  Make sure that JAVA_HOME in the Makefile points to the correct JAVA compiler.

- Makefile failed while linking or missing Oracle library error generated during run time.

  The sample HTTP application was written to use the Oracle OCI protocol for communicating with the database. Oracle OCI requires machine-dependent libraries, so make sure that the path to the libraries is set correctly in Makefile.

- Socket creation error.

  This error occurs when *port_num* is changed in start_dm and stop_dm. To correct it, restart the program /opt/CSCOcbt/bin/start_dm program.

- DbQuery returns error status.

  It is possible that some of the Oracle environment did not get set up correctly. To check this, review oracle.csh, update it as required, and source it. Then, run start_dm again.

Sample Java Code

1. Set up the HTTP Connection to receive a request from CBT (HttpManager.java).

```
HttpRequest = new HttpRequest()
```

2. Retrieve the message header (HttpRequest.java):

```
StringBuffer sb = new StringBuffer();
InputStream is = sock.getInputStream();
DataInputStream fromBrowser = new DataInputStream(is);
while (true) {
        msg = fromBrowser.readLine();
        if (msg.equals("")) break;
        sb.append(msg + nl);  // put back the '\n\r' for newline
}
```

3. Retrieve the message body (HttpRequest.java):

```
DataInputStream  fromBrowser = sock.getInputStream();
StringBuffer    sb = new StringBuffer();
```

```
    for (int x=0; x < len; x++) {
            // return int range 0-255, -1: end of stream
            i = fromBrowser.read();
            if (i == -1) break;
            // however, char is 16 bits for unicode 2 bytes
            // now use 2 bytes to hold just one byte data.
            sb.append((char)i);
            //System.out.println (sb);
    }
```

4. Parse the message (HttpObj.java):

```
    //parse and save message into Hashtable

        String req_command = decodeString(command);
        //System.out.println ("req_command=" + req_command);

        // Parse the data from the servlet
        StringTokenizer rdata = new StringTokenizer(req_command, "&");

        if (rdata == null)  return;

        //System.out.println ("data=" + data + ", rdata=" + rdata);
        while (rdata.hasMoreTokens()) {
            String cmd = rdata.nextToken();
            int index = cmd.indexOf("=");
            if (index == -1) return;
            String key = cmd.substring(0, index);
            String val = cmd.substring(index+1);
            hParams.put(key, val);
        }
```

5. Process the message (HttpManager.java & DataManager.java):

```
 //if POST message then process the message:
  if (req.getReqType().equals ("POST")) {
                req.fetchBody(sock_);
                hParams = req.getParams();
                resp = new PostResponse(req);
                String status = null;
                status = dm_.processCommand(req);
  ...
  }
//examine the request type and process the message accordingly:
Hashtable hParams = req.getParams();
String task = (String)hParams.get("REQUEST");
int status = 1;

        if (task.equals(GET_PROV_INFO)) {
            String sStr = "";
            String sType =  (String)hParams.get("SEARCH_TYPE");
            if (sType.equals("MAC"))
            {
               String sTmp =  (String)hParams.get("IN_PARAM");
               String sMac = sTmp.toLowerCase();
               Logger.debug("In Mac is " + sMac);
               sStr = DbQuery.getProvInfo(sMac);
               Logger.debug("Prov Info is " + sStr);
            }
.....
        }
```

```
  // retrieve information from the database (DbQuery.java):
    public static String getProvInfo(String Mac) {

        String query = "";
        String subscriberId =  "";
        String modemIp = "";
        String cmtsIp = "";
        Statement stmt = null;

           try {
               long start = (new java.util.Date()).getTime();

               start = (new java.util.Date()).getTime();
               stmt = vital_db_conn.createStatement();
              query = "select subscriberid, ipaddress,giaddr from device where deviceid =
'" + Mac + "'";
               Logger.debug(query);
               boolean bFound = false;
               ResultSet rs = stmt.executeQuery(query);
               while (rs.next()) {
                   bFound = true;
                   subscriberId = rs.getString(1);
                   modemIp = rs.getString(2);
                   cmtsIp = rs.getString(3);
                   rs.clearWarnings();
               }
               rs.close();
               rs = null;
               stmt.close();
               stmt = null;

            long end = (new java.util.Date()).getTime();
            Logger.debug("total msec to execute getProvInfo: " + (end - start));
            if (!bFound)
               return ("OUT_DATA=ERROR^" + Mac + " does not exist in the Provisioning
Database.^");
            String sStr = "OUT_DATA=PROVISION^" + cmtsIp + "^" + modemIp + "^";
            return (sStr);
           }
           catch (SQLException se) {
               String sError = "OUT_DATA=ERROR^" + se.getMessage() + "^";
               if (stmt != null)
               {
                  try {
                     stmt.close();
                  }
                  catch (Exception e)
                  {
                  }
                  stmt = null;
               }
               return sError;
           }
    }
```

6. Build a Post-HTTP Response to return data to CBT (HttpResponse.java):

```
        //send header to CBT
        OutputStream os = null;
        try {
                os = sock.getOutputStream();
        }
```

```
        catch (IOException e) {
                Logger.error("DataManager.HttpResponse:sendHeader: error: " + e);
                e.printStackTrace(DataManager.getLogPW());
                e.printStackTrace();
        }

        StringBuffer sb = new StringBuffer();
        sb.append ("HTTP/1.0 200 OK" + nl);
        sb.append ("Content-type: text/html" + nl);
        byte[] b = (sb.toString()).getBytes();
        try {
                os.write(b);
                os.flush();
        }
        catch (IOException e) {
                Logger.error ("DataManager.HttpResponse:sendHeader: error write: " + e);
                e.printStackTrace(DataManager.getLogPW());
                e.printStackTrace();
        }

Send result data off to CBT

        os.write(result_data);
        os.flush();
```

