

Physical Topology (PTOP0) Working Group
Internet-Draft

Kendall S. Jones
Bay Networks, Inc.

Expires in 6 months

March 25, 1997

Physical Topology Framework

[<draft-kjones-ptopo-framework-00.txt>](#)

Ken Jones

[1.0](#) Status of this Memo

This document is a submission to the IETF Physical Topology (PTOP0) Working Group. Comments are solicited and should be addressed to the working group mailing list (ptopo@3com.com) or to the author.

This document is an Internet-Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as a Work in Progress.

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow directories on ds.internic.net (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim). Distribution of this memo is unlimited.

Table of Contents

1. Status of this Memo	1
2. Abstract	2
3. Introduction	2
4. Background and Concepts	2
4.1. A word on physical and logical topology	3
4.2. Goals of a physical topology framework	4
5. Glossary of Terms	4
6. The Framework	5
7. Security Considerations	16
8. Open Issues	16
9. Acknowledgments	16
10. References	16
11. Author's Address	16

2. Abstract

This memo defines a framework for the collection of physical topology information. Physical topology is defined as the physical interconnection of communication ports between communication devices. The framework allows for topology determination within and across a broad spectrum of devices. It establishes a set of guidelines for topology mechanisms used to distribute and collect topology information, and describes the behavior of a management station required to collect topology information across a potentially large and diverse set of communication devices. A companion memo will provide an experimental portion of the Management Information Base (MIB) which is derived from this framework and allows the management workstation to gather physical topology information from the devices in the network.

3. Introduction

The purpose of this memo is to provide a framework for discussing physical topology mechanisms, the role of management agents in collecting and reporting topology data (via a MIB), and the methods by which a management application can collect physical topology information across arbitrarily large and complex networks. This memo defines a set of common terminology and presents the framework. It is expected that a physical topology MIB will be defined in a companion memo, and possibly several companion memos will be defined describing specific topology mechanisms for different underlying communication technologies (e.g shared media Ethernet, Token Ring, switched networks, etc.).

4. Background and Concepts

The need to understand the physical relationships between devices in a network has always been an important aspect of network management. With the increasing complexity of multi-segment and multi-media hubs and switching devices, and management applications that want to identify hot spots in the network, isolate complex faults (such as configuration problems) to the physical port, and manage virtual networks (VLANs), the need for accurate, timely, and system wide physical topology is becoming more and more critical to maintaining a mission critical network.

Most of today's management platforms do a good job at discovering the logical topology at the network layer but do not help much in understanding the actual physical interconnection. This is due to a lack of standard topology mechanisms at the physical layer to collect the physical topology information as well as a standard MIB to return topology information to the management workstation.

Standard topology mechanisms exist for certain media types (such as FDDI) and proprietary mechanisms exist for other media such as shared media Ethernet, switched Ethernet, and Token Ring. While standardizing these or other mechanisms for each of these technologies could be a painstaking task, standardizing a common MIB and a topology framework that allows co-existence of multiple topology mechanisms to populate these MIBs can go a long way toward achieving the goal of providing network-wide physical topology information at the network management workstation. The topology framework should specify the core requirements for topology mechanisms in order to provide the data necessary to populate the common topology MIB. These requirements form a set of guidelines to direct the eventual standardization of a set of topology mechanisms for the various communication media. In the meantime, the common MIB will allow creation of physical topology databases which will allow applications to provide value added services on top of this rich set of data.

4.1. A word on physical and logical topology

Before launching into the physical topology framework it may be helpful to review the differences between physical and logical topology. Physical topology describes how physical devices in a network are interconnected. Logical topology indicates how devices are related based on some system level attribute. Often this is based on the OSI communication layer. For instance, network layer topology, or layer 3 topology, uses network layer address hierarchies to construct a topological relationship. Management platforms typically present Layer-3-based topology. This is done by SdiscoveringT the IP addresses on a network and then grouping these logically by the subnet portion of the address. Other logical views of topology include layer-2 based views based on vlan membership, and higher layer views such as workgroup membership. Most higher-layer topology views use network address or user name to represent members of the topology space.

Physical topology represents the topology model for layer 1 of the OSI stack - the physical layer. Physical topology consists of identifying the devices on the network and how they are physically interconnected. By definition of this document, physical topology does not imply a physical relationship between ports on the same device. Other means exist for determining these relationships (e.g. entity MIB).

