MCN:0081792-01
Revision B
December 12, 2002

CopperEdge®

Installation and Operating Guide

CopperEdge 200 DSL Concentrator

Copper Mountain Networks, Inc.

San Diego Facility:
10145 Pacific Heights Boulevard
San Diego, California 92121

Palo Alto Facility:
1850 Embarcadero Road
Palo Alto, California 94303

Important Information about Networking Models

Beginning with Release 7.0, the former HDIA and Copper-VPN networking models have been superseded by a new, more capable, more secure, and much easier-to-use net-Model called CopperVPN+ (referred to in the software as simply CopperVPN). The functionality of the older netmodels continues to be supported for backward compatibility, but those configurations, when saved, are upgraded to the new CopperVPN netModel and format automatically.

Part Number 0081792-01
Revision B
December 12, 2002

Limited Warranty

Copper Mountain Networks, Inc. ("Copper Mountain") warrants the CopperEdge® 200 to be free from defects in materials or workmanship for a period of one (1) year from the date of shipment from Copper Mountain's factory. Should your CE200 fail during the warranty period, Copper Mountain will, at its option and as its sole and exclusive obligation under this warranty, repair or replace it with a like product, which may include new or refurbished parts or components. This warranty is extended only to the original purchaser and only covers failures due to defects in materials and workmanship which occur during normal use during the period of the warranty. It does not cover damage which occurs in shipment or failures resulting from misuse, negligence, accident, improper storage, installation or testing, unusual electrical stress, fire, lightning, other environmental hazards, unauthorized attempts at repair, operation inconsistent with published electrical and environmental specifications, or if the Product was maintained in a manner other than described in this document, or if the serial number or other identifications markings have been altered, removed or rendered illegible. Expendable components such as batteries or cabling external to the unit are not covered by this warranty.

For specific terms and conditions of the product warranty and Copper Mountain's obligations thereto, please refer to the warranty section of your purchase agreement.

In order to exercise your rights to repairs under this warranty, you must first contact Copper Mountain's Customer Service Department at 888-611-4266 (from US locations) or 858-410-7100 (from outside the US) to obtain authorization (including tracking number) and instructions for return of the product(s).

EXCEPT FOR THE EXPRESS WARRANTIES SET FORTH HEREIN AND IN THE APPLICABLE PURCHASE AGREEMENT, COPPER MOUNTAIN MAKES NO WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY AND HEREBY EXPRESSLY DISCLAIMS ANY AND ALL SUCH WARRANTIES, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE PRODUCTS PURCHASED. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT, WHETHER MADE BY COPPER MOUNTAIN OR OTHERWISE, SHALL BE DEEMED TO BE A WARRANTY FOR ANY PURPOSE OR GIVE RISE TO ANY LIABILITY OF COPPER MOUNTAIN.

The laws in some localities do not allow limitations on the warranty period so the above exclusions or limitations may not be applicable. This warranty provides specific legal rights. You may have other rights with respect to this product, depending on local laws.

Software License Agreement

Use of this Product is subject to the Software License Grant contained in the purchase agreement used to originally purchase this Product from Copper Mountain.

Except as expressly permitted by applicable law, Customer or any third party to which the software may be licensed as permitted under the original purchase agreement, may not alter, modify, adapt or translate, create any derivative works based on the software, reverse engineer, decompile, disassemble or otherwise attempt to derive the human-readable source code for the software licensed with this Product.

FCC Information

This device complies with Part 15 of the FCC rules. Operation is subject to the following two conditions:

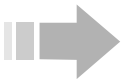
(1) This device may not cause harmful interference, and

(2) This device must accept any interference received, including interference that may cause undesired operation.

Changes or modifications to this device not expressly approved by the party responsible for compliance could void the user's authority to operate this device.

W A R N I N G

This equipment has been tested and found to comply with the limits for a Class "A" digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause interference to radio communications. Unprotected operation of this type of commercial equipment in a residential area is likely to cause harmful interference which the user would be required to correct



N O T E

The CopperEdge™ 200 was FCC verified under test conditions that included the use of shielded I/O cables and connectors on certain system components. To be in compliance with FCC regulations, you must use properly installed shielded cables and connectors on all connections to the System Control Module, and the V.35 and DS3 Frame WAN Modules. Shielding is not required on cables to DSL port connectors (LJ1 through LJ8 on the rear of the equipment chassis), the alarm status connector (JA2 on the rear of the chassis), or on the DC power input connections.

10145 Pacific Heights Blvd.
Suite 100
San Diego, CA 92121 USA

Declaration of Conformity

We, Copper Mountain Networks, Inc., declare under our sole responsibility that the following product to which this declaration relates is in conformity with the Essential Requirements and Harmonised Standards identified below.

1999/5/EC of the European Parliament and of the Council relating to Radio & Telecommunications Terminal Equipment. *The mutual recognition of conformity with this directive is based on compliance with the following Essential Requirements*:

89/336/EEC, EMC Directive

EN 55022:1998, Limits and methods of measurement of radio interference characteristics of Information Technology Equipment (ITE)

EN 55024:1998, Information Technology Equipment - Immunity Characteristics - Limits and Methods of Measurement

73/23/EEC, Low Voltage Directive

EN60950:1992, Safety of Information Technology Equipment, Including Electrical Business Equipment, Including amendments A1:1993, A2:1993, A3:1995, A4:1997, A11:1997

PRODUCT

Manufacturer	Copper Mountain Networks, Inc.
Trade Name/Model Number	CopperEdge 200 DSL concentrator, CE200

Alternate Trade Names/Model Numbers-Identical to that shown above except for Trade Name/Model Number.

- CopperEdge 200 made for Lucent Technologies by Copper Mountain Networks, Inc.

Year of First Issue 2000

Tom Lavka, Member of Technical Staff
Typed Name and Title of Representative

+01.858.410.7110, tlavka@coppermountain.com
Telephone Number & E Mail address

Signature of Manufacturer's Representative

May 15, 2000
Date

Form Copper Mountain: EU D of C

Issue: 05/15/00, Rev B

Industry Canada CS-03 Telecommunications Notice

The Industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective operational and safety requirements in the appropriate Terminal Equipment Technical Requirements documents. The Department does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be coordinated by a representative designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines, and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

C A U T I O N

Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

The Ringer Equivalence Number (REN) of this device is 0.6.

The Ringer Equivalence Number (REN) assigned to each terminal device provides an indication of the maximum number of terminals allowed to be connected to a telephone interface. The termination of an interface may consist of any combination of devices subject only to the requirement that the sum of the Ringer Equivalence Numbers of all the devices does not exceed 5.

This equipment uses Canadian CA11A Jacks.

Table of Contents

List of Figures

Revision History

Document MCN & Date	Release	Summary of Changes
0081792-01	7.0-L	Limited Availability release.
DDC 4/25/02	7.0-L	Regenerated book and PDF to correct formatting inconsistencies.
DDC 4/30/02	7.0-L	Updated Chapters 5, 6 and 9 to incorporate materials omitted from original draft.
0081792-01	7.0-GA	General Availability release.
DDC 11/6/02	7.0-GA	Replaced pp 42-49 to incorporate latest CopperVPN MIB tables and correct errors in CopperVPN configuration procedure.
DDC 11/25/02	7.0-GA	Minor, non-substantive corrections relating to current CPE availability.
Rev. B 12/12/02	7.0-GA	Revised procedures for obtaining service and support and Return Authorization (Ch. 1 and Ch. 6) for consistency with current service offerings and policies.

System Software and Applicability

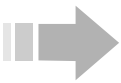
This document applies to CopperEdge 200 systems delivered under Hardware/Software Release 7.0 (General Availability Release).

Document Conventions

The following conventions are used throughout this document:

`this bold-face font`	Indicates commands that you type in order to command and control the *CopperEdge*. Examples of typed input are preceded by the system prompt, shown as "`Craft>`" or "`System>`"
`This fixed-space font`	Indicates output from the system that displays on your screen.
This italic font	Emphasizes new terms, names, titles, or trademarked words, and is also used to highlight examples.

Throughout this document, you will encounter examples of configurations or commands showing link- or user-specific information such as IP addresses, MAC addresses, etc. Unless otherwise specified, all such data is fictitious and is provided for illustrative purposes only.



N O T E

Information or instructions to which you should pay particular attention.

C A U T I O N

Information alerting you to a hazard, either to personnel or to the systems and equipment.

C A U T I O N

Information alerting you to an electrostatic discharge hazard which could damage equipment or cause the loss of stored information.

W A R N I N G

Information alerting you to a situation that could result in damage to the network, and/or violation of local or national laws.

CopperEdge 200 DSL Concentrator

Chapter 1
Introducing the CopperEdge 200

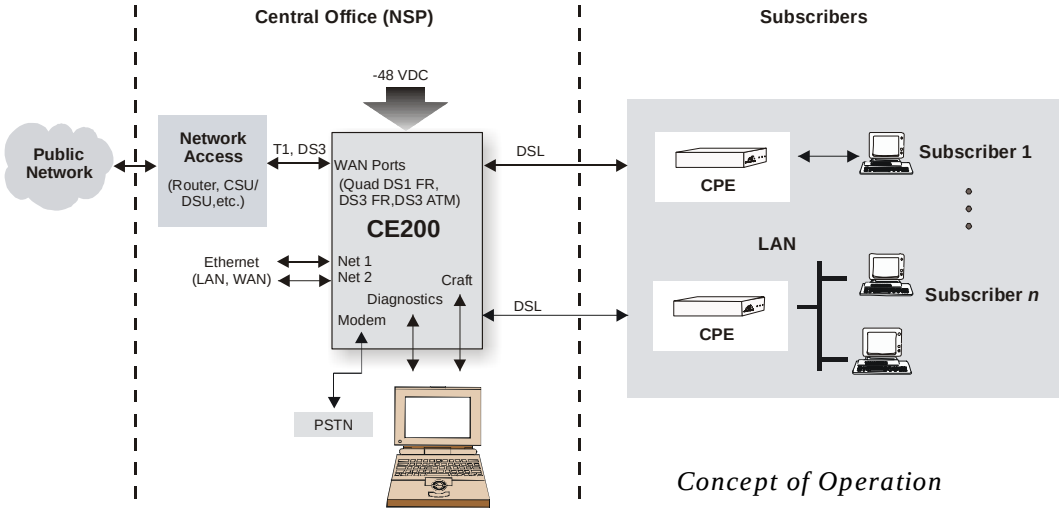
The *CopperEdge*™ 200 (CE200) DSL Concentrator is the central-office half of Copper Mountain's end-to-end high speed DSL connectivity solution for Network Service Providers (NSPs) and their subscribers. Each CE200 unit can distribute DSL data packets at high speeds to and from as many as 192 physical subscriber lines over two-conductor twisted-pair copper wire.

At the subscriber premises, a high-performance DSL access device such as Copper Mountain's *CopperRocket*™ family of Customer Premise Equipment (CPE) seamlessly connects the subscriber's PC or LAN, through the DSL physical link, to the CE200 at the Central Office, and from there to the high-speed facilities of the network.

Concept of Operation

The following illustration shows a typical deployment of the elements of the Copper Mountain DSL system.

At the subscriber site, the Customer Premise Equipment (a DSL Modem or Integrated Access Device) connects to the subscriber's Ethernet LAN or PC. The CE200 receives data from each DSL link (subscriber line) at the central office. It adds packet header information to identify each of its connected subscribers, and aggregates the subscriber lines for transmission over the network.

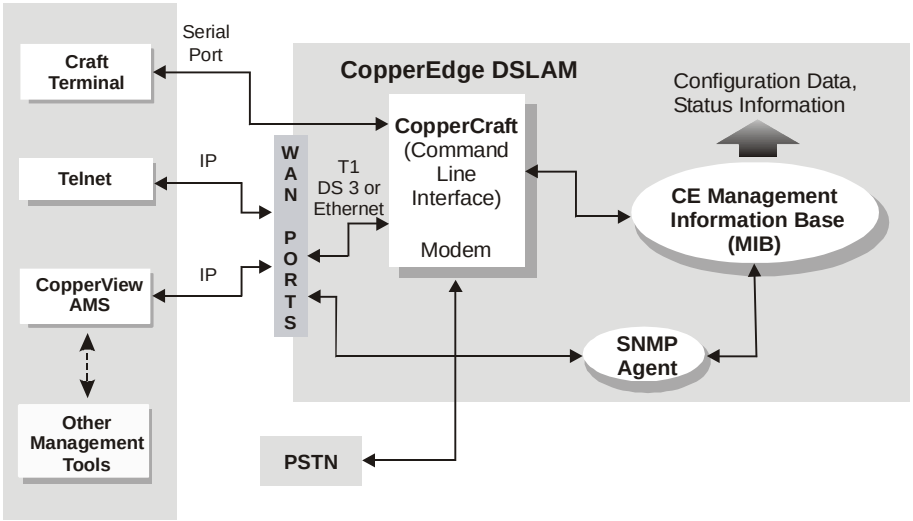


Concept of Operation

Management Tools

Software to configure and manage the system consists of the *CopperCraft*™ command line/Craft interface (also accessible remotely via Telnet), and an SNMP agent. An external graphic user interface is available through the Element Manager software component of the *CopperView*™ Access Management System (AMS). The CopperView AMS facilitates centralized remote control and management of multiple CopperEdge concentrators through SNMP. The CE200's software tools may be used individually or together to configure the CE200 and each of its links and to monitor status and performance.

As shown in the following illustration, the CopperCraft interface provides an intelligible, text-based mechanism by which an operator can directly monitor and control the CE200. When controlled by the CopperView Element Manager or other network management system, the CE200's SNMP agent responds to commands originated by another computer (SNMP Manager).



Status/Configuration Data Flow

Third-Party Customer Premise Equipment (CPE)

As a leader in the xDSL standardization initiative, Copper Mountain conceived and operated a comprehensive program to facilitate interoperability of the CE200 with the DSL CPE of other manufacturers.

A large number of manufacturers have submitted their CPE (operating with various DSL formats, protocols and digital rates) for test and certification of their interoperability with the CopperEdge family of concentrators. Check with your Copper Mountain sales representative for a current list of *CopperCompatible* CPE products.

The full collection of CPE Application Notes, with instructions for configuring your system to operate with CopperCompatible CPE, can be found on the Copper Mountain Support website.

Hardware Features

Specifications

All specifications are subject to change without notice. New modules are being added all the time. In addition, special releases occur; supplemental documentation describes those modules and their releases. This documentation is later incorporated in the standard operating guides.

DSL Platform

- Flexible, modular design supports a broad range of DSL interfaces, transmission protocols, and backbone network technologies
- Eighteen-slot chassis; up to eight DSL modules (maximum of 192 physical ports), in any combination of SDSL, IDSL, ADSL (G.dmt or G.lite), or DS1/DSL per CE200 shelf

Physical Dimensions

- Overall Dimensions:
  21" (53.3 cm) H x 18" (45.7 cm) W x 10" (25.54 cm) D
- Weight (approximate) with full circuit-module complement: 89 lbs (40.05 kg)
- Rack mounting options: EIA 19 inches (48.3 cm) or 23-inch (58.4 cm) CO style

Input Power

- -40 to -60 Volts DC
- 15 amps @ -48 VDC max.

Operating Environment

- Temperature: 32° to 122°F (0° to 50°C)
- Humidity: 10 to 80%, non-condensing
- Altitude: to 10,000 ft/3,048 m

System Control Modules

- SCM-1:
  - ⇒ Intel Pentium processor (P54C)
  - ⇒ Up to 200 MHz internal clock, 66 MHz front side bus (FSB)
  - ⇒ Up to 512 Kbytes of secondary cache (L2 cache)
  - ⇒ 32 MB DRAM, 66 MHz
  - ⇒ 8 MB of on-board flash memory
  - ⇒ One 10Base-T/100Base-TX Ethernet port with an RJ-45 connector
  - ⇒ Two DB9 male serial ports (Craft and Diagnostic) with EIA/TIA-232 Serial Interface connectors
- SCM-2 is the same as the SCM-1, with these exceptions:
  - ⇒ 128 MB DRAM, 66 MHz
  - ⇒ 8 MB of on-board flash memory; minimum of 40 MB of on-board flash memory through an IDE flash module

- SCM-3:
  - ⇒ Intel Pentium III processor (EMC2)
  - ⇒ 500 MHz internal clock, 100 MHz front side bus (FSB)
  - ⇒ 256 kBytes of secondary cache (L2 cache)
  - ⇒ 256 MB DRAM, 100 MHz
  - ⇒ 64 MBytes of on-board flash memory through a Compact-Flash type 1 module
  - ⇒ Two 10BASE-T/100BASE-T Ethernet ports, RJ-45
  - ⇒ Two 16C550 PC-Compatible serial ports (Craft and Diagnostic) with RJ-45 connectors
  - ⇒ 33,600 kbps analog modem with error correction and data compression support
  - ⇒ Pinhole reset switch
  - ⇒ For optimal system performance, we recommend you use the Buffer Control Module 2 with the SCM-3

Buffer Control Modules

- BCM-1:
  - ⇒ i960CF (33 MHz) processor
  - ⇒ 4 kB of internal instruction cache
  - ⇒ 1 kB of internal data cache (direct-mapped, write-through)
  - ⇒ 1 kB internal SRAM
  - ⇒ Can be used with SCM-1, SCM-2, or SCM-3
- BCM-2:
  - ⇒ Clock-doubled i960HD (33/66 MHz) processor
  - ⇒ 16 kB of internal instruction cache (four-way set associative)
  - ⇒ 8 kB of internal data cache (four-way set associative)
  - ⇒ 2 kB internal RAM
  - ⇒ Can only be used with SCM-3

DSL Line Modules

- SDSL 30X module (24 physical ports): multi-speed 1.568 Mbps, 1.040 Mbps; and 784, 416, 320, 208, and 160 kbps 2B1Q
- IDSL module (24 Ports): multi-speed 144, 128, and 64 kbps 2B1Q for use in systems offering IDSL service
- ADSL G.lite module (24 physical ports): multi-speed 64 kbps to 2.336 Mbps
- ADSL G.dmt module (24 physical ports): multi-speed 64 kbps to 6.144 Mbps
- G.SHDSL Module (24 physical ports): multi-speed *n x 64*, 19.2 kbps to 2.304 Mbps
- DS1/DSL Module (12 physical ports): 1.544 Mbps ± 50 bps
- Input/Output Impedance at all DSL ports: Balanced, 135 Ω

WAN Modules

- Two-port Universal Serial I/O—EIA-449 (CCITT V.36), X.21, and V.35
- One-port DS3 (45 Mbps) Frame Relay, dual (Tx/Rx) 75-Ohm Coaxial
- One-port DS3 (45 Mbps) ATM, dual (Tx/Rx) 75-Ohm Coaxial
- Four-port DS1 (T1 PMC), 100-Ohm unshielded twisted pair, using modular RJ-48C connectors
- One-port OC-3c/STM-1 (155.52 Mbps), Single Mode fiber interface
- One-port OC-3c/STM-1 (155.52 Mbps), MultiMode fiber interface
- Eight-port T1/E1 IMA
- Up to 2048 WAN VCs can be supported per CE200

Software Features

DSL Link Protocols

- Internet RFC-1483 Multiprotocol Encapsulation over Frame-Based User-to-Network Interface (FUNI)
- DSL ports also support RFC-1490, Q.922, Q.922-1490, 1973, 2364, HDLC, and HDLC PPP as implemented in various third-party CPEs

Packet Multiplexing Protocols

- TCP/IP
  - ⇒ RFC-791 Internet Protocol (IP)
  - ⇒ RFC-792 Internet Control Message Protocol (ICMP)
  - ⇒ RFC-768 User Datagram Protocol (UDP)
  - ⇒ RFC-826 Ethernet Address Resolution Protocol (ARP)
  - ⇒ RFC-1058 Gateway and Hosting Protocol
- Virtual Wide Area Net (VWAN) multiplexing and CopperVPN IP multiplexing
- Frame multiplexing for third-party CPEs
- Policy-based packet multiplexing
- IP filtering using source or destination address or port

Network Standard Protocols

- ANSI T1.606 Frame Relay Architectural Framework
- ANSI T1.606 Addendum 1 for Congestion Management –CIR/BIR/BECN/FECN support
- ANSI T1.618 Core Aspects of Frame Protocol for Frame Relay
- Multiple standards of LMI management protocols including T1.617 Annex D, Q.933 Annex A, and LMI Rev 1.0a
- Frame Relay Forum UNI FRF.1
- Frame Relay NNI (FRF.2) supported on WAN links
- Frame Relay/ATM service internetworking per (FRF.8)

- RFC-1157 SNMP Protocol
- RFC-1483 Multiprotocol Encapsulation over ATM
- RFC-1490 Multiprotocol Encapsulation over Frame Relay
- RFC-1973 PPP over Frame Relay
- RFC-2364 PPP over ATM
- Inverse ARP over Frame Relay (per RFC-2390)
- Inverse ARP over ATM (per RFC-2225)

Network Management

- Full SNMP functionality
- Supported MIBS include subsets of Internet RFC-1213 MIB-II, RFC-1315 Frame Relay MIB, RFC-1573 DS3/ES MIB, RFC2558 SONET MIB, and full support of various Copper Mountain Proprietary MIBs
- Proxy management for Copper Mountain CPEs
- EIA-2332D Console port/Craft interface
- Telnet (password protected) to CopperCraft CLI
- CopperCraft interface accepts up to four simultaneous management sessions through IP (Telnet and FTP), up to five sessions through SNMP, and one through the Craft serial port
- Ping utility (both originate and respond)
- CopperView DSL Access Management System with graphics-based element manager, a self-contained SNMP Manager
- Radius authentication and accounting for system-level operators, including those on the CE200

Reliability/Serviceability

- Hot-swap supported for all DSL modules; front-panel status indicators to aid fault isolation
- Control and WAN redundancy supported when equipped with optional redundant modules (System Control, Buffer Control, and WAN Modules)
- Optional power redundancy through dual load-sharing power supplies with automatic failover and fault monitoring
- Redundant inputs (A&B) for -48 VDC primary power
- Software and configuration downloads from flash memory or from external server through FTP
- Extensive performance monitoring through system-, trunk-, and port-level statistics
- Event and Alarm log, and extensive event trapping capability provide status information to NMS/SNMP Managers; front panel module shows alarm status, and network of contact closures enable audible and visible CO alarm devices

Certification

United States and Canada

- Network Equipment Building Systems (NEBS): Meets or exceeds NEBS levels 1, 2, and 3 and the following Bellcore Specifications:
  - ⇒ NEBS: GR-63-CORE
  - ⇒ Electromagnetic Compatibility and Electrical Safety Generic Criteria for Telecom Equipment: GR-1089-CORE
  - ⇒ Generic Physical Design Requirements for Telecom Products: TR-NWT-000078, Issue 3
- Product Safety:
  - ⇒ UL 1950 3rd Edition, Safety of Information Technology Equipment, Including Electrical Business Equipment, and CAN/CSA C22.2 No. 950-95 3rd Edition. UL/CUL Recognized Component
- Product Emissions:
  - ⇒ Federal Communications Commission (FCC), CFR 47, Part 15, Subpart B, Class A

European Union

- Product Emissions:
  - ⇒ EN 55022:1998, Limits and methods of measurement of radio interference characteristics of Information Technology Equipment (ITE), Class A.
- Product Immunity:
  - ⇒ EN 55024:1998, Information Technology Equipment - Immunity Characteristics - Limits and Methods of Measurement
  - ⇒ IEC 61000-4-1:1992, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Section 1: Overview of immunity tests. Basic EMC Publication
  - ⇒ IEC 61000-4-2:1995, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Sect. 2: Electrostatic discharge immunity test. Basic EMC Publication
  - ⇒ IEC 61000-4-3:1998, Electromagnetic compatibility (EMC) - Part 4-3: Testing and measurement techniques - Radiated, radio-frequency, electromagnetic field immunity test
  - ⇒ IEC 61000-4-4:1995, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Section 4: Electrical fast transient/burst immunity test. Basic EMC Publication
  - ⇒ IEC 61000-4-5:1995, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Section 5: Surge immunity test

⇒ IEC 61000-4-6:1996, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Section 6: Immunity to conducted disturbances, induced by radio-frequency fields

⇒ IEC 61000-4-8:1993, Electromagnetic compatibility (EMC) - Part 4: Testing and measurement techniques - Section 8: Power frequency magnetic field immunity test. Basic EMC Publication

- Product Safety:
  - ⇒ EN60950:1992, Safety of Information Technology Equipment, Including Electrical Business Equipment, Including amendments A1:1993, A2:1993, A3:1995, A4:1997, A11:1997
- Product Telecommunications:
  - ⇒ G.lite Line Card, ITU G.992.2, ATU-C Characteristics
  - ⇒ G.dmt Line Card, ITU G.992.1, ATU-C Characteristics
  - ⇒ SDSL Line Card, ETS TS 101 135 V1.5.1 (1998-11) for Pulse-shape, PSD output power and Electrical Characteristics of a single 2B1Q Transceiver
  - ⇒ IDSL Line Card, ETSI ETR 080, Nov 1996, Transmitter Output Characteristics and Transmitter/Receiver Termination
  - ⇒ T1 Line Card, ANSI T1.403-1999, Network and Customer Installation Interfaces - DS1 Electrical Interface.
  - ⇒ G.SHDSL Line Card, ITU-T, G.991.2

Release Notes

For complete information about new features, hardware/software compatibility, and late-breaking issues, consult the Release Notes document corresponding to the version of hardware and installed software in this CE200. Electronic copies of release notes, along with complete user documentation for all hardware and software documentation are available on line at the Copper Mountain web site, **http://www.coppermountain.com**.

From the Copper Mountain home page, click on ***Support***, and then log onto ***CopperCSO Interactive***. If you have purchased an Extended Service Agreement, simply use your assigned access codes. If you do not have a service agreement, you can obtain limited access to the site using the following username and password:

```
Username:   coppercso
Password:   customer
```

Technical Support

Expert help with your Copper Mountain equipment and software is available through our Technical Assistance Center (TAC). Optional service agreements are available which can provide you with basic telephone support, extended (24-hour, seven-day) support, and other premium services; contact your Copper Mountain sales representative for details.

Optional service agreements are also available for those requiring extended support (24-hour, seven-day) and other premium services. Contact your Copper Mountain sales representative for details.

For assistance or information via e-mail, contact:

support@coppermountain.com

Technical information about DSL and Copper Mountain products and technology is also available at our web site at:

http://www.coppermountain.com

Chapter 2
Installation

Installation Instructions are no longer contained in this document. For complete installation instructions, refer to the CE200 Installer Guide.

Chapter 3
Initial Configuration

This chapter describes procedures for performing initial provisioning and other basic configuration tasks. Before you attempt to modify any of the software settings, however, you should be familiar with the sections on General Configuration Guidelines, Command Structure, and the *Permanent Interface Identifier* scheme of specifying interfaces on the CE200.

The term *DSL module* is generally used throughout this chapter to mean *any supported* DSL module: SDSL, IDSL, ADSL (G.lite or G.dmt), G.SHDSL, or T1. If there is a difference in the way one or the other types of DSL modules behave or what they require, it will be explicitly specified.

System Configuration Guidelines

Configuration of the CE200 is accomplished by entering SNMP-like commands to modify the Management Information Base (MIB). This section contains instructions for querying and configuring the CE200 as part of the installation process.

Accessing the CE200

You can log in to the CE200 using the Command Line Interface (CLI) or the Graphical User Interface (GUI):

- You can use CopperView or some other SNMP-based NMS to remotely configure the CE200.
- You can use a LAN and a telnet session to access the CLI to remotely configure the CE200.
- You can use a terminal (or terminal emulator) to the Craft serial port on the SCM front panel and then employ the CLI to configure the CE200 on site.

To access the CopperCraft command line interface (the CLI) through your PC COM port, use Hyperterminal or an equivalent terminal emulation application. Set the serial port rate to 9600 bps, 8-N-1 (8 data bits, no parity, 1 stop bit). If you want to use flow control, choose the hardware (RTS/CTS) settings.

CE200 installation, on-site verification, and establishment of upstream data links are normally done using the CopperCraft interface, either through direct connection method or through Telnet. Thus, this document puts greatest emphasis on those methods.

For information on the CopperView Element Manager and its graphic user interface, see the *CopperView Element Manager, Installation and Operation Guide*.

CopperCraft Login

As the CE200 boots, the Copper Mountain Networks copyright and system information is displayed, followed by the user name prompt.

```
Copper Mountain Networks
CopperEdge200
Copyright 1996 - 2002, All Rights Reserved
System Name: Hillcrest 3
Build Date: Mar 27 2002, 20:29:54

USERNAME>
```

To log in, enter **ce200** for both user name and password.

```
USERNAME> ce200
PASSWORD> *****
```

Once you are logged in, the factory default command line prompt is displayed:

```
ce200(1.2)>
```

This default prompt consists of the system name (ce200), as defined in the Name object of the cmSystemTable, plus the slot number of the active System Control Module (1.2 or the redundant slot, 1.15). Since all CE200s initially have this default prompt, we recommend you change the system name in the cmSystemTable to a more meaningful name. You cannot change the slot number.

In this manual, we use Craft as the system name, which produces a default command line prompt of:

```
Craft (1.2)>
```

CopperCraft Logout

When you are ready to log out of the system, enter **exit**, **quit**, or **logout**.

If you changed any system settings and did not save the configuration, the following prompt is displayed:

```
CE configuration has been changed.

Save the configuration?(Y / y)>
```

Enter Y or **y** to save the configuration. If the save was successful, the following messages are displayed:

```
.....

CE configuration save succeeded.

2002/3/14-15:33:17  USER LOGGED OUT
```

If you enter a response other than Y/y, or if you do not respond within 30 seconds, the configuration will not be saved, but the logout process will continue.

If you entered Y/y to save the configuration but do not want to wait for the save process, press **Esc**. Your terminal is released, while the configuration is saved and the logout process completes.

Command Line Interface

In general, commands and other data can be entered without regard to case; either capital letters or lowercase letters are acceptable. However, to specify a directory path on a file server, use lowercase letters, as many servers are case-sensitive and will not download their software if commands are not lowercase.

If a long command line wraps, or for any reason you are in doubt about what the entire line actually contains, and in what order, type ^L (Control-L) to refresh the display of the entire command. For more information about line editing, see *CopperCraft Line Editor* on the next page.

CopperCraft Line Editor

Command strings for SNMP object groups can be very short or very long, especially when you are setting a series of new parameters in an object group. Here is an example of a short getall command, which returns data about all of the boards in the CE200's chassis:

```
Craft (1.2)> getall cmboard
```

Here is an example of a longer Set command, which sets the parameters on an ATM quality of service table:

```
Craft (1.2)> set cmcircuitparam [6]
rowstatus=createandgo servicecategory=nrtvbr
pcr=104000 scr=604 mbs=4 cdv=unspecified
```

The line editor function on the CopperCraft CLI lets you recall as many as 20 previous commands from the system's memory buffer. You can move through the command lines and edit them before you press Enter and send the command to the system.

To display the list of valid Line Editor commands, press **esc-?** at the system prompt.

```
Craft (1.2)>
^D            Delete current character
DEL           Delete current character
^H            Backspace
^E            End of line
^A            Beginning of line
Escape F      Forward one word
Escape B      Backward one word
^K            Delete to end of line
^U            Delete to beginning of line
^U^K          Delete entire line
^L            Redisplay current line
Left Arrow    Cursor left
Right Arrow   Cursor right
Up Arrow      Scroll up through history
^P            Scroll up through history
Down Arrow    Scroll down through history
^N            Scroll down through history
```

Control characters are shown with a caret (^). Press and hold the Control key, press the letter key, then release both keys.

Escape sequences are indicated by the word, Escape. Press and release the Escape key, and then press the letter key.

If you are using a terminal emulator, set it to VT100 mode. If a command line will not fit on a single line, an arrow character (< or >) appears at either end of the line indicating Scroll mode. Use the arrow keys to move to the end of the line; the next 10 characters will be displayed.

Permanent Interface Identifier (PII)

Fully populated CE200s can have over 200 physical interfaces, including DSL ports, WAN ports and management ports. When virtual interfaces (those supporting frame relay or ATM virtual circuits) are applied, the total number of interfaces rises to the thousands. All of these ports and circuits must be uniquely identifiable so they can be configured, and so the traffic over them can be monitored. To accomplish the job, the CopperCraft CLI uses a system called Permanent Interface Identifiers (PIIs).

The PII system consists of four elements, which must be written in a series format in a specific order: c.s.p.v, where the letters represent the following:

c = Chassis shelf number (always 1)

s = Slot number (1 to 18)

p = Port number (For DSL, 1 to 24; for WAN, 1 to 4)

v = Virtual circuit number (1 to 976)

Slots in the CE200 chassis are reserved for specific types of modules, as shown in the following table:

Slot No.	Module
1	DC Power Module
2	System Control Module
3-4	V.35, ATM, Frame, T1/E1 IMA, OC-3c/STM-1 or Quad T1
5	Buffer Control Module
6-13	DSL Modules
14	Redundant Buffer Control Module
15	Redundant System Control Module
16-17	Redundant WAN Modules
18	Second DC Power Module

If the ports in slots 3 to 4 and 16 to 17 are on a V.35 board, their numbers are 1 to 2 from top to bottom. If the ports are on Quad T1 boards, their numbers are 1 to 4 from top to bottom. Slot 2 always should have a System Control Module in it; slot 5 always should have a Buffer Control Module in it. The CE200 system, will not function without a System Control Module in slot 2 and a Buffer Module in slot 5.

The PII system of identifying the interfaces on a CE200 works in this way:

- A PII of 1.2.1 identifies the CE200 in chassis 1, the System Control module in slot 2, and port 1 (the Ethernet port).
- A PII of 1.3.1.41 identifies the CE200 chassis 1, the WAN module in slot 3, port 1, and virtual circuit 41.
- A PII of 1.7.24 identifies the CE200 chassis 1, the DSL module in slot 7, and port 24 (it could be SDSL, IDSL, ADSL, or G.SHDSL).

If you are using an SNMP manager to configure the CE200, you may encounter a slightly different PII format. In such cases, you enter the identifier without periods to separate the segments, and you fill parts of the fields with leading zeros. For example, a PII of 1.3.1.24 entered at the CopperCraft CLI becomes 103010024 in this SNMP-manager format.

SNMP Command Structure

Command strings for SNMP object groups must have certain elements in them, which must appear in a certain order. If elements are missing or if they are out of order, the CE200 will not be able to process the commands that you sent through the CopperCraft CLI. Instead, the system will issue an error message.

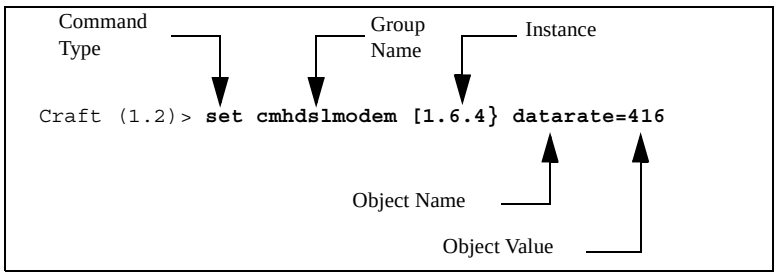
Command strings for SNMP object groups have a minimum of two elements:

- a command type
- an object group name

Command strings for SNMP object groups have a maximum of five separate elements:

- a command type
- an object group name
- an instance (typically a PII)
- one or more object names
- an object value for each object name

The illustration shows the structure of a typical *Set* command (in this case, to assign a port's DSL data rate to 416 Mbps. The port is the fourth one on a board in the sixth slot of the chassis:



Example of a Set Command

You must enter the elements of command strings in the following order, and you must adhere to the following restrictions:

- Command type: *set*, *get*, *getnext*, *getall*, *help*, *find*, *ping*.

  Always enter the command type first, separated from the group name by a space.

  For operators with Security or System privileges, the following commands are also available: elog, alarms, LCRestart, and SCMRestart.
- Group name: *cmSystem*, *cmIface*, *cmADSLModem*, etc.

  Always enter a group name for the set, get, getnext, getall, and find commands. Otherwise, the system will return an error message. All of the group names are listed alphabetically in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.
- Instance (if required): *IP address*, *PII*, etc.

Enter an instance for a Set or Get command. A port—either a WAN port or a DSL port—can be divided into VCs. To identify the VC, include the number of the VC after the number for the slot. For example, the instance, 1.3.1.30, refers to the WAN module in slot 3 and the first port. The number, 30, identifies the VC.

Always enclose an instance in square brackets [ ]. If an entry requires more than one index, place the indexes in the same order as displayed in the help objects listing. Always separate multiple indexes with a comma, but always enclose them with a set of brackets: [*index1*, *index2*]. Multiple indexes form a single instance.

- Object name: *NetModel*, *IpAddr*, *NetMask*, etc.

  Enter an object name when performing a *Set* command. (In the three different *Get* commands, object names are unnecessary.) Separate the object name from the instance by a space. Also, an object name must be accompanied by an object value.

- Object value: *possible values or value ranges are specified in each of the object group tables*.

  All of the objects and their values or value ranges are listed in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual. In the command string, separate an object value from an object name with an equal sign (=), like this:

  netmodel=coppervpn.

  Do *not* use spaces between object names and object values.

  In a Set command, you will be assigning values to multiple objects. In these situations, separate different objects (and their values) from each other with spaces, as follows:

  ```
  CRAFT (1.2)> set cmiface [1.4.24] netmodel=vwan
      destpii=1.3.1.500 encapsulationtype=rfc1483
      cmcpcompatible=yes
  ```

  You can enter multiple Get commands on the same line by separating them with a semicolon, as in the following example:

  ```
  CRAFT (1.2)> get cmiface [1.4.24]; get iftable [1.2.1.109]
  ```

Any of the CopperCraft object names can be truncated up to the point of its shortest unique character string. For example, in the sample Set command above, cmIface could be abbreviated to cmif, and ifTable could be abbreviated to ifta.

Some object names may also be entered using a shorter alias. Alias strings that differ from the object name, and that consist of something other than a truncated version of the name, are shown in the MIB-definition tables in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Whenever you assign IP addresses or subnet masks for the CE200, accuracy is essential; always issue a *Get* command to verify the entry when you are done. The CE200 will report an error if you try to assign the same IP address to more than one of its interfaces, but it does not currently report an error if two IP addresses for the same subnet are assigned to separate interfaces.

Helpful Shortcuts: Getall and Find

In addition to the standard SNMP commands, the CopperCraft interface provides two special command, Getall and Find. They are briefly described below. For more complete information about these commands, see the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Getall

The Getall command allows you to review the status of all objects for all instances of a specified group at the same time. The output is more condensed than the output for a Get command.

The basic syntax is:

getall *mib_group*

Since the list could have hundreds of entries, the screen will fill up and stop at four or five entries. To view additional entries, press any key except Esc. Continue paging until you find the entries you want or until you reach the end of the list. To quit the listing at any time, press Esc.

Find

The Find command provides an easy way of compiling lists of interfaces based on common criteria. The lists are useful for troubleshooting and record-keeping. You can use the find command in several different ways.

To specify up to three objects and their values for all instances, the basic syntax is:

find *mib_group object1=value1...object3=value3*

To specify up to three objects and their values starting with a specific instance, the basic syntax is:

```
Craft (1.2)> find mib_group [instance]
        object1=value1... object3=value3
```

You can include up to five additional object names after the three objects and values to refine the output. The system will return only the values for the instance and the object names for those entries that matched the specified criteria.

Initial Configuration

With the CE200 successfully started, you can now configure the CE200, associated host devices, and links (both DSL links and upstream WAN links). Initial configuration consists of setting up attributes for two-way communication over the network.

The required steps, described in the following sections, consist of:

1. Setting up records (user names and passwords) for authorized system operators
2. Storing the CE200 system name, location and contact information (optional)

3. Setting the system clock
4. Specifying the SNMP manager(s) to which traps will be sent (trap destinations)
5. Configuring the CE200 for communication over the network to enable remote access and system management
6. If your CopperEdge Concentrator is equipped with a DS3 Protection Switch/Alarm Panel, test the redundancy function.

The *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual contains a reference of all of the CE200 MIB groups and their objects, and additional information about configuring or viewing them.

Configure Operator Names

This section describes the method of configuring and managing operator names and passwords through a local database (*cmOperatorTable*) internal to the CE200. For more information about the cmOperator group, see the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

The initial tasks to be performed include:

- Create at least one unique operator name and password.
- Change the password for the factory default operator name (ce200).
- If your system will be managed locally (not through an external Radius server) and an SNMP manager will be used, create the SNMP Community names.

Note the following restrictions for entering operator usernames and passwords. These restrictions are enforced even if you are using RADIUS authentication:

- Do not use: double quotes (" or "), or ending bracket (]). The CE200 may misinterpret them as other commands.
- Pound (#) and semicolon (;) characters are not allowed.
- Do not use + or = as the last character of a Username or Password.
- Spaces are not allowed in UserName, Password or other text strings. If first and last names are needed, use upper case or an underscore to separate them: JohnDoe or john_doe.
- UserNames and Passwords are case-sensitive in their first letter only; other text strings are NOT case sensitive except for directory paths which should always be lower case.

In addition to the username and password, you will set the operator's context (authorizing system access through SNMP, nonSNMP, or All) and Privilege level (View, Monitor, or Provision). Only one operator, the Security Administrator, is allowed the

highest privilege level (Security). The default operator (ce200) is always the Security Administrator for the system.

You can use the Help command to view the list of available configuration options for each cmOperator object.

```
help cmoperator
```

Management by a Radius Server

If your system will be managed by an external Radius server, you do not need to create the SNMP Community names from CopperCraft. When Radius authentication is used, the user and password databases on all CE200s on the network, including CopperView (SNMP) sessions, will be consolidated on the Radius server, and all login requests are forwarded to the Radius server for processing.

Once one or more Radius servers are provisioned and enabled, security features are normally under their control. All operators will be created on the Radius server instead of through CopperCraft; you can still configure the *CopperCraft cmOperatorTable*, but its contents will only be used if no radius server can be contacted and the *LocalFallback* option has been enabled. Operators configured in the *cmOperatorTable* and not contained in the Radius server database will not be able to log in. The Radius protocol has fewer restrictions as to which characters are permitted in user names and passwords, but it does enforce full case sensitivity for both names and passwords.

Create a Unique Operator

Create at least one unique operator name and password with the CLI using the Craft port, Telnet, or the modem; you cannot use SNMP. We recommend you use this operator to perform the remaining configuration tasks.

To create a new operator:

1. From the control console, enter the name, password, state, context, and privilege level of the new operator.

```
CRAFT (1.2)> set cmoperator [mick] password=stones
    state=create context=nonsnmp privilege=provision
    Set Successful
```

2. Verify the new operator configuration:

```
CRAFT (1.2)> get cmoperator [mick]
    Group: cmOperator
    Name                = Mick
    Password            = *******
    State               = active
    Context             = nonSNMP
    Privilege           = Provision
```

   Passwords are not displayed. If a user forgets a password, the Security Administrator can change it to another one, or else delete the operator and start again.

3. Test the new operator by logging out of the Craft session and logging in with the new username and password.

Change the Factory Default Operator Password

To ensure continued system security, you must change the password for the factory default operator name. Otherwise, unauthorized users could gain access to your system, with full privileges to change all settings.

Passwords are not completely case sensitive; the first character of a password may be accepted regardless of case. The system is case-sensitive to all successive password characters.

To change the password:

```
CRAFT (1.2)> set cmoperator [ce200] password=b5c312r
Set Successful
```

Verify the change:

```
CRAFT (1.2)> get cmoperator [ce200]
Group: cmOperator
Instance: [ce200]
Name                = ce200
State               = active
Context             = All
Privilege           = Security
```

If your system will be managed through an external Radius server, you can skip the next section. The SNMP Community names will be created on the Radius server.

Create the SNMP Community Names

To use CopperView AMS, CopperView EM, or another SNMP-based manager, you must first configure two operators (any community names, but typically, *public* and *private*). The first operator must have the View privilege, and the second must have the Provision privilege. They are to be used in the Read Community and Write Community login prompts, respectively.

To create the Public community name:

1. From the control console, enter the name, context, state, and operator privilege level.

```
CRAFT (1.2)> set cmoperator [public] state=create
    context=snmp privilege=view
    Set Successful
```

2. Verify the configuration:

```
CRAFT (1.2)> get cmoperator [public]
Group: cmOperator
Instance: [public]
Name                = public
Password            = **********
State               = Active
Context             = SNMP
Privilege           = View
```

To create the Private community name:

1. From the control console, enter the name, context, state, and operator privilege level.

```
CRAFT (1.2)> set cmoperator [private] state=create
    context=snmp privilege=provision
Set Successful
```

2. Verify the configuration:

```
CRAFT (1.2)> get cmoperator [private]
Group: cmOperator
Instance: [private]
Name                = private
Password            = **********
State               = Active
Context             = SNMP
Privilege           = Provision
```

System Information

Although not essential for operation, you should identify this CE200 by configuring the following objects in the System group.

Name = A unique identifier for this CE200

Location = Site information (CO, city, state)

Contact = Point of contact for this CE200 (name, title, phone)

Example:

```
set system name=newname location="E St. COLLO"
       Contact="Network Administrator 800-555-4141"

Set Successful
```

- Use quotes for character strings that include spaces.
- Character strings cannot contain backslash (\), angle brackets (< and >), or apostrophe (‘) or pound sign (#) characters.

C A U T I O N

Using a pound sign (#) when configuring ports and VCs may cause the CE200 *to not read its config file correctly on reboot.*

The Craft command line prompt automatically changes to display the system name. It also displays the slot number of the active System Control Module. For example:

```
NEWNAME(1.2)>
```

In this manual, examples of the Craft command line prompt do not show the slot number.

System Clock

Check the system clock and observe the date and time displayed in the format shown (time in 24-hour format) in the example:

```
get cmsystem calendartime

CalendarTime = 2002/06/15-12:04:17
```

If necessary, set the clock to the correct date and time:

```
set cmsystem calendartime=2002/06/15-15:05:00

Set Successful
```

Verify that the clock displays the correct date and time:

```
get cmsystem calendartime

CalendarTime = 2002/06/15-15:05:09
```

Trap Destination

Configure at least one trap destination (a managing device to which SNMP traps should be sent).

```
set cmtrapdest [n.n.n.n, 162] community=trapcomm
       rowstatus=active
```

where **n.n.n.n** is the IP address of the SNMP manager.

Depending on the SNMP manager you are using, it may also be necessary to configure an IP port number or a community string for the receiver. For more information, see the description of the cmTrapDestination group in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Connecting the CE200 to the Network

To enable the CopperEdge to communicate over the network, and to allow a remote host or server to connect to the CE200 through Telnet or FTP, you must enable one of the system's network interfaces. It may be the Ethernet interface (1.2.1), or it could be one of the WAN interfaces in slots 3 or 4. The following steps outline the basic procedure.

1. Configure the network interface to be used for control and management of the system with an IP address and mask:

   ```
   CRAFT (1.2)> set cmiface [1.2.1] netmodel=ip
       ipaddr=204.16.191.254 netmask=255.255.255.0
       encapsulationtype=none
   ```

   If this interface will use a NetModel/ForwardingMode other than Full IP, specify those options as necessary.

2. If the network interface is a Frame Relay or ATM virtual circuit, you must also specify the VC number in the PII:

   ```
   CRAFT (1.2)> set cmiface [1.3.1.23]
       farendaddr=202.114.66.188 netmodel=vwan
       encapsulationtype=rfc1490
   ```

3. For the example above, an interface configured for VWAN, you would also need to configure and activate the virtual circuit as described on page 35. Note that full provisioning of a Frame Relay or ATM WAN link may require additional configuration, depending on the type of digital facility used, connected upstream equipment, etc. See the information on MIB groups related to WAN link provisioning in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual for specifics.

4. Verify the physical connection of the CE200 network interface (Ethernet, WAN, or DSL), and ensure that any third-party equipment (router, CSU/DSU, etc.) is configured to communicate with the CE200.

5. Configure the CE200's default IP route. In most cases, the routes to connected networks that are followed by outgoing message packets are automatically defined by the IP addresses and subnet masks of the interfaces. For any messages that do not have an assigned route, the default IP route (specified as 0.0.0.0) establishes the "next hop" address. This default route is mapped to its next hop through the port on the CE200 used for "upstream" packet-based communication. Typically this will be one of the WAN ports, but could be the Ethernet port, if that interface is used for network access. For example:

   ```
   CRAFT (1.2)> set iproute[0.0.0.0] nexthop=192.176.1.88
   ```

   where: `192.176.1.88` is the IP address of the router interface to which packets for unknown destinations are routed. The mask of the default route is always 0.0.0.0. If no default route is configured, packets addressed to unknown destinations are discarded.

6. Configure other entries in the IpRoute table as necessary.

7. For certain destinations, you may need to insert additional routing-table entries. In fact, any other IP Route information, beyond the default, must be configured as static routes, as there is no provision for dynamic route assignment in the CE200. If additional routes are needed, add them to the ipRoute table as described in the previous section.

8. Using the CopperCraft interface, first view the existing route table, beginning with the default (0.0.0.0) route:

   ```
   CRAFT (1.2)> get iproute [0.0.0.0]
   Group: ipRoute
   Instance: [0.0.0.0]
   Dest          = 0.0.0.0
   IfIndex       = 2
   Metric1       = 0
   Metric2       = -1
   Metric3       = -1
   Metric4       = -1
   NextHop       = 192.176.1.88
   Type          = Direct
   Protot        = netmgmt
   Age           = 64
   Mask          = 0.0.0.0
   Metric5       = -1
   Info          = 0.0
   ```

   The ipRouteTable is indexed by the destination IP address. A route entry requires at least a destination address and a Next Hop address. The Next Hop address specifies where to send a packet with the specified destination address.

   For more details on ipRoute group and its objects, including valid responses and ranges for each, see the reference listing for the ipRoute group in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

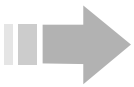
9. Save the interface configuration:

   ```
   CRAFT (1.2)> set cmsystem command=save
   ```

10. Verify that the configuration information has been saved:

```
CRAFT (1.2)> get cmsystem  configsynch
Group: cmSystem
ConfigSynch          =  Saved
```

11. The CE200 responds with the cmSystem configuration. The CommandStatus object should display `Succeeded`. If Command Status shows `InProgress`, repeat the **`get cmsystem`** command; if Command Status shows `Failed`, contact Technical Application Support.

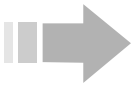


N O T E

If this CE200 is equipped for Control and WAN Redundancy (duplicate control and WAN modules installed in slots 2 to 5 and 14 to 17 of the CE200), you will need to repeat the previous initialization steps for the Redundancy Complex. If you are not already well acquainted with the redundancy functionality of the CE200, then we recommend that you first read about Control and WAN Redundancy in Chapter 8.

DS3 Protection Switch

For CE200s equipped with a DS3 Protection Switch/Alarm Panel, check that the redundant complex (the System Control, Buffer, and DS3 WAN modules on the Backup side) is set up, and test the functionality as described in the following section.



N O T E

The terms Primary *and* Secondary *are different from the terms* Preferred *and* Backup. *Preferred refers to modules located in slots 2 to 5 on the left side of the CE200 chassis, and Backup refers to the redundant modules located in slots 14 to 17 on the right side of the CE200 chassis (also called the* Redundant Complex*). Primary refers to the side that is controlling operation and traffic, and Secondary refers to the side that will be active in case of a failover.*

When the DS3 Protection Switch detects the failure of a Primary DS3 WAN module:

- The Backup side takes over the Primary role within approximately five seconds.
- The Preferred-side SCM assumes the Secondary role. You can no longer configure the CE200 through its Craft port.
- The circuits configured across the original DS3 are automatically switched over to the backup DS3 and will continue to operate normally. No additional circuit configuration is required.

  Although the circuits have been switched over to the backup modules, there is no indication of these circuits in the circuit tables of the CE200. They are not available for configuration on the Backup side, since configuration should only be performed on the Preferred side of the CE200.

- Redundancy is disabled every time a switchover occurs. This prevents the Preferred side from assuming the Primary role before any problems can be fixed.
- If redundancy is not enabled, the WAN modules on the Backup side show Operstate=Disabled, and the redundant System Control Module (SCM) shows Role=Secondary.
- When redundancy is enabled, all WAN modules in the CE200 show Operstate=Enabled.

Configure for Redundancy

Check that redundancy is enabled for the CE200:

```
get cmsystem

Group: cmSystem
ObjectClass          =  System
OperState            =  Enabled
Version              =  E 4.0
Master               =  0.0.0.0
ConfigFileName       =  config.tgz
CalendarTime         =  2002/10/20-09:27:05
MyPII                =  1.2.0.0
PrimaryPII           =  1.2.0.0
SecondaryPII         =  1.15.0.0
MgmtPII              =  0.0.0.0
Redundancy           =  Enabled
ShelfCount           =  1
ExpIpSubNet          =  192.168.250.0
ConfigSynch          =  NotSaved
Command              =  None
CommandStatus        =  None
```

Verify that the SCM on the Preferred side (instance 1.2) shows Role=Primary, and that the SCM on the Backup side (instance 1.15) shows Role=Secondary. All WAN modules on both sides should show Operstate=Enabled.

```
geta cmboard

Index               ObjectClass         OperState     HwType
HwVersion           SwVersion           PromVersion   Role
ClusterRole         UpTime              NumPorts      FileName
FileDate            ConfigChange        Command       SerialNumber
Information

Instance: [1.2.0.0]
1.2.0.0            SystemControlModule Enabled           SystemControlModule
R 1.0               4.0.72              4.0.59        Primary
Master              0 day 0 hour 4 min  5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.3.0.0]
1.3.0.0             DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3               4.0.72              2.5.11        Active
NotApplic           0 day 0 hour 3 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None          MI4FT03SX_003470105
""
```

```
Instance: [1.4.0.0]
1.4.0.0             DS3FR-WAN           Enabled       DS3FR-WAN
R 1.0               4.0.72              1.35.19       Active
NotApplic           0 day 0 hour 3 min  1             ds3fr.bin
Oct 13 2002, 00:25: 0                   None          MI3SM01GL
""
etc.

Instance: [1.15.0.0]
1.15.0.0            SystemControlModule Enabled       SystemControlModule
R 1.0               4.0.72              4.0.59        Secondary
Master              0 day 0 hour 10 min 5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.16.0.0]
1.16.0.0            DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3               4.0.72              2.5.11        Active
NotApplic           0 day 0 hour 9 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None          MI4FT03SV_003470105
""
```

Enable Redundancy

If your CE200 is not set up as described above, follow these steps:

1. On the Preferred side of the CE200, enable redundancy for the system:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   ```

2. Save the configuration:

   ```
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes because the Backup side will restart.

3. Check the redundancy settings of the System Control Modules on both sides of the CE200.

   a) Make sure redundancy is enabled.

   ```
   CRAFT (1.2)> get cmsystem
   ```

   The Redundancy object should return Enabled.

   a) Check the roles of the System Control Modules on both sides of the CE200:

   ```
   CRAFT (1.2)> geta cmboard
   ```

   The Preferred SCM should show Role=Primary, and the Backup SCM should show Role=Secondary.

Test the Functionality

1. Create a failover by disconnecting a cable from a WAN module on the Preferred side.

2. Wait for two minutes for failure information to be written to the database.

3. Verify that the Preferred side shows Role=Secondary:

```
geta cmboard
Index               ObjectClass         OperState     HwType
HwVersion           SwVersion           PromVersion   Role
ClusterRole         UpTime              NumPorts      FileName
FileDate            ConfigChange        Command       SerialNumber
Information

Instance: [1.2.0.0]
1.2.0.0             SystemControlModule Enabled       SystemControlModule
R 1.0               4.0.72              4.0.59        Secondary
Master              0 day 0 hour 4 min  5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

No more instances
```

Because this SCM is no longer in control, you cannot see any other modules on this side of the CE200.

4. Log out of the CE200, and log in to the Backup side of the CE200, which is now controlling operation and traffic.

5. Check the Role and OperState settings of the SCM and WAN modules:

   ```
   CRAFT (1.2)> geta cmboard
   ```

   a) SCM [1.2] shows Role=Secondary.

   b) The WAN modules on the Preferred side show Operstate=Disabled.

   c) SCM [1.15] shows Role=Primary.

   d) The WAN modules on the Backup side show Operstate=Enabled.

6. Reconnect the cable to the WAN module on the Preferred side.

7. Enable redundancy for the system and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save will take a few minutes until the restart is complete.

8. Tell the Backup SCM to relinquish control:

   ```
   CRAFT (1.2)> set cmsys command=relinquish
   ```

9. Log in to the Preferred side of the CE200.

10. Verify that the Preferred SCM shows Role=Primary and the Backup SCM shows Role=Secondary.

    ```
    CRAFT (1.2)> geta cmboard
    ```

    Redundancy is automatically disabled, and the WAN modules on the backup side show Operstate=Disabled.

11. Enable redundancy on the CE200 and save the configuration:

    ```
    CRAFT (1.2)> set cmsystem redundancy=enabled
    CRAFT (1.2)> set cmsystem command=save
    ```

    The save process will take a few minutes while the Backup side restarts. The WAN modules on the backup side will show Operstate=Enabled.

Chapter 4
Advanced Configuration

This chapter describes procedures for establishing DSL and network interfaces, and performing other configuration tasks. Important tasks include:

- Establishing subscriber DSL links (that is, configuring the Interface Table entries for each of the CE200's connected DSL ports).
- Configuring the upstream (WAN side) links, and setting up any additional parameters needed for your specific operation.

The term *DSL module* is generally used throughout this chapter to mean *any supported* DSL module: SDSL, IDSL, ADSL (G.lite or G.dmt), G.SHDSL, or T1. If there is a difference in the way one or the other types of DSL modules behave or what they require, it will be explicitly specified.

The Varieties of DSL

Although there are many technical differences among the varieties of DSL that your system may offer, the most visible variation is that each type of DSL module supports a separate set of data rates. This profusion of DSL types allows the Network Services Provider to serve the largest possible universe of potential customers. Most of the DSL modules offer a range of configurable data rates, and the operator selects as the subscriber's maximum data rate, the one that will provide the fastest consistent with the distance from the CO and the loop quality. For ADSL services, both the upstream and downstream rates must be set. Use the following MIB object groups to set the maximum data rate.

- For standard SDSL: *cmHdslModemTable*
- For IDSL: *cmIdslModemTable*
- For DS1 Subscriber Module: Max rate fixed at DS1 rate (1.544 Mbps)
- For G.SHDSL: *cmGShdslModemTable*
- For ADSL (G.lite or G.dmt): *cmADSLModemTable*

Beyond the basic rate settings, symmetrical DSL interfaces (SDSL, G.SHDSL and DS1) can also be configured to limit their speeds on

a per-VC basis for systems operating tiered service levels. For interfaces using the IP networking model or any of the aggregated netmodels (VWAN, CopperVPN, or legacy HDIA) you can configure rate limiting via the *cmFrCircuitTable* MIB group. It is possible to rate-limit this way for VCs using the Cross-connect netmodel, as well, but for Cross-connect VC's, you can achieve the desired end more efficiently using ATM or FR QoS (e.g. *cmCircuitParam* or *frCircuit*).

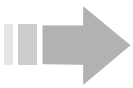
Configuring DSL Links for Network Models

Configuration of DSL links (SDSL, IDSL, ADSL, G.SHDSL, and T1 line modules) on a CE200 focuses on networking models (netmodels) and encapsulations. You must choose a netmodel for each DSL (and WAN) link. You can configure a CE200 with the following netmodels, depending on what your applications are:

- IP Routing
- Virtual Wide Area Network (VWAN)
- CopperVPN
- Cross-Connect

The IP netmodel can be Full-IP or Policy-based. If Policy-based, it can run over a WAN or Ethernet link. The VWAN netmodel can be set in point-to-point or bridge (aggregate) mode over either a WAN or Ethernet link. Similarly, CopperVPN can be set in point-to-point or aggregate mode, but, unlike VWAN, it can specify the IP address of the far-end router or learn the router's IP address with inverse ARP.

In contrast to the aggregation netmodels, *VWAN* and *CopperVPN*, the *Cross-Connect* netmodel only provides point-to-point connections, but it offers many encapsulation types and forwarding modes. Finally, the CopperVPN model can serve voice streams and data streams on the same port when they are separated into different VCs. For more details on the netmodels and their encapsulation types, see the cmIfaceTable in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

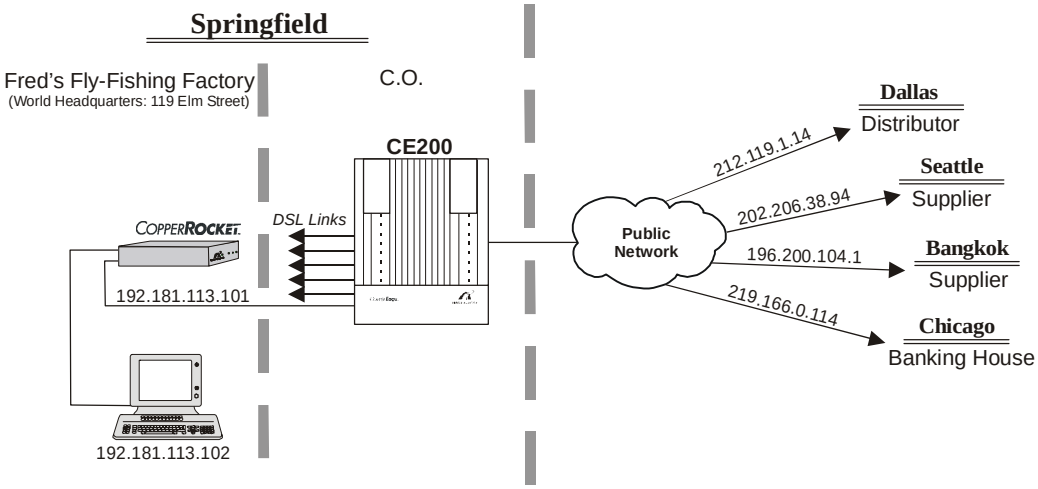


N O T E

Configuring DSL interfaces for Voice-over-SDSL or for Multilink operation entails special requirements and precautions. Please read the pertinent discussions in Chapters 4 and 6.

IP Netmodel

With this netmodel, the CE200 will route both inbound and outbound packets according to the IP address of their origination and destination devices. In the full IP netmodel (and also the IP policy variation), subscriber links appear as wide-area extensions of a remote LAN.



IP Network Model

Full IP Routing

The IP netmodel with full IP forwarding is one that most users know well. The NSP Central Office serves as a gateway to the net, and individual message packets are exchanged and routed over the public network based on their destination IP address. The diagram above describes the model.

N O T E

Because it is simple, easy to configure, and well-known, the full IP netmodel is often used in configuring temporary links or dedicated management interfaces. However, to maximize network security, economy of address-assignment, and efficient use of bandwidth, other net models, such as the new CopperVPN should be given first consideration.

To configure an interface for full IP:

1. Set the IP address, and other required parameters for each connected CPE. Use the PII to designate the interface to be configured.

