**CIM Network Model White Paper**

**CIM Version 2.7**

**Version 1.1, December 2, 2003**

# Abstract

The DMTF Common Information Model (CIM) is a conceptual information model for describing computing and business entities in enterprise and Internet environments. It provides a consistent definition and structure of data, using object-oriented techniques. The CIM Schema establishes a common conceptual framework that describes the managed environment.

This document provides an overview of each of the model fragments composing the overall CIM 2.7 Network model and discusses its major concepts. As way of illustration, a simple scenario is presented showing how to use some of the main classes defined in the model.

# Notice

***DSP0152*** ***Status: Preliminary***

# Table of Contents

# 1 Introduction

This paper provides a general overview of the information model defined by the DMTF Network Working Group. The model is divided into a total of twenty-one (21) fragments; this facilitates model navigation and promotes clarity. While no explicit groupings exist, for the sake of discussion the various fragments may be grouped into categories as follows:

1. Relationship of networks to other parts of CIM (Overview)
2. Systems and Collections (Systems, Collections)
3. Logical Interconnection and Access (Pipes, ProtocolEndpoints)
4. Routing and Routing Protocols (Routing and Forwarding, Routing, OSPF, BGP)
5. L2 Protocols and Technologies (Switching & Bridging)
6. Quality of Service technologies (QoS, QoS Conditioning Services)
7. Management Protocols (SNMP)
8. Filtering and Buffers
9. Supporting definitions (Associations, Dependencies, Aggregations, MemberOfCollections)

## 1.1 Overview

The information model, CIM, characterizes a network as a type of administrative domain, which in itself may contain other networks or sub-networks. It may be defined according to criteria such as management domains, geography, etc. In order to operate the network infrastructure, networking services are required, and thus a number of "services" are defined in the model. Given the wide use of the word service, it is important to point out that within the context of the current information model, "service" refers to infrastructure services or offerings required by the individual network elements to exchange information or perform a function. Examples of these services are Routing (OSPF, BGP), Forwarding, and QoS.

Services are accessed or made available by Network Elements (NEs) via ProtocolEndpoints (subclass of ServiceAccessPoint). ProtocolEndpoints describe the characteristics that uniquely identify the address or location where the services are available. In addition to addressing and identification information, Endpoints may also contain configuration information that further defines the characteristics of the protocols. ProtocolEndpoint configuration may include information such as total bandwidth, available bandwidth, keepalive timers, retry intervals, etc.

Two ProtocolEndpoints may be associated via the ActiveConnection association to represent a potential or actual connection, and the exchange of information between the two entities. Typically, this association occurs at the same level within a communication protocol stack or application level protocol. This type of relationship is used when the connection needs to be represented, but the connection itself is not being managed. That is, the connection exists, but it does not have a state nor configuration information associated with it. In situations where the there is a need for a managed connection, the NetworkPipe association is used. All of these logical entities can be associated to the PhysicalElements, LogicalDevices, etc., that are being managed. Although this is not specified as part of the Network Model, it is an integral part of the overall schema needed to manage the network. The reader is encouraged to consult the CIM Core, Physical and Device Models for further details.

In addition to the general aspects of the Network Model, technology and protocol specifics are also addressed. Two commonly used routing protocols are defined in the model, OSPF and BGP. In both cases, the respective models focus on the configuration of the routing protocol. The protocols are characterized in terms of the services they offer, the protocol endpoints through which these services are made available, and protocol-specific configuration parameters.

In the switching arena, the schema covers switching and bridging functions, which includes Spanning Tree Protocol (STP) support. The Filtering Sub-Model defines the filtering criteria that may be defined and applied to device interfaces for the purpose of controlling the ingress of traffic into the network. The Buffers Sub-Model simply defines the availability of buffer space (BufferPools), which may be applicable to a device or interface.

Finally, the model has supporting sections defining the various associations, dependencies, and aggregations used throughout the model.

In conclusion, the Network Model covers both generic aspects required to represent connectivity between systems and relationships to the underlying physical components, as well as network technology and protocol specifics.

