



Send documentation comments to mdsfeedback-doc@cisco.com



Cisco MDS 9000 Family SMI-S Programming Reference

March, 2008

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

CCVP, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, LightStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0711R)

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco MDS 9000 Family SMI-S Programming Reference
© 2008 Cisco Systems, Inc. All rights reserved.

Send documentation comments to mdsfeedback-doc@cisco.com



C O N T E N T S

New and Changed Information vii

Preface ix

Audience ix

Organization ix

Document Conventions x

Related Documentation x

 Release Notes xi

 Compatibility Information xi

 Regulatory Compliance and Safety Information xi

 Hardware Installation xi

 Cisco Fabric Manager xii

 Command-Line Interface xii

 Intelligent Storage Networking Services Configuration Guides xii

 Troubleshooting and Reference xii

 Installation and Configuration Note xiii

Obtaining Documentation and Submitting a Service Request xiii

 Cisco.com xiii

 Product Documentation DVD xiii

 Ordering Documentation xiv

Documentation Feedback xiv

Cisco Product Security Overview xiv

 Reporting Security Problems in Cisco Products xv

Obtaining Technical Assistance xvi

 Cisco Technical Support & Documentation Website xvi

 Submitting a Service Request xvi

 Definitions of Service Request Severity xvii

Obtaining Additional Publications and Information xvii

CHAPTER 1

Overview 1-1

 About the Common Information Model 1-1

 About the Storage Management Initiative Specification 1-2

 About the WBEM Initiative 1-3

 Understanding CIM and Unified Modeling Language Notation 1-3

Send documentation comments to mdsfeedback-doc@cisco.com

Understanding CIM Classes 1-3
 Understanding UML 1-4
 About SMI-S and CIM in the Cisco MDS 9000 Family 1-4

CHAPTER 2

Cisco MDS 9000 Family CIM Server Support 2-1
 Managing SANs Through SMI-S 2-1
 Service Location Protocol 2-2
 Server Profile 2-2
 Switch Profile 2-2
 Blade Subprofile 2-3
 Access Point Subprofile 2-4
 Fabric Profile 2-4
 Zone Control Subprofile 2-5
 Enhanced Zoning and Enhanced Zoning Control Subprofile 2-6
 Using the Zoning Subprofile 2-6
 FDMI Subprofile 2-7
 Using the FDMI Subprofile 2-8
 Cisco MDS Extensions to the Switch and Fabric Profiles 2-8
 VSAN Extensions 2-8
 TE Port Extensions 2-10
 PortChannel Extensions 2-12
 FCIP Extensions 2-13
 iSCSI Extensions 2-14
 Fabric Profile Extensions 2-14
 Zoning Subprofile Extensions 2-16
 FDMI Subprofile Extensions 2-16
 CIM Indications 2-16

CHAPTER 3

Configuring and Using the CIM Server 3-1
 Configuring the CIM Server 3-1
 Creating a Certificate Using OpenSSL 3-1
 Installing the Certificate and Enabling the CIM Server 3-2
 Setting CIM Server Loglevel 3-2
 Performing Discovery and Performance Monitoring 3-3
 Modeling a Module Using the Blade Subprofile 3-4
 Configuring Zoning 3-4

Send documentation comments to mdsfeedback-doc@cisco.com

Managed Object Format Files A-1

Cisco MOF Files for Cisco SAN-OS Release 3.0(1) or Later **A-1**

Cisco Fabric MOF **A-1**

Cisco Zone MOF **A-4**

Cisco FDMI MOF **A-5**

Cisco MOF Files for Cisco SAN-OS Release 2.x **A-12**

Cisco Fabric MOF **A-12**

Cisco Zone MOF **A-14**

Cisco Indications MOF **A-15**

INDEX

Send documentation comments to mdsfeedback-doc@cisco.com



New and Changed Information

Table 1 summarizes the new and changed features for the *Cisco MDS 9000 Family SMI-S Programming Reference* and tells you where they are documented. The table includes a brief description of each new feature and the release in which the change occurred.

Table 1 Documented Features for the Cisco MDS 9000 Family SMI-S Programming Reference

Feature	Description	Changed in Release	Where Documented
Zone member addition	Updated the Zone MOF	3.3(1)	Appendix A, “Managed Object Format Files”
FDMI MOF	Added the FDMI MOF	3.3(1)	Appendix A, “Managed Object Format Files”
New Indications	Added new indications	3.3(1)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
FDMI Subprofile	Added FDMI Subprofile	3.3(1)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
FDMI Subprofile Extensions	Added FDMI Subprofile extensions	3.3(1)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
SMI-S 1.2.0 compliance	Added support for compliance with SMI-S 1.2.0 with caveats.	3.3(1)	Chapter 1, “Overview”
SMI-S 1.1.0 compliance	Added support for full compliance with SMI-S 1.1.0.	3.0(1)	Chapter 1, “Overview”
Access point subprofile	Added support for supplying the URL to the switch.	3.0(1)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
Server profile	Added support for announcing services and capabilities using the Server profile.	2.0(1b)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
SLP	Added support for services discovery using Service Location Protocol (SLP).	2.0(1b)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”
CIM indications	Added support for CIM asynchronous event notifications.	2.0(1b)	Chapter 2, “Cisco MDS 9000 Family CIM Server Support”

Send documentation comments to mdsfeedback-doc@cisco.com

Table 1 Documented Features for the Cisco MDS 9000 Family SMI-S Programming Reference

Feature	Description	Changed in Release	Where Documented
CIM configuration	Added steps to configure the CIM server.	1.3(1)	Chapter 3, “Configuring and Using the CIM Server”
CIM support	Introduction of CIM support for the Cisco MDS 9000 Family.	1.3(1)	This guide



Preface

This preface describes the audience, organization, and conventions of the *Cisco MDS 9000 Family SMI-S Programming Reference*. It also provides information on how to obtain related documentation.

Audience

To use this programming guide, you must be familiar with general object-oriented programming techniques and the following items:

- Storage Management Initiative Specification (SMI-S)
- Common Information Model (CIM)
- Managed Object Format (MOF) files
- Unified Modeling Language (UML)
- Secure Socket Layer (SSL), if increased security is desired when accessing the CIM server

Organization

This guide is organized as follows:

Chapter	Title	Description
Chapter 1	Overview	Provides an overview of the support provided for CIM and other standards.
Chapter 2	Cisco MDS 9000 Family CIM Server Support	Describes the supported profiles, indications, and Cisco-specific extensions.
Chapter 3	Configuring and Using the CIM Server	Provides CLI commands to configure the CIM server, and sample scenarios for using CIM to manage your SAN.
Appendix A	Managed Object Format Files	Provides the text from the MOF files for the Cisco MDS 9000 Family CIM server extensions.



Document Conventions

Command descriptions use these conventions:

boldface font	Commands and keywords are in boldface.
<i>italic font</i>	Arguments for which you supply values are in italics.
[]	Elements in square brackets are optional.
[x y z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Screen examples use these conventions:

screen font	Terminal sessions and information the switch displays are in screen font.
boldface screen font	Information you must enter is in boldface screen font.
<i>italic screen font</i>	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following conventions:



Note

Means reader *take note*. Notes contain helpful suggestions or references to material not covered in the manual.



Caution

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

Related Documentation

The documentation set for the Cisco MDS 9000 Family includes the following documents. To find a document online, use the Cisco MDS SAN-OS Documentation Locator at:
http://www.cisco.com/en/US/docs/storage/san_switches/mds9000/roadmaps/doclocator.htm

Send documentation comments to mdsfeedback-doc@cisco.com



For information on IBM TotalStorage SAN Volume Controller Storage Software for the Cisco MDS 9000 Family, refer to the IBM TotalStorage Support website:
<http://www.ibm.com/storage/support/2062-2300/>

Release Notes

- *Cisco MDS 9000 Family Release Notes for Cisco MDS SAN-OS Releases*
- *Cisco MDS 9000 Family Release Notes for Storage Services Interface Images*
- *Cisco MDS 9000 Family Release Notes for Cisco MDS SVC Releases*
- *Cisco MDS 9000 Family Release Notes for Cisco MDS 9000 EPLD Images*

Compatibility Information

- *Cisco MDS 9000 SAN-OS Hardware and Software Compatibility Information*
- *Cisco MDS 9000 Family Interoperability Support Matrix*
- *Cisco MDS SAN-OS Release Compatibility Matrix for IBM SAN Volume Controller Software for Cisco MDS 9000*
- *Cisco MDS SAN-OS Release Compatibility Matrix for Storage Service Interface Images*

Regulatory Compliance and Safety Information

- *Regulatory Compliance and Safety Information for the Cisco MDS 9000 Family*

Hardware Installation

- *Cisco MDS 9500 Series Hardware Installation Guide*
- *Cisco MDS 9200 Series Hardware Installation Guide*

Send documentation comments to mdsfeedback-doc@cisco.com



- *Cisco MDS 9216 Switch Hardware Installation Guide*
- *Cisco MDS 9100 Series Hardware Installation Guide*
- *Cisco MDS 9020 Fabric Switch Hardware Installation Guide*

Cisco Fabric Manager

- *Cisco MDS 9000 Family Fabric Manager Quick Configuration Guide*
- *Cisco MDS 9000 Family Fabric Manager Configuration Guide*
- *Cisco MDS 9000 Fabric Manager Online Help*
- *Cisco MDS 9000 Fabric Manager Web Services Online Help*

Command-Line Interface

- *Cisco MDS 9000 Family Software Upgrade and Downgrade Guide*
- *Cisco MDS 9000 Family CLI Quick Configuration Guide*
- *Cisco MDS 9000 Family CLI Configuration Guide*
- *Cisco MDS 9000 Family Command Reference*
- *Cisco MDS 9000 Family Quick Command Reference*
- *Cisco MDS 9020 Fabric Switch Configuration Guide and Command Reference*
- *Cisco MDS 9000 Family SAN Volume Controller Configuration Guide*

Intelligent Storage Networking Services Configuration Guides

- *Cisco MDS 9000 Family Data Mobility Manager Configuration Guide*
- *Cisco MDS 9000 Family Storage Media Encryption Configuration Guide*
- *Cisco MDS 9000 Family Secure Erase Configuration Guide - For Cisco MDS 9500 and 9200 Series*

Troubleshooting and Reference

- *Cisco MDS 9000 Family Troubleshooting Guide*
- *Cisco MDS 9000 Family MIB Quick Reference*
- *Cisco MDS 9020 Fabric Switch MIB Quick Reference*
- *Cisco MDS 9000 Family SMI-S Programming Reference*

Send documentation comments to mdsfeedback-doc@cisco.com



- *Cisco MDS 9000 Family System Messages Reference*
- *Cisco MDS 9020 Fabric Switch System Messages Reference*

Installation and Configuration Note

- *Cisco MDS 9000 Family SSM Configuration Note*
- *Cisco MDS 9000 Family Port Analyzer Adapter Installation and Configuration Note*

Obtaining Documentation and Submitting a Service Request

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

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

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

Cisco.com

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

<http://www.cisco.com/techsupport>

You can access the Cisco website at this URL:

<http://www.cisco.com>

You can access international Cisco websites at this URL:

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

Product Documentation DVD

The Product Documentation DVD is a comprehensive library of technical product documentation on a portable medium. The DVD enables you to access multiple versions of installation, configuration, and command guides for Cisco hardware and software products. With the DVD, you have access to the same HTML documentation that is found on the Cisco website without being connected to the Internet. Certain products also have .PDF versions of the documentation available.

Send documentation comments to mdsfeedback-doc@cisco.com



The Product Documentation DVD is available as a single unit or as a subscription. Registered Cisco.com users (Cisco direct customers) can order a Product Documentation DVD (product number DOC-DOCDVD= or DOC-DOCDVD=SUB) from Cisco Marketplace at this URL:

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

Ordering Documentation

Registered Cisco.com users may order Cisco documentation at the Product Documentation Store in the Cisco Marketplace at this URL:

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

Nonregistered Cisco.com users can order technical documentation from 8:00 a.m. to 5:00 p.m. (0800 to 1700) PDT by calling 1 866 463-3487 in the United States and Canada, or elsewhere by calling 011 408 519-5055. You can also order documentation by e-mail at tech-doc-store-mkpl@external.cisco.com or by fax at 1 408 519-5001 in the United States and Canada, or elsewhere at 011 408 519-5001.

Documentation Feedback

You can rate and provide feedback about Cisco technical documents by completing the online feedback form that appears with the technical documents on Cisco.com.

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

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

We appreciate your comments.

Cisco Product Security Overview

Cisco provides a free online Security Vulnerability Policy portal at this URL:

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

From this site, you will find information about how to:

- Report security vulnerabilities in Cisco products.
- Obtain assistance with security incidents that involve Cisco products.
- Register to receive security information from Cisco.

Send documentation comments to mdsfeedback-doc@cisco.com



A current list of security advisories, security notices, and security responses for Cisco products is available at this URL:

<http://www.cisco.com/go/psirt>

To see security advisories, security notices, and security responses as they are updated in real time, you can subscribe to the Product Security Incident Response Team Really Simple Syndication (PSIRT RSS) feed. Information about how to subscribe to the PSIRT RSS feed is found at this URL:

http://www.cisco.com/en/US/products/products_psirt_rss_feed.html

Reporting Security Problems in Cisco Products

Cisco is committed to delivering secure products. We test our products internally before we release them, and we strive to correct all vulnerabilities quickly. If you think that you have identified a vulnerability in a Cisco product, contact PSIRT:

- For Emergencies only—security-alert@cisco.com

An emergency is either a condition in which a system is under active attack or a condition for which a severe and urgent security vulnerability should be reported. All other conditions are considered nonemergencies.

- For Nonemergencies—psirt@cisco.com

In an emergency, you can also reach PSIRT by telephone:

- 1 877 228-7302
- 1 408 525-6532



Tip

We encourage you to use Pretty Good Privacy (PGP) or a compatible product (for example, GnuPG) to encrypt any sensitive information that you send to Cisco. PSIRT can work with information that has been encrypted with PGP versions 2.x through 9.x.

Never use a revoked or an expired encryption key. The correct public key to use in your correspondence with PSIRT is the one linked in the Contact Summary section of the Security Vulnerability Policy page at this URL:

http://www.cisco.com/en/US/products/products_security_vulnerability_policy.html

The link on this page has the current PGP key ID in use.

If you do not have or use PGP, contact PSIRT at the aforementioned e-mail addresses or phone numbers before sending any sensitive material to find other means of encrypting the data.

Send documentation comments to mdsfeedback-doc@cisco.com



Obtaining Technical Assistance

Cisco Technical Support provides 24-hour-a-day award-winning technical assistance. The Cisco Technical Support & Documentation website on Cisco.com features extensive online support resources. In addition, if you have a valid Cisco service contract, Cisco Technical Assistance Center (TAC) engineers provide telephone support. If you do not have a valid Cisco service contract, contact your reseller.