```
CRAFT (1.2)> set cmiface [1.10.1] netmodel=ip
       ipaddr=192.181.113.101 netmask=255.255.255.0
       encapsulationtype=rfc1483
Set Successful

CRAFT (1.2)> get cmiface [1.10.1]
Group: cmIfaceTable
Instance: [1.2.1.0]
PII                 =  1.2.1.0
IfIndex             =  1.2.1.0
Name                =  ""
GroupName           =  ""
AdditionalInfo      =  ""
NetModel            =  IP
```

```
IpAddr              =  192.181.113.101
NetMask             =  255.255.255.0
MacAddr             =  0.80.50.1.b3.ca
BurnedInMacAddr     =  0.80.50.1.b3.ca
FarEndAddr          =  0.0.0.0
DestPII             =  0.0.0.0
CMCPCompatible      =  No
EncapsulationType   =  rfc1483
FwdMode             =  Full-IP
Pix                 =  2
ServiceClass        =  None
```

1. The CE200 retrieves the MAC address from the CPE automatically each time a DSL link is established.
2. Remember also that *all* CE200 ports—Ethernet, WAN, or DSL—are referenced with a PII. For example, PII 1.3.2 refers to the lower connector (Port 2) on a V.35 WAN module.
3. Hosts connected to the CPE LAN must be configured with IP addresses on the subnet defined by the DSL port's IP address and netmask.
4. When you have completed the Initial Configuration of the CE200 and its interfaces, save the configuration data to a file:

   ```
   CRAFT (1.2)> set cmsystem command=save
   ```

5. The file is saved on the appropriate volume (P: or Q:) as: \ce200\system\config.tgz (or config.txt). For more information, see "cmSystem, Managing Your Configuration" on page 77.
6. Use the ping command to verify connectivity.

   ```
   CRAFT (1.2)> ping 192.181.113.102
   ```

7. Log out of the Command Line Interface by entering **`exit`**, **`quit`**, or **`logout`** at the prompt.
8. If you fail to log out, the system will perform an auto logout after approximately 15 minutes (same as Telnet).

Policy-based IP Routing

With this variation on the IP netmodel, the CE200 is able to route inbound packets according to the IP address of the origination device and outbound packets, regardless of IP address, to a designated interface on the CE200. Two types of interfaces exist:

- In IP Policy over a WAN interface, packets outbound from the CPE are routed to a destination WAN-port interface (usually an ATM VC or Frame Relay PVC) on the CE200.
- In IP Policy Over Ethernet, packets outbound from the CPE are routed to a specific LAN host device through the CE200 Ethernet port.

To configure interfaces for Policy-based IP routes and to establish WAN virtual circuits for IP-Policy, follow the procedures in the "IP Policy Over WAN" on page 85.

VWAN Netmodel

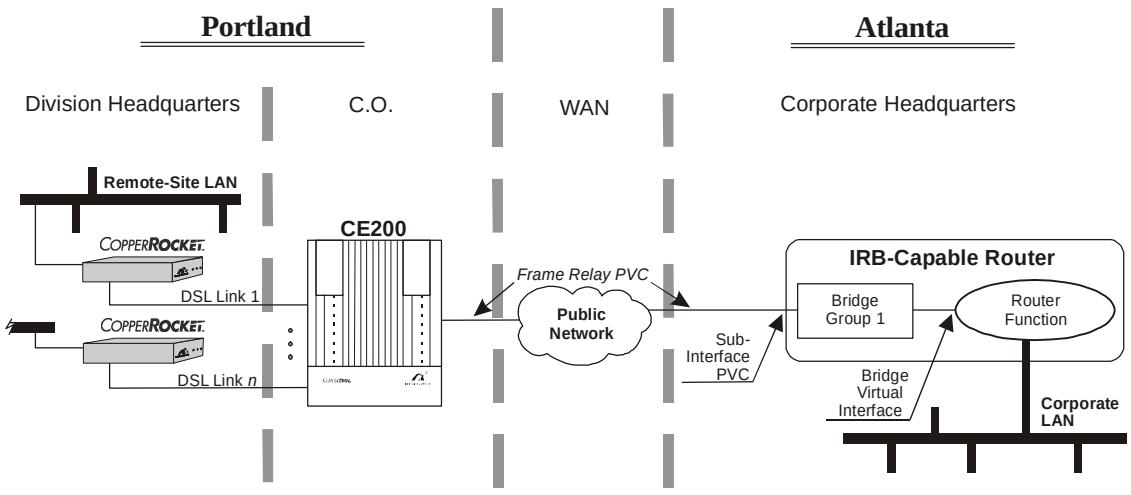
With this netmodel, subscriber links act as wide-area extensions of a remote LAN or other private network. Packets can be exchanged over an ATM or Frame Relay virtual circuit or, as MAC PDUs, over the CE200's Ethernet port. VWAN is a layer 2 LAN extension to an upstream router.

The VWAN netmodel requires an upstream IRB router, allowing the CE200 and the router together to function as a virtual bridge. Hosts on the LANs on the downstream side of the CPEs become remote or *wide-area* extensions of a private LAN on the upstream side of the router.

Enabling VWAN in the CE200 depends on a specific operating mode of certain routers called Integrated Router/Bridge or IRB. Routers without this capability may not be suitable for use in VWAN applications.

The VWAN netmodel allows you to link a single DSL line to a single virtual bridge (WAN VC or Ethernet) in point-to-point mode, or aggregate many DSL links into a single virtual bridge, provided they all have the same DestPII; that is, provided they are served by a common circuit (WAN VC or Ethernet) on a common WAN (or Ethernet) interface linked to a common bridge group on a distant IRB router.

Each CE200 can support as many virtual bridge groups as it can virtual circuits, each connecting to a different far-end terminal point. Each bridge group, in such a case, must be associated with one DLCI or ATM Virtual Link number (or with the Ethernet port).



VWAN Network Model

When many DSL links comprise a virtual bridge group, the cmIface FwdMode object is VWAN BRIDGED. When only a single DSL link comprises a bridge group, the FwdMode object in the

cmIface group is VWAN POINT-TO-POINT. When the Ethernet port is configured as the destination WAN interface, the FwdMode object is always VWAN BRIDGED, even for a single member group.

Configuration of VWAN links must be coordinated with the telecom/datacom management at the far end of the upstream link. An IRB-capable router with a compatible operating system (IOS) version must be installed at the far end.

The following procedure provides a model for end-to-end configuration of an SDSL subscriber for VWAN, using a Frame Relay WAN link. Configuration of the network port for VWAN over Ethernet appears in Chapter 5.

1. **Configure the DSL interface for VWAN** and specify the outbound facility to use.

2. Remember that, for a Frame Relay or ATM WAN interface, the DESTPII object includes the number (Frame Relay DLCI or ATM Virtual Link) that identifies the associated virtual circuit. For example:

```
CRAFT (1.2)> set cmiface [1.10.1] netmodel=vwan
       encapsulationtype=rfc1483 destpii=1.3.1.27
Set Successful

CRAFT (1.2)> get cmiface [1.10.1]
Group: cmIfaceTable
Instance: [1.10.1.0]
PII                  = 1.10.1.0
IfIndex              = 1.10.1.0
Name                 = Zebra
GroupName            = X-ray
AdditionalInfo       = Comp account
NetModel             = VWAN
IpAddr               = 0.0.0.0
NetMask              = 0.0.0.0
MacAddr              = 0.60.58.1.0.36
BurnedInMacAddr      = 0.77.62.18.63.bc
FarEndAddr           = 0.0.0.0
DestPii              = 1.3.1.27
CMCPCompatible       = Yes
EncapsulationType    = rfc1483
FwdMode              = VWAN Point-to-Point
Pix                  = 72
Service Class        = D
```

   The MAC address listed for the DSL port in VWAN mode is actually the address of the connected router at the far end of the WAN link. The CE200 still displays the Mac Address of the CPE, however, as the BurnedInMacAddr object.

3. Enable the WAN interface for Frame Relay.

```
CRAFT (1.2)> set cmfrdlcmi [1.3.1] adminstate=enabled
Set Successful
```

   The\example here is a simplified one that does not specify LMI. To provide a link with LMI, use the State object in the standard frDlcmi group to specify the LMI formatting scheme.

For more information on Frame Relay configuration, including LMI, see Chapter 6.

4. Activate the virtual circuit.

```
CRAFT (1.2)> set frcircuit [1.3.1.27] state=active
Set Successful
```

5. Repeat step 1 for each port in the Virtual Bridge group.

6. Optional: Configure the Virtual Bridge to block broadcast packets (such as IPX) to the wide area network.

```
CRAFT (1.2)> set cmvbridge [1.3.1.27] option=ip-special
Set Successful
```

   For more about cmVbridge, see the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

7. When you have completed the initial configuration of the CE200 and its interfaces, save the configuration data:

```
CRAFT (1.2)> set cmsystem command = save
```

   The file is saved on the appropriate volume (P: or Q:) as: \ce200\system\config.tgz (or config.txt). For more information, see "cmSystem, Managing Your Configuration" on page 77.

8. At the far end of the VWAN link (the main LAN site) configure the IRB-capable router to provide a bridge virtual interface.

   As an example, here is a typical config.txt file for a Cisco Systems IRB-capable router showing the relevant entries for VWAN (in this example, the DLCI number is 30):

```
!
interface Ethernet0
ip address 192.1.1.188 255.255.255.0
!
interface Serial0
 no ip address
 encapsulation frame-relay IETF
 no keepalive
 ignore-dcd
!
interface Serial0.1 point-to-point
 frame-relay interface-dlci 30
 bridge-group 1
!
interface Serial1
 no ip address
shutdown
!
interface BVI1
mac-address 0000.0c00.338c
ip address 206.41.72.1 255.255.255.0
no keepalive
!
bridge irb
 bridge 1 protocol ieee
 bridge 1 route ip
!
```

For additional information, consult the documentation supplied by the manufacturer.

9. Log out of the Command Line Interface by entering **`exit`**, **`quit`**, or **`logout`** at the prompt.

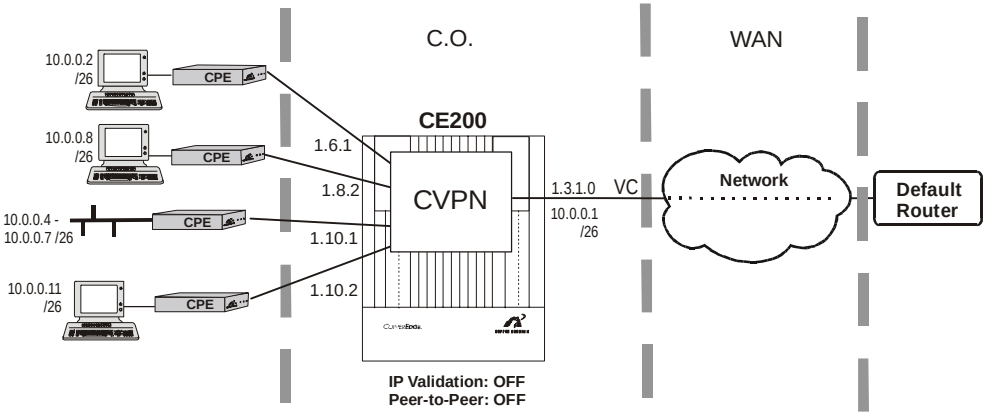
   If you fail to log out, the system will perform an auto logout after approximately 15 minutes (same as Telnet).

CopperVPN Netmodel

Beginning with Version 7.0, *CopperVPN* has been completely retooled into a more powerful, much more secure, yet much easier-to-use networking model which provides the functional equivalent of a Transparent Virtual Router. CopperVPN (sometimes referred to as *CopperVPN+*) combines and expands the capabilities of the previous CopperVPN and HDIA networking schemes into a powerful new, layer-three model. This new and enhanced version of CopperVPN eases subscriber configuration, increases security, and greatly improves the flexibility and efficiency of IP-Address utilization.

Operating with CopperVPN requires that you configure groups of interfaces, with each group aggregated onto a common WAN VC for transport. These groups, also referred to as *transparent virtual routers,* perform the same basic functions as an IP router, but are invisible to the external network. The following two diagrams provide examples of traffic flow in two variations of the CVPN netModel.

Example 1: DSL Aggregation



PII	IP Addr	MAC Addr	Type	TTL
1.6.1	10.0.0.2	00-50-82-75-8a-20	dynamic-ARP	30
1.10.1	10.0.0.5	00-70-28-00-8b-10	dynamic-ARP	10
1.10.1	10.0.0.7	00-50-28-00-8a-20	dynamic-ARP	5

CopperVPN, Simple DSL Aggregation

In this example, CVPN is configured as a simple DSL aggregator (equivalent to the former CVPN netmodel). Both IP validation and Peer to Peer are disabled. The Address Translation/Forward table

is empty at the system start, and all of the CPE hosts are manually configured with their IP address and gateway IP address.

Upstream Direction:

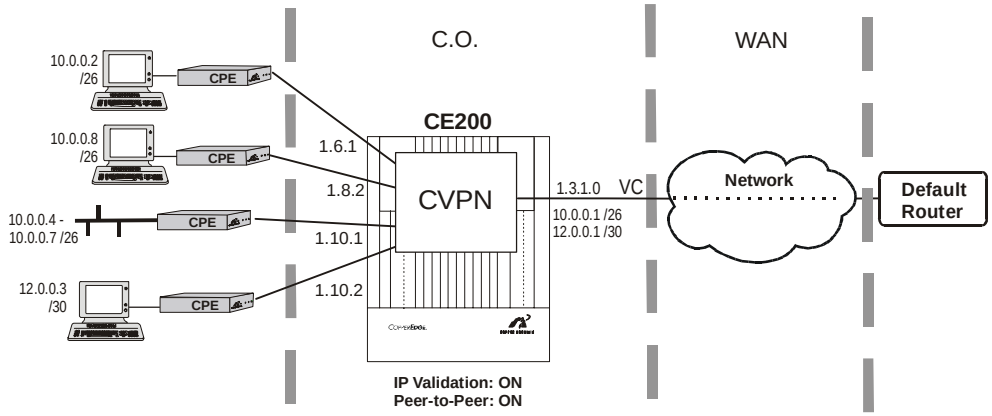
1. The Host (10.0.0.7) sends an ARP request for its gateway (10.0.0.1)
2. The CE matches the gateway in its WAN VC (1.3.1.100) gateway/subnet table and proxy ARP replies to the request. The CE also fills up a route in the Address Translation/Forward (ATF) table for that host.
3. The Host sends the subsequent packets to DSL port 1.10.1.
4. The CE receives the packets and forwards them to the upstream WAN gateway.

Downstream Direction:

1. The gateway (10.0.0.1) sends back the response packets to host 10.0.0.7
2. The CE receives the packets and finds the DSL port by looking up the ATF table and sends the packets down to the host.
3. The gateway (10.0.0.1) sends a packet to host (10.0.0.5).
4. The CE receives the packet and could not find the matching route in the ATF table. It holds the packet and sends an ARP request to every DSL port in the group. If CE receives another packet before it receives the ARP reply, it will discard the holding packet and always only hold the last received packet.
5. The CE receives the ARP reply from one of the DSL ports and adds a route to the ATF table for that host and forwards the held packet down to the DSL port.

Similarly the route for host 10.0.0.2 is added to the ATF table. With the previous version of CopperVPN, this ATF table was not accessible by operator. With today's CVPN, you can read and write to it, allowing you to add permanent routes for some hosts as needed. Also if you use DHCP to assign the IP addresses, the routes will be added for those hosts automatically via snooping DHCP messages.

Example 2: HDIA Functionality



PII	IP Addr	MAC Addr	Type	TTL
1.6.1	10.0.0.2	00-50-82-75-8a-20	static	
1.8.2	10.0.0.8	00-50-82-75-8a-30	static	
1.10.1	10.0.0.5	00-70-28-00-8b-10	dynamic-ARP	10
1.10.1	10.0.0.7	00-50-28-00-8a-20	dynamic-ARP	5

CopperVPN, HDIA Functionality

In this example, CVPN is configured to provide an equivalent to the former HDIA netmodel, with both IP validation and Peer-to-Peer enabled. CopperVPN provides two ways to fill up the routes in the Address Translation/Forward (ATF) table. You can either enter them manually and ensure that the CPE hosts are also manually configured with their IP address and gateway as in the former HDIA method, or you can allow the hosts to get their IP address and gateway through DHCP. In the latter case, the CopperEdge automatically adds the routes in the ATF table via snooping of DHCP messages (as with the routes for hosts in DSL port 1.10.1 in the example). With IP validation enabled, the CE will simply discard any packets that fail their source IP validation, together with any packets that did not find their destination IP in the ATF table (as in the example, with all packets from/to host 12.0.0.3).

Peer to Peer Communication:

1. The Host (10.0.0.7) sends an ARP request for its neighbor host (10.0.0.8).
2. The CE validates the source IP of the ARP request coming from DSL port 1.10.1 by looking it up in the ATF table. It then verifies that the destination IP matches an IP address in the ATF table. If there is a match, it sends a proxy ARP reply to the request.
3. The Host (10.0.0.7) sends the subsequent packets to DSL port 1.10.1.
4. The CE receives the packets. Before the CE forwards them to DSL port 1.8.2, it sends an ARP request to DSL port 1.8.2 to query for the host's Mac address. After receiving the ARP reply, the CE forwards subsequent packets to port 1.8.2.

Also the CE does source IP validation for all packets it receives from DSL ports.

When the CE receives downstream packets from the WAN VC, it follows the same procedure as described in the step 4. For IP-1490 encapsulation, the ARP functions in steps 1, 2 and 4 are skipped.

Configuring CopperVPN

To configure a CopperVPN group (the "+" is a marketing term and is not used in actual operation (in this document, the terms *CopperVPN* and *CopperVPN+* are used interchangeably), you will perform the following basic steps in sequence:

1. **Create the Group:** Create a *cmIfaceTable* entry for a WAN VC with *NetModel=CopperVPN*. This will automatically identify it as the PII of a CopperVPN group.

   Use *cmVPNGroupTable* to establish any other settings applicable to the group (peer-to-peer, IpValidation, etc.).

2. **Add Subnets:** If using static addressing, assign the IP address and netmask for the IP gateway for the group. You can also assign a *diagnosticIpAddress* in *cmVpnSubnetTable*. Up to eight subnets can be associated with a CVPN group.

3. **Add the Subscriber Group Members:** Create the entries in the *cmIfaceTable* for the DSL interfaces that you wish to assign as members of the group, specifying DestPii=VPNGroupPii (the PII of the interface configured in step 1).

   When configuring a subscriber interface for CVPN+, you do not configure its IpAddr, NetMask or FarEndAddr objects.

4. **Add Static Routes:** As needed, use *cmVPNRoute* to associate group destination addresses with the appropriate interfaces.

5. **Configure Filters:** As needed, use *cmFilter* and *cmGroupFilterTable* to create and apply policy filters to individual interfaces and groups.

6. **Configure DHCP:** Configure the *cmDHCPTable*, one entry per subscriber interface. For CopperVPN, set the *Function* object to *DHCPForward*.

For additional information on configuring CVPN groups, see the CopperEdge Command Reference and MIB Definitions document. Relevant groups include: cmIfaceTable, cmVPNGroupTable, cmVPNRouteTable, cmVPNSubnetTable, and cmGroupFilter.

The following examples show a typical configuration for CopperVPN.

1. Set the WAN VC network model:

```
Craft> set cmif[1.3.1.101] netm=CopperV Encap=ip-1483
Set Successful

Craft> get cmif[1.3.1.101]
Group: cmIfaceTable
Instance: [1.3.1.101]
PII                 =  1.3.1.101
IfIndex             =  1.3.1.101
Name                =  ""
GroupName           =  ""
AdditionalInfo      =  ""
NetModel            =  CopperVPN
IpAddr              =  0.0.0.0
NetMask             =  0.0.0.0
MacAddr             =  ff.ff.ff.ff.ff.ff
BurnedInMacAddr     =  ff.ff.ff.ff.ff.ff
FarEndAddr          =  0.0.0.0
DestPII             =  0.0.0.0
CMCPCompatible      =  No
EncapsulationType   =  ip-1483
FwdMode             =  CopperVPN
Pix                 =  274
ServiceClass        =  None
```

2. Set the DSL network model:

```
Craft> set cmif[1.6.1] netmodel=coppervpn
   encap=mac-1483 dest=1.3.1.101
Set Successful

Craft> get cmif[1.6.1]
Group: cmIfaceTable
Instance: [1.6.1.0]
PII                 =  1.6.1.0
IfIndex             =  1.6.1.0
Name                =  ""
GroupName           =  ""
AdditionalInfo      =  ""
NetModel            =  CopperVPN
IpAddr              =  0.0.0.0
NetMask             =  0.0.0.0
MacAddr             =  ff.ff.ff.ff.ff.ff
BurnedInMacAddr     =  0.0.0.0.0.0
FarEndAddr          =  0.0.0.0
DestPII             =  1.3.1.101
CMCPCompatible      =  Yes
EncapsulationType   =  mac-1483
FwdMode             =  CopperVPN
Pix                 =  3
ServiceClass        =  D
```

3. Set group attributes:

```
Craft> set cmVPNGroup[1.3.1.101] PeerToPeer=enable
   IPValidation=enable
Set Successful

Craft> get cmVNPGroup[1.3.1.101]
Group: cmVNPGroupTable
Instance: [1.3.1.101]
VPNGroupPii         =  1.3.1.101
Name                =  PTCC
DefaultUplink       =  1.3.1.101
WANUplink1          =  0
WANUplink2          =  0
WANUplink3          =  0
PeerToPeer          =  enabled
IpValidation        =  enabled
GatewayByInARP      =  enabled
DHCPSnooping        =  disabled
WanArpPeerHosts     =  enabled
DefaultTTL          =  30
Command             =  none
PolicyBlocked       =  9
UnresolvedAddr      =  0
InvalidPackets      =  0
InternalLimits      =  0
DIPFailures         =  0
```

4. Set the gateway/subnet:

```
Craft> set cmVPNSubnet[1.3.1.101, 1] RowStat=active
    gatewayIpAddr=10.24.23.1 netmask=255.255.255.0
Set Successful

Craft> get cmVPNSubnet[1.3.1.101, 1]
Group: cmVPNSubnetTable
Instance: [1.3.1.101, 1]
VPNGroupId           =  1.3.1.101
Number               =  1
RowStatus            =  active
GatewayIpAddr        =  10.24.23.1
NetMask              =  255.255.255.0
DiagnosticIpAddr     =  10.14.22.2
Type                 =  Static
TimeToLive           =  0
```

5. Set IP entries:

```
Craft> set cmVPNRoute[1.3.1.101, 10.24.23.2]
    RowStatus=active Interface=1.6.1
Set Successful

Craft> get cmVPNRoute[1.3.1.101, 0]
Group: cmVpnRouteTable
Instance: [1.3.1.101, 10.24.23.2]
VPNGroupPii         =  1.3.1.101
Destination         =  10.24.23.2
RowStatus           =  active
Interface           =  1.6.1
MacAddr             =  2.30.3E.2.0.D
Type                =  Static
TimeToLive          =  0
Command             =  none
DefaultGateway      =  10.24.23.1
SubnetMask          =  255.255.255.0
```

Cross-Connect Netmodel

The Cross-Connect netmodel allows conversion of disparate frame formats between DSL and WAN interfaces. Under this netmodel, different forwarding modes support multiple encapsulation types as well as translation between different encapsulation types. For example, under the VC-VC Payload forwarding mode, a frame encapsulation on the DSL interface can be converted into an ATM encapsulation on the WAN interface.

In addition to cross-connecting DSL connections to WAN interfaces, the Cross-Connect netmodel is used in connecting two ATM WAN interfaces on the same CE200, one of which functions as an inter-shelf trunk (IST) and connects to a subtended DSLAM, and the other connects to the network. To use this configuration, the two interfaces must be on separate modules.

There are two types of Cross-Connect forwarding modes, determined by the DSL port encapsulation: *per-VC* forwarding modes and *per-port* forwarding modes. P*er-VC* forwarding modes result from a DSL encapsulation that allows multiple VCs per DSL port. P*er-port* forwarding modes result from a DSL encapsulation that does not support VCs.

Per-VC Forwarding Modes

All Cross-Connect per-VC encapsulations support up to 64 PVCs on the DSL port multiplexed to a single WAN VC. Each DSL VC inherits its encapsulation from the DSL port. DLCI numbers identifying DSL VCs obey the same rules as they do on WAN VCs: numbers must be between 17 and 991, and they do not have to be sequential. However, the numbers 16, 528, and 529 cannot be used.

In general, the DSL VC's forwarding mode is determined by the WAN VC. If the WAN encapsulation is not specified (None), packet forwarding is performed in VC-VC Payload forwarding mode. If the DSL and WAN encapsulations are the same, packet forwarding is performed in a *transparent* mode. If the DSL and WAN encapsulations differ, packet forwarding is performed in a *translation* mode.

Since the cmSubIface table has no FwdMode object, the Cross-Connect forwarding mode is found in the cmIfaceTable entry for the WAN VC. The forwarding mode for the DSL port is per-VC, since each DSL VC has its own FwdMode.

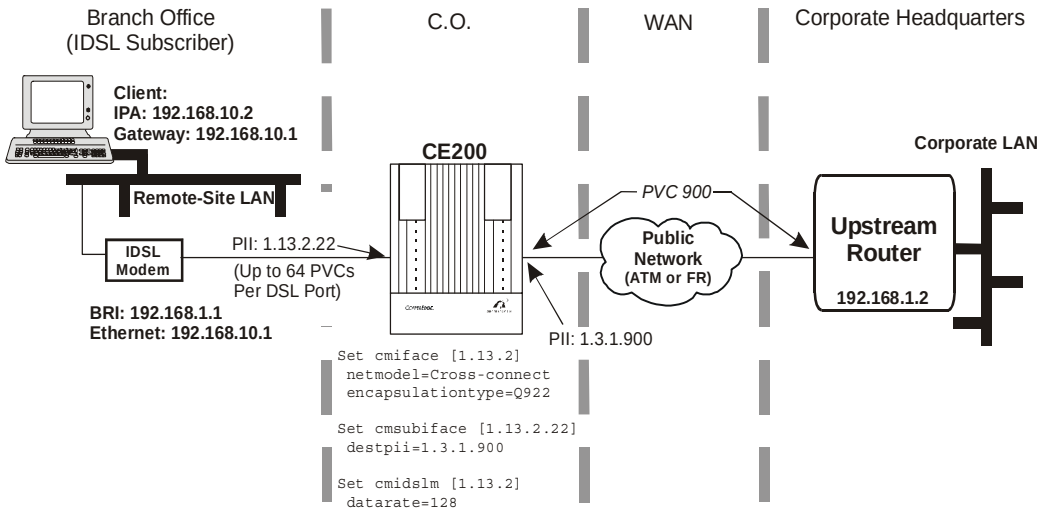
LMI is available on DSL links with per-VC forwarding modes. When LMI is in force, the CE200 acts as a DCE, and the CPE acts as a DTE. For more information, see the discussion on the cmIface and cmSubIface object groups in the CopperEdge 200 CopperCraft Reference and MIB Definitions manual. Also see the discussion of encapsulation in Appendix A.

DSL Encapsulation	Line Card Type	WAN Encapsulation	Forwarding Mode
Q922-1490, Q922	Frame	None	VC-VC-payload (transparent)
Q922-1490, Q922	Frame	FRF5	FRF5 (translation)
Q922-1490	Frame	RFC-1483	FRF8-1490-1483 (translation)
RFC-1973	Frame	RFC-1973	PPP-Transparent
RFC-1973	Frame	RFC-2364 null	PPP-Translation
RFC-1973	Frame	RFC-2364 LLC	PPP-Translation
RFC-1973	Frame	None	VC-VC-Payload

VC-VC Payload Forwarding Mode

The VC-VC payload forwarding mode allows third-party CPEs to use encapsulation formats other than RFC-1483 so they can function as terminals on a remote LAN. In such a configuration, as the following graphic shows, the WAN interface and backbone can be either Frame Relay or ATM. DSL VCs configured for VC-VC payload using the Cross-Connect netmodel generally use the Q922-1490 encapsulation type. This setting, used *only* on DSL VCs, specifies:

- The DSL physical port encapsulation is Q.922, thus allowing Frame-Relay or FUNI VCs on it.
- The encapsulation on every VC using that port is RFC-1490.



VC-VC Payload Forwarding Model

FRF.5 Forwarding Mode

This translation mode allows a DSL port on the CE200 to receive Q.922 HDLC frames, transform them into ATM AAL5 PDUs, and send them to an ATM WAN interface for transmission across an

ATM network. At the other end of the network, a reverse translation allows the Q.922 HDLC frames to be recovered.

To configure this mode, set the netmodel and encapsulation type on the DSL and WAN interfaces:

```
set cmiface [1.3.1.100] netmodel=cross-connect
       encapsulationtype=frf5

set cmiface [1.6.1] netmodel=cross-connect
       encapsulationtype=q922

set cmsubiface [1.6.1.15] destpii=1.3.1.100
```

FRF.8 Translation and Transparent Forwarding Modes

This translation/forwarding mode is for Q.922 frames being sent to ATM VCs on an ATM network where they terminate. The CE200 decides to use translation mode or transparent mode based on the encapsulations set on the DSL VC and WAN VC. (In the Cross-Connect netmodel, DSL VC encapsulation is implied by the DSL port encapsulation.)

For instance, if the two encapsulations are the same (or if the WAN VC encapsulation is None), transparent mode is used. If they differ, the Translation mode is used. Since DSL encapsulation applies to an entire DSL port, all DSL VCs that require translation must use the same encapsulation. DSL encapsulation is not relevant or used on DSL VCs that are not translated (i.e., whose cross-connected WAN VC does not specify a translation).

To configure two data VCs on one DSL port, and map them to two WAN VCs where one has an encapsulation type of RFC-1483 and the other has an encapsulation type of None, enter the following commands:

```
set cmiface [1.3.1.1] netmodel=cross-connect
       encapsulationtype=rfc1483

set cmiface [1.3.1.2] netmodel=cross-connect
       encapsulationtype=none

set cmiface [1.8.1] netmodel=cross-connect
       encapsulationtype=q.922

set cmsubiface [1.8.1.19] destpii=1.3.1.1

set cmsubiface [1.8.1.20] destpii=1.3.1.2
```

After configuration, WAN VC 1.3.1.1 has a forwarding mode of FRF8-1490-1483, while WAN VC 1.3.1.2 has a forwarding mode of VC-VC-payload.

Not all translations described by the FRF.8.1 specification are implemented. Translations are implemented for Frame Relay PDUs with:

- NLPID=0x80, which include but are not limited to Bridged PDUs (802.3 protocol) and Other routed PDUs (AppleTalk data, AppleTalk ARP, IPX, Inverse ARP but not ARP)
- NLPID=0xCC, which are Routed IP Version 4 PDUs
- NLPID=0x08, Connection oriented protocols (Q.933/ Q.2931 NLPID) PDUs that include but are not limited to SNA

Packets that do not fall into these categories will be discarded.

PPP Translation and Transparent Forwarding Modes

The PPP-translation forwarding mode supports RFC-1973 to RFC-2364 interworking. PPP-transparent forwarding mode supports RFC-1973 to RFC-1973 interworking. At present, RFC-2364 to RFC-1973 and RFC-2364 to RFC-2364 interworking are not supported.

RFC-2364 actually specifies two encapsulations, the Null (VC Multiplexing) and the Logical Link Control (LLC) encapsulations. ATM based line cards and WAN interfaces in the CE200 support both RFC-2364 encapsulations.

To convert an RFC-1973 frame to RFC-2364, the flag and Q.922 header are stripped away. For the RFC-2364 Null encapsulation, Control and NLPID are removed as well. For the RFC-2364 LLC encapsulation, an LLC header is prepended to the frame and the NLPID is retained. With both RFC-2364 encapsulations, an optional pad and a CPCS-PDU trailer are appended to the frame at the ATM driver level. The process is reversed to convert an RFC-2364 frame to RFC-1973.

The following frame-based line cards and WAN interfaces in the CE200 support RFC-1973 encapsulation:

- SDSL
- IDSL
- T1
- DS3 Frame
- Quad T1
- V.35

The following ATM-based line cards and WAN interfaces in the CE200 support RFC-2364 encapsulation:

- G.lite
- G.dmt
- DS3 ATM
- G.SHDSL

RFC-1973 is configured on the port level for DSL links. Then DSL VCs are configured with cmSubIface entries. The DSL VCs assume the DSL port encapsulation.

To configure DSL RFC-1973 to WAN RFC-2364-null:

1. Set the DSL port netmodel and encapsulation:

```
CRAFT (1.2)> set cmiface [1.6.1] netmod=cross-connect
       encapsulationtype=rfc1973
```

2. Create the cmSubIface row and set the destination PII:

```
CRAFT (1.2)> set cmsubiface [1.6.1.20] destpii=1.3.1.100
```

3. Set WAN VC encapsulation:

```
CRAFT (1.2)> set cmiface [1.3.1.100] encapsulationtype= rfc2364-
       null
```

4. Create a cmAtmVcl row for the WAN VC:

```
CRAFT (1.2)> set cmatmvcl [1.3.1.100] vpi=1 vci=121
       adminstatus=up
```

After configuration, WAN VC 1.3.1.100 has a forwarding mode of PPP-translation. As with other per-VC encapsulations, the DSL VC's forwarding mode is the same as that of the cross-connected WAN VC.

To configure DSL RFC-1973 to WAN RFC-1973:

1. Set the netmodel and encapsulation on the DSL port:

```
CRAFT (1.2)> set cmiface [1.6.1] netmodel=cross-connect
       encapsulationtype=rfc1973
```

2. Create the cmSubIface row and set the WAN destination PII:

```
CRAFT (1.2)> set cmsubiface [1.6.1.20] destpii=1.3.1.100
```

3. Set the WAN VC encapsulation:

```
CRAFT (1.2)> set cmiface [1.3.1.100] encapsulationtype= rfc1973
```

   After configuration, both WAN VC 1.3.1.100 and DSL VC 1.6.1.20 have a forwarding mode of PPP-transparent.

Per-Port Forwarding Modes

The Cross-Connect netmodel per-port forwarding modes result from a DSL encapsulation that defines the DSL port as a single logical interface that does not support VCs.

DSL Encapsulation	Line Card Type	WAN Encapsulation	Forwarding Mode
PPP-HDLC	Frame	RFC-1973	PPP-HDLC-1973
PPP-HDLC	Frame	None	HDLC-VC-payload

PPP-HDLC Forwarding Modes

The Cross-Connect netmodel supports PPP-HDLC encapsulation, which is valid only for DSL ports, not for WAN VCs. The forwarding mode in the previous table applies to both the DSL port and the WAN VC.

PPP-HDLC (RFC-1662) is a point-to-point protocol that allows a DSL port on the CE200 to receive PPP-HDLC frames. The CE200 can transform them into RFC-1973 frames, and then send them to a WAN VC. Different ports carrying PPP-HDLC are mapped to VCs

on the same WAN interface. The operation is called *Point-to-Point Protocol Multiplexing*.

To configure the PPP-HDLC-1973 forwarding mode, set the netmodel and encapsulation on the DSL and WAN interfaces:

1. Set the DSL port's netmodel and encapsulation, and specify the WAN VC as the destination PII:

   ```
   CRAFT (1.2)> set cmiface [1.6.1] netmodel=cross-connect
          encapsulationtype=ppp-hdlc destpii=1.3.1.100
   ```

2. Set the netmodel and encapsulation for the WAN VC:

   ```
   CRAFT (1.2)> set cmiface [1.3.1.100) netmodel=cross-connect
          encapsulationtype=rfc1973
   ```

   After configuration, both WAN VC 1.3.1.100 and DSP port 1.6.1 have a forwarding mode of PPP-HDLC-1973.

3. To configure the HDLC-VC-payload forwarding mode, set the netmodel and encapsulation on the DSL and WAN interfaces:

4. Set the DSL port's netmodel and encapsulation, and specify the WAN VC as the destination PII:

   ```
   CRAFT (1.2)> set cmiface [1.6.1] netmodel=cross-connect
          encapsulationtype=ppp-hdlc destpii=1.3.1.100
   ```

5. Set the netmodel and no encapsulation for the WAN VC:

   ```
   CRAFT (1.2)> set cmiface [1.3.1.100) netmodel=cross-connect
          encapsulationtype=none
   ```

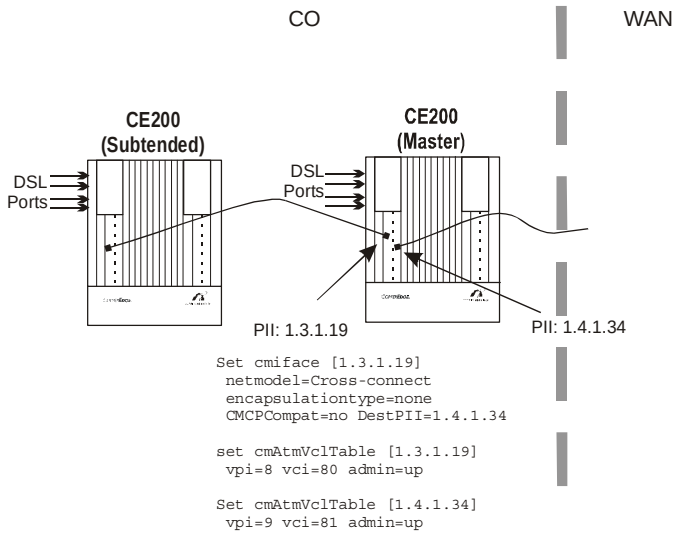
   After configuration, both WAN VC 1.3.1.100 and DSP port 1.6.1 have a forwarding mode of HDLC-VC-payload.

For more information about the Cross-Connect netmodel, refer to Appendix A, *Data Encapsulation*.

WAN-To-WAN Cross-Connect (Subtending)

Beginning with Release 7.0, the CE200 has the capability of cross-connecting two ATM WAN ports (DS3-ATM, DS1/E1 IMA, or OC-3c/STM1) on the same CE200 in order to enable subtending of a second CE200 or CE150. Other, third-party ATM devices can also be subtended by the CE200, provided those subtended devices use ATM traffic classes and AAL types that are supported on the CE200.

Subtending allows you to aggregate the subscriber interfaces from two DSLAMs onto a single ATM facility (DS3, T1/E1 IMA or OC-3c/STM-1) for transmission over the network. Subtending should not be confused with multi-shelf operation. When a CE200 subtends another DSLAM in a so-called "master" configuration, it has no provisioning or management function, and the master is transparent to the "subtended" DSLAM. To the master, the entire subtended shelf is treated as if it were a DSL interface, and the WAN port used to physically connect the master and the subtended DSLAM is configured as if it were a DSL interface. The egress (WAN) port on the subtended unit, however, is configured as a normal WAN interface. The diagram below illustrates the concept.



WAN-to-WAN Cross-Connect (Subtending)

High Density Internet Access (HDIA) Netmodel

Beginning in Release 7.0, the HDIA networking model has been replaced by the revised and improved CopperVPN (also referred to as CopperVPN+), described previously in this chapter.

Configuring G.SHDSL Interfaces

A 24-port G.SHDSL (ITU-T G.991.2) line card is available which, for subscribers with compatible CPE, can provide a range of data rates from 19.200 kbps to 2.034 Mbps. Rates can be fixed or, using a rate-adaptive configuration option, can be allowed to fluctuate with the line quality of the local loop.

The G.SHDSL protocol as implemented in the CE line card is a cell-based ATM protocol. Besides allowing higher maximum rates, the G.SHDSL standard has a greater reach than SDSL, which can be further extended through use of repeaters (regenerator units).

In the CopperEdge MIB, a number of object groups (the *hdsl2Shdsl* groups) have been added to allow you to monitor performance of the G.SHDSL module, its links and subscribers, and to control maintenance functions such as loopback tests.

G.SHDSL interfaces are established using the *cmIfaceTable*, and the basic procedure is the same as any other SDSL or ADSL interface. To configure the G.SHDSL characteristics of the interface, however, you use *cmGSHDSLModemTable* to set the minimum and maximum data rates, up- and downstream SNR margins, and training mode (fixed or rate-adaptive).

Example:

```
CRAFT (1.2)> set cmgshdslmodem [1.8.11] maxdata=2304000
       mindata=768000 conftrainmode=rateadaptive

Group: cmGSHDSLModemTable
Instance: [1.8.11.0]
Index             =  1.8.11.0
MaxDataRate       =  2304000
CurrDataRate      =  768000
TransmissionMode  =  1
RemoteEnabled     =  disabled
HostMode          =  central
DSLMode           =  GSHDSL
SuccessfulTrains  =  1
FailedTrainingAt  =  0
CpeResponses      =  2
ModemCommand      =  None
Debug             =  00.00.00.00
MinDataRate       =  192000
CurrSNRDown       =  0
CurrSNRMarginDown=  6
MinimumSNRMarginD=  2
CurrSNRUp         =  0
CurrSNRMarginUp   =  6
MinimumSNRMArginU=  4
ConfPSD           =  symmetric
CurrPSD           =  symmetric
ConfTrainMode     =  rateadaptive
CurrTrainMode     =  rateadaptive
TxPower           =  0
PBOVal            =  0
RxGain            =  0
```

For additional details, see the *CopperCraft Reference and MIB Definitions* document.

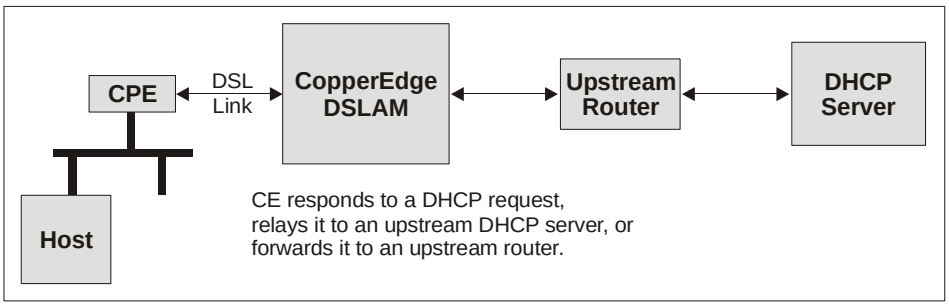
Configuring the CE200 for DHCP

For specific ports or VCs, you can configure the CE200 to process DHCP requests in one of the following modes:

- *DHCP Server*—The CE200 responds to a DHCP request from any connected CPE, sending the IP address entered in the cmDHCPTable for that DSL PII.
- *DHCP Relay Agent*—The CE200 sends the DHCP request from any CPE or premise LAN host to an upstream DHCP server. (To use this mode, the netmodel must be IP).
- *DHCP Forwarding Agent*—The CE200 sends the DHCP request from any CPE or premise LAN host to an upstream DHCP server. The netmodel must be Cross-Connect, VWAN, or CopperVPN.

Configured as a DHCP server, the CE200 cannot distinguish between non-CMCP CPEs and their hosts on the premise LAN. It can assign only one IP address from the cmDHCPTable, sending it to both non-CMCP CPEs and their hosts. Therefore, configure the CE200 to relay or forward the requests to an upstream DHCP server, which can respond with different IP addresses for both the CPEs and their hosts.

The CE200's DHCP features, based on RFC-1542, give service providers the option of having their own DHCP servers provide IP configuration parameters for CPEs and hosts on the premise LAN.



DHCP Functionality in the CE200

As described below, the CE200 assigns a client/circuit identifier when sending DHCP requests to an upstream DHCP server if that server is to be able to uniquely identify the clients (CPEs and their hosts) sending the requests.

- A unique circuit identifier is automatically entered in the CircuitID object in the *cmDHCPTable*. The default is the CE200 system name plus the DSL PII. You can use a different ID, but it must start with an alpha character.
- The CircuitID (Option 82) and Client Identifier (Option 61) are inserted on their respective fields in the DHCP request message.

An upstream DHCP server uses Option 82 (Circuit ID) to identify which DSL port the DHCP request originated on. It uses Option 61, Client Identifier, to distinguish between requests originating from a CPE and requests originating from a host. If Option 61 is present, the request originated from a CPE; if it is not present, the request originated from a host.

The CE200's DHCP functionality depends on which netmodel is being used and whether the CPE is a CMCP CPE.

DHCP Function	Network Model	CMCP CPE	Host with CMCP CPE	Non-CMCP CPE or Host
CE200 as a Server	IP	Response	Relay	Response[1]
	CopperVPN	Response	Forward	Response[1]
	VWAN	Response	Forward	Response[1]
	Cross-Connect	Response	Forward	Response[1]
CE200 as a Relay Agent	IP	Relay[2]	Relay[3]	Relay[3]
CE200 as a Forwarding Agent	CopperVPN	Forward[2]	Forward[3]	Forward[3]
	VWAN	Forward[2]	Forward[3]	Forward[3]
	Cross-Connect	Forward[2]	Forward[3]	Forward[3]

1. The CE200 cannot distinguish between non-CMCP CPEs and their hosts on the premise LAN. It can only assign one IP address, which comes from the cmDHCPTable. The CE200 sends that IP address to both the non-CMCP CPEs and their hosts. Therefore, configure the CE200 to either Relay or Forward these requests to an upstream DHCP server, which can assign different IP addresses for both.
2. Both the Circuit ID and Client Identifier are inserted in the request.
3. Only the Circuit ID is inserted in the request.

The following sections describe the CE200's DHCP functionality, both upstream and downstream.

Upstream DHCP Processing

In the upstream direction, the CE200 can function as a DHCP Server, Relay Agent, or Forwarding Agent for a DSL PII, depending on the netmodel.

You use the cmDHCPTable to configure DHCP functionality for the DSL PII. If there is no entry in the cmDHCPTable for the DSL PII, the CE200's default mode is to pass the DHCP request to the upstream router or DHCP server without inserting the Circuit ID or Client Identifier in the message. See Chapter 3 of the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual for information about the cmDHCPTable.

IP Netmodel

When the netmodel is IP, you can configure the CE200 to act as a DHCP Server or as a DHCP Relay Agent for a DSL port or VC.

The CE200 as a DHCP Server

For CMCP CPEs:

- The CE200 responds to DHCP requests from CMCP CPEs for which there is an entry in the cmDHCPTable.
- The CE200 generates a response based on the configured objects in the cmDHCPTable and sends the response back to the DSL PII that it arrived on.

For hosts on the premise LAN:

- The CE200 relays the DHCP requests to an upstream DHCP server. The Relay function is described on page 54.
- The CE200 must know the IP address of the upstream DHCP server; you enter this address in the cmDHCPTable.
- The CE200 must also have an IP address assigned to itself, which the upstream DHCP server will use to route the DHCP responses back to the CE200.

For non-CMCP CPEs and hosts on the premise LAN:

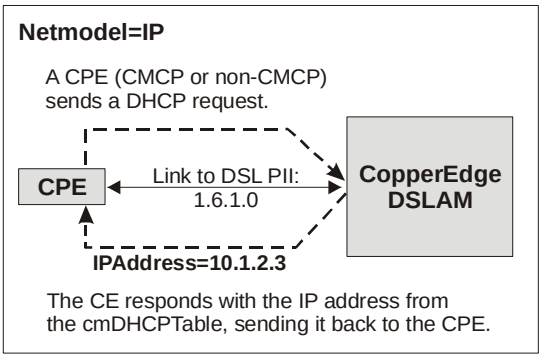
- The CE200 responds to DHCP requests from non-CMCP CPEs for which there is an entry in the cmDHCPTable.
- However, the CE200 cannot differentiate between DHCP requests coming from non-CMCP CPEs or hosts on the premise LAN. It sends the same response for both.
- Therefore, for non-CMCP CPEs, configure the CE200 as a Relay Agent instead of as a DHCP Server. See page 54.

To configure the CE200 to act as a DHCP Server for a port or VC:

In cmDHCPTable, enter the upstream DHCP server's IP address in the ServerIPAddr object and set the Function object to

DHCPRespond. You can leave the CircuitID object at the default (the system name and the PII of the port or VC).

```
set cmdhcp [1.6.1.0] function=dhcprespond
       serveripaddr=204.20.20.30

Set Successful

get cmdhcp [1.6.1.0]

Group: cmDHCPTable
Instance: [1.6.1.0]
PII                          = 1.6.1.0
RowStatus                    = Active
IpAddress                    = 10.1.2.3
NetMask                      = 255.255.255.255
DefaultRoute                 = 204.20.20.10
DNSServer                    = 204.20.20.20
Function                     = DHCPRespond
ServerIPAddr                 = 204.20.20.30
CircuitID                    = CMTN-1.6.1.0
```



CE200 *as DHCP Server*

The CE200 as a DHCP Relay Agent

The CE200 relays DHCP requests from CPEs (both CMCP and non-CMCP) and hosts on the premise LAN to an upstream DHCP server.

To configure the CE200 to act as a DHCP Relay Agent for a port or VC:

In cmDHCPTable, enter the upstream DHCP server's IP address and set the Function object to DHCPRelay. You can leave the CircuitID object at the default (the system name and the PII of the port or VC).

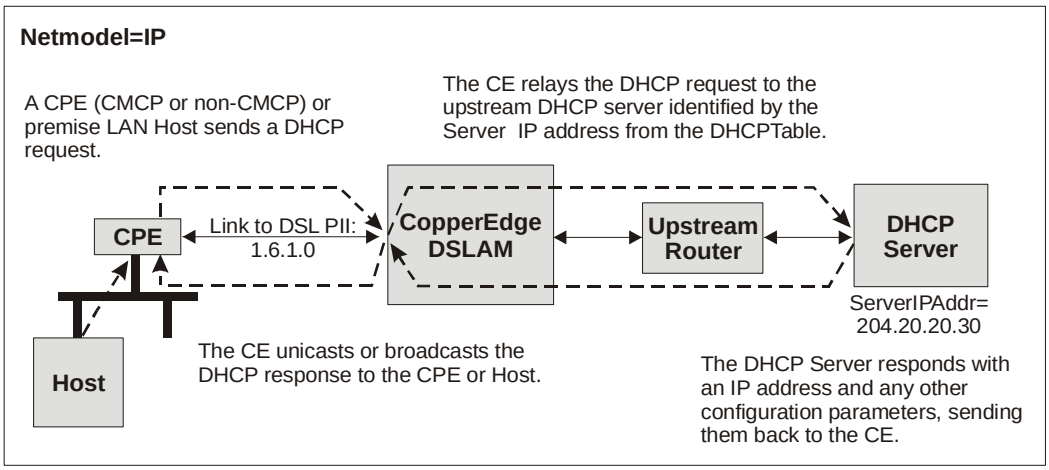
```
set cmdhcp [1.6.1.0] function=dhcprelay
       serveripaddr=204.20.20.30

Set Successful

get cmdhcp [1.6.1.0]

Group: cmDHCPTable
Instance: [1.6.1.0]
```

```
PII                         = 1.6.1.0
RowStatus                   = Active
IpAddress                   = 10.1.2.3
NetMask                     = 255.255.255.255
DefaultRoute                = 204.20.20.10
DNSServer                   = 204.20.20.20
Function                    = DHCPRelay
ServerIPAddr                = 204.20.20.30
CircuitID                   = CMTN-1.6.1.0
```

The CE200 performs the following functions when it receives a DHCP request:

1. The CE200 inserts the IP address of the DSL PII, over which the request was received, in the DHCP request. This IP address will be used by the upstream DHCP server to send the DHCP response back to the CE200.
2. For a CMCP CPE, the CE200 modifies the DHCP request by inserting the CircuitID from the cmDHCPTable into the Relay Agent Information and Client Identifier fields in the DHCP request message.

   For a non-CMCP CPE or a host on the premise LAN, the CE200 modifies the DHCP request by inserting the CircuitID only into the Relay Agent Information field.
3. The CE200 sends the request upstream as a unicast IP packet destined to the DHCP Server IP address configured in the cmDHCPTable.



CE200 as DHCP Relay Agent

Pass Through Mode

If the CE200 receives a DHCP request and there is no entry in the cmDHCPTable for that DSL PII, the CE200 passes the request to the upstream router or DHCP server. The DHCP request is not modified (the Relay Agent Information and Client Identifier fields are not inserted in the message).

When the netmodel is IP, the CPE or premise LAN host *cannot* obtain an IP address from the upstream DHCP server. The CE200 simply drops the response it receives from the upstream DHCP server.

Cross-Connect, VWAN, and CopperVPN Netmodels

When the netmodel is Cross-Connect, VWAN, or CopperVPN, you can configure the CE200 to act as a DHCP Server or as a DHCP Forwarding Agent for a port or VC.

The CE200 as a DHCP Server

For CMCP CPEs:

- The CE200 responds to DHCP requests from CMCP CPEs for which there is an entry in the cmDHCPTable.
- The CE200 generates a response based on the configured objects in the cmDHCPTable and sends the response back to the DSL port that it arrived on.

For hosts on the premise LAN:

- The CE200 forwards the DHCP requests to an upstream DHCP router, which is responsible for sending the request to the appropriate DHCP server.
- The Forwarding function is described on page 57.

For non-CMCP CPEs and hosts on the premise LAN:

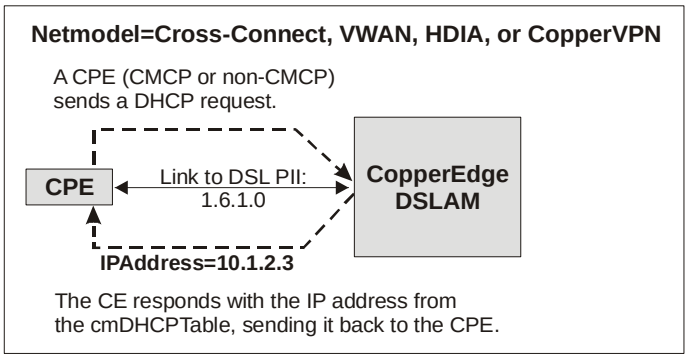
- The CE200 responds to DHCP requests from non-CMCP CPEs for which there is an entry in the cmDHCPTable.
- However, the CE200 cannot differentiate between DHCP requests coming from non-CMCP CPEs or hosts on the premise LAN. It sends the same response for both.
- Therefore, for non-CMCP CPEs, configure the CE200 as a Forwarding Agent instead of as a DHCP Server. See page 57.

To configure the CE200 to act as a DHCP Server for a port or VC:

In cmDHCPTable, set the Function object to DHCPRespond. You can leave the CircuitID object at the default (the system name and the PII of the port or VC).