## 1.2 Background Reference Material

CIM Core and Common Models - Versions 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, and 2.7 - Downloadable from http://www.dmtf.org/standards/standard_cim.php

Common Information Model (CIM) Specification, V2.2, June 14, 1999 - Downloadable from http://www.dmtf.org/standards/documents/CIM/DSP0004.pdf

DMTF Specifications - Approved Errata - Downloadable from http://www.dmtf.org/standards/standard_cim.php

Unified Modeling Language (UML) from the Open Management Group (OMG) - Downloadable from http://www.omg.org/uml/

Internet Engineering Task Force (IETF) - MIBs and Work Group information at http://www.ietf.org

Core White Paper, DSP111, June 2003 – Downloadable from http://www.dmtf.org/standards/published_documents.php

## 1.3 Terminology

| Term | Definition |
|---|---|
| 802.1D | IEEE 802.1D defines the Bridge Protocol Data Unit / Spanning Tree |
| 802.1P | IEEE 802.1P defines the Priority bits in the L2 Ethernet frame |
| 802.1Q | IEEE 802.1Q defines the operation of Virtual LAN (VLAN) Bridges that permit the definition, operation and administration of VLAN topologies within a Bridged LAN infrastructure. The standardproposes a standardized format for MAC layer frame tagging (also known as encapsulation). Frame tagging uniquely assigns a user-defined ID to each frame. It also enables a switch to communicate VLAN membership information across multiple devices by frame tagging. |
| ATM | Asynchronous Transfer Mode |
| BGP | Border Gateway Protocol |
| EBGP | Exterior BGP |
| EMS | Element Management System |
| FDDI | Fiber Distributed Data Interface |
| IBGP | Interior BGP |
| IGP | Interior Gateway Protocol. A terminology used in MPLS for label hierarchy.<br>If a packet transfers through the following label path: R1, R2, R21, R22, R3, R4, then R1, R2, R3, R3 are level 1 LSR and R21, R22 are level 2. R2, and R21 are IGP neighbors. |
| IGP | Interior Gateway Protocol |
| L1 | Layer 1 |
| L2 | Layer 2 |
| L3 | Layer 3 |
| MPLS | Multiprotocol Label Switching |
| NE | Network Element |
| OSPF | Open Shortest Path First (Routing Protocol) |
| OSS | Operations Support System |
| QoS | Quality of Service |
| SNMP | Simple Network Management Protocol |
| SONET | Synchronous Optical Network |
| VC | Virtual Circuit |

# 2 The Network Model

## 2.1 Background and Assumptions

It is assumed that the reader has general knowledge of networking and is familiar with the concepts and terminology covered in the CIM Specification [1]. For general background information, the reader is referred to the CIM Concepts White Paper [2].

## 2.2 Conceptual Areas Addressed by the Model

As previously described, the Network Model is divided into multiple fragments addressing each of the areas listed below. The following sections provide an overview on each of these areas.

- Systems
- Collections
- Protocol Endpoints
- Pipes
- Filtering
- Buffers
- Routing and Forwarding
- OSPF
- BGP
- Switching and Bridging
- SNMP

## 2.3 Systems

This model fragment defines network systems and introduces the concept of Network into CIM. A Network is a type of administrative domain (**AdminDomain**), which may be composed of other networks or systems. In other words, there is a recursive relationship that allows the user to create nesting relationships between networks for the purpose of either defining management domains, or to reflect the underlying physical network, or both. For example, networA may be de-composed into multiple sub-networks (sub-network B, sub-network C, and sub-network D), and each of these sub-networks can be managed by a different organization within the company (administrative domains).

## 2.4 Collections

Collection is a class defined in the Core Model that is used to define groupings of ManagedElements. In this fragment of the Network model, five (5) classes are defined: ConnectivityCollection, IPConnectivitySubnet, LANConnectivitySegment, IPXConnectivityNetwork, and RangeOfIPAddreses.