Cisco Technical Support & Documentation Website

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

<http://www.cisco.com/techsupport>

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

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



Note

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

Submitting a Service Request

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

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

Send documentation comments to mdsfeedback-doc@cisco.com



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

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

Asia-Pacific: +61 2 8446 7411 (Australia: 1 800 805 227)

EMEA: +32 2 704 55 55

USA: 1 800 553-2447

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

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

Definitions of Service Request Severity

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

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

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

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

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

Obtaining Additional Publications and Information

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

- The *Cisco Product Quick Reference Guide* is a handy, compact reference tool that includes brief product overviews, key features, sample part numbers, and abbreviated technical specifications for many Cisco products that are sold through channel partners. It is updated twice a year and includes the latest Cisco offerings. To order and find out more about the Cisco Product Quick Reference Guide, go to this URL:

<http://www.cisco.com/go/guide>

Send documentation comments to mdsfeedback-doc@cisco.com



- Cisco Marketplace provides a variety of Cisco books, reference guides, documentation, and logo merchandise. Visit Cisco Marketplace, the company store, at this URL:
<http://www.cisco.com/go/marketplace/>
- *Cisco Press* publishes a wide range of general networking, training and certification titles. Both new and experienced users will benefit from these publications. For current Cisco Press titles and other information, go to Cisco Press at this URL:
<http://www.ciscopress.com>
- *Packet* magazine is the Cisco Systems technical user magazine for maximizing Internet and networking investments. Each quarter, Packet delivers coverage of the latest industry trends, technology breakthroughs, and Cisco products and solutions, as well as network deployment and troubleshooting tips, configuration examples, customer case studies, certification and training information, and links to scores of in-depth online resources. You can access Packet magazine at this URL:
<http://www.cisco.com/packet>
- *iQ Magazine* is the quarterly publication from Cisco Systems designed to help growing companies learn how they can use technology to increase revenue, streamline their business, and expand services. The publication identifies the challenges facing these companies and the technologies to help solve them, using real-world case studies and business strategies to help readers make sound technology investment decisions. You can access iQ Magazine at this URL:
<http://www.cisco.com/go/iqmagazine>
or view the digital edition at this URL:
<http://ciscoiq.texterity.com/ciscoiq/sample/>
- *Internet Protocol Journal* is a quarterly journal published by Cisco Systems for engineering professionals involved in designing, developing, and operating public and private internets and intranets. You can access the Internet Protocol Journal at this URL:
<http://www.cisco.com/ipj>
- Networking products offered by Cisco Systems, as well as customer support services, can be obtained at this URL:
<http://www.cisco.com/en/US/products/index.html>
- Networking Professionals Connection is an interactive website for networking professionals to share questions, suggestions, and information about networking products and technologies with Cisco experts and other networking professionals. Join a discussion at this URL:
<http://www.cisco.com/discuss/networking>
- World-class networking training is available from Cisco. You can view current offerings at this URL:
<http://www.cisco.com/en/US/learning/index.html>



CHAPTER 1

Overview

The Cisco MDS 9000 Family of multilayer directors and fabric switches provide an industry standard application programming interface (API) using the Storage Management Initiative Specification (SMI-S). SMI-S facilitates managing storage area networks (SANs) in a multivendor environment.

This chapter includes the following sections:

- [About the Common Information Model, page 1-1](#)
- [Understanding CIM and Unified Modeling Language Notation, page 1-3](#)
- [About SMI-S and CIM in the Cisco MDS 9000 Family, page 1-4](#)

About the Common Information Model

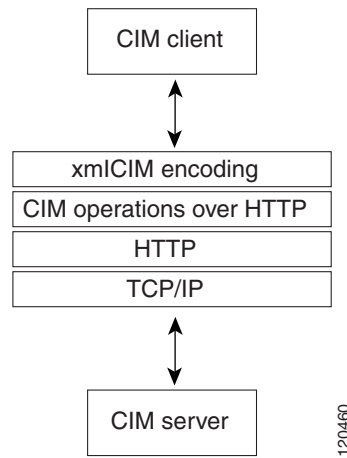
The Common Information Model (CIM) is an object-oriented information model that describes management information in a network or enterprise environment. Because it is object-oriented, CIM provides abstraction, inheritance, and dependency or association relationships between objects within the model. CIM is based on XML and is platform-independent and technology neutral. The management application developer does not need any information about how CIM was implemented on a vendor product; only the API is required to interact with a vendor product.

CIM uses a client/server model. The CIM server can be embedded into the vendor product or can be implemented by a proxy server that provides the CIM server functionality for the legacy vendor product. The CIM client is the management application that communicates to multiple CIM servers to manage the SAN. The CIM client discovers CIM servers through the Service Location Protocol, version 2 (SLPv2) as defined in RFC 2608. SLPv2 uses UDP port 427 for communication and is a discovery protocol that is separate from the CIM client/server communication path.

CIM defines the communications between the client and server in terms of technologies defined in the Web-Based Enterprise Management Initiative (WBEM). [Figure 1-1](#) shows the full CIM client/server communications path.

Send documentation comments to mdsfeedback-doc@cisco.com

Figure 1-1 CIM Client/Server Communications

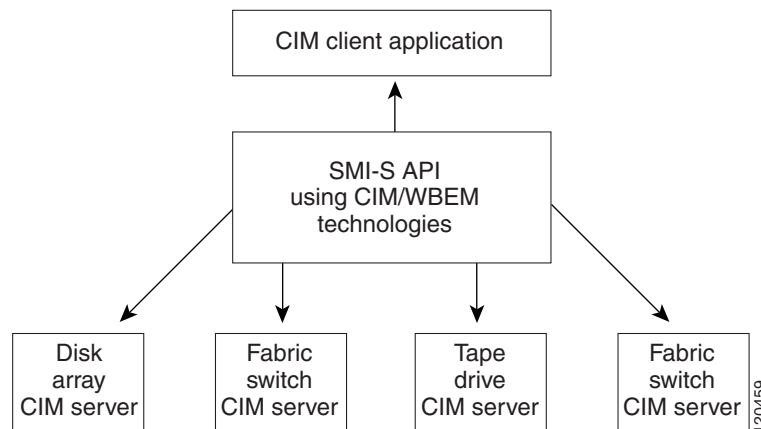


For more information about CIM, refer to the specification available through the Distributed Management Task Force (DMTF) website at <http://www.dmtf.org>.

About the Storage Management Initiative Specification

The Storage Management Initiative Specification (SMI-S) uses an object-oriented model based on CIM to define a set of objects and services that can manage elements of a SAN. By using a standardized architecture, SMI-S helps management application developers create common and extensible applications that work across multiple SAN vendor products. Figure 1-2 exemplifies SMI-S in a multivendor SAN.

Figure 1-2 SMI-S in a Multivendor SAN



SMI-S provides a set of standard management objects collected in a *profile*. Several profiles are defined in SMI-S that cover common SAN elements, including switches, fabrics, and zoning. These standardized profiles insure interoperability across products within the SAN. SMI-S also defines an automated discovery process, using SLPv2. SMI-S uses CIM defined by the DMTF as part of the WBEM.

For more information about SMI-S, refer to the Storage Networking Industry Association (SNIA) website at <http://www.snia.org>.

Send documentation comments to mdsfeedback-doc@cisco.com



Note

Cisco SAN-OS Release 3.0(1) is compliant with SMI-S 1.1.0.

About the WBEM Initiative

The WBEM initiative is a set of management and Internet standards developed to unify the management of enterprise computing environments.

The WBEM initiative includes:

- CIM, which provides a common format, language and methodology for collecting and describing management data.
- The CIM-XML Encoding Specification, a standards-based method for exchanging CIM information. CIM-XML uses an xmlCIM encoded payload and HTTP as the transport mechanism. CIM-XML consists of the following specifications:
 - xmlCIM encoding, a standard way to represent CIM information in XML format.
 - CIM operations over HTTP, a transportation method that describes how to pass xmlCIM encoded messages over HTTP.

For more information about the WBEM initiative, refer to the DMTF website at <http://www.dmtf.org>

Understanding CIM and Unified Modeling Language Notation

SMI-S relies on object-oriented classes as defined in CIM. These classes are frequently defined using Unified Modeling Language (UML). To understand the SMI-S and the Cisco extensions present in this document, you must have a basic understanding of CIM classes and UML.

Understanding CIM Classes

A class is a collection of properties and methods that define a type of object. As an example, a generic network device is a type of object. We could define the `NetworkDevice` class to describe this object. The `NetworkDevice` class contains properties or attributes of a network device. Some properties for our `NetworkDevice` class are `IpAddress` and `DeviceType`. Further, we want to control our network device through the `NetworkDevice` class. So we add methods or routines we can use to trigger actions on our network device. Some methods are `enablePort()` and `rebootDevice()`.

Now that we have a `NetworkDevice` class, we can define a class for just switches. Because a switch is a special type of `NetworkDevice`, we use the object-oriented concept of *inheritance* to define our `Switch` class. We define the `Switch` class as a child of the `NetworkDevice` class. This means the `Switch` class automatically has the properties and methods of its parent class. From there, we add properties and methods unique to a switch.

CIM defines a special type of class called an *association class*. An association class represents relationships between two or more classes. As an example, we define an association class to show the relationship between a `NetworkDevice` class and an `OperatingSystem` class. If there is a many-to-one or many-to-many relationship, the association class is considered an *aggregation*.

Refer to <http://www.dmtf.org> for a full explanation of CIM.

Send documentation comments to mdsfeedback-doc@cisco.com

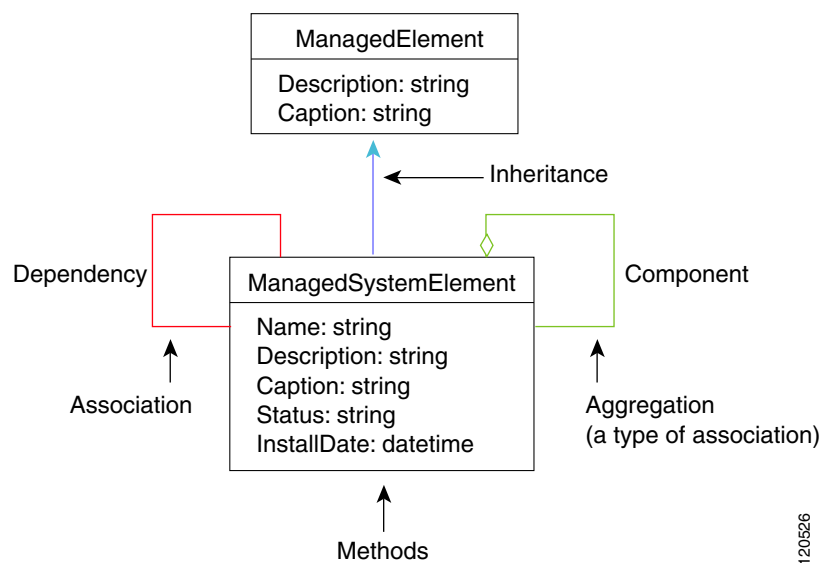
Understanding UML

UML draws a visual representation of the classes that describe a product or technology. UML contains many visual elements, but only a subset are described here. Refer to <http://www.uml.org> for a full explanation of UML.

Figure 1-3 shows an example section from a UML diagram for CIM classes. This diagram shows:

- blue lines for inheritance between classes
- green lines for aggregation between classes
- red lines for associations between classes

Figure 1-3 UML Example Diagram



About SMI-S and CIM in the Cisco MDS 9000 Family

Each switch or director in the Cisco MDS 9000 Family includes an embedded CIM server. The CIM server communicates with any CIM client to provide SAN management compatible with SMI-S. The CIM server includes the following standard profiles, subprofiles, and features as defined in SMI-S:

- Service Location Protocol version 2 (SLPv2).
- Server profile.
- CIM indications.
- Fabric profile.
 - Zoning Control subprofile.
 - Enhanced Zoning and Enhanced Zoning Control subprofile.
 - FDMI subprofile.
- Switch profile, including the Blade subprofile and Access Point subprofile.
- xmlCIM encoding and CIM operations over HTTP as specified by the WBEM initiative.

Send documentation comments to mdsfeedback-doc@cisco.com

- HTTPS, which uses Secure Socket Layer (SSL). HTTPS is optional but provides enhanced security by encrypting communications between the CIM server and the CIM client.

Table 1-1 shows the Cisco SAN-OS release that supports different versions of SMI-S.

Table 1-1 Cisco SAN-OS Support for SMI-S

Cisco SAN-OS Release	SMI-S Support	Description
3.3(1)	SMI-S 1.2.0 compliant with caveats	<ul style="list-style-type: none">• All required indications are not supported• limited WQL/CQL support
3.0(1)	SMI-S 1.1.0 compliant	Additional support for Server profile and Access Port subprofile.
2.0(1b)	Supports SMI-S 1.0.2	Supports SLPv2, CIM indications, and the Server profile.

Send documentation comments to mdsfeedback-doc@cisco.com



CHAPTER 2

Cisco MDS 9000 Family CIM Server Support

SMI-S defines a number of profiles that specify the managed objects used to control and monitor elements of a SAN. The Cisco MDS 9000 Family CIM server supports the standard profiles listed in this chapter. The CIM server also supports extensions to these profiles to support features in Cisco MDS SAN-OS that are not available from the standard profiles.

This chapter includes the following sections:

- [Managing SANs Through SMI-S, page 2-1](#)
- [Service Location Protocol, page 2-2](#)
- [Server Profile, page 2-2](#)
- [Switch Profile, page 2-2](#)
- [Fabric Profile, page 2-4](#)
- [Cisco MDS Extensions to the Switch and Fabric Profiles, page 2-8](#)
- [CIM Indications, page 2-16](#)

Managing SANs Through SMI-S

SANs are created in a multivendor environment. Hosts, fabric elements (switches, directors), and data storage devices are integrated from different vendors to create an interoperable storage network. Managing these elements from different vendors is problematic to the network administrator. Each element has its own management interface that may be proprietary. A network administrator must work with these disparate management APIs to build a cohesive management application that controls and monitors the SAN.

The SMI-S addresses this management problem by creating a suite of flexible, open management API standards based on the vendor- and technology-independent CIM. Using the SMI-S APIs, collected in *profiles* of common management classes, a network administrator can create a simplified management application CIM client to control and monitor the disparate SAN elements that support SMI-S and CIM. With CIM servers either embedded on the SAN elements or supported by a proxy CIM server, these elements are accessible to the network administrator's CIM client application.

SMI-S uses the Service Location Protocol version 2 (SLPv2) to discover CIM servers. Once the CIM servers are identified, the CIM client determines the profiles supported on the CIM servers through the Server profile. This profile is mandatory on all SMI-S based CIM servers.

Besides the control and monitoring support provided by profiles, the CIM server also supports asynchronous delivery of events through CIM *indications*. Indications provide immediate notification of important occurrences such as when an interface goes down.

Send documentation comments to mdsfeedback-doc@cisco.com

Service Location Protocol

The first step in managing a network of SAN elements with CIM servers is discovering the location and support available on the CIM servers. The SLPv2 provides this discovery mechanism. A CIM client uses SLPv2 to discover CIM servers, gathering generic information about what services the CIM servers provide and the URL where these services are located.

The Cisco MDS 9000 Family CIM server supports SLPv2 as defined in RFC 2608.



Note

Cisco MDS SAN-OS Release 2.0(1b) and later support SLPv2 for the Cisco MDS 9000 Family CIM server.

Server Profile

Once the CIM client discovers the CIM servers within the SAN, the CIM client must determine the level of support each CIM server provides. The Server profile defines the capabilities of the CIM server. This includes providing the namespace and all profiles and subprofiles supported by the CIM server.