```
set cmdhcp [1.6.1.0] function=dhcprespond
Set Successful

get cmdhcp [1.6.1.0]
Group: cmDHCPTable
Instance: [1.6.1.0]
PII                         = 1.6.1.0
RowStatus                   = Active
IpAddress                   = 10.1.2.3
NetMask                     = 255.255.255.255
DefaultRoute                = 204.20.20.10
```

```
DNSServer                    = 204.20.20.20
Function                     = DHCPRespond
ServerIPAddr                 = 0.0.0.0
CircuitID                    = CMTN-1.6.1.0
```



CE200 as DHCP Server

The CE200 as a DHCP Forwarding Agent

The CE200 forwards DHCP requests from CPEs (both CMCP and non-CMCP) and hosts on the premise LAN to an upstream WAN VC or router.

To configure the CE200 to act as a DHCP Forwarding Agent for a port or VC:

In cmDHCPTable, set the Function to DHCPForward. You can leave the CircuitID object at the default (the system name and the PII of the port or VC).

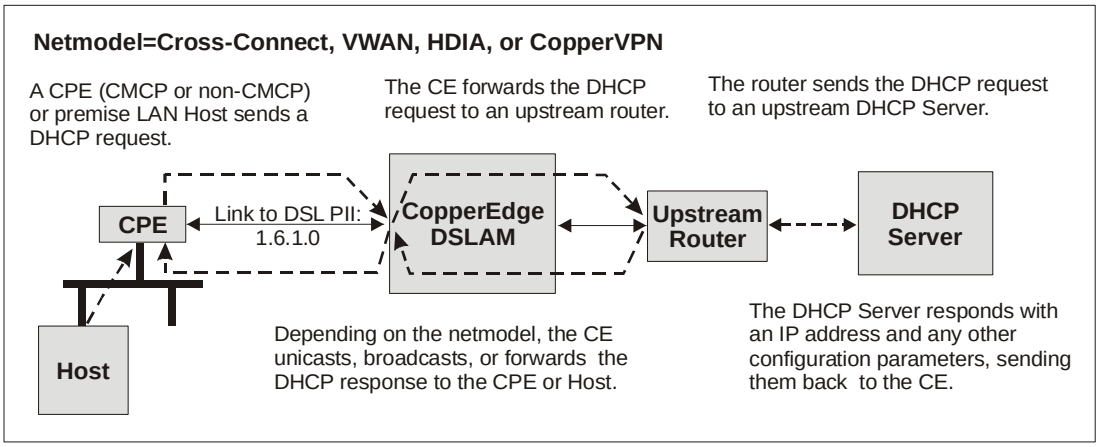
```
set cmdhcp [1.6.1.0] function=dhcpforward
       serveripaddr=204.20.20.30

Set Successful

get cmdhcp [1.6.1.0]

Group: cmDHCPTable
Instance: [1.6.1.0]
PII                          = 1.6.1.0
RowStatus                    = Active
IpAddress                    = 10.1.2.3
NetMask                      = 255.255.255.255
DefaultRoute                 = 204.20.20.10
DNSServer                    = 204.20.20.30
Function                     = DHCPForward
ServerIPAddr                 = 0.0.0.0
CircuitID                    = CMTN-1.6.1.0
```

The CE200 performs the following functions when it receives a DHCP request:

1. For a CMCP CPE, the CE200 modifies the DHCP request by inserting the CircuitID from the cmDHCPTable into the Relay Agent Information and Client Identifier fields in the DHCP request message.

For a non-CMCP CPE or a host on the premise LAN, the CE200 modifies the DHCP request by inserting the CircuitID only into the Relay Agent Information field.

2. The CE200 forwards the request to the upstream WAN VC or router, which is responsible for responding to or relaying the request to the upstream DHCP server.



CE200 *as DHCP Forwarding Agent*

Pass Through Mode

If the CE200 receives a DHCP request and there is no entry in the cmDHCPTable for that DSL PII, the CE200 passes the request to the upstream router or DHCP server. The DHCP request is not modified (the Relay Agent Information and Client Identifier fields are not inserted in the message).

When the netmodel is Cross-Connect, VWAN, or CopperVPN, the CPE or premise LAN host can obtain an IP address from the upstream DHCP server. There is enough information in the message for the CE200 to forward the response to the correct DSL PII and ultimately to the correct CPE or premise LAN host.

Downstream DHCP Processing

In the downstream direction, the following DHCP processing takes place for DHCP Response messages for the various netmodels. The CE200 will drop any DHCP requests coming downstream from a WAN VC or an Ethernet port.

IP Netmodel

The DHCP response is routed to the DSL PII based on the IP address of that DSL PII and the assigned Client IP address of the received DHCP response. The DHCP response is either broadcast downstream on the DSL PII or sent as a unicast packet based on the DHCP broadcast flag.

Cross-Connect and VWAN Netmodels

The downstream DHCP responses are simply forwarded to the appropriate DSL PIIs based on the provisioning configured in the CE200. No additional processing is required.

CopperVPN Netmodel

If the received DHCP response is a broadcast IP packet, it is flooded to all members of the CopperVPN group as a broadcast, and the DHCP client identifies its response based on the MAC Address in the DHCP response.

If the received DHCP response is an IP unicast packet, the CE200 replicates this packet to all DSL PIIs in the CopperVPN group as a unicast packet, and uses the MAC Address in the DHCP response as the unicast MAC address. Therefore, only the correct DHCP client will be able to receive the response.

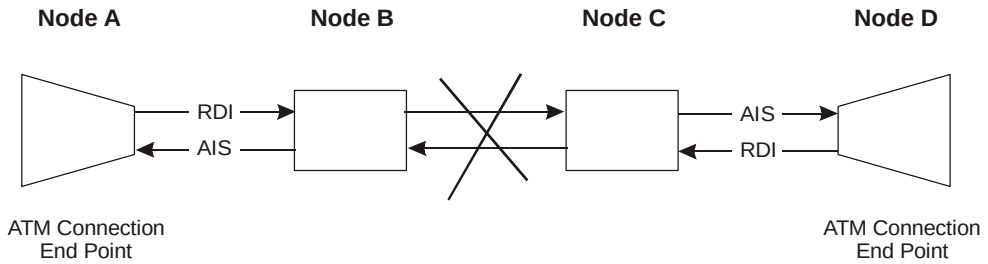
OAM Fault Management for ATM WAN Links

The OAM fault management function on the CE200 allows system administrators to monitor basic conditions for transmitting and receiving on end-to-end ATM WAN links, made up of many segments. It also allows them to locate a break on one of the segments.

Transmitting/Receiving Fault Messages

When a fault occurs on upstream or downstream segments, OAM fault management software installed on switching or routing devices on the segments sends out Alarm Indication Signal (AIS) messages. The messages travel to devices (such as DSLAMs and routers) at opposite ends of the link. In response, the DSLAM and routers send out Remote Defect Indication (RDI) messages.

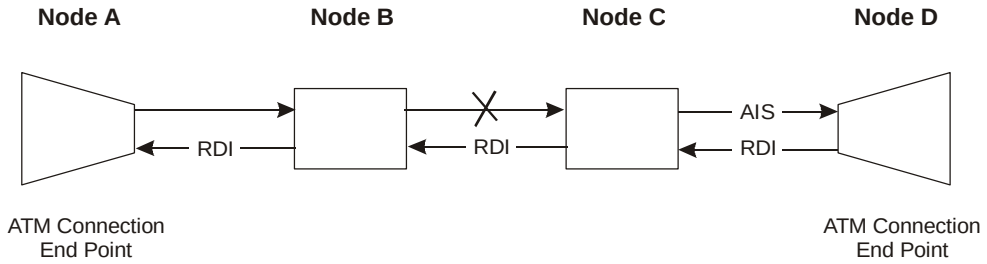
The diagrams below show two possible scenarios: first, a fault on both the upstream and downstream parts of a connection; second, a fault on the upstream part of a connection.



Connection Faults Upstream and Downstream

In the diagram above, the devices at Node B and Node C sense the fault between them. The device at Node B sends an AIS message to the device at the endpoint on Node A. Similarly, the device at node C sends an AIS message to the device on the endpoint at Node D. Each device on the endpoint sends an RDI message to the device at the other endpoint, notifying it of the fault.

But, neither message gets through the fault between Node B and Node C. Neither message needs to. The devices at both ends of the link know the connection is broken because of the AIS messages they have already received. Manual loopback testing can begin from the devices at either end to determine the exact location of the fault.



Upstream Connection Fault

In the diagram above, the device at Node C senses the fault on the upstream between Node B and Node C. It sends an AIS message to the device on the endpoint at Node D. The device at Node D responds and sends a RDI message to the device on the other end of the WAN connection. The message goes through because no fault is present on the downstream part of the connection.

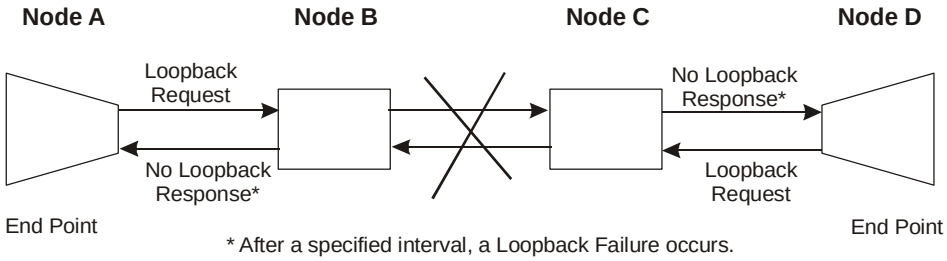
At this time, devices at both ends of the connection know the upstream path on the connection has a fault is broken, and two-traffic has stopped. Loopback testing can begin from the devices at either end to determine the exact location of the fault.

Transmitting and Receiving Loopback Messages

To determine the exact location of the fault, system administrators must use the OAM fault management function on one or both of the devices at the endpoints of the connection. Administrators can set a device to automatically send Requests for Loopback cells to a device at the other end of the connection on a regular basis; or they can set a device to send Requests for Loopback cells on a manual basis. Then administrators can read the counters for Requests for Loopback cells and the counters for Responses to Loopback Requests.

But to locate the fault, if the connection has multiple segments, someone must look at the counters for Requests and Responses on devices located on intervening segments. The fault on the connection will be between the last device to receive Requests for

Loopback cells and the first device to not receive those same Requests for Loopback cells.



Loopback Request/No Response

Configuring the OAM Function

To configure the OAM function, first configure the ATM WAN port, then the WAN VC, and finally the OAM object itself.

1. Configure the ATM WAN port for AdminStatus using the object group, IfTable. If the status is not already Up, set it to Up.

   ```
   CRAFT (1.2)> set cmiftable [1.3.1] adminstatus=up
   ```

2. Configure the ATM VC for AdminStatus, using the object group, cmAtmVcl. If the status is not already Up, set it to Up.

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.1.101] adminstatus=up
   ```

3. Configure the ATM WAN port for OAM AdminState, using the object group, cmAtmIfExt. If the status is not already Enabled, set it to Enabled.

   ```
   CRAFT (1.2)> set cmatmifext [1.3.1] oamadminstate=enabled
   ```

4. Configure the ATM VCL for OAMAdminState, using the object group, cmAtmVcl. If the state is not already Enabled, set it to Enabled.

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.1.101] oamadminstate=enabled
   ```

5. Check your work in the cmAtmVcl group:

   ```
   CRAFT (1.2)> get cmatmvcl [1.3.1.101]

   Group: cmAtmVclTable
   Instance: [1.3.1.101]
   PII                  =  1.3.1.101
   Vpi                  =  2
   Vci                  =  11
   AdminStatus          =  Up
   OperStatus           =  Up
   LastChange           =  12 day 1 hour 44 min 54.0 sec
                           (2002/09/28-14:44:46)
   AalType              =  Aal5
   Aal5CpcsTransmitSduS =  1600
   Aal5CpcsReceiveSduSi =  1600
   RowStatus            =  Active
   TransmitTrafficDescr =  2
   OAMState             =  normal (0)
   OAMAdminState        =  enabled
   OAMAutoLBState       =  disabled
   OAMManualLBCmd       =  none
   OAMManualLBCmdStatus =  none
   OAMLBInterval        =  17
   OAMLBTimeOut         =  18
   OAMTxAISCells        =  0
   OAMRxRDICells        =  0
   ```

```
OAMRxAISCells        =  0
OAMTxRDICells        =  0
OAMTxLBRequestCells  =  0
OAMRxLBResponseCells =  0
OAMRxLBRequestCells  =  0
OAMTxLBResponseCells =  0
OAMRxUnsupportedCell =  0
OAMTxDiscards        =  0
OAMRxDiscards        =  0
```

With this configuration, the OAM function will respond to AIS messages from the network by sending out RDI messages. It also will respond to Loopback Request messages sent by a device on the other end of the network. But, it will not send out Loopback Requests.

Configuring the OAM Loopback Function

The loopback function has two parts: an automatic function and a manual function. You can operate them at the same time or separately. The automatic function is simply for discovering when a fault occurs; the manual function is for discovering where the fault is on the connection.

Automatic Loopback Requests

The automatic loopback function will operate as long the port and the VC are UP and as long as the OAMAdminState function is Enabled. It will operate at the periods specified in the Interval and Timeout objects. It will record loopback failure in the OAMState object.

To configure the loopback function within the OAM function, first configure the ports and VCs as described in the previous section, then the OAMAutoLBState object. You may want to make changes in the loopback interval and the loopback timeout. Follow these steps:

1. Check the ATM WAN port and the ATM VC for their AdminStatus. Make sure they are both set to Up.
2. Configure the ATM VC for OAMAutoLBState by setting it to Enable:

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.1.101] oamautolbstate=enabled
   ```

3. Check the OAMLBInterval object. Its default is 17 seconds; it has a range of 1 to 999 seconds. Reset it if necessary.

   OAMLBInterval specifies how often a Loopback Request will be sent until either a Loopback Response is received or a timeout occurs. If no response is received before timeout, the OAMState object indicates a loopback failure.

4. Check the OAMLBTimeOut object. Its default is 18; it has a range of 1 to 999 seconds. Reset it if necessary to a longer period.

   OAMLBTimeOut specifies how long Loopback Requests will be sent. The timeout period must always be longer than the interval period.

With this configuration, the OAM function will send out periodic f5End2End Loopback requests to the device on the far end of the network. If the device at the far end receives the requests—if no breaks exist on the connection—it will return Loopback responses during each period. If the device on the near end does not receive a response before the timeout period end, it records a loopback failure in the OAMState object.

Manual Loopback Requests

The manual loopback function will operate only when the port and the VC are UP, the OAMAdminState function is Enabled, and a command for the OAMManualLBCmd object has been sent. It will operate at the periods specified in the Interval and Timeout objects, that you either accepted or changed in the configuration of automatic loopback requests. The success or failure of a manually sent loopback will be recorded only in the OAMManualLBCmdStatus object, never in the OAMState object.

After the automatic function discovers the existence of a fault, this function will help you to discover which segment on the connection has the fault.

To send a manual loopback, follow these steps:

1. Set the OAMManualLBCmdStatus object to f5end2end:

```
CRAFT (1.2)> set cmatmvcl [1.3.1.101] oammanuallbcmd=f5end2end
```

2. Check the OAMManualLBCmdStatus object. You will see one of three conditions: in progress; succeeded; failed.

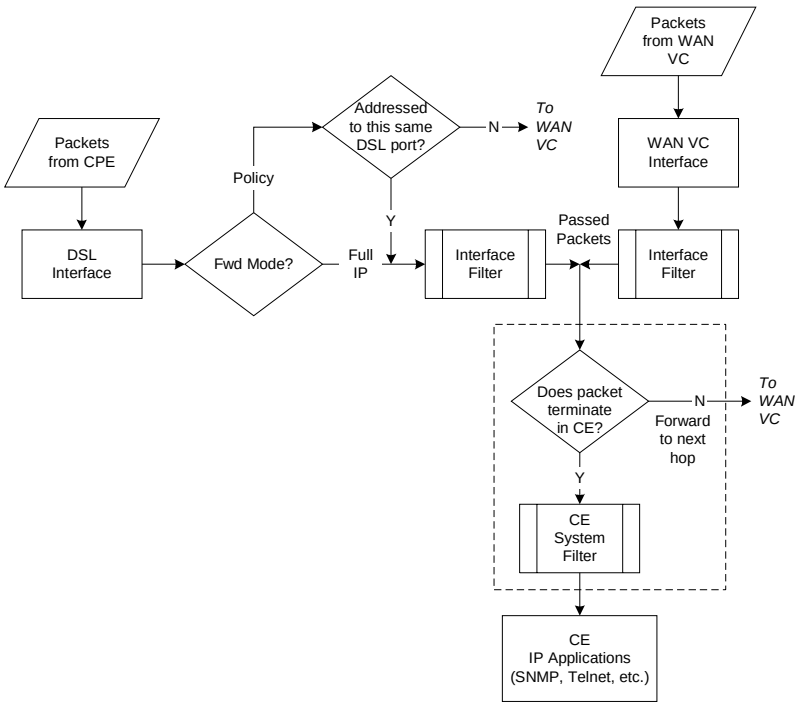
```
CRAFT (1.2)> get cmatmvcl [1.3.1.101]
Group: cmAtmVclTable
Instance: [1.3.1.101]
PII                  =  1.3.1.101
Vpi                  =  2
Vci                  =  11
AdminStatus          =  Up
OperStatus           =  Up
LastChange           =  12 day 1 hour 44 min 54.0 sec
                        (2002/09/28-14:44:46)
AalType              =  Aal5
Aal5CpcsTransmitSduS =  1600
Aal5CpcsReceiveSduSi =  1600
RowStatus            =  Active
TransmitTrafficDescr =  2
OAMState             =  normal (0)
OAMAdminState        =  enabled
OAMAutoLBState       =  enabled
OAMManualLBCmd       =  f5end2end
OAMManualLBCmdStatus =  succeeded
OAMLBInterval        =  17
OAMLBTimeOut         =  18
OAMTxAISCells        =  0
OAMRxRDICells        =  0
OAMRxAISCells        =  0
OAMTxRDICells        =  0
OAMTxLBRequestCells  =  0
OAMRxLBResponseCells =  0
OAMRxLBRequestCells  =  0
OAMTxLBResponseCells =  0
OAMRxUnsupportedCell =  0
OAMTxDiscards        =  0
OAMRxDiscards        =  0
```

IP Filters

When you configure a DSL interface with the IP netmodel, the CE200 allows all inbound messages to pass through without regard to IP address. Normally, the subscriber equipment processes only message packets that contain its IP address and ignores packets with a different address. Occasionally, data should not even *reach* a subscriber's equipment. A business, for example, would want to ensure that its message traffic is not accessible to its competition.

IP filtering lets you configure links to selectively pass or discard inbound IP packets based on parameters that you set. In general, filtering applies only to packets that are full-IP routed. The one exception is for packets that are policy routed to the IP address of the same DSL interface through which they entered the CE200 (such as a Ping from a device on a LAN at the customer premise to the IP address of its own associated CE200 DSL interface). Otherwise, policy-routed packets are not filtered.

Filtering cannot be done on interfaces with a networking model *other than* IP or (using the Group Filter function) on CopperVPN groups. Also, filtering works only on *inbound* packets, before they reach the IP stack. Note, however, that packets may be inbound from either a WAN interface or from a DSL interface, provided they are destined for an IP address that terminates in the CE200 itself. The following flow chart shows the general flow of filtered packets in the CE200.



IP Filtering and Packet Flow in the CopperEdge DSLAM

Every source of IP packets can be identified by its PII. Each DSL port, network port, and every virtual circuit aggregated into a

high-speed digital facility can be referenced by its PII. Each PII can be configured with a separate, independent set of IP filters.

Each filter, in turn, consists of a specific set of criteria against which incoming packets are compared. If the incoming packet matches the criteria specified by the filter, the packet is subjected to the action (Pass, Block, or Chain) specified by that filter.

When a packet arrives, it is checked against the filters in the list, in order. The first filter which matches the packet determines which action will be taken. An action code is also programmed into the filter, defining whether the matched packet will be passed or blocked, or if the filter will simply be combined with another (chained) to further specify the applicable range of packet values.

For maximum security, a packet which matches no filters is automatically blocked. However, *an interface for which no filters have been configured* ***passes*** *all packets.*

Since IP packets can also be addressed to the CE200 itself, system security may be enhanced by establishing a list of filters applicable to the CE200. To configure filters that apply to the overall system, use the CE200's "virtual PII": 1.0.0.0. As a further security measure, and to counteract an inherent vulnerability in Internet-compliant systems, the cmFilter table for the CE200 includes a static filter that will immediately discard any ICMP Redirect messages destined for the CE200.

Viewing Filters

As we have seen, each interface can have its own set of filters. In certain cases, this list could be a long one. If you are using the *CopperView* EM to control the CE200, then you can display the entire contents of the filter table. From the local *CopperCraft* interlace, you can retrieve them in batches using the *Getall* command and the interface of interest:

```
getall cmfilter [1.7.7]
```

To view filters singly, begin by retrieving the first filter in the list:

```
get cmfilter [1.7.7, 1]
```

Then, to view succeeding filters for the same interface, simply use the Getnext command:

```
getnext cmfilter
```

Specifying, Activating, and Deleting Filters

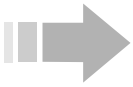
Activating or deleting filters from the list is accomplished using the cmFilter function object. For example:

```
set cmfilter [1.8.4, 3] function=delete
```

This command will delete filter number 3 for the interface specified (a DSL port at location 1.8.4). With the function object, the operator can command a filter to become active, or be deleted. When the filter specifications are displayed in response to a Get or *Getnext* command, the filter-function always displays as `active`. The operator can set it to any value, to perform the following

functions (recall that a filter definition must have **both** a PII and a filter-number):

- ACTIVE—Activates the filter criteria specified within the accompanying command (that is, in the same set statement as the active command). If there is already a filter at the location (list number) specified, it will be replaced with the filter you have defined. If you specify a filter-number larger than any on the existing list, the new filter will then be appended to the end of the filter-list.
- DELETE—Removes the specified filter. As shown in the example above, no other filter values should be entered. All filters following the deleted filter have their filter-numbers reduced by 1 as a result.



N O T E

Two previously available options for the Function *object (*insert *and* DeleteList*) are no longer supported. Consistent with standard SNMP usage, the filter list for an interface is now fully controlled by the operator, and the value of the FilterNumber must be explicitly specified in every case.*

Filter Criteria

Each IP filter includes the five filtering criteria listed below. In order for a packet to be considered as matching the filter, it must match *all five* of the categories. This means that all five items must be configured, even if the specification is *don't care*. For *don't care* fields, set the field to a value that will be true for every packet; for example, if you do not care about the Source Address, you can simply specify the source IP address and source subnet mask values as zero. Here are the five categories of filter criteria:

1. Source address and source mask.

   Match if (SA & mask) = value

2. Destination address and destination mask.

   Match if (DA & mask) = value

   IP Protocol Identifier of the filter. The ID is entered as an integer which translates to the type of protocol as shown in the table below. A match occurs if:

   Filter Protocol ID = Protocol ID of the packet, or

   Filter Protocol ID = 0 (any).

   Valid IP Protocol IDs are shown in the following table:

IP Protocol IDs

0	IP	Pseudo protocol number; matches any Internet protocol
1	ICMP	Internet control message protocol
2	IGMP	Internet group multicast protocol
3	GGP	Gateway-gateway protocol

6	TCP	Transmission control protocol
17	UDP	User datagram protocol

3. Source port comparison.

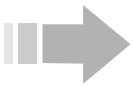
   For TCP and UDP packets only, compares the source TCP/UDP port with the port specified by the filter. The comparison standard can be set to detect a match if the value of the packet's source port is:

ANY	anything
LT	less than the filter port
GT	greater than the filter port
EQ	equal to the filter port
NE	not equal to the filter port.

4. Destination port comparison.

   For TCP and UDP packets only, compares the destination TCP/UDP port of the packet with the port of the filter. The comparison standard can specify a match if the value of the packet's destination port is:

ANY	anything
LT	less than the filter port
GT	greater than the filter port
EQ	equal to the filter port
NE	not equal to the filter port.



N O T E

In specifying Source or Destination Port Comparisons, please be aware that the options "Any" and "NE" are homophones of each other. Always check for understanding to be sure that any oral instruction to configure this option as "Any" is not misinterpreted as "NE," or vice versa.

The *pass* and *block* actions are self-explanatory; the *chain* and *redirect* actions are described below. The *chain* option is used to invoke multiple filtering criteria for the same packet, for example to specify a range of values; *redirect* forwards packets matching the filter to a specified upstream interface.

Chaining Filters

The *Chain* action combines two or more sequential filters in a list. Only when a packet matches all filters in the chain will the action in the last filter in the chain be taken. Thus, to block packets with source ports between 120 and 130, you would combine two filters:

Match source_port > 119, action: CHAIN

Match source_port < 131, action: BLOCK

Redirect

The *action=redirect* option is available for packets arriving at the CE from subscriber interfaces (not on WAN side interfaces).

Redirected packets are sent to the interface specified in the *RedirectPii* object. The outgoing interface must be a WAN-side interface. Note that *redirect* messages arriving at the CE from an upstream source and aimed at a subscriber interface are automatically discarded.

Specifying Filters

Remember, the first filter that matches the packet determines which action will be taken. So not only the selection criteria, but the order of their presentation is crucial to an effective filtering scheme.

To specify a complete packet filter, issue a Set command that includes all of the configurable objects in the cmFilter group, including the instance with its two index objects. For example:

```
*Index            = PII for this filter
*FilterNumber     = Filter Number for this filter
Function          = Sets status of the current filter:
                    Active or Delete
SrcAdrs           = Source IP Address
SrcMask           = Source Subnet Mask Address
SrcAdrsCompare    = None, equal, less, greater, not equal
DstAdrs           = Destination IP Address
DstMask           = Destination Subnet Mask address
DstAdrsCompare    = None, equal, less, greater, not equal
IpProtocol        = Filter Protocol Identifier
SrcPort           = Source port
SrcPortCompare    = None, equal, less, greater, not equal
DstPort           = Destination Port
DstPortCompare    = None, equal, less, greater, not equal
Action            = Filter Action: pass, block, chain,
                    redirect
RedirectPII       = PII of outgoing interface if
                    action=redirect
NumMatches        = Number of packets matching the rules
                    of this filter. Enter 0 to reset.
```

A full **`set cmfilter`** command would take the form shown in the following example:

```
set cmfilter [1.7.7, 1] function=active
        srcadrs=n.n.n.n srcmask=n.n.n.n
        dstadrs=n.n.n.n dstmask=n.n.n.n ipprotocol=0
        srcport=5 srcportcompare=eq dstport=8
        dstportcompare=ne action=pass
```

The cmFilter table is sorted by PII (SNMP first index), and then by filter-number (SNMP second index).

Sorting in the Filter Table

1.6.1	1	•••
1.6.1	2	•••
1.6.1	3	•••

Sorting in the Filter Table

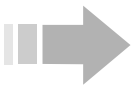
1.6.2	1	• • •
1.6.2	2	• • •
:	:	:

DSL IMUX

For subscriber endpoints equipped with the appropriate CPEs, the CE200 supports a technique called *Inverse Multiplexing*, or *IMUX*, in which multiple DSL ports are designated as members of a *bundle*, and treated as a single logical DSL port. The IMUX logical port and its DSL Multilink (multiple copper pairs originating at different physical DSL ports on the CE200, but terminating at a common endpoint CPE) combine to cumulatively increase both the bandwidth and the throughput of a single DSL link. Thus, an IMUX bundle containing two SDSL ports, each set at 1,568 Kbps, has a theoretical throughput of 3,136 Kbps. A small portion of the total, however, is reserved for management overhead. IMUX links may be either IDSL or SDSL, provided they are matched with the appropriate CPE and line module type.

In the current release, the CE200 supports two DSL ports in one bundle. But the application is capable of supporting four-port IMUX bundles in software.

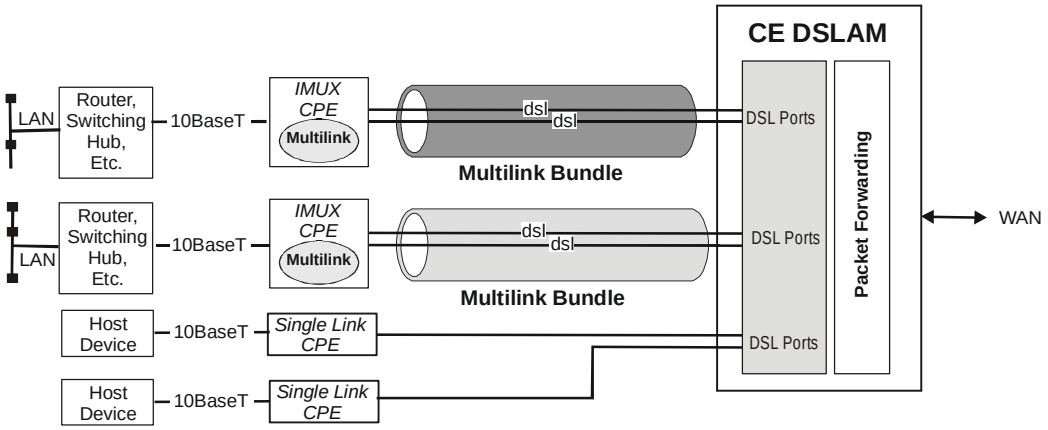
With few exceptions, IMUX bundles are treated in the same way as any other DSL port. Each bundled port is designated with its own PII and assigned to a logical slot 51, such as 1.51.3.



N O T E

Only IMUX-capable CPEs can be assigned as endpoints of an IMUX bundle; if a regular (single link) CPE is designated as the endpoint of an IMUX bundle and the CPE attempts to train, the connection will fail and the CE200 will generate an alarm. Moreover, any mismatch or conflict between the ports in an IMUX bundle and the bundle's endpoint CPE will result in an alarm condition.

In the following diagram, two-port DSL bundles are shown connecting the CE200 to separate IMUX CPEs. For contrast, two standard, single-port links are shown with their standard CPEs.



DSL Multilink (IMUX) Concept

Configuring an IMUX Bundle

In general, the procedure for setting up an IMUX bundle and configuring it is the same as for an individual DSL port, but there are a few special considerations.

When choosing which DSL ports to put into an IMUX bundle, note that they can be on the same DSL Module or on different ones, but all ports in the bundle must be on the same CE200.

1. Using cmHDSLModem, set the rates at which you want each of the ports in the bundle to operate.
2. For best multilink performance, set the two lines to the same speed. For example, if you want approximately 2000 Kbps throughput, it is better to set both lines at 1040 Kbps (for a total of 2080) than it is to set one at 1568 Kbps and the other at 416 Kbps (for a total of 1984).

```
CRAFT (1.2)> set cmhdsl [1.6.1] datarate=1040
Set Successful

CRAFT (1.2)> set cmhdsl [1.6.2] datarate=1040
Set Successful
```

3. Using cmBundle, create the bundle and enter the PIIs of the DSL ports on the CE200 that you want to assign to the bundle. You can enter all of the ports in the bundle with one command or you can enter one and the others later or you can create the bundle and add no ports initially.

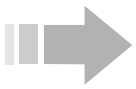
```
CRAFT (1.2)> set cmbundle [1.51.1] member1pii=1.6.1
       member2pii=1.6.2
Set Successful
```

4. If the CPE connected to the bundle is an IMUX CPE, then it will train normally, first on the Member1PII port, and then on the Member2PII port.

5. If the bundle is miswired or misconfigured, an alarm will be generated and a trap sent. If the configuration and wiring are correct, but there is no CPE connected or trained, the member ports in that bundle will show an endpoint ID consisting of zeros, and their status will indicate `WaitForAdd`. If a CPE is later connected, it will train normally and the member ports' endpoint and status information will automatically update.

6. Using cmIface, set the NetModel and the EncapsulationType for the IMUX bundle. Typically, the Cross-Connect netmodel is used for IMUX. Also, if you want to change the default for Service Class, you can do so at the same time. For example:

```
CRAFT (1.2)> set cmiface [1.51.1] netmodel=cross-connect
       encapsulationtype=q922
Set Successful
```



N O T E

When creating IMUX bundles, do not attempt to preconfigure the individual DSL Member ports with cmIFace or cmSubIface before assigning them to a bundle. If you try to assign configured ports to an IMUX bundle, the cmBundle command will fail, and the system will return an error message. If this happens, use cmIface to reset the DSL member port causing the error to NetModel=None. Configuration of NetModel and the resultant forwarding mode can only be done by configuring the bundle.

7. Using cmSubIface, create the virtual circuit(s) for the IMUX bundle. In the process, set the destination PII to point to the desired virtual circuit on the WAN link.

```
CRAFT (1.2)> set cmsubiface [1.51.1.16] rowstatus=create
       destpii=1.3.1.200
Set Successful
```

8. Using frCircuit, create the entry for the virtual circuit on the WAN link if it does not already exist. At the same time, set the values for the Frame Relay congestion-management parameters.

```
CRAFT (1.2)> set frcircuit [1.3.1.200] throughput=n1
       committedburst=n2 excessburst=n3
Set Successful
```

DSL Voice and Data Service

In addition to conventional packet data traffic, the CE200 can accommodate voice over IP and other types of real time IP traffic, such as video conferencing. These real time packets and non-real time packets can coexist within the CE200, and both types of signals can share the composite data stream over DSL ports and WAN ports.

Both types of packets can also flow simultaneously over any of the CE200's Virtual Circuits: ATM, Frame Relay, or DSL. To optimize voice performance in this mixed environment, the CE200 provides

for the assignment of various levels of priority to delay-sensitive, real time packets over non-real time packets.

Separate mechanisms are used for allocating priorities to traffic received over the DSL port (Class of Service), and traffic transmitted over the DSL port (Priority Queuing) without regard to the data rates on the various DSL links.

Class of Service

This function applies to packets *received* by the DSL ports on the CE200. It allows you to assign a level or class of service to each of the DSL ports, using *cmIface*. When you indicate the netmodel, the net mask, and the IP address for a DSL port, you also can enter a class of service for it. Typically, you will give a DSL port carrying voice a higher level of service than a DSL port carrying regular IP data traffic.

The CE200 allows you to set four levels of service: A, B, C, and D, where A is the highest class and D is the lowest. The default is D. The relative weight of each of the levels or classes is configured using the cmServiceClass MIB group. For more information on how these relative weights are derived, and how they affect their assigned interfaces, see the description in the cmServiceClass MIB group definition in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Once configured, the value selected as the weight for a particular class of service is the same for that class for every DSL port to which it is applied. For example:

```
get cmiface [1.6.24]

Group: cmIfaceTable
Instance: [1.6.24.0]
PII                 = 1.6.24.0
IfIndex             = 1.6.24.0
Name                = ....
GroupName           = ....
AdditionalInfo      = ....
NetModel            = IP
IpAddr              = 172.20.1.2
NetMask             = 255.255.255.0
MacAddr             = 0.60.58.1.0.76
BurnedInMacAddr     = 0.60.58.1.0.76
FarEndAddr          = 0.0.0.0
DestPii             = 1.3.1.43
CMCPCompatible      = Yes

EncapsulationType   = rfc1483
FwdMode             = IP-Policy
Pix                 = 3
Service Class       = A
```

In the example above, class of service on port 24 on the board occupying the slot 6 has been set to A, the highest class. To change the class of service on any of the other ports, use cmIface and change the defaults one at a time.

Class of service only comes into play when the CPU on the CE200 becomes congested. At that point, packets from ports with higher classes of service have a greater chance of being processed and passed on to the WAN interface. The higher the class of service—and the greater the weight of the higher class—the greater the chance of being processed.

Priority Queuing

Preference for delay-sensitive packets *transmitted* over the DSL port (from the CE200 to the subscriber interface) is provided through the Priority Queuing function. This configuration group applies only to DSL VCs that are created on the CE200, using the Cross-Connect netmodel. The other netmodels—IP, VWAN, and CopperVPN—automatically give these kinds of packets priority in the downstream direction over DSL links. But, in the Cross-Connect netmodel, after you have set up the Permanent Virtual Circuits, you have to use cmSubIface to set each DSL VC for priority queuing: High for real time packets such as voice, low for non-real time packets.

The default is Low level. For a PVC that has voice packets or video conferencing packets and regular IP data packets on it, set the priority level to High. For a PVC with only regular IP data packets on it, leave the priority level at Low. For example:

```
get cmSubIface [1.8.24.16]

Group: cmSubIface
Instance: [1.8.24.16]
PII                 = 1.8.24.16
IfIndex             = 1.8.24.16
Name                =
Priority            = High
RowStatus           = Active
```

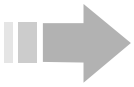
In the example above, priority queuing on the 16th PVC on the 24th port on the board occupying the 8th slot has been set to High priority. To change the default on any of the other PVCs from the default (Low priority), you will have to use cmSubIface and change the defaults one at a time.

Priority queuing, once a PVC has been configured, comes into play whenever real time and non-real time packets are sent from the CE200 over the PVCs on the DSL port. Real Time packets will be processed before non-real time packets.

Radius Authentication

Enhanced system security and management of the CopperEdge user base is available through *Radius authentication*. All user and password databases on all CE200s on the network, including CopperView (SNMP) sessions, can be consolidated in a remote data base administered via one or more remote Radius servers. All login requests for CE200s then go to the Radius server for verification. A separate server supports the Radius accounting function. It keeps a record of user and SNMP access for each CE200.

Putting control at a regional or corporate level reduces chances of unauthorized entry and reduces the complexity of the user-management task. Up to three different server addresses can be specified for both the authentication server and the accounting server, enabling virtually 100 percent reliability.



N O T E S

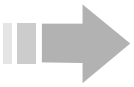
The CopperEdge does not send passwords for SNMP operators (such as Public and Private) in authentication requests.

In the CopperEdge implementation of the Radius Accounting Function, the Radius Attribute Acct-Status-Type, used in the Radius accounting function, always has a value of 7 (Accounting On). Since the CE system does not utilize a shutdown process, the "Accounting Off" packet (Acct-Status-Type=8) is not supported. Instead, the server should be configured to treat Accounting On as signifying the beginning of a new session and the ending of the previous one.

Preparing to Configure CopperEdge for Radius Servers

Here is what's needed to implement Radius in the CE200:

- You must have at least one Radius Authentication server; at least one accounting server is also required to implement the accounting functions
- The IP addresses of your Radius servers must be available through the CopperEdge routing table
- The Authentication Servers must be pre-configured to translate objects that are specific to the CopperEdge products, to object names recognized by the server
- The final step is to enable the cmRadius authentication object on the CopperEdge unit.



N O T E

Once the Radius server is provisioned and cmRadiusAuth is enabled, security features are under control of the Radius server. You can still configure the CopperCraft cmOperatorTable, but its contents will not be used in normal operation (the exception results if LocalFallback is enabled and none of the configured RADIUS servers can be reached). Operators configured in cmOperator and not contained in the Radius server database will not be allowed to log in. Note also that the Radius protocol differs from the CopperCraft controlled authentication in that Radius operator names and passwords are fully case-sensitive.

Configuring the Radius Servers for CopperEdge

Radius servers and their software are available in a number of different versions and implementations. Because setup and configuration procedures may vary, consult the documentation for whichever type you are using for full configuration particulars.

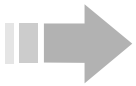
Similarly, while the CopperEdge Radius Client uses all the applicable Radius attributes as documented in RFC-2138 and RFC-2139, certain CopperEdge-specific parameters must first be translated so the server can interpret and deal with them appropriately:

1. Radius Attribute NAS-Identifier may be omitted in the packet if the CopperEdge SystemName object is empty.

2. The value of Radius Attribute *Filter-Id* substitutes for the CopperEdge Context and Privilege attributes (objects). Filter-Id attribute should contain the strings Context=HH (with HH a hexadecimal value) and Privilege=DD (with DD a decimal value) as described below. The syntax of the Filter-Id string is important because the CopperEdge unit must extract the values of HH and DD to properly complete the login process. Two Filter-Id strings are expected in an Access_Accept packet.

   The filter ID values for the *CopperEdge Context* attributes are:

   01 = HTTP (Hex 01)
   02 = SNMP (Hex 02)
   04 = Telnet (Hex 04)
   08 = Serial (Hex 08)
   10 = System (Hex 0a)
   20 = FTP (Hex 14)
   40 = Shell (Hex 28)



N O T E

Valid numeric values in the Context string can generally be combined to enable the same operator record to be used in multiple contexts. For example, decimal 12 (Hex 0c) enables both Telnet (04) and Serial (08) contexts. But to distinguish SNMP sessions from normal operator logins, and to prevent an operator from logging in using only the community string, SNMP user records must specify a Context of SNMP only. If an SNMP user has a login context value indicating anything other than SNMP, authentication will be rejected by the CopperEdge unit even if accepted by the RADIUS server.

   The filter ID values for the *CopperEdge Privilege* attributes are:

   1 = View
   2 = Monitor
   3 = Provision
   4 = Security

3. The Radius *Reply-Message* attribute may be configured with a text string (up to 128 characters) to serve as an error message in case an attempted login is denied (Access Reject). This string accompanies the *Access_Reject* packet, and is entered in the CopperEdge Event Log.

4. The Radius *Reply-Message* attribute may also contain a second 128-character (maximum) text string for use if you want the server to ask the user for additional information or to repeat the previous login attempt. The CopperEdge Radius client will respond to one *Access_Challenge* by resending the

same UserName and Password. If the server does not accept it, the login fails. The string goes with the *Access_Challenge* packet. It appears in the CopperEdge Event Log.

5. If you are configuring a CE200 equipped for Control and WAN redundancy, you must be sure to make the Radius server aware of the presence of the Secondary System Control module by entering its *NAS-IP-Address*, so the server can be ready to process requests from the Secondary in case of a failover.
6. Finally, the Radius server must use the same shared secret (*AuthKey*) string when configuring the both Primary and Secondary SCM clients. We recommend that you use the same AuthKey system-wide, but like a good password, it should be short-lived, unpredictable, and changed at irregular intervals.

The two tables below summarize the server configuration sequence.

To request authentication, the CopperEdge sends an Access-Request packet with the following data to the RADIUS server:

Attribute	Value	Note
User-Name	Text string input by user	Maximum of 32 characters.
User-Password	Text string input by user	Maximum of 32 characters. This field is one-way hashed using the MD5 algorithm. **Note:** The CE does not send passwords for SNMP operators (such as Public and Private) in authentication requests.
NAS-IP-Address	CE-IP-Address	This is the IP-Address of Ethernet port 1.2.1 or 1.15.1, depending on which SCM is being accessed by the operator. This field will contain 0.0.0.0 if there is no IP address configured for that port CE Management Ethernet IP address.
NAS-Identifier	CE-System-Name	This attribute is omitted if CE-System-Name is an empty string.

Upon receiving of the request from CopperEdge, the Radius server may respond with an Access_Accept packet with the following data in it:

Attribute	Value	Note
Idle-Timeout	Idle Timeout value in second (0 to 2147483647)	If this field is empty, the CE uses a predefined system Idle Timeout of 15 minutes.
Filter-Id	"Context = HH" **Note:** Be sure to enter this exactly as shown, with a space before and after the equal sign.	Where HH is a Hex number of the login context for this operator. Possible context is a bit map: 01 = CONTEXT_HTTP; 02 = CONTEXT_SNMP; 04 = CONTEXT_TELNET; 08 = CONTEXT_SERIAL; 10 = CONTEXT_SYSTEM; 20 = CONTEXT_FTP; 40 = CONTEXT_SHELL
Filter-Id	"Privilege = DD" **Note:** Be sure to enter this exactly as shown, with a space before and after the equal sign.	Where DD is a decimal number specified level of privilege designated to the particular operator. A number represents level of privilege: 1 = VIEW; 2 = MONITOR; 3 = PROVISION; 4 = SECURITY

Configuring CopperEdge for Radius Servers

When the Radius server is configured and ready to communicate, and its database has been provisioned with the user data base, the CopperEdge cmRadius MIB group can be configured:

1. With cmRadius in its default state (Authentication= Disabled), configure all of its objects except authentication. *Example:*

   ```
   CRAFT (1.2)> set cmradius authkey=4cr37ei
          authprimaryipaddr=10.122.4.4 authprimaryport=1645
          acctprimaryipaddr=10.122.4.6 acctprimaryport=1646
   ```

   Be sure to use whichever port numbers are recognized by your Radius servers. Although the officially assigned port numbers for Radius Authentication and Accounting are 1812 and 1813, respectively, many current radius servers still use the port numbers of the original RFC: 1645 for authentication, and 1646 for accounting. Be sure that the numbers you assign in the CopperEdge cmRadius configuration match those used by your remote Radius servers.

2. When you are sure the rest of the configuration is complete and correct, activate the Radius capability:

   ```
   CRAFT (1.2)> set cmradius authentication=enabled
   ```

For more data about the cmRadius MIB group, see your *CopperCraft Reference and MIB Definitions* manual.

cmSystem, Managing Your Configuration

The cmSystem group is used to control and manage a number of functions that are key to operating the CopperEdge:

CalendarTime	Set or display the system real time clock (date and time set at initial configuration)
Redundancy	Enables or Disables redundancy functionality. When Enabled, the secondary Control and WAN complex can take over as primary.
Command	The command **`get cmsystem`** responds with the last operator command. Use the command **`set cmsystem`** to save the Config (configuration) file, to manually switch (takeover, relinquish) between two redundancy complexes, and to restart the system.

When you have finished adjusting the system attributes to your preferred settings, save the configuration data:

```
set cmsystem command=save
```

To verify that the save command was successful:

```
get cmsystem

Group: cmSystem
ObjectClass          =  System
```

```
OperState            =  Enabled
Version              =  E 3.0
Master               =  0.0.0.0
ConfigFileName       =  config.tgz
CalendarTime         =  2002/07/21-10:01:11
MyPII                =  1.2.0.0
PrimaryPII           =  1.2.0.0
SecondaryPII         =  0.0.0.0
Redundancy           =  NotAvailable
ShelfCount           =  1
ExpIpSubNet          =  192.168.250.0
ConfigSynch          =  Saved
Command              =  SaveConfig
CommandStatus        =  Succeeded
```

In the resulting cmSystem listing, the CommandStatus field should indicate `Succeeded`. If Command Status shows `InProgress`, repeat the command **`get cmsystem`**; if Command Status shows `Failed`, contact Copper Mountain Tech Support.

The cmSystem readConfig command is not supported in this release. If it becomes necessary to reload a previous configuration, use the cmMaintCmd group's ConfigRestore command. (See "Restoring a Backed Up Configuration" on page 80).

Restarting the System

The cmSystem group can also be used to do a warm restart of the system. A restart will interrupt any processes in progress, and the CE200 will reboot and reload its entire code base. Restart is *service-affecting*, with consequent (temporary) interruption of service for all subscribers connected to this CE200. Except in cases of emergency, restarts should only be attempted during scheduled maintenance periods at off-peak times.

To restart the system (using any of the available control interfaces):

```
set cmsystem command=restart
```

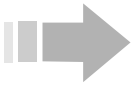
To restart the system if you have Security or System privileges:

```
scmrestart
```

Configuration Backup

Even with a modest number of connected subscribers, the saved file containing the CE200 configuration data can be extensive. Copper Mountain recommends that you periodically back up the file to a remote machine for reference (such as when the System Control Module is replaced) or emergency use.

With an on-line file server available to act as receiver, the CE200 (Version 2.0 and later) can be commanded to back up its current (saved) configuration on demand, or it can be configured to automatically send a Config file backup at specified recurring intervals. Using the cmMaintCmd table and its ConfigBackup command, you can specify the exact time to perform the backup, and how often, if ever, you wish it to recur.



N O T E

If you back up or otherwise gain access to a config.tgz (or config.txt) file, you can view the file with a text editor, but do ***not*** *try to edit it with any offline (non-CE200) application. While it may appear to be in text format, the config file cannot be reliably revised or updated with a text editor. If the file should become corrupted, routing information may be affected, with unintended effects on service.*

Backing Up the Saved Config File

To configure a backup of the config.tgz (or config.txt) file, one or (preferably) two file servers should be available and provisioned with FTP servers to receive the configuration file from the CE200.

The backed up file is always an uncompressed text file (config.txt).

Using the cmMaintCmd table and the configbackup command, configure the CE200 for config backup. Use the following command example as a model for your own configuration.

Full configuration of cmMaintCmd requires setting a fairly lengthy list of parameters. To reduce the required complexity of any single command, you can issue multiple Set commands, provided you begin by specifying the ConfigBackup command and creating the appropriate row in the cmMaintCmd table.

```
set cmmaintcmd [configbackup] rowstatus=createandwait

set cmmaintcmd [configbackup]
       primaryipaddr=206.71.190.4

set cmmaintcmd [configbackup]
       secondaryipaddr=206.71.190.5

set cmmaintcmd [configbackup] directory= /
       ce200config/sys1

set cmmaintcmd [configbackup] basefilename=sysdeltabu

set cmmaintcmd [configbackup] username=anonymous
       pass=""

set cmmaintcmd [configbackup] recurrence=168

set cmmaintcmd [configbackup] start=99/04/01-03:20

set cmmaintcmd [configbackup] rowstatus=active
```

In the above example, the CE200 has both primary and secondary file servers for its FTP transaction, which is set to recur weekly (recurrence=168 hours) at 0320 hours (3:20 a.m.).

If you set the recurrence at 0 and start time at 0 or any time in the past, the command will execute immediately upon row activation.

In a recurring configuration backup, you would specify the time of the first occurrence, and an interval for subsequent file uploads. When configured, the CE200 will send the current configuration saved in its flash. *A recurring backup will only be attempted if this is the first attempt since powerup, or if the config file saved in flash has changed since the previous backup.*

If a fresh backup is needed, the CE200 attempts to transfer the config to the primary file server. If the transfer succeeds, the CE200 logs an informational event, and sends an SNMP trap. If the transfer to the primary fails, the CE200 will also log and trap the event, and will then attempt to save the config to the secondary server. If the transfer to the secondary server fails, the CE200 will log the event as a minor alarm and send the applicable trap.

For more information about the maintenance commands, see the description of cmMaintCmd in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Restoring a Backed Up Configuration

Restoring a configuration is very similar to backing it up, except that the configuration data is flowing back toward the CE200 rather than toward the file server. The ConfigRestore command is also slightly simpler, as there is no need to specify a secondary server or recurrence-related objects.

As is the case with the configbackup command, you can issue multiple Set commands to set up the appropriate row in the cmMaintCmd table.

```
set cmmaintcmd [configrestore] rowstatus=
       notinservice

set cmmaintcmd [configrestore] primaryipaddr=
       206.71.190.4

set cmmaintcmd [configrestore] directory=
       /ce200config/sys1

set cmmaintcmd [configrestore] basefilename=
       config206_71_190_419991101_0000

set cmmaintcmd [configrestore] username= anonymous
       password=""

set cmmaintcmd [configrestore] rowstatus= active
```

The final command in the string will activate the config restore, and the file will be downloaded to the CE200 flash, but will not be loaded into the active system. To complete the reload, you must reset the system:

```
set cmsystem command=restart
```

The restored configuration file is always a compressed binary file.

Chapter 5
Routing, Forwarding, and Link Management

This chapter discusses the CE200's network interfaces and considerations in routing DSL links. It describes ATM and Frame Relay functionality as implemented in the CE200, and provides instructions with examples for configuring circuits and connections for communication over Frame Relay and ATM links, and over the Ethernet when that interface is used for network access.

A reference listing of all of the CE200 MIB objects, including those related to ATM and Frame Relay functionality (configuration, status, and performance monitoring) is contained in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Features

No special commands or objects are needed for configuring either Frame Relay or ATM WAN VCs. All parameters are accessible from the existing CE200 MIB tables.

DS3 Frame Relay Module

The Frame Relay WAN VCs currently support:

- RFC-1973, point-to-point protocol over Frame Relay
- RFC-1483, encapsulation (FUNI)
- RFC-1490, bridged or routed-IP encapsulation
- NNI on WAN Links
- RFC-1315, DLCMI and link (PVC) performance/error monitoring
- Traffic Shaping, with settings for Committed Information Rate (CIR), Committed Burst (CB), and Excess Burst (EB)
- Link Management Interface (LMI), with protocols LMI Rev1, ANSIT1-617-D, and Q9.33-Annex-A
- IP packet filtering on VCs
- Redundancy, with modules in Backup slots 16 and 17 supporting modules in Preferred slots 3 and 4

DS3 ATM Module

The ATM WAN VCs currently support:

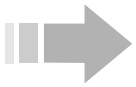
- RFC-1483, multi-protocol encapsulation over ATM
- RFC-1490, multi-protocol encapsulation over Frame
- RFC-2364, point-to-point protocol over ATM
- ATM Permanent Virtual Circuits (Switched Virtual Circuits not supported at this time)
- ATE, functioning as an ATM Host
- Traffic Shaping, with service categories (ubr, nrtVBR, rtVBR) and with settings for Peak Cell Rate (PCR), Sustained Cell Rate (SCR), Maximum Burst Size (MBS), and CDV (Cell Delay Variation)
- OAM Fault Management, with Alarm Indication Signals (AISs) and Remote Defect Indications (RDIs) along with functions for automatic and manual (diagnostic) loopbacks
- virtual connections (no virtual path tunneling
- IP packet filtering on VCs
- Redundancy, with modules in Backup slots 16 and 17 supporting modules in Preferred slots 3 and 4.

DS1 Frame Relay Module (or Quad T1)

The ATM WAN VCs currently support:

- RFC-1973, point-to-point protocol over Frame Relay
- RFC-1483, encapsulation (FUNI)
- RFC-1490, bridged or routed-IP encapsulation
- NNI on WAN Links
- RFC-1315, DLCMI and link (PVC) performance/error monitoring
- Traffic Shaping, with settings for Committed Information Rate (CIR), Committed Bursts (CB), and Excess Bursts (EB)
- Link Management Interface (LMI), with protocols including LMI Rev1, ANSIT1-617-D, and Q9.33-Annex-A
- IP packet filtering on VCs identical to that on DSL links
- Receive timing from a clock recovered from the network line signal; transmit timing from a local/internal clock or a transmit clock recovered from the network.
- ESF framing mode
- Loopbacks, with settings for inward loop, near-end line loop, near-end payload loop, far-end line loop, and far-end payload loop

- DS1 modules in slots 3 and 4 can support redundant DS1 modules in slots 16 and 17. A link down on any T1 port in either slot 3 or slot 4 can trigger a failover to the redundant side.



N O T E

In this release (7.0), redundancy is not supported for T1/E1 IMA Modules (page 97).

Configuring CE200 Network Interfaces

You can review the basic configuration of any CE200 interface by checking the contents of its entry (PII) in the *c*mIface table (see the sample table entry below). Then, use the command, **`set cmiface`**, to specify the interface and configure its various parameters (objects).

In the following example (and in the tables throughout this chapter), shaded objects are read-only; you cannot directly set them. The objects without shading are operator-configurable.

```
Group: cmIfaceTable
Instance: [1.10.1.0]
PII                 =  1.10.1.0
IfIndex             =  1.10.1.0
Name                =  Name of this port
AdditionalInfo      =  ....
NetModel            =  IP
IpAddr              =  192.168.1.1
NetMask             =  255.255.255.0
MacAddr             =  0.60.58.0.0.b
BurnedInMacAddr     =  0.60.58.0.0.b
FarEndAddr          =  0.0.0.0
DestPii             =  1.3.2.32
CMCPCompatible      =  Yes
EncapsulationType   =  rfc1483
FwdMode             =  IP Policy
Pix                 =  54
```

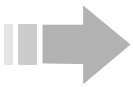
Full IP vs. Policy-Routed Links

As described in the previous chapter, you can configure a DSL link to make the CPE behave as an ordinary IP router. If you configure a DSL link with an IP address, and you also specify a destination PII, the resulting link is said to be *IP-Policy routed* and the status of the fwdMode field in the cmIface table will display `IP Policy`.

Policy routing is a forwarding process that ignores any destination IP address embedded in packets. Instead, packets are sent to the destination PII specified by the configuration of the packet's source PII (the DSL interface). If no destPII is specified, the link is Full-IP routed.

Policy routing allows you to enhance security between DSL links, since traffic arriving on one DSL link cannot be routed out to another DSL link nor to any other IP address, no matter what its contents. The one exception to the Policy Routing scheme occurs when a premise host sends a packet to the CE200's IP address. In that case, the host is talking *to* the CE200 rather than *through* it.

The CE200 receives and processes the packet, but it is not policy routed. Such a setup allows a host on a LAN served by a CPE to ping the DSL interface.



N O T E

In the IP networking model, packets from the WAN interface or any non-DSL interface are always full-IP routed; i.e., the system sends the packet to a DSL link (or elsewhere) based on its destination IP address. Only packets arriving on DSL links can be policy routed.

As noted in previous discussions of the IP networking model, we continue to support it for its utility in configuring temporary or dedicated management links. However, to maximize network security, economy of address-assignment, and efficient use of bandwidth, a flexible aggregation net model, such as the new CopperVPN should be given first consideration.

Configuring Full-IP DSL Links

You configure DSL links for Full-IP routing as described in the previous chapter. In brief, assign the DSL interface an IP address, optional subnet mask, and leave the DestPii object set to zero. (FarEndAddr is normally ignored on DSL links, unless configured for policy routing over Ethernet). Be sure to configure the EncapsulationType, NetModel, and any other mandatory settings, as well. For example:

```
Group: cmIfaceTable
Instance: [1.10.1.0]
PII                 =  1.10.1.0
IfIndex             =  1.10.1.0
Name                =  System-assigned port name
GroupName           =  User-Group ID String
AdditionalInfo      =  User-defined text string
NetModel            =  IP
IpAddr              =  192.168.1.1
NetMask             =  255.255.255.0
MacAddr             =  0.60.58.0.0.b
BurnedInMacAddr     =  0.60.58.0.0.b
FarEndAddr          =  0.0.0.0
DestPii             =  0.0.0.0
CMCPCompatible      =  Yes
EncapsulationType   =  rfc1483
FwdMode             =  Full-IP
Pix                 =  88
```

You can change a policy-routed DSL link to full-IP at any time by setting destPii to `0.0.0.0`. The fwdMode field will display the mode (`Full-IP` or `IP Policy`) based on the setting of the destPii object.

Configuring Policy-Routed DSL Links

DSL links may be policy routed to a virtual circuit on a CE200 WAN port, or to a specific host device on a LAN over the CE200 Ethernet port.

IP Policy Over WAN

To configure a DSL link for policy routing over a WAN link, configure the DSL link (**`set cmiface`**) to specify its IP address, net mask, and a DestPii. The DestPii includes shelf number, slot, port and the virtual identifier (Frame Relay DLCI or ATM virtual circuit number), in the full PII format as shown in the following sample configuration. For example:

```
Group: cmIfaceTable
Instance: [1.10.1.0]
PII                  =  1.10.1.0
IfIndex              =  1.10.1.0
NetModel             =  IP
IpAddr               =  192.168.1.1
NetMask              =  255.255.255.0
MacAddr              =  0.60.58.0.0.b
BurnedInMacAddr      =  0.60.58.0.0.b
FarEndAddr           =  0.0.0.0
DestPii              =  1.3.1.32
CMCPCompatible       =  Yes
EncapsulationType    =  rfc1483
FwdMode              =  IP Policy
Pix                  =  27
```

As shown in this example, policy-routed DSL links *do* have IP addresses and subnet masks, because packets are routed *to* them with full-IP routing.

Packets forwarded in IP-Policy mode are capable of generating the IETF standard `TTL Exceeded` ICMP error message if the packets expire before reaching their destination (V1.6 and later). This capability enables use of standard TraceRoute diagnostic utilities.

Note, however, that because outbound policy-routed packets bypass the CE200 routing table, a TraceRoute sequence cannot identify the CE200 leg of the route. To resolve the hop to and from the CE200, you can add an entry to the routing table that will let the ICMP error message track through and identify the route segment from the WAN VC, through the CE200, to the DSL port/CPE.

IP Policy Over Ethernet

Performance implications when using the Ethernet port of the CE200 as a WAN interface, using the IP Policy netmodel, are not clear. If you establish other network connections over a WAN port (FR, ATM, V.35) along with a connection over the Ethernet port, you may have performance degradation on one link or all of the links. Also, restriction to a single virtual bridge group in VWAN over Ethernet may present a security problem on your network.

Any DSL link can also be policy routed to a specific host on an Ethernet LAN through the CE200 Network (Ethernet) port.

First, if the CE200 Ethernet port has not yet been configured, assign its IP address. Second, set its networking model and encapsulation type parameters. For example:

```
set cmiface [1.2.1] ipaddr=192.168.1.1
        netmask=255.255.255.0 netmodel=ip
        encapsulationtype=none

Set Successful
get cmiface [1.2.1]
```

The response will display all objects in the cmIface group for the specific instance (the Ethernet interface).

```
Group: cmIfaceTable
Instance: [1.2.1]
PII                   =  1.2.1.0
IfIndex               =  1.2.1.0
Name                  =  EthernetPort
AdditionalInfo        =  System Master
NetModel              =  IP
IpAddr                =  192.168.1.1
NetMask               =  255.255.255.0
MacAddr               =  0.80.56.34.da.87
BurnedInMacAddr       =  0.80.56.34.da.87
FarEndAddr            =  0.0.0.0
DestPii               =  0.0.0.0
CMCPCompatible        =  No
EncapsulationType     =  None
FwdMode               =  Full-IP
Pix                   =  1
```

As shown, the CE200 Ethernet port is configured as full IP. To configure the DSL port for policy routing, configure its IP address and netmask, destination PII (the System Control Module Ethernet port, 1.2.1) and far-end address (the IP address of the Ethernet host machine). For example:

```
set cmiface [1.6.1] ipaddr=200.10.10.1
        netmask=255.255.255.252 destpii=1.2.1
        farendaddr=192.168.1.2 netmodel=ip
        encapsulationtype=rfc1483

Set Successful
```

Note that this use of FarEndAddr is very different from its function in configuring WAN virtual circuits. Note also that, if the DestPII is the Ethernet port on the System Control Module, then the FarEndAddr must be a separate IP address, but it must be on the same subnet as the Ethernet Port.