While the framework presented here is focused on physical topology, it may well be that the topology mechanisms and MIBs described could be extended to include logical topology information as well. That is not a focus of this memo at this time, although some consideration may be given to the framework itself and some notes may be included within this memo. They will be clearly identified as work for further study.

Expires September 1997

[Page 3]

4.2 Goals of a physical topology framework

The following goals can be described for the physical topology framework.

- allow for a common MIB that represents the physical topology information available to any one agent in the network. This MIB should be universal across all types of agents. The MIB should provide enough topology information to allow a management workstation to navigate across agents to build a complete physical topology for an arbitrarily large network.
- provide a set of requirements for topology mechanisms that can populate the agent MIBs. These requirements should allow for many different standard and non-standard topology mechanisms to be used within a given network, with clear rules describing how these mechanisms should interact.
- specify, where appropriate, topology mechanisms for specific media types. Where standard or industry-accepted mechanisms exist, indicate how these mechanisms can be used to populate the topology MIBs.
- provide a description of the management station procedures to navigate among topology agents and gather topology information

5. Glossary of Terms

The following glossary describes the important terms used in this memo. Many of these terms are taken from the Entity MIB.

Device = Managed System

This term is used to define the physical devices within the topology model. Physical topology consists of understanding the physical links between ports on one device and ports on another. The managed system is referred to in the Entity MIB as follows:

"...there is some 'overall' physical entity which houses the sum of the things managed by that one agent, i.e., there are multiple 'logical' entities within a single physical entity. Sometimes, the overall physical entity contains multiple (smaller) physical entities and each logical entity is associated with a particular physical entity. Sometimes, the overall physical entity is a 'compound' of multiple physical entities (e.g., a stack of stackable hubs)."

Device Identifier

some unique way of identifying a device is encouraged. This can be via the address or addresses of the agent(s) within the system or through some sort of 'device id' or 'chassis id' that uniquely identifies this device, such as a serial number. A MAC address may also serve as a device ID. As described later, for devices that contain multiple agents, it may be useful to create a device ID that concatenates an agent identifier that is unique for each agent and a managed system identifier that is known by all agents.

Expires September 1997

[Page 4]

Physical Port

A physical port referred to in this memo corresponds to a physical entity with a PhysicalClass value of port(10) as defined in the Entity MIB.

Port Number

The port number referred to in this memo corresponds to the value of entPhysicalParentRelPos for the Physical Port as defined in the Entity MIB. As stated in the Entity MIB: "This value should match any external labeling of the physical component if possible." and "The agent should retain parent-relative position values across reboots...".

Slot Number

The slot number referred to in this memo corresponds to the value of entPhysicalParentRelPos for a Physical Entity with a PhysicalClass value of container(5), as defined in the Entity MIB. Often times when this memo refers to a Physical Port, there is an implied slot number associated with that port.

Cloud

A cloud indicates that insufficient information is known about a portion of the topology to completely infer the devices that make up that portion of the topology.

Topology Mechanism

topology mechanism is a means, possibly requiring some sort of protocol, by which devices determine topology information. Note that the definition does not preclude a mechanism that requires users to manually enter neighbor identification information into each device (similar to static routing).

Local Physical Topology

For a device on the network, local physical topology is defined as the interconnection of the ports on this device to other devices on the network. Local physical topology specifically does not include the interconnection of ports between other devices on the network.

6. The Framework

First, a concise definition of physical topology is provided followed by a description of the components of the framework. Each of these components and their interactions with other components is described in detail in subsequent sections.

6.1 Physical topology

Physical topology, for the purposes of this document, is defined as the interconnection of physical ports between devices. It does not represent the geographic location of devices (e.g. device A is on the second floor of building 1). It also does not concern itself with how physical ports are interconnected within a device. Devices are treated as black boxes, although they should be uniquely identified by a 'device id' as defined in the terminology section above.

The goal of representing the physical topology is to map as closely as possible to the actual physical wiring used to interconnect devices in a network. For managed devices which are interconnected using point to point wiring, the representation can be extremely accurate. If non-managed devices are deployed or wiring is not point-to-point (such as Ethernet vampire taps), then the physical topology representation will not be as accurate. It is important to realize that it may be impossible to create a completely accurate representation of the physical wiring for most networks. The topology framework must deal with uncertainties and inconsistencies and represent the physical topology as accurately as possible.