For each supported profile, the Server profile instantiates the `RegisteredProfile` class. Each instance of this class gives the CIM client the profile name and unique ID that is supported by the CIM server. Similarly, the CIM server lists all supported optional subprofiles, using the `RegisteredSubProfile` class and the `SubprofileRequiresProfile` association class to associate the subprofile with the profile.



Note

Cisco MDS SAN-OS Release 2.0(1b) and later support the Server profile for the Cisco MDS 9000 Family CIM server.



Note

For a Server profile instance diagram, refer to SMI-S at <http://www.snia.org>.

Switch Profile

The Switch profile models the physical and logical aspects of switches. The CIM client uses the Switch profile to identify that the CIM server is on a switch and uses classes in the Switch profile to identify and manage Fibre Channel ports on the switch.

The Switch profile also supports the optional Blade subprofile (see the “Blade Subprofile” section on page 2-3) and the optional Access Point subprofile (see the “Access Point Subprofile” section on page 2-4).



Note

For a Switch profile instance diagram, refer to the SMI-S at <http://www.snia.org>.

Table 2-1 shows how to use the Switch profile classes and association classes to model the switch and ports.

Send documentation comments to mdsfeedback-doc@cisco.com

Table 2-1 **Using the Switch Profile**

Class	How Used
ComputerSystem	Identifies the switch, with the <code>Dedicated</code> property set to <code>Switch</code> .
PhysicalElement	Identifies the physical aspects of a device.
FCPort	Identifies logical aspects of the port link and the data layers.
FCPortCapabilities	Defines configuration options supported by the ports.
FCPortStatistics	Identifies port statistics, showing real-time port traffic information for each instance of <code>FCPort</code> class.
FCSwitchCapabilities	Defines configuration options supported by the switch.
FCSwtichSettings	Requests configuration changes on the switch.
FCPortSettings	Requests configuration changes on the ports.

Blade Subprofile

The CIM client uses the optional Blade subprofile to model the physical and logical aspects of a supervisor module, switching module, or services module in a switch. Combining this with the Switch profile, the CIM client gains a chassis-level view into the switch, associating ports to modules and modules to a switch.

[Table 2-2](#) shows how to use the classes and association classes to model a module.

Table 2-2 **Using the Blade Subprofile**

Class	How Used
LogicalModule	Identifies a supervisor module, switching module, or services module as an aggregation point for the switch ports.
ModulePort	Associates the ports to a module.



Note

For a Blade subprofile instance diagram, refer to the SMI-S at <http://www.snia.org>.

Send documentation comments to mdsfeedback-doc@cisco.com

Access Point Subprofile

The CIM client uses the Access Point subprofile to return the URL to access the switch and install or launch Fabric Manager or Device Manager. If Fabric Manager or Device Manager have not been installed, then the URL gives the option to install them. If Fabric Manager or Device Manager have been installed, then the URL gives the option to launch either of them.

Table 2-3 shows how to use the classes and association classes to model a module.

Table 2-3 Using the Access Point Subprofile

Class	How Used
HostedAccessPoint	Associates the <code>RemoteServiceAccessPoint</code> to the system on which it is hosted.
RemoteServiceAccessPoint	A <code>ServiceAccessPoint</code> for management tools. Returns the URL for the switch that can be used to install or launch Fabric Manager or Device Manager.
SAPAvailableForElement	Identifies the subset of devices in the system that are serviced by <code>RemoteServiceAccessPoint</code> .



Note

For an Access Point subprofile instance diagram, refer to the SMI-S at <http://www.snia.org>.

Fabric Profile

A fabric is composed of one or more switches and network elements interconnected in a SAN. The Fabric profile models the physical and logical aspects of the fabric containing the SAN switches listed by the Switch profile.

Fabrics can contain one or more virtual SANs, or VSANs. See the “[Cisco MDS Extensions to the Switch and Fabric Profiles](#)” section on page 2-8 for more information on the Cisco VSAN extension. Because routing in the Cisco MDS 9000 Family is based on the VSAN, the `ConnectivityCollection` and `ProtocolEndpoint` classes must be associated with the VSAN, not the fabric.

Table 2-4 shows how to use the classes and association classes of the Fabric profile to model the fabric.

Table 2-4 Using the Fabric Profile

Class	How Used
AdminDomain	Identifies fabrics and VSANs.
ContainedDomain	Associates a VSAN to a fabric.
ConnectivityCollection	Groups a set of <code>ProtocolEndpoint</code> classes together that can communicate with each other directly and represents the foundation necessary for routing. Associates to a VSAN using the <code>Component</code> association class.
ComputerSystem	Represents the fabric elements that contain ports, such as switches, hosts, and storage systems. The <code>Dedicated</code> property is set to <code>Switch</code> . Associates to a VSAN using the <code>Component</code> association class.

Send documentation comments to mdsfeedback-doc@cisco.com

Table 2-4 Using the Fabric Profile (continued)

Class	How Used
FCPort	Represents the logical aspects of the link and data layers. Associates to the <code>ProtocolEndpoint</code> class by the <code>DeviceSAPImplementation</code> association class and associates to the <code>ComputerSystem</code> class by the <code>SystemDevice</code> association class.
<code>ProtocolEndpoint</code>	Represents the higher network layers for routing. Associates to the <code>ConnectivityCollection</code> class by the <code>ConnectivityMemberOfCollection</code> association class.
<code>ActiveConnection</code>	Represents a link that associates two <code>ProtocolEndpoint</code> classes as a connection that is currently carrying traffic.



Note

The Cisco MDS 9000 Family CIM server only provides the `LogicalPortGroup` class for the fabric, not for hosts or storage systems.

The CIM server requires that the name of the fabric or VSAN be unique within the same CIM namespace. Names are identified by the `Name` class property with an associated optional `NameFormat` property. A VSAN identifier is the VSAN identification followed by the world-wide name (WWN) of the principal switch, for example—“1_2001000530000A0A” (the `NameFormat` indicates that it is a WWN). For VSANs, the fabric identifier is a string because there is no principal switch per fabric.



Note

For a Fabric profile instance diagram, refer to the SMI-S at <http://www.snia.org>.

The CIM server supports the following optional subprofiles from the Fabric profile:

- Zone Control subprofile (see the “Zone Control Subprofile” section on page 2-5)
- Enhanced Zoning and Enhanced Zoning Control subprofile (see the “Enhanced Zoning and Enhanced Zoning Control Subprofile” section on page 2-6)
- FDMI subprofile (see the “FDMI Subprofile” section on page 2-7)

The Fabric profile also supports a number of extensions specific to the Cisco MDS 9000 Family. See the “Cisco MDS Extensions to the Switch and Fabric Profiles” section on page 2-8.

Zone Control Subprofile

Zoning enables the CIM client to set up access control between storage devices or user groups. The Zone Control subprofile is a subprofile of the Fabric profile and models zoning information for the fabric. It incorporates read and write functionality including the following operations:

- Creating and deleting zones and zone sets
- Creating and deleting zone members (using `ZoneMembershipSettingData`)
- Adding and removing zone members to zones
- Adding and removing zones to zone sets
- Activating and deactivating a zone set

Send documentation comments to mdsfeedback-doc@cisco.com

The CIM server supports all the CIM classes and association classes described by the SMI-S zoning model.

Enhanced Zoning and Enhanced Zoning Control Subprofile

The Enhanced Zoning and Enhanced Zoning Control subprofile is a subprofile of the Fabric profile and provides additional modeling of Cisco zoning information for management purposes. This includes support for the following:

- Creating and deleting zone aliases
- Adding and removing zone members to zone aliases

This subprofile supports all CIM classes and association classes described by the SMI-S zoning model except the concept of sessions for zoning.

Using the Zoning Subprofile

In the Cisco MDS CIM implementation, zoning occurs under the VSAN, not the fabric.



Note

For Zoning subprofile instance diagrams, refer to the SMI-S at <http://www.snia.org>.

Table 2-5 shows how to use the classes and association classes of the Zoning subprofiles to model zoning.

Table 2-5 Using the Zoning Subprofile

Class	How Used
<code>ZoneMembershipSettingData</code>	Identifies zone members and indicates the member ID (defined in the CIM schema) and how the device was zoned.
<code>ZoneAlias</code>	Identifies zone aliases. Contains zone members (<code>ZoneMembershipSettingData</code> class) associated by the <code>ElementSettingData</code> association class.
<code>ZoneSets</code>	Identifies zone sets. Contains zones associated by the <code>MemberOfCollection</code> association class.
<code>AdminDomain</code>	Identifies VSANs. Only contains zone sets that are associated by the <code>HostedCollection</code> association class.
<code>ZoneControl</code>	Provides operations to control zone objects, such as creating, removing, and activating both zones and zone sets.
<code>ZoneService</code>	Manages the creation of zone sets, zones, zone aliases, and zone members, as well as activation of the zone set. The <code>ZoneService</code> class is hosted on the <code>CISCO_Vsan</code> class, which is a subclass of <code>AdminDomain</code> .
<code>ActiveConnection</code>	Represents a link that associates two <code>ProtocolEndpoint</code> classes as a connection that is currently carrying traffic.

Zones and zone sets that are active have the `Active` property set to `True` by the CIM server. Zones can only contain the following types of objects:

Send documentation comments to mdsfeedback-doc@cisco.com

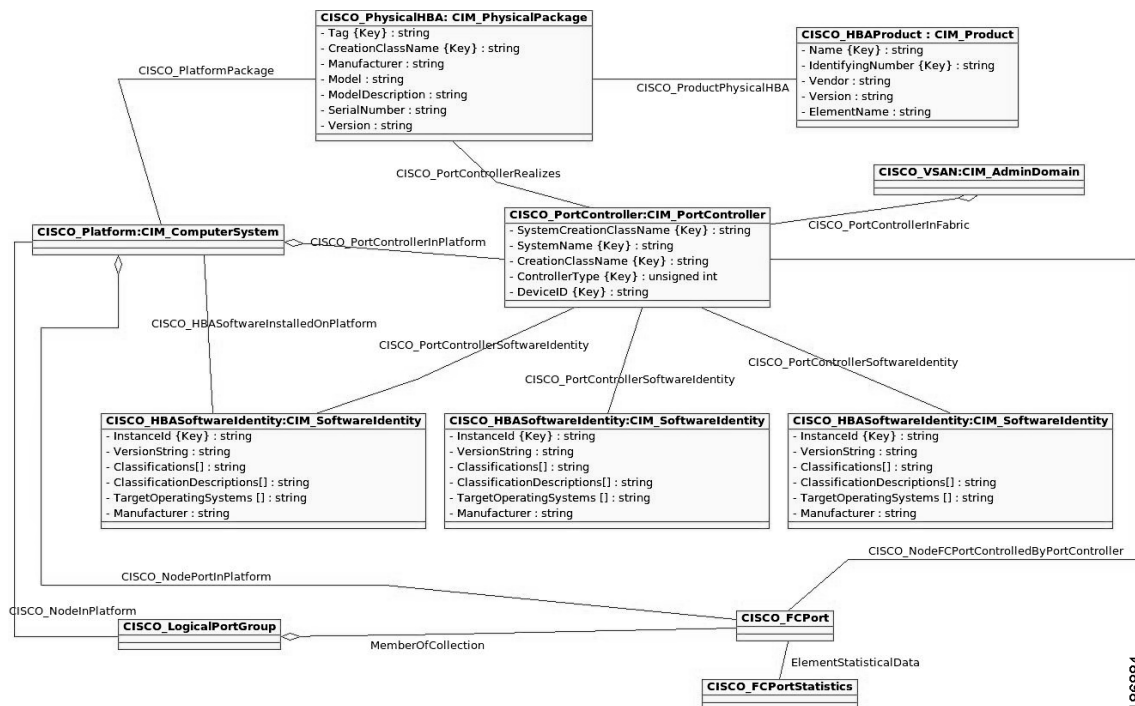
- Zone members (ZoneMembershipSettingData class) associated by the ElementSettingData association class.
- Zone aliases (ZoneAlias class; defined by SMI-S as NamedAddressCollections class) associated by the MemberOfCollection association class.

FDMI Subprofile

The Fabric Device Management Interface (FDMI) manages host bus adapters (HBA) through the Fabric and complements data in the fabric profile. It allows any entity in the fabric to expose the HBA information through the SMI without having an agent resident on the host containing the HBA. The fabric profile only addresses HBA type devices. The HBA Management Interface defined by FDMI is a subset of the interface defined by the Fibre Channel HBA API specification.

Figure 2-1 shows the FDMI subprofile instance diagram. The classes are defined in CISCO_HBA.mof. If the FDMI-enabled HBA supports the host name, then CISCO_PortController associates to a platform through CISCO_PortControllerInPlatform. If the FDMI-enabled HBA does not support the host name, then CISCO_PortController associates to a fabric, through CISCO_PortControllerInFabric.

Figure 2-1 UML Diagram for FDMI Subprofile



Send documentation comments to mdsfeedback-doc@cisco.com

Using the FDMI Subprofile

In the Cisco MDS CIM implementation, the FDMI subprofile occurs under the fabric.



Note

For FDMI subprofile instance diagrams, refer to the SMI-S at <http://www.snia.org>.

Table 2-6 shows how to use the classes and association classes of the FDMI subprofile.

Table 2-6 Using the FDMI Subprofile

Class	How Used
CISCO_PhysicalHBA	Represents FDMI enabled physical HBA card attached to a switch.
CISCO_HBAProduct	Represents product information of FDMI enabled physical HBA card attached to a switch.
CISCO_Platform	Represents a fabric-connected entity, containing one or more Node objects, that has registered with a fabric's Management Server service.
PortController	Represents the Port Controller of a FDMI enabled HBA.

Cisco MDS Extensions to the Switch and Fabric Profiles

The Cisco MDS 9000 Family CIM server supports additional classes that provide management for SAN features not covered by the standard SMI-S profiles. These extensions include:

- [Figure 2-1 VSAN Extensions, page 2-8](#)
- [TE Port Extensions, page 2-10](#)
- [PortChannel Extensions, page 2-12](#)
- [FCIP Extensions, page 2-13](#)
- [iSCSI Extensions, page 2-14](#)
- [Fabric Profile Extensions, page 2-14](#)
- [Zoning Subprofile Extensions, page 2-16](#)
- [FDMI Subprofile Extensions, page 2-16](#)

VSAN Extensions

A VSAN is a virtual SAN that is created by partitioning the physical fabric into one or more logical fabrics. The Cisco MDS switches base routing on VSANs. The CIM client uses these VSAN extensions to identify VSANs and their associations to physical fabrics and switches.

The VSAN model in the CIM server uses the DMTF partition model. Partitioning, as defined by DMTF, is the virtual division of a single entity into multiple entities. It applies to any resource and can span namespaces and CIM object managers. Each partitioning entity manages its underlying partitions. A partitioned entity may be unaware that it is partitioned, and users may be unaware that a resource is shared. Refer to the standard partitioning model described in the CIM 2.8 schema, available from the DMTF website at <http://www.dmtf.org>.

Send documentation comments to mdsfeedback-doc@cisco.com



Note

For more information about VSANs, refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* or the *Cisco MDS 9000 Family CLI Configuration Guide*.