The configured DSL interface should be similar to the following example.

```
Group: cmIfaceTable
Instance: [1.6.1]
PII                 =  1.6.1
IfIndex             =  1.6.1
NetModel            =  IP
IpAddr              =  200.10.10.1
NetMask             =  255.255.255.252
MacAddr             =  0.60.12.34.56.78
BurnedInMacAddr     =  0.60.12.34.56.78
FarEndAddr          =  192.168.1.2
DestPii             =  1.2.1.0
CMCPCompatible      =  Yes
EncapsulationType   =  rfc1483
FwdMode             =  Policy-IP
Pix                 =  14
```

A key difference between policy routing over Ethernet versus policy routing over WAN is that over Ethernet, the far-end address of the DSL interface is configured with the IP address of the Ethernet host. In contrast, over a WAN link, the far-end address (FarEndAddr) is not significant for the DSL link. You don't need to configure it.

Configuring Ports for VWAN Over Ethernet

Configuration of DSL ports and WAN VCs for VWAN is described in Chapter 4. Although the Ethernet port does not support virtual circuits, it must still be configured in order to function properly in the VWAN networking model.

If the CE200 Ethernet port has not yet been configured, explicitly assign its networking model and encapsulation type. If the Ethernet port will be used for system management, you will also need to specify its IP address and mask. For example:

```
set cmiface [1.2.1] ipaddr=192.168.1.1
       netmask=255.255.255.253 netmodel=vwan
       encapsulationtype=none

Set Successful

get cmiface [1.2.1]
```

The response will display all objects in the cmIface group for the specified instance (the Ethernet interface).

```
Group: cmIfaceTable
Instance: [1.2.1.0]
PII                   =  1.2.1.0
IfIndex               =  1.2.1.0
Name                  =  EthernetPort
GroupName             =  Garfield
AdditionalInfo        =  Brighton AP Office
NetModel              =  VWAN
IpAddr                =  192.168.1.1
NetMask               =  255.255.255.0
MacAddr               =  0.80.56.34.da.87
BurnedInMacAddr       =  0.80.56.34.da.87
FarEndAddr            =  0.0.0.0
DestPii               =  0.0.0.0
CMCPCompatible        =  No
EncapsulationType     =  None
FwdMode               =  VWAN bridge
Pix                   =  1
ServiceClass          =  D
```

VWAN over Ethernet functions as a “nailed-up” point-to-point link to a single router, and there can be no other device on the LAN. Note that the Ethernet port, when configured for VWAN, operates in promiscuous mode. Thus, *you cannot specify an IP address for the router*. Rather the CE200 learns the MAC address of the router from the received packet stream.

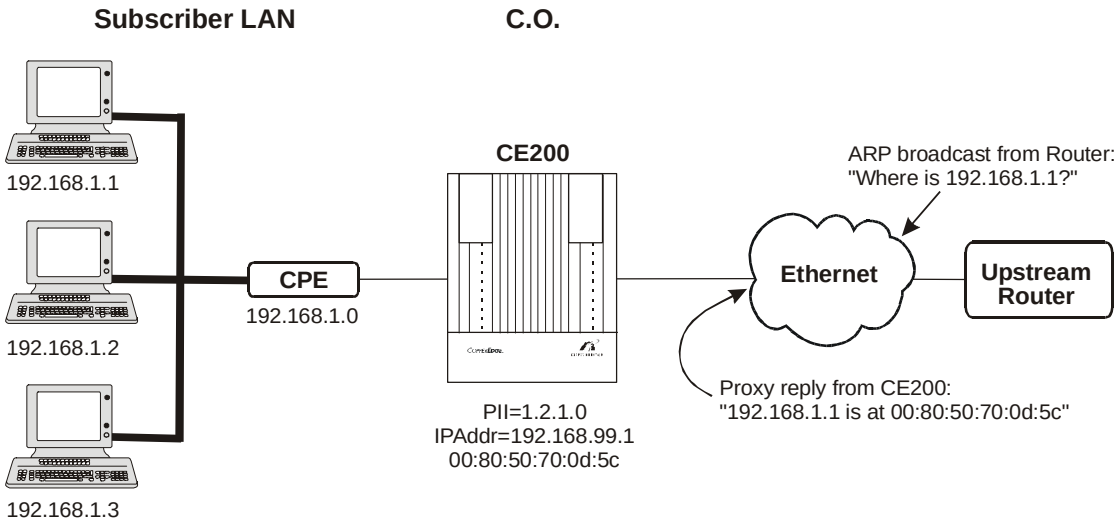
Note also, that the forwarding mode over Ethernet is always displayed as `bridge`, regardless of the number of DSL ports served.

Proxy ARP

For DSL interfaces that are routed over the 10/100BASE-T Ethernet interface, the CE200 supports proxy ARP functionality. The feature simplifies configuration by allowing the CE200, when queried, to use its own Ethernet (MAC) address as a proxy for LAN destinations on its connected CPE.

Proxy ARP for Ethernet is actually the reverse of ARP used in the CopperVPN networking model and described in Chapter 4. In CopperVPN, the CE200 replies to queries from CPEs on LANs as a proxy for the upstream router. In Proxy ARP for Ethernet, the CE200 responds to queries from the upstream device as a proxy for CPEs on LANs.

The diagram below illustrates Proxy ARP for Ethernet. But Proxy ARP is valid only if the DSL port is configured with NetModel=IP and also if the network or WAN connection is through the 10/100BASE-T Ethernet interface on the primary System Control Module. For details on configuring DSL ports for Proxy ARP, see the cmProxyARPTable in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.



Proxy ARP Concept

Configuring WAN VCs

Virtual Circuits, either Frame Relay or ATM, used only as destinations of policy IP routes do not need to be elaborately configured. For those VCs, simply specify the Frame Relay Data Link Circuit Identifier (DLCI) or ATM virtual circuit number associated with the circuit in the destPii of the DSL links. Activate the circuit as described below.

In this mode, the CE200 has no knowledge of IP addresses at the far end of the VC. As a result, the CE200 is fully independent of the far-end IP numbering scheme. In fact, two different VCs can have duplicate, overlapping IP addresses (impossible with full IP routing), and the CE200 will not be affected.

If you wish, you can give IP addresses to the VCs. The CE200 treats each VC as a separate point-to-point (unnumbered) IP link. Unnumbered IP links are ones with no IP subnet number. Each end of an unnumbered IP link may have an IP address, and if both are present, the two addresses can be on different subnets.

However, the preferred way to configure an unnumbered IP link is to assign it a FarEndAddr, but not an IP address. The FarEndAddr tells the CE200 what the IP address of the device at the far end of the point-to-point link is. The WAN port at the near (CE200) end of the link typically has no use for an IP address. See the configuration below:

```
Group: cmIfaceTable
Instance: [1.3.1.32]
PII                  =  1.3.1.32
IfIndex              =  1.3.1.32
Name                 =  ....
AdditionalInfo       =  ....
```

```
GroupName              =  ....
NetModel               =  IP
IpAddr                 =  0.0.0.0
NetMask                =  0.0.0.0
MacAddr                =  ff.ff.ff.ff.ff.ff
BurnedInMacAddr        =  ff.ff.ff.ff.ff.ff
FarEndAddr             =  192.166.101.8
DestPii                =  0.0.0.0
CMCPCompatible         =  No
EncapsulationType      =  rfc1490
FwdMode                =  Full-IP
Pix                    =  141
```

Once the FarEndAddr is defined for the VCs, you can use those IP addresses as the next-hop for route table entries. For more information, see the ipRoute table description in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual. For example:

```
set iproute [192.168.250.0] nexthop=192.166.101.8
```

Currently, when you delete a FarEndAddr (by setting it to 0.0.0.0), you must manually delete any associated routes in the ipRoute table. For example, if you move an IP address from one VC to another, you must first delete, and then recreate any routes using that address. You can delete routes at any time, but you cannot create a new route until *after* the interface IP address is assigned.

You can make a VC both a policy-route destination and a full-IP interface. The two functions are independent of each other.

Configuring DS3 Frame Relay

To establish a Frame Relay virtual circuit on a WAN interface, first, *enable* the WAN port for Frame Relay. Second, specify the LMI scheme, if any. VCs now can be specified and activated.

Initial Configuration

To create a Frame Relay link, you must use two similarly named MIB groups: frDlcmi (from the Frame Relay MIB) and cmFrDlcmi (the Copper Mountain MIB). Both groups configure links on the physical *port*, and thus apply to all of the Frame Relay virtual circuits that you may subsequently assign to that port. For example, to enable a link on a WAN port with no LMI:

```
set cmfrdlcmi [1.3.1] adminstate=enable
```

To provide a link with LMI, use the State object in the standard frDlcmi group to specify the LMI formatting scheme. Available options are: ansiT1-617-D (ansiT1.617 Annex D), lmirev1 (LMI Revision 1), Q9.33-Annex-A, and noLmiConfigured. For example:

```
set frdlcmi [1.3.1] state=ansit1-617-d
```

Always disable the link using cmfrdlcmi (Admin=dis) before changing the LMI state or any other link attributes. Modifying link attributes is service-affecting.

When you enable LMI on a WAN port, remember to configure your Frame Relay router as DCE LMI.

Adding a PVC

Once you have enabled a link as described above, individual WAN PVCs can be configured. Do this using the frCircuit group to specify and activate the circuit numbers. For Frame Relay VCs, you can use any DLCI from 16 to 991.

To establish a VC (DLCI 32, port 1 in the example):

```
set frcircuit [1.3.1.32] state=active
```

Note that the index in this case includes the complete PII entry, that is the interface location (1.3.1) and the PVC number (32).

To disable PVC 32 on port 1:

```
set frcircuit [1.3.1.32] state=inactive
```

To delete PVC 32:

```
set frcircuit [1.3.1.32] state=invalid
```

Before a PVC can be deleted, its link must be disabled through cmFrDlcmi. An active circuit on an enabled link cannot be deleted.

Configure the Frame Relay DCE to Connect with the CE200

The CE200 behaves as a Frame Relay UNI DTE (User Network Interface, Data Terminal Equipment). Therefore, the CE200 must connect to a Frame Relay UNI DCE (Data Communications Equipment) interface, such as a Frame Relay switch. The DCE provides clocking for the CE200 WAN interface.

Throughput Management

For any Frame Relay link (PVC), only a finite amount of data can be accommodated between any two points in time. If the network becomes congested, connected devices at one or both ends of a PVC are automatically notified by the network that the volume of transmitted data will be temporarily retarded until the congestion is alleviated. To ensure a continuous and orderly flow of data, even during these periods of congestion, Frame Relay relies on the concept of a *Committed Information Rate* (*CIR*).

The CE200 supports three configurable command objects related to Committed Information Rate, all through the frCircuit group used to configure Frame Relay virtual circuits (PVCs) established at *WAN* interfaces. They are:

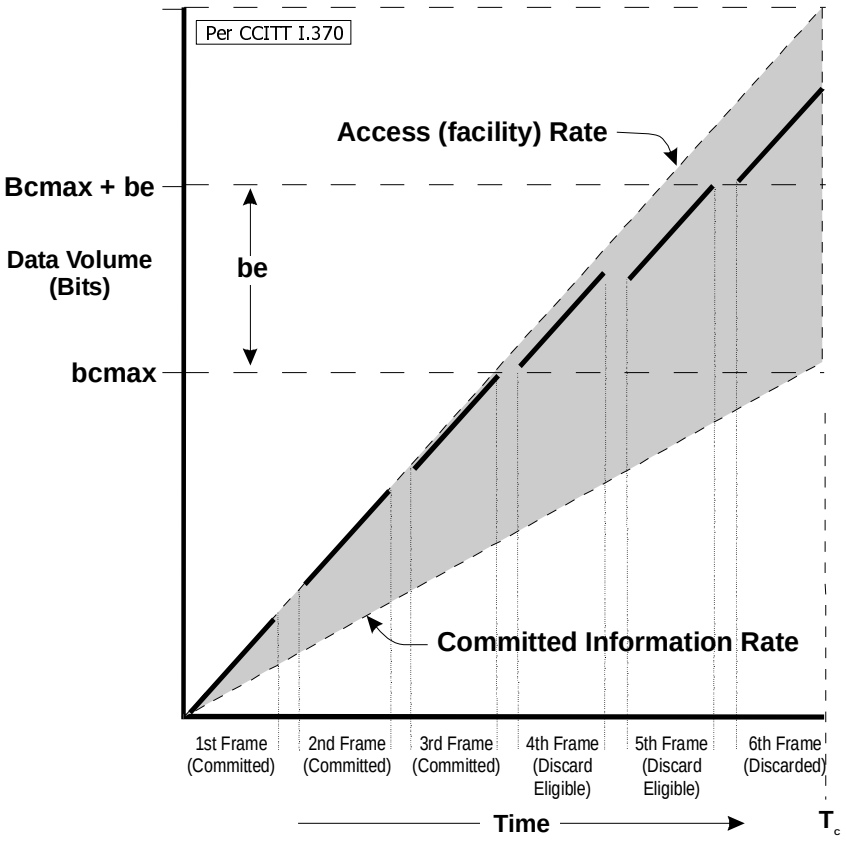
- Throughput
- CommittedBurst
- ExcessBurst

Throughput

Throughput (a term synonymous with CIR) specifies the average amount of data, in bits-per-second, that can be sent over a given PVC (assuming a reasonably well designed network) on demand, 24 hours a day. Since PVCs may be shared by a number of users, the Committed Information Rate for a CE200 PVC will typically approximate some reasonable sum of the data rates of all of the subscribers who have access to the interface.

As a practical matter, however, a user of a real-world data link can almost always transmit and receive at a much faster rate than the nominal baseline indicated by the CIR. As we know, most networked data communication is bursty in nature, that is, it consists of bursts of packets (some of which may be very large), separated by relatively long idle periods. Thus, there is another, typically larger, number that represents a burst of data which the network will accept with a reasonable probability of reliable delivery. This is the CE200 *CommittedBurst (bcMax*).

The third congestion-management setting specifies the amount of data which, though in excess of the Committed Burst, can still be handled by the network and the circuit under normal conditions. This amount is the *ExcessBurst* (*BE).* However, because this ExcessBurst does exceed the commitment, it necessarily is at greater risk of causing or encountering congestion on the link. Frames that fall within the BE range are automatically marked with the *discard-eligible* (*DE*) bit, which means that if the network becomes congested, the frame will simply be dropped. Data within the commitment time interval that exceed the value of BE will be discarded, regardless of whether the link is congested or not. The following illustration, adopted from the CCITT standard on congestion management, shows the relationship between the three variables within the commitment interval (T_c).



Throughput Management Variables

A definition for the commitment interval has been established by CCITT and adopted by ANSI. The actual value will depend on the settings of *CIR*, *bcMax* and *be*. For details on calculating T_c, refer to the Throughput object in the frCircuitTable in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Note that while the values of bcMax and BE remain constant over time, the values of CIR and Access Rate appear as slopes. This is because bcMax and BE are expressed as absolute values (fixed number of bits transmitted over the commitment-interval period), while CIR and Access Rate are expressed in bits per second.

Setting CIR Parameters for Frame Relay PVCs

By default, the CIR setting on newly created WAN PVCs is zero. The default value of bcMax is also zero. In this case, ExcessBurst automatically assumes the value of the Speed object in the IfTable. Thus, Speed in this instance may be considered as synonymous with Access Rate as shown in the preceding illustration.

Congestion management parameters (CIR, bcMax, and BE) are not configurable for DSL VCs. Since there is no enforcement of CIR on the receive side of WAN VCs, bandwidth commitments cannot be assigned or honored on the transmit side of the corresponding DSL VCs.

With the factory default settings, every frame is technically discard-eligible. But, as frames will not normally be discarded unless the Access Rate is exceeded, the default scenario represents the most conservative situation.

If you decide to configure the congestion-management parameters to a value other than the default, then we recommend that you first agree upon appropriate values for CIR, bcMax and BE with your network provider.

To set the information commitment parameters, all three of the relevant objects *must* be set as part of the same command to ensure that all three are simultaneously applied. For example:

```
set frcircuit [1.3.1.101] Throughput=n1
       committedburst=n2 ExcessBurst=n3
```

Where:

n1 is the Throughput value in bits per second

n2 is the size of the maximum committed burst in bits

n3 is the size of the ExcessBurst credit in bits

For more information about throughput management and the tools available to monitor congestion on your Frame Relay links, see frCircuitTable in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Configuring DS3 ATM

The CE200 provides concentration of DSL links originating at the Customer Premises onto DS3 ATM trunks. If you have configured ATM for other types of equipment, you will likely notice that the ATM MIB used here comprises only a small subset of the full MIB. Since the CE200 is an ATM host and not a switch, configuration is relatively simple. In fact, only the atmInterfaceConfTable was extracted from the ATM MIB. Two tables were also added to the proprietary cm MIB, cmAtmVclTable and cmDS3Atm, to support the CE200's implementation of ATM.

The current release will support up to 976 ATM VCs and up to 16 Virtual Paths per DS3 ATM module. The CE200 software supports RFC-1483 encapsulated bridge and IP traffic, PPP over ATM, and an implementation of FRF.8 and FRF.5.

The DS3 ATM Module has the same loopback options (LineLoop, InwardLoop, NoLoopback) and performance parameters (through the dsx3 MIB objects) as the DS3 Frame Module, except that dsx3 table listings for PESs, PSESs, and PCVs will always be zero, as those objects apply only when the LineType is M23. The M23 line type is not supported by the CE200's DS3 ATM Modules.

Configuring ATM VCs

To configure a WAN link for ATM, the appropriate DS3 ATM WAN module must be installed and operating. The DS3 link must also be set to use your selected timing source (local or loop).

Example:

```
set dsx3config [1.3.1] transmitclocksource=loop
```

Next, use cmDS3ATM to configure scrambling and cell mapping. For example:

```
set cmds3atm [1.3.1] cellpayloadscrambling=enabled
       cellmapping=plcp
```

Configuration of the ATM interface and its circuits can then proceed as follows:

1. Configure the physical interface. For example*:*

   ```
   CRAFT (1.2)> Set cmiface [1.3.1] netmodel=none
          encapsulationtype=rfc1483
   ```

2. Create and configure the ATM VCL. For example*:*

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.1.66] rowstatus=create vpi=0
          vci=55 adminstatus=up
   ```

3. Configure the protocol interface (*cmIface*) for the ATM VC. For example*:*

   ```
   CRAFT (1.2)> set cmiface [1.3.1.66] netmodel=ip
          farendaddr=192.166.100.1 encapsulationtype=rfc1483
   ```

4. The above example configures an ATM VC for the IP netmodel.

If the DS3 ATM physical port is disabled, the port does not go idle, but transmits AIS signals.

For more information about the ATM-related objects in these examples and their configuration options, see the MIB Definitions in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Configuring ATM VCs for Quality of Service

The CE200 supports ATM quality of service (QoS) for all cells transmitted upstream over an ATM WAN VC. Different flavors of voice, full-range video, and video conferencing can be accommodated. The CE200 allows you to configure QoS using either the CopperView EM or the CopperCraft CLI. You must use three different MIB groups or tables:

- cmCircuitParam
- cmATMVcl
- cmParamSummary

You can create as many as 255 different subclasses of service using the cmCircuitParam group. You select either real time Variable Bit Rate (rtVBR) or non-real time Variable Bit Rate (nrtVBR), and then configure the subclasses with different sizes for Peak Cell Rate (PCR), Sustained Cell Rate (SCR), and Maximum

Burst Size (MBS). You also can configure Unspecified Bit Rate (UBR).

Afterward, you can apply the different subclasses (or the different values for PCR, SCR, and MBS) to the different WAN VCs by using the cmATMVcl group.

Meanwhile, you can manage the WAN VCs and different subclasses of service by monitoring the total bandwidth assigned to ports using the cmParamSummary group. You can also adjust upward or downward subscription factors on the ports, allowing as much as 2500 percent oversubscription.

For more information about the MIB groups for applying QoS over ATM, see the MIB Definitions in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

T1/E1 IMA

If your CopperEdge system is equipped with a T1/E1 IMA WAN Module, you can use Inverse Multiplexing ATM to take advantage of disparities in cost and/or availability of bandwidth for your network (WAN) facilities. The cost of a given amount of dedicated bandwidth may be significantly less if it comes in the form of a bundle of T1 or E1 lines, vs. a single higher speed facility. With inverse multiplexing, up to eight T1 or E1 links can be used to carry the combined bandwidth of a virtual high-speed ATM facility whose rate is approximately the sum of the link rates. This is referred to as an IMA group.

IMA Overview

The T1/E1 IMA WAN Module provides eight network physical ports, which you can configure in software as either T1 or E1. These eight ports, as a group, serve as a single virtual/logical port which will always have the port number of 41. Hence, only one group can be supported per T1/E1 IMA Module. DSL interfaces that are to be part of the group are configured with their DESTPII as the PII of the IMA group, for example, 1.3.41.101. Note that as an ATM interface, the IMA group also takes a VC number.

IMA groups terminate at each end of the IMA virtual link. In the transmit direction, the ATM cell stream received from the ATM layer is distributed on a cell by cell basis, across the multiple links within the IMA group. At the far-end, the receiving IMA unit (typically an IMA-capable router) recombines the cells from each link, on a cell by cell basis, recreating the original ATM cell stream. The aggregate cell stream is then passed to the ATM layer.

The IMA interface periodically transmits special cells that contain information that permit reconstruction of the ATM cell stream at the receiving end of the IMA virtual link. The receiver end reconstructs the ATM cell stream after accounting for the link differential delays, smoothing CDV introduced by the control cells, etc. These cells, defined as IMA Control Protocol (ICP) cells, define an IMA frame.

IMA Configuration

To configure an IMA group, you must configure the T1/E1 component of the connection via the *dsx1Config* and *cmDsx1Config* groups, establish the group itself, set the required ATM parameters, and configure the individual (DSL) interfaces involved. The IMA MIB consists of four groups:

- imaGroupMappingTable
- imaGroupNumber
- imaGroupTable
- imaLinkTable

The following steps provide an example of the configuration sequence for an IMA group:

1. Configure the port for T1.

   ```
   CRAFT (1.2)> set dsx1ConfigTable [1.3.1] linetype=ESF
   ```

   Note that this operation has secondary effects. Once you set any port on a module for LineType=ESF, all other ports on the same IMA module will be set to the same line type, and the LineCoding for all ports will be set to B8zs. This single action configures the module as a T1 device (versus E1).

2. Set the line build out based on the application. If ShortHaul, set *linelength* only; if longhaul, you must also set the *LongHaulTxAttenuation*.

   ```
   CRAFT (1.2)> set cmDsx1ConfigTable [1.3.1] LineLength=longhaul
          LongHaulTxAttenuation=-7.5dB
   ```

3. If this port will be configured for E1, enter:

   ```
   CRAFT (1.2)> set dsx1ConfigTable [1.3.1] LineType=E1CRC
   ```

   Just as with the T1 configuration in step 1, the E1 configuration also has secondary effects. Once you set any port on a module for LineType=E1CRC, all other ports on the same IMA module will be set to the same line type, and the LineCoding for all ports will be set to HDB3. This configures the module as an E1 device. No buildout setting is required for E1 ports.

4. Create an IMA group.

   ```
   CRAFT (1.2)> set imaGroupTable ImaGroupIndex=1.3.41
   ```

5. Create a link and add it to the IMA group.

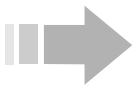
   ```
   CRAFT (1.2)> set imaLinkTable [1.3.1] impGroupIndex=1.3.41
   ```

6. Create a VC on the IMA group.

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.41.101] vpi=12 vci=222 row=create
          admin=up
   ```

7. Configure DSL ports for transport via the IMA group.

   ```
   CRAFT (1.2)> set cmiface [1.6.1] netmodel=ip destpii=1.3.41.101
   ```



N O T E

1. *In the current release, only one IMA group can be supported on a T1/E1 IMA Module. You can configure the module's network port as a single T1 interface, in which case the you would not create an IMA group; rather port 1 would become a dedicated T1 port, and ports 2 through 8 would remain inactive for as long as the single-T1 configuration persisted.*

2. *Although an IMA group may be an aggregation of eight separate T1 or E1 links, the composite bandwidth available may vary if one or more of the T1/E1 lines goes out of service or is restored after having been unavailable. If you have allocated the entire composite bandwidth, and something happens to disable even one of the T1 links, there will be traffic congestion, and packet loss will ensue. A well-founded bandwidth management plan should take note of the reliability of transport facilities.*

8. Typical configurations for imaGroupTable and imaLinkTable are presented in the following examples.

```
CRAFT (1.2)> get imaGroupTable [1.3.41]
    Group: imaGroup
    Index= 1
    RowStatus= Active
    IfIndex= 1.3.41
    NeState= operational
    FeState= operational
    FailureStatus= noFailure
    Symmetry= symmetricOperation
    MinNumTxLinks= 1
    MinNumRxLinks= 1
    NeTxClkMode= ctc
    FeTxClkMode= ctc
    TxTimingRefLink= 1.3.1
    RxTimingRefLink= 1.3.1
    LastChange= 10/21/00  16:54:45
    TxImaId= 1
    RxImaId= 12
    TxFrameLength= 128
    RxFrameLength= 128
    DiffDelayMax= 25
    LeastDelayLink= 1.3.1
    DiffDelayMaxObs= 14
    AlphaValue= 2
    BetaValue= 2
    GammaValue= 1
    RunningSecs= 76491
    UnavailSecs= 34
    NeNumFailures= 0
    FeNumFailures= 0
    TxAvailCellRate= 7244
    RxAvailCellRate= 7244
    NumTxCfgLinks= 2
    NumRxCfgLinks= 2
    NumTxActLinks= 2
    NumRxActLinks= 2
    TestLinkIfIndex= 0
    TestPattern= -1
    TestProcStatus= disabled
    ValidIntervals= 65
    InvalidIntervals= 0
    TimeElapsed= 165
    TxOamLabelValue= 1
    RxOamLabelValue= 1

CRAFT (1.2)> get imaLinkTable [1.3.1]
    Group:  imaLink
    IfIndex= 1.3.1
    RowStatus= Active
    GroupIndex= 1.3.41
    NeTxState= active
    NeRxState= active
    FeTxState= active
    FeRxState= active
    NeRxFailureStatus= noFailure
    FeRxFailureStatus= noFailure
    TxLid= 1
    RxLid= 1
    RelDelay= 9
    ImaViolations= 0
    OifAnomalies= 0
    NeSevErroredSecs= 0
    FeSevErroredSecs= 0
    NeUnavailSecs= 0
    FeUnavailSecs= 0
    NeTxUnusableSecs= 2
    NeRxUnusableSecs= 2
    FeTxUnusableSecs= 2
    FeRxUnusableSecs= 2
    NeTxNumFailures= 0
    NeRxNumFailures= 0
    FeTxNumFailures= 0
    FeRxNumFailures= 0
    imaLinkTxStuffs= 2
    imaLinkRxStuffs= 0
```

Configuring OC-3c/STM-1 Links

Configuration of an OC-3c/STM-1 interface is virtually the same as for any other ATM WAN interface, including netmodels, encapsulations, etc. The only configuration specific to the OC-3c/STM-1 have to do with matching the WAN module options to the transport facility:

sonetMediumTable — This MIB group includes commands to specify the *Type* of signal supported at the interface, either SONET or SDH; the *circuitIdentifier*, a string supplied by the transmission vendor; and the *loopbackConfiguration*, as required for test purposes.

cmSonetSdhTable — This group, part of the Copper Mountain proprietary MIB, allows you to specify the ClockSource to be used by the module, the operating wavelength of the WAN module, and whether or not to enable *cellPayloadScrambling*.

cmAlarmTable — Now includes five sets of variables for setting Threshold Crossing Alerts related to performance of the SONET link.

cmCircuitParamTable — Has been modified to add a new value (*oc3stm1*) to specify the WAN card type for the *PhyType* object.

Besides the individual objects mentioned above the following MIB groups have been added, which allow you to query the status, monitor performance, and collect statistics from your SONET links:

cmSonetSdhTable	SonetMediumTable
SonetFarEndLineCurrentTable	SonetLineCurrentTable
SonetFarEndLineIntervalTable	SonetLineIntervalTable
SonetFarEndPathCurrentTable	SonetPathCurrentTable
SonetFarEndPathIntervalTable	SonetPathIntervalTable
SonetSectionCurrentTable	SonetSectionIntervalTable
SonetSESThreshSetTable	

For more information on the SONET/SDH MIB groups, their objects and options, see your *CopperCraft Reference and MIB Definitions* volume, Chapter 3.

Configuring Quad T1 Frame Relay

The Quad T1 Frame Relay module, unlike the DS3 module for either Frame Relay or ATM, does not require the installation of software to run it. The software already resides on the System Control module in the CopperEdge.

The Quad T1 Frame Relay module is very similar to a DS3 Frame Relay module. The Quad T1 Frame Relay module has the same features and functions as a DS3 Frame Relay module, but the Quad T1 Frame Relay module has significantly lower throughput rates. Like the DS3 Frame Relay module, The Quad T1 Frame Relay module will allow the configuration of VCs and the shaping of traffic using CIR on the VCs.

Performance Monitoring

Many interfaces and operation modes collect summary packet-counts useful in monitoring and evaluating link and interface performance. In most cases the counts are cumulative since the system was last booted. Counters can only be reset by resetting the CE200. For complete information on available performance monitoring groups and their objects, see the specific tables in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

IP Monitoring with IfTable group

The IfTable is a MIB-II standard table where information on IP interfaces and their data performance is collected and stored. The IfTable includes *only* IP interfaces. Therefore, Frame Relay PVCs used only as policy-routed destinations do not appear in the table. PVCs that have IP addresses on either end appear in the IfTable, but *only IP-routed packets are counted; policy-routed packets are not counted in the IfTable*.

VC Monitoring with frCircuitTable

The frCircuit group is a standard Frame Relay MIB table with configuration and performance information about the configured PVCs. Information in this table is specific to Frame Relay and does not include data on IP addresses and routing.

Monitoring Frame Relay Errors with frErrTable

The frErrTable, also derived from RFC-1315, is a standard table of information about the most recent errored frame detected over a specified Frame Relay interface.

Disabling/Deprovisioning an Interface

To temporarily take any interface or port out of service, set the IfTable AdminStatus to Down. To restore service at the same level and to the same users, enter this command:

```
set iftable [pii] adminstatus=up
```

To completely *deprovision* a port (that is, remove it as an active interface or prepare it for a fresh configuration), set the cmIface table entry for the interface to NetModel=None, and clear the Name, Group, and AdditionalInfo objects.

Chapter 6
Voice over DSL

This chapter describes the procedures used to plan and configure networks for voice over SDSL and T1 line modules. It discusses the configuration of DSL and WAN interfaces using different netmodels along with, in the case of Integrated Access Devices (IADs), configuration of both voice and data subnets on a single port.

An IAD serving both voice and data circuits, in contrast to a CPE serving only data circuits, requires a dual netmodel and dual pathways over the IAD and the CE200.

Overview of VoDSL

In delivering digitized voice over SDSL or T1, the CE200 is able to work with a series of different IADs on local loops, and a series of different voice gateways on DS-1s, DS-3s, and OC-3s. At present, the CE200 is able to interface with voice gateways from a number of other manufacturers.

Signaling Protocols

Two signaling protocols for call setup and for voice communications over the CE200 exist: GR-303 and Media Gateway Control Protocol (MGCP). Some IADs are able to work only with the GR-303 protocol; others are able to work only the MGCP standard.

A few IADs, such as Copper Mountain's CR408, CR508, and CR508T, are able to work with gateways using either the GR-303 or the MGCP standard. But, if the CopperMountain IADs have code compatible with GR-303, you must download new code to make them compatible with MGCP, and vice versa.

Service Architecture

Two types of service architecture for carrying call setup messages and voice packets exist: ATM and IP. Some voice gateways as well as the IADs capable of working with them use an ATM AAL2-based architecture. In contrast, other voice gateways as well as the IADs capable of working with them use an IP-based architecture.

Dual Pathways on a Port

All IADs provide data and voice access to upstream networks for users downstream. But, although the data streams and voice streams go over the same DSL line to and from the CE200, the data and voice packets are separated into two VCs on DSL lines and on DS1 and DS3 lines. Two pathways on the WAN interface must be created. Two paths over the CE200 are necessary because the voice packets must go to their own voice gateway and the data packets must go to their own data gateway. The voice gateway performs call setup and routes voice packets; the data gateway reads header information and, in addition, routes data packets.

Dual Netmodels on a Port

To accommodate the dual pathways over DSL ports and WAN ports on the CE200, dual netmodels are necessary. For example, on data VCs on DSL ports, you can use any of four netmodels:

- IP
- VWAN
- CopperVPN
- Cross-Connect

Voice VCs on DSL ports, which must be assigned VC 22, can use either of two netmodels:

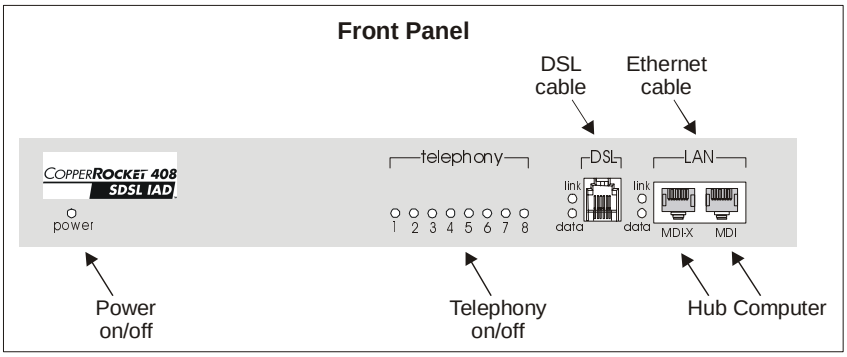
- Cross-Connect
- CopperVPN

To configure data VCs on DSL ports for the IP, VWAN, CopperVPN, and Cross-Connect netmodels, the normal configuration procedures apply. Since you must set the netmodel on the port, you do not need a VC. The system uses a default of 528.

Dual Pathways on an IAD

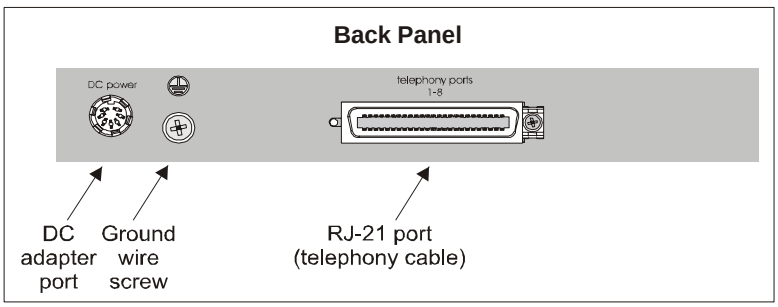
All IADs, provide two types of access. They permit voice and data VCs on a single SDSL line, running at speeds ranging from 128 kbps to 1.5 Mbps. For sending and receiving voice packets, Copper Mountain's *CopperRocket* IADs are compatible with both GR-303 gateways and other types of gateways. In this chapter, we use the *CopperRocket* (CR) 408 as an example of an IAD.

The front panel of a CR408 has connectors for a DSL twisted pair and an Ethernet cable. With the 10/100BaseT connector, users on a LAN can access the data subnet on the CE200.



Front Panel of the CR408

The rear panel of a CR408 has an RJ-21 connector. With the RJ-21 connector, up to eight phone lines at a time can be connected to a DSL physical port via the voice subnet on the CE200. In addition, for emergency service, a CR408 has a LifeLine feature that allows two regular phone lines to be attached to its RJ-21 connector.



Back Panel of the CR408

CopperVPN (Plus) Netmodel

To manage separate voice and data circuits on DSL ports and WAN ports and to allow economical use of IP addressing, the CE200 provides an enhanced netmodel, called CopperVPN (CVPN). CopperVPN functions as a so-called “dual netmodel” in that it allows both voice and data channels on the same physical port.

With CVPN, you can have a router and hosts on different premises but on the same subnet.

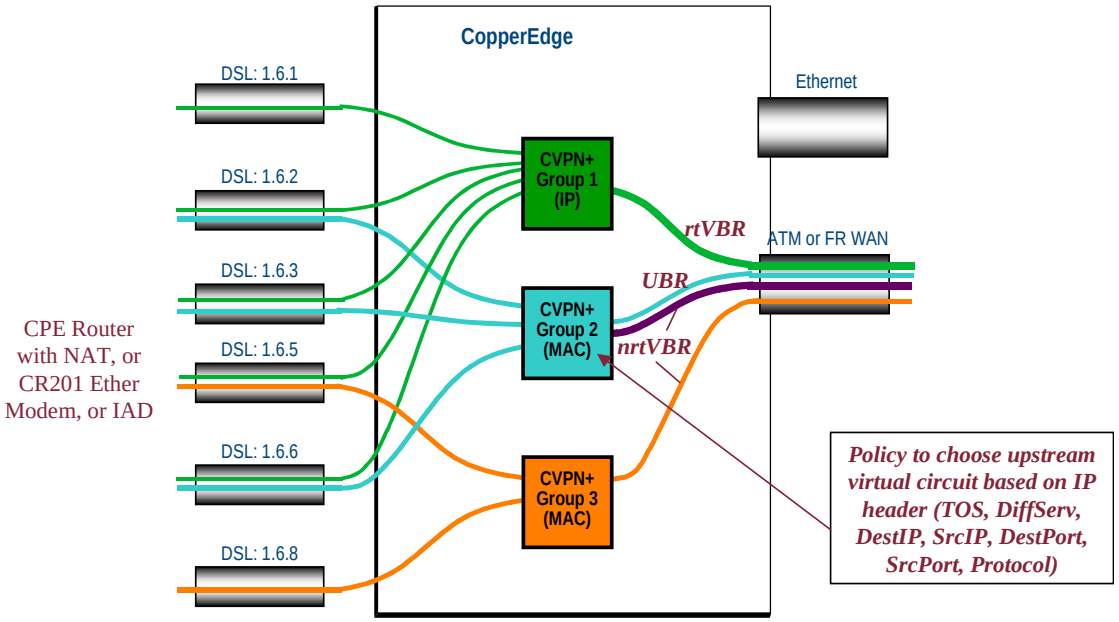
To support the broadest use of the CVPN netmodel for both data and voice, CVPN supports both Ethernet (rfc1483 and rfc1490) and IP (IP-1490) encapsulations. In general, the Ethernet (MAC) encapsulation is for data traffic, and IP encapsulation is for voice IADs as in the former HDIA netmodel. Configuration of CopperVPN interfaces was tested using all types of available IAD, with verified support for the following encapsulations:

- *DSL data interface* (DSL port Pii): rfc1483(MAC), rfc1490(MAC)

- *DSL voice interface* (VCID 22): ip1490(IP), rfc1483(MAC)
- *Ethernet*: None

Note that under IP encapsulation, there is no address translation nor ARP. All entries in the Address Translation/Forward table are either manually entered or automatically learned by DHCP snooping. The WAN-side encapsulation is IP frame in either rfc1483 or rfc1490.

The diagram below illustrates some forwarding examples of CVPN as used in conjunction with dual (voice/data) netmodels. In this example, Group 1, a Voice-over-IP service, uses an IP encapsulation on the DSL, and is associated with an rtVBR ATM WAN virtual circuit. Group 2 is for an ISP. There are two upstream virtual circuits, a best-effort UBR VC, and a guaranteed nrtVBR VC. Group 3 is for another ISP.



Dual NetModels with CopperVPN

Chapter 7
ADSL: G.lite and G.dmt

This chapter describes the procedures used to plan and configure networks for data over ADSL modules, both G.lite and full-rate G.dmt. It discusses the configuration of DSL and WAN interfaces using different netmodels along with the setup and configuration of CPEs in router or in bridge mode.

The procedures include setting up virtual circuits on both the WAN and DSL interfaces of the CE200 with VPIs and VCIs. They do not include setting up virtual circuits on Telco routers on the WAN side of the network, or on CPE routers on the DSL side of the network.

Overview of G.dmt and G.lite

Both the G.dmt and G.lite modules have 24 ports. For G.lite the supported rates for downstream data on each physical port are 64 kbps to 2.336 Mbps. On the upstream, each of the ports supports multiples of 32 kbps from 32 kbps to 512 kbps.

In the downstream direction, each of the 24 G.dmt ports supports data rates in multiples of 32 kbps from 32 kbps to 6.144 Mbps. In the upstream direction, each of the ports supports rates in multiples of 32 kbps from 32 kbps to 640 kbps.

Configuration of ADSL ports can be done either by the CopperView EM or by the CopperCraft command line interface. Using the *cmAdslModem* group, you can set parameters for upstream and downstream signal-to-noise margins and upstream and downstream receive and transmit rates.

CPEs Supporting G.dmt and G.lite

Both the G.dmt and G.lite modules are compatible with a variety of CPEs, including models from Alcatel, Lucent, Motorola, 3Com, Intel, Cisco, FlowPoint, Cayman, Efficient, and Westell. The G.dmt and G.lite modules are also capable of communicating with CPEs in either router or bridge mode. However, some CPEs will run only in bridge mode.

Netmodels Supporting G.dmt and G.lite

Both the G.dmt and G.lite modules are compatible with all of the current netmodels in the CE200, including IP, VWAN, CopperVPN, and Cross-Connect. However, the LAN extension netmodels (IP, VWAN, and CopperVPN) require the CPEs on the DSL links to be in bridge mode, not router mode. None of the CPEs is CMCP compatible; they must have the CMCPCompatible object in the cmIface group set to No.

Encapsulations Supporting G.dmt and G.lite

Both the G.dmt and G.lite modules support RFC-1483 for DSL port encapsulation. They also support RFC-2364 for DSL port encapsulation, but the CE200 allows only RFC-2364 encapsulation with VC-VC forwarding on the DS3 ATM WAN port.

When using RFC-2364 for DSL line encapsulation, the CE200 will not allow RFC-1973 encapsulation with PPP Translation forwarding on the DS3 ATM WAN port. It also will not allow RFC-2364 encapsulation with PPP-Transparent forwarding or with PPP-Translation forwarding on the DS3 ATM WAN port.

In addition, when using FRF.8, the CE200 will not allow a translation from RFC-1483 for DSL port encapsulation to an RFC-1490 encapsulation on the DS3 Frame Relay port.

VCs Supporting G.dmt and G.lite

Both the G.dmt and G.lite modules support multiple VCs on their DSL ports. The CE200 will allow you to map the VCs on the DSL ports to multiple VCs on its WAN VCs. The WAN VCs can be on DS3 ATM boards, DS3 Frame Relay boards, or DS1 boards. The number of VCs available are:

- On G.dmt and G.lite ports, the CE200 will support as many as eight VCs. In addition, the VPIs and VCIs on each VC on the ADSL ports must match the VPIs and VCIs on the CPE.
- On the WAN ports, the CE200 will support as many as 976 VCs. In addition, the VPIs and VCIs on each VC on an ATM board must match the VPIs and VCIs on the upstream router.

Planning the G.dmt and G.lite Network

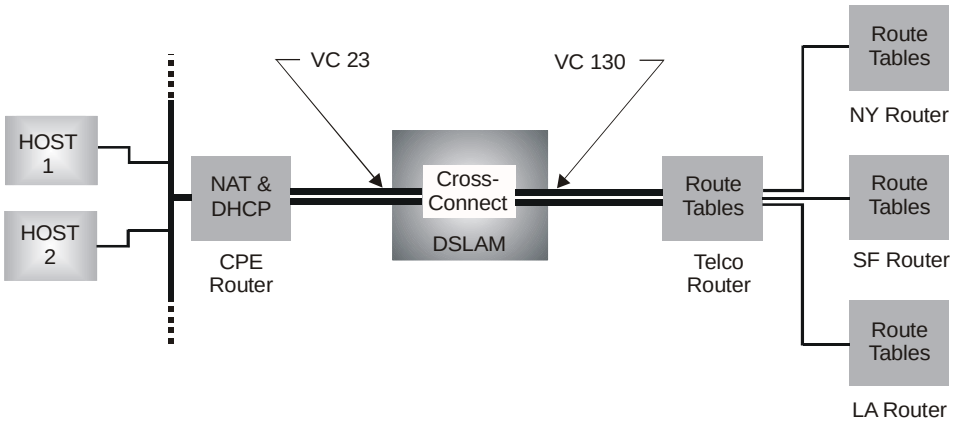
Although the ADSL modules (both G.dmt and G.lite) support all netmodels, we will discuss only two of them: Cross-Connect in Point-to-Point Protocol (PPP) and Full IP. The two netmodels have different capabilities and different requirements for configuring CE200s and CPEs.

Cross-Connect Netmodel and Point-to-Point Protocol over ATM

If you have a LAN and a series of hosts downstream from your CPE and if you have it configured in bridge mode, the Cross-Connect netmodel can be set up with either public or private IP addresses for the hosts. In bridge mode, the CPE treats the hosts on the LAN

as extensions of a subnet configured at the other end of the PPP connection. But, if you have a LAN and hosts downstream from your CPE and if you have it configured in router mode, the setup is different. A CPE in router mode can function as a simple or a complex router, using a DHCP function and a Network Address Translation (NAT) function.

In simple router mode, the setup typically requires a public IP address for the CPE and the hosts. In complex router mode, the setup typically requires an IP public address for the CPE. Using a DHCP function, the CPE then dynamically assigns private IP addresses to hosts on its LAN. When the hosts send messages upstream, the CPE sets up a table, recording all source and destination addresses. Then, before the CPE forwards packets upstream to a destination, it substitutes its own public address for the hosts' private addresses. In response, when messages come downstream for hosts on the LAN, the CPE consults its routing table for source and destination addresses. Before it forwards packets downstream, it substitutes the hosts' private addresses for its own public address.



Cross-Connect Netmodel and Point-to-Point Protocol

IP Netmodel

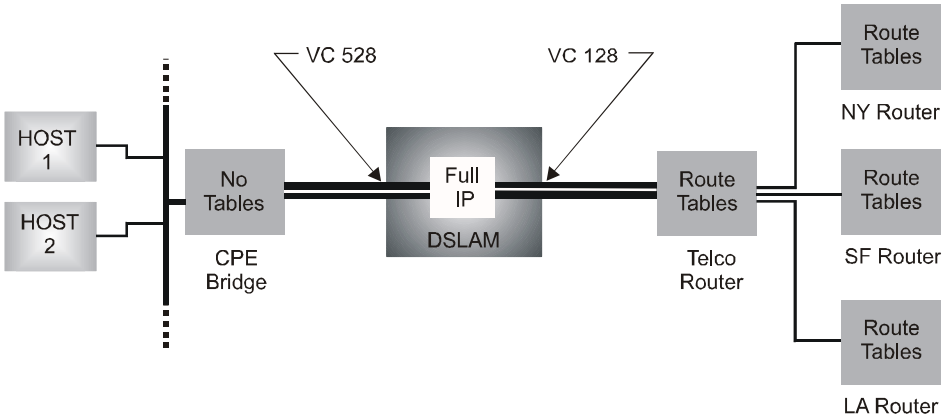
If you have a LAN and a series of hosts downstream from your CPE and if you have it configured in router mode, the IP netmodel can be set up with public IP addresses or private IP addresses for the hosts. In router mode, the CPE treats all of the hosts on the LAN as extensions of a subnet configured on the CE200.

But, if you have a LAN and a series of hosts downstream from your CPE and if you have it configured in bridge mode, the setup is different. A CPE in bridge mode also can be set up with either public or private IP addresses for hosts.

In bridge mode, hosts on a LAN use a DSL port on the CE200 as a gateway. Once packets arrive at a DSL port, the CE200 looks up its default IP route and sends them to a WAN VC. It forwards them over a Point-to-Point link to a Telco router. If the packets have been sent by hosts with public addresses, the router sends them

out over the Internet to a destination address with their source address embedded.

If the packets have been sent by hosts with private addresses, the Telco router uses its NAT function and replaces the private addresses with its own public address. Then the router sends the packets out over the Internet to their destination with its public address (or source address) embedded. In response, when messages come downstream for hosts on the LAN, the router looks through its routing tables for the private addresses (or destination addresses). Before the router forwards the packets downstream, it replaces its own public address with the appropriate private addresses for the hosts.



IP Netmodel

Configuring the G.dmt and G.lite Data Network

The procedure for configuring the CE200 and CPEs for a Cross-Connect netmodel with PPP is different from the procedure for configuring them for a full IP netmodel. But, in both cases, it is best to start with the WAN VCs and their upstream routers. Then, work your way down to the DSL ports and link them to the WAN VCs.

Finally, configure the CPE routers or CPE bridges and the hosts connected to them. The procedures for configuring the CPE routers or CPE bridges and the attached hosts, however, will vary according to the type and brand of CPE router or bridge and the type of operating system running on the host. But several basic steps are always present.

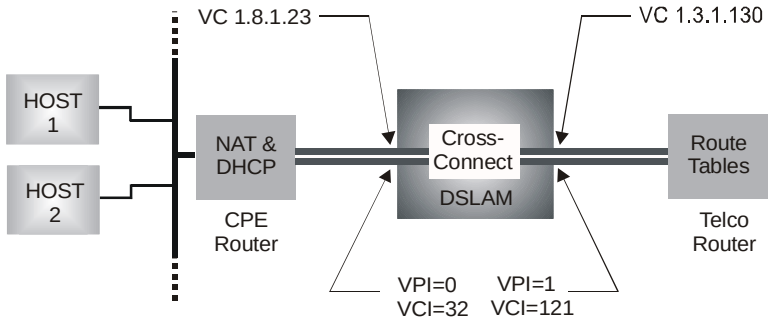
Cross-Connect Netmodel

In the Cross-Connect netmodel with PPP, the end points of the VCs must be manually configured for each stage in the link. (See the diagram below.)

VCs on the Telco router going to the CE200 must be mapped to VCs on the WAN interface of the CE200. WAN VCs on one side of the DSLAM must be mapped to the DSL VCs on the other side. And, finally, VCs on the DSL interface going to the CPE router must be mapped to VCs on the CPE. For each PPP link, the

procedure must be repeated. A new WAN VC must be created for each new DSL VC.

In the following diagram, only one DSL port and one WAN port appear. In the procedure that follows, first the WAN VC, at 1.3.1.130, will be created. Then the DSL VC, at 1.8.1.23, will be created. Finally, the DSL VC will be mapped to the WAN VC. Configuration of the VC on the Telco router and configuration of the VC on the CPE router are not shown.



Cross-Connect Netmodel—VPIs and VCIs

For the first PPP connection on the DSL port, follow these steps:

1. Set the VC on the WAN port. The VPI and VCI on the WAN VC must match the ones on the Telco router, but the VCL does not have to match its counterpart on the Telco router. Set AdminStatus to Up.

   ```
   CRAFT (1.2)> set cmatmvcl [1.3.1.130] vpi=1 vci=121
          adminstatus=up
   ```

   Issue a *Get* for the VC and you will see this table:

   ```
   CRAFT (1.2)> get cmatmvcl [1.3.1.130]

       Group: cmAtmVclTable
       Instance: [1.3.1.130]
       PII                  =  1.3.1.130
       Vpi                  =  1
       Vci                  =  121
       AdminStatus          =  Up
       OperStatus           =  Up
       LastChange           =  0 day 0 hour 0 min 0.0 sec
                               (2002/12/23-04:03:39)
       AalType              =  Aal5
       Aal5CpcsTransmitSduS =  1600
       Aal5CpcsReceiveSduSi =  1600
       RowStatus            =  Active
       TransmitTrafficDescr =  1
       OAMState             =  adminDown(0)
       OAMAdminState        =  disabled
       OAMAutoLBState       =  disabled
       OAMManualLBCmd       =  none
       OAMManualLBCmdStatus =  none
       OAMLBInterval        =  5
       OAMLBTimeOut         =  15
       OAMTxAISCells        =  0
       OAMRxRDICells        =  0
       OAMRxAISCells        =  0
       OAMTxRDICells        =  0
       OAMTxLBRequestCells  =  0
       OAMRxLBResponseCells =  0
       OAMRxLBRequestCells  =  0
       OAMTxLBResponseCells =  0
       OAMRxUnsupportedCell =  0
   ```

```
OAMTxDiscards        =  0
OAMRxDiscards        =  0
```

2. Set the NetModel and the EncapsulationType for the WAN VC. The NetModel must be Cross-Connect. The EncapsulationType must be None.

```
CRAFT (1.2)> set cmiface [1.3.1.130] netmodel=cross-connect
       encapsulationtype=none
```

Issue a *Get* for the VC and you will see this table:

```
CRAFT (1.2)> get cmiface [1.3.1.130]

Group: cmIfaceTable
Instance: [1.3.1.130]
PII                  =  1.3.1.130
IfIndex              =  1.3.1.130
Name                 =  ""
GroupName            =  ""
AdditionalInfo       =  ""
NetModel             =  Cross-Connect
IpAddr               =  0.0.0.0
NetMask              =  0.0.0.0
MacAddr              =  ff.ff.ff.ff.ff.ff
BurnedInMacAddr      =  ff.ff.ff.ff.ff.ff
FarEndAddr           =  0.0.0.0
DestPII              =  0.0.0.0
CMCPCompatible       =  No
EncapsulationType    =  None
FwdMode              =  VC-VC-payload
Pix                  =  274
ServiceClass         =  None
```

3. Set the VC on the DSL port. Use a PII such as 1.8.1.23 and specify a VPI and VCI. They must match the ones on the CPE. Typically, they are 0 and 32. Documentation for CPEs will give defaults, indicate if they can be changed, and show when multiple VCs are possible. The number of the VC must be in the range of 16 to 23.

```
CRAFT (1.2)> set cmatmvcl [1.8.1.23] vpi=0 vci=32 adminstatus=up
```

Issue a *Get* for the VC and you will see this table:

```
CRAFT (1.2)> get cmatmvcl [1.8.1.23]

Group: cmAtmVclTable
Instance: [1.8.1.23]
PII                  =  1.8.1.23
Vpi                  =  0
Vci                  =  32
AdminStatus          =  Up
OperStatus           =  Up
LastChange           =  0 day 0 hour 0 min 0.0 sec
                        (2002/12/23-08:03:37)
AalType              =  Aal5
Aal5CpcsTransmitSduS =  1600
Aal5CpcsReceiveSduSi =  1600
RowStatus            =  Active
TransmitTrafficDescr =  1
OAMState             =  adminDown(0)
OAMAdminState        =  disabled
OAMAutoLBState       =  disabled
OAMManualLBCmd       =  none
OAMManualLBCmdStatus =  none
OAMLBInterval        =  5
OAMLBTimeOut         =  15
OAMTxAISCells        =  0
OAMRxRDICells        =  0
OAMRxAISCells        =  0
OAMTxRDICells        =  0
OAMTxLBRequestCells  =  0
OAMRxLBResponseCells =  0
OAMRxLBRequestCells  =  0
OAMTxLBResponseCells =  0
OAMRxUnsupportedCell =  0
OAMTxDiscards        =  0
```

```
    OAMRxDiscards          =  0
```

4. Set the NetModel and the EncapsulationType for the DSL port. Use the same PII, 1.8.1, that you used in step 3. The VC number is not necessary. Set CMCP to No (the default is Yes).

```
CRAFT (1.2)> set cmiface [1.8.1] netmodel=cross-connect
       encapsulationtype=atm cmcpcompatible=no
```

   Issue a *Get* for the Port and this table is returned:

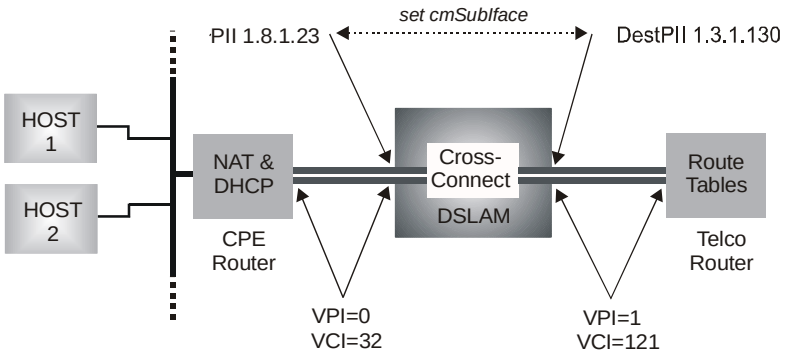
```
CRAFT (1.2)> get cmiface [1.8.1]

    Group: cmIfaceTable
    Instance: [1.8.1.0]
    PII                    =  1.8.1.0
    IfIndex                =  1.8.1.0
    Name                   =  ""
    GroupName              =  ""
    AdditionalInfo         =  ""
    NetModel               =  Cross-Connect
    IpAddr                 =  0.0.0.0
    NetMask                =  0.0.0.0
    MacAddr                =  ff.ff.ff.ff.ff.ff
    BurnedInMacAddr        =  0.0.0.0.0.0
    FarEndAddr             =  0.0.0.0
    DestPII                =  0.0.0.0
    CMCPCompatible         =  No
    EncapsulationType      =  atm
    FwdMode                =  PER-VC
    Pix                    =  3
    ServiceClass           =  D
```

5. Set the VC on the DSL port so that it points toward the VC on the WAN port. All data from the VC on the DSL port will automatically go to the VC on the WAN port, and vice versa. Use the DestPII setting to accomplish the mapping.

```
CRAFT (1.2)> set cmsubiface [1.8.1.23] destpii=1.3.1.130
```

   Perform a Get on the VC and you will see this table:

```
CRAFT (1.2)> get cmsubiface [1.8.1.23]

    Group: cmSubIfaceTable
    Instance: [1.8.1.23]
    PII                    =  1.8.1.23
    DestPII                =  1.3.1.130
    Name                   =  ""
    RowStatus              =  Active
    Priority               =  Low
```



Cross-Connect Netmodel—Destination PII

IP Netmodel

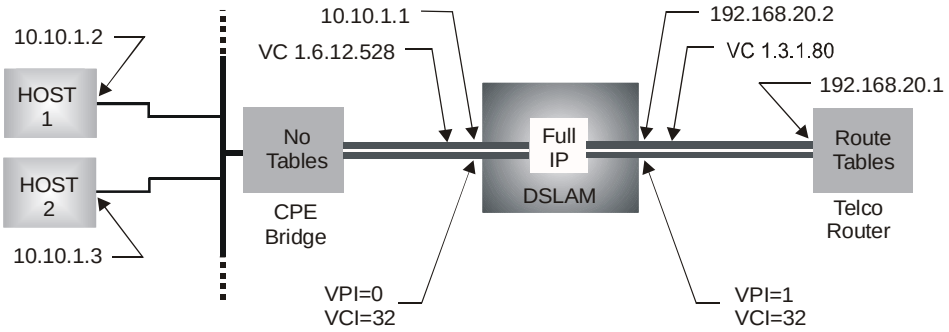
In the IP netmodel, the end points of the VCs must be manually configured for each of the stages on the link, but this time, each of the interfaces requires an IP address. (In the diagram below, all addresses are private, although they could be public.)

The VC going from the Telco router to the WAN port on the CE200 must have an IP address at each end. Its endpoints must be on different subnets. The WAN VC and the DSL port also must have IP addresses. They also must be on different subnets. Finally, the DSL port and its downstream hosts must have IP addresses, but they must be on the same subnet.

Since the CPE is in bridge mode, it doesn't require an IP address. The DSL port, not the upstream CPE port, is the gateway for hosts on the LAN. But, since the CE200 in the IP netmodel is in router mode, it sends packets coming upstream from its DSL ports to its WAN VC based on destination addresses in the packets. It sends packets coming downstream from its WAN VC to the proper DSL ports based on destination addresses.

For each of the DSL ports with G.dmt and G.lite CPEs connected to them, part of the configuring process must be repeated. New DSL VCs must be created on each DSL port.