6.2 Components of the Framework

The following components make up the framework for physical topology

Network Devices - these are the devices, along with their physical connectivity, that make up the physical topology. Some of these devices (but maybe not all) provided management agents that report their local physical topology information to a manager via the physical topology MIB. These devices include communication infrastructure devices, such as hubs, switches, and routers, as well as 'leaf' devices, such as workstations, printers, and servers. Generally, user data passes through infrastructure devices while leaf devices are sources and sinks of data. Both types of devices may implement the physical topology MIB, although it will be shown later that implementation within leaf devices is much less critical.

Topology Mechanism - the topology mechanism allows the management agents to collect the information required to populate the topology MIB. Many instances of a particular topology mechanism may be in use on a given network, and many different mechanisms may be employed. In some cases, multiple mechanisms may overlap across part of the physical topology. Agents may need to be configured so that they know which mechanism(s) are in use on any given portion of the network. Most topology mechanisms need to be bounded to a subset of the network to contain their impact on the network and quantize the effort of the manager to collect topology information. Most topology mechanisms are naturally bounded by the media on which they run (e.g. FDDI topology mechanism) or by routers in the network that intentionally block these mechanisms from crossing into other parts of the network.

Local Topology Data and the Local Topology MIB - each managed device collects physical topology information from the network, based on the topology mechanisms it is configured to use. The data represents this agent's view of the physical network and may or may not provide enough data to understand the local physical topology surrounding this agent. It may be necessary to gather local topology data from a number of agents in order to completely understand the local physical topology. Part of the local topology data collected must include the identification of other local agents which may contain additional topology information. The definition of 'local' varies based on the topology mechanism or mechanisms being used.

Manager process - a manager is responsible for querying management agents to obtain their local topology information and their knowledge of additional local agents. The manager may need to query some or all local agents to build an accurate view of the physical network.

6.3 Topology Mechanism

A topology mechanism is a means, possibly requiring some sort of protocol, by which devices determine topology information. The formal requirement is that the mechanism should provide sufficient information such that a manager can accurately determine the physical topology of a set of devices by querying all of those devices for their local view of the topology. In other words, the mechanism may not be robust enough to allow the manager to accurately determine any part of the network by querying a single agent - rather it may need to query all agents to understand the topology. This is discussed in more detail in the section below on topology data.

Topology mechanisms can be active or passive. Active mechanisms require a device to send and receive topology protocol packets. These packets provide the device ID of the source of the packet and may also indicate out which port the packet was transmitted. When receiving these packets, devices typically are required to identify on which port that packet was received.

Expires September 1997

[Page 7]

Passive mechanisms take advantage of data on the network to populate the topology MIB. As described below, by maintaining a list of device IDs seen on each port of all devices in a network, it is possible to construct an accurate physical topology of the network.

A device can act as a boundary device or an intermediate point of a topology mechanism. If it is a boundary device, it will prevent active topology mechanisms from passing through to other ports on that device. Routers are often boundary devices for active topology mechanisms. Boundary devices serve a critical role in containing topology mechanisms within a set of devices. This limits the size of topology tables maintained by the agent, reduces calculations required by managers, and prevents proliferating topology protocols across the network.

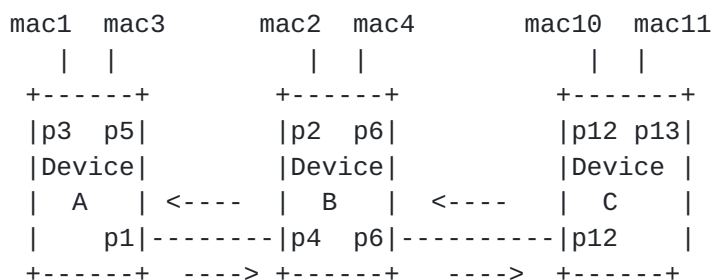
It is possible to have ports that support more than one topology mechanism. In general this simply allows the port to collect more robust topology information which should allow the manager to create more accurate physical topologies. If the set of devices that support each mechanism has only minimal overlap, the manager may need to work with the union of the two sets which could increase the processing requirement substantially. the impact on the manager is described in a later section.