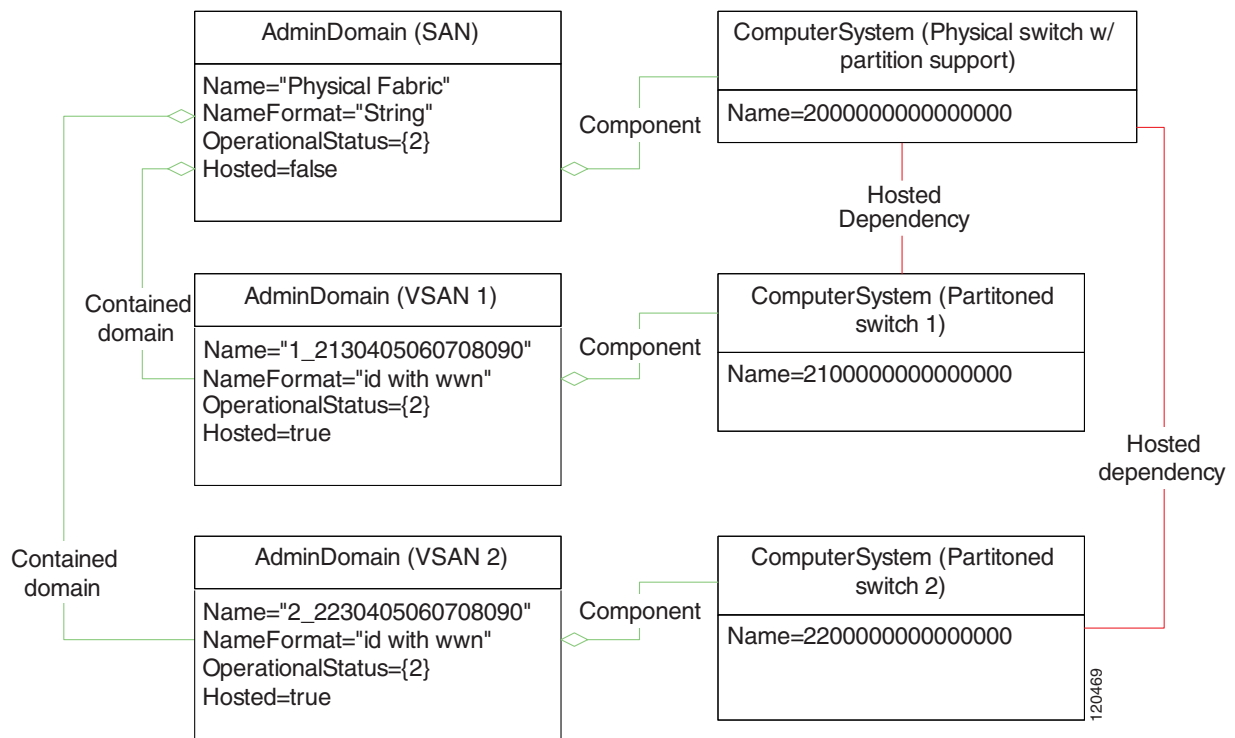
The VSAN extension provided by the Cisco MDS 9000 Family CIM server is both compatible with, and an extension of, the standard partition model. It models VSANs as a partitioned physical fabric. The E ports, F ports, PortChannels, and ports supporting FC IP and iSCSI on Cisco MDS switches all support the partitioning model.

The `HostedDependency` association class can describe the following relationships:

- Partitioning (fan in)
 - Antecedent is the partitioning entity
 - Dependent is the partitioned entity
- Clustering (fan out)

Figure 2-2 shows a UML diagram of a fabric partitioned into two VSANs. The physical switch is partitioned into two logical switches, Partitioned Switch 1 and Partitioned Switch 2. The partitions are identified as belonging to the physical switch using the `HostedDependency` association class. The VSANs are identified as belonging to the corresponding switch partitions using the `Component` association class.

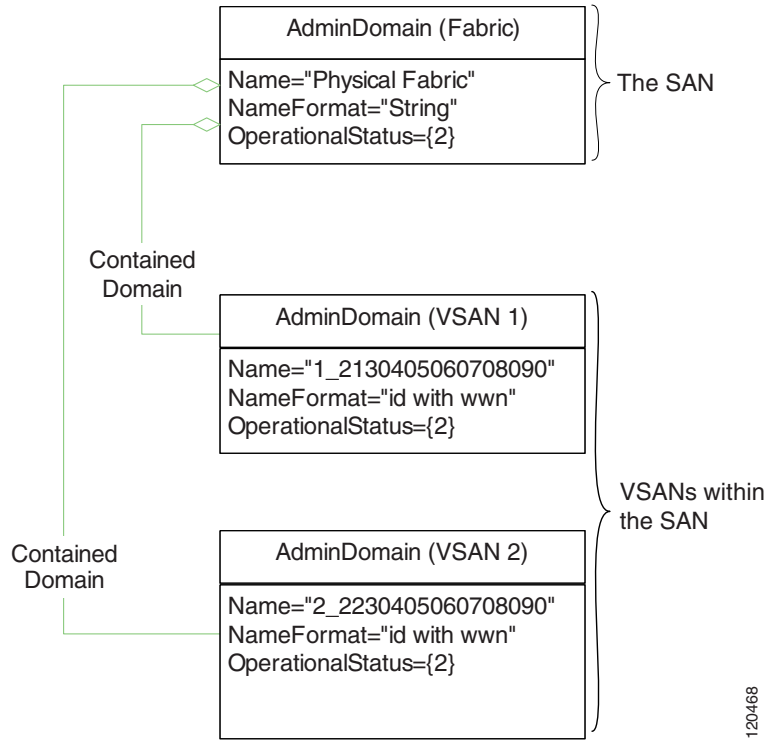
Figure 2-2 UML Diagram of Fabric Partitioning



Send documentation comments to mdsfeedback-doc@cisco.com

Figure 2-3 isolates the VSAN component from Figure 2-2. The physical fabric is partitioned into two VSANs, VSAN 1 and VSAN 2. Each VSAN is identified by the `AdminDomain` class. The VSANs can be identified as belonging to the physical fabric using the `ContainedDomain` association class.

Figure 2-3 VSAN Partitioning Example



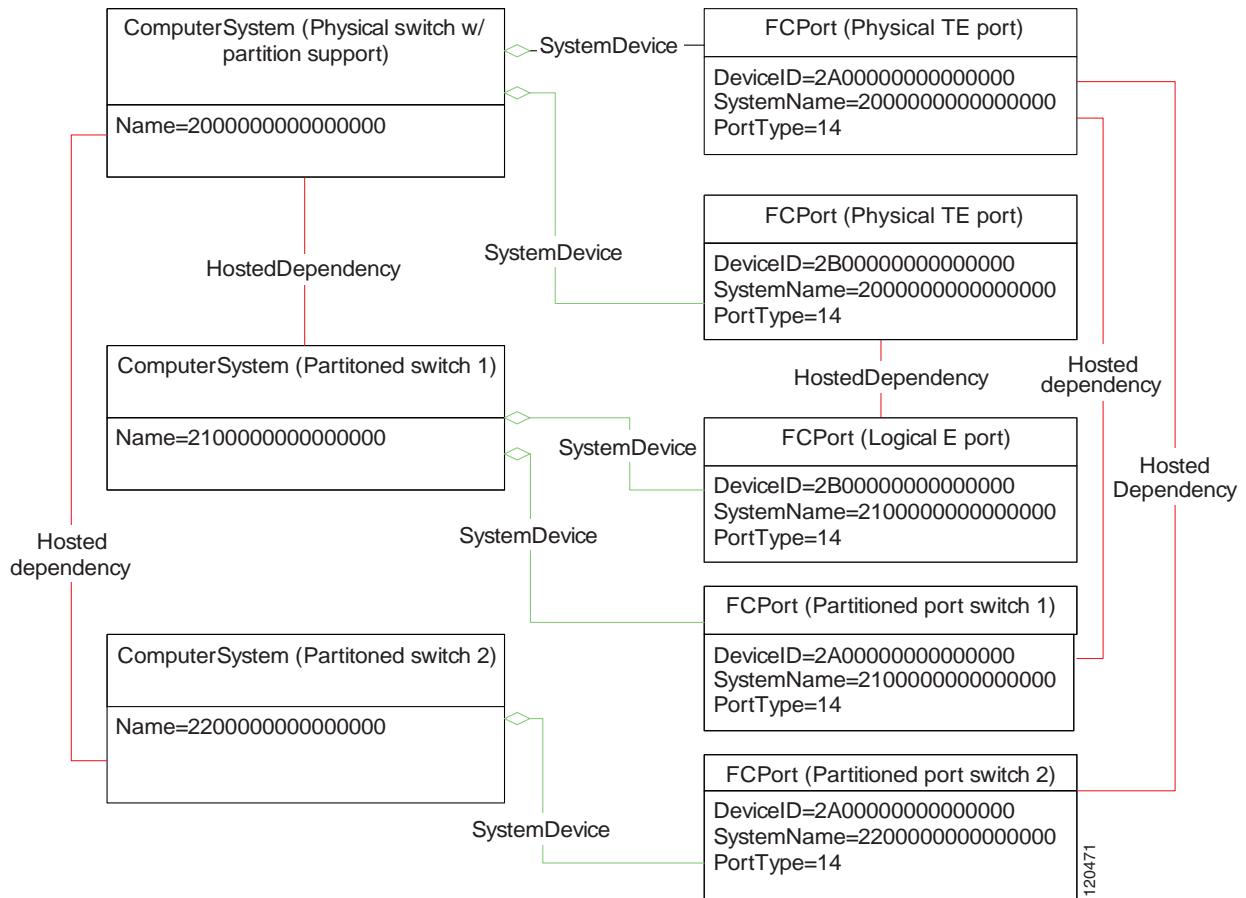
TE Port Extensions

TE ports are E ports that can carry traffic for multiple VSANs. The CIM server uses the existing fabric-to-FC port association classes to model membership of TE ports in multiple VSANs. Figure 2-4 shows the physical and logical port relationship to the switch. The two illustrated physical ports are partitioned into logical ports, and the logical ports are identified as belonging to the physical ports using the `HostedDependency` association class. A physical TE port is partitioned into two logical ports, one for Partitioned switch 1 (associated to VSAN 1 in Figure 2-2) and one for Partitioned switch 2 (associated to VSAN 2 in Figure 2-2).

The physical ports are identified as components of the physical switch using the `SystemDevice` association class, and the partitioned ports are identified as components of the corresponding partitioned switch using the `SystemDevice` association class.

Send documentation comments to mdsfeedback-doc@cisco.com

Figure 2-4 TE Port Partitioning Example



Note

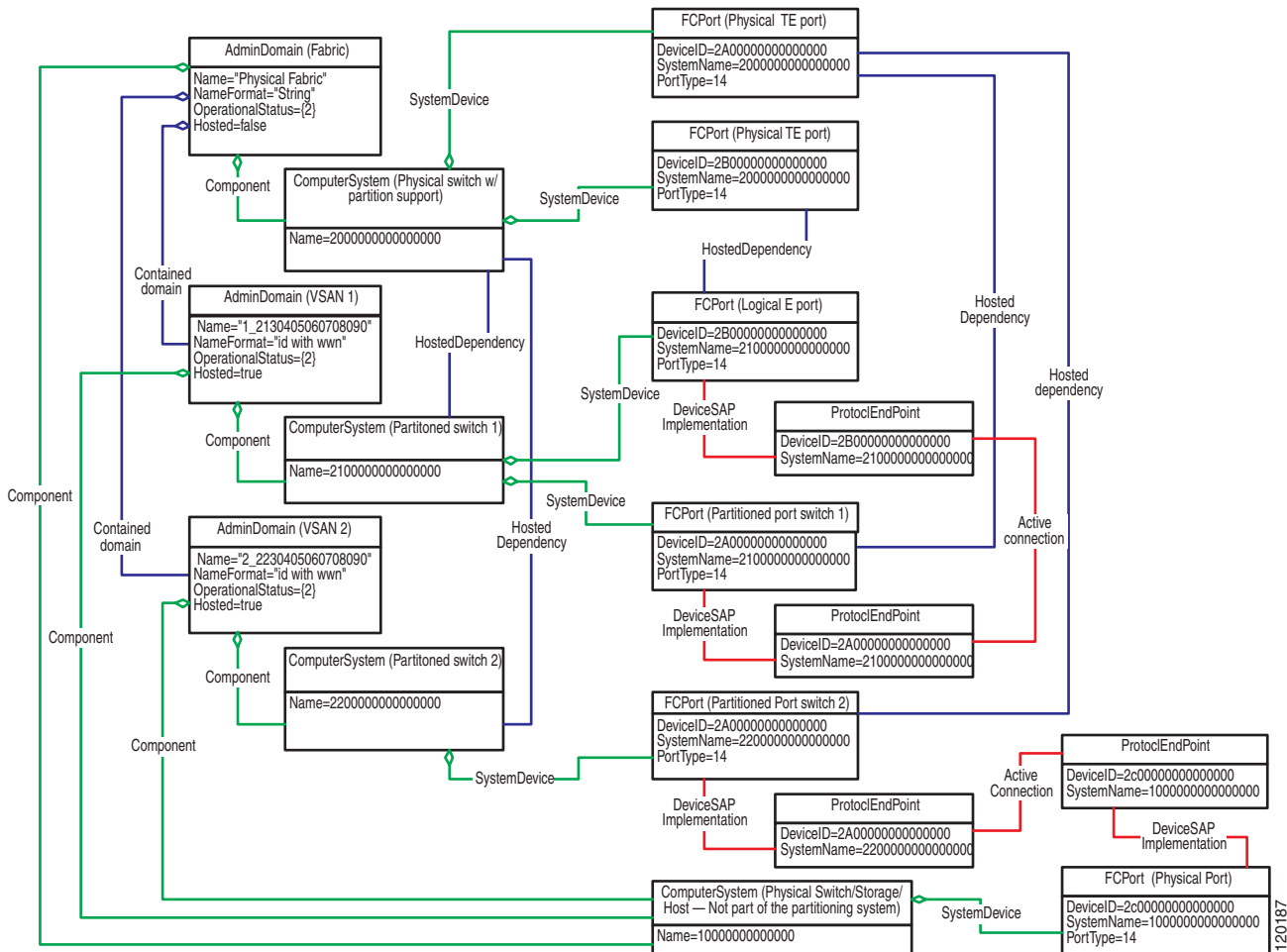
For more information about trunking, refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* or the *Cisco MDS 9000 Family CLI Configuration Guide*.

120471

Send documentation comments to mdsfeedback-doc@cisco.com

Figure 2-5 shows the full UML diagram for VSAN fabric and port partitioning in a SAN switch.

Figure 2-5 UML Diagram for VSAN Partitioning



PortChannel Extensions

A PortChannel is the aggregation of multiple physical Fibre Channel ports into one logical port to provide aggregated bandwidth, load balancing, and link redundancy. The CIM server supports a PortChannel port type in the `Cisco_FCPort` class. The `Component` association class can be used to associate individual ports with a PortChannel.

PortChannels are supported by the CIM server only for the local switch on which the CIM server is running. The CIM server also exports active connections for remote PortChannels, with two limitations:

- The remote PortChannel WWN is not available; the remote switch WWN and port index are provided.
- The `Component` and `LogicalIdentity` association classes of the remote PortChannel are not available.