In the following diagram, only one DSL and one WAN port appear. In the procedure on the following pages, first a WAN VC with an IP address of 192.168.20.2 will be created; then a DSL VC with an IP address of 10.10.1.1. Finally, the default IP route on the CE200 will be set. Configuration of addresses on the Telco router and downstream hosts will not be shown.



IP Netmodel—VPIs and VCIs

1. Set the VC on the WAN port. The VPI and VCI on the WAN VC must match those on the Telco router. Set AdminStatus to Up (the default is Down).

```
CRAFT (1.2)> set cmatmvcl [1.3.1.80] vpi=1 vci=122
       adminstatus=up
```

   Issue a *Get* for the VC and you will see this table:

```
CRAFT (1.2)> get cmatmvcl [1.3.1.80]
    Group: cmAtmVclTable
```

```
    Instance: [1.3.1.80]
    PII                   =  1.3.1.80
    Vpi                   =  1
    Vci                   =  122
    AdminStatus           =  Up
    OperStatus            =  Up
    LastChange            =  0 day 0 hour 0 min 0.0 sec
                             (2002)04/14-14:41:37)
    AalType               =  Aal5
    Aal5CpcsTransmitSduS  =  1600
    Aal5CpcsReceiveSduSi  =  1600
    RowStatus             =  Active
    TransmitTrafficDescr  =  1
    OAMState              =  adminDown(0)
    OAMAdminState         =  disabled
    OAMAutoLBState        =  disabled
    OAMManualLBCmd        =  none
    OAMManualLBCmdStatus  =  none
    OAMLBInterval         =  5
    OAMLBTimeOut          =  15
    OAMTxAISCells         =  0
    OAMRxRDICells         =  0
    OAMRxAISCells         =  0
    OAMTxRDICells         =  0
    OAMTxLBRequestCells   =  0
    OAMRxLBResponseCells  =  0
    OAMRxLBRequestCells   =  0
    OAMTxLBResponseCells  =  0
    OAMRxUnsupportedCell  =  0
    OAMTxDiscards         =  0
    OAMRxDiscards         =  0
```

2. Set NetModel, IPAddr, NetMask, and Encapsulation- Type for the WAN port and the FarEndAddr. The NetModel must be IP. The EncapsulationType must be rfc1483.

```
CRAFT (1.2)> set cmiface [1.3.1.80] netmodel=ip
       ipaddr=192.168.20.2 netmask=255.255.255.255
       farendaddr=192.168.20.1 encapsulationtype=rfc1483
```

Issue a *Get* for the VC and you will see this table:

```
CRAFT (1.2)> get cmiface [1.3.1.80]

    Group: cmIfaceTable
    Instance: [1.3.1.80]
    PII                   =  1.3.1.80
    IfIndex               =  1.3.1.80
    Name                  =  ""
    GroupName             =  ""
    AdditionalInfo        =  ""
    NetModel              =  IP
    IpAddr                =  192.168.20.2
    NetMask               =  255.255.255.255
    MacAddr               =  ff.ff.ff.ff.ff.ff
    BurnedInMacAddr       =  ff.ff.ff.ff.ff.ff
    FarEndAddr            =  192.168.10.1
    DestPII               =  0.0.0.0
    CMCPCompatible        =  No
    EncapsulationType     =  rfc1483
    FwdMode               =  Full-IP
    Pix                   =  28
    ServiceClass          =  None
```

3. Set the VC on the DSL port. Use a PII such as 1.6.12.528 and specify a VPI and VCI. They must match the ones on the CPE. Typically, they are 0 and 32. The number of the VC itself must be 528. It cannot be any other number for rfc1483.

```
CRAFT (1.2)> set cmatmvcl [1.6.12.528] vpi=0 vci=32
       adminstatus=up
```

Perform a Get on the VC and you will see this table

```
CRAFT (1.2)> get cmatmvcl [1.6.12.528]

    Group: cmAtmVclTable
    Instance: [1.6.12.528]
```

```
    PII                  =  1.6.12.528
    Vpi                  =  0
    Vci                  =  32
    AdminStatus          =  Up
    OperStatus           =  Up
    LastChange           =  0 day 0 hour 0 min 36.0 sec
                            (2002/04/14-14:42:13)
    AalType              =  Aal5
    Aal5CpcsTransmitSduS =  1600
    Aal5CpcsReceiveSduSi =  1600
    RowStatus            =  Active
    TransmitTrafficDescr =  0
```

4. Set the NetModel and the EncapsulationType for the DSL port. Use the same PII, 1.6.12, that you used in step 3. The VC number is not necessary. Set CMCPCompatible to No (the default is Yes).

```
CRAFT (1.2)> set cmiface [1.6.12] netmodel=ip
       encapsulationtype=rfc1483 cmcpcompatible=no
```

Perform a Get on the port and observe the following table:

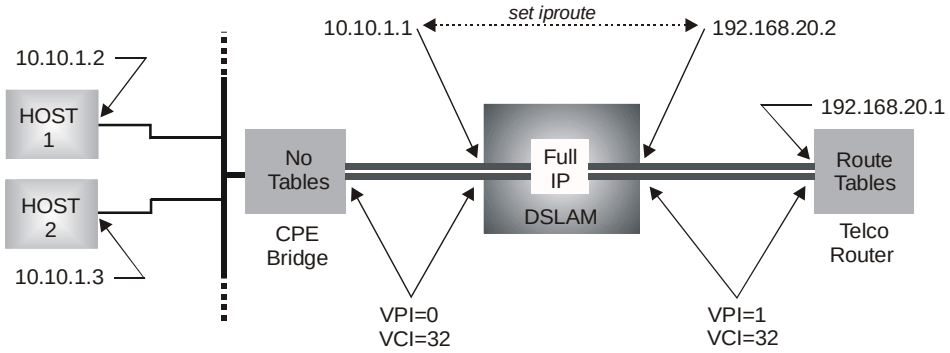
```
CRAFT (1.2)> get cmiface [1.6.12]

    Group: cmIfaceTable
    Instance: [1.6.12.0]
    PII                  =  1.6.12.0
    IfIndex              =  1.6.12.
    Name                 =  ""
    GroupName            =  ""
    AdditionalInfo       =  "
    NetModel             =  IP
    IpAddr               =  10.10.1.1
    NetMask              =  255.255.255.0
    MacAddr              =  ff.ff.ff.ff.ff.ff
    BurnedInMacAddr      =  0.0.0.0.0.0
    FarEndAddr           =  0.0.0.0
    DestPII              =  0.0.0.0
    CMCPCompatible       =  No
    EncapsulationType    =  rfc1483
    FwdMode              =  Full-IP
    Pix                  =  14
    ServiceClass         =  D
```

5. Perform a set for the default IP route leading from the DSL ports to the WAN VC and, alternatively, from the VC to the DSL ports. The IP netmodel should work without a default IP route, however, since the CE200 acts as a router for the traffic going between the DSL ports (VCs) and the WAN ports (VCs).

```
CRAFT (1.2)> set iproute [0.0.0.0] nexthop=192.168.20.1
       type=direct
```

Issue a *Get* and observe the following table:

```
CRAFT (1.2)> get iproute [0.0.0.0]

    Group: ipRouteTable
    Instance: [0.0.0.0]
    Dest                 =  0.0.0.0
    IfIndex              =  1.3.1.80
    Metric1              =  1
    Metric2              =  -1
    Metric3              =  -1
    Metric4              =  -1
    NextHop              =  192.168.20.2
    Type                 =  Direct
    Proto                =  netmgmt
    Age                  =  755624
    Mask                 =  0.0.0.0
    Metric5              =  -1
    Info                 =  0.0
```



IP Netmodel—Default Route

Chapter 8
Control and WAN Redundancy

This chapter discusses the concept of system control in the CE200 with a redundancy complex installed. It explains how to configure and enable a redundancy complex and how to recover after a failover has occurred and after the failed module has been replaced.

A redundancy complex in the CE200 contains a System Control Module, a Buffer Control Module, and one or two WAN modules in slots 14 through 17 of the CE200. The modules must reflect exactly the modules installed in slots 2 through 5 of the CE200. When a module fails in slots 2 through 5, the redundancy complex in slots 14 through 17 automatically takes over with little or no interruption in service.

Overview of Redundancy

The redundant modules are installed in slots 14 through 17 of the CE200 chassis. In non-redundant systems, these slots are empty. If necessary, the CE200 can operate satisfactorily with Control and WAN modules in slots 14 through 17 and with slots 2 through 5 empty. With such a configuration, however, system startup will be slightly slower. It is best to replace the failed module on the *Preferred* side and switch control of the CE200 back to the *Preferred* side.

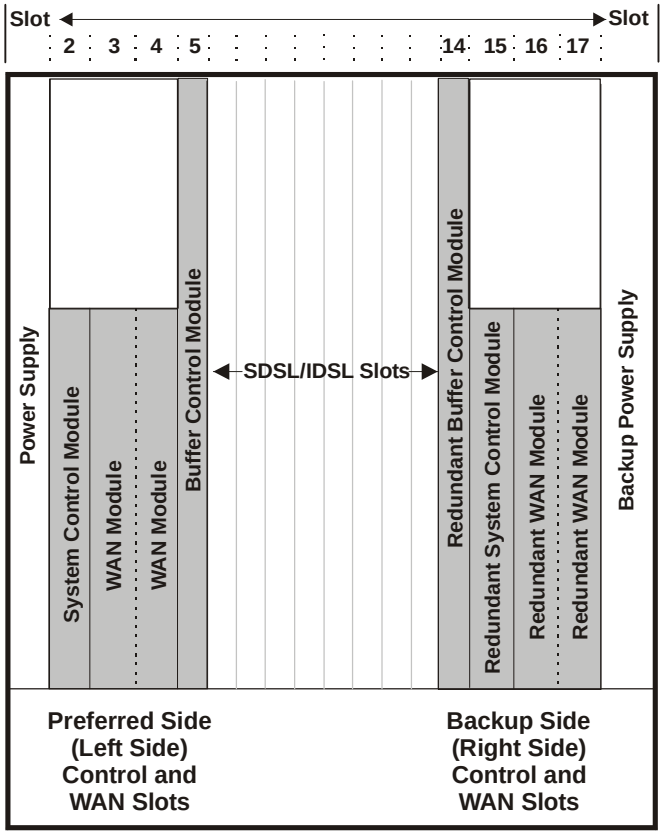
Preferred Side and Backup Side

The left side of the CE200 chassis containing slots 2 through 5 is the *Preferred* side. Cards in the slots normally control operations. The right side of the CE200 chassis containing slots 14 through 17 is the *Backup* side. Cards in the Backup side must reflect exactly cards in the Preferred side.

First, if a System Control Module (SCM) 2 is in the Preferred side, the redundancy complex must have an SCM 2. Also, if a Buffer Control Module 1 is in the Preferred side of the chassis, the redundancy complex must have a BCM-1.

Second, if a Buffer Control Module 2 and SCM 3 are in the Preferred side, the redundancy complex must also have a BCM-2 and SCM-3. These two cards support each other.

Finally, if an ATM DS3 card and Frame Relay DS3 card are in slots 3 and 4 on the Preferred side, the redundancy complex must have an ATM DS3 card anda Frame Relay DS3 card in slots 16 and 17.



Module Arrangement in a Redundant System

Redundancy Configurations

A redundancy complex can be set up in one of these two configurations:

- *Standard redundancy complex*—All WAN modules are connected to ports on an upstream device, e.g., a router or a switch. This setup uses twice as many ports on the device. See page 125 for more information.
- *DS3 Switch complex*—All WAN modules are connected to a DS3 Protection Switch installed in the CE200 chassis. The DS3 switch, not the WAN modules, is connected to the device. See page 127 for more information.

Primary Complex and Secondary Complex

For the sake of this discussion, the side of the chassis that is currently controlling operations and traffic has the name, *Primary complex*; the other side has the name, *Secondary complex*. Such a designation is relative, since either the Preferred side or the Backup side can become the Primary complex.

Whether or not a CE200 is equipped for Control and WAN Redundancy is transparent to the DSL side of the CE200.

Configuration of DSL interfaces is unaffected, and the additional hardware is transparent to upstream devices that may be directing data traffic to end points served by the CE200 (if the DS3 switch is in use).

When redundancy is enabled and power is recycled, the Preferred side automatically becomes the Primary complex. When redundancy is disabled, whichever side is the Primary remains in that role, even through a power cycle.

Operation

With the redundancy complex installed and enabled, the CE200 will automatically transfer control from the Preferred side to the Backup side when one of the Control or WAN modules fail. It will also transfer control if a WAN link is taken down administratively or if it fails operationally. However, if you configure a WAN link as administratively disabled, any subsequent signal transitions on that interface will not be used in failover decisions.

A module failure is different from a startup failure; if a Buffer Control or WAN module simply fails to start up, failover will not occur. This exception allows you to start up a CE200 that may not have a full complement of modules installed.

On a fully functional CE200, the command, getall cmBoard, will provide different information depending on whether you are logged in to the Primary or Secondary complex:

- The Primary System Control Module displays information about all of the boards in the chassis.
- The Secondary System Control Module only displays information about itself.

After a failover to the Backup side, you cannot access the Preferred side (now the Secondary complex) if the management connection is through the WAN interface. However, you can access the Secondary complex if access is through the Ethernet port.

After you fix the problem on the Preferred side and before you transfer control back to the Preferred side, be sure to save the configuration so that the Config files will be identical on both sides of the CE200. This save and transfer process, while normally of brief duration, is service-affecting and should be scheduled for a time of reduced traffic.

Setting Up Redundancy

Installing the Modules

To install a redundancy complex in a CE200:

1. If you are installing a Standard Alarm Panel or a DS3 Protection Switch/Alarm Panel:
   a) Power down the CE200.
   b) Install the Alarm PANEL.

c) Install the redundant Control and WAN modules. If necessary, refer to Chapter 9 for information about installing or replacing modules.

d) Reapply power to the CE200.

2. Ensure that the redundancy feature is disabled:

```
CRAFT (1.2)> set cmsystem redundancy=disabled
```

3. Connect the WAN ports:

e) For a standard redundant complex, connect the WAN ports to the upstream router ports.

f) For a DS3 Protection Switch, connect the WAN ports to the DS3 switch. If you need specific instructions on connection, consult your *CopperEdge Installer Guide*.

4. Provision the redundant System Control Module with its own IP address and interface identity, just as you did for the Preferred System Control Module.

5. Ping the redundant System Control Module over the LAN and WAN.

6. Check that the primary and secondary PIIs exist, and that the Ethernet port of the active System Control Module is listed.

```
CRAFT (1.2)> get cmsystem

    Group: cmSystem
    ObjectClass          =  System
    OperState            =  Enabled
    Version              =  E 4.0
    Master               =  0.0.0.0
    ConfigFileName       =  config.tgz
    CalendarTime         =  2002/10/30-18:46:09
    MyPII                =  1.2.0.0
    PrimaryPII           =  1.2.0.0
    SecondaryPII         =  1.15.0.0
    MgmtPII              =  0.0.0.0
    Redundancy           =  Disabled
    ShelfCount           =  1
    ExpIpSubNet          =  192.168.250.0
    ConfigSynch          =  Saved
    Command              =  SaveConfig
    CommandStatus        =  Succeeded
```

7. Check the Control and WAN modules:

g) The modules are operating with the same software version.

h) The Preferred System Control Module shows Role=Primary, and the Backup System Control Module shows Role=Secondary.

i) The Backup WAN modules show Operstate=Disabled.

```
CRAFT (1.2)> geta cmboard

    Index          ObjectClass    OperState      HwType
    HwVersion       SwVersion       PromVersion     Role
    ClusterRole     UpTime          NumPorts        FileName
    FileDate        ConfigChange    Command         SerialNumber
    Information

    Instance: [1.2.0.0]
    1.2.0.0         SystemControlModule Enabled     SystemControlModule
    R 1.0           4.0.72              4.0.59      Primary
    Master          0 day 4 hour 5 min  5           P:/ce200/scm
    Oct 13 2002, 00:39: 0               None        ****
    RAM 256Mb;Flash 64M
```

```
Instance: [1.3.0.0]
1.3.0.0         DS3ATM-WAN        Enabled      DS3ATM-WAN
R 1.3           4.0.72            2.5.11       Active
NotApplic       0 day 4 hour 6 min  1         DS3ATM.HEX
Oct 13 2002, 00:31: 0              None       I4FT03SX_003470105
""

Instance: [1.4.0.0]
1.4.0.0         DS3FR-WAN           Enabled      DS3FR-WAN
R 1.0           4.0.72              1.35.19      Active
NotApplic       0 day 4 hour 5 min  1            ds3fr.bin
Oct 13 2002, 00:25: 0               None         MI3SM01GL
""

Instance: [1.5.0.0]
1.5.0.0         BufferControlModule Enabled      BufferControlModule
R 1.0           4.0.72              1.40.18      Active
NotApplic       0 day 4 hour 6 min  0            bc.bgz
Unknown         0                   None         BC1489076_003160111
""

             <OTHER BOARDS>

Instance: [1.14.0.0]
1.14.0.0        BufferControlModule Enabled      BufferControlModule
R 1.0           4.0.72              1.40.18      Active
NotApplic       0 day 2 hour 59 min 0            bc.bgz
Unknown         0                   None         BCFT00IQ_003160109
""

Instance: [1.15.0.0]
1.15.0.0        SystemControlModule Enabled      SystemControlModule
R 1.0           4.0.72              4.0.59       Secondary
Master          0 day 2 hour 59 min 5            P:/ce200/scm
Oct 13 2002, 00:39: 0                    None         ****
RAM 256Mb;Flash 64M

Instance: [1.16.0.0]
1.16.0.0        DS3ATM-WAN          Disabled     DS3ATM-WAN
R 1.3           4.0.72              2.5.11       Active
NotApplic       0 day 3 hour 0 min  1            DS3ATM.HEX
Oct 13 2002, 00:31: 0               None         MI4FT03SV_003470105
""

Instance: [1.17.0.0]
1.17.0.0        DS3FR-WAN           Disabled     DS3FR-WAN
R 1.0           4.0.72              1.35.19      Active
NotApplic       0 day 2 hour 59 min 1            ds3fr.bin
Oct 13 2002, 00:25: 0               None         MI3SM01BF
""
```

Enabling Redundancy

Now that the redundancy complex is installed and the system is operating normally, you must enable redundancy on the system so failover can occur. To enable redundancy:

```
set cmsystem redundancy=enable

Set Successful
get cmsystem

Group: cmSystem
ObjectClass          =  System
OperState            =  Enabled
Version              =  E 4.0
Master               =  0.0.0.0
ConfigFileName       =  config.tgz
CalendarTime         =  2002/10/30-18:46:09
MyPII                =  1.2.0.0
PrimaryPII           =  1.2.0.0
SecondaryPII         =  1.15.0.0
MgmtPII              =  0.0.0.0
Redundancy           =  Enabled
ShelfCount           =  1
ExpIpSubNet          =  192.168.250.0
ConfigSynch          =  Saved
Command              =  SaveConfig
CommandStatus        =  Succeeded
```

Redundancy is now fully enabled and automatic failover is possible. The WAN modules on the Backup side show Operstate=Enabled.

```
CRAFT (1.2)> geta cmboard [1.15]

    Index           ObjectClass          OperState     HwType
    HwVersion       SwVersion            PromVersion   Role
    ClusterRole     UpTime               NumPorts      FileName
    FileDate        ConfigChange        Command       SerialNumber
    Information


    Instance: [1.15.0.0]
    1.15.0.0       SystemControlModule Enabled  systemControlMod
    R 1.0            4.0.72              4.0.59    Secondary
    Master            0 day 0 hour 10 min 5          P:/ce200/scm
    Oct 13 2002, 00:39: 0                    None      ****
    RAM 256Mb;Flash 64M

    Instance: [1.16.0.0]
    1.16.0.0         DS3ATM-WAN          Enabled       DS3ATM-WAN
    R 1.3             4.0.72              2.5.11        Active
    NotApplic         0 day 3 hour 0 min  1             DS3ATM.HEX
    Oct 13 2002, 00:31: 0              None  MI4FT03SV_003470105
    ""

    Instance: [1.17.0.0]
    1.17.0.0         DS3FR-WAN           Enabled       DS3FR-WAN
    R 1.0            4.0.72              1.35.19       Active
    NotApplic        0 day 2 hour 59 min 1             ds3fr.bin
    Oct 13 2002, 00:25: 0                None          MI3SM01BF
    ""
```

Next you will configure the Control and WAN modules, and, for standard redundancy, the upstream router.

Configuring the Redundant Modules

To accommodate redundancy features, each WAN module has both a physical address and a logical address:

- The logical address denotes the WAN module's *virtual slot number* for redundancy considerations. Use it when referring to the module in connection with any routing-related considerations (such as modifying the DestPII object in cmIface).
- The physical address denotes the WAN module's actual location. Use it for all other purposes (such as setting thresholds and using the command, getall cmBoard).

There are only two logical WAN slots on a CE200: slots 3 and 4. Use of the logical address in the cmIface DestPII object implies that the table is identical on both sides of the CE200 (both internally and externally).

The Preferred and Backup System Control Modules have separate IP addresses for system management through their respective Ethernet ports. These addresses stay with the physical interface, regardless of its role.

A separate Config file resides in each complex (they should be identical). When you issue a Save Config command, the CE200 saves the Config file in the Primary complex, copies it to the Secondary complex, and then restarts the Secondary complex. The integrity of routing information for a redundant system depends on these two files remaining identical, so it is important to save

the configuration any time you change a permanent value. The reliability provided by redundancy is greatly reduced if the Config file is not current when failover occurs.

The following sections discuss the different steps for configuring the modules in a standard redundancy system versus a DS3 Switch redundant system.

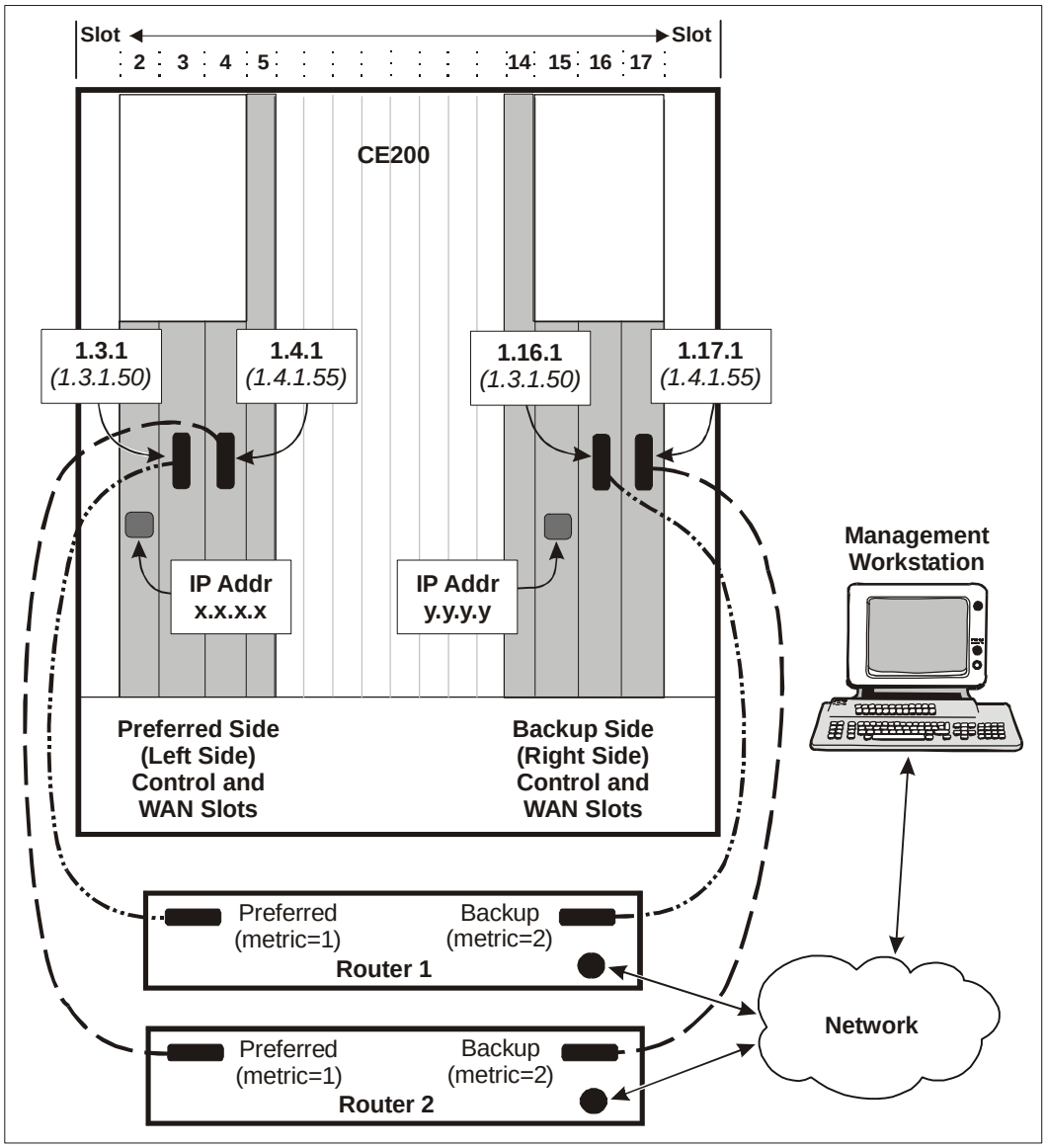
Standard Redundancy Complex

In a standard redundancy system, all WAN modules are connected to ports on the upstream device. The Preferred and Backup WAN modules share the same logical address and are connected to preferred and backup ports on the same device.

In the CE200, assign one logical address to the WAN modules in slots 3 and 16, and another logical address to the WAN modules in slots 4 and 17. Then configure the upstream device with the appropriate metrics and virtual circuit numbers to ensure compatibility.

For example, the following illustration shows the following configuration:

- WAN modules in slots 3 and 16 have a logical PII of 1.3.1.50. Data traffic flows over separate lines to upstream device 1.
- WAN modules in slots 4 and 17 have a logical PII of 1.4.1.55. Data traffic flows over separate lines to upstream device 2.
- The IP address for system management of the Preferred side is shown as x.x.x.x; the management IP address of the Backup side is shown as y.y.y.y.



Standard Redundancy—Physical and Logical WAN Links

If the upstream device is a router, you must also configure the router to establish priority for the Preferred side interface to route user traffic to the CE200. The router should have only a single route to the CE200 management addresses x.x.x.x and y.y.y.y. In the event of a failover to the Backup side, the redundancy complex transmits a protocol message to the router, and, as the new Primary, becomes the priority connection for the router.

DS3 Protection Switch Redundant Complex

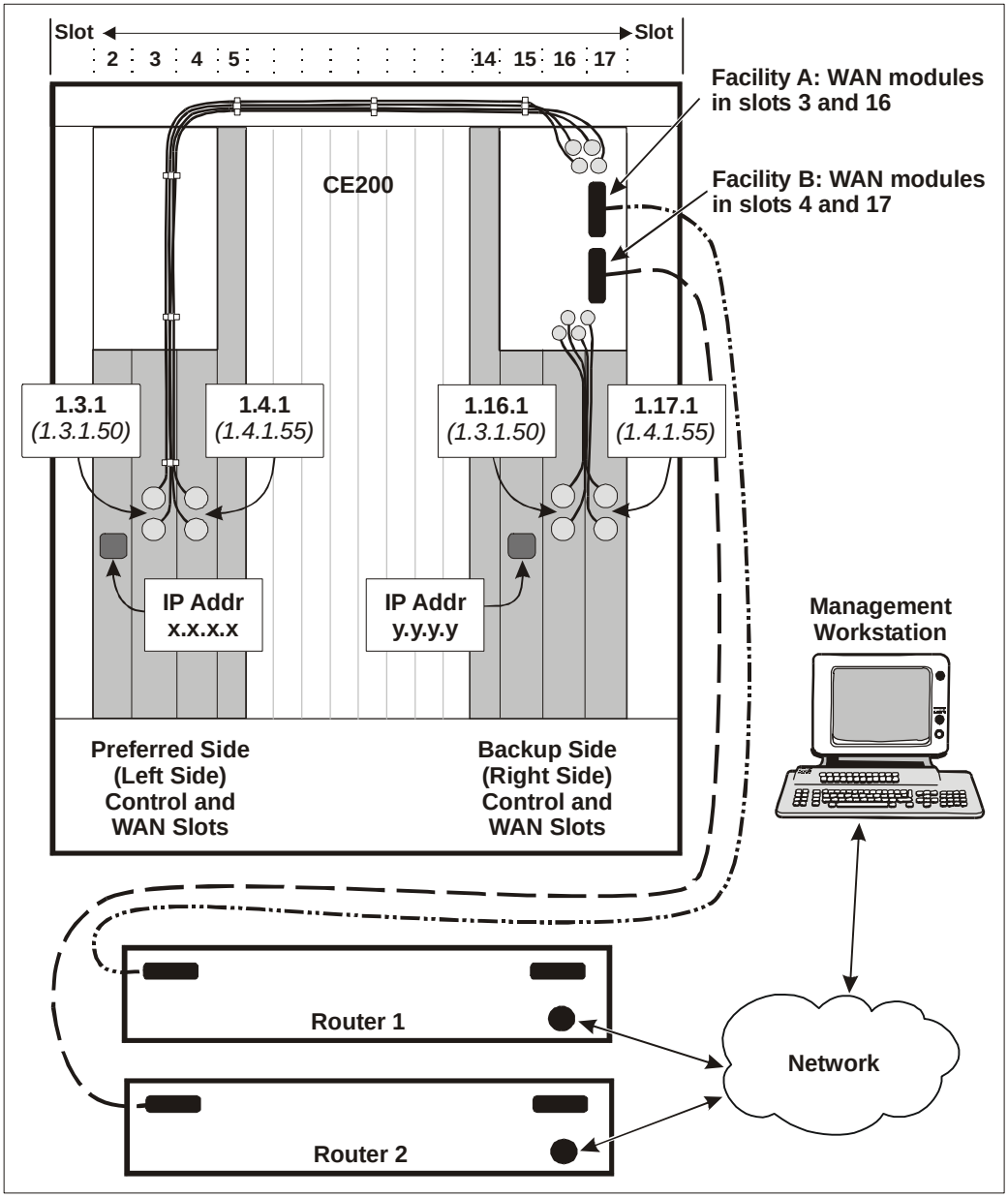
If your CE200 is equipped with a DS3 Protection Switch, the DS3 WAN modules can be connected to a single WAN facility (such as a T3 line) or to two WAN facilities. In the current release, there are two possible configurations:

- *Control and WAN redundancy using only two WAN modules*—This provides complete complex redundancy using only one facility.
- *Control and WAN redundancy using all four WAN modules*—The WAN modules in slots 3 and 16 are routed to Facility A, and WAN modules in slots 4 and 17 are routed to Facility B.

The Preferred and Backup WAN modules share the same logical address and are connected to either Facility A or Facility B on the DS3 switch. Assign one logical address to the WAN modules in slots 3 and 16, and another logical address to the WAN modules in slots 4 and 17.

For example, the following illustration shows the following configuration:

- WAN modules in slots 3 and 16 have a logical PII of 1.3.1.50. Data traffic flows over separate lines to the DS3 Switch Facility A ports, and then to upstream device 1.
- WAN modules in slots 4 and 17 have a logical PII of 1.4.1.55. Data traffic flows over separate lines to the DS3 Switch Facility B ports, and then to upstream device 2.
- The IP address for system management of the Preferred side is shown as x.x.x.x; the management IP address of the Backup side is shown as y.y.y.y.



DS3 Switch Redundancy—Physical and Logical WAN Links

Module Failures

Automatic Failover

With the redundancy complex installed and enabled, the CE200 will automatically transfer control from the Preferred side to the Backup side when:

- The System Control Module fails
- The Buffer Control Module fails
- A WAN link goes down or a WAN module fails

- An operational WAN link is taken down administratively

If you configure a WAN link as administratively disabled, any subsequent signal transitions on that interface will not be used in failover decisions.

A module failure is different from a startup failure; if a Buffer Control or WAN module simply fails to start up, failover will *not* occur. This exception allows you to start up a CE200 that may not have a full complement of modules installed.

When a Preferred side Control or WAN module fails:

- There is a short interruption in service (typically less than 10 seconds) while any WAN links to the failed complex are disabled.
- The Backup side takes over the Control and WAN functions and switches to the role of Primary within approximately five seconds.
- The Backup side automatically disables redundancy. This prevents the Preferred side from assuming the Primary role before any problems can be fixed.
- The configured circuits are automatically switched over to the Backup side and will continue to operate normally. No additional circuit configuration is required.
- User traffic and control are resumed as soon as this sequence is completed.
- The System Control Module on the Preferred side assumes the Secondary role. You will not be able to configure the CE200 through its Craft port.

Commanded Failover

In addition to automatic failover, you can command a transfer of control from the Primary complex to the Secondary complex at any time. You can do this while connected to either the Primary or the Secondary complex.

To give up control while connected to the Primary complex:

```
set cmsystem command=relinquish
```

To take over control while connected to the Secondary complex:

```
set cmsystem command=takeover
```

In either case, the new Primary complex immediately disables redundancy and restarts the complex that gave up control.

Restoring Redundancy after a Failover

After a failover to the Backup side, you cannot access the Preferred side when the management connection is through the WAN interface. Most control operations are no longer accessible, such as:

- Commanded takeover by the Preferred side (**set cmsystem command=takeover**)
- Verification of Preferred side WAN link status
- External Preferred side WAN loopback
- Upgrading software on the Preferred side over the WAN interface

When you have resolved the cause of a failover, the redundancy functionality must be restored to the original (pre-failover) condition. This process, while normally of brief duration, is service-affecting and should be scheduled for a time of reduced traffic. Before you transfer control back to the Preferred side, be sure to save the configuration so that the Config files will be identical on both sides of the CE200.

The following procedures assume that redundancy is disabled, the Backup side has the Primary role, and a functional hardware complex is installed and running in the Preferred side.

Recovery on a Standard Redundant System

To recover from a failure that has been resolved:

1. Reconfigure the upstream device to prefer the Backup side by changing the Backup side from metric=2 to metric=0 (the lower metric is the preferred one). This ensures that the device prefers the Backup side until the CE200 is fully operational on its Preferred side.
2. Reenable redundancy in the system and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes while the Preferred side restarts.
3. If the WAN facility is a DS3, perform an external loopback test using dsx3Config to verify that the link is usable.
4. Verify connectivity (Ping or Telnet) from the management terminal or NMS to the CE200, using the IP address for the Preferred side management link.
5. Transfer control to the Preferred side:

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```

   The Preferred side assumes the Primary role. The new Primary System Control Module disables redundancy and administratively disables the WAN links on the Backup side.
6. On the device, verify that the CE200 Backup side WAN links are seen as down, and that traffic has reverted to the Preferred side interfaces.
7. Reconfigure the external device to prefer the Preferred side by resetting the Backup side port's routing metric to 2.
8. Log in to the Preferred side of the CE200.

9. Enable redundancy on the system and save the configuration.

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

The save process will take a few minutes while the Backup side restarts.

```
CRAFT (1.2)> set cmsystem redundancy=enabled
```

10. Verify that the WAN modules on the Backup side show Operstate=Enabled:

```
geta cmboard [1.15]

Index             ObjectClass         OperState     HwType
HwVersion         SwVersion           PromVersion   Role
ClusterRole       UpTime              NumPorts      FileName
FileDate          ConfigChange        Command       SerialNumber
Information

Instance: [1.15.0.0]
1.15.0.0            SystemControlModule Enabled
SystemControlModule
R 1.0             4.0.72              4.0.59        Secondary
Master            0 day 0 hour 10 min 5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.16.0.0]
1.16.0.0          DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3             4.0.72              2.5.11        Active
NotApplic         0 day 3 hour 0 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0               None    MI4FT03SV_003470105
""

Instance: [1.17.0.0]
1.17.0.0          DS3FR-WAN           Enabled       DS3FR-WAN
R 1.0             4.0.72              1.35.19       Active
NotApplic         0 day 2 hour 59 min 1             ds3fr.bin
Oct 13 2002, 00:25: 0                   None        MI3SM01BF
""
```

11. Resume normal operations.

Recovery from a DS3 Failover

To recover from a failure that has been resolved:

1. Log in to the Backup side of the CE200.
2. Enable redundancy for the system and save the configuration:

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

The save process will take a few minutes while the Preferred side restarts.

3. Tell the System Control Module on the Backup side to relinquish control:

```
CRAFT (1.2)> set cmsys command=relinquish
```

The Preferred side assumes the Primary role. The new Primary System Control Module disables redundancy and administratively disables the WAN links on the Backup side.

4. Log in to the Preferred side of the CE200.

5. Verify that the Preferred System Control Module [1.2] now shows Role=Primary, and the Backup System Control Module [1.15] shows Role=Secondary.

geta cmboard

```
Index            ObjectClass         OperState           HwType
HwVersion        SwVersion           PromVersion         Role
ClusterRole      UpTime              NumPorts            FileName
FileDate         ConfigChange        Command         SerialNumber
Information

Instance: [1.2.0.0]
1.2.0.0         SystemControlModule  Enabled    SystemControlMod
R 1.0               4.0.72            4.0.59             Primary
Master              0 day 0 hour 4 min  5           P:/ce200/scm
Oct 13 2002, 00:39: 0                   None                ****
RAM 256Mb;Flash 64M


                <OTHER BOARDS>

Instance: [1.15.0.0]
1.15.0.0       SystemControlModule Enabled      SystemControlMod
R 1.0               4.0.72              4.0.59          Secondary
Master              0 day 2 hour 59 min 5           P:/ce200/scm
Oct 13 2002, 00:39: 0                   None                ****
RAM 256Mb;Flash 64M

<OTHER BOARDS>
```

6. Enable redundancy on the CE200 and save the configuration (remember, redundancy is automatically disabled with each switchover).

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

The save process will take a few minutes while the Backup side restarts.

7. Verify that the WAN modules on the Backup side show Operstate=Enabled:

geta cmboard [1.15]

```
Index               ObjectClass         OperState     HwType
HwVersion           SwVersion           PromVersion   Role
ClusterRole         UpTime              NumPorts      FileName
FileDate           ConfigChange       Command      SerialNumber
Information

Instance: [1.15.0.0]
1.15.0.0            SystemControlModule Enabled
SystemControlModule
R 1.0               4.0.72              4.0.59        Secondary
Master             0 day 0 hour 10 min 5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.16.0.0]
1.16.0.0            DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3               4.0.72              2.5.11         Active
NotApplic           0 day 3 hour 0 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None
MI4FT03SV_003470105
""

Instance: [1.17.0.0]
1.17.0.0            DS3FR-WAN           Enabled       DS3FR-WAN
R 1.0               4.0.72              1.35.19       Active
NotApplic           0 day 2 hour 59 min 1             ds3fr.bin
Oct 13 2002, 00:25: 0                   None          MI3SM01BF
""
```

8. Resume normal operations.

Chapter 9
Troubleshooting

This chapter describes features of the CE200 that can aid in evaluating system performance, and in diagnosing and recovering from trouble.

CE200 Diagnostic Features

The CE200 includes features to help you to diagnose problems in case something should go wrong. Should a malfunction occur, these features can help you to pinpoint the device causing the problem, and indicate possible avenues for recovery.

Diagnostic features include:

- Visual indicators (front panel LEDs)
- Traps and alarms
- Loopbacks and Loop Quality Tests
- Built-in performance monitors
- Diagnostic port (assisted by Copper Mountain Technical Support)

Front Panel LED Indicators

Module Status Indicators

Circuit modules for the CE200 are equipped with front-panel LEDs that show module functionality at a glance. The module status LEDs on the DC power modules, DSL line modules, and Buffer Control Module indicate the general status of the circuit pack whenever power is applied.

Under normal circumstances, the module status indicator LEDs should be on and green. The Buffer Control Module status LED provides additional information as discussed below.

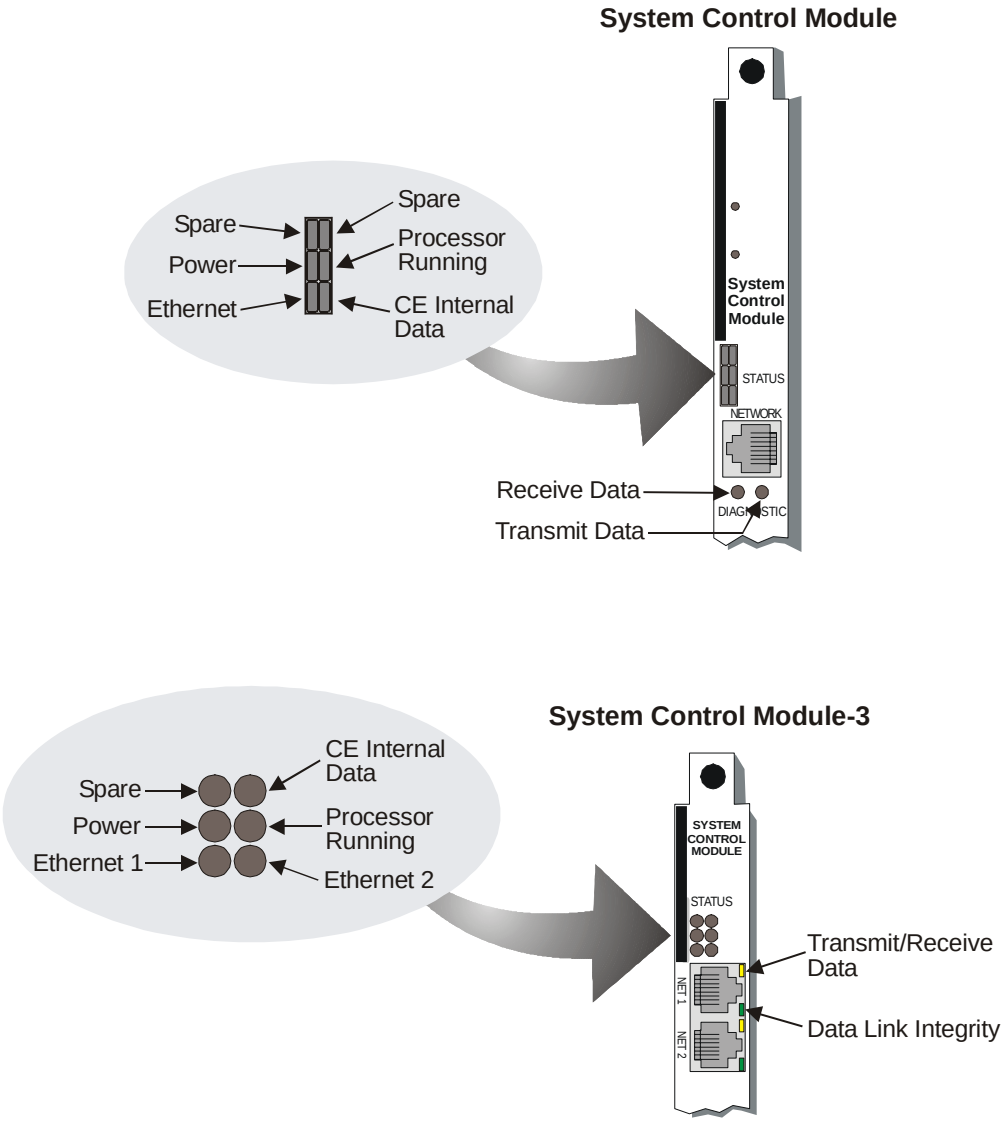
Buffer Control Module Status

During normal operation (startup sequence completed), the Buffer Control Module should be green and flickering to indicate data flow. In case of trouble, however, Buffer Control Modules of Version 2.1.63 and later could also display the following:

- Alternating *Red*/*Amber*—Abnormal operation. At least one invalid descriptor has been received from the System Control Module, the affected DSL ports have been deactivated, unaffected DSL ports will continue to operate normally.
- Continuous cycle of *Red*/*Amber*/Off/*Green*—The Buffer Control Boot PROM cannot properly initialize the module. Not usually indicative of a failure in the Buffer Control Module, but more likely in the PCI bus or other PCI device. Restarting the System Control Module may clear this condition; cycling power to the CE200 should always clear it.
- *Red*—Buffer Control Module is in latched reset because the CPU has failed to start, or the SRAM test failed. A full CE200 power cycle is required to clear this state.

System Control Module Status

The System Control Module and System Control Module-3 have six LEDs above the Ethernet connector. When power is applied, the Power LED immediately turns green, and the Processor Running indicator turns green within a second or two.



System Control Module LEDs

If the System Control Module is connected to an Ethernet network, the module will synchronize and negotiate its speed with the network. The Ethernet LED turns green if the network is a 10BASE-T (10 Mbps) system, and red if it is a 100BASE-T (100 Mbps) system.

A green-lit Ethernet LED is not a signal that the CE200 is connected to or communicating with a live Ethernet network; the LED turns green by default, whether or not the port is actually connected.

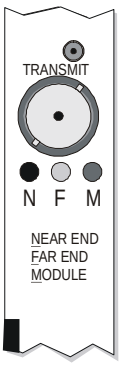
The CE Internal Data LED indicates data activity internal to the CE200 itself. The state of this LED does not correspond to the status of the system. At any time, the Internal Data LED may be off, blinking, pulsing, or solidly green.

On the System Control Module panel, the Receive Data and Transmit Data LEDs indicate data traffic. The Receive Data indicator turns or flickers green when Ethernet data packets are received by the CE200. The Transmit Data indicator turns or flickers green when Ethernet data packets are transmitted by the CE200.

On the System Control Module-3 panel, the Ethernet port has two LEDs: a single Transmit/Receive Data LED and a Data Link Integrity LED. The Transmit/Receive Data LED indicates data traffic. It turns or flickers yellow when data is received or transmitted on the Ethernet line. The Data Link Integrity LED turns green when the integrity of the Ethernet link for 10BASE-T or 100BASE-TX is acceptable.

DS3 WAN Module Status

The DS3 WAN module (ATM or Frame Relay) has three LED indicators on its front panel, just below the Transmit BNC connector. At power-up, all three LEDs light briefly as a functional test.



DS3 WAN Module

If the module is connected to a DS3 facility through a suitable access device, the green M (Module) LED will light, and the red N (Near End) and yellow F (Far End) LEDs will be off.

If there is a local failure, such as loss of signal or frame errors at the CE200 or within the DS3 module, the red N LED will light to indicate the failure.

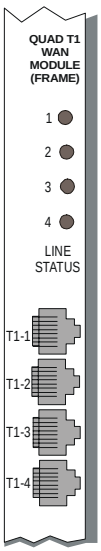
If a failure occurs on the link or with upstream equipment, or when diagnostic testing is in progress, the yellow F LED illuminates. When a software download is in progress, the yellow F LED blinks.

Quad T1 Frame WAN Module Indicators

The DS1 WAN module has four LEDs that indicate the condition of the T1 line or the type of traffic on the T1 line, including alarm signals and normal operating signals.

The LEDs are above the connectors, in the same order as the connectors (for example, the top LED is for port T1-1, and the bottom LED is for port T1-4). Each LED has three colors that indicate different states:

- *Green*—Normal operation. The line has been synchronized with the network and shows no local or far-end alarms.
- *Yellow*—Far-end failure, either the link or upstream equipment. The LED stays yellow until the condition is cleared.
- *Red*—A near-end failure condition, such as a loss of signal or frame errors at the CE200, or within the Quad T1 Frame Module itself. The LED stays red until the condition is cleared.



Quad T1 Frame WAN Module LEDs

DSL Module and Port Status

All SDSL, IDSL, G.SHDSL, ADSL (G.dmt or G.lite), and DS1 subscriber modules have a cluster of LED indicators (12 or 24 LEDs, depending on the type of module), each of which corresponds to one of its Subscriber ports.

If a port has been correctly wired and configured, the CPE at the far end of the subscriber link will train on the port and the corresponding LED on the CopperEdge DSL module will turn green. If the indicator is off, or if it continually alternates between off and on, there is a fault with the port, link, or CPE. Note, however, that a green LED indicates that a CPE is present and trained; it does not indicate whether or not the port is configured or configured properly.

T1 DSL Line Module (DS1 Subscriber Module) Port Status

The T1 DSL line module, like the other DSL modules, has a cluster of 12 LED indicators, each of which corresponds to one of its DSL ports. This module's port-status LEDs, however, may display in three different colors, indicating the following states:

- *Off*—Reset state, power down state, or administratively down (Admin=Down).
- *Green*—Normal operation, synchronized with the network; no local or far-end alarms; administratively up (Admin=Up).
- *Yellow*—Far-end failure with no near-end failure; administratively up (Admin=Up).
- *Red*—Near-end failure, such as loss of signal, loss of frame, or receiving AIS; administratively up (Admin=Up).

Events and Alarms

The CE200 is designed to be in continuous operation, 24 hours a day. During operation, various events will occur: operators will log in and out; configuration files will be modified and saved; the status of DSL or network links will change; and occasional failures or other operational anomalies may occur.

Events can be sent by any resource in the system. They are logged in memory (locally on the CE200 system), and are also translated into SNMP Trap PDUs to be sent to external SNMP managers. Here are the three groups of events on the CopperEdge:

- *Alarms*—Events that indicate the occurrence of an alarm condition; alarms will affect the status of the alarm panel; in general an alarm will remain in effect until cleared (either automatically or manually).
- *Alarm Clearing Notifications*—Events that clear a particular alarm (if one exists); if no corresponding alarm condition exists for such an event to clear, it is treated as a normal notification.
- *Event Notifications*—Events, such as warnings and informational messages, that neither trigger nor clear an alarm.

For a description of all of the Alarm, Alarm-Clearing, and Event Notifications that may occur on your CopperEdge system, see Appendix C.

Event Contents

Each event notification includes data identifying the type of event, the resource reporting the event, and the severity of the event. There may also be event-specific data with details about the particular event. The common data contained in every event consists of the following objects from the cmTrapEventTable: SeqNum, Type, ObjectClass, ClassId, Severity, and TimeTag.

Resource Identification

The resource that reported the event can be uniquely identified using two fields: ObjectClass and ClassId. Each resource in the system belongs to a specific object class (such as Board or Port). The specific resource within the class can be identified using the ClassID, which is the PII of the resource.

The class identifies the type of resource reporting the event and the ClassID identifies the specific resource in that class. For example, in a BoardDown event, the ObjectClass might be SystemControlModule and the ClassId would be 1.2.0.0. This event indicates the SCM in the second slot has gone down.

Object Classes

The following object classes are used to represent all of the resources in the CE200 system.

System Classes

These classes apply to the system as a whole.

- System
- Operator

Shelf Classes

These classes designate the type of shelf (chassis) reporting the event.

- Shelf
- CE200Shelf
- CE150Shelf

Board Classes

These classes specify the type of circuit pack/module.

- Board
- SystemControlModule
- SystemControlModule3
- SDSL10xModule-1
- SDSL10xModule
- SDSL30xModule
- SDSL30xModule-24
- IDSLModule-24
- BufferControlModule
- BufferControlModule2
- V.35-WAN
- DS3FR-WAN
- DS3ATM-WAN
- QuadT1-WAN
- IMAWANModule
- G.liteModule-24
- DualT1-WAN
- SDSLModule-8
- T1Module-12
- SystemControlModule3
- ADSLMultiModeModule-24A
- ADSLMultiModeModule-24B
- SONETSingleMode-WAN
- SONET MultiMode--WAN
- GSHDSLATMModule-24
- GSHDSLPacketModule-24

Port Classes

These classes designate the various physical interfaces.

- Port
- SDSLPort
- EthernetPort

- V.35Port
- RS232Port
- DS3FRPort
- IDSLPort
- DS3ATMPort
- T1Port-WAN
- T1Port
- G.litePort
- G.dmtPort
- T1Port-LC
- SONETPort
- GSHDSLPort
- IMAPort

Link Classes

These classes designate the logical data connections (links) connected to the physical interfaces.

- Link
- LCPortLink
- EthernetLink
- FrameRelayLink
- FrameRelayPVC
- ATMLink
- InterShelfTrunk

CPE Classes

These classes identify various types of connected CPEs.

- CR
- CR201-10x
- CR201-30x
- CR201-SDSL
- CR201-IDSL
- CPE-SDSL
- CPE-IDSL
- Netopia-SDSL
- CPE-T1
- CPE-GSHDSL

Support Classes

These classes identify supporting entities in the system.

- Support
- PowerModule
- FanModule

Logs

All events that occur on the CE200 are captured in three locations:

- The Event Log contains all monitored events, whether a major alarm or a notification.
- The Alarm Table contains all alarms.
- The Audit Log contains a record of attribute changes: the exact command entered to effect each change, and the IP address and name of the operator originating the command.

Event Log

As events occur, the information is logged, and can be viewed in real time or captured to a file. The CE200 maintains a running record of the most recent 1,000 events since the system was last reset, and by default, the log will display all of them. A list of 1,000 events can be unwieldy to use, however, so the system will allow you to specify how many events (looking backward from the most recent) to display.

There are two ways to see the contents of the Event Log. You can view the log in real time by connecting to the system either directly or through Telnet and entering the command:

```
CRAFT (1.2)> elog
```

If you enter the command without a numeric modifier, the entire contents of the log file will print to your screen, and additional events will displayed as they occur, for as long as you remain in the elog mode.

For a more usable display, you can specify the number of events to display by entering a number from 1 to 1000 as part of the elog command. For example,

```
CRAFT (1.2)> elog 25
```

will return the 25 most recent events. If you enter a number greater than the capacity of the log file (>1,000), the elog will display 1,000 records. If you will be dealing with more data than can be displayed on a single screen, you can also capture the information by logging your Telnet session and saving it for later analysis.

To exit the elog function, press **Esc**.

Event log information can also be viewed through the *cmTrapEvent* group (described in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual).

Alarm Log

Alarm messages are written to the event log, and also to a separate alarm table that is updated every time the alarm status changes. Once an alarm condition is cleared, it is dropped from the table.

Like the Event Log, two different methods are available for viewing information about current alarms. To view the Alarm Table in real time, connect to the system either directly or through Telnet and

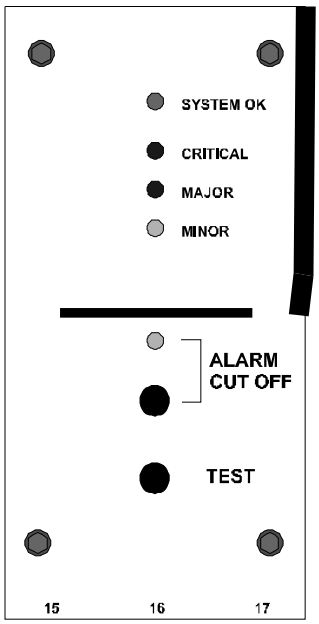
enter the command, `alarms`. A list of all current (uncleared) alarms will be displayed. To exit the Alarm Table, press **Esc**.

The second method is to view Alarm table information through the *cmTrapAlarm* group (described in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual).

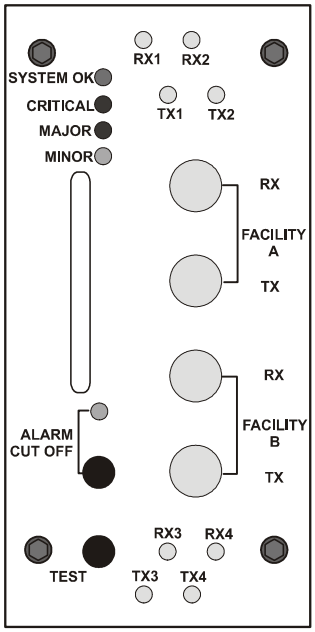
Alarm Module LEDs

When an alarm does occur, not only does it generate a message for entry in the AlarmTable, the alarm also causes a signal to be sent to the CE200 Alarm Module (either the standard Alarm Module or the DS3 Protection Switch Alarm Panel).

When the alarm data is received, an LED turns red (for a critical or major alarm) or yellow (for a minor alarm), depending on the severity, and two sets of corresponding dry contacts close. These contacts can be used to control external alarm devices, either local or remote.



Standard Alarm Panel



DS3 Protection Switch/ Alarm Panel Module

When all alarm conditions of a given severity have been corrected, the corresponding alarm-relay contacts will open, the Alarm LED will extinguish, and (if no other alarms are outstanding) the System OK LED again turns green.

Two front panel switches are provided to enable an on-site (CO) user to silence an active alarm (Alarm Cut Off) or to close the alarm contacts momentarily and test the alarm relays and LEDs (TEST).

Audit Log

When isolating the cause of a problem, it can be helpful to know what might have triggered it. On systems with an SCM-2 or SCM-3, the CE200 software includes uploadable logfiles containing a record of the most recent attribute changes (up to 1,000 events) on this CE200. Attribute changes are generally changes that result from a Set command.

With the time-stamped AttributeChange known, you can logically infer the precipitating command. The Audit Log also lists the name and IP address of the operator initiating the change, as well as the command context (such as Telnet or SNMP) and the privilege level of the operator. The addition of this information invests the Audit Log with an implicit role in maintaining system security.

The full audit log information base consists of four files stored in the Logs directory on the CE200 flash system (the Q: (IDE) drive) of the SCM-2 and SCM-3:

- *audit.txt*—The audit log that is currently filling; contains the most recent AttibuteChange events.
- *audit_count.txt*—This file contains a running count of the events in the *audit.txt* file. When this number reaches 500, new *audit.txt* and *audit_count.txt* files are created and the old files are renamed.
- *audit.bak*—The previous *audit.txt* file, now filled with the 500 AttributeChange events immediately preceding those in the current *audit.txt* file.
- *audit_count.bak*—The previous *audit_count.txt* file, now indicating the total number of records (500) in the *audit.bak* file. When the next cycle is completed and the current *.txt* files are filled and renamed, the current *.bak* files are overwritten.

The audit log files are accessible only through FTP, and only by an operator with Security level privilege. Once uploaded, the raw files have the format of one record per line, with fields delimited by a vertical bar (|), but this can be easily re-formatted using a spreadsheet program.

Traps

In the CE200, every internal event generates an SNMP *trap*. A trap is a record of the event that is captured and sent to the configured trap destinations for the system; it contains the same information as the event. For a listing of all currently supported traps associated with each event, see Appendix C.

There is no one-to-one mapping between events and traps. Because of the way alarms are cleared internally, one or more events may translate into the same trap. In these cases, one or more data fields can be used to distinguish the traps. For example, both the DLCIStateDisabled and DLCIStateEnabled events result in a frDLCIStatusChange trap being generated.

To view a chronological list of the 1,000 most recent traps (events) you can use the elog or cmTrapEvent group as described above.

The same information is also available sorted by the type of trap or event. To access this list, use the cmTrapType group as described in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Traps can be sent to as many as ten different SNMP managers. The list of those designated to receive traps is contained in the cmTrap-Destination table in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Alarms

Alarms constitute one subset of events. When generated, an alarm message contains the following information:

- Name of the alarm (such as FanFailure)
- Source (module, port, link)
- Severity (Critical, Major, Minor)
- Time the event occurred
- Other event-specific information

Alarm/Trap Severity

Every trapped event is assigned a *severity*. There are five levels of severity. If a trapped event occurs which is assigned one of the three most severe levels, it triggers an Alarm. The alarming levels are:

- *Critical*—A service-affecting condition that requires immediate corrective action. Used when the entity is totally out of service and its capability must be restored.
- *Major*—A service-affecting condition that requires urgent corrective action. Used when there is a severe degradation of the entity's capability, which must be restored.
- *Minor*—An event or condition that does not affect service, but corrective action should be taken to prevent a more serious fault.

Events assigned to one of the two lower severities do not trigger alarms; in many cases they represent a change of state which removes a previous alarm. The non-alarming severities are:

- *Warning*—An informational notice intended to call the operator's attention to a condition indicating a vulnerability or future risk.
- *Notification*—An event or condition that either clears an alarm, or that may simply be advisory to the operator, such as a login or logout notice. This is the lowest severity level.