When modeling a network, there is typically a need to show and monitor the communication points within a system and across systems (these are modeled as ProtocolEndpoints –see Section 2.5). These ProtocolEndpoints may need to be grouped (via **ConnectivityCollection**) to show the existence of multiple access/communication points within a system or an administrative domain. From a management perspective, this is a useful and powerful grouping that allows management applications to show these grouping graphically, perform operations on the group, and monitor the endpoints as a group[1]. Three of the remaining four classes are specializations of ConnectivityCollection (**IPConnectivitySubnet, IPXConnectivityNetwork and LANConnectivitySegment**). The first two classes deals exclusively with L3 endpoints (IP and IPX respectively), the latter deals with any type of LAN endpoint (L2), which may include technologies such as Ethernet, Token Ring, or FDDI.

Finally, **RangeOfIPAddresses** is used to represent a pool (group) of IP addresses. A potential use of this class is to represent a pool of addresses managed by a system, such as a DCHP server.

## 2.5 Protocol Endpoints

This model fragment defines communication points called ProtocolEndpoints for the following protocols:

- UDP
- TCP
- IP
- IPX
- BGP
- OSPF

In addition, this fragment defines two other classes representing a LAN and Switch endpoint, which provide the means to associate a communication endpoint with the underlying hardware.

The aforementioned **ProtocolEndpoint** types may be used to represent the communication path through the protocol stack within a system (via the BindsTo relationship), and also the communication between two or more endpoints of the identical type across two or more systems.[2]

[1] It is important to note that these are simply examples of the potential uses and benefits of having such groupings. The management application is required to provide the needed functionality to operate on the information represented through the model.
[2] This allows the representation of point-to-point, point-to-multi-point, or multi-point-to-multi-point communication.

## 2.6 Pipes

In Section 1.1, there was mention of a relationship called "ActiveConnection," and that it was used to tie together two ProtocolEndpoints that were exchanging data. A pipe serves a similar purpose, but with added semantics and a somewhat different focus. A **NetworkPipe** represents the connection in a network, through one or more ProtocolEndpoints. Through the NetworkPipe class, management of the connection is possible, since it contains state as well as data flow direction (uni-directional or bi-directional). Furthermore, it provides the means to represent network layering, by being able to define a client connection (pipe) with respect to a lower-level pipe or server connection. A typical example of this is the representation of an ATM Virtual Circuit (VC) riding on top of a Layer 1 (L1) SONET path.

The classes defined in this fragment are:

- NetworkPipe
- HostedNetworkPipe
- EndpointOfNetworkPipe
- NetworkPipeComposition

## 2.7 Filtering

This section of the model defines classes representing the filtering criteria used in Network Elements (NEs) to control the flow of traffic through the device. The classes defined here are:

- FilterList
- FilterEntryBase
- FilterEntry
- IPHeadersFilter
- 802.1Filter
- PreambleFilter

An NE may have been configured with a **FilterList**, which in turn is composed of one or more filter entries (subclasses of **FilterEntryBase**). A packet or frame (depending on the filter type) is subject to the filter criteria and disposition (forward, drop, etc.) specified within **FilterEntry**. Two specialized classes – **8021Filter and PreambleFilter** – are defined to handle the most commonly used fields for filtering within an IP header and an Ethernet header.

## 2.8 Buffers

This is a rather simple part of the network model in CIM 2.7, consisting of a single class defining a pool of buffers (**BufferPool**). It is used to represent a collection of one or more buffers, which may be associated directly with an interface or globally with the NE.

## 2.9 Routing and Forwarding

The purpose of the Routing and Forwarding model is to provide the means for a management system to manage (configure and view) the routing criteria (policies) within a system. The management station, via the **RoutingPolicy**, may configure such policies, and thus manipulate how the router handles traffic (accept, accept but modify, etc.). In addition, a management station maintains information describing the Endpoints used to calculate the routes, and those to which packets are being forwarded (**RouteCalculationService, ForwardingService**). Furthermore, these two classes also contain information about the routing protocol in use, as well as the protocols being forwarded.