[Send documentation comments to mdsfeedback-doc@cisco.com](mailto:mdsfeedback-doc@cisco.com)



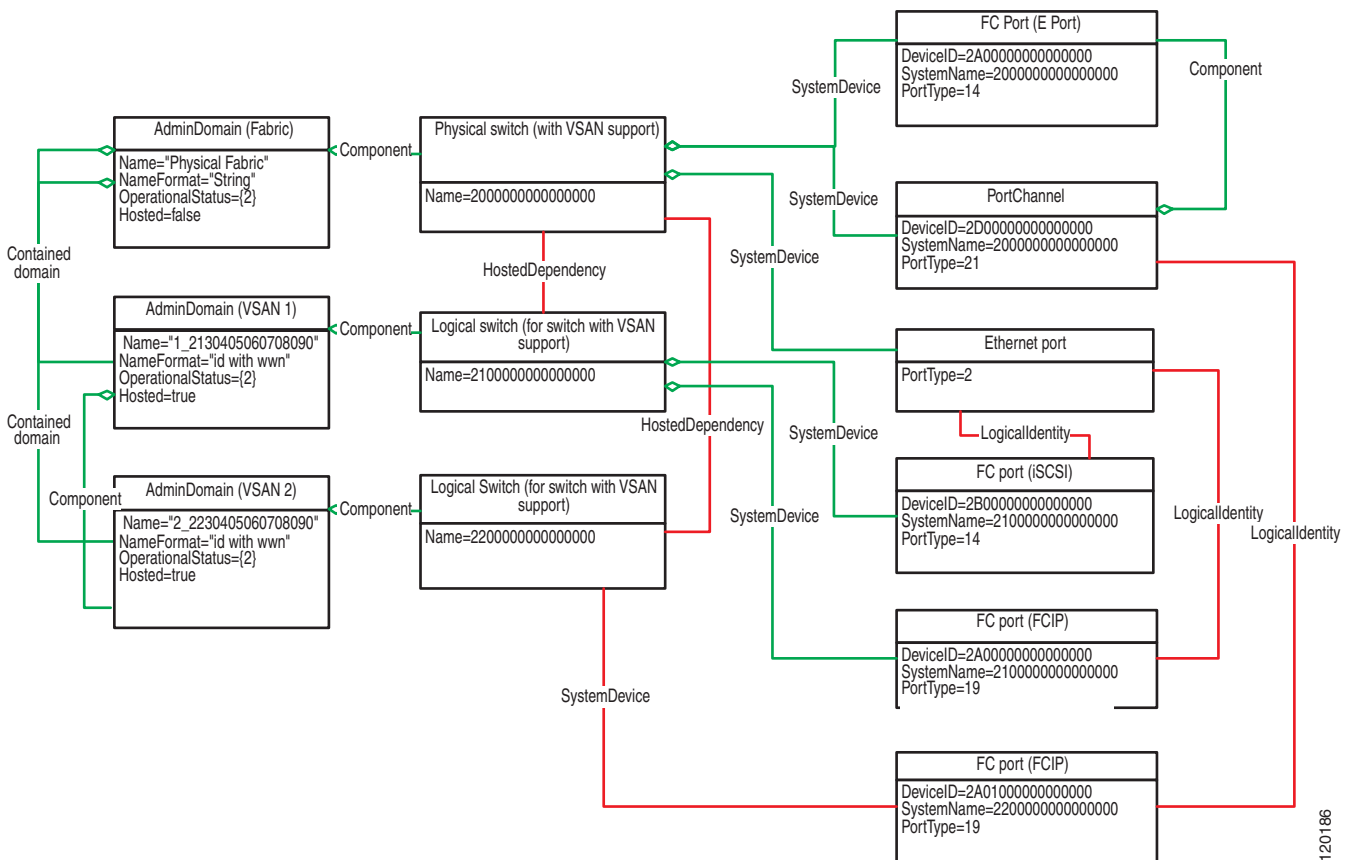
Note

For more information about PortChannels, refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* or the *Cisco MDS 9000 Family CLI Configuration Guide*.

Figure 2-6 shows the relationships among ports and PortChannels on the switch that is running the CIM server. In this example:

- The PortChannels and ports are identified as belonging to the physical switch using the `SystemDevice` association class.
- The individual ports are identified as belonging to the PortChannels using the `Component` association class.

Figure 2-6 UML Instance Diagram of the Relationships Among Ports Using FCIP, PortChannels, and Ethernet Ports



FCIP Extensions

The CIM server uses the current `FCPort` class to discover information about ports supporting FCIP. For the local switch (the switch on which the CIM server is running), the CIM server uses the `LogicalIdentity` association class to link ports supporting FCIP that are on the same module.

The CIM server exports active connections for remote ports running FCIP, with two limitations:

Send documentation comments to mdsfeedback-doc@cisco.com

- The WWN of the port running FCIP is not available; the remote switch WWN and port index are provided.
- The `LogicalIdentity` association class of the port running FCIP is not available.



Note

For more information about FCIP, refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* or the *Cisco MDS 9000 Family CLI Configuration Guide*.

Figure 2-6 shows the relationships among ports running FCIP and other entities. In this example:

- The ports running FCIP are associated with other entities using the `LogicalIdentity` association class. One port running FCIP is a logical entity of an individual Ethernet port, and the other is a logical entity of a PortChannel that is comprised of Ethernet ports.
- The port running FCIP, Ethernet port, and PortChannel are identified as belonging to the physical switch using the `SystemDevice` association class.

iSCSI Extensions

You can use the current `EthernetPort` class to discover information about the port and use the `LogicalIdentity` association class to associate Gigabit Ethernet ports with iSCSI. This association class is only available for ports local to the CIM server.

Figure 2-6 shows the relationships among ports running iSCSI and other entities. In this example:

- The port running iSCSI is identified as belonging to the Ethernet port using the `LogicalIdentity` association class.
- The port running iSCSI is identified as belonging to the physical switch using the `SystemDevice` association class.



Note

For more information about iSCSI, refer to the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* or the *Cisco MDS 9000 Family CLI Configuration Guide*.

Fabric Profile Extensions

In addition to the standard Fabric profile, the following classes and association classes that are specific to the Cisco MDS 9000 Family are supported:

```

CISCO_ActiveConnection
CISCO_AdminDomain
CISCO_FCPort
CISCO_FCPortCapabilities
CISCO_FCPortSettings
CISCO_Vsan
CISCO_Component
CISCO_ComputerSystem
CISCO_ConnectivityCollection
CISCO_ConnectivityMemberOfCollection
CISCO_ContainedDomain
CISCO_DeviceSAPImplementation
CISCO_FCPortStatistics
CISCO_HostedAccessPoint
CISCO_HostedCollection
CISCO_ProtocolEndPoint

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

CISCO_PhysicalPackage
CISCO_PhysicalElement
CISCO_Product
CISCO_Realizes
CISCO_SystemDevice
CISCO_ComputerSystemPackage
CISCO_ElementStatisticalData
CISCO_LogicalPortGroup
CISCO_LogicalModule
CISCO_ModulePort
CISCO_HostedDependency
CISCO_LogicalIdentity
CISCO_PhysicalComputerSystem
CISCO_LogicalComputerSystem
CISCO_FCNodeMemberOfCollection

```

The port identifiers for the CISCO_FCPort class that are supported by the CIM server are described in [Table 2-7](#). Port Identifiers 16004 through 16012 are Cisco extensions.

Table 2-7 Port Identifiers Supported by the Cisco MDS 9000 Family CIM Server

Port Identifier	Port Type
0	Unknown
1	Other
10	N
11	NL
12	F/NL
13	Nx
14	E
15	F
16	FL
17	B
18	G
16004(cisco specific) etc.	PortChannel
16010	FCIP
16011	iSCSI-F
16012	iSCSI-N
16000...65535	Vendor reserved

See the “[Cisco Fabric MOF](#)” section on page A-1 for the full definition of the Cisco fabric extensions.

Send documentation comments to mdsfeedback-doc@cisco.com

Zoning Subprofile Extensions

In addition to the standard zoning subprofiles, the following classes and association classes that are specific to Cisco are supported:

```
CISCO_HostedService
CISCO_ZoneMemberOfCollection
CISCO_ZoneMembershipSettingData
CISCO_ZoneSet
CISCO_Zone
CISCO_ZoneCapabilities
CISCO_ZoneAlias
CISCO_ElementSettingData
CISCO_ZoneService
CISCO_SystemSpecificCollection
```

See the “[Cisco Zone MOF](#)” section on page A-4 for the full definition of the Cisco zoning extensions.

FDMI Subprofile Extensions

In addition to the standard FDMI subprofile, the following classes and association classes that are specific to the Cisco MDS 9000 Family are supported:

```
PortControllerRealizes
PlatformPackage
PortControllerSoftwareIdentity
HBASoftwareInstalledOnPlatform
NodeFCPortControlledByPortController
ProductPhysicalHBA
PlatformInFabric
NodePortInPlatform
NodeInPlatform
PortControllerInPlatform
PortControllerInFabric
```

See the “[Cisco FDMI MOF](#)” section on page A-5 for the full definition of the Cisco FDMI extensions.

CIM Indications

SMI-S provides asynchronous *indications* for changes in the CIM server or the managed elements controlled by the CIM server. These indications can inform a CIM client that:

- The SAN configuration has changed.
- The SAN switch health has degraded.
- The SAN fabric performance has degraded.
- Nameserver Database has changed.
- VSAN added/deleted/modified.
- Fan status has changed.
- Temperature status has changed.
- Power Supply status has changed.

Send documentation comments to mdsfeedback-doc@cisco.com

- FRU inserted/ removed/changed.

Indications can also be used when a CIM class method is invoked that will take a long time to finish. Rather than tie up the CIM server (block) until the operation completes, the CIM server responds that the operation started, and the CIM server continues handling other requests (non-blocking). When the original, long operation completes, the CIM server sends a CIM indication asynchronously to the CIM client, showing the result of the operation. A CIM client must subscribe to indications it wants to receive from the CIM server.

The Cisco MDS 9000 Family CIM server supports the following Cisco-specific indications:

```
CISCO_LinkStateChange
CISCO_LinkUp
CISCO_Linkdown
CISCO_MediaFRUInserted
CISCO_MediaFRURemoved
CISCO_VSANChanged
CISCO_ZoneSetAlert
CISCO_EnvironmentalAlert
CISCO_FanAlert
CISCO_PowerAlert
CISCO_TempAlert
CISCO_NameServerDatabaseChanged
```

See the “[Cisco Indications MOF](#)” section on page A-15 for the Cisco Indications MOF file.

Send documentation comments to mdsfeedback-doc@cisco.com



CHAPTER 3

Configuring and Using the CIM Server

This chapter provides the steps to configure the CIM server in Cisco MDS 9000 Family products and gives some sample scenarios for using CIM objects to manage your SAN. This chapter includes the following sections:

- [Configuring the CIM Server, page 3-1](#)
- [Performing Discovery and Performance Monitoring, page 3-3](#)
- [Modeling a Module Using the Blade Subprofile, page 3-4](#)
- [Configuring Zoning, page 3-4](#)



Note

For information about CLI commands, refer to the *Cisco MDS 9000 Family Command Reference*.

Configuring the CIM Server

The CIM server can be configured through the CLI. To configure the CIM server, you must first enable it. For added security, you can install an SSL certificate to encrypt the login information and enable HTTPS before enabling the CIM server. The CIM server requires HTTP or HTTPS or both to be enabled. By default, HTTP is enabled and secure HTTPS is disabled. Using HTTPS encrypts all management traffic between the CIM client and the CIM server and is the recommended configuration.

Creating a Certificate Using OpenSSL

You need a valid certificate to configure the CIM server. You can use OpenSSL to create the private key and certificate needed by the CIM server. Refer to <http://www.openssl.org>.

To create a self-signed certificate and private key using OpenSSL, follow these steps:

Step 1

Create a file called `ssl.conf` on your workstation. This is used to specify the distinguished name. Sample contents of this file are:

```
[ req ]
distinguished_name = req_distinguished_name
prompt = no
[ req_distinguished_name ]
CN = Common Name
emailAddress = test@email.address
```

Send documentation comments to mdsfeedback-doc@cisco.com

- Step 2** Use the **openssl** command to create the private key and the certificate by typing the following:
- ```
/usr/bin/openssl req -x509 -days 365 -newkey rsa:2048 -nodes -config ./ssl.conf -keyout ./key.pem -out ./cert.pem
```
- Step 3** Concatenate the private key and the certificate into a single file.
- ```
cat key.pem cert.pem > cimserver1.pem
```
- Step 4** Copy cimserver1.pem to bootflash: on your switch. You use this as the certificate when configuring the CIM server.

Installing the Certificate and Enabling the CIM Server

To configure a CIM server using the HTTPS protocol, follow these steps:

	Command	Purpose
Step 1	switch# config t	Enters configuration mode.
Step 2	switch(config)# cimserver certificate bootflash:cimserver1.pem	Installs a Secure Socket Layer (SSL) certificate specified in the file named with a .pem extension.
	switch(config)# cimserver clearcertificate Certificate1	Optional. Clears the specified SSL certificate (Certificate1).
Step 3	switch(config)# cimserver enableHttps	Enables HTTPS (secure protocol).
	switch(config)# no cimserver enableHttps	Optional. Disables HTTPS (default).
Step 4	switch(config)# cimserver enable	Enables the CIM server.
	switch(config)# no cimserver enable	Optional. Disables the CIM server (default).

To configure a CIM server using the HTTP protocol, follow these steps:

	Command	Purpose
Step 1	switch# config t	Enters configuration mode.
Step 2	switch(config)# cimserver enable	Enables the CIM server using the default HTTP (non-secure) protocol.
	switch(config)# no cimserver enable	Optional. Disables the CIM server (default).
	switch(config)# no cimserver enableHttp	Optional. Disables HTTP.
	switch(config)# cimserver enableHttp	Optional. Enables HTTP and reverts to the switch default.

Setting CIM Server Loglevel

To configure a CIM server loglevel, follow these steps:

	Command	Purpose
Step 1	switch# show cimserver logs	Displays CIM server logs.

Send documentation comments to mdsfeedback-doc@cisco.com

	Command	Purpose
Step 2	switch# <code>conf t</code>	Configures CIM server loglevels.
Step 3	switch(config)# <code>cimserver logLevel</code>	Sets the CIM server loglevel. The loglevels range from 1-5 (1-trace, 2-information, 3-warning, 4-severe, 5-fatal).

Performing Discovery and Performance Monitoring

You can use the Fabric and Switch profiles to implement discovery and performance monitoring. See the “[Fabric Profile](#)” section on page 2-4 and the “[” section on page 2-7 for more information on these profiles.](#)

Discovery provides information about the physical and logical entities within the SAN. This information changes dynamically as SAN entities are added, moved, or removed. Discovery also includes the discovery of object classes as well as related association classes, properties, and return status codes that are provided by servers in the managed environment.

[Table 3-1](#) shows how to perform discovery, using the intrinsic methods defined by CIM. Use these methods to retrieve information about the switch and fabric.

Table 3-1 *Performing Discovery*

Method	How Used
<code>enumerateInstances()</code>	Enumerate instances of a CIM class.
<code>enumerateInstanceNames()</code>	Enumerate names of instances of a CIM class.
<code>getInstance()</code>	Get a CIM instance.
<code>associators()</code>	Enumerate associators of a CIM object.
<code>associatorName()</code>	Enumerate names of associators of a CIM object.
<code>references()</code>	Enumerate references to a CIM object.
<code>referenceName()</code>	Enumerate names of references to a CIM object.

The target of these methods is the location of the CIM server, which is identified by the switch IP address.