Assignment of Severity Levels

In individual systems, large numbers of events may occur that have a default severity level that triggers an alarm, but which are not considered alarming in that system's operational model. For

example, a LinkDown alarm may be generated if the CPE at a subscriber location is powered down or disconnected. The large number of resulting alarms may disrupt normal operations or even mask the occurrence of a "real" alarm. Conversely, some systems may wish to elevate the severity level of an event that has particular importance for that system.

A capability in this release of the CE software allows you to select and assign the severity appropriate for your system to any trapped CopperEdge event. For details see *cmTrapEventConfigTable* in your *CopperCraft Reference and MIB Definitions*.

Alarm Clearing

The phrase "clear an alarm" describes an occurrence (a CE200-generated event or manual action) that causes visual and audible alarm indications to disappear (usually but not always corresponding to correction of the underlying problem).

Some alarms (such as BoardDown) are cleared by the CE200 when a corresponding event (in this case, BoardUp) occurs. Other alarms (such as RoleChanged) have no such corresponding event. In the latter case, you must manually clear the alarm by setting the cmAlarmTable as discussed in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Manually clearing an alarm in the CE200 does not imply fixing the problem. For instance, if a RoleChange alarm occurs (indicating that a redundant SCM assumed control of the system) the corrective action could be quite involved (for instance, replacing a failed board or re-routing a WAN connection). Clearing the alarm merely causes the alarm unit to no longer display alarm indications.

You can manually clear alarms that support automatic clearing. For example, you would do this if a line card is being removed from a system with no immediate plans to replace it. The alarm that would result from the subsequent BoardDown event could be suppressed by manually clearing the alarm. However, we recommend that you do not manually clear such alarms very often, because information about the status of your system will be discarded in the process.

When an alarm is cleared, a *Notification* message is displayed showing the new operational condition. Not all Notification messages are alarm-clearing; some are simply informational or advisory. A *Warning* is essentially a Notification, but with the added dimension of calling the operator's attention to a situation that may include some vulnerability or future risk.

Trap and Event Filtering

In the normal course of operation, and assuming the factory default severities are in force, the CopperEdge event log will soon be filled to its capacity of 1,000 events. Similarly, minor alarms can be expected to occur as the CE200 interacts with the real world of the network.

Although huge amounts of data are generated, it may be of little use to you, the system operator, unless it can be accessed selectively and/or the extraneous or unimportant data can be discarded. Thus, two filtration mechanisms have been introduced to the traps and events functionality of your CopperEdge:

- *Trap Filters*—A system by which you can prevent those traps which are of no concern in your system from being sent to configured Trap Managers. Up to 32 separate filters can be set to screen trapped events based on Type, Class ID, Object Class, Severity Level, or any combination of these.
- *Event Log Filters*—Event Log filters, in some ways, fulfill the opposite purpose of the Trap Filters. With event log filters, you can collect events based on the criteria you specify, such that the event must match: type of event, PII of the reporting interface, severity (EQ or GT), events occurring between specified start and end times, or the most recent number of events that match whatever combination of the previous criteria you impose. Up to 10 separate filters can be set.

Neither of these filters affects the contents of the Event Log, only whether or not they are reported as traps, or whether or not a particular event is included in a filtered collection.

To configure a Trap Filter, see the CopperCraft Reference and MIB Definitions volume. For Trap filters you will configure *cmTrapTrapFilterTable* [No, that's not a typo. Ed.], *cmtrapDestinationTable*, and *cmTrapSummary*. For EventLogFilters, you will use *cmTrapEvenFilterTable*, and *cmTrapFilteredLog*.

Recovery from a DS3 Failover

If your CE200 is equipped with a DS3 Protection Switch/Alarm Panel and it has detected the failure of a primary DS3 WAN module, the Backup side now controls operation and traffic (it has assumed the Primary role).

The circuits configured across the original DS3 have automatically switched over to the Backup DS3 and will continue to operate normally.

The term Preferred *refers to modules located in slots 2 to 5 on the left side of the CE200 chassis, and* Backup *refers to the redundant modules located in slots 14 to 17 on the right side of the chassis.* Primary *refers to the side that is controlling operation and traffic.*

To restore control to the Preferred side of the CE200, follow these steps:

1. Log in to the Backup side of the CE200.
2. Correct the problem on the Preferred side.
3. Enable redundancy for the system and save the configuration:

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

The save process will take a few minutes while the Preferred side restarts.

4. Tell the System Control Module (SCM) on the Backup side to relinquish control:

```
CRAFT (1.2)> set cmsystem command=relinquish
```

5. Log in to the Preferred side of the CE200 and verify that the Preferred SCM [1.2] now shows Role= Primary, and the Backup SCM[1.15] shows Role=Secondary.

```
geta cmboard

Index               ObjectClass         OperState     HwType
HwVersion           SwVersion           PromVersion   Role
ClusterRole         UpTime              NumPorts      FileName
FileDate           ConfigChange        Command       SerialNumber
Information

Instance: [1.2.0.0]
1.2.0.0             SystemControlModule Enabled
SystemControlModule
R 1.0               4.0.72              4.0.59        Primary
Master             0 day 0 hour 4 min  5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.3.0.0]
1.3.0.0             DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3               4.0.72              2.5.11        Active
NotApplic           0 day 0 hour 3 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None
MI4FT03SX_003470105
""

Instance: [1.4.0.0]
1.4.0.0             DS3FR-WAN           Enabled       DS3FR-WAN
R 1.0               4.0.72              1.35.19       Active
NotApplic           0 day 0 hour 3 min  1             ds3fr.bin
Oct 13 2002, 00:25: 0                   None          MI3SM01GL
""
<etc.>

Instance: [1.15.0.0]
1.15.0.0            SystemControlModule Enabled
SystemControlModule
R 1.0               4.0.72              4.0.59        Secondary
Master             0 day 0 hour 10 min 5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M

Instance: [1.16.0.0]
1.16.0.0            DS3ATM-WAN          Disabled      DS3ATM-WAN
R 1.3               4.0.72              2.5.11        Active
NotApplic           0 day 0 hour 9 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None
MI4FT03SV_003470105
""
```

6. Enable redundancy on the CE200 and save the configuration (remember, redundancy is automatically disabled with each switchover).

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

The save process will take a few minutes while the Backup side restarts.

7. Verify that the WAN modules on the Backup side show Operstate=Enabled:

```
geta cmboard [1.15]

Index               ObjectClass         OperState     HwType
HwVersion           SwVersion           PromVersion   Role
ClusterRole         UpTime              NumPorts      FileName
FileDate           ConfigChange        Command       SerialNumber
Information


Instance: [1.15.0.0]
1.15.0.0            SystemControlModule Enabled
SystemControlModule
R 1.0               4.0.72              4.0.59        Secondary
Master             0 day 0 hour 10 min 5             P:/ce200/scm
Oct 13 2002, 00:39: 0                   None          ****
RAM 256Mb;Flash 64M


Instance: [1.16.0.0]
1.16.0.0            DS3ATM-WAN          Enabled       DS3ATM-WAN
R 1.3               4.0.72              2.5.11        Active
NotApplic           0 day 0 hour 9 min  1             DS3ATM.HEX
Oct 13 2002, 00:31: 0                   None
MI4FT03SV_003470105
""
```

Save Configuration Failure

If the system configuration is not successfully saved when you log out of the CE200, the following message is displayed and the logout process is stopped:

```
CE configuration has been changed.
Save the configuration?(Y / y)>y
.....
CE configuration save failed.
```

To see information about the failure, look for the ConfigWriteFailed event in the event log (it will be at or near the end of the event log).

```
elog

2002/10/03-14:37:03 ConfigWriteFailed
  cmSystem           ConfigFileName       = config.tgz
  cmTrapEventTable   Type                 = ConfigWriteFailed
  cmTrapEventTable   ObjectClass          = System
  cmTrapEventTable   ClassId              = 1.0.0.0
  cmTrapEventTable   SeqNum               = 11
  cmTrapEventTable   TimeTag              = 2002/10/03-
                                            14:37:03
```

```
    cmTrapEventTable   Severity            = Information
```

To see the entire description of the event, find the instance number the ConfigWriteFailed event in the cmTrapEvent group:

```
geta cmTrapEvent

SeqNum        Type              ObjectClass        ClassId
ProbableCause Severity          TimeTag            Text

Instance: [11]
11            ConfigWriteFailed  System            1.0.0.0
0             Information       2002/10/02-10:22:23 Unable to
```

To view the full description in the cmTrapEvent group for that instance:

```
geta cmTrapEvent [11]

Group: cmTrapEventTable
Instance: [11]
SeqNum              =  11
Type                =  ConfigWriteFailed
ObjectClass         =  System
ClassId             =  1.0.0.0
ProbableCause       =  0
Severity            =  Information
TimeTag             =  2002/10/02-10:22:23
Text                =  Unable to open the file
```

See Appendix C, *Events and Alarms*, for more information about the ConfigWriteFailed event.

Terminate an Operator Session

An operator may forget to log out and a script may still be running, preventing the CE200 from automatically logging the session out after a time-out expires. Using the cmActiveSession group, you can terminate any Craft, FTP, or SNMP sessions, whether they were opened locally or through the Radius authentication server. But to terminate a session, an operator must have security level privileges.

First, list all active sessions to find the instance of the one you want to terminate:

```
geta cmactivesession

SessionID           OperatorName        Context
Privilege
IpAddress           StartTime           IdleTimeout
RowStatus

Instance: [1982]
1982                private             SNMP
Provision
10.64.20.162       18 day 18 hour 33 m 900               Active

Instance: [1983]
1983               ce200              Telnet             Security
10.64.20.249        18 day 18 hour 36 m 900               Active

Instance: [1984]
1984                public              SNMP               View
```

```
10.64.40.66        18 day 18 hour 36 m 900                Active

Instance: [1985]
1983              ce200             Telnet             Security
10.64.20.234       18 day 18 hour 36 m 900                Active
```

You can optionally view the information for that session only:

```
get cmactivesession [1983]

Group: cmActiveSessionTable
Instance: [1983]
SessionID            =  1983
OperatorName         =  ce200
Context              =  Telnet
Privilege            =  Security
IpAddress            =  10.64.20.249          .
StartTime            =  18 day 18 hour 36 min 27.0 sec
                        (2002/11/18-09:19:39)
IdleTimeout          =  900
RowStatus            =  Active
```

To terminate the active session, delete the row from the table:

```
set cmactivesession [1983] rowstatus=destroy

Set successful
```

Any application running in the specified active session is halted, the session is terminated, and this event is listed in the Event log as a termination of the session. The following message is displayed on the operator's screen:

```
2002/11/18-09:36:30  USER LOGGED OUT
```

Repeat the **geta cmactivesession** command to check that the terminated session is no longer in the table.

The cmActiveSession table will not let you accidentally terminate your own session. If you enter your session's instance, the following message will be displayed:

```
set cmactivesession [1985] rowstatus=destroy

Delete failed
Group: cmActiveSessionTable
```

DHCP Problems

The proper functioning of the DHCP feature depends on the correct configuration of the CE200 as well as of the upstream router and DHCP server.

If the CPE or premise LAN host does not receive the expected response for a DHCP request that it originated, follow these diagnostic steps.

In general:

1. Check the cmDHCPTable to see if an entry exists for DSL PII.
2. If it exists, verify that the Function object is set correctly: DHCPRespond, DHCPRelay, or DHCPForward.

For DHCP Relaying:

1. Verify that the upstream router is able to perform the DHCP Relay or DHCP Server function.
2. If the netmodel is IP, verify that the DHCP Server's IP address configured in the cmDHCPTable is reachable from the CE200.

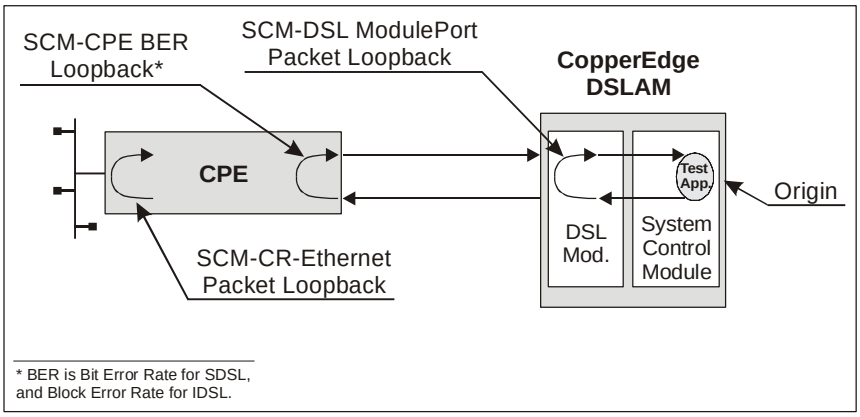
For DHCP Relay and Forwarding:

Check the cmDHCPTable and verify that the CircuitID object is correct. This is used by the upstream DHCP server for generating the appropriate DHCP responses.

Loopbacks

The CE200 can be configured for a number of loopback tests useful not only in isolating failures but also in evaluating line performance over various segments of the end-to-end circuit. In the current CE200 product release, loopbacks are available for test and evaluation of the following: SDSL/IDSL ports; DS3 ATM and Frame Relay ports; and DS1 ports.

Three DSL loopbacks (between the CE200 main processor and the subscriber CPE) are controlled through the cmLoop group. All of the loopback tests are packet-oriented, except the SCM-CPE BER loopback, which is a bit-error rate test. The cmLoopHistory group provides information on the ten most recent tests run through the cmLoop group. For information on configuring cmLoop and cmLoopHistory, see the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual. The following diagram illustrates the DSL loopback tests supported on the CE200.



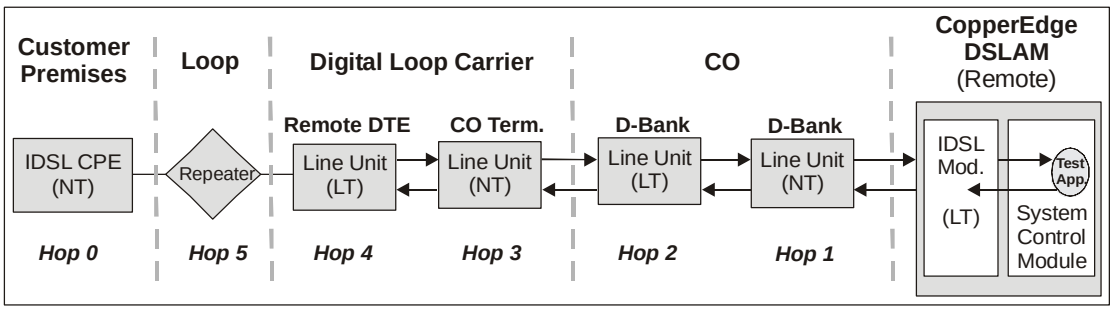
DSL Subscriber-Side Loopbacks

EOC Loopbacks

The EOC Loopback is used for troubleshooting IDSL links. Using a small portion of the ISDN band called the *Embedded Operations Channel* (EOC) is helpful in isolating IDSL line problems by allowing you to test the various segments between the CE200 and the CPE that comprise an IDSL loop. Using the EOC, each network element (that is, each segment or hop) reports its presence and is counted by the CE200. With the number of elements (number of

hops) between the CE200 and the CPE known, specific network elements can then be placed into loopback mode by addressing the test command to the device that is one, two, three (or more) elements downstream from the CE200. The following diagram illustrates the concept. Note that the endpoint CPE is always referred to as Hop 0 or Element 0, regardless of how many intervening elements may be present.

The IDSL Module includes a CPE monitor that detects the presence or absence of a CPE through the EOC, and a standard IDSN network element counter. The monitor sends both standard EOC messages and a Copper Mountain specific message that the CPE use to learn the configured data rate. If for some reason the EOC path fails, the CPE may still be able to determine the data rate based on receiving Copper Mountain CMCP packets. The monitor state machine is called once a second.



IDSL Loop with Multiple Network Elements

If the entire IDSL loop is intact, EOC will detect and report the presence of the CPE at the downstream end of the loop. If the link is *not* complete, EOC will attempt to count the network elements that are in place and functioning, and report the number through the cmIdslModem NetworkElements object. Thus, if you know how many hops are supposed to comprise a link, then isolating the area of the malfunction becomes much less complicated.

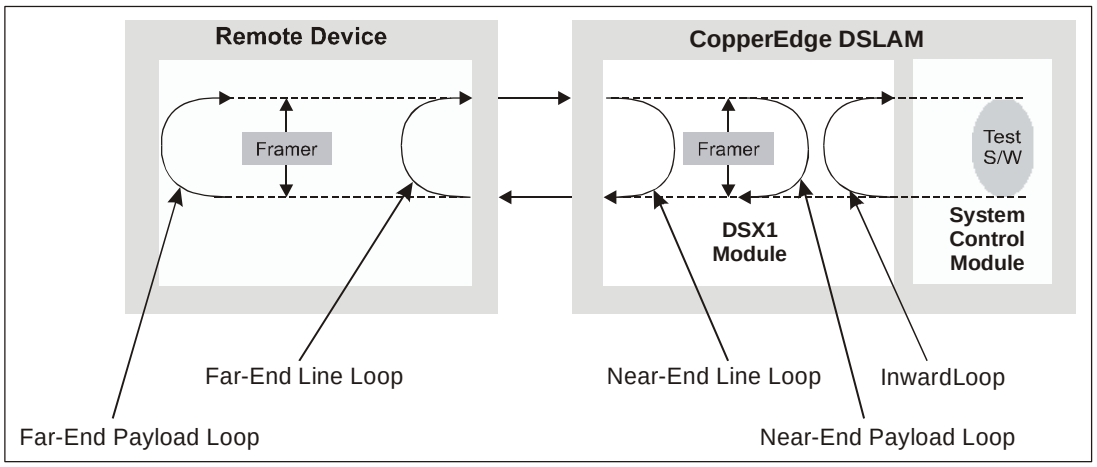
Similarly, if the CPE at the end of an IDSL link is able to train, but other performance factors are degraded or marginal, the SCM-DSLModuleEOC test lets you command any of the preceding network elements into loopback mode. By looping back from successive hops, you should be able to isolate the link segment where the problem is occurring. For details on configuring a CE200 for EOC loopback testing, see the cmLoop MIB description in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

DS1 Loopbacks

To configure loopbacks on DS1 SDSL Modules, or on Quad T1 WAN Modules, you may have to configure *dsx1Conig*, *cmDsx1Config*, and *cmLoop* tables.(See the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.)

A number of types of loopbacks are possible for DS1 boards: inward loops, near-end payload loops, near-end line loops, far-end line loops, and far-end payload loops. For DS1 SDSL Modules, you can also configure custom FDL loopbacks as described below (page 153).

For the inward loop test, the far-end line loop test, and the far -end payload loop test, the CopperEdge must generate test packets, using a generator on its System Control Module. For the near-end line loop test and the near-end payload loop test, a remote device must generate the test packets and send them to the CopperEdge.



DS1 Loopback Modes

After setting the loopback tests for DS1 boards with the dsx1ConfigTable group of objects, you are ready to send the test packets and then read the results of the tests, using the cmLoop group of objects.

Setting up loopback and then starting, interrupting, and completing loopback tests are trapped events. As such, all of them are reported to the trap receiver.

The current CE200 product release does not support multiple simultaneous loopback tests. Only one test can be run at any one time. Moreover, the CE200 loopback test software makes no assumptions and has no prior intelligence about the loop under test. If the loop is open, even if only temporarily because up- or downstream equipment is busy, the test will run but the CE200 will report errors.

Special Loopback Options for DS1 SDSL Modules

There are a number of alternative methods you can use for performing loopback tests on a DS1 DSL port. With the standard method, you choose either *LineCode* or *PayloadCode* for the *SendCode* object on the *Dsx1ConfigTable,* and then run a selected test pattern using one of the ds1 Types from the *cmLoopTable*. In contrast, the Custom FDL method allows you to send non-standard FDL codes to devices that are to be looped. Finally, there are two

loop tests, *scm-DSLModulePort*, and *scm-CPE-BER*, that integrate the loop setup and teardown with the transmission of the test patterns.

Standard Line Loops

To use the standard method with line code, proceed as follows:

Set Dsx1Config SendCode=LineCode

set cmLoop interface=*pii* Type=*D* duration=*n*

where "*pii*" is the PII of the interface under test, "*d*" is the selected DS1 pattern: *ds1All-0s*, *ds1All-1s*, *ds1QRS*, or *ds13in24*), and "*n*" is the number of seconds the test will run.

set cmLoop action=start

get cmLoop interface=pii

Observe the results in the listing for the *TermCode*, *TxCount*, and *RxErrors* objects.

Set Dsx1Config SendCode=reset

Standard Payload Loops

To use the standard method with payload code, proceed as follows:

Set Dsx1Config SendCode=payloadCode

set cmLoop interface=*pii* Type=Ds1qrs duration=*n*

where "*pii*" is the PII of the interface under test, and "*n*" is the number of seconds the test will run (note that Payload Loops require the ds1QRS pattern)

set cmLoop action=start

get cmLoop interface=pii

Observe the results in the listing for the *TermCode*, *TxCount*, and *RxErrors* objects.

Set Dsx1Config SendCode=reset

Custom FDL Loopbacks

The procedure for the custom method is as follows:

1. On the *cmDsx1ConfigTable*, set the *FdlCode* object to the proper value to set up the loop on the far end device (for many SmartJack devices, the value is 18).
2. On the *Dsx1ConfigTable*, set the *SendCode* object with the value, *OtherPattern*. This will send the FDL code toward the far end, and set up the loop.
3. On the *cmLoopTable*, set the *Type* object for the desired value: *ds1All-0s*, *ds1All-1s*, *ds1QRS*, or *ds13in24*. Note: For payload Loop testing, the only permissible Type is *ds1QRS*.
4. On the *cmLoopTable*, set *action=start* to begin the test.

5. When the test is complete, issue a *get cmloop* for the interface and observe the results in the listing for the *TermCode*, *TxCount*, and *RxErrors* objects.
6. On the *cmdsx1ConfigTable*, set the FDL object with the proper value to tear down the loop between the CE200 and the device at the other end (for many SmartJacks the value is 36).
7. On the *Dsx1ConfigTable*, set the *SendCode* object with the value, *OtherPattern*. This will send the FDL code toward the far end and tear down the loop.

Integrated cmLoop Test

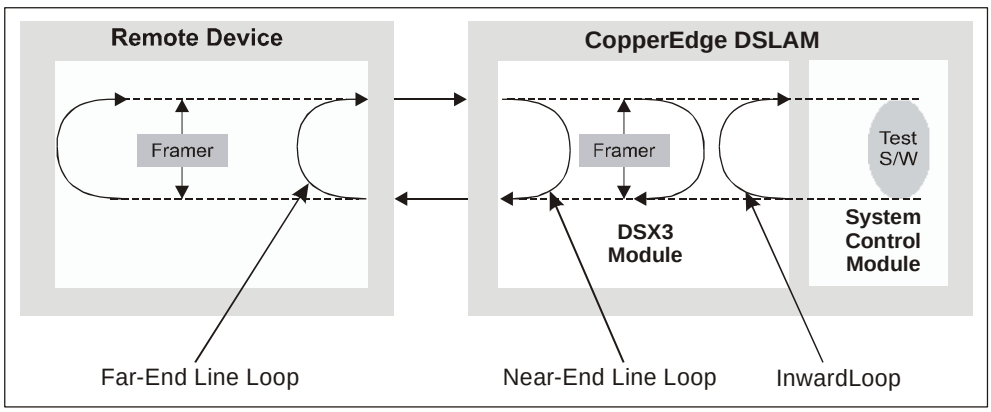
The procedure for the Integrated method is as follows:

1. On the *cmDsx1ConfigTable*, set the *Type* object to either *scm_DSLModulePort* (for a near-end inward loop test), or *scm-CPE-BER* (for a far-end payload loop test).
2. Set the *cmLoopTable* to *Action=Start* to begin the test.
3. When the test is complete, issue a *get cmloop* for the interface and observe the results in the listing for the *TermCode*, *TxCount*, and *RxErrors* objects.

DS3 Loopbacks

For DS3 boards, either ATM or Frame, you must configure loopbacks through the dsx3Config group. (See the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.)

Three types of loopbacks are possible for DS3 boards: inward loops, near-end line loops, and far-end line loops. For the inward loop test and the far-end line loop test, the CopperEdge must generate test packets, using a generator on its System Control Module. For the near-end line loop test, a remote device must generate test packets and send them to the CopperEdge.



DS3 Loopback Modes

After setting the loopback tests for DS3 boards with the dsx3ConfigTable group of objects, you are ready to send test pack-

ets and then read the results of the tests using the cmLoop group of objects.

Setting up loopback and then starting, interrupting, and completing loopback tests are trapped events. As such, all of them are reported to the trap receiver.

The current CE200 product release does not support multiple simultaneous loopback tests. Only one test can be run at any one time. Moreover, the CE200 loopback test software makes no assumptions and has no prior intelligence about the loop under test. If the loop is open, even if only temporarily because up- or downstream equipment is busy, the test will run but the CE200 will report errors.

Evaluating SDSL Loops

The CE200 also includes a special test capability which examines certain analog properties of the SDSL loop. The cmSDSLTest group (described in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual) provides two tests that are useful for evaluating and troubleshooting SDSL loops. The SeekMaxRate test can help to establish the fastest data rate that a specified DSL loop will sustain with a connected CPE. The Loop Profile test examines the overall condition of the loop between physical ports on the CE200 and their respective CPE. The Loop Profile test can also be used to verify the integrity of the SDSL loop between the CE200 and the first 500 feet of the loop, typically the portion of the loop within the CO itself.

Typically, a loop test is run at the time of installation in order to establish a baseline for future assessment. Data from the test is kept on file and can then be used to perform line-quality analysis or for troubleshooting if needed.

To check continuity over the entire length of the SDSL loop, you need the cooperation of an assistant at the CPE site who can communicate in real time with the person operating the CE200. To perform this test, follow these steps:

1. Tell the assistant at the CPE site to disconnect the DSL line from the CPE, thus opening the loop.
2. At the CE200 console, enter the following command:

   ```
   CRAFT (1.2)> set cmsdsltest [pii of the sdsl port]
          type=type_of_test action=start
   ```

3. At the CE200 console, wait five or ten seconds, then enter a **`get cmsdsltest`** command.

   Observe and record the value displayed in the ContinuityTestResult field.

4. At the CE200 console, while observing the value displayed in the ContinuityTestResult field, tell the assistant at the CPE site to apply a short between pins 2 and 3 of the RJ-11 plug at the end of the DSL line.
5. At the CE200 console, wait five seconds and repeat the **`get cmsdsltest`** command.

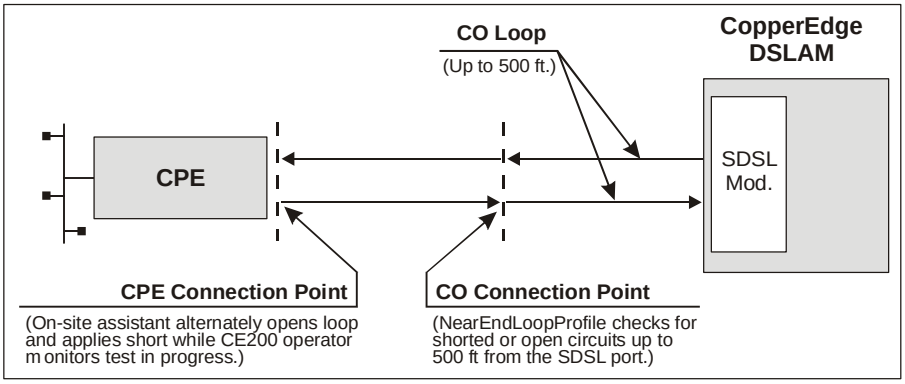
There should be a perceptible change in the value of the integer displayed in the ContinuityTestResult field; observe and record the new value.

6. At the CE200 console, while observing the value displayed in the ContinuityTestResult field, tell the assistant at the CPE site to remove the short from the RJ-11 plug at the end of the DSL line.

7. At the CE200 console, wait five seconds and repeat the **`get cmsdsltest`** command.

   The value of the integer displayed in the ContinuityTestResult field should return to the value you noted in step 3.

When using the cmSDSLTestTable for troubleshooting, the absolute numerical values displayed in the above procedure have no meaning in themselves, and will vary depending on the length of the loop and a number of other physical and electrical factors. The significant points are the change in the value as the line is monitored while being opened or shorted, and whether and by how much the resulting values deviate from the baselines recorded at the time of installation. The following diagram shows the cmSDSLTest function.



SDSL Loop Test Functionality

Built-in Performance Monitors

In addition to hard failures, the CE200 also collects information that can be helpful in diagnosing marginally performing lines or other link-related hardware. Statistics related to SDSL and IDSL loops and statistics related to LAN and WAN links are collected as part of the routine operation of the system. You can configure the threshold points at which the various indicators trigger an alert. As part of a regular program of system maintenance, these statistics can help to provide early warning of link-related problems.

Statistics on system performance can also be collected for selected time intervals using the Bulk Statistics feature of the cmMaintCmd group (described in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual).

IMUX Configuration Issues

For setting up IMUX bundles, the CE200 offers a number of features to help identify and isolate configuration and/or wiring errors. Two types of warnings may be received:

- *Error messages*—These messages provide an immediate indication that some configuration rule has been violated. For example, if an operator, using cmBundle, tries to assign a port that has already been configured with a netmodel or a filter to a bundle.
- *TrapEvent and TrapAlarm Tables*—These tables indicate an improper step in a procedure. For example, an alarm occurs when a CPE that is not IMUX-capable trains on a bundled DSL port, such that the CE200 cannot change its status to Active (ready to receive and transmit data).

Problems Configuring Multilink Bundles

If you try to place ports in an IMUX bundle incorrectly and the CPE's ports are already trained, the system returns an error message. DSL ports that have already been configured as individual interfaces, and ports that have mismatched EndPointIDs, will trigger error messages. The error messages related to IMUX configuration are listed below. (For the full configuration procedure for IMUX bundles, see "Configuring an IMUX Bundle" on page 70.)

Multilink Bundle Error Messages

Message	Description	Action
CPE is not IMUX capable	You cannot assign a CPE to a bundle unless it has IMUX software in its code. When a CPE trains on a port, the CE200 knows whether it is IMUX capable.	
IMUX CPE has EndPointID inconsistent with existing members	You cannot assign a port to a bundle when the CPE trained on the port has an EndPointID that does not match the EndPointID of a port already in the bundle. All ports in a bundle must have the same EndPointID.	Enter the command **`getall cmEndPointPort`** to list all EndPointIDs in the CE200, starting with the first port on the first line card. Create a new bundle, and then either switch the mismatched port or attach the correct phone to it. Finally, create other ports for the other phone lines to the bundles.
IMUX CPE has EndPointID that matches member in different bundle	You cannot assign a port to a bundle when the CPE trained on the port has an EndPointID that matches the EndPointID of a port in another bundle. All ports in a bundle must have the same EndPointID.	Enter the command **`getall cmEndPointPort`** to list the EndPointIDs in the CE200, starting with the first port on the first line card. Either switch the port to the other bundle or attach the correct phone line to the port. Then add another port to the proper bundle and attach the correct phone line.

Multilink Bundle Error Messages

Message	Description	Action
Member PII must be a DSL port	You cannot assign a WAN or Ethernet port to a bundle.	
Member PII must be on same shelf as BundlePII	Member ports in the same bundle must be controlled by a common control module.	
Member Port must have NetModel set to None to be added to bundle	You must set the netmodel for the IMUX bundle. You cannot set the netmodel separately for each ports in the bundle.	
Member Port must not be configured in cmDHCP table	You must set the IMUX bundle for Dynamic Host Configuration Protocol. You cannot separately configure each port in the bundle for DHCP.	
Member Port must not be configured in cmFilter table	You must set the filters for the IMUX bundle. You cannot set filters separately for each port in the bundle.	
Member Port must not be configured in cmMember table	You must assign the IMUX bundle to a user group. You cannot separately assign each port in a bundle to a group.	
Member Port must not be under loopback testing	You cannot assign a port to a bundle when it is in loopback testing. Similarly, you cannot perform loopback testing on a port already in a bundle.	To perform testing on a port in a bundle, remove it from the bundle, perform the test, and put it back in the bundle.
PII value must be a DML Bundle	The PII for an IMUX bundle represents a logical, not a physical, interface. It must have the format, 1.51.n, where n is any number from 1 to 63 inclusive.	
Port cannot be configured into multiple entries in a bundle	A bundle can have as many as four ports assigned to it. Each port must have a different PII. The ports can be on the same line card or on different line cards.	
Port is configured in a different bundle	You tried to assign the same port to different bundles. You cannot assign the same port to different bundles.	Enter the command **`getall cmBundle`** to list all IMUX bundles on the CE200 and all ports assigned to them.
Valid RowStatus values: Active, Destroy	The default value for RowStatus is Active. The only other acceptable value, Destroy, allows you to delete the bundle. You can delete the bundle without deleting ports from the bundle.	

Problems during CPE Training

If there is no CPE connected and trained on a port, any associated configuration and wiring errors will not be immediately apparent. It is only when a CPE is connected and attempts to train that the EndPointID objects are reported and inconsistencies can be detected. Without a connected and trained CPE, the CE200 simply accepts any assigned ports as part of the IMUX bundle, registers their EndPointIDs as 0.0.0.0.0.0.0.0.0, and sets the port status to WaitForAdd.

Later, if a CPE is connected and attempts to train, the CE200 can determine if the CPE is IMUX capable, and if it is, it can then compare EndPointIDs to see if they match. If the EndPointIDs of the ports in a bundle don't match, the CE200 sends a trap to the TrapEvent log with a resulting alarm. If this occurs, the ports remain in their previous WaitForAdd status until the conflict between the physical wiring and the logical bundles is resolved.

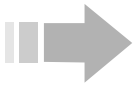
The TrapAlarm messages resulting from this conflict are listed below. To review the configuration of existing bundles and their member ports and isolate the source of the conflict, use the cmEndPointPort group. (See the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.)

CPE Training Error Messages

Message	Description	Action
EndPoint Conflict, CPE in bundle is not IMUX capable	A CPE has trained on a bundle that does not have the appropriate IMUX software. Typically, the CPE has a single port instead of multiple ports.	
EndPoint Conflict, IMUX CPE has EndPointID inconsistent with existing members	When the CPE attached to the port trains, the CE200 discovers that it already has an active port in the bundle with a different EndPointID.	Either replace the port in the bundle, making sure that you have the correct phone line attached, or keep the port but switch the phone line.
EndPoint Conflict, IMUX CPE has EndPointID that matches member in different bundle	When the CPE attached to the port trains, the CE200 discovers that it has a CPE with the same EndPointID attached to a port in another bundle.	Either switch the mismatched ports between the bundles, or switch the phone lines attached to the ports.

Diagnostic Port

The Diagnostic port on the front panel of the System Control Module is for CopperMountain personnel. If you have a problem with your system, a Copper Mountain Technical Support engineer may ask you connect to the Diagnostic port to the serial port of a local computer. The Diagnostic port does provide access to certain events related to the functioning of the CE200, but not in a form readily usable by an operator.



N O T E

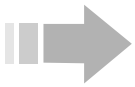
Copper Mountain recommends that you not connect to the Diagnostic port except as directed by Copper Mountain Customer Support. Restarting a computer with one of its serial ports connected to the Diagnostic port may cause the CE200 to reset. Also, certain service options that are available only through the Diagnostic port (such as options that facilitate recovery from software lockups) may cause undesirable results.

If you connect a cable to the DIAG port, it is crucial that the connected line be noise free and that it not be subject to receiving control characters, whether intentional or the result of spurious signals. The boot process can be interrupted or otherwise negatively affected if such signals are introduced to the DIAG port.

Packet Tracing

In troubleshooting local loops and WAN links, it may be helpful to display packet traffic over a selected interface as it occurs. Using the cmTracePIITable, an operator with security privilege can trace inbound packets from DSL port, WAN ports, or VCs as the data arrives at the CopperEdge.

Tracing packets arriving via a specified interface amounts to creating a row in a table, and the configuration uses the familiar RowStatus create/destroy mechanism to start and stop the trace operations. On a heavily congested high-capacity interface, the trace operation can result in a blizzard of data on the screen which may be gone before it can be observed. If you Telnet to the Craft session, you can use the Telnet log to record your trace.



N O T E

Use of the packet trace utility can impose a heavy performance penalty (throughput reduced by as much as 40 percent), so it should be applied judiciously.

For details on configuring a packet trace, see *cmTracePIITable* in the *CopperCraft Reference and MIB Definitions* volume.

Syslog

In certain situations, it can be helpful to collect diagnostic information from a number of different CopperEdge units (CE200 or CE150), such as when multiple units are exhibiting the same or

similar kinds of symptoms. To facilitate collecting this data at a central location, the CopperEdge unit can be configured as a UNIX Syslog client.

With the Syslog capability, diagnostic information from a number of CopperEdge units, which was previously available only at each unit's control module Diagnostic port, can be sent directly to a file on a UNIX-based server (Solaris or Linux) in standard Syslog message format.

While the Syslog capability requires that you configure the CopperEdge units and UNIX server as clients, the log file should not be considered a user-accessible function.

Since Syslog messages are identical in content and format to those at the CopperEdge Diagnostic ports, the same cautions apply: Syslog message information cannot be reliably interpreted by an operator, and should only be retrieved with the active participation of Copper Mountain Technical Support. Data from the log can then be returned to the factory where the contents can be analyzed by Copper Mountain technical staff.

Syslog messages occur as the result of certain normal operational events. The mere presence of data in the Syslog should not be construed as indicative of trouble with the system, and should not trigger a service call.

Configuring the Syslog Server

To implement Syslog capability on your system, you must first identify and configure a suitable UNIX machine as your Syslog file server. If you need more information on how to do this, consult the on-line UNIX manual:

- *man syslog*—Provides an overall description of the function and lists the available options and modifiers
- *man syslogd*—Describes the syslog daemon and lists its available options
- *man syslog.conf* —Provides server configuration details

When your Syslog server is operating and accessible to the network, client machines can be configured as described in the subsequent section.

Once you have begun collecting information in the Syslog, managing the records becomes a significant consideration. By default, most UNIX machines will purge or truncate log files at preset intervals. It is important that your server's "flushing" interval be set at a large enough value to allow a meaningful body of data to be gathered. We recommend at least one week.

Syslog Client Configuration

Four objects in the cmFile MIB group (SysLogAddr, SysLogPort, SysLogFacility, and SysLogPriority) control the Syslog capability on the CopperEdge unit. When configuring the CopperEdge unit as a

Syslog client, all you need to do is enter the IP address of the server. In most cases, the default settings of all of the other objects should be the appropriate ones. For example:

```
set cmfile syslogaddr=209.141.14.4
```

For more information about other Syslog configuration options for the CE200, refer to the cmFile MIB definition in the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

CPE Message Log Table

If you need to perform diagnostics for an IAD CPE (such as a CR408, CR508, or CR508T), you can log debug messages to a CPE log table, which you can view from the Craft prompt.

To log messages, you will use two MIB groups: cmCpeIADTable and cmCpeLogTable.

- cmCpeIADTable—Create and delete the log table entries using the LogAction, LogType, LogLevel, and LogDuration objects.
- cmCpeLogTable—View up to 1000 logged message entries. The first entry contains the oldest information.

In the following examples, we will turn on message logging and generate some messages.

To start logging messages:

1. First check the cmCpeBoardTable to see if the CPE supports cmCpeLogTable.

```
CRAFT (1.2)>  get cmcpeboard [cpe:1.13.2] groupmap

    Group: cmCPEBoardTable
    Instance: [CPE:1.13.2]
    GroupMap              =  3a.40.39.91.92.93.94.9b.
                             a5.a0.a7
```

   In the GroupMap, a5 indicates cmCpeLogTable.

2. Then check the cmCpeIADTable to see how the LogAction object is set.

```
CRAFT (1.2)> geta cmcpeiad
Index                CurrentIpAddr       CurrentCAIpAddr
SavedIpAddr
SavedCAIpAddr        NumVoicePorts       NumConnections
Command
TOSByte              LogAction           LogType
LogLevel
LogDuration          TotalMsgsLogged     Trace1Mask
Trace2Mask



Instance: [CPE:1.13.2]
CPE:1.13.2           10.10.1.5           10.10.1.1
10.10.1.5
10.10.1.1            8                   16                  None
10                   None                0                   0
30                   0                   0                   0
```

3. Since LogAction=None, there should be no messages logged in the cmCpeLogTable.

```
CRAFT (1.2)> geta cmcpelog
    No more instances
```

4. Turn on the debug message logging feature for the duration of the timer, set in the LogDuration object:

```
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2] logaction=startdebug
    Set Successful
```

When the timer expires, the LogAction object is automatically set to StopDebug and messages are no longer logged.

5. Select a log type by entering either the text or the bit number for the text.

```
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2] logtype=dhcp
    OR
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2] logtype=4
    Set Successful
```

6. Specify the message levels to be logged by entering either the text or the bit numbers for the text.

```
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2] loglevel=0+1+2+
    OR
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2]
       loglevel=info+error+detail+other
    Set Successful
```

7. (Optional) You can change the timer for logging messages (for example, from 30 minutes to 10 minutes):

```
CRAFT (1.2)> set cmcpeiad [cpe:1.13.2] logduration=10
    Set Successful
```

8. For this example, we can generate some messages to be logged by changing the link's administrative state to Down and then back to Up. This forces a resend of the DHCP request, causing messages to be logged.

```
CRAFT (1.2)> set iftable [1.13.2] admin=down
    Set Successful
CRAFT (1.2)> set iftable [1.13.2] admin=up
    Set Successful
```

9. Check the IAD table and note that the TotalMsgsLogged object has changed from 0 to 5.

```
CRAFT (1.2)> geta cmcpeiad
Index               CurrentIpAddr       CurrentCAIpAddr
SavedIpAddr
SavedCAIpAddr       NumVoicePorts       NumConnections
Command
TOSByte             LogAction           LogType
LogLevel
LogDuration         TotalMsgsLogged     Trace1Mask
Trace2Mask


Instance: [CPE:1.13.2]
CPE:1.13.2          10.10.1.5           10.10.1.1
10.10.1.5
```

```
10.10.1.1          8                  16                  None
10                StartDebug          4                  0+1+2+3
30                 5                  0                   0
```

To see more details:

```
CRAFT (1.2)> get cmcpeiad [cpe:1.13.2]

    Group: cmCpeIADTable
    Instance: [CPE:1.13.2]
    Index               =  CPE:1.13.2
    CurrentIpAddr       =  10.10.1.5
    CurrentCAIpAddr     =  10.10.1.1
    SavedIpAddr         =  10.10.1.5
    SavedCAIpAddr       =  10.10.1.1
    NumVoicePorts       =  8
    NumConnections      =  16
    Command             =  None
    TOSByte             =  10
    LogAction           =  StartDebug
    LogType             =  DHCP(4)
    LogLevel            =  INFO(0)+ERROR(1)+DETAIL(2)
                           +OTHER(3)
    LogDuration         =  30
    TotalMsgsLogged     =  5
    Trace1Mask          =  0
    Trace2Mask          =  0
```

10. Display the message entries in the log table:

```
CRAFT (1.2)> geta cmcpelog
Index               MsgNbr              MsgType
TimeStamp
Message

Instance: [CPE:1.13.2, 1]
CPE:1.13.2          1                   DHCP                587565
Sending DHCP reques

Instance: [CPE:1.13.2, 2]
CPE:1.13.2          2                   DHCP                587575
Got a DHCP Reply

Instance: [CPE:1.13.2, 3]
CPE:1.13.2          3                   DHCP                587575
ipaddr 10.10.1.5

Instance: [CPE:1.13.2, 5]
CPE:1.13.2          5                   DHCP                587575
caipaddr 10.10.1.1

No more instances
```

You can see that five new message entries have been added to the log table. The Geta command does not always show the entire message, so perform a Get to see more details for some of the entries.

```
CRAFT (1.2)> get cmcpelog [cpe:1.13.2,1]

    Group: cmCpeLogTable
    Instance: [CPE:1.13.2, 1]
    Index               =  CPE:1.13.2
    MsgNbr              =  1
    MsgType             =  DHCP
    TimeStamp           =  587565
    Message             =  Sending DHCP request

CRAFT (1.2)> get cmcpelog [cpe:1.13.2,4]

    Group: cmCpeLogTable
    Instance: [CPE:1.13.2, 4]
```

```
Index                =  CPE:1.13.2
MsgNbr               =  4
MsgType              =  DHCP
TimeStamp            =  587575
Message              =
```

To stop logging messages:

Turn off the message logging feature:

```
set cmcpeiad [cpe:1.13.2] logaction=stopdebug

Set Successful
```

The log table still displays the messages that were already logged. However, any new messages will not be added to the log table.

To delete entries from the message log:

Set the LogAction object to delete the entries.

```
set cmcpeiad [cpe:1.13.2] logaction=deletedebug

Set Successful
```

The TotalMsgsLogged object has been reset to 0.

```
geta cmcpeiad

Index                CurrentIpAddr       CurrentCAIpAddr
SavedIpAddr
SavedCAIpAddr        NumVoicePorts       NumConnections
Command
TOSByte              LogAction           LogType
LogLevel
LogDuration          TotalMsgsLogged     Trace1Mask
Trace2Mask



Instance: [CPE:1.13.2]
CPE:1.13.2           10.10.1.5           10.10.1.1
10.10.1.5
10.10.1.1            8                   16                  None
10                 DeleteDebug         4                  0+1+2+3
30                   0                   0                    0
```

All existing messages have been deleted from the cmCpeLogTable.

```
geta cmcpelog

No more instances
```

Set the LogAction object to None:

```
set cmcpeiad [cpe:1.13.2] logaction=none

Set Successful
```

A Note on ATM CPE

When troubleshooting an ATM CPE Virtual Circuit remember that a number of object groups may affect the interface. You may need to investigate: *cmIfaceTable* (cmif), *cmFrCircuitTable* (cmfrc), *frCircuitTable* (Alias: frpvc), *cmSubIfaceTable*, and *cmAtmVclTable*.

Remember also that ATM circuits between the CPE and CE must be enabled using the *frCircuitTable* (Alias: frpvc). Example: "set frc[1.6.15.16] admin=up. The frCircuitTable queries the SCM, the *cmAtmVclTable* queries the Line Card.

Restart Options

As mentioned elsewhere, the CE200 is designed to provide uninterrupted service, 24 hours a day. Resetting or restarting the system or any of its modules is service-affecting, and should only be done when warranted by a failure that cannot be resolved any other way.

Even so, situations may arise that can only be resolved by resetting a CPE, a module in the CE200, or by rebooting the CE200 itself.

CPE Soft Restart

If you suspect that a particular subscriber CPE is experiencing a software fault, it may be possible to recover by restarting the unit from the CE200. When this command is issued, the CPE should restart and retrain on the DSL port, and reload its software. Normal data transmission to and from the CPE will be interrupted while the restart sequence completes.

To restart a CPE, enter the following command:

```
set cmcpeboard [cpe:pii] command=restart
```

In this case, the PII is the permanent interface identifier of the CE200 DSL port to which the specified CPE is connected.

Module Soft Restart

You can also perform a soft restart on the CE200 System Control Module, any of the DSL modules, or a DS3 Frame module (but *not* to Buffer Control modules, to V.35 WAN modules, or DS3 ATM modules). If you suspect a software fault within a specific module, it may be possible to recover by restarting *only* the specific module. In some cases this may be a less drastic solution than restarting the entire chassis. That is, it might affect fewer subscribers (as in restarting only a single DSL module). Also, restarting a single module generally takes less time than a complete reboot.

To restart a specific module, enter the following command:

```
set cmboard [pii] command=restart
```

In this case, the PII consists only of the shelf and slot number. For example, a DSL module in slot 6 would be specified as [1.6].

A Restart command directed to a module that does not support individual module restart may return a "`Set Successful`*" message, but in fact the command will be ignored.*

Line Card Restart (Hardware Restart)

If, for some reason, you are unable to restart a DSL module using the normal software restart option, (**`set cmboard`**) a hardware restart is available through the CopperCraft interface that will achieve the same result.

To perform a hardware restart of a DSL module, connect to the system control module (in system equipped for redundancy, this must be the current Primary control module) through the CopperCraft interface and issue the following command:

```
lcrestart n.n
```

where *n.n* indicates the shelf and slot number of the target DSL module. Note that this command cannot be abbreviated.

SCM-3 Reset Switch

If the software restarts do not work and your CE200 is equipped with a System Control Module-3 (SCM-3), press the Reset switch on the front panel of the module. It resets the System Control Module, Buffer Control Module, and any WAN modules in a complex.

Restarting a Redundancy Complex

In CE200s equipped for Control and WAN Redundancy, you can re-start the "redundancy complex" of System Control Module, Buffer Control Module, and WAN modules by commanding a transfer to the secondary redundancy complex.

To command a redundant system transfer while connected to the primary redundancy complex, enter:

```
set cmsystem command=relinquish
```

To command a redundant system transfer while connected to the secondary redundancy complex, enter:

```
set cmsystem command=takeover
```

In either case, a commanded failover will disable redundancy and restart the modules of the "failed" redundancy complex.

System Configuration

If you suspect the CE200's configuration data has become unstable or corrupt, you may need to reload the backup configuration file from a remote file server using cmMaintCmd and its ConfigRestore option as described in "Restoring a Backed Up Configuration" on page 80. Implementing the recovered data file, however, does require a full restart of the CE200 as discussed in the next section.

The cmSystem *ReadConfig* option is no longer supported.

System Soft Restart

In a major software fault occurs, there may be no alternative to restarting the system. To restart/reboot the CE200, enter the following command:

```
set cmsystem command=restart
```

Using SCMRestart

In certain relatively rare cases, a system failure could be accompanied (or caused) by an "out of control buffers" state. If you have tried unsuccessfully to soft restart the Control Module as described above (**`set cmboard [1.2] command=restart`**) and the system (**`set cmsystem command=restart`**), there is one other alternative recovery procedure: SCMRestart.

The SCMRestart command does not require any control buffers to be available. To run this command, enter:

```
scmrestart
```

Note that the *entire word* must be entered, but it isn't case-sensitive. You must have Provisioning or greater privileges to execute this command.

Removing or Replacing Modules

This section describes the procedures for installing or replacing the System Control Module, Buffer Module, WAN modules, DSL modules, and Power Supply modules.

C A U T I O N

All CE200 modules contain static-sensitive devices. If you must remove or handle modules for any reason, observe standard ESD precautions (such as using ground straps for personnel and equipment, and storing removed modules in anti-static bags). If you are unsure of the necessary precautions, contact Copper Mountain Technical Support for assistance.

System Control Module

Non-Redundant Systems

Replacement of a System Control Module is service-affecting, with consequent interruption of service for all subscribers served by this CE200. Except in cases of emergency (system down), we recommend that you replace the module during a scheduled maintenance period at off-peak hours.

To replace the SCM module:

1. If the installed System Control Module is operational:
   - Set FTP to Binary mode and copy the Configuration file to a workstation.
   - Or, back up the existing configuration file as described in “Backing Up the Saved Config File” on page 79.
2. Power down the system and disconnect all front panel cables from the System Control Module.
3. Use a Phillips screwdriver to loosen the two screws securing the System Control Module.
4. Lift the module extraction levers to gently separate the System Control Module from its connector inside the CE200.

   When the module is free, remove it from the CE200 and transfer it to a suitable anti-static container.
5. Remove the new System Control Module from its anti-static bag, and install in the empty CE200 System Control slot.
6. Gently but firmly seat the module into its connector inside the CE200.

   If the module does not readily mesh with the connector pins, do not force it; pull the module out and try again.
7. When the module is seated in the connector, ensure that the extraction levers lay flat (module-inserted position), and use your fingers to screw in the two retainer screws.

   Finger tighten only; do not over-torque.

8. Reconnect the front-panel cables disconnected in step 2.
9. Apply power to the system and wait for boot-up to complete.
10. Connect to the Craft port and use a terminal emulation program to log in to the CE200.
11. Configure the CE200 to talk to the rest of the network:

    j) Set the netmodel, IP address, and netmask of the CE200.

    ```
    CRAFT (1.2)> set cmiface [1.2.1] netmodel=ip
           ipaddress=xxx.xxx.xxx.xxx netmask=255.255.255.xxx
    ```

    k) Set the default route for this CE200.

    ```
    CRAFT (1.2)> set iproute [0.0.0.0] mask=255.255.255.xxx
           nexthop=xxx.xxx.xxx.xxx
    ```

    l) Set the SNMP command strings.

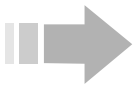
    ```
    CRAFT (1.2)> set cmoperator [public] context=snmp privilege=view

    CRAFT (1.2)> set cmoperator [private] context=snmp
           privilege=provision
    ```

    m) Save the new configuration.

    ```
    CRAFT (1.2)> set cmsystem command=save
    ```

12. If necessary, load the software.
13. Restore the saved configuration:
    - Set FTP to Binary mode and copy the Configuration file from the workstation.
    - Or, refer to “Restoring a Backed Up Configuration” on page 80 to reinstate the saved system configuration.
14. Restart the system.

    ```
    CRAFT (1.2)> set cmsystem command=restart
    ```



N O T E

If you are upgrading to a Buffer Control Module 2, you must also upgrade to a System Control Module 3. The BCM-2 will not work with a SCM-1 or SCM-2.

Redundant Systems

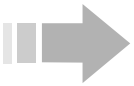
If the SCM to be replaced is installed in a CE200 with a Control and WAN redundancy complex, it can be replaced without affecting service, provided it is in a Secondary role (set in cmBoard) and cm-Sytem Redundancy is disabled. For instructions, see “Enabling Redundancy” on page 123 and “Restoring Redundancy after a Failover” on page 129.

If you are also upgrading to a Buffer Control Module 2, both it and a SCM-3 must be installed in both sides of the chassis. The BCM-2 will not work with a SCM-1 or SCM-2.

Buffer Control and WAN Modules

Non-Redundant Systems

Replacement of the Buffer Control and WAN modules is service-affecting, with consequent interruption of service for all subscribers served by this CE200. Except in cases of emergency (system down), we recommend that you replace any modules during a scheduled maintenance period at off-peak hours.



N O T E

If you are upgrading to a Buffer Control Module 2, you must also upgrade to a System Control Module 3. The BCM-2 will not work with a SCM-1 or SCM-2.

To replace a Buffer Control or WAN module:

1. Power down the system and disconnect any front panel cables from the affected module.
2. Use a Phillips screwdriver to loosen the two screws securing the module to be replaced.
3. Lift the module extraction levers to gently separate the module from its connector inside the CE200.
4. When the module is free, remove it from the CE200 and transfer it to a suitable anti-static container.
5. Remove the new module from its anti-static bag and insert it into the empty slot in the CE200.
6. Gently but firmly seat the module into its connector inside the CE200.
7. If the module does not readily mate to the connector, do not force it; back the module out and try again.
8. When the module is seated in the connector, ensure that the extraction levers lay flat (module-inserted position), and use your fingers to screw in the two retainer screws.
9. Finger tighten only; do not over-torque.
10. Reconnect the front-panel cables disconnected in step 1.
11. Apply power to the system and wait for boot-up to complete.

Redundant Systems

If the Buffer Control or WAN module to be replaced is installed in a CE200 with a Control and WAN redundancy complex, it can be replaced without affecting service, provided it is in a Secondary role (set in cmBoard) and cmSytem Redundancy is disabled. For instructions on enabling and disabling redundancy, see “Enabling Redundancy” on page 123 and “Restoring Redundancy after a Failover” on page 129.

If you are upgrading to a Buffer Control Module 2, it must be installed in both sides of the chassis. Also, you must also upgrade

to a System Control Module 3 in both sides of the chassis. The BCM-2 will not work with a SCM-1 or SCM-2.

DS3 Protection Switch

If you are installing or replacing a DS3 Protection Switch/Alarm Panel, you will need to power down the chassis. We recommend that install the module during a scheduled maintenance period at off-peak hours.

The term Preferred refers to modules located in slots 2 through 5 on the left side of the CE200 chassis, and Backup refers to modules located slots 14 through 17 in the right side of the CE200 chassis (also called the Redundancy Complex). Primary refers to the side that is controlling operation and traffic, and Secondary refers to the side that will be active in case of a failover.

C A U T I O N

With a DS3 Protection Switch installed, you cannot downgrade your CE200 software below release 4.0. If you need to install an earlier release, you should first remove the DS3 Protection Switch/Alarm panel and reinstall the standard alarm panel.

Before you begin:

- Make sure all WAN modules required for your configuration are installed and configured properly.
- Make sure you have installed version 4.0 (or later) of the CE200 and AMS/EM software packages.
- Check that the upgrade kit includes a DS3 Protection Switch/Alarm Panel, eight coaxial cable assemblies, cable ties, and a 7/64″ Allen wrench.
- This procedure is service-affecting; no traffic can pass while performing this procedure.

C A U T I O N

*All CE200 modules contain static-sensitive devices. If you must remove or handle modules for **any** reason, observe standard ESD precautions (such as ground straps for personnel and equipment). If you are unsure of the necessary precautions, contact Copper Mountain Technical Support for assistance.*

To replace a DS3 Protection Switch/Alarm Panel:

1. Remove the four hex-head screws holding the standard alarm module using the supplied hex-wrench.
2. Carefully pull the alarm module straight out of the chassis.
3. Carefully slide the DS3 Protection Switch/Alarm Panel into the card-guides of the empty slot until it is fully seated in the backplane connector.

4. Re-install and tighten the four hex-head screws using the supplied hex-wrench.
5. Using the following table, select the type of configuration you want to set up.

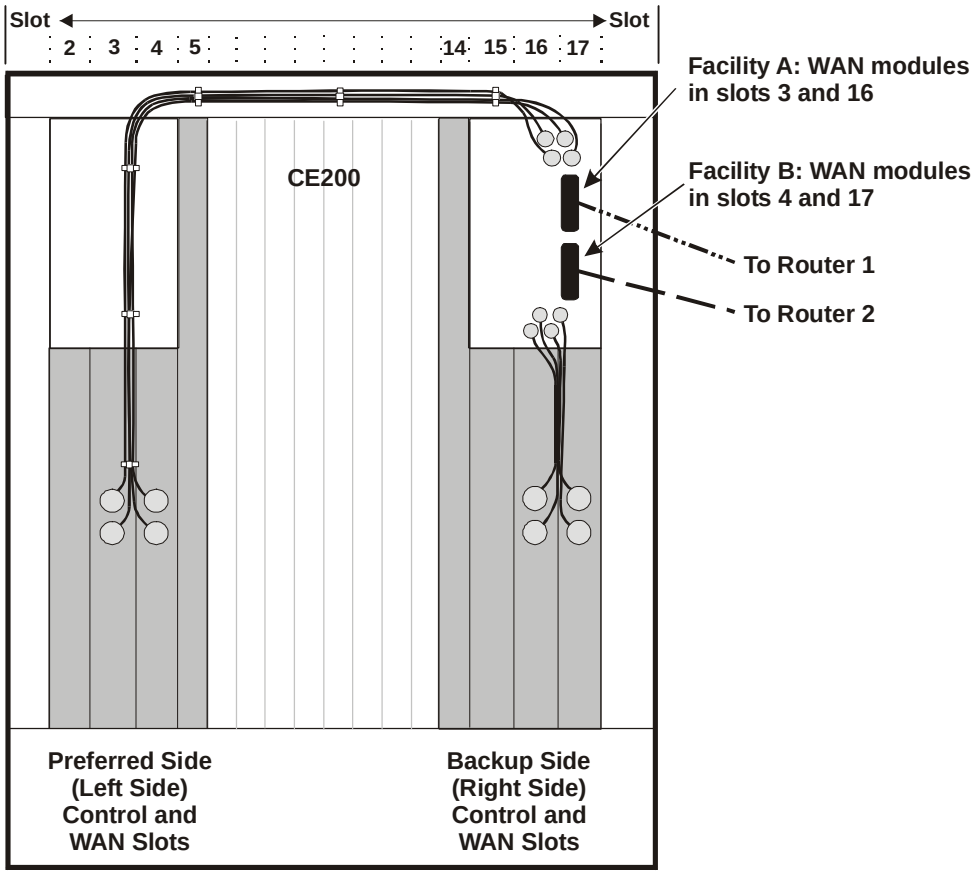
	Preferred			Backup			
Configuration Type	**SCM** (Slot 2)	**A** (Slot 3)	**B** (Slot 4)	**SCM** (Slot 15)	**A** (Slot 16)	**B** (Slot 17)	**Facilities Connected**
Control and WAN	SCM	ATM	–	SCM	ATM	–	A only
(Single WAN) (or)	SCM	F-R	–	SCM	F-R	–	A only
Control, WAN, and Facility	SCM	ATM	F-R	SCM	ATM	F-R	A & B
(Dual WAN) (or)	SCM	F-R	ATM	SCM	F-R	ATM	A & B

Legend:
SCM = SCM module installed.
ATM = DS3 ATM WAN module installed.
F-R = DS3 Frame Relay module installed.
– = No module installed.