INDEX

Symbols

2-1

.INI files

- CONFIGS.INI **3-16, 3-17**
- GUNSLINGER.INI **3-16, 3-22**
- POLLER.INI **3-16, 3-19**
- setting parameters **3-16**
- SPECTRUM.INI **3-16, 3-20, 5-2**

A

Applet to Servlet hostname mapping **5-1**

B

BPR **3-28**

C

cable qos profile command **3-4**

Carrier-to-Noise ratio **4-9**

CBT 3.3

- adding router information **3-13**
- adding subscriber information **3-15**
- CBT Log In Window **2-4, 3-12**
- configuring JRE **3-1**
- installing and starting on Linux **2-6**
- installing and starting on Windows **2-5**
- Minimum Cisco Universal Broadband Router and Cisco IOS Requirements **1-5**
- passwords, users, and data **3-10**
- poller **6-4**
- Spectrogram **4-7**
- Spectrum Management Tools **3-1**
- Subscriber Traffic Management **3-4**
- system requirements **1-2**
- TCP ports **2-1**
- uninstalling **2-7**
- verifying installation status **6-3**
- verifying license **6-3**

CBT Task Menu **3-5**

Check CPU Util dialog box **4-16**

CiscoView

- configuring **3-4**
- installing and starting on Solaris **2-4**

CMTS Dashboard dialog box **3-37**

CNR **3-29**

CNR Analysis **4-13**

CNR Analysis Criteria dialog box **4-10**

CNR Analysis dialog box **4-9**

CNR Analysis Playback Criteria dialog box **4-13**

CNR Analysis Playback Result dialog box **4-14**

commands (Cisco IOS)

- cable qos profile **3-4**
- configure terminal **3-4**
- show cable modem registered **3-4**

configure terminal command **3-4**

Continuous Sweep Spectrum Operation **6-1**

CPU Assessment **4-16**

CPU Util dialog box **4-16**

D

DST OS patches **1-5, 2-6**

E

external interfaces 3-22

F

Flap List Analysis 3-38, 3-40, 3-41
Flap List Analysis Details dialog box 3-40
Flap List Analysis dialog box 3-39
Flap List parameters 3-42
Flap List purging 3-43

H

hostname mapping 5-1

I

Import Subscriber Data dialog box 3-16, 3-30

J

Java Runtime Environment 1-5, 3-1
JRE 3-1
    OS-level requirements 2-6

L

LDAP 3-28
Linux 1-3, 2-6
Linux Operating System 1-5
local database 3-29, 3-31
Locate a Broadband Modem dialog box 3-36
Locate Broadband Modems dialog box 3-35

M

Manage Users dialog box 3-13
MIBs 1-6

O

Operating Systems
    CBT 3.3 support 1-4

P

passwords
    disabling or enabling text display 3-11
Purge Flap List Data File dialog box 3-43

R

Real-Time Modem Status Report dialog box 3-34
Real Time Status of a Broadband Modem dialog box 3-36

S

Server Ports in XML Script 6-2
Set External Interfaces dialog box 3-23
show cable modem registered command 3-4
SNMP
    enabling on routers 1-6
    MIBs in Cisco CBT 3.3 1-6
Solaris 1-4
Solaris Operating System 1-5
Spectrogram 4-7
Spectrogram Criteria Confirmation dialog box 4-8
Spectrogram Criteria dialog box 4-8
Spectrum Analysis
    administraive parameters 5-2
Spectrum Polling 4-16
Subscriber Traffic Management 3-4
Sunfire V100 1-2
Sunfire V210 1-2
Sun Fire V240 1-2

Sun Fire V440 1-2
System Message Logs 6-2

T

TCP port
- 2100 2-2, 6-2
- 2101 2-2
- 2102 2-2
- 2500 2-2, 6-2
- 2640 2-1, 6-1
- 3100 2-1, 6-2
- 3500 2-1, 6-2
- 5100 2-2, 6-2
- 5500 2-2, 6-2
- 8005 2-1
- 8012 7-7
- 8020 2-1, 6-1
- 8040 3-20
- 8080 2-1
- 8105 2-1, 6-1, 6-2
- 9080 2-1, 6-1, 6-2
- 9082 2-1, 6-1, 6-2
- 9443 2-1, 6-1, 6-2

TCP ports 2-1
Tomcat Application Server 1-5
Trace Window 4-1
Trace Window Criteria dialog box 4-5
Trace Window Playback Criteria dialog box 4-11
Trace Window Playback List dialog box 4-12

U

UNIX
- UNIX client machine with Solaris 1-4
- UNIX server systems with Linux 1-3
- UNIX server systems with Solaris 1-3

UNIX Client Machine with Linux 1-4
UNIX Client Machine with Solaris 1-4
UNIX server Systems with Linux 1-3
UNIX Server systems with Solaris 1-3

W

Web Server Applications
- CBT support 1-4

Widnows
- PC Server systems for Windows 1-4

Windows 2000 1-2
Windows Operating System 1-5
Windows PC Client Machine 1-4
Windows PC Server systems 1-4
Windows systems 1-3
Windows XP 1-2
workflow
- administrator and RF technician tasks 3-9