Performance monitoring provides the status and statistics for the local ports. Only ports on the local switch can be monitored. You can retrieve statistics from the properties of the `FCPortStatistics` class for `FCPort` class instances on the CIM server.



Note

The namespace of the CIM server is `root/cimv2`.

Send documentation comments to mdsfeedback-doc@cisco.com

Modeling a Module Using the Blade Subprofile

You can use the blade subprofile to model a supervisor module, switching module, or services module within a switch. [Table 3-2](#) shows how to use the association classes in this subprofile to map ports to modules and modules to switches.

Table 3-2 Using the Blade Subprofile Association Classes

Class	How Used
Realizes	Associates the <code>LogicalModule</code> class to the <code>PhysicalPackage</code> class. Use this class to map modules to the switch.
ModulePort	Associates the <code>FCPort</code> class to the <code>LogicalModule</code> class. Use this class to map individual ports to modules within the switch.

See the “Blade Subprofile” section on page 2-3 for more information about the Blade subprofile.

Configuring Zoning

The zoning model in the SMI-S uses extrinsic and intrinsic methods to manage zoning within the SAN fabric. Extrinsic methods are methods specific to a particular class. Intrinsic methods are methods inherited from the CIM and present in every applicable class.

To create a zone member (referred to as `ZoneMembershipSettingData`), zone, zone alias, or zone set, use `invokeMethod(operand)`. The operand can be one of the extrinsic methods from the zoning subprofiles as shown in [Table 3-3](#).

Table 3-3 Zoning Extrinsic Methods

Extrinsic Method	How Used
<code>CreateZoneMembershipSettingData()</code>	Creates a <code>ZoneMembershipSettingData</code> and adds it to the specified <code>Zone</code> or <code>NamedAddressCollection</code> . The <code>ConnectivityMemberID</code> is dependent upon the <code>ConnectivityMemberType</code> .
<code>CreateZone()</code>	Creates a <code>Zone</code> and associates it to <code>AdminDomain</code> where the <code>ZoneService</code> is hosted.
<code>CreateZoneAlias()</code>	Creates a <code>ZoneAlias</code> and associates it to <code>AdminDomain</code> where the <code>ZoneService</code> is hosted.
<code>CreateZoneSet()</code>	Creates a <code>ZoneSet</code> and associates it to the <code>AdminDomain</code> where the <code>ZoneService</code> is hosted.
<code>AddZone()</code>	Adds the <code>Zone</code> to the specified <code>ZoneSet</code> . Adding a <code>Zone</code> to a <code>ZoneSet</code> extends the zone enforcement definition of the <code>ZoneSet</code> to include the members of that <code>Zone</code> . If adding the <code>Zone</code> is successful, the <code>Zone</code> should be associated to the <code>ZoneSet</code> by <code>MemberOfCollection</code> .
<code>AddZoneMembershipSettingData()</code>	Adds <code>ZoneMembershipSettingData</code> to the <code>Zone</code> or <code>NamedAddressCollection</code> .

Send documentation comments to mdsfeedback-doc@cisco.com

Table 3-3 Zoning Extrinsic Methods (continued)

Extrinsic Method	How Used
AddZoneAlias()	Adds the ZoneAlias to the Zone.
ActivateZoneSet ()	Sets the ZoneSet to active.

Use the DeleteInstance(*instance_name*) intrinsic method to remove a zoning item from a collection or to delete the zoning item entirely. The DeleteInstance() method requires a reference to one of the instances shown in Table 3-4.

Table 3-4 Deleting Zoning Entities

Class	How Used
CIM_ElementSettingData	Removes a zone member from a zone or zone alias. Use deleteInstance() to delete the instance of ElementSettingData that associates the zone member to the zone.
CIM_MemberOfCollection	Removes a zone or zone alias from a zone set. Use deleteInstance() to delete the instance of MemberOfCollection that associates the zone or zone alias to the zone set.
CIM_ZoneMembershipSettingData	Deletes a zone member. This automatically removes it from any zone or zone alias.
CIM_Zone	Deletes a zone.
CIM_ZoneAlias	Deletes a zone alias.
CIM_ZoneSet	Deletes a zone set.

See the “Zone Control Subprofile” section on page 2-5 and the “Enhanced Zoning and Enhanced Zoning Control Subprofile” section on page 2-6 for information about the zoning subprofiles.



Note

These methods are supported for the CIM protocol only and cannot be entered as commands at the CLI. For more information about SMI-S, refer to the SNIA website at <http://www.snia.org>. For more information about CIM, refer to the DMTF website at <http://www.dmtf.org>.

Send documentation comments to mdsfeedback-doc@cisco.com



APPENDIX **A**

Managed Object Format Files

This appendix provides the text from the Managed Object Format (MOF) files for the Cisco MDS 9000 Family CIM server extensions. These MOF files are an extension to the standard MOF files and provide management for VSANs, PortChannels, FCIP, and iSCSI.

For information about the standard MOF files, refer to the DMTF website at the following URL: <http://www.dmtf.org>.

This appendix includes the following sections:

- [Cisco MOF Files for Cisco SAN-OS Release 3.0\(1\) or Later, page A-1](#)
- [Cisco MOF Files for Cisco SAN-OS Release 2.x, page A-12](#)
- [Cisco Indications MOF, page A-15](#)

Cisco MOF Files for Cisco SAN-OS Release 3.0(1) or Later

This section includes the MOF files supported by Cisco SAN-OS Release 3.0(1) or later. It includes the following topics:

- [Cisco Fabric MOF, page A-1](#)
- [Cisco Zone MOF, page A-4](#)
- [Cisco FDMI MOF, page A-5](#)

Cisco Fabric MOF

The Cisco Fabric MOF for Cisco SAN-OS Release 3.0(1) or later provides extensions to the Fabric profile to manage VSANs, PortChannels, and other Cisco-specific entities within the fabric. See the [“” section on page 2-7](#).

```
[Version ("1.0.0"), Description (
    "cisco fabric and switch profile classes")]
class CISCO_ActiveConnection : CIM_ActiveConnection
{
};
```

```
class CISCO_AdminDomain : CIM_AdminDomain
{
};
```

```
    [Version ( "2.7.1"), Description (
        "Capabilities and management of a Fibre Channel Port Device." ) ]
class CISCO_FCPort : CIM_FCPort {
```

Send documentation comments to mdsfeedback-doc@cisco.com

```

[Override ( "PortType"), Description (
    "The specific mode currently enabled for the Port. The "
    "values: \"N\" = Node Port, \"NL\" = Node Port supporting FC "
    "arbitrated loop, \"E\" = Expansion Port connecting fabric "
    "elements (for example, FC switches), \"F\" = Fabric "
    "(element) Port, \"FL\" = Fabric (element) Port supporting "
    "FC arbitrated loop, \"B\" = Bridge and \"G\" = Generic "
    "Port. PortTypes are defined in the ANSI X3 standards. "
    "When set to 1 (\"Other\"), the related property "
    "OtherPortType contains a string description of the port's "
    "type."),
ValueMap { "0", "1", "10", "11", "12", "13", "14", "15", "16",
    "17", "18", "16004", "16010", "16011", "16012", "16000.65535"},
Values { "Unknown", "Other", "N", "NL", "F/NL", "Nx", "E", "F",
    "FL", "B", "G", "PortChannel", "FCIP", "ISCSI-F", "ISCSI-N", "Vendor Reserved"}
]
uint16 PortType;
uint16 PortAvailability = 2;
};

class CISCO_Vsan : CIM_AdminDomain {
};
class CISCO_Component : CIM_Component
{};

class CISCO_ComputerSystem : CIM_ComputerSystem
{};

class CISCO_ConnectivityCollection : CIM_ConnectivityCollection
{};

class CISCO_ConnectivityMemberOfCollection : CIM_MemberOfCollection
{};

class CISCO_ContainedDomain : CIM_ContainedDomain
{};

class CISCO_DeviceSAPImplementation : CIM_DeviceSAPImplementation
{};

class CISCO_FCPortStatistics : CIM_FCPortStatistics
{};

class CISCO_HostedAccessPoint : CIM_HostedAccessPoint
{};

class CISCO_HostedCollection : CIM_HostedCollection
{};

class CISCO_ProtocolEndPoint : CIM_ProtocolEndPoint
{};

class CISCO_PhysicalPackage : CIM_PhysicalPackage
{};

class CISCO_PhysicalElement : CIM_PhysicalElement
{};

class CISCO_Product : CIM_Product
{};

class CISCO_Realizes : CIM_Realizes
{};

```

Send documentation comments to mdsfeedback-doc@cisco.com

```
class CISCO_SystemDevice : CIM_SystemDevice
{};

class CISCO_ComputerSystemPackage : CIM_ComputerSystemPackage
{};

class CISCO_ProductPhysicalComponent : CIM_ProductPhysicalComponent
{};

class CISCO_ElementStatisticalData : CIM_ElementStatisticalData
{};

class CISCO_LogicalPortGroup : CIM_LogicalPortGroup
{};

class CISCO_LogicalModule : CIM_LogicalModule
{};

class CISCO_ModulePort : CIM_ModulePort
{};

class CISCO_EthernetPort : CIM_EthernetPort
{};

class CISCO_HostedDependency : CIM_HostedDependency
{};

class CISCO_LogicalIdentity : CIM_LogicalIdentity
{};

class CISCO_PhysicalComputerSystem : CISCO_ComputerSystem
{};

class CISCO_LogicalComputerSystem : CISCO_ComputerSystem
{};

class CISCO_FCNodeMemberOfCollection : CIM_MemberOfCollection
{};

class CISCO_FabricHostedService : CIM_HostedService
{};

class CISCO_ObjectManagerHost : CIM_System
{};

class CISCO_FCPortCapabilities : CIM_FCPortCapabilities
{};

class CISCO_FCSwitchCapabilities : CIM_FCSwitchCapabilities
{};

class CISCO_FCPortSettings : CIM_FCPortSettings
{};

class CISCO_FCSwitchSettings : CIM_FCSwitchSettings
{};

class CISCO_ElementCapabilities : CIM_ElementCapabilities
{};

class CISCO_ElementSettingDataSys : CIM_ElementSettingData
{};

class CISCO_SoftwareIdentity : CIM_SoftwareIdentity
```

Send documentation comments to mdsfeedback-doc@cisco.com

```
{};

class CISCO_ElementSoftwareIdentity : CIM_ElementSoftwareIdentity
{};

class CISCO_SAPAvailableForElement : CIM_SAPAvailableForElement
{};

class CISCO_RemoteServiceAccessPoint : CIM_RemoteServiceAccessPoint
{};
```

Cisco Zone MOF

The Cisco Zone MOF for Cisco SAN-OS Release 3.0(1) or later provides extensions to the zoning subprofiles. See the [“” section on page 2-7](#).

```
[Version ("1.0.0"), Description (
    "cisco zoneset class")]
class CISCO_ZoneSet : CIM_ZoneSet
{};

class CISCO_Zone : CIM_Zone
{};

class CISCO_ZoneAlias : CIM_NamedAddressCollection
{};

class CISCO_ZoneMemberSettingData : CIM_ZoneMembershipSettingData{

[Override ( "ConnectivityMemberType" ), Description (
    "ConnectivityMemberType specifies the type of identification "
    "used in the ConnectivityMemberID field. For Fibre Channel, "
    "several of the enumerated values require additional "
    "explanation: \n"
    "** A ConnectivityMemberType equal to 2 (Permanent Address) "
    "indicates that an NxPort WWN (pWWN) value should be specified in "
    "the related ConnectivityMemberID property. \n"
    "** A ConnectivityMemberType of 3 (FCID) indicates "
    "that an NxPort Address ID(FCID) value should be specified in the "
    "related ConnectivityMemberID property. \n"
    "** A ConnectivityMemberType of 4 (Switch Port ID) indicates "
    "that a Domain or Port Number(DomainID) value should be specified in "
    "the related ConnectivityMemberID property.(eg. 06:40) \n"
    "** A ConnectivityMemberType of 5 (fcalias) "
    "indicates that alias name which denotes a port ID or WWN should be "
    "specified in the related ConnectivityMemberID property."
    "** A ConnectivityMemberType of 6 (Interface) "
    "indicates that a interface of local switch. The fc interface should"
    "be specified in the related ConnectivityMemberID property(eg. fc1/9)"
    "** A ConnectivityMemberType of 7 (fWWN) "
    "indicates that Fabric port WWN.The WWN of the fabric "
    "port value should be specified in the "
    "related ConnectivityMemberID property."
    "** A ConnectivityMemberType of 8 (Network Address IpV4) "
    "indicates that IPv4 address of an attached device in 32 bits"
    "in dotted decimal format should be specified in the "
    "related ConnectivityMemberID property."
    "** A ConnectivityMemberType of 9 (Network Address IpV6) "
    "indicates that IPv6 address-The IPv6 address of an attached device "
    "in 128 bits in colon(:)-separated hexadecimal format should be specified"
    " in related ConnectivityMemberID property."
```

Send documentation comments to mdsfeedback-doc@cisco.com

```

    ** A ConnectivityMemberType of 10 (Interface with Remote SWWN) "
    "indicates that a interface of remote switch. The fc interface should"
    "be specified along with Switch WWN in the related ConnectivityMemberID"
    "property(eg. fc1/9:20000005300084DF)"
    ** A ConnectivityMemberType of 11 (Interface with DomainID) "
    "indicates that a interface of local switch. The fc interface should"
    "be specified along with the Domain Id in the related "
    "ConnectivityMemberID property(eg.fc1/9:25)" )]
    ** A ConnectivityMemberType of 12 (Symbolic-node name) "
    "indicates that a symbolic-node name"
    "should be specified in the "
    "related ConnectivityMemberID property."

    uint16 ConnectivityMemberType;
};

class CISCO_ZoneService : CIM_ZoneService
{};

class CISCO_SystemSpecificCollection : CIM_SystemSpecificCollection
{};

class CISCO_ZoneMemberOfCollection : CIM_MemberOfCollection
{};

class CISCO_ElementSettingData : CIM_ElementSettingData
{};

class CISCO_HostedService : CIM_HostedService
{};

class CISCO_ZoneHostedCollection : CIM_HostedCollection
{};

class CISCO_ZoneCapabilities : CIM_ZoneCapabilities
{};