6. Connect the cable assemblies according to your particular configuration.
7. Using the following table and illustration to guide you, connect the thin coaxial cables according to the configuration you selected.

From WAN Module	To DS3 Protection Switch/ Alarm Panel	Cable Part No.
Slot 3, Preferred WAN A (Transmit)	RX1	005993-4-XX[1]
Slot 3, Preferred WAN A (Receive)	TX1	005993-3-XX
Slot 4, Preferred WAN B (Transmit)	RX2	005993-4-XX
Slot 4, Preferred WAN B (Receive)	TX2	005993-3-XX
Slot 16, Backup WAN A (Transmit)	RX3	005993-2-XX
Slot 16, Backup WAN A (Receive)	TX3	005993-1-XX
Slot 17, Backup WAN B (Transmit)	RX4	005993-2-XX
Slot 17, Backup WAN B (Receive)	TX4	005993-1-XX

1. "XX" is a two-digit revision number (01, 02, etc.).



DS3 Protection Switch Connections

8. Dress the cables using the supplied cable ties, as shown in the previous illustration.
9. For a single WAN, connect two shielded 75-Ohm coax cables (RG-59/U) to the Facility A BNC connectors on the Alarm Panel
10. For a dual WAN, also connect two cables to the Facility B BNC connectors.
11. Connect the other ends of the shielded 75-Ohm coax cables (RG-59/U) to the DS3 access device (WAN concentrator).

To configure the CE200:

1. Enable redundancy and save the configuration:

```
CRAFT (1.2)> set cmsystem redundancy=enabled
CRAFT (1.2)> set cmsystem command=save
```

2. Check DS3 connectivity:
   a) On each DS3 WAN module, all three LEDs will illuminate during the reboot process.

b) When the line is established on each connected DS3 WAN module, the red and yellow LEDs will go out and the green LED will remain on.

3. Check the alarm panel indicators.

   a) The System OK LED should be illuminated, and the Critical and Major LEDs should be off.

   b) If any of the alarm LEDs are on, check the event log to make sure the alarms are not significant.

To perform a redundancy test:

1. Create a failover by disconnecting a cable from the WAN module on the Preferred side.
2. Wait for one to two minutes for the failure information to be written to the database. The actual hardware switchover takes no more than five seconds.
3. Verify that the System Control Module on the Preferred side [1.2] shows Role=Secondary:

```
geta cmboard

Index              ObjectClass         OperState           HwType
HwVersion           SwVersion           PromVersion          Role
ClusterRole         UpTime              NumPorts
FileName
FileDate            ConfigChange        Command
SerialNumber
Information

Instance: [1.2.0.0]
1.2.0.0            SystemControlModule Enabled
SystemControlModule
R 1.0              4.0.72              4.0.59
Secondary
Master             0 day 0 hour 4 min  5                    P:/
ce200/scm
Oct 13 2002, 00:39: 0                  None                 ****
RAM 256Mb;Flash 64M

No more instances
```

   Because this SCM is no longer in control, you cannot see any other modules on this side of the CE200.

4. Log out of the CE200 and log in to the Backup side.
5. Check the *Role* and *OperState* of the SCM and WAN modules:

```
CRAFT (1.2)> geta cmboard
```

   a) SCM [1.2] shows Role=Secondary.

   b) The WAN boards on the Preferred side show Operstate=Disabled.

   c) SCM [1.15] shows Role=Primary.

   d) The WAN boards on the Backup side show Operstate=Enabled.

6. Reconnect the cable to the WAN board on the Preferred side.

7. Enable redundancy for the system and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes while the Preferred side restarts.

8. Tell the Backup SCM to relinquish its Primary role:

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```

9. Log in to the Preferred side of the CE200.

10. Verify that the Preferred SCM [1.2] shows Role=Primary, and the Backup SCM [1.15] shows Role=Secondary.

    ```
    CRAFT (1.2)> geta cmboard
    ```

    Redundancy is automatically disabled, and the WAN modules on the backup side show Operstate=Disabled.

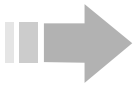
11. Enable redundancy on the CE200 and save the configuration:

    ```
    CRAFT (1.2)> set cmsystem redundancy=enabled
    CRAFT (1.2)> set cmsystem command=save
    ```

    The save process will take a few minutes while the Backup side restarts. The WAN modules on the backup side will show Operstate=Enabled.

Hot Insert Modules in a Redundant System

If your CE200 is set up for redundancy, you can *hot insert* the System Control Module-3, Buffer Control Module, and WAN modules on the secondary side (the side that is not controlling operation and traffic). You will not have to power down the chassis, and it will not affect service on the primary side (the side that is controlling operation and traffic). Be aware of the following limitations, however:



N O T E

1. *SCM-1 and SCM-2 modules are **not** hot insertable.*

2. *Due to an operational incompatibility, removal of the SCM3 causes the Buffer Control Module to enter a "hard reset" state, with the result that the SCM3 is not automatically recognized after replacement, even after pushing the front-panel reset button. In this situation the Buffer Control Module requires a power cycle to reinitialize. Thus, in order to hot insert an SCM-3, use the following procedure:*

   a) Remove the current (failed) SCM.

   b) Install the new SCM-3.

   c) Pull out the Buffer Control Module to disengage it from the backplane, thus removing its power.

   d) Reseat the Buffer Control Module to restore its power.

   e) Press the reset button on the SCM-3.

If you hot insert or extract a WAN module from the primary side, traffic will be disrupted, including some disruption to traffic on oth-

er WAN modules or the Ethernet interface. If redundancy is enabled, a failover to the secondary side will occur.

Once you have followed the workaround steps in Note 2, above, the newly inserted SCM-3 module will reset itself and the Buffer Control and WAN modules. When you hot insert a Buffer Control or WAN module, it will restart when you save the configuration after enabling redundancy.

Recover from a Control or WAN Failure

If a Control or WAN module on the Preferred (left) side of a redundant system fails and the Backup (right) side taken control, you can replace the failed module without affecting service. Follow these steps:

1. With the Backup side is now controlling operation and traffic, extract the failed SCM-3, Buffer Control, or WAN module from the Preferred side.
2. Hot-insert the replacement module in the Preferred side.
3. Log in to the Backup side, enable redundancy for the system, and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes while the Preferred side restarts.
4. Tell the Backup side to relinquish its Primary role:

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```
5. Log in to the Preferred side, enable redundancy on the CE200, and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

Upgrade a Control or WAN Module

You can upgrade a Control or WAN module by replacing it without affecting service.

To upgrade a module only on the Backup side of the CE200:

1. Extract the SCM-3, Buffer Control, or WAN module to be upgraded from the Backup side.
2. Hot insert the replacement module in the Backup side.
3. Restart the Backup side (see “A Note on ATM CPE” on page 166).

To upgrade a module only on the Preferred side of the CE200:

1. Tell the Preferred side to relinquish control, allowing the Backup side to take over.

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```
2. Log in to the Backup side.

3. Extract the SCM-3, Buffer Control, or WAN module from the Preferred side.
4. Hot-insert the replacement module in the Preferred side.
5. Enable redundancy for the system and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes while the Preferred side restarts.
6. Tell the Backup side to relinquish its Primary role:

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```
7. Log in to the Preferred side, enable redundancy on the CE200, and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

To upgrade modules on both the Preferred and Backup sides of the CE200:

1. First extract the SCM-3, Buffer Control, or WAN module to be upgraded from the Backup side.
2. Hot insert the replacement module in the Backup side.
3. Restart the Backup side (see “A Note on ATM CPE” on page 166).
4. Tell the Preferred side to relinquish control, allowing the Backup side to take over.

   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```
5. Extract the module to be upgraded from the Preferred side.
6. Hot-insert the replacement module in the Preferred side.
7. Log in to the Backup side, enable redundancy for the system, and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

   The save process will take a few minutes while the Preferred side restarts.
8. Tell the Backup side to relinquish its Primary role:

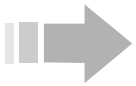
   ```
   CRAFT (1.2)> set cmsystem command=relinquish
   ```
9. Log in to the Preferred side, enable redundancy on the CE200, and save the configuration:

   ```
   CRAFT (1.2)> set cmsystem redundancy=enabled
   CRAFT (1.2)> set cmsystem command=save
   ```

DSL Line Modules

Replacement of any DSL line module--SDSL, IDSL, ADSL (G.dmt or G.lite), or T1--is service-affecting to the ports served by that module. Except in cases of emergency (system down), we recommend that you replace any modules during a scheduled maintenance pe-

riod at off-peak hours. Unlike the other modules, however, you can reliably replace DSL line modules without powering down the CE200.



N O T E

*Before you remove a module that will **not** be replaced by another module of the same type, change the netmodel to None for all VCs, ports, and IMUX bundles associated with that module.*

To replace a DSL line module:

1. Use a Phillips screwdriver to loosen the two screws securing the module to be replaced.
2. Lift the module extraction levers to gently separate the module from its connector inside the CE200.
3. When the module is free, remove it from the CE200 and transfer it to a suitable anti-static container.
4. Remove the new module from its anti-static bag, and insert it into the empty slot in the CE200.
5. Gently but firmly seat the module into its connector inside the CE200.
6. If the module does not readily mate to the connector, do not force it; back the module out and try again.
7. When the module is seated in the connector, ensure that the extraction levers are lying flat (module-inserted position), and use your fingers to screw in the two retainer screws.
8. Finger tighten only; do not over-torque.
9. After the new DSL module has been installed, you must reconfigure the data rate of each DSL port as appropriate; connected CPEs will follow the rate of the DSL interface. For example:

```
CRAFT (1.2)> set hdslmodem [1.8.1] datarate=416
```

IDSL Modules

To replace an IDSL module with another IDSL module, follow the procedure for DSL Modules (page 179).

However, there is a different procedure if you want to replace an SDSL module with an IDSL module. *Note that you will have to re-start the system*. Follow these steps:

1. Remove the SDSL module as described on page 179.
2. Save the Config.txt file:

```
CRAFT (1.2)> set cmsystem command=save
```

3. Check that the save command was successful:

```
CRAFT (1.2)> get cmsystemm configsynch

    Group: cmSystem
    ConfigSynch          =  Saved
```

4. Insert the new IDSL card as described on page 179.
5. Restart the system:

   ```
   CRAFT (1.2)> set cmsystem command=restart
   ```

6. All alarms should clear, and the CPE should train normally.
7. Configure ports as required to support the IDSL CPE.

DC Power Modules

Non-Redundant Systems

Replacing a DC Power module is service-affecting, with consequent interruption of service for all subscribers served by this CE200. Except in cases of emergency (system down), we recommend that you replace the module during a scheduled maintenance period at off-peak hours.

To replace the power module:

1. Power down the CE200.
2. Remove and replace the module using the general technique described in “Buffer Control and WAN Modules” on page 172.
3. Power up the CE200.

Redundant Power Systems

If the CE200 is equipped with a redundant power supply, you can hot swap either the Preferred or Backup power supply while the unit is running without affecting service. Simply remove and replace the module using the procedure described in steps 2 through 6 in “Buffer Control and WAN Modules” on page 172. When the new module is installed, its Status LED should illuminate green.

Preventive Maintenance

Preventive maintenance on the CE200 consists of periodically removing the fan assembly and cleaning the air filter. Buildup of dust can interfere with heat dissipation, and could affect electrical performance. The interval between cleanings will vary depending on the installed environment. We recommend that filters be inspected at least every six months and cleaned or replaced as necessary.

To clean or replace the filter:

1. Unscrew the two thumbscrews at the base of the fan assembly.
2. Using the drawer-pull handle, carefully pull the fan tray forward and out of the CE200.
3. Locate the filter at the top of the fan try and slide it out toward the rear of the fan tray.
4. Tap the filter lightly on a hard surface to loosen dust and soil.
5. If a vacuum cleaner is available, vacuum the back of the filter (that is, the reverse of normal air flow). You can also use a stiff brush such as a whisk broom to sweep away loose particles.

Replacement filters are available from Copper Mountain Networks; contact your sales representative for details.

6. To reinstall the filter, slide it back into the bracket at the top of the fan tray assembly. Airflow in the CE200 is from bottom to top, so be sure that the arrow stamped on the frame of the filter is pointing up.
7. Seat the Fan Assembly into its connector inside the CE200. *If the assembly does not readily mesh with the connector pins, do not force it*; pull the assembly out and try again.
8. When the assembly is seated in the connector, use your fingers to tighten the two thumbscrews at the base of the Fan Assembly. *Finger-tighten only; do not over-torque*.

Copper Mountain Service and Support

Expert help with your Copper Mountain equipment and software is available through our Technical Assistance Center (TAC). Optional service agreements are available which can provide you with basic telephone support, extended (24-hour, seven-day) support, and other premium services; contact your Copper Mountain sales representative for details.

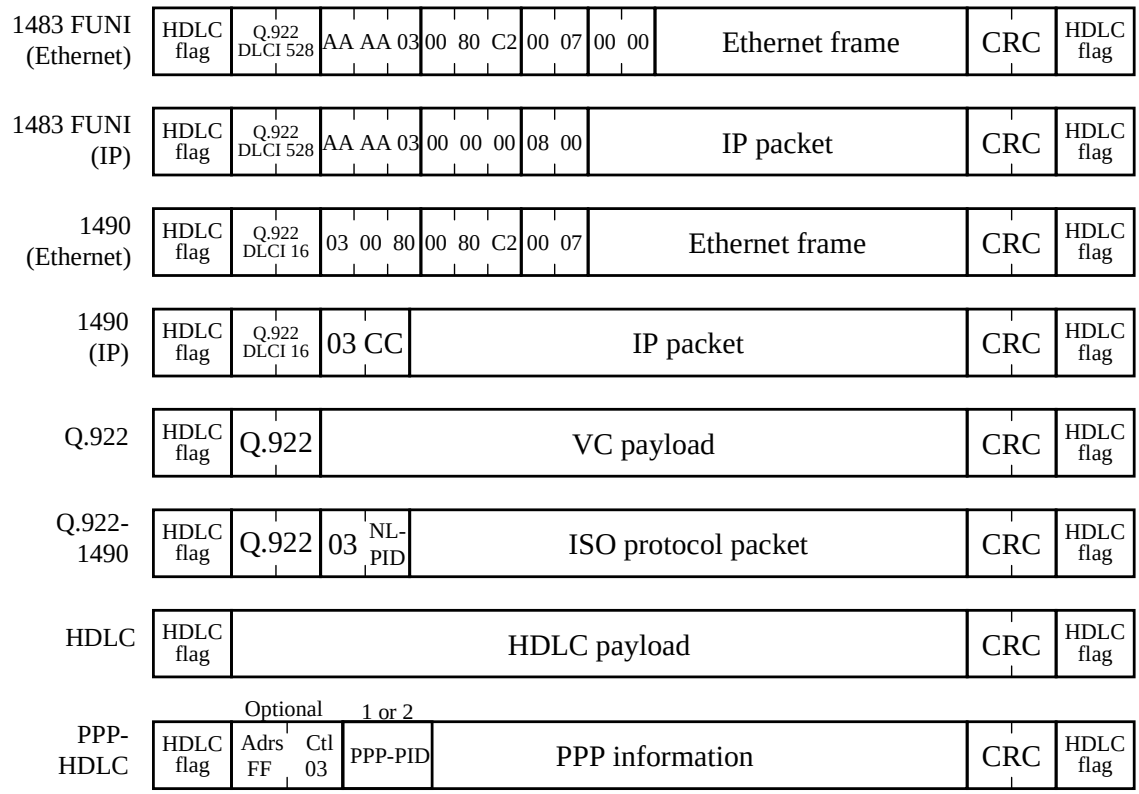
For assistance or information via e-mail, contact *support@coppermountain.com*. Technical information about DSL and Copper Mountain products and technology is also available at our web site, *http://www.coppermountain.com*.

If you suspect a failure in your CE200 or one of its modules, contact the Copper Mountain TAC for help and instructions. Should it be necessary to return any of your Copper Mountain equipment for service or repair, you will be provided with shipping instructions and the Repair Authorization numbers. These numbers should always be clearly marked on the outside of the package to ensure the fastest possible repair or replacement.

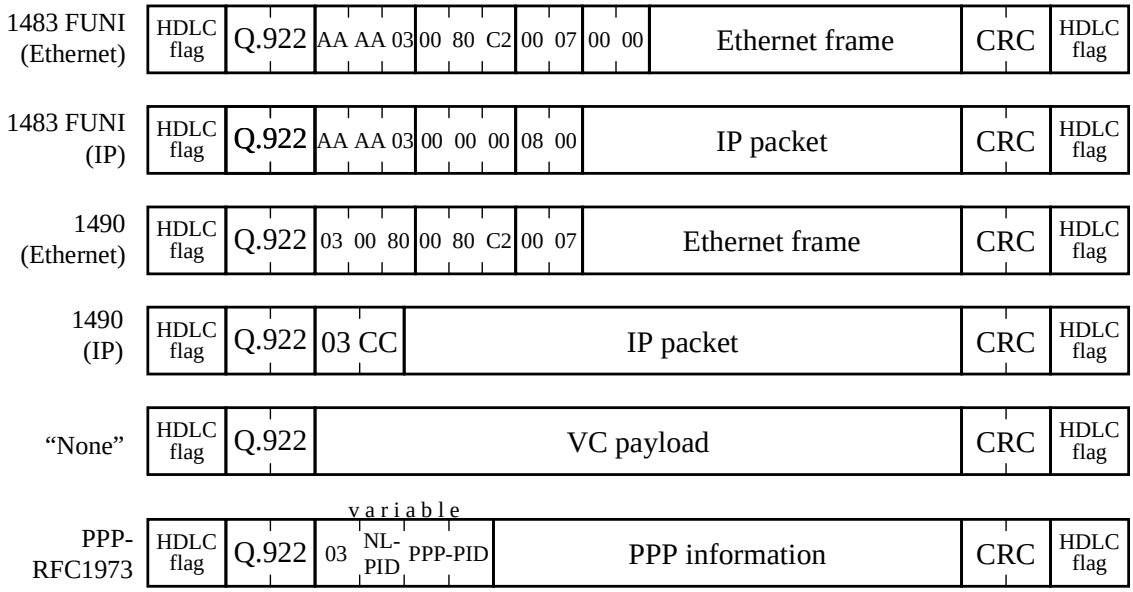
Appendix A
Overview of Data Encapsulation

As the capabilities of the CopperEdge product family have grown to embrace an ever wider pool of users and service providers, the number of supported protocols and encapsulations has grown as well. The diagrams presented below and on the following page provide a graphic representation of each of the encapsulation types supported in Version 2.1, which interfaces and networking models they apply to, and how they resemble and differ from each other.

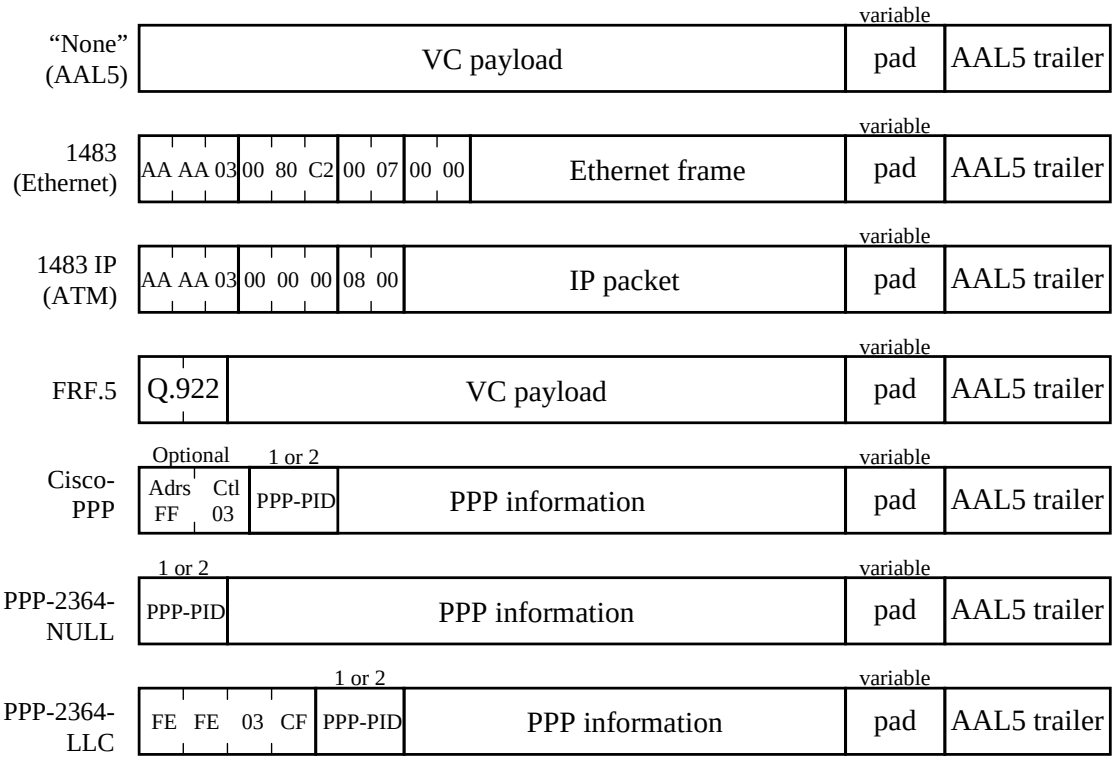
DSL Port Encapsulations



Frame Relay WAN VC Encapsulations



ATM WAN VC Encapsulations



Netmodels, Encapsulations, and Translations

It is important to distinguish between *port encapsulations* and *VC encapsulations*. The following sections describe DSL port encapsulations, DSL VC encapsulations, WAN port encapsulations, and WAN VC encapsulations. Known inconsistencies appear at the end.

DSL Port Encapsulations

For DSL ports, the port encapsulation defines whether the DSL port is a single logical interface or it supports multiple VCs. Encapsulations of HDLC or PPP-HDLC are a single logical interface and cannot support VCs. All other encapsulations support multiple VCs on a DSL port.

DSL physical ports support the following encapsulations. Note that in the cross-connect Netmodel w/ Q.922, ATM or RFC1973, DSL VCs defined with cmSubIfaceTable entries assume an encapsulation implied by the port encapsulation.

DSL Port Encapsulation	Implied DSL VC Encapsulation	Netmodels	Description	Notes
None	cmSubIface not allowed	None		The interface is not yet configured for data transport
RFC1483	cmSubIface not allowed	VWAN, IP, CVPN	Funi VPI/VCI of 1/32 + 1483(MAC) = Frame Relay DLCI 528 + 1483 (MAC)	
RFC1490	cmSubIface not allowed	VWAN, IP, CVPN	Frame Relay DLCI 16 + 1490(MAC)	ICP is optional
Q922	None	Cross-Connect	Q.922 VCs of unspecified content	Used to specify Layer-2 Frame Relay interworking modes.
ATM	None	Cross-Connect	G.lite ATM VC unspecified content	Used to specify Layer-2 ATM interworking modes.
Q922-1490	RFC1490	Cross-Connect	Q.922 VCs, each translated VC containing RFC1490 encapsulation.	Used for FRF.8 translation. Non-translated VCs can have arbitrary (unknown) encapsulation
HDLC	cmSubIface not allowed	Cross-Connect	Raw HDLC frame of unspecified content. No VCs.	Used for Cisco-PPP. Cannot allow VCs or ICP, because there is no Q.922 header.

DSL Port Encapsulation	Implied DSL VC Encapsulation	Netmodels	Description	Notes
PPP-HDLC	cmSubIface not allowed	Cross-Connect	PPP (RFC1662) + PPP Payload. No VCs.	Payload excludes HDLC address and control fields. No VCs or ICP, because there is no Q.922 header.
RFC1973	RFC1973	Cross-Connect	Either PPP-transparent or PPP-translation FwdMode, depending on WAN encapsulation	FRF.8 translation performed when WAN encapsulation is RFC2364-null or RFC2364-LLC

Dual Netmodel

Except for HDLC and PPP-HDLC, all DSL port encapsulations (including None) allow a dual-netmodel cmIface entry on VCID=22. This means that such encapsulations imply a port-level encapsulation that supports VCs, which is ATM or RFC2364-null/RFC2364-LLC for ATM-based DSLs, and Q.922 or RFC1973 for frame-based DSLs.

DSL VC Encapsulations

In one special case (DSL VC VCID=22), you specify a DSL VC encapsulation in a cmIfaceTable row. Such DSL VCs support these encapsulations:

DSL VC Encapsulation	Valid Netmodels	Description	Notes
None	Cross-Connect	Q.922 (SDSL) or AAL5 (G.lite and G.dmt)	Used for transparent forwarding
RFC1490	Cross-Connect CopperVPN	1490 IP or MAC frames	Used for FRF.8 translation
IP-1490	CopperVPN	Routed IP frames	

WAN Port Encapsulations

Currently, WAN ports do not have cmIface entries or explicit configuration. Since the hardware for each WAN port supports only one encapsulation, the encapsulation is implied by the hardware.

Frame-based WAN interfaces (such as DS3-Frame and Quad-T1) imply a WAN port encapsulation of Q.922 (Frame Relay). ATM WAN interfaces (such as DS3-ATM) imply a WAN port encapsulation of ATM. In the future, frame-based WANs may support non-Q.922 encapsulations and you will then have to explicitly configure the WAN port encapsulations.

WAN VC Encapsulations

WAN VCs (Frame Relay and ATM) support the following encapsulations. For 1483 or 1490 encapsulation, the netmodel determines whether the frame-type is IP PDU or MAC PDU.

WAN VC Encapsulation	Valid Netmodels	Frame-based VC Format	ATM VC Format
None	None, Cross-Connect	Q.922 address + payload	AAL5 payload
RFC1483	VWAN, IP, CVPN, Cross-Connect	Q.922 address + 1483 (MAC or IP)	AAL5 + 1483
RFC1490	VWAN, IP, CVPN	Q.922 address + 1490 (MAC or IP)	AAL5 + 1490
RFC1973	Cross-Connect	Q.922 address + RFC1973	
RFC2364-null	Cross-Connect	none	PPP payload + AAL5 trailer
RFC2364-LLC	Cross-Connect	none	LLC header + PPP payload + AAL5 trailer
FRF.5	Cross-Connect		AAL5 + FRF.5

Ethernet Port Encapsulations

Since industry standards unambiguously define a single encapsulation for each protocol, the CE200 only accepts Encapsulation=None.

Translations

For the Cross-Connect netmodel, the CE200 does not terminate protocols, but may convert headers. For non-VC DSL port encapsulations, there is only one logical interface per DSL port. For DSL port encapsulations of Q.922, RFC1973, or ATM, each DSL VC may be a separate logical interface.

DSL Port Translations

This table covers the DSL port encapsulations that do not support VCs. It describes Cross-Connects from a DSL port to a WAN VC.

Netmodel	DSL Port Encapsulations	WAN VC Encapsulations	FwdMode	Notes
Cross-Connect	HDLC	None	HDLC-VC-payload	Used for PPP-HDLC to Cisco-PPP-ATM
	PPP-HDLC			
Cross-Connect	PPP-HDLC	RFC1973	PPP-HDLC-1973	PPP-HDLC to RFC1973

DSL VC Translations

This table covers the DSL port encapsulations that support VCs. It describes Cross-Connects from a DSL VC to a WAN VC. Since DSL VCs defined in the cmsubIface table have no FwdMode object, operators must look to the Cross-Connected WAN VC for the FwdMode, which the CE200 uses for both the DSL VC and the WAN VC. In these cases, the DSL port has a FwdMode=Per-VC, since each DSL VC has its own FwdMode.

Netmodel	DSL VC Encapsulations[a]	WAN VC Encapsulations	FwdMode	Notes
Cross-Connect	None	None	VC-VC-payload	payload transparent, including PPP-RFC2364.This is not really a translation.
		FRF.5	FRF.5	FRF.5
Cross-Connect	RFC1490	RFC1483	FRF.8-1490-1483	FRF.8 translate
		None	VC-VC-payload	payload transparent, including PPP-RFC2364
		FRF.5	FRF.5	FRF.5
Cross-Connect	RFC1973	RFC 1973	PPP-Transparent	Like VC-VC-payload, but with encapsulations specified and validated
		RFC 2364-null	PPP-Translation	FRF.8 translate
		RFC 2364-LLC	PPP-Translation	FRF.8 translate
		None	VC-VC-payload	payload transparent

a. [1] For DSL VCs configured with a cmSubIfaceTable entry, the VC encapsulation is implied by the DSL port encapsulation. For DSL VCs configured with a cmIfaceTable entry, the VC encapsulation is specified directly by that cmIfaceTable entry.

Appendix B
cmIface Configuration

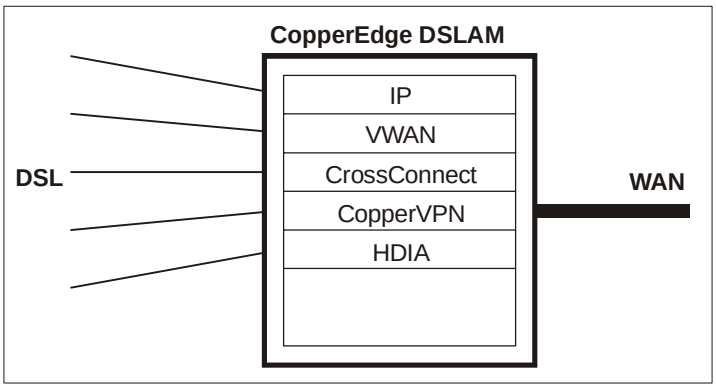
Data Forwarding Configuration

The CE200 is capable of a wide variety of protocol processing functions for user data. It requires a significant amount of operator configuration to implement functions.

Each interface on the CE200 operates in one of the CE's networking models (or *netmodels*):

- IP
- VWAN
- Cross-Connect
- CopperVPN

You can configure interfaces independently, and all netmodels may be simultaneously operating in a single CE200.



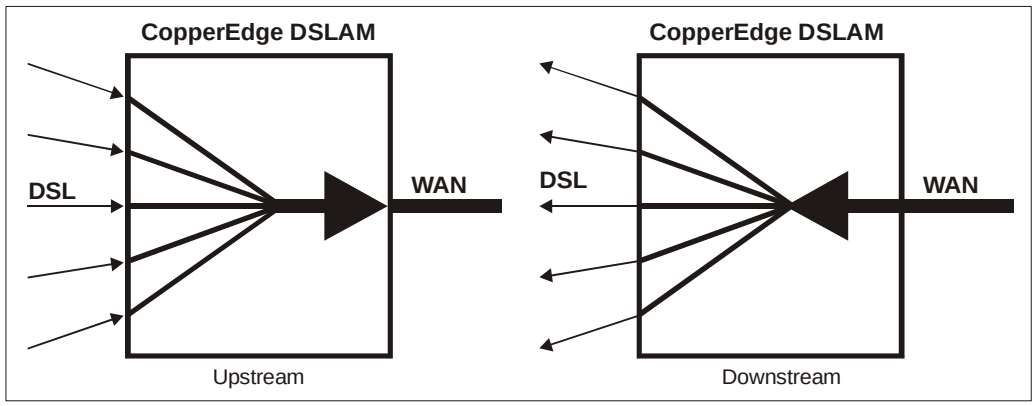
CE200 with Five Simultaneous Netmodels

In general, the netmodels behave quite differently and require somewhat different configuration. Later in this appendix we will discuss each netmodel separately. But first we describe some common principles used in the CE200.

The CE200 is primarily a data forwarding and multiplexing device that connects DSL subscribers to Network Service Providers

(NSPs). As part of the forwarding process, the CE200 often modifies the data in some way, but very little user data actually terminates in the CE200.

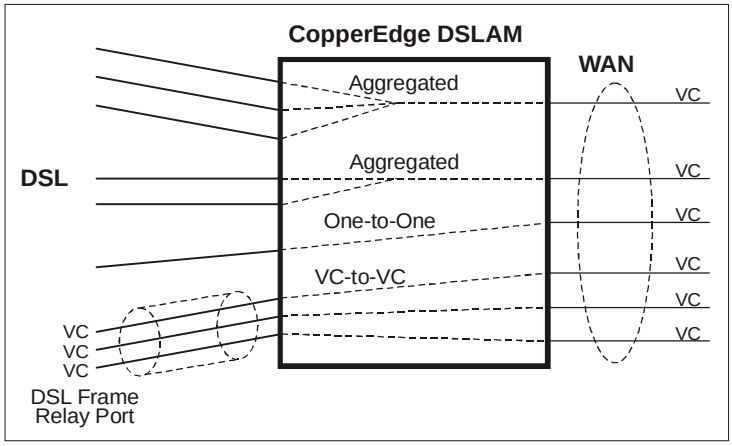
Although the CE200 can forward data between any two interfaces, most applications involve the CE200 aggregating upstream traffic from multiple DSL subscribers into one (or a few) multiplexed WAN interfaces, and demultiplexing and distributing downstream traffic from the multiplexed WAN interface to many DSL subscribers. A multiplexed WAN interface is typically a high-capacity Frame Relay or ATM interface, with many independent virtual circuits (VCs) on it. Each virtual circuit is a logical point-to-point link to an NSP. Note that in some configurations, the WAN interface can be a CE200 Ethernet port, or even another DSL port.



CE200 with Upstream and Downstream Traffic

DSL subscribers are usually connected, at subscription time, to a single NSP, and therefore, to a single VC on the WAN interface. Thus, you can often think of CE200 configuration as connecting CE200 DSL ports to CE200 WAN VCs.

However, the CE200 does much more than just connect ports; it also performs many other important networking functions, as described in the netmodel sections. In some netmodels, many DSL ports can be aggregated together onto a single WAN VC. In other netmodels, it is only meaningful to map a single DSL interface to a single WAN VC. In these cases, the CE200 enforces this one-to-one connection requirement at configuration time. A third possibility is that the DSL port is a Frame Relay or ATM interface containing VCs within it. Then the VCs on the DSL port are connected to VCs on the WAN port.



CE200 with Aggregated and One-to-One Connections

Because many DSL ports may be connected to a single aggregated interface, you always specify connections at the DSL interface. That is, you define connections by configuring the DSL port to connect to a WAN VC. The connection is two-way; data traffic can flow in both directions. But, you always configure a connection by connecting a DSL port to a WAN VC, not by connecting a WAN VC to a DSL port.

The cmIfaceTable

At the center of all user data processing configuration is a single table: the cmIfaceTable (often abbreviated cmIface). All user-data configuration starts with the cmIface table. In some cases, several other support tables augment the cmIface table. But each support table is needed only in some configurations, and only for some functions. Different applications and interfaces use different support tables.

The cmIface table consists of many rows, or instances, of cmIface entries. Each entry configures a single interface, and in some cases, connects one interface to another. Each entry consists of several objects. The exact usage of the objects is highly dependent on what kind of interface is being configured, and on which netmodel you choose for user-data processing.

In all cases, a single cmIface entry configures a DSL port, a DSL voice circuit, an Ethernet port, or a WAN virtual circuit.

Permanent Interface Identifier

The cmIface table is indexed by a Permanent Interface Identifier (PII). This means each cmIface entry is uniquely identified by its PII. A single PII can have no more than one cmIface entry. Note that PIIs are used for other purposes besides the cmIface table, so not all PIIs in the system have a cmIface entry.

PIIs are called permanent because they do not change across power cycles of the CE200, and the PII for an interface does not change as a result of any configuration of other interfaces. Each DSL port, Ethernet port, and WAN VC is uniquely identified by a PII. In some cases, the CE200 may require some enabling configuration to make the PIIs for some VCs valid; not all PIIs are necessarily valid with all configurations. The CE200 enforces any requisite configuration by disallowing the use of PIIs that are not currently valid. PIIs are unique across all interfaces in a single CopperEdge chassis.

The PII format is completely described elsewhere, but briefly, a PII is a sequence of one to four decimal numbers separated by periods. The four numbers are the chassis, slot, port, and circuit ID of the interface. For example, 1.3.1.99 is the PII for chassis 1, slot 3, port 1, (virtual) circuit ID 99. While a PII sometimes resembles an IP address, PIIs are totally different and should not be confused with IP addresses.

cmIface Objects

Each cmIface entry contains several objects. Three of the objects, Name, GroupName, and AdditionalInfo, are configurable identification for operator convenience. You can assign arbitrary strings to the interface Name and AdditionalInfo objects. Also, you can create interface groups and optionally assign each interface to a single group. These objects have no effect on how the CE200 processes user data. They are for customized interface identification only.

All other cmIface objects configure or display various aspects of how the CE200 processes and forwards user data. Not all objects are meaningful in all contexts. A typical cmIface entry for a DSL port might look similar to this example:

```
get cmiface [1.6.1]
Group: cmIfaceTable
Instance:            = [1.6.1.0]
PII                  = 1.6.1.0
IfIndex              = 1.6.1.0
Name                 = ""
GroupName            = ""
AdditionalInfo       = ""
NetModel             = Cross-Connect
IpAddr               = 0.0.0.0
NetMask              = 0.0.0.0
MacAddr              = ff.ff.ff.ff.ff.ff
BurnedInMacAddr      = 0.0.0.0.0.0
FarEndAddr           = 0.0.0.0
DestPII              = 1.3.1.61
ICPCompatible        = Yes
EncapsulationType    = PPP-HDLC
FwdMode              = PPP-HDLC-1973
```

Some objects are operator-configurable and others are read-only. Read-only objects display information about the interface; you cannot set or change the values. A general description of each object follows, but typically they may have different significance for different interface types (DSL, DSL VC, Ethernet, WAN VC), and in different netmodels.

IfIndex (Read/Write)	The PII of the interface. You cannot set this object, but you must specify it to identify the cmIface entry of interest. You can create and delete cmIface entries.
Name (Read/Write)	An arbitrary string that names the interface.
GroupName (Read/Write)	The name of the interface group, if any, to which this interface belongs.
AdditionalInfo (Read/Write)	An arbitrary string of information for the interface. For example, the name and phone number of a person to contact when there is a problem with this interface.
NetModel (Read/Write)	The netmodel that the CE200 uses for this interface. A NetModel of None means the interface is not configured for user data.
IpAddr (Read/Write)	In the IP netmodel, the standard IP address of this interface
NetMask (Read/Write)	In the IP netmodel, the standard IP subnet mask for this interface
MacAddr (Read-only)	In some netmodels, the working MAC address of this interface.

BurnedInMacAddr (Read-only)	For CPEs that support this feature, the burned-in MAC address of the CPE at the end of the DSL link.
FarEndAddr (Read/Write)	For IP netmodel WAN VCs, the IP address of the device at the opposite end of the WAN VC.
DestPII (Read/Write)	For most netmodels on DSL ports, the aggregated interface to which this DSL port is connected.
ICPCompatible (Read/Write)	For DSL ports, specifies whether the CPE at the end of the DSL link supports Copper Mountain's Internal Control Protocol (ICP) for enhanced network manageability.
EncapsulationType (Read/Write)	The data format of data frames or packets exchanged on this interface.
FwdMode (Read-only)	The kind of processing that the CE200 will perform for packets on this interface. The CE200 displays this result as a function of all other configuration.

Automatic Creation of WAN VC cmIface Entries

For ease of configuration file storing and loading, the CE200 allows the full set of cmIface entries to specified in any order. This means that you can specify a WAN VC as a DestPii of a DSL port, even if the WAN VC's cmIface entry does not yet exist. In this case, the CE200 automatically creates a minimal cmIface entry for the WAN VC. The CE200 sets the WAN VC's netmodel to the netmodel of the DSL port which implicitly created it. The CE200 sets all other parameters to the simplest defaults. In general, you should explicitly configure your WAN VCs to be sure all the parameters are set as you expect.

IP Netmodel

This section assumes you are familiar with the networking concepts of IP, and the CE200's IP capabilities. It provides details of how to configure the IP functions of the CE200. The IP netmodel is supported on DSL ports, Ethernet ports, and WAN VC interfaces.

Full IP Routing

DSL and Ethernet interfaces

The CE200 treats DSL interfaces the same as Ethernet interfaces, because in IP netmodel, the DSL link acts as a long cable to the CPE's Ethernet interface. The CE200 requires you to configure IP subnets on DSL and Ethernet interfaces. You must specify the NetModel, IpAddr, NetMask, and Encapsulation objects. For example:

```
Craft> set cmiface[1.6.1] netmodel=ip ipaddr=10.0.0.1
       netmask=255.0.0.0 encapsulationtype=rfc1483
```

WAN VC interfaces

The CE200 treats WAN VCs as unnumbered IP point-to-point links. Unnumbered links have no subnet on them, and each end may have an IP address. You can assign addresses to either end of the VC, both ends, or neither end. To assign an IP address to the CE200's end of the VC, set the IpAddr object. The NetMask is ignored for WAN VCs, because there is no subnet associated with a WAN VC. To assign an IP address to the far end of a WAN VC, set the farEndAddr object. Note that for the CE200 to route a packet to a WAN VC, one end or the other of the VC must have an IP address, because the routing table requires an IP address for the NextHop object (specified through the IPRoute MIB group). For example:

```
set cmiface[1.3.1.61] netmodel=ip ipaddr=10.0.0.1
       farendaddr=192.168.1.1
```

Policy-to-WAN-VC IP Routing

You can configure DSL interfaces for policy IP routing to a WAN VC. To do so, set the NetModel, IpAddr, NetMask, Encapsulation, and DestPii objects in the DSL port's cmIface entry. You must specify a WAN VC as the DestPii. This means that inbound packets that are not addressed to the DSL's own IP address are always forwarded to the specified DestPii. For policy routing, the CE200 does not consult any routing table to forward received packets. For example:

```
set cmiface[1.6.1] netmodel=ip ipaddr=192.168.1.1
       netmask=255.255.255.0
       encapsulationtype=rfc1483 destpii=1.3.1.61
```

Ethernet and WAN VCs do not support policy routing.

Policy-to-Ethernet IP Routing

You can configure DSL interfaces for policy IP routing to the CE200's Ethernet port. To do so, set the NetModel, IpAddr,

NetMask, Encapsulation, DestPii, and farEndAddr objects in the DSL port's cmIface entry. You must set the DestPii to a CE200 Ethernet port, usually 1.2.1. You must also set the DSL port's farEndAddr object to the next hop IP address on the Ethernet subnet to receive the forwarded packets. Each DSL port can specify a different farEndAddr on the same Ethernet interface. In this forwarding mode, the CE200 forwards inbound DSL packets that are not addressed to the DSL's own IP address, to the specified Ethernet port and next hop IP address. For policy routing, the CE200 does not consult any routing table to forward received packets. For example:

```
set cmiface[1.6.1] netmodel=ip ipaddr=192.168.1.1
       netmask=255.255.255.0
       encapsulationtype=rfc1483 destpii=1.2.1
       farendaddr=172.16.1.1
```

Ethernet and WAN VCs do not support policy routing.

VWAN Netmodel

This section assumes you are familiar with the networking concepts of Copper Mountain's VWAN netmodel. It provides details of how to configure the VWAN functions of the CE200.

VWAN is the simplest mode to configure for DSL interfaces. To do so, set the NetModel, Encapsulation, and DestPii. You must specify a WAN VC or an Ethernet port for the DestPii. If only one DSL port refers to a given WAN VC, both interfaces are in VWAN point-to-point mode. If two or more DSL ports refer to the same WAN VC, all the interfaces in the VWAN group are in VWAN bridge mode. The CE200 automatically selects the FwdMode based on the given configuration. For example:

```
set cmiface[1.6.1] netmodel=vwan
       encapsulationtype=rfc1490 destpii=1.3.1.61
```

WAN VCs do not need to be configured for VWAN mode because they assume VWAN mode automatically when referenced by VWAN DSL ports.

WAN VC and Ethernet interfaces cannot be configured as members of a WAN group.

Cross-Connect Netmodel

This section assumes you are familiar with the networking concepts of Copper Mountain's Cross-Connect netmodel. It provides details of how to configure the Cross-Connect functions of the CE200.

Ethernet interfaces don't support the Cross-Connect netmodel.

WAN VCs do not need to be configured for Cross-Connect mode because they assume Cross-Connect mode automatically when referenced by DSL ports.

VC to VC Frame Forwarding

In this mode, the CE200 treats the physical DSL interface as a Frame Relay interface. To configure this mode, set the DSL port's NetModel, and set Encapsulation=Q922. For each VC on the DSL interface, configure a cmSubIface entry with the DSL VC's PII, and set the cmSubIface DestPii to the WAN VC to which the DSL VC is Cross-Connected. For example:

```
set cmiface[1.6.1] netmodel=cross-connect
       encapsulationtype=q922

set cmsubiface[1.6.1.40] destpii=1.3.1.61
```

HDLC to VC Frame Forwarding

In this mode, the CE200 forwards HDLC frame payloads into WAN VC payloads. To configure this mode, set the DSL port's NetModel, Encapsulation=HDLC, and DestPii. You must specify a WAN VC as the DestPii. For example:

```
set cmiface[1.6.1] netmodel=cross-connect
       encapsulationtype=hdlc destpii=1.3.1.61
```

PPP Frame Conversion

In this mode, the CE200 converts PPP-HDLC frames on the DSL link to PPP-RFC1973 format on a WAN VC. To configure this mode, set the DSL port's NetModel, Encapsulation=PPP-HDLC, and DestPii. You must specify a WAN VC as the DestPii. For example:

```
set cmiface[1.6.1] netmodel=cross-connect
       encapsulationtype=ppp-hdlc destpii=1.3.1.61


set cmiface[1.3.1.16] netmodel=cross-connect
       encapsulationtype=rfc1973
```

CopperVPN+ Netmodel

This section provides details of how to configure the CopperVPN+ functions of the CE200.

To configure a CopperVPN group (the "+" is a marketing term and is not used in actual operation; this document may use the terms *CopperVPN* and *CopperVPN+* interchangeably), you will perform the following basic steps in sequence:

1. **Create the Group:** Create a *cmIfaceTable* entry for a WAN VC with *NetModel=CopperVPN*. This will automatically identify it as the PII of a CopperVPN group.

   Use *cmVPNGroupTable* to establish any other settings applicable to the group (peer-to-peer, IpValidation, etc.).

2. **Add Subnets:** If using static addressing, assign the IP address and netmask for the IP gateway for the group. You can also assign a *diagnosticIpAddress* in *cmVpnSubnetTable*. Up to eight subnets can be associated with a CVPN group.

3. **Add the Subscriber Group Members:** Create the entries in the *cmIfaceTable* for the DSL interfaces that you wish to assign as members of the group, specifying DestPii=VPNGroupPii (the PII of the interface configured in step 1).

4. **Add Static Routes:** As needed, use *cmVPNRoute* to associate group destination addresses with the appropriate interfaces.

5. **Configure Filters:** As needed, use *cmFilter* and *cmGroupFilterTable* to create and apply policy filters to individual interfaces and groups.

6. **Configure DHCP:** Configure the *cmDHCPTable*, one entry per subscriber interface. For CopperVPN, set the *Function* object to *DHCPForward*.

CopperVPN (Legacy) Netmodel

Beginning with Release 7.0, the CopperVPN netmodel provided in previous releases is no longer supported as a configuration option. Rather it has been superseded by the new CopperVPN (referred to in some locations in this document as CopperVPN+). Should you import a configuration from a CopperEdge running a previous code version, any CopperVPN configurations will be identified and will work, but when the configuration is next saved, the interfaces will be automatically converted to the new CopperVPN netmodel.

HDIA (Legacy) Netmodel

Beginning with Release 7.0, the HDIA netmodel has been superseded by CopperVPN+ and is no longer supported as a configuration option. Should you import a configuration from a CopperEdge running a previous code version, the HDIA configurations will be identified and will work, but when the configuration is next saved, the HDIA interfaces will be automatically converted to CopperVPN.

Appendix C
Events and Alarms Reference

This appendix contains a reference listing of all CE200 events and alarms. Items are presented alphabetically with separate subsections for Alarms, Alarm-Clearing Notifications, and Event Notifications.

For descriptions of the groups and objects, see the *CopperEdge 200 CopperCraft Reference and MIB Definitions* manual.

Exceptions

In addition to the items presented in this section, a small number of events have been incorporated in the *CopperEdge* software in anticipation of their implementation in a future release. Thus, you may see reference in the *CopperCraft* Interface to *StateChange*, *TestStatus*, *TestError*, and several *SHDSL* events which are not yet reported as events, nor supported as alarms. Only those events and alarms listed in this section are actually supported in the current release.

Alarms

Type	Default Severity	Description
AtmVccDown	Minor	Sent by the System Control Module when an ATM WAN VC OperState causes changes to one of the fault conditions. *Information included in the event:* PII of the VC VPI of the VC VCI of the VC Operational status of the VC Cause of the operational status *Cleared by:* AtmVccUp *Trap:* Enterprise
BoardDown	Major	Sent by the System Control Module when a board stops responding to polls. *Information included in the event:* Index *Cleared by:* BoardUp *Trap:* cmBoardDown
BootFileFail	Major	Sent when a file download request from a board is not successful. *Information included in the event:* File name Board PII Reason code Descriptive text *Cleared by:* BootFileSucceed *Trap:* cmBootFileError
ConfigReadFailed	Major	Sent by the System Control Module when an error is encountered while reading the Config file. *Information included in the event:* File name Reason *Cleared by:* ConfigReadSucceed *Trap:* cmConfigFileReadFailed
ConfigWriteFailed	Major	Sent by the System Control Module when an error is encountered while writing the Config file. *Information included in the event:* File name Reason *Cleared by:* ConfigWriteSucceed *Trap:* cmConfigFileWriteFailed
CpePlugAndPlayFailure	Minor	Sent by the System Control Module when a CPE Plug and Play update fails. *Information included in the event:* Status string *Cleared by:* CpePlugAndPlayClear *Trap:* cmCpePlugAndPlayFailure

Type	Default Severity	Description
DLCIStateDisabled	Minor	Sent by the System Control Module when a Frame Relay PVC becomes disabled. *Information included in the event:* IF index DLCI State Port name Port PII Group name *Cleared by:* DLCIStateEnabled *Trap:* frDLCIStatusChange
DS1LineStatusAlarm	Major	Sent by a Quad T1 Wan module or a T1 line module when a line status changes to an error status. *Information included in the event:* Value of the DS1 line status Time at last line status change IF index of the DS1 line Port name Port PII Group name *Cleared by:* DS1LineStatusClear *Trap:* dsx1LineStatusChange
DS3LineStatusAlarm	Major	Sent by a DS3 module when a line status changes to an error status. *Information included in the event:* Value of the DS3 line status Time at the last line status change IF index of the DS3 line Port name Port PII Group name *Cleared by:* DS3LineStatusClear *Trap:* dsx3LineStatusChange
EndPointConflictAlarm	Minor	Sent by System Control Module if there is a bundle configure conflict; the end points of the IMUX bundles are improperly configured or connected. *Information included in the event:* PII of port that just trained EndPoint ID reported by port Logical IMUX PII Status string *Cleared by:* Manually *Trap:* cmEndPointConflictAlarm
FanFault	Major	Sent by the System Control Module when a Fan fault is detected. *Information included in the event:* None *Cleared by:* FanFaultClear *Trap:* cmFanFailure

Type	Default Severity	Description
IDSLTimingLossAlarm	Major	Sent by an IDSL line module when the network timing is lost. *Information included in the event:* Board PII Configured timing mode *Cleared by:* IDSLTimingLossClear *Trap:* cmIDSLBoardTimingLoss
LinkDown	Minor	Sent by the System Control Module when a DSL link, Frame Relay link, or ATM link transitions to disabled from the enabled state. *Information included in the event:* Port index Port name Port PII Group name *Cleared by:* LinkUp *Trap:* linkDown
LoginSaturated	Warning	Sent by the System Control Module when an operator login is attempted and the number of active sessions is already at the maximum. *Information included in the event:* Operator IP Reason *Cleared by:* LoginAvailable *Trap:* cmLoginsSaturated
LoginSuspended	Warning	Sent by System Control Module when three successive login attempts for a specific user are unsuccessful. This is cleared when the suspended user logs in successfully. *Information included in the event:* Operator IP Reason *Cleared by:* LoginAllowed *Trap:* cmLoginsSuspended
MaintFailed	Minor	Sent by System Control Module when a maintenance command fails. *Information included in the event:* Maintenance command attempted Status of command attempted Reason code Status string *Cleared by:* MaintSucceeded *Trap:* cmMaintCmdStatusChange

Type	Default Severity	Description
PortMisprovisioned	Minor	Sent by the System Control Module for a DSL port when the line trains but is not configured; that is, the operational state of the port is Up, but the port is not configured. An interface is considered as “configured” if it has a valid entry in the cmIfaceTable. This event alerts the operator that a CPE is trying to connect on a port that is not configured. *Information included in the event:* Status string *Cleared by:* Manually *Trap:* cmPortMisprovisioned
PowerSupplyFault	Major	Sent by the System Control Module when a Power Supply fault is detected. *Information included in the event:* None *Cleared by:* PowerSupplyClear *Trap:* cmPowerSupplyFailure
RateFallbackAlarm	Minor	Sent by the DSL line module when the link trains at rate lower than the configured data rate. *Information included in the event:* Port PII Current data rate *Cleared by:* RateFallbackClear *Trap:* cmRateFallback
RedundancyChanged	Minor	Sent by the System Control Module when a change in the redundancy state (Enable/disable) is detected. *Information included in the event:* SCM board PII Redundancy is enabled or disabled *Cleared by:* Manually *Trap:* cmRedundancyChange
RedundancyConflict	Minor	Sent by the Secondary System Control Module when a secondary module thinks it should be locked as primary, but the other side is already locked. Both modules are trying to assume the Primary role. *Information included in the event:* Index *Cleared by:* Manually *Trap:* cmRedundancyConflict
RoleChanged	Major	Sent by the System Control Module when its redundancy role (Primary/Secondary) changes. *Information included in the event:* SCM board PII New primary SCM board PII Failover was automatic or manual *Cleared by:* Manually *Trap:* cmRoleChange

Type	Default Severity	Description
SONETLineALarm	Major	Sent by the OC-3c/STM-1 Module when line status changes to *Error*. *Information included in the event:* IF Index of SONET Line Value of LineStatus object *Cleared by:* SonetLineStatusClear *Trap:* cmSONETLineStatusChange
SONETPathALarm	Major	Sent by the System Control Module when its redundancy role (Primary/Secondary) changes. *Information included in the event:* IF Index of SONET Path Value of PathStatus object *Cleared by:* SonetPathStatusClear *Trap:* cmSONETPathStatusChange
SONETSectionALarm	Major	Sent by the System Control Module when its redundancy role (Primary/Secondary) changes. *Information included in the event:* IF Index of SONET Section Value of SectionStatus object *Cleared by:* SonetSectionStatusClear *Trap:* cmSONETSectionStatusChange
WANLinkDown	Critical	Sent by the System Control Module when a WAN link state transitions to disabled from the enabled state. *Information included in the event:* Port index Admin Status OperStatus Port name Port PII Group name *Cleared by:* WANLinkUp *Trap:* WANLinkDown

Alarm-Clearing Notifications

Type	Default Severity	Description
AtmVccUp	Notification	Sent by the System Control Module when an ATM WAN VC OperState changes to Up or enabled. *Information included in the event:* PII of the VC VPI of the VC VCI of the VC Operational status of the VC Cause of the operational status *Trap cleared by this advisory:* AtmVccDown *Trap:* Enterprise
BoardUp	Notification	Sent by the System Control Module when a new module is detected in the system. *Information included in the event:* Index *Trap cleared by this advisory:* BoardDown *Trap:* cmBoardUp
BootFileSucceed	Notification	Sent when a file download request to a module is successfully carried out. *Information included in the event:* File name Board PII Reason code Descriptive text *Trap cleared by this advisory:* BootFileFail *Trap:* cmBootFileOK
ConfigReadSucceed	Notification	Sent by the System Control Module when the Config file is successfully read. *Information included in the event:* File name *Trap cleared by this advisory:* ConfigReadFailed *Trap:* cmConfigFileReadOk
ConfigWriteSucceed	Notification	Sent by the System Control Module when the config file is successfully written. *Information included in the event:* File name *Trap cleared by this advisory:* ConfigWriteFailed *Trap:* cmConfigFileWriteOk
CpePlugAndPlayClear	Notification	Sent by the System Control Module when a CPE Plug and Play failure has been cleared. *Information included in the event:* Status string *Trap cleared by this advisory:* CpePlugAndPlayFailure *Trap:* cmCpePlugAndPlayClear

Type	Default Severity	Description
DLCIStateEnabled	Notification	Sent by the System Control Module when a Frame Relay PVC becomes enabled. *Information included in the event:* IF index DLCI State Port name Port PII Group name *Trap cleared by this advisory:* DLCIStateDisabled *Trap:* frDLCIStatusChange
DS1LineStatusClear	Notification	Sent by a Quad T1 Wan module or a T1 line module when line status changes to a no-error status. *Information included in the event:* Value of DS1 line status Time at last line status change IF index of DS1 line Port name Port PII Group name *Trap cleared by this advisory:* DS1LineStatusAlarm *Trap:* dsx1LineStatusChange
DS3LineStatusClear	Notification	Sent by a DS3 module when line status changes to a no-error status. *Information included in the event:* Value of DS3 line status Time at last line status change IF index of DS3 line Port name Port PII Group name *Trap cleared by this advisory:* DS3LineStatusAlarm *Trap:* dsx3LineStatusChange
FanFaultClear	Notification	Sent by the System Control Module when a Fan fault is cleared. *Information included in the event:* None *Trap cleared by this advisory:* FanFault *Trap:* cmFanFailureClear
IDSLTimingLossClear	Notification	Sent by an IDSL line module when the network timing is reacquired. *Information included in the event:* Board PII Configured timing mode *Trap cleared by this advisory:* IDSLTimingLossAlarm *Trap:* cmIDSLBoardTimingLoss

Type	Default Severity	Description
LinkUp	Notification	Sent by the System Control Module when a DSL link, Frame Relay Link, or ATM link transitions to enabled from the disabled state. *Information included in the event:* Port index Port name Port PII Group name *Trap cleared by this advisory:* LinkDown *Trap:* linkUp
LoginAllowed	Notification	Sent by the System Control Module when a successful login ends a login suspension for a particular operator. *Information included in the event:* Operator IP Reason *Trap cleared by this advisory:* LoginSuspended *Trap:* cmLoginsAvailable
LoginAvailable	Notification	Sent by the System Control Module when a login session ends and causes an existing LoginsSaturated condition to clear. *Information included in the event:* Operator IP Reason *Trap cleared by this advisory:* LoginSaturated *Trap:* cmLoginsAvailable
MaintSucceeded	Notification	Sent by the System Control Module when a maintenance command succeeds. *Information included in the event:* Maintenance command attempted Status of command attempted Reason code Status string *Trap cleared by this advisory:* MaintFailed *Trap:* cmMaintCmdStatusChange
PortMisprovisioned-Clear	Notification	Sent by the System Control Module when a portmisprovisioned alarm has been cleared. *Information included in the event:* Status string *Trap cleared by this advisory:* PortMisprovisioned *Trap:* cmPortMisprovisionedClear
PowerSupplyClear	Notification	Sent by the System Control Module when a Power Supply fault clears. *Information included in the event:* None *Trap cleared by this advisory:* PowerSupplyFault *Trap:* cmPowerSupplyFailure

Type	Default Severity	Description
RateFallbackClear	Notification	Sent by a DSL line module when a link is not trained. *Information included in the event:* Port PII Current data rate *Trap cleared by this advisory:* RateFallbackAlarm *Trap:* cmRateFallbackClear
SONETLineStatusClear	Notification	Sent by the OC-3c/STM-1 Module when line status changes to normal (no errors). *Information included in the event:* PII (IfIndex) associated with the SONET line Current value of SONETLineStatus *Trap cleared by this advisory:* SONETLineStatusALarm *Trap:* cmSONETLineStatusChange
SONETPathStatusClear	Notification	Sent by the OC-3c/STM-1 Module when path status changes to normal (no errors). *Information included in the event:* PII (IfIndex) associated with the SONET path Current value of SONETPathStatus *Trap cleared by this advisory:* SONETPathStatusAlarm *Trap:* cmSONETPathStatusChange
SONETSectionStatusClear	Notification	Sent by the OC-3c/STM-1 Module when section status changes to normal (no errors). *Information included in the event:* PII (IfIndex) associated with the SONET section Current value of SONETSectionStatus *Trap cleared by this advisory:* SONETSectionStatusAlarm *Trap:* cmSONETSectionStatusChange
WANKLinkUp	Notification	Sent by the System Control Module whenever a WAN link (FR or ATM) state transitions from disabled to enabled. *Information included in the event:* Port PII (from IfIndex) Admin status (from IfTable) Oper Status (from IfTable) Port Name (from cmNameTable) Group Name (from cmGroupTable) *Trap cleared by this advisory:* WANLinkDown *Trap:* cmWANLinkUp

Event Notifications

Type	Severity	Description
atmfVccChange	Minor	The attributes of the ATM WAN VC have been modified. *Information included in the event:* IP of the VC VPI of the VC VCI of the VC *Trap:* atmfVccChange
AttributeChanged	Notification	Sent by the System Control Module when a Set command completes successfully (a group, instance, or object has changed value). There is one event per attribute changed. *Information included in the event:* Operator name Operator IP Operator context String with Group, Instance, Object, Value *Trap:* cmAttributeChange
BoardRestart	Notification	Sent by any module before a software initiated reset due to the reason supplied. *Information included in the event:* Reason code Descriptive text *Trap:* cmBoardRestart
ColdStart	Notification	Sent by the System Control Module when it powers up. *Information included in the event:* None *Trap:* coldStart
Diagnostic	Notification	Sent by any module when a miscellaneous unexpected event occurs that the operator should know about. *Information included in the event:* Text *Trap:* cmDiagnostic
FallingThresholdAlert	Notification	Sent by the System Control Module when a performance value falls below its threshold. *Information included in the event:* Index, Threshold variable, Sampling method, Actual value, Threshold value to compare against, Port name, Port PII, Group name *Trap:* cmFallingAlarm
LoginFailed	Notification	Sent by the System Control Module when an attempted operator login fails; the authentication is invalid or incorrect. *Information included in the event:* Operator IP (Telnet sessions only) Name, Context, Server-IP-Address (config for RADIUS authentication, but unable to contact RADIUS server) Name, Context, Server-IP-Address, Error-msg-reply-from-server (if login rejected by RADIUS server) *Cleared by:* None *Trap:* cmLoginError

Type	Severity	Description
LoginSucceeded	Notification	Sent by the System Control Module when an attempted operator login succeeds; the authentication is valid. *Information included in the event:* Operator IP (Telnet sessions only) Name, Context, Server-IP-Address (Radius authentication) *Trap:* cmLoginOK
Logout	Notification	Sent by the System Control Module when an operator logs out. *Information included in the event:* Operator IP Reason *Trap:* cmLogout
LoopStatusChange	Notification	Sent by the System Control Module at the start and end of a loopback test. *Information included in the event:* Status of loopback test Unique identifier for this test PII of interface being tested Type of test being performed Person or entity running the test If Status=Done, the termination code If Status=Done, the total elapsed time of the test If Status=Done, the total transmit count for the test If Status=Done, the total received errors for the test Test sub type Additional error information Port name Port PII Group name *Trap:* cmLoopStatusChange
RadiusLocalFallback	Notification	Sent by the SCM when none of the specified RADIUS servers can be found and the DSLAM is attempting to fall back to local authentication. *Information included in the event:* Operator IP Operator Name Operator Context Reason for fallback (text string) *Trap:* cmLocalFallback
RisingThresholdAlert	Notification	Sent by the System Control Module when a performance value exceeds its threshold. *Information included in the event:* Index Threshold variable Sampling method Actual value Threshold value to compare against Port name Port PII Group name *Trap:* cmRisingAlarm

Appendix D
Software Upgrade Procedures

This appendix provides procedures for installing upgrades to the CE200 software. Installation of a software upgrade is, by definition, a service-affecting event and should only be performed during scheduled maintenance at off-peak times.

