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

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

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 {
```

```
        [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
{};
```

```
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
```

```
{};

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 shoud 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."
```

```
            "* 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 "
```

```
        "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;
```

```
};

// CISCO_Platform
[Provider("FDMIProvider"),
    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[];
```

```
    [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.")]
```

```
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 SofwareInstalledOnPlatform 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
```

```
[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 (
```

```
        "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;
};
```

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

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"},
```

```
        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
{};
```

```
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
{};
```

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;
```

```
   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;
```

```
  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
{
};
```