```

Cisco FDMI MOF

The Cisco FDMI MOF for Cisco SAN-OS Release 3.0(1) or later provides extensions to the Fabric profile to manage VSANs, PortChannels, and other Cisco-specific entities within the fabric. See the [section on page 2-7](#).

```

[Provider("FDMI_Provider"),Description (
    "This class represents FDMI enabled physical HBA card attached "
    "to a switch" )]
class CISCO_PhysicalHBA: CIM_PhysicalPackage {

    [Override("Tag"), Key, MaxLen (256), Description (
        "A unique physical identifier that serves as the key for "
        "the HBA. The HBA serial number could be used as a tag.\n" )]
    string Tag;

    [Override("CreationClassName"), Key, MaxLen (256), Description (
        "CreationClassName indicates the name of the class or the "
        "subclass used in the creation of an instance.")]
    string CreationClassName= "CISCO_PhysicalHBA";

    [Override("Manufacturer"), MaxLen (256), Description (
        "The name of the organization responsible for "

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

        "manufacturing the HBA.")]
string Manufacturer;

[Override("Model"), MaxLen (64), Description (
    "The name by which the HBA is generally known.")]
string Model;

[Description (
    "The detailed description of the model of the HBA. The "
    "value might provide a more detailed identification of the "
    "HBA than the Model property does."),
    MaxLen (256)]
string ModelDescription;

[Override("SerialNumber"), MaxLen (64), Description (
    "A manufacturer-allocated number used to identify the HBA. "
    "This value SHOULD match a serial number engraved or "
    "printed in the HBA.")]
string SerialNumber;

[Override("Version"), MaxLen (64), Description (
    "A string indicating the version of the HBA card.")]
string Version;
} ;

/// CISCO_HBAProduct
[Provider("FDMIProvider"),
    Description ("This class represents product information of FDMI enabled physical
        HBA card attached to a switch."
)]

class CISCO_HBAProduct: CIM_Product {

    [Override("Name"),Key, Description (
        "Commonly used Product name."),
        MaxLen ( 256 )]
string Name;

    [Override("IdentifyingNumber"),Key, Description (
        "A manufacturer-allocated number used to identify the HBA. "
        "This value SHOULD match a serial number engraved or "
        "printed in the HBA."),
        MaxLen ( 64 )]
string IdentifyingNumber;

    [Override("Vendor"),Key, Description (
        "The name of the Product's supplier, or entity selling the "
        "Product (the manufacturer, reseller, OEM, etc.). "
        "Corresponds to the Vendor property in the Product object in "
        "the DMTF Solution Exchange Standard."),
        MaxLen ( 256 )
    ]
string Vendor;

    [Override("Version"),Key, Description (
        "A string indicating the version of the HBA card."),
        MaxLen ( 64 )]
string Version;

    [Override("ElementName"), Description(
        "The detailed description of the model of the HBA. The "
        "value might provide a more detailed identification of the "
        "HBA than the Model property does ")]
string ElementName;

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

};

// CISCO_Platform
[Provider("FDMIPProvider"),
  Description (
    "CISCO_Platform represents a fabric-connected entity, "
    "containing one or more Node objects, that has registered "
    "with a fabric's Management Server service.")]

class CISCO_Platform: CIM_ComputerSystem {

  [Override ("CreationClassName"), Key, MaxLen (256),
    Description (
      "CreationClassName indicates the name of the class or the "
      "subclass used in the creation of an instance.")]
  string CreationClassName= "CISCO_Platform";

  [Override ("Name"), Key, MaxLen (256), Description (
    "The inherited Name serves as key of the platform in an "
    "enterprise environment. This value has the following "
    "format:\n"
    "\"WWN\": \"Platform Name\".")]
  string Name;

  [Override ("ElementName"), Required, Description (
    "A user-friendly name for the object. This property allows "
    "each instance to define a user-friendly name IN ADDITION TO "
    "its key properties/identity data, and description "
    "information.")]
  string ElementName;

  [Override ( "NameFormat" ),Required, Description (
    "The ComputerSystem object and its derivatives are Top Level "
    "Objects of CIM. They provide the scope for numerous "
    "components. Having unique System keys is required. The "
    "NameFormat property identifies how the ComputerSystem Name "
    "is generated. The NameFormat ValueMap qualifier defines the "
    "various mechanisms for assigning the name. Note that "
    "another name can be assigned and used for the "
    "ComputerSystem that better suit a business, using the "
    "inherited ElementName property."),
    ValueMap { "Other", "IP", "Dial", "HID", "NWA", "HWA", "X25",
      "ISDN", "IPX", "DCC", "ICD", "E.164", "SNA", "OID/OSI",
      "WWN", "NAA" }}
  string NameFormat = "Other";

  [Write, Override ("Dedicated"), Description(
    "Platform type. Although this is represented as an array, "
    "only one type is specified at any given time (array size is "
    "always 1). When writing this property, users should "
    "specify only a single type in an array size of exactly 1. "
    "Specifying more or less than 1 type results in an exception "
    "with an invalid argument error code."),
    Values{"Unknown", "Others", "Gateway", "dummy3", "dummy4",
      "Converter", "HBA", "Swproxy", "StorageDev", "Host",
      "Storsubsys", "Module", "Driver", "StorAccess"},
    ValueMap {"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
      "11", "12", "13"}}
  uint16 Dedicated[];

  [Override ("OtherIdentifyingInfo"), Description(
    "Platform name: for example, host name.")]
  string OtherIdentifyingInfo[];

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

[Write, Description(
    "The set of management IP Addresses used to access this "
    "platform.")]
string MgmtAddressList[];
};
// CISCO_PortController
[Description("CISCO_PortController represents the port controller of an FDMI enabled
HBA.")]
class CISCO_PortController: CIM_PortController {

    [Override("SystemCreationClassName"), Key, MaxLen (256), Description (
        "The scoping system's creation class name. The "
        "scoping system is the CISCO_Platform or "
        "CISCO_Fabric of which this device is part.")]
    string SystemCreationClassName;

    [Override("SystemName"), Key, MaxLen (256), Description (
        "The scoping system's Name property. The value "
        "is equivalent to the platform name if the scoping system is an "
        "instance of CISCO_Platform or the Proxy Switch WWN if the "
        "scoping system is an instance of CISCO_Fabric.")]
    string SystemName;

    [Override("CreationClassName"), Key, MaxLen (256),
    Description (
        "CreationClassName indicates the name of the CISCO_PortController "
        "class that, when used with the other key properties of this "
        "class, uniquely identifies an instance of the "
        "CISCO_PortController class.")]
    string CreationClassName= "CISCO_PortController";

    [Override("DeviceID"), Key, MaxLen (64), Description (
        "This is the Serial Number of the HBA")]
    string DeviceID;

    [Override("ControllerType"), Required, Description (
        "The type or model of the port controller. Specific values "
        "will be enumerated in a later release of this schema. When "
        "set to 1 (\"Other\"), the related property "
        "OtherControllerType contains a string description of the "
        "controller's type."),
    ValueMap { "0", "1", "2", "3", "4", "5", "6", "7", "8" },
    Values { "Unknown", "Other", "Ethernet", "IB", "FC", "FDDI",
        "ATM", "Token Ring", "Frame Relay" }]
    uint16 ControllerType = 4;
};

class CISCO_HBASoftwareIdentity : CIM_SoftwareIdentity
{};

class CISCO_ElementSoftwareIdentity : CIM_ElementSoftwareIdentity
{};

// Associations

// CISCO_PortControllerRealizes

[Association,
    Provider("FDMIProvider"),
    Description (
        "CISCO_PortControllerRealizes is the association that defines "
        "the mapping between devices and the physical elements "
        "that implement them.")]

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

class CISCO_PortControllerRealizes: CIM_Realizes {

    [Override ("Antecedent"), Description (
        "The physical HBA that implements the Device.")]
    CISCO_PhysicalHBA REF Antecedent;

    [Override ("Dependent"), Description (
        "The Device.")]
    CISCO_PortController REF Dependent;
};
// CISCO_PlatformPackage

[Association,
    Description (
        "This association denotes one or more physical HBAs that "
        "realize a Platform.")]
class CISCO_PlatformPackage: CIM_ComputerSystemPackage {

    [Override ("Antecedent"), Description (
        "The physical HBA that realizes a Platform.")]
    CISCO_PhysicalHBA REF Antecedent;

    [Override ("Dependent"), Description (
        "The Platform.")]
    CISCO_Platform REF Dependent;
};
// CISCO_PortControllerSoftwareIdentity

[Association,
    Description (
        "The PortControllerSoftwareIdentity relationship identifies any "
        "software that is associated with the device and this association "
        "can return multiple instances.")]
class CISCO_PortControllerSoftwareIdentity: CIM_ElementSoftwareIdentity {

    [Override ("Antecedent"), Description (
        "The SoftwareIdentity on the device.")]
    CISCO_HBASoftwareIdentity REF Antecedent;

    [Override ("Dependent"), Description (
        "The logical device that requires or uses the software.")]
    CISCO_PortController REF Dependent;
};
// CISCO_HBASoftwareInstalledOnPlatform

[Association,
    Description (
        "The SoftwareInstalledOnPlatform relationship allows the "
        "identification of the platform on which HBA driver "
        "is installed and this association can return multiple instances.")]
class CISCO_HBASoftwareInstalledOnPlatform: CIM_InstalledSoftwareIdentity {

    [Key, Override("System"), Max (1), Description (
        "Reference to the platform hosting a particular "
        "SoftwareIdentity.")]
    CISCO_Platform REF System;

    [Key, Override("InstalledSoftware"), Description (
        "Reference to the driver that is installed on the "
        "platform.")]
    CISCO_HBASoftwareIdentity REF InstalledSoftware;
};
// CISCO_NodeFCPortControlledByPortController

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

[Association,
  Description (
    "This association represents the relationship between a "
    "device and ports.")]
class CISCO_NodeFCPortControlledByPortController: CIM_ControlledBy {
  [Override ("Antecedent"), Description (
    "The device that controls the port.")]
  CISCO_PortController REF Antecedent;

  [Override ("Dependent"), Description (
    "The port being controlled.")]
  CISCO_FCPort REF Dependent;

  [Override("DeviceNumber"), MaxLen(255), Description (
    "Address of associated port in context of the antecedent "
    "device. This may be a comma-separated list in case there "
    "are multiple addresses.")]
  string DeviceNumber;
};
// CISCO_ProductPhysicalHBA

[Association,
  Description (
    "The HBA is shipped to the customer by a third party "
    "(OEM/reseller) to the customer. This class associates "
    "the HBA with the product.")]
class CISCO_ProductPhysicalHBA: CIM_ProductPhysicalComponent {

  [Override ("GroupComponent"), Description (
    "The product.")]
  CISCO_HBAProduct REF GroupComponent;

  [Override ("PartComponent"), Description (
    "The HBA that is shipped as a product.")]
  CISCO_PhysicalHBA REF PartComponent;
};

CISCO_PlatformInFabric

[Association, Aggregation,
  Description (
    "CISCO_PlatformInFabric is a generic association used to "
    "establish membership relationships between the fabric and "
    "platforms connected to the fabric.")]
class CISCO_PlatformInFabric: CIM_Component {

  [Override("GroupComponent"), Aggregate, Key, Description (
    "The fabric that has connected platforms.")]
  CISCO_VSAN REF GroupComponent;

  [Override("PartComponent"), Key, Description (
    "The platforms connected to this fabric.")]
  CISCO_Platform REF PartComponent;
};
// CISCO_NodePortInPlatform

[Association, Aggregation,
  Description (
    "CISCO_NodePortInPlatform is a generic association used to "
    "establish membership relationships between a platform and the "
    "node ports contained within that platform.")]
class CISCO_NodePortInPlatform: CIM_SystemDevice {

  [Override("GroupComponent"), Description (

```

Send documentation comments to mdsfeedback-doc@cisco.com

```

        "The platform that has contained node ports.")]
    CISCO_Platform REF GroupComponent;

    [Override("PartComponent"), Description (
        "The node ports contained in this platform.")]
    CISCO_FCPort REF PartComponent;
};

// CISCO_NodeInPlatform

[Association,
    Description (
        "CISCO_NodeInPlatform defines a SystemSpecificCollection "
        "in the context of a scoping system. Only nodes that are "
        "present in the platform database and also present in the "
        "Name Server are considered.")]

class CISCO_NodeInPlatform: CIM_HostedCollection {

    [Override ("Antecedent"), Description (
        "A platform hosts a collection of nodes.")]
    CISCO_Platform REF Antecedent;

    [Override ("Dependent"), Description (
        "The nodes that are hosted on a platform.")]
    CISCO_LogicalPortGroup REF Dependent;
};

// CISCO_PortControllerInPlatform

[Association,
    Description (
        "CISCO_PortControllerInPlatform defines a SystemSpecificCollection "
        "in the context of a scoping system. The node registered "
        "in the platform database must also be registered in the "
        "Name Server.")]
class CISCO_PortControllerInPlatform: CIM_SystemDevice {

    [Override ("GroupComponent"), Description (
        "A platform hosts a collection of devices.")]
    CISCO_Platform REF GroupComponent;

    [Override ("PartComponent"), Description (
        "The devices hosted on a platform.")]
    CISCO_PortController REF PartComponent;
};

// CISCO_PortControllerInFabric

[Association,
    Provider("FDMIProvider"),
    Description (
        "CISCO_PortControllerInFabric defines a SystemSpecificCollection "
        "in the context of a scoping system.")]
class CISCO_PortControllerInFabric: CIM_SystemDevice {

    [Override ("GroupComponent"), Description (
        "A platform hosts a collection of devices.")]
    CISCO_VSAN REF GroupComponent;

    [Override ("PartComponent"), Description (
        "The devices hosted on a platform.")]
    CISCO_PortController REF PartComponent;
};

```

Send documentation comments to mdsfeedback-doc@cisco.com

Cisco MOF Files for Cisco SAN-OS Release 2.x

This section includes the MOF files supported by Cisco SAN-OS Release 2.x. It includes the following topics:

- [Cisco Fabric MOF, page A-12](#)
- [Cisco Zone MOF, page A-14](#)

Cisco Fabric MOF

The Cisco Fabric MOF for Cisco SAN-OS Release 2.x provides extensions to the Fabric profile to manage VSANs, PortChannels, and other Cisco-specific entities within the fabric. See the “[Cisco MDS Extensions to the Switch and Fabric Profiles](#)” section on page 2-8.

```
[Version ("1.0.0"), Description (
    "cisco fabric and switch profile classes")]
class CISCO_ActiveConnection : CIM_ActiveConnection
{};

class CISCO_AdminDomain : CIM_AdminDomain
{};

[Version ( "2.7.1"), Description (
    "Capabilities and management of a Fibre Channel Port Device." ) ]
class CISCO_FCPort : CIM_FCPort {

    [Override ( "PortType"), Description (
        "The specific mode currently enabled for the Port. The "
        "values: \"N\" = Node Port, \"NL\" = Node Port supporting FC "
        "arbitrated loop, \"E\" = Expansion Port connecting fabric "
        "elements (for example, FC switches), \"F\" = Fabric "
        "(element) Port, \"FL\" = Fabric (element) Port supporting "
        "FC arbitrated loop, \"B\" = Bridge and \"G\" = Generic "
        "Port. PortTypes are defined in the ANSI X3 standards. "
        "When set to 1 (\"Other\"), the related property "
        "OtherPortType contains a string description of the port's "
        "type."),
        ValueMap { "0", "1", "10", "11", "12", "13", "14", "15", "16",
            "17", "18", "16004", "16010", "16011", "16012", "16000..65535"},
        Values { "Unknown", "Other", "N", "NL", "F/NL", "Nx", "E", "F",
            "FL", "B", "G", "PortChannel", "FCIP", "ISCSI-F", "ISCSI-N", "Vendor Reserved"}
    ]
    uint16 PortType;
};

class CISCO_Vsan : CIM_AdminDomain {
    [Override ( "NameFormat"), Description (
        "The NameFormat property identifies how the Name of the "
        "AdminDomain is generated, using the heuristic specified in "
        "the CIM V2 System Model spec. It assumes that the "
        "documented rules are traversed in order, to determine and "
        "assign a Name. The NameFormat Values list defines the "
        "precedence order for assigning the Name of the "
        "AdminDomain. \n"
        "\n"
        "\"FC\" has been deprecated and replaced by \"WWN\" to be "
        "consistent with the other ValueMaps."),
        ValueMap { "Other", "AS", "NAP", "NOC", "POP", "RNP", "IP",
            "IPX", "SNA", "Dial", "WAN", "LAN", "ISDN", "Frame Relay",
            "ATM", "E.164", "IB", "FC", "Policy Repository", "WWN", "ID with WWN"},
    ]
};
```