If you receive notice of a software upgrade to the CE200, be sure to read any printed materials or files that accompany the new software. Those specific instructions will always be more important; the procedures presented here are a more generalized view of the sequence required to update the CE200 file system.

In the procedures, the term *local machine* refers to the computer (such as a PC) that you are using to control and manage the installation of new/upgraded software. We assume that your local machine is provisioned with suitable FTP and Telnet applications and that you are familiar with their use.

For CE200s Running V2.1 or Later

Beginning with Version 2.1, the CopperEdge system software supports enhanced versions of CE200 system control modules that include a 128 MB IDE flash disk.

This appendix is intended for use with CE200s that are already running Version 2.1 or later, and which are upgrading to a newer version of the software, whether a full version change or an interim release. There are two important considerations concerning these procedures:

- In a CE200 System Control Module equipped with an IDE flash disk, the software files (Code subdirectory) and Config file (System subdirectory) reside on the IDE flash disk (drive Q:). The SCM file resides on the main disk (drive P:).

  If the System Control Module does not have an IDE disk, all files reside on the main disk (drive P:).
- Directory and file names on the IDE disk are case-sensitive, so to ensure consistency and prevent the creation of unusable duplicate files, use lower case for all directory and file names.

Drive letters are also case-sensitive, so use upper case when specifying the P: or Q: drives. Throughout the text, directory and file name are capitalized to make them easier to read.

- Only the SCM-2 has IDE flash memory. Neither the SCM-1 nor the SCM-3 has IDE flash memory.
- Support for the SCM-3 was added to Version 4.0.

If this CopperEdge unit is running any release of version 2.1 or older of the software, use the *CopperEdge 200 Release 2.1 Software Upgrade Instructions*, available through your Copper Mountain Support FTP account.

This appendix includes separate procedures for upgrading CE200s equipped with IDE flash disks, and for CE200s that do not have IDE flash disks.

Determine if an IDE Flash Disk Exists

Before you continue, see if your system is equipped with an IDE Flash Disk:

1. Initiate an FTP session to the CE200 and log in.

   ```
   C:\>ftp DSLAM_IpAddress
   Connected to DSLAM_IpAddress
   User <DSLAM_IpAddress:<none>>:
   ```

2. Enter **ce200** when you are asked for the user name and password.
3. Change to drive Q:

   ```
   ftp> cd Q:
   ```

4. If the CE200 has an IDE Flash Disk, the following will be displayed:

   ```
   250 Changed directory to "Q:"
   ```

   Log off the ftp session and follow the procedure starting with "Upgrade Procedure, Units with IDE Flash Disk" on page 222.

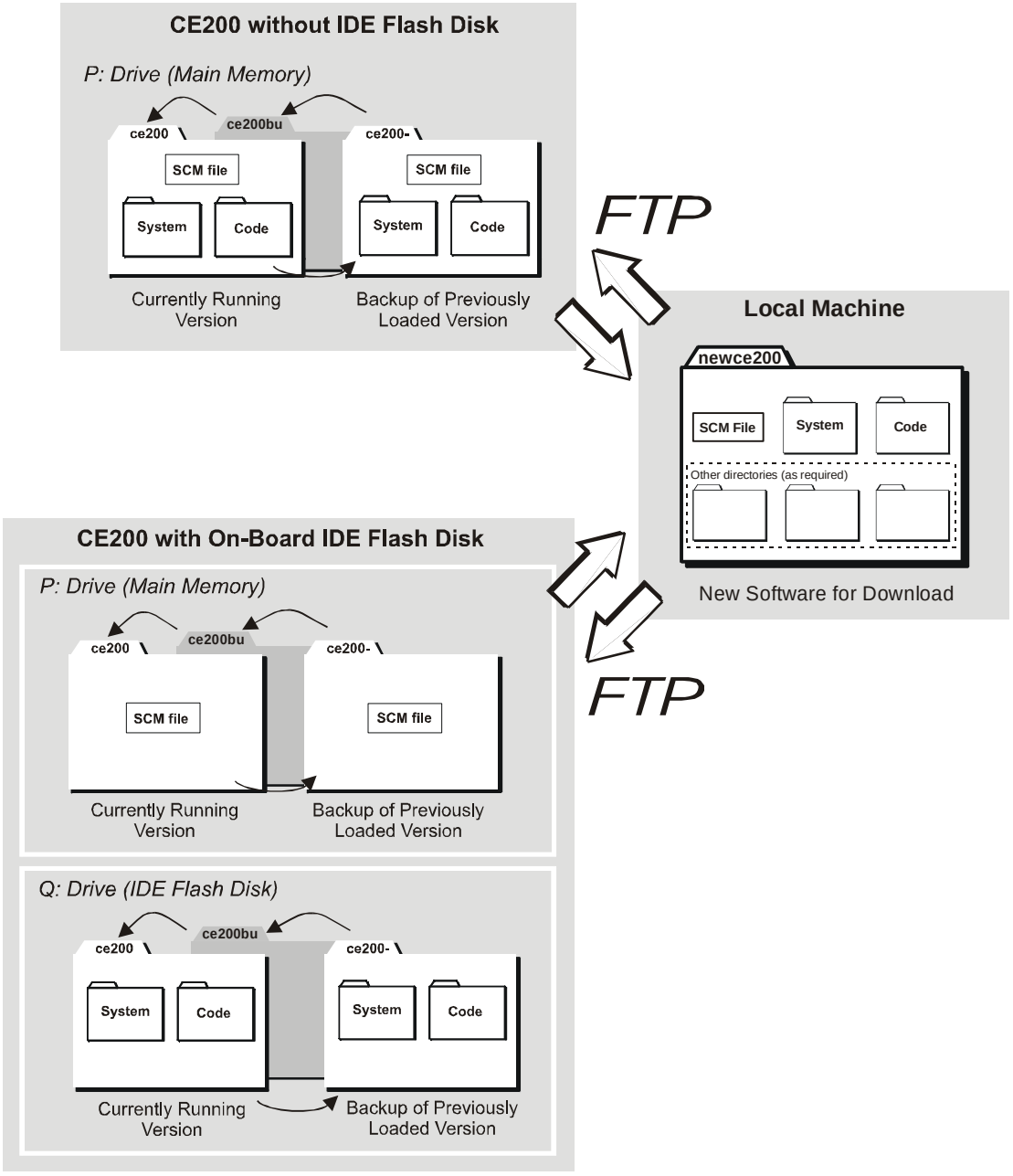
5. If the CE200 does not have an IDE Flash Disk, the following will be displayed:

   ```
   501 Directory non existent or syntax error
   ```

   Log off the ftp session and follow the procedure starting with "Upgrade a CE200 that has no IDE Flash Disk" on page 217.

Software Upgrade Overview

While specific software upgrades may have unique aspects, the basic tasks and their sequence include the following steps, as illustrated in the following diagram.



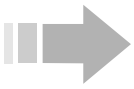
Directory Structure for Software Upgrade

1. On the local machine, prepare a mirror of the CE200 directories containing the new software.
2. Telnet to the CE200 and save its current configuration (this updates the Config file).

3. Exit the Telnet session and then FTP to the CE200.
4. Rename the backup directory to deactivate it (rename ce200- to ce200bu).
5. Rename the current directory to make it the new backup directory (rename ce200 to ce200-).
6. Set the FTP utility to Binary mode and make sure the Prompt mode is turned on.

   If the screen prompt indicates that you have uploaded a file in ASCII mode, Binary mode has not been turned on. Delete the ASCII-transferred file to the CE200, and upload the file in Binary mode.

   By default, the FTP Prompt feature is turned on. Because of the switching among multiple directories on the local machines and the CE200s, complexity is built into the upgrade process. Keep the FTP Prompt feature active as a safety measure.
7. In the ce200bu directory, delete the System Control Module (SCM) file, and then upload (FTP *put*) the new SCM file from the local machine to the CE200.
8. If there is an IDE Flash Disk, first change to drive Q: and then perform the following steps.
9. In each subdirectory in the ce200bu directory, delete the files and then upload (FTP *put*) the corresponding new files from the local machine to the CE200.
10. Restore the system configuration by copying the Config file from the ce200- directory (FTP *get*) to the ce200bu directory (FTP *put*).
11. Rename the ce200bu directory (which now contains your new software) to ce200, thus making it the default code base when the CE200 starts up.
12. If this CE200 is equipped for Control and WAN redundancy, end the FTP session with the primary Control module, and repeat the previous steps for the redundancy complex.
13. Restart the CE200 to activate the new software.
14. If the upgrade applies to the CopperRocket CPE as well as to the CE200, run the upgrade command (cmCPEBoard group) for each connected CPE unit. (See “Upgrade the CPE Software” on page 229.)



N O T E

*The compressed configuration file (config.tgz) is **not** compatible with any version of CE200 software prior to Version 3.0.*

FTP Errors

If you quickly perform several FTP operations, you may see a transient error from the CE200 because it has run out of sockets for an operation. Transient errors are numbered from 400 to 499, as in the following example:

```
ftp> dir
200 Port set okay
426 Data Connection error
```

Wait a few seconds and try the operation again. Under normal conditions, the problem will clear within 60 seconds.

Upgrade a CE200 that has no IDE Flash Disk

If your CE200 does not have an IDE Flash Disk, this section provides the generic procedures for upgrading the system software. If there is a conflict between these procedures and the specific instructions accompanying the new version of your software, follow the version-specific instructions.

Be sure to perform these procedures at the local machine (such as a PC) from where the new software will be copied.

Prepare the New Software on the Local Machine

1. On the local machine, create a directory (such as *newce200*).
2. Copy the new software to the newce200 directory.

   The new software is located at the Copper Mountain ftp site; you can access it directly through ftp or through the Copper Mountain web site. If you do not know how to access the ftp site, contact the Copper Mountain Technical Assistance Center.

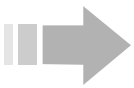
   Although individual upgrades may vary, your new directory should contain at least the following file and directories:

   scm—a file with the new System Control Module code

   code—a directory containing the software files

   system—a directory containing the Config file

   These items are identical to the contents of the ce200 directory on the CE200.
3. Set the local machine's default directory to newce200.



N O T E

If the file or directory structure of the new software is not the same as shown here, refer to the instructions/release notes accompanying the new software for specific instructions.

Save the Current Config File on the CE200

1. Use Telnet or an SNMP Manager to connect to the CE200.
2. Log in to the CE200.

3. Enter **ce200** when you are asked for the login name and password.

4. At the prompt, save the current configuration:

```
CRAFT> set cmsystem command=save
```

5. Verify that the save was successful:

```
CRAFT> get cmsystem
Group: cmSystem
     .
     .
     .
CommandStatus        =  Succeeded
```

6. If the CommandStatus object displays `InProgress` instead of `Succeeded`, repeat the **get cmsystem** command.

   If it displays `Failed`, contact Copper Mountain Technical Support.

7. Log out of the CE200.

Upgrade the Software on the CE200

Rename the Current and Backup Directories

1. Initiate an FTP session to the CE200 and log in.

```
C:\>ftp DSLAM_IpAddress
Connected to DSLAM_IpAddress
User <DSLAM_IpAddress:<none>>
```

2. Enter **ce200** when you are asked for the user name and password.

3. Check the current directory structure:

```
ftp> dir
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------       ------     ------    --------
  512        Nov-09-2001  12:08:08   CE200      <DIR>
  512        Nov-09-2001  12:08:08   CE200-     <DIR>

1024 bytes.  1248768 bytes free.
226 Transfer complete
252 bytes received in 0.00 seconds <252000.00 kBytes/sec>
```

   Note that two directories are listed with similar names: ce200 and ce200- ("ce200 minus"). The ce200- directory contains the code and supporting files for the last version of the software that ran on this CE200 *before* the version currently installed.

   If the CE200 is unable to boot from its current software (contained in the ce200 directory), it automatically looks for the ce200- directory and tries to boot using the previous software version.

4. Rename the ce200- directory to ce200bu. This version of the software is now obsolete and is about to be replaced:

```
ftp> rename ce200- ce200bu
350 Accepted source filename. Ready for destination name
354 Rename done.
```

5. Rename the ce200 directory to ce200-. The current code is now designated as the backup:

```
ftp> rename ce200 ce200-
350 Accepted source filename. Ready for destination name
354 Rename done.
```

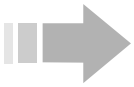
Upload New Files

1. On the CE200, change to the ce200bu directory:

```
ftp> cd ce200bu
250 Changed directory to "P:/ce200bu"
```

2. List the ce200bu directory:

```
ftp> dir
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------       ------     ------    --------
     512    Jan-01-2001  12:08:08   .          <DIR>
     512    Jan-01-2001  12:08:08   ..         <DIR>
     512    Jan-01-2001  12:08:08   code       <DIR>
     512    Jan-01-2001  15:01:32   system     <DIR>
 2287440    Jan-29-2001  12:08:08   scm

2289488 bytes.  56043520 bytes free.
226 Transfer complete
431 bytes received in 0.01 seconds <43.10 kBytes/sec>
```



N O T E

You will only be upgrading the code and system directories. Ignore any other directories (such as html, image, or other).

3. Set up the FTP utility:

   f) Set the FTP utility to interactive mode:

```
ftp> prompt
    Interactive mode On.
```

   If the following message is displayed, enter **prompt** again to turn Interactive mode on:

```
Interactive mode Off.
```

   g) Set the FTP utility to binary code transfer mode:

```
ftp> bin
    200 Type set to I, binary mode
```

   h) Set the FTP utility to view the progress of your uploads:

```
ftp> hash
Hash mark printing On <2048 bytes/hash mark>
```

4. On the local machine, change to the new software directory:

```
ftp> lcd c:\newce200
    Local directory now C:\newce200
```

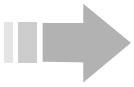
5. On the CE200, delete the SCM file in the ce200bu directory:

```
ftp> del scm
    250 Command accepted.
```

6. Upload the new SCM file from the local machine to the CE200:

```
ftp> put scm
    200 Port set okay
    150 Opening BINARY mode data connection
    ############################################################
    ############################################################
    ############################################################
    ############################################################
```

```
############################################################
############################
226 Transfer complete
2254784 bytes sent in 50.94 seconds <44.26 kBytes/sec>
```



N O T E

When transferring software using FTP, a certain amount of delay is normal. Do not interrupt the process and try to start over—the original file on which the processor is running has been deleted from flash. Note also that while upload of the SCM file is in progress, the CopperCraft command line interface will not respond to commands. Wait until the SCM upload is complete before you enter further commands. If upload of the SCM file has not completed after 20 minutes, contact Copper Mountain's Technical Assistance Center.

7. Upgrade the Code subdirectory:

   a) Change to the Code subdirectory on both the CE200 and local machines:

   ```
   ftp> cd code
       250 Changed directory to "P:/ce200bu/code"

   ftp> lcd code
       Local directory now C:\newce200\code
   ```

   b) On the CE200, delete all *.bgz or *.bin files, one by one:

   ```
   ftp> del bc.bgz  or  del bc.bin
       250 Command accepted
   ```

   c) Upload the new code files (*.bgz) from the local machine to the CE200:

   ```
   ftp> mput *.bgz
       mput bc.bgz?
   ```

   Enter **y** for each file to confirm the upload.

   ```
       200 Port set okay
       150 Opening BINARY mode data connection
       ####
       226 Transfer complete
       14509 bytes sent in 0.10 seconds <145.09 kBytes/sec>
   ```

8. Upgrade the System subdirectory:

   a) *On the local machine only*, change to the System subdirectory:

   ```
       ftp> lcd ../system
       Local directory now C:\newce200\system
   ```

   b) On the CE200, change to the System subdirectory in the ce200- directory. It contains the Config file that you saved at the beginning of this process:

   ```
       ftp> cd /ce200-/system
       250 Changed directory to "P:/ce200-/system"
   ```

   c) Copy the Config file:

   ```
   ftp> get config.tgz  or  get config.txt
       200 Port set okay
       150 Opening BINARY mode data connection
       #
       226 Transfer complete
       3218 bytes received in 0.17 seconds <18.93 kBytes/sec>
   ```

d) Return to the System subdirectory in the ce200bu directory:

```
ftp> cd /ce200bu/system
    250 Changed directory to "P:/ce200bu/system"
```

e) Delete the existing Config file in the ce200bu directory:

```
ftp> del config.tgz  or  del config.txt
250 Command accepted.
```

f) Paste the Config file that you copied in step c:

```
ftp> put config.tgz  or  put config.txt
200 Port set okay
150 Opening BINARY mode data connection
#
226 Transfer complete
3218 bytes sent in 0.17 seconds <18.93 kBytes/sec
```

9. At this point, the ce200bu directory on the CE200 contains all of the files needed for your upgrade.

10. Return to the system root directory:

```
ftp> cd /
250 Changed directory to "P:/"
```

Rename the Directory CE200 the New Software

From the CE200 root directory, rename the newly rebuilt ce200bu directory to the default name for the CE200 software (ce200):

```
ftp> rename ce200bu ce200
350 Accepted source filename. Ready for destination name
354 Rename done.
```

Upgrade the Redundancy Complex

If this CE200 is equipped for Control and WAN Redundancy, the software on the redundant (Backup) side must also be upgraded.

Log off your FTP session on the Preferred side, and log in to the Backup side. Repeat the previous steps for the redundancy complex, beginning with "Upgrade the Software on the CE200" on page 223.

Restart the CE200 to Implement the New Software

1. Exit the FTP session:

```
ftp> bye
```

2. Use Telnet or the serial (Craft*)* interface to connect to the CE200.

3. Log in to the CE200, using **ce200** for the login name and password.

4. At the prompt, restart the system:

```
CRAFT> set cmsystem command=restart
Set Successful
```

It may take several minutes for the system to restart. During this time, the command line interface will not accept commands

A good indicator that enough time has elapsed for the restart sequence to complete is when connection to the CE200 is lost.

5. When the system has completed its restart sequence, log into the CE200, using **ce200** for the login name and password.

6. Verify that the new code is running on all System Control, Buffer Control, WAN, and DSL modules:

```
CRAFT> geta cmboard swver
SwVersion

Instance: [1.2.0.0]
5.0.83

Instance: [1.3.0.0]
5.0.83

Instance: [1.4.0.0]
5.0.83

Instance: [1.5.0.0]
5.0.83

Instance: [1.6.0.0]
5.0.83

Instance: [1.7.0.0]
5.0.83

Instance: [1.8.0.0]
5.0.83
```

   Check that the displayed software version matches the version of your upgrade.

Upgrade Procedure, Units with IDE Flash Disk

If your CE200 is equipped with an IDE Flash Disk, this section provides the generic procedures for upgrading the system software on both drives P: and Q:. If there is a conflict between these procedures and the specific instructions accompanying the new version of your software, follow the version-specific instructions.

Be sure to perform these procedures at the local machine (such as a PC) from where the new software will be copied.

Prepare the New Software on the Local Machine

1. On the local machine, create a new directory (such as *newce200*).

2. Copy the new software to the newce200 directory.

   The new software is located at the Copper Mountain ftp site; you can access it directly through ftp or through the Copper Mountain web site. If you do not know how to access the ftp site, contact the Copper Mountain Technical Assistance Center.

   Although individual upgrades may vary, your new directory should contain at least the following file and directories:

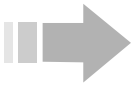
   scm—a file with the new System Control Module code

   code—a directory containing the software files

   system—a directory containing the Config file

   These items are identical to the contents of the ce200 directory on the CE200.

3. Set the local machine's default directory to newce200.



N O T E

If the file or directory structure of the new software is not the same as shown here, refer to the instructions/release notes accompanying the new software for specific instructions.

Save the Current Config File on the CE200

1. Use Telnet or an SNMP Manager to connect to the CE200.
2. Log in to the CE200.
3. Enter **ce200** when you are asked for the login name and password.
4. At the prompt, save the current configuration:

```
CRAFT> set cmsystem command=save
```

5. Verify that the save was successful:

```
CRAFT> get cmsystem
Group: cmSystem
ObjectClass          =  System
OperState            =  Enabled
Version              =  E 3.0
Master               =  0.0.0.0
ConfigFileName       =  config.tgz
CalendarTime         =  2002/07/21-10:01:11
MyPII                =  1.2.0.0
PrimaryPII           =  1.2.0.0
SecondaryPII         =  0.0.0.0
Redundancy           =  NotAvailable
ShelfCount           =  1
ExpIpSubNet          =  192.168.250.0
ConfigSynch          =  Saved
Command              =  SaveConfig
CommandStatus        =  Succeeded
```

6. If the CommandStatus object displays `InProgress` instead of `Succeeded`, repeat the **get cmsystem** command.

   If it displays `Failed`, contact Copper Mountain Technical Support.

7. Log out of the CE200.

Upgrade the Software on the CE200

The following steps describe how to upgrade the SCM file on drive P: and the software files on drive Q:.

Rename the Directories on Drive P:

1. Initiate an FTP session to the CE200 and log in.

```
C:\>ftp DSLAM_IpAddress
Connected to DSLAM_IpAddress
User <DSLAM_IpAddress:<none>>
```

2. Enter **ce200** when you are asked for the user name and password.
3. Check the current directory structure:

```
ftp> dir
```

```
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------       ------     ------    --------
     512    Oct-05-2002  12:08:08   CE200       <DIR>
     512    Oct-05-2002  12:08:08   CE200-      <DIR>

    1024 bytes.  3210240 bytes free.
226 Transfer complete
252 bytes received in 0.01 seconds <25.20 kBytes/sec>
```

Note that two directories are listed with similar names: ce200 and ce200- ("ce200 minus").

The ce200- directory contains the last version of the SCM file that ran on this CE200 *before* the version currently installed.

4. Rename the ce200- directory to ce200bu. This version of the software is now obsolete and is about to be replaced:

```
ftp> rename ce200- ce200bu
350 Accepted source filename. Ready for destination name
354 Rename done.
```

5. Rename the ce200 directory to ce200-. The current code is now designated as the backup:

```
ftp> rename ce200 ce200-
350 Accepted source filename. Ready for destination name
354 Rename done.
```

Upload the New SCM File to Drive P:

1. On the CE200, change to the ce200bu directory:

```
ftp> cd ce200bu
250 Changed directory to "P:/ce200bu"
```

2. List the ce200bu directory:

```
ftp> dir
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------       ------     ------    --------
     512    Oct-05-2001  12:08:08   .          <DIR>
     512    Oct-05-2001  12:08:08   ..         <DIR>
 1899688    Jul-27-2002  15:01:32   SCM

 1900702 bytes.  3210240 bytes free.
226 Transfer complete
308 bytes received in 0.03 seconds <10.27 kBytes/sec>
```

3. Set up the FTP utility:

   g) Set the FTP utility to interactive mode:

```
ftp> prompt
Interactive mode On.
```

   If the following message is displayed, enter **prompt** again to turn Interactive mode on:

```
Interactive mode Off.
```

   h) Set the FTP utility to binary code transfer mode:

```
ftp> bin
200 Type set to I, binary mode
```

   i) Set the FTP utility to view the progress of your uploads:

```
ftp> hash
Hash mark printing On <2048 bytes/hash mark>
```

4. On the local machine, change to the new software directory:

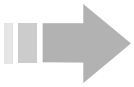
```
ftp> lcd c:\newce200
Local directory now C:\newce200
```

5. On the CE200, delete the SCM file in the ce200bu directory:

```
ftp> del scm
250 Command accepted.
```

6. Upload the new SCM file from the local machine to the CE200:

```
ftp> put scm
200 Port set okay
150 Opening BINARY mode data connection
############################################################
############################################################
############################################################
############################################################
################################
226 Transfer complete
2254784 bytes sent in 50.94 seconds <44.26 kBytes/sec>
```



N O T E

When transferring software using FTP, a certain amount of delay is normal. Do not interrupt the process and try to start over—the original file on which the processor is running has been deleted from flash. Note also that while upload of the SCM file is in progress, the CopperCraft command line interface will not respond to commands. Wait until the SCM upload is complete before you enter further commands. If upload of the SCM file has not completed after 20 minutes, contact Copper Mountain's Technical Assistance Center.

7. Return to the system root directory:

```
ftp> cd /
250 Changed directory to "P:/"
```

Rename the New Software Directory on Drive P:

From the CE200 root directory, rename the ce200bu directory to the default name for the CE200 software (ce200):

```
ftp> rename ce200bu ce200
350 Accepted source filename. Ready for destination name
354 Rename done.
```

Rename the Directories on Drive Q:

1. Change to drive Q:

```
ftp> cd Q:
250 Changed directory to "Q:"
```

2. Check the current directory structure (note that the SCM file is not listed):

```
ftp> dir
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------      ------     ------     --------
     512    Apr-25-2002  12:08:08   ce200-     <DIR>
     512    Apr-25-2002  12:08:08   logs       <DIR>
     512    Sep-28-2002  15:01:32   ce200      <DIR>

 1536 bytes.  37560320 bytes free.
226 Transfer complete
314 bytes received in 0.01 seconds <31.40 kBytes/sec>
```

Note that two directories are listed with similar names: ce200 and ce200- ("ce200 minus"). The ce200- directory contains the code and supporting files for the last version of the software that ran on this CE200 *before* the version currently installed.

If the CE200 is unable to boot from its current software (contained in the ce200 directory), it automatically looks for the ce200- directory and tries to boot using the previous software version.

3. Rename the ce200- directory to ce200bu. This version of the software is now obsolete and is about to be replaced:

```
ftp> rename ce200- ce200bu
350 Accepted source filename. Ready for destination name
354 Rename done.
```

4. Rename the ce200 directory to ce200-. The current code is now designated as the backup:

```
ftp> rename ce200 ce200-
350 Accepted source filename. Ready for destination name
354 Rename done.
```

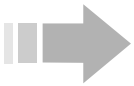
Upload New Software Files to Drive Q:

1. On the CE200, change to the ce200bu directory:

```
ftp> cd ce200bu
250 Changed directory to "Q:/ce200bu"
```

2. List the ce200bu directory:

```
ftp> dir
200 Port set okay
150 Opening ASCII mode data connection
  size          date       time       name
--------       ------     ------    --------
     512    Apr-25-2002  12:08:08   .          <DIR>
     512    Apr-25-2002  12:08:08   ..         <DIR>
     512    Apr-25-2002  12:08:08   html       <DIR>
     512    Apr-25-2002  12:08:08   code       <DIR>
     512    Sep-28-2002  15:01:32   system     <DIR>
     512    Apr-25-2002  12:08:08   image      <DIR>
     512    Apr-25-2002  12:08:08   other      <DIR>

 3584 bytes.  37560320 bytes free.
226 Transfer complete
558 bytes received in 0.02 seconds <27.90 kBytes/sec>
```



N O T E

You will only be upgrading the code and system directories. Ignore any other directories (such as html, image, or other).

3. Upgrade the Code subdirectory:

   a) Change to the Code subdirectory on both the CE200 and local machines:

```
ftp> cd code
250 Changed directory to "Q:/ce200bu/code"

ftp> lcd code
Local directory now C:\newce200\code
```

   b) On the CE200, delete all *.bgz or *.bin files, one by one:

```
ftp> del bc.bgz  or  del bc.bin
250 Command accepted
```

c) Upload the new code files (*.bgz) from the local machine to the CE200:

```
ftp> mput *.bgz
mput bc.bgz?
```

Enter **y** for each file to confirm the upload.

```
200 Port set okay
150 Opening BINARY mode data connection
####
226 Transfer complete
14509 bytes sent in 0.10 seconds <145.09 kBytes/sec
```

4. Upgrade the System subdirectory:

a) *On the local machine only*, change to the System subdirectory:

```
ftp> lcd ../system
Local directory now C:\newce200\system
```

b) On the CE200, change to the System subdirectory in the ce200- directory. It contains the Config file that you saved at the beginning of this process:

```
ftp> cd /ce200-/system
250 Changed directory to "Q:/ce200-/system"
```

c) Copy the Config file:

```
ftp> get config.tgz  or  get config.txt
200 Port set okay
150 Opening BINARY mode data connection
#
226 Transfer complete
3218 bytes received in 0.17 seconds <18.93 kBytes/sec>
```

d) On the CE200, return to the System subdirectory in the ce200bu directory:

```
ftp> cd /ce200bu/system
250 Changed directory to "Q:/ce200bu/system"
```

e) Delete the existing Config file in the ce200bu directory:

```
ftp> del config.tgz  or  del config.txt
250 Command accepted.
```

f) Paste the Config file that you copied in step c:

```
ftp> put config.tgz or  put config.txt
200 Port set okay
150 Opening BINARY mode data connection
#
226 Transfer complete
3218 bytes sent in 0.17 seconds <18.93 kBytes/sec
```

5. At this point, the ce200bu directory on the CE200 contains all of the files needed for your upgrade, except the SCM file.

6. Return to the system root directory:

```
ftp> cd /
250 Changed directory to "Q:/"
```

Rename the New Software Directory on Drive Q:

From the CE200 root directory, rename the newly rebuilt ce200bu directory to the default name for the CE200 software (ce200):

```
ftp> rename ce200bu ce200
350 Accepted source filename. Ready for destination name
354 Rename done.
```

The new software is now resident on drive Q: of the CE200.

Upgrade the Redundancy Complex

If this CE200 is equipped for Control and WAN Redundancy, the software on the redundant (Backup) side must also be upgraded.

Log off your FTP session on the Preferred side, and log in to the Backup side. Repeat the previous steps, beginning with "Upgrade the Software on the CE200" on page 223.

Restart the CE200 to Implement the New Software

1. Exit the FTP session:

   ```
   ftp> bye
   ```

2. Use Telnet or the serial (Craft) interface to connect to the CE200.

3. Log in to the CE200, using **ce200** for the login name and password.

4. At the prompt, restart the system:

   ```
   CRAFT> set cmsystem command=restart
   Set Successful
   ```

   It may take several minutes for the system to restart. During this time, the command line interface will not accept commands.

5. When the system has completed its restart sequence, log in to the CE200, using **ce200** for the login name and password.

   If you cannot establish a connection to the CE200, the restart process has not yet finished.

6. Verify that the new code is running on all System Control, Buffer Control, WAN, and DSL modules:

   ```
   CRAFT> geta cmboard swver
   SwVersion

   Instance: [1.2.0.0]
   5.0.83

   Instance: [1.3.0.0]
   5.0.83

   Instance: [1.4.0.0]
   5.0.83

   Instance: [1.5.0.0]
   5.0.83

   Instance: [1.6.0.0]
   5.0.83

   Instance: [1.7.0.0]
   5.0.83

   Instance: [1.8.0.0]
   5.0.83
   ```

Check that the displayed software version matches the version of your upgrade.

View the Compressed Configuration File

There may be occasions when you want to view the Configuration file (config.tgz), which has been compressed using the gzip utility.

If you do not have the gzip utility on your PC, you can download it from the following web site:

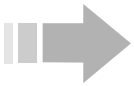
http://www.gzip.org

Follow these steps to view the Configuration file:

1. Through FTP, manually upload the **config.tgz** file to your PC.
2. At the DOS prompt, rename the **config.tgz** file to **config.txt.gz**.
3. Run the gzip utility to unzip the file:

```
gzip -d config.txt.gz
```

The file is unzipped and named config.txt. You can view it in any text editor.



N O T E

Although you can view the config files with a text editor, do not attempt to modify the files except via the CopperCraft interface. Using a text editor to modify the files could result in file corruption with consequent loss of configuration data.

Upgrade the CPE Software

If an upgrade affects operation of CPEs attached to this CE200, the CPEs may also need to have their software updated.

CopperRockets, as well as some third-party CPEs (consult the manufacturer's documentation for each equipment type), may be upgraded using software from the CE200 either singly, using *cmCPEBoard Upgrade*, or in groups using *cmEndPointConfig*.

For CPEs with unique capabilities (IMUX/Multilink or IAD devices used with derived voice applications), use the special instructions in "CPE Upgrades from an External Server (IMUX and IAD CPE)" on page 230.

Upgrading the CPE software is service-affecting for the CPE being upgraded. During the upgrade process, communication with the CPE will be interrupted while it reboots, downloads and installs its new code, and then reboots from the new software. Should the CPE fail to train using the new code, the CPE will reboot and request a new download.

Upgrading an Individual CPE

Depending on the PROM version installed, some CopperRocket 201 CPEs do not have the 1.568 Mbps data rate hard-coded in their CPE PROM.

If you are performing upgrades of individual CR201s, be sure to perform the entire procedure below. To upgrade individual CPEs *other than* CR201, however, simply issue the upgrade command shown in step 2.

1. Set the SDSL Module Port to any data rate other than 1.568Mbps:

```
CRAFT> set cmhdslmodem [pii] datarate=784
```

2. Issue the upgrade command to the CPE:

```
CRAFT> set cmcpeboard [cpe:pii] command=upgrade
```

3. When the upgrade is complete, set the SDSL Module Data Rate to any supported rate.

Downloading software to a CPE may require a matter of seveal minutes to complete.

Upgrading Multiple CPEs

To upgrade or otherwise download software to multiple CPEs, it is generally more efficient to designate DSL physical ports that are eligible for bulk CPE download with cmEndPointConfig, and then use the cmMaintCmd group to execute a BulkDownload on all eligible CPEs at the same time.

Note that any necessary speed adjustments are performed automatically and the procedure described in *Upgrading an Individual CPE* is not necessary.

See the *CopperEdge 200 Installation and Operating Guide* for information on bulk downloading software to multiple CPEs.

CPE Upgrades from an External Server (IMUX and IAD CPE)

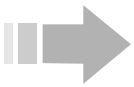
To upgrade the software on connected IMUX CPEs or VoDSL IADs, perform the following steps:

1. Upgrade the software in the CE200 flash (both primary and secondary sides, if equipped with redundant control and WAN modules).
2. Restart the CE200 unit (System Restart).
3. From the *CPE* subdirectory of the *Release* directory on the Copper Mountain Support FTP site, **get** the following files from the subdirectory and transfer them to the host machine to be used as your remote file server. Be sure that the server IP address is reachable from the CE200's route table.

   cpe5_b.bgz (code for CopperRocket 202 IMUX SDSL)

   cpe5_h.bgz (code for CopperRocket 212 IMUX IDSL)

   cpe5_t.bgz (code for CopperRocket 408 IAD)



N O T E

These code files (identified by the .bgz file extension) are contained within a compressed archive. The CopperEdge will automatically expand the files when installing them on the CPE, so do not take any specific action except as specified below.

4. When the system has fully initialized and all connected CPEs have trained, log into the CE200 and configure the cmFile group as follows:

   FSAddr: IP Address of the file server.

FSUsername: User Name used to log in to the file server

FSPass: The Password used to log in to the file server.

FSDir: Directory path to be traversed to reach the file archives transferred in step 3.

5. Once the CPE code files have been placed on the remote server, and the cmFile group has been configured to locate them, follow the procedure in "Upgrading an Individual CPE" on page 229 (in this release, Bulk Upgrade is not supported for IAD or IMUX CPE).

   When you issue the **`upgrade`** command, the CE200 will identify the CPE as a type whose code is stored on the remote server. The files will then be copied to the CE200, expanded, and downloaded to the specified CPE.

   For IMUX CPEs, you only need to enter the PII of *one* of the DSL physical interfaces that comprise the IMUX link.

Appendix E
Glossary

This glossary is provided as an aid to recognizing some of the specialized terms used in this manual that relate to the Copper Mountain DSL system, the CE200 and the CopperView EMS. Selected telephony and data communications terms commonly used in DSL networking are also included.

A

ADSL Asymmetrical DSL. DSL scheme in which upstream and downstream data rates are different, typically with the downstream channel running much faster than the upstream channel.

address A character string used as a unique identifier, usually of a specific location, user, machine, or logical name.

agent An active object with a specific purpose. *See* SNMP Agent.

AIS Alarm Indication Signal used in DS1 and DS3 status messages.

alarm An event category that requires an operator's attention.

ARP Address Resolution Protocol. A protocol within the TCP/IP suite that maps IP addresses to Ethernet addresses. TCP/IP requires ARP for use with Ethernet.

ATM Asynchronous Transfer Mode. A typically high-speed (T3+) method of wide-area data transfer in which the information is organized into cells. Asynchronous in the sense that there may not be any periodicity in the recurrence of information cells originating with a specific end point or user.

B

backup side Slots 14 through 18 in a CE200 chassis. It is also called Side B.

BER Bit Error Rate; also Block Error Rate.

boot, bootstrap The process of starting a computer or a routine, the first few instructions are sufficient to bring the rest of it's necessary components into memory.

BootP server An external device that provides bootstrap data through the Bootstrap Protocol.

bps (bits per second) Measurement unit used to specify the rate or speed of data communications.

Buffer Control Module A module used with the CE200 that manages the transfer of data between the proprietary interface to the DSL Modules, and between the PC interface and the System Control Module.

C

CCV C-bit Coding Violation Error (for C-bit Parity and SYNTRAN DS3 applications).

CE CopperEdge. The generic term that refers to all of the Copper Mountain DSLAM class of equipment. *See also* DSLAM.

CE150, CE200 CopperEdge 150 and CopperEdge 200 are high-density, multi-speed DSL systems (fast packet access concentrators), which allow network service providers to offer high-speed local access to the Internet and to private enterprise networks.

CES C-bit Errored Seconds. A second with one or more CCVs, one or more Out of Frame defects, or a detected incoming AIS.

Clear An event (CE-generated or manual) that causes alarm indications (visual and audible) to disappear. *See also* Event.

CLI Command Line Interface. *See* CopperCraft.

CMCP Copper Mountain Compatible Protocol.

COE Central Office Equipment. Communications apparatus that normally resides in a telco central office or with a Network Service Provider. The CopperEdge 200 is one type of COE.

cold boot The process of restarting a computer system as if it were powered off, then back on. Also called cold start.

complex Consists of a System Control Module (SCM), a Buffer Control Module (BCM), and optionally one or two WAN modules. *See also* Primary complex *and* Secondary complex.

control master The single SCM in a CE system that is the point of egress for all user data traffic to the WAN, and the point of operator management of the system.

control module The System Control Module (SCM) or Buffer Control Module (BCM).

CopperCraft Copper Mountain's product name for its embedded command-line software that utilizes an SNMP-like command language to configure and monitor performance of DSL and other ports on the CE200.

CopperVPN Copper Mountain's name for a networking model by which subscribers are aggregated into a virtual private network of DSL links, connecting transparently with a remote LAN or corporate network, but without reliance on any special capabilities or software in router devices connected to the CoppperVPN circuit.

CPE Customer Premise Equipment. Telecommunications equipment that resides at the subscriber's location. Copper Mountain's CopperRocket 201 SDSL Modem is one type of CPE.

D

datagram A message unit that contains source and destination address information, as well as the data itself, which is routed through a packet-switched network.

DCE Data Communication Equipment. In defining interface standards for connection of equipment to a network, the classification of a device as Data Communications Equipment (typically a modem or printer) or Data Terminal Equipment (typically a computer or remote terminal device) determines the wiring of the interface connector. In networking parlance, the network access device is normally considered the DCE.

DHCP server A host processor accessed through Dynamic Host Configuration Protocol.

DLCI Data Link Circuit Identifier. A logical name (generally 10 bits in length) assigned to a Frame Relay virtual circuit and included in the Frame Relay header.

DLCMI Digital Link Connection Management Interface. Software mechanism for configuring and controlling Frame Relay virtual circuits.

DSL Digital Subscriber Line. A digital telephone line capable of reliably transporting data at higher speeds than either conventional analog mdems or such dedicated digital formats as ISDN (e.g. G.lite DSL speeds range from 160 kbps up to 2 Mbps; G.dmt rates may be as great as 6 Mbps) over ordinary twisted-pair copper loops. Also, the hardware and software technology utilized to implement DSL.

DSLAM Digital Subscriber Line Access Multiplexer. A network device, usually at a telephone company central office, that receives signals from multiple customer DSL connections and sends the signals on a high-speed backbone line using multiplexing techniques. Depending on the product, DSLAM multiplexers connect DSL lines with some combination of asynchronous transfer mode (ATM), Frame Relay, or IP networks.

DSL Module A CE200 module that interfaces between DSL links and the switched network. CE200 DSL modules may either be 12- or 24-port types, and, depending on the model, may support a variety of xDSL formats.

DTE Data Terminal Equipment. In defining interface standards for connection of equipment to a network, the classification of a device as Data Communications Equipment (typically

a modem or printer) or Data Terminal Equipment (typically a computer or remote terminal device) determines the wiring of the interface connector. Network interfaces on the CE200 (Ethernet and WAN) are wired as DTE.

E

Ethernet A local area network used for connecting computers and peripheral devices over short distances and using private facilities. Ethernet operates over twisted pair wire or coaxial cable at rates to 10 Mbps or 100 Mbps.

Event An indication of an occurrence that an operator may want to know about.

F

filter A logical construct that allows selective reception or blocking of data packets based on the criteria (such as the IP address of the originator) assigned and built into the filter specification.

Frame Relay A high-performance shared-bandwidth packet-switching protocol, which has become the de-facto standard in wide-area data communications.

FTP File Transfer Protocol.

G

G.SHDSL Single-pair High Speed Digital Subscriber Line, a symmetric DSL technology based on PAM to provide 256 Kbps to 2.3 Mbps.

H

HDIA High Density Internet Access. A networking model that allowed separate voice and data circuits on DSL and WAN ports and that provided economical use of IP ad- dressing. This netmodel was capable and robust, but could be difficult to use, as it required that you set up base IP addresses and address ranges (or net masks) for voice and data subnets according to a complex set of rules. In Release 7.0 HDIA was superseded by CopperVPN Plus, which provides the same functionality but without the complexity.

HDSL High-rate Digital Subscriber Line. Originally referred to DSL links at T1 (1.544 Mbps) and E1 (2.048 Mbps) rates. This use of the term is becoming less common, however, as DSL rates have generally increased.

host A processor (computer) that provides services such as database access or computation to other devices. In commands used to configure a CE200, *host* may refer to an external file server or work station, or to the CE200 itself, depending upon the operational organization of the site.

hot insert Capability of inserting or removing a circuit pack/module from a powered up operational system without danger of damage to the DSLAM or to the board itself.

hot swap A module is considered to be hot swappable if it can be removed and replaced into a powered system, and upon insertion, the DSLAM automatically detects the board's presence, initializes it, and configures it for operation without affecting service to or from any connected interface.

I

IAD Integrated Access Device. A special type of CPE that provides dual circuits, accommodating digitized on one circuit and regular data on the other. On the upstream side, both circuits communicate with a CE200 over an SDSL line. On the downstream, for voice, POTS lines interface with the IAD; for data, an Ethernet line interfaces with the IAD.

IDSL DSL service in which the data rates are those used in ISDN services (64, 128, or 144 kbps). IDSL services typically allow longer loops between the CPE and COE, and may allow use of existing ISDN CPE in the DSL system.

IMUX An acronym for *Inverse Multiplexing*, a technique by which data from a single source is separated into multiple streams for transmission over multiple physical links, effectively multiplying the throughput available to an individual subscriber.

interface A point of demarcation (may be physical or logical) between two devices where the electrical signals, connectors, timing and handshaking are defined. It can also refer to the procedures, codes, and protocols that enable a connection and exchange of information between two entities.

IP address Internet Protocol address. A 32-bit address assigned to hosts using TCP-IP. The address is written as four octets that denote the addressee's network, subnet, and host location.

IRB Integrated Router/Bridge. A special operating mode of certain routers that allows the router to support virtual bridge groups, and which thus enables the CE200 Virtual Wide Area Network (VWAN) capability.

ISDN Integrated Services Digital Network. A CCITT standard for a digital network as implemented and delivered by telecommunications common carriers.

L

LAN Local Area Network. A short distance network used to link computers and peripheral devices under standard control. LANs allow users to access centralized servers and anyone on the LAN can send messages to and work jointly with others on the LAN.

LCV Line Coding Violation (a class of DS3 error).

LMI Local Management Interface. A Frame Relay specification related to information exchange between network devices (bridges, routers, etc.).

LOS DS1 and DS3 Loss of signal message.

M

MAC address Media Access Control address. A unique 48-bit character string (address) permanently assigned to an individual device at the time of manufacture.

Mbps Megabits per second, used to denote a data rate. A rate of 1 Mbps is 1×10^6 bits per second.

MIB Management Information Base. A set of variables forming a database contained in an SNMP-managed node on a network. SNMP managers can fetch/store information from/to this database.

MIO Multiplexed Input/Output. A term applied to CE200 network interfaces.

multilink A technique in which data to or from a single endpoint is divided into multiple streams and transported over multiple physical links, effectively multiplying the transmission rate. *See* IMUX.

multi-mode fiber Fiber optic cable which supports multi-mode propagation. Mode refers to the path that light travels over the fiber. In multimode, light is reflected off the sides of the core as it travels through the fiber. Multimode fiber has a typical core diameter of 50 to 100 µm with a graded or stepped refractive index. Multi-mode allows use of inexpensive LED light sources and connector alignment and coupling is less critical than single mode fiber with laser light sources. Distances of transmission and transmission bandwidth are less than with single mode fiber due to dispersion.

N

node A point where connection may be achieved into a network. In practical terms, a node is usually the site of a switch or router.

O

octet An eight-bit data byte. An IP address is comprised of four octets, each ranging in number from 0 to 255.

operational CE200 A CE200 that is powered up and passing data (user or management traffic).

P

PDU Protocol Data Unit. The standard abbreviation used to refer to SNMP protocol packets.

PES DS3 P-bit Errored Seconds. One second of time in which one or more P-bit coding violations, frame defects or incoming alarm signals occur.

PII Permanent Interface Identifier. A character string that provides a means of relating a *physical* interface location to its corresponding network identifiers (IP address, MAC address, etc.).

PIIs are formatted as c.ss.pp.uuuu. Where c is the chassis (shelf) number, ss is the slot number on the CE200 where the interface terminates, pp is the port number of the interface. If a virtual circuit is associated with the interface, uuuu is an identifying number for the virtual circuit, if applicable.

For example, the PII 1.3.2.27 refers to system shelf 1, the WAN card in Slot 3 of that shelf, port number 2, and a virtual circuit with a DLCI or ATM Virtual Link number of 27.

PPP Point-to-Point Protocol.

Preferred side Slots 1 through 5 in a CE200 chassis. It is also called Side A.

Primary complex The side of the CE200 that is controlling operation and traffic; it can be either the Preferred side or Backup side of a CE200 chassis. In cmBoard, Role=Primary.

protocol A set of rules relating to the format and timing of data transmission between devices. In packet-switched networks, protocols break a lengthy message into equal parts called packets. The packets are transmitted across the network, the receiving processor then error-checks them, and returns an acknowledgment to the originating device.

protocol interface While a PII points to and focuses on a physical interface (a port or connector), a protocol interface is a logical name for any interface through which packet date passes: a physical port (DSL, Ethernet, etc.), or a virtual circuit. A protocol interface may take the form of a full PII (*always* including a circuit identifier), or it may simply be the logical name (cmName, IfName) assigned to that interface.

PSES DS3 P-bit Severely Errored Second. One second of time in which 44 or more P-bit coding violations, frame defects or incoming alarm signals occur.

PVC Permanent Virtual Circuit. A Frame Relay term for a configured connection between two devices, usually over a wide-area link. The "permanent" circuit will persist as long as its configuration remains in place. Once a PVC is configured between two devices, no setup or disconnect process is required to initiate or conclude communication over the channel.

R

RADIUS "Remote Authentication Dial-In User Service", a protocol for carrying authentication, authorization, and configuration information between a device that requires authentication service and a shared server on which the authentication data resides.

redundant complex The System Control Module, Buffer Control Module, and WAN modules installed in the Backup (right) side of a CE200.

runt frame Term used to describe an invalid or incomplete Ethernet data frame.

S

SDH Synchronous Digital Hierarchy: A hierarchical set of digital transport structures, standardized for carriage of suitably adapted payloads over physical (primarily optical) networks. The frame structure is very similar to SONET STS, and can be viewed as the European equivalent of SONE

SDSL Symmetrical DSL. DSL service in which data rates are the same in both directions.

Secondary complex The side of the CE200 that is not controlling operation or traffic; it can be either the Preferred side or Backup side of a CE200 chassis. In cmBoard, Role=Secondary.

shelf A single CE200 unit, consisting of a chassis and all of its circuit modules.

single-mode fiber An optical fiber that allows only one mode to propagate (straight-through path, with no reflections off the sides of the core). Single-mode fiber has a very small core diameter (approximately 8 µm). It permits signal transmission at extremely high bandwidth frequencies over very long transmission distances. A laser source is required for signals transported over a single-mode fiber.

SNMP Simple Network Management Protocol. A protocol used to monitor and provision nodes and devices across a network.

SNMP agent Device-based software used to monitor and report status when queried by an SNMP manager. By definition, SNMP agents run on managed devices; an SNMP manager is required to perform active (command and configuration) functions.

SNMP manager System software used to remotely query and configure network devices. A more generic term, also used in this document, is Element Management System.

SONET Synchronous Optical NETwork: A synchronous hierarchy developed by Bellcore. It defines framing structures (STS-n) and optical signaling characteristics for physical layer transmission.

subnet address An extension of the Internet addressing scheme; allows use of a single IP address for multiple physical networks.

subnet mask A 32-bit address mask used in IP to specify a particular subnet.

System Control Module The CE200 circuit module that controls timing, protocol processing and overall system functionality.

T

TCA Threshold Crossing Alert. A type of event/alarm which indicates that a user configured error count or performance limitation has been exceeded. condition which generates an alarm on the CE200. Using the cmAlarmTable which itself uses counters that measure conditions like overruns, underruns, and errors, you can tell the CE200 to indicate when the number of errors becomes unacceptable.

trap An unsolicited messaged generated by an SNMP agent on a network device due to a threshold value being exceeded or the occurrence of a pre-defined event.

TCP/IP Transmission Control Protocol/Internet Protocol. Currently the most widely used layered transmission protocol for connecting dissimilar computers across networks.

TDR Time Domain Reflectometer. A test device that acts on RADAR-like principles to determine the location of metallic faults.

Telco The Incumbent Local Exchange Carrier (ILEC) or the Competitive Local Exchange Carrier (CLEC).

TFTP Trivial file Transfer Protocol. A simplified version of FTP used to transfer files based on UDP (User Datagram Protocol) with no guarantee of delivery. TFTP lacks password protection and certain other capabilities of a full-featured FTP application.

U

UBR Unspecified Bit Rate. Used to describe certain ATM links.

V

VC Virtual Channel. An ATM communications channel; often used to mean any multiplexed packet stream aggregated into a virtual circuit.

VCC Virtual Channel Connection. A concatenation of ATM VCLs. *See* VCL.

VCI Virtual Channel Identifier. Numeric (16-bit) field in an ATM cell header that identifies the virtual channel the cell is to take.

VCL Virtual Channel Link. A discrete end-to-end ATM connection that encompasses both directions of the data exchange. Analogous to the Frame Relay PVC.

VP Virtual Path. A unidirectional ATM path consisting of a logical group or bundle of VCs.

VPI Virtual Path Identifier. A 8-bit field in a cell header designating the virtual path to be taken by the cell.

VPL Virtual Path Link. The full path between the point at which a VPI is assigned to a Cell and the point at which the VPI is translated or removed (i.e, between the starting line and finish line of the path).

VWAN Virtual Wide Area Network. A networking model that makes use of the IRB (Integrated Router/Bridge) capability of compatible routers to allow transparent connectivity of LAN ports over wide area (Frame Relay) links.

W

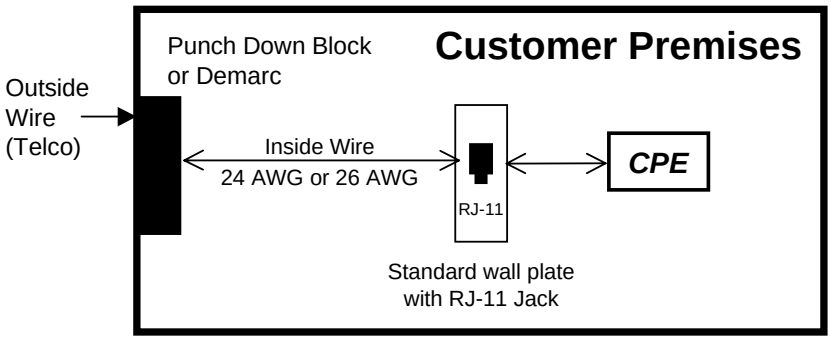
WAN Wide Area Network. WANs cover extended geographical areas in contrast to LANs. WANs generally use links provided by the public switched telephone network to connect distant sites.

Appendix F
CPE Inside Wiring

Inside wiring for Copper Mountain's CopperRocket xDSL family of CPEs consists of a standard, single-pair telephone facility to which the DSL line is connected, and to which the CPE itself connects with a standard 2- or 4-conductor (RJ-11/RJ-14) cable. A diagram is included on the next page for your reference.

If you are using a CPE manufactured by someone other than Copper Mountain Networks, check the manufacturer's documentation for any special requirements.

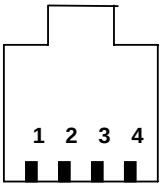
Inside Wiring for xDSL CPE with Copper Mountain Networks COE



Inside Wiring Specifications:

Pinouts

Pin 1 = Yellow
Pin 2 = Green
Pin 3 = Red
Pin 4 = Black



Connect CPE using standard RJ-11/RJ-14 telephone cable included with the CPE.

Only Pins 2 and Pin 3 are used; connection is straight through with no crossover. Unit automatically adjusts for Tip and Ring on these pins.

Recommend inside wire: 24 AWG or 26 AWG standard telephone wire; CAT 3 and CAT 5 UTP can also be used.

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X

COPPER MOUNTAIN

DSL Physical Port Configuration

Perm. Ifc. Index Chassis . Slot . Port			Subscriber IP Address	Account Number	Account Name	Install Date	Status / Notes
		1					
		2					
		3					
		4					
		5					
		6					
		7					
		8					
		9					
		10					
		11					
		12					
		13					
		14					
		15					
		16					
		17					
		18					
		19					
		20					
		21					
		22					
		23					
		24					

COPPER MOUNTAIN

Sheet _____ of _____

WAN Port Configuration Worksheet

Perm. Ifc. Index Shelf . Slot . Port . VC No.				Port IP Address	Subnet Mask	FWD Mode	Net Model	Encaps. Type	CIR	EIR	Interface/Link Notes