## 2.10 Routes

The Route model fragment is comprised of three classes: **NextHopRoute, NextHopIPRoute**, and **AdministrativeDistance**. The purpose of this section of the model is to represent the routing table configuration in a device. These class definitions may be used to model the data needed to manage routing tables in either an end-station or an NE. Typically, the end-station is configured with a small set of static routes, but the router is running one or more routing protocols, with most of the routes being dynamic. The indication as to whether the route is static or dynamic is conveyed through a property in NextHopRoute.

An entry in a routing table describes the next hop a packet should take as well as the interface used to forward the packet. This is captured in the model via the associations **AssociatedNextHop** and **RouteUsesEndpoint**. The relationship between a route entry to the end-station, or NE in which it is configured, is done by establishing the **HostedRoute** relationship (association) between NextHopRoute/NextHopIPRoute and the device (system).

## 2.11 OSPF

The Open Shortest Path First (OSPF) Sub-Model addresses the configuration of the OSPF routing protocol. Because OSPF is a hierarchical routing protocol, the model addresses both the inter-area and intra-area level configurations. Please refer to the paper, "The Network OSPF Sub-Model White Paper," DSP0160, for complete documentation [3].

## 2.12 BGP

The Border Gateway Protocol (BGP) Sub-Model is being revised in CIM V2.9, to simplify and clarify the class hierarchy. Currently, the model addresses:

- AutonomousSystem
- BGPPeerGroup, BGPCluster and Confederation

- BGPService
- BGPAttributes and BGPPathAttributes
- FilterEntry, FilterList, and RoutingPolicy
- BGPProtocolEndpoint
- AdministrativeDistance

By and large, these concepts will be maintained in CIM V2.9, although simplified. And, aspects of how the protocol works will be removed – since these are not manageable. The revised design will also make use of policy.

## 2.13 QoS

The QoS Sub-Model is heavily influenced by the CIM-based IETF (Internet Engineering Task Force) "QoS Device Model", which is defined by the IETF Policy Framework Working Group. Many of the contributors to that document worked on this model as well. Please refer to the Internet-Draft, "Information Model for Describing Network Device QoS Datapath Mechanisms," for complete documentation [4].

## 2.14 Switching and Bridging

Switching and Bridging captures the L2 configuration and capabilities of an Ethernet switch in the form of **SwitchServices**. A switch provides a SwitchService, which includes transparent bridging (**TransparentBridgingService**). TransparentBridgingServices forward frames (one hop at a time) transparently, to an end-station. The forwarding decision is done by inspection of the forwarding tables, which may be manually configured (**StaticForwardingEntry**) or learned by the switch (**DynamicForwardingEntry**). Another aspect captured in the model is the configuration of the Spanning Tree Algorithm (STP), which is used by the switches to discover the network topology and prevent loops in the network. In addition, this fragment of the CIM network model provides performance measurements associated with the switch endpoint (SwitchPortStatistics), as well as the services(SwitchPortSourceRoutingStatistics, SwitchPortTransparentBridgingStatistics, and SwitchPortSpanningTreeStatistics).

Classes defined in this model are:

- SwitchPort
- SourceRoutingService
- SwitchService
- TransparentBridgingService
- SpanningTreeService
- StaticForwardingEntry
- DynamicForwardingEntry
- SwitchPortSourceRoutingStatistics
- SwitchPortTransparentBridgingStatistics
- SwitchPortSpanningTreeStatistics

- SwitchPortStatistics

## 2.15 SNMP

This fragment captures the basics of the SNMP v1 and v2 protocols, and their configuration. It represents SNMP as a service (for example, running on a NE[3]), with options to configure community strings and the associated access level, as well as SNMP Trap destinations. Access to the SNMP services is represented via a protocol endpoint. The set of classes defined in this model are:

- SNMP Service
- SNMPCommunityStrings
- SNMPTrapTarget

## 2.16 VLANs

The CIM 2.7 VLAN Sub-Model was focused on the representation of VLANs within a switching system. And while it does make the VLANs visible via AccessPoints, it does not capture many important concepts necessary to provide an adequate view and facilitate management of a VLAN. As a result, this sub-model is being re-worked in CIM 2.8 to address many of the omitted areas. It is recommended that potential users of this sub-model wait until the updated VLAN work is available in CIM 2.8.