Send documentation comments to mdsfeedback-doc@cisco.com

```
        Values { "Other", "Autonomous System",
                "Network Access Provider", "Network Operations Center",
                "Point of Presence", "Regional Network Provider", "IP",
                "IPX", "SNA", "Dial", "WAN", "LAN", "ISDN", "Frame Relay",
                "ATM", "E.164", "Infiniband", "Fibre Channel",
                "Policy Repository", "Fibre Channel Worldwide Name", "Virtual SAN ID and
Worldwide Name"},
        ModelCorrespondence { "CIM_AdminDomain.Name" } ]
    string NameFormat;
};

class CISCO_Component : CIM_Component
{};

class CISCO_ComputerSystem : CIM_ComputerSystem
{};

class CISCO_ConnectivityCollection : CIM_ConnectivityCollection
{};

class CISCO_ConnectivityMemberOfCollection : CIM_MemberOfCollection
{};

class CISCO_ContainedDomain : CIM_ContainedDomain
{};

class CISCO_DeviceSAPImplementation : CIM_DeviceSAPImplementation
{};

class CISCO_FCPortStatistics : CIM_FCPortStatistics
{};

class CISCO_HostedAccessPoint : CIM_HostedAccessPoint
{};

class CISCO_HostedCollection : CIM_HostedCollection
{};

class CISCO_ProtocolEndPoint : CIM_ProtocolEndPoint
{};

class CISCO_PhysicalPackage : CIM_PhysicalPackage
{};

class CISCO_PhysicalElement : CIM_PhysicalElement
{};

class CISCO_Product : CIM_Product
{};

class CISCO_Realizes : CIM_Realizes
{};

class CISCO_SystemDevice : CIM_SystemDevice
{};

class CISCO_ComputerSystemPackage : CIM_ComputerSystemPackage
{};
class CISCO_ProductPhysicalComponent : CIM_ProductPhysicalComponent
{};
class CISCO_ElementStatisticalData : CIM_ElementStatisticalData
{};
class CISCO_LogicalPortGroup : CIM_LogicalPortGroup
{};
```

Send documentation comments to mdsfeedback-doc@cisco.com

```

class CISCO_LogicalModule : CIM_LogicalModule
{};

class CISCO_ModulePort : CIM_ModulePort
{};

class CISCO_EthernetPort : CIM_EthernetPort
{};

class CISCO_HostedDependency : CIM_HostedDependency
{};

class CISCO_LogicalIdentity : CIM_LogicalIdentity
{};

class CISCO_PhysicalComputerSystem : CISCO_ComputerSystem
{};

class CISCO_LogicalComputerSystem : CISCO_ComputerSystem
{};

class CISCO_FCNodeMemberOfCollection : CIM_MemberOfCollection
{};

```

Cisco Zone MOF

The Cisco Zone MOF for Cisco SAN-OS Release 2.x provides extensions to the zoning subprofiles. See the [“Cisco MDS Extensions to the Switch and Fabric Profiles”](#) section on page 2-8.

```

[Version ("1.0.0"), Description (
    "cisco zoneset class")]
class CISCO_ZoneSet : CIM_ZoneSet
{
};

class CISCO_Zone : CIM_Zone
{};

class CISCO_ZoneAlias : CIM_NamedAddressCollection
{};

class CISCO_ZoneMemberSettingData : CIM_ZoneMembershipSettingData
{};

class CISCO_ZoneService : CIM_ZoneService
{};

class CISCO_SystemSpecificCollection : CIM_SystemSpecificCollection
{};

class CISCO_ZoneMemberOfCollection : CIM_MemberOfCollection
{};

class CISCO_ElementSettingData : CIM_ElementSettingData
{};

class CISCO_HostedService : CIM_HostedService
{};

```

Send documentation comments to mdsfeedback-doc@cisco.com

Cisco Indications MOF

The Cisco Indications MOF provides extensions to the SMI-S standard indications to provide indications of link state changes. This MOF supports Cisco SAN-OS Release 2.0(1a) or later. See the “[FDMI Subprofile Extensions](#)” section on page 2-16.

```
[Version ("2.2.0")]
class CISCO_LinkStateChange : CISCO_AlertIndication
{
    [Description (
        "The desired state of the interface. The testing (3) state"
        "indicates that no operational packets can be passed. When a"
        "managed system initializes, all interfaces start with"
        "ifAdminStatus in the down(2) state. As a result of either"
        "explicit management action or per configuration information"
        "retained by the managed system, ifAdminStatus is then"
        "changed to either the up(1) or testing(3) states (or remains"
        "in the down(2) state)."),
    ValueMap {"1", "2", "3"},
    Values { "up", "down", "testing"}}
    uint32 ifAdminStatus;

    [Description (
        "The current operational state of the interface. "),
    ValueMap {"1", "2", "3", "4", "5", "6", "7"},
    Values { "up", "down", "testing", "unknown", "dormant",
        "notPresent", "lowerLayerDown"}}
    uint32 ifOperStatus;
    uint32 ifIndex;
};

class CISCO_LinkUp : CISCO_LinkStateChange
{};

class CISCO_LinkDown : CISCO_LinkStateChange
{};

class CISCO_MediaFRU : CISCO_AlertIndication
{
    uint32 PhysicalIndex;
    string PhysicalDescr;
    uint32 PhysicalVendorType_len;
    uint32 PhysicalContainedIn;
    [
        Description ("Entity Physical Class Type "),
        ValueMap {"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11" },
        Values {"ENT_OTHER", "UNKNOWN_ENTITY", "CHASSIS", "BACKPLANE", "CONTAINER",
            "POWERSUPPLY", "FAN", "SENSOR", "MODULE", "PORT", "STACK"}
    ]
    uint32 PhysicalClass;

    uint32 PhysicalParRelPos;
    string PhysicalName;
    string PhysicalHardwareRev;
    string PhysicalFirmwareRev;
    string PhysicalSoftwareRev;
    string PhysicalSerialNum;
    string PhysicalMfgName;
    string PhysicalModelName;
    string PhysicalAlias;
    string PhysicalAssetID;
    boolean PhysicalIsFRU;
};
```

Send documentation comments to mdsfeedback-doc@cisco.com

```

boolean Valid;

[
  Description ( "Module Admin Status Status"),
  ValueMap {"1", "2", "3", "4"},
  Values {"CEFC_PHYS_STATUS_OTHER ", "CEFC_PHYS_STATUS_SUPPORTED",
"CEFC_PHYS_STATUS_UNSUPPORTED", "CEFC_PHYS_STATUS_INCOMPATIBLE"}
]
uint16 PhysicalStatus;

string PhySecondSerialNum;
string PhyProductNumber;
string PhyPartRevision;
string PhyMfgDate;
string PhysicalCLEICode;
uint16 PhySramSize;
string PhysicalNameofSlot;
};

class CISCO_MediaFRUInserted : CISCO_MediaFRU
{};

class CISCO_MediaFRURemoved : CISCO_MediaFRU
{};

class CISCO_MediaFRUChanged: CISCO_AlertIndication
{
  uint32 PhysicalIndex;
  [Description (
    "Module Operational Status"),
  ValueMap {"1", "2", "4", "5", "6", "7", "8", "9", "10", "11", "12",
    "13", "14", "15", "16", "17", "18", "19", "20", "21"},
  Values {
"MOD_OPER_UNKNOWN", "MOD_OPER_OK", "MOD_OPER_DISABLED", "MOD_OPER_OKBUTDIAGFAILED",
    "MOD_OPER_BOOT", "MOD_OPER_SELFTEST", " MOD_OPER_FAILED", "MOD_OPER_MISSING",
    "MOD_OPER_MISMATCHWITHPARENT", "MOD_OPER_MISMATCHCONFIG",
"MOD_OPER_DIAGFAILED",
    "MOD_OPER_DORMANT" , " MOD_OPER_OUTOFSERVICEADMIN",
"MOD_OPER_OUTOFSERVICEENVTEMP",
    "MOD_OPER_POWEREDDOWN", "MOD_OPER_POWEREDUP", " MOD_OPER_POWERDENIED",
    "MOD_OPER_POWERCYCLED", "MD_OPER_OKBUTPOWEROVERWARNING", "
MOD_OPER_OKBUTPOWEROVERCRITICAL",
    "MOD_OPER_SYNCINPROGRESS" }
  ]
  uint16 ModuleOperStatus;

  [Description (
    "Module Admin Status Status"),
  ValueMap {"1", "2", "3", "4"},
  Values {"Admin Enabled", "Admin Disabled", "Admin Reset", "Admin Out of Service"}
  ]
  uint16 ModuleAdminStatus;
  [Description (
    "Module Admin Status Status"),
  ValueMap {"1", "2", "3", "4", "5"},
  Values {"UNKNOWN_RESET ", "POWERUP", "PARITYERROR",
"CLEARCONFIGRESET", "MANUALRESET"}
  ]
  uint16 ModuleResetReason;
  string ModuleResetReasonDescription;
  uint32 numPorts;
  uint32 boot_mode;
  uint8 isValid;
};

```

Send documentation comments to mdsfeedback-doc@cisco.com

```
    uint8  mod_state;
    uint8  mod_type;
    uint8  pad[2];
    uint32 mod_no;
    uint32 ModuleUpTime;
    uint32 numFcPorts;
};

class CISCO_VSANChanged: CISCO_AlertIndication
{};

class CISCO_ZoneSetAlert: CISCO_AlertIndication
{
    string ZoneSetName;
    uint32 VsanId;
};

class CISCO_EnvironmentalAlert: CISCO_AlertIndication
{
    string EnvAlertDescription;
    uint32 PhysicalIndex;
    uint32 OperationalStatus;
};

class CISCO_FanAlert: CISCO_EnvironmentalAlert
{};

class CISCO_PowerAlert: CISCO_EnvironmentalAlert
{
    uint32 FRUPowerAdminStatus;
    uint32 FRUCurrent;
};

class CISCO_TempAlert: CISCO_EnvironmentalAlert
{
    uint32 SensorValue;
    uint32 SensorThresholdValue;
    uint32 SensorThresholdIndex;
};

class CISCO_NameServerDatabaseChanged: CISCO_AlertIndication
{
};
```

Send documentation comments to mdsfeedback-doc@cisco.com



INDEX

A

- Access Point subprofile
 - description [2-4](#)
 - using Access Point subprofile classes (table) [2-4](#)

B

- Blade subprofile
 - description [2-3](#)
 - using Blade subprofile classes (table) [2-3, 3-4](#)

C

- CIM
 - client/server communications path (figure) [1-2](#)
 - configuring the server [3-1](#)
 - creating a certificate (procedure) [3-1](#)
 - description [1-1](#)
 - Operations over HTTP [1-3](#)
 - sample scenarios for using objects [3-1](#)
 - support by Cisco SAN-OS release (table) [1-5](#)
- CIM indications. See indications
- CIM server [1-4](#)
- Cisco extensions. See extensions
- Cisco FDMI MOF [A-5](#)
- Common Information Model. See CIM
- configuring the CIM server [3-1](#)

D

- discovery and performance data [3-3](#)
- DMTF website [1-1](#)

documentation

- additional publications [1-x](#)
- conventions [1-x](#)
- related documents [1-xiii](#)

E

- Enhanced Zoning and Enhanced Zoning Control subprofile description [2-6](#)
- extensions
 - Fabric profile [2-14](#)
 - FCIP [2-13](#)
 - iSCSI [2-14](#)
 - PortChannel [2-12](#)
 - TE port [2-10](#)
 - VSAN [2-8](#)
 - Zone Control subprofile [2-16](#)

F

- Fabric profile
 - description [2-4](#)
 - discovery and performance data [3-3](#)
 - extensions [2-14](#)
 - using Fabric profile classes (table) [2-4](#)
- FCIP extensions [2-13](#)
- FDMI Extensions [2-16](#)

H

- HTTP and HTTPS protocols (procedures) [3-2](#)

Send documentation comments to mdsfeedback-doc@cisco.com

I

indications

- Cisco-specific [2-17](#)

- description [2-16](#)

- iSCSI extensions [2-14](#)

M

Managed Object Format files. See MOFs

- managing zones [3-4](#)

- module, modeling [3-4](#)

MOFs

- Cisco Fabric for Release 2.x [A-12](#)

- Cisco Fabric for Release 3.x [A-1, A-5](#)

- Cisco Zone for Release 2.x [A-14](#)

- Cisco Zone for Release 3.x [A-4](#)

- description

N

- new and changed information (table) [1-vii](#)

O

Obtaining Documentation and Submitting a Service Request [1-xiii](#)

OpenSSL

- using to create certificate (procedure) [3-1](#)

P

- performance data, gathering [3-3](#)

PortChannels

- description [2-12](#)

- extensions [2-12](#)

- relationship to ports (figure) [2-13](#)

- port identifiers by port type (table) [2-15](#)

procedures

- configuring the CIM server [3-1](#)

- configuring zoning [3-4](#)

- gathering performance data [3-3](#)

- performing discovery [3-3](#)

profiles

- Fabric [2-4](#)

- Server [2-2](#)

- Switch [2-2](#)

S

- Server profile, description [2-2](#)

- Service Location Protocol. See SLP

- SLP, description [2-2](#)

SMI-S

- description [1-2](#)

- in multivendor SAN (figure) [1-2](#)

- managing SANs [2-1](#)

- support by Cisco SAN-OS release (table) [1-5](#)

- support in Cisco MDS 9000 Family [1-4](#)

- SNIA website [1-2](#)

- standards, supported [1-4, 1-5](#)

- Storage Management Initiative Specification. See SMI-S

subprofiles

- Access Point [2-4](#)

- Blade [2-3](#)

- Enhanced Zoning and Enhanced Zoning Control [2-6](#)

- Zone Control [2-5](#)

Switch profile

- description [2-2](#)

- discovery and performance data [3-3](#)

- using Switch profile classes (table) [2-3](#)

T

- TE port extensions [2-10](#)

Send documentation comments to mdsfeedback-doc@cisco.com

U

UML

- description [1-4](#)
- diagram for fabric partitioning (figure) [2-9](#)
- diagram for TE port partitioning (figure) [2-11](#)
- diagram for VSAN partitioning (figure) [2-12](#)
- diagram of UML example (figure) [1-4](#)
- FCIP and PortChannels example (figure) [2-13](#)

Unified Modeling Language

See UML

extrinsic methods (table) [3-4](#)

modeling [2-6, 2-8](#)

using the Zoning subprofile classes (table) [2-6, 2-8](#)

V

VSAN

- extensions [2-8](#)
- partitioning example (figure) [2-10](#)
- UML diagram for VSAN partitioning (figure) [2-12](#)

W

WBEM

- description [1-3](#)
- website [1-3](#)

Web-Based Enterprise Management Initiative. See WBEM

X

xmlCIM Encoding Specification [1-3](#)

Z

Zone Control subprofile

- description [2-5](#)
- extensions [2-16](#)

zoning

- configuring [3-4](#)
- deleting entities (table) [3-5](#)

Send documentation comments to mdsfeedback-doc@cisco.com