Major areas covered in the soon to be released VLAN Sub-Model include:

- Expanded view of VLAN to cover network devices, as well as end-stations
- Association of Switch endpoints to end-station endpoints
- Port modes (trunk, access, etc)
- Statistics

[3] This is applicable to any system, however the NE is used as an example here

# 3 Relationships to Other Standards and Specifications

## 3.1 Overlapping Standards and Specifications

### 3.1.1 IETF

Various areas of overlap exist between the CIM Network Model and work at the IETF. In general, it is the intention of this group to align its work with existing IETF MIBs [5] whenever possible, since their primary emphasis is on device instrumentation and hence there are no conflicts with higher level constructs in this model. Related areas with IETF include:

- OSPF (RFC 1850 - OSPF Version 2 Management Information Base)
- BGP (RFC 1657 - Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIv2)
- QoS (RFC 3289 - Management Information Base for the Differentiated Services Architecture)
- Information Model for Describing Network Device QoS Datapath Mechanisms http://www.ietf.org/internet-drafts/draft-ietf-policy-qos-device-info-model-10.txt
- Switching and Bridging (RFC 1493 - Definitions of Managed Objects for Bridges)
- SNMP (RFC 1157 - Simple Network Management Protocol (SNMP))

IETF MIBS can be found at http://www.ietf.org/.

### 3.1.2 ITU

The Network model (Systems and Pipes fragments) incorporate the following concepts:

- Networks, Pipes and Termination Points (ITU-T Recommendation M.3100 [6])

# 4 Network Model Use Case

Previous sections described the various model fragments. This section provides an overview of how to use various components of the model to represent and manage an IP network. In our example, we identify what needs to be modeled and we have the following[4]:

- Network as a whole
- Sub-networks
- Network Topology[5]
- Communication points
- Network Element and Components
- Services

A network is a type of AdminDomain that is composed of multiple ManagedSystemElements. These elements are interconnected and exchange information with each other. In some situations, it is necessary or desirable to decompose a network into sub-networks. For example, a network may be composed of multiple technologies (SONET, IP, etc.) and the L1 and L3 networks may be managed by distinct entities. As shown in the diagram below, this is accomplished via the ContainedDomain association.

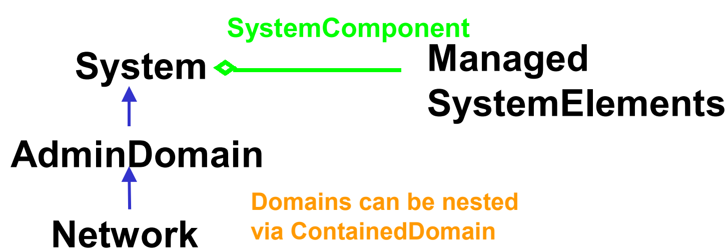


**Figure 1: Network and AdminDomain Classes**

Access points into the network are represented as a collection of ProtocolEndpoints. This grouping is done via a Collection class called ConnectivityCollection and linked to the network via the HostedCollection association.

[4] This does not represent a comprehensive list of areas to be modeled. This is simply used for illustration purposes.
[5] Network topology is an important aspect of network modeling and this is not part of CIM v2.7, however, this is introduced in CIM v2.8 (See Future Work in Section 5).

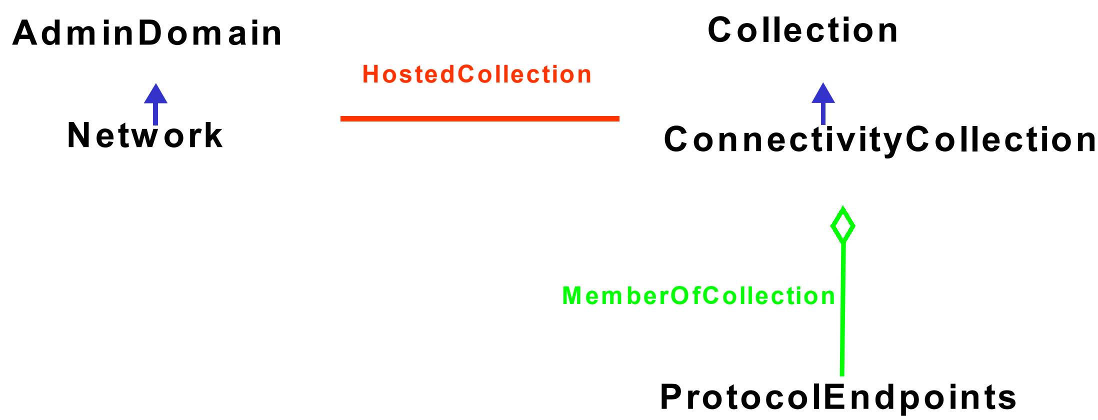


**Figure 2: Network access points (ProtocolEndpoints)**

At the network or sub-network level, there are two methods by which connections may be represented. They are an ActiveConnection association between ProtocolEndpoints, or instantiation of the NetworkPipe class between ProtocolEndpoints hosted in the Network itself. Aside from the distinction made above, the other major distinction is that ActiveConnection simply indicates that the associated ProtocolEndpoints have the potential to communicate/exchange data, whereas a NetworkPipe represents the connection, and as such has state and holds configuration information.

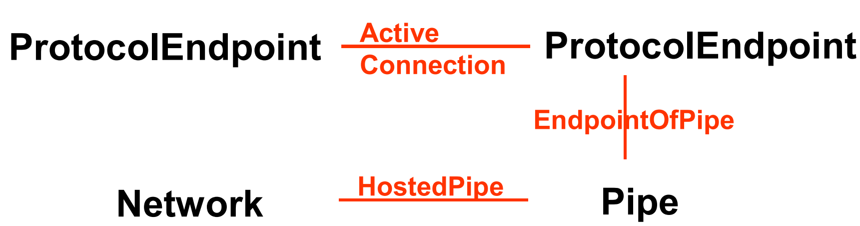


**Figure 3: Representation of network connections**

Eventually, a sub-network is de-composed into a series of ManagedSystemElements, each of which hosts one or more ProtocolEndpoints. Communication occurs via the addresses and interfaces provided by the ProtocolEndpoints. In addition to this, other aspects of the device at both the logical and physical level need to be modeled. The diagram below (Figure 4) illustrates this.

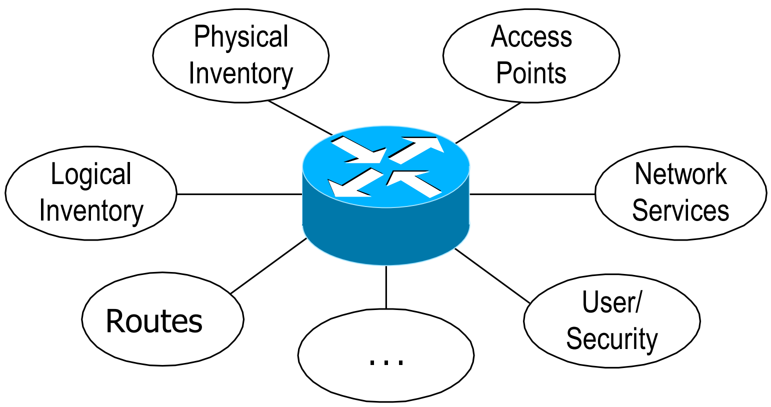


**Figure 4: Logical and Physical Device Modeling Areas**

Ultimately, objects modeling logical network aspects must be related to the objects representing the underlying physical components. Following the ProtocolEndpoint example, looking at Figure 5 we can identify the relationship from a ProtocolEndpoint (LANEndpoint) to the NetworkPort, and from there to the physical Card in the device.

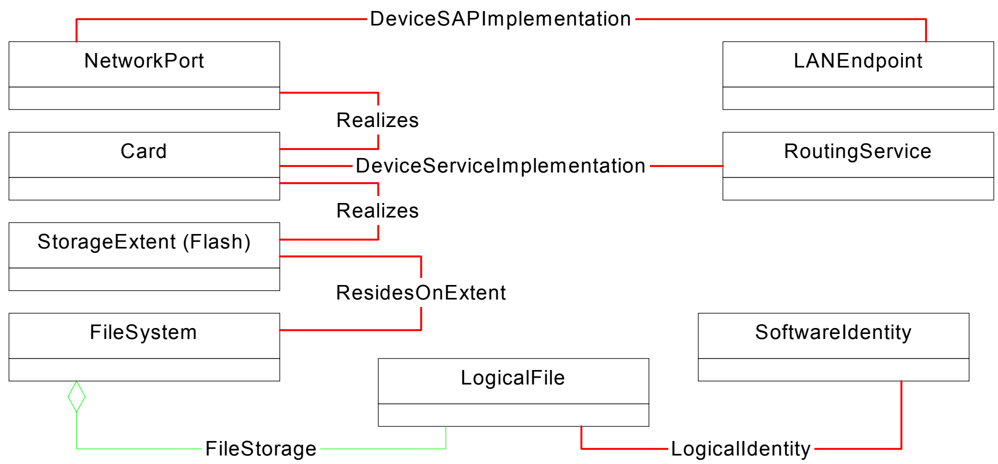


**Figure 5: Physical to Logical Relationships**

ProtocolEndpoints are not only used for modeling external communication. Modeling communication points across layers in the protocol stack, within a device, also makes use of ProtocolEndpoints.

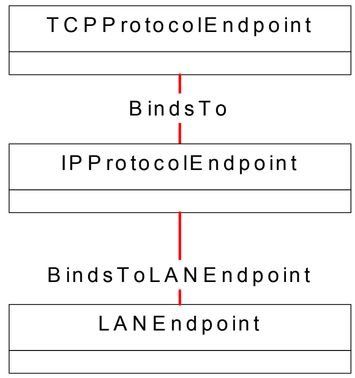


**Figure 6: Local ProtocolEndpoint relationship**

# 5 Future Work

In the upcoming version of the model (v2.8), the plan is to include support in the following areas:

Virtual Local Area Networks (VLANs)
Multi-Protocol Label Switching (MPLS)
Network Topology Representation
IP Secure (IP Sec)

For CIM v2.9, proposed work items include:

- BGP model cleanup
- Support for various Link Aggregation mechanisms
- QoS model cleanup and enhancements to align with the latest Internet-Drafts in the IETF
- Definition of guidelines for the creation and configuration of ProtocolEndpoints
- Definition of a model for Virtual Private Networks (VPN)
- Further enhancements to VLAN Sub-Model
- Internet Small Computer System Interface (iSCSI)

# Appendix A – Change History

| | | |
|---|---|---|
| Version 0.9 | June 19, 2003 | Initial Draft |
| Version 1.0 | August 26, 2003 | Preliminary |
| Version 1.1 | December 2, 2003 | Editorial changes and clean up |

# Appendix B – References

[1] Common Information Model (CIM) Specification, V2.2, June 14, 1999 - Downloadable from http://www.dmtf.org/spec/cims.html

[2] CIM Concepts White Paper, DSP0110 – Downloadable from http://www.dmtf.org/standards/published_documents.php

[3] CIM Network OSPF Sub-Model, DSP0160 – Downloadable from http://www.dmtf.org/standards/published_documents.php

[4] IETF Internet-Draft, Information Model for Describing Network Device QoS Datapath Mechanisms, http://www.ietf.org/internet-drafts/draft-ietf-policy-qos-device-info-model-10.txt

[5] Various IETF standard protocols and SNMP MIBs – Downloadable from http://www.ietf.org/rfc.html

[6] ITU-T Generic Network Information Model, M.3100 – Obtainable from http://www.itu.int/home/index.html

# Appendix C – Extending the Model

The Network Model covers general concepts and is vendor independent. Therefore, it may be necessary to create specialized classes in situations where proprietary implementations need to be supported. Furthermore, areas not currently covered by the model may be addressed by proprietary extensions, but also *should* be submitted to the DMTF for discussion and inclusion into the next CIM release.