6.4 Local Topology Data

This section describes what is required for local topology data using a simple network example. First, it describes the data that an agent would provide to present a complete view of its local physical topology. Next, several examples are provided to show how agents may provide partial data that must be collated by a manager to determine the actual physical topology. Next, some degenerate cases are described where only partial topology data is available. Further examples of ring-based networks are provided. The final subsection describes the MIB requirements.

6.4.1 Network Example

Consider the simple network shown below:



For this network, the physical topology can be described by the following information:

```
device_id = A, port 1 connects to device_id = B, port = 4
device_id = A, Port 3 connects to device_id = mac1, port = 1
device_id = A, Port 5 connects to device_id = mac3, port = 1
device_id = B, Port 8 connects to device_id = C, port = 11
device_id = B, Port 2 connects to device_id = mac2, port = 1
device_id = B, Port 6 connects to device_id = mac4, port = 1
device_id = C, Port 12 connects to device_id = mac10, port = 1
device_id = C, Port 13 connects to device_id = mac11, port = 1
```

It is easy to see that this information provides an accurate view of the physical topology for this network. This is the sort of information that might be available from a management workstation or mid-level manager after it has collected and processed topology data from the agents in the network. Let's examine what information is required to be maintained by each individual agent in this network. For now, assume that each device in the diagram above includes an agent. This information would be known by the agent by employing one or more topology mechanisms as described above.

6.4.2 Agent knows complete local topology

If the topology mechanism is robust enough, then it is possible that each agent knows the device_id and port of attachment for any devices connected to any of its ports. In this case, for the example above, each agent returns the following information.

```
Agent for device_id = A
  Port 1 connects to device_id = B, port = 4
  Port 3 connects to device_id = mac1, port = 1
  Port 5 connects to device_id = mac3, port = 1
Agent for device_id = B
  Port 8 connects to device_id = C, port = 11
  Port 2 connects to device_id = mac2, port = 1
  Port 6 connects to device_id = mac4, port = 1
Agent for device_id = C
  Port 11 connects to device_id = B, port = 4
  Port 12 connects to device_id = mac10, port = 1
  Port 13 connects to device_id = mac11, port = 1
Agent for device_id = mac1
  Port 1 connects to device_id = A, port 3
Agent for device_id = mac2
  Port 1 connects to device_id = B, port 2
Agent for device_id = mac3
  Port 1 connects to device_id = A, port 5
Agent for device_id = mac4
  Port 1 connects to device_id = B, port 6
Agent for device_id = mac10
  Port 1 connects to device_id = C, port 12
Agent for device_id = mac11
  Port 1 connects to device_id = C, port 13
```

Expires September 1997

[Page 9]

This level of information is generally easier for store and forward type devices to ascertain. For example, a topology mechanism might require a device to send a packet out each port that contains its device ID and the port number the packet is sent out. The receiving device obtains the packet from the incoming port and from it has the data required to populate a row of the table described above. For devices that normally do not send unique packets out specific ports or pass packets through the device transparently, such as shared media hubs, creation of the table above could be quite expensive. Fortunately, alternatives exist.

6.4.3 Agent knows partial local topology

There is a lot of redundant returned by the agents in the example above. Agents could maintain significantly less information and still allow a manager to determine the physical topology. Reducing the information that an agent must report can significantly simplify the underlying topology mechanism used to collect the topology data. For instance, if each agent reports the remotely-connected device_id for each of its ports, it does not have to report the port number for that remotely-connected device. That information can be filled in once the agent on the remotely-connected device is queried. In this case, each agent returns the following information:

```
Agent for device_id = A
  Port 1 connects to device_id = B
  Port 3 connects to device_id = mac1
  Port 5 connects to device_id = mac3
Agent for device_id = B
  Port 8 connects to device_id = C
  Port 2 connects to device_id = mac2
  Port 6 connects to device_id = mac4
Agent for device_id = C
  Port 11 connects to device_id = B
  Port 12 connects to device_id = mac10
  Port 13 connects to device_id = mac11
Agent for device_id = mac1
  Port 1 connects to device_id = A
Agent for device_id = mac2
  Port 1 connects to device_id = B
Agent for device_id = mac3
  Port 1 connects to device_id = A
Agent for device_id = mac4
  Port 1 connects to device_id = B
Agent for device_id = mac10
  Port 1 connects to device_id = C
Agent for device_id = mac11
  Port 1 connects to device_id = C
```

Sometimes adding redundant data can simplify the implementation of the topology mechanism and provide for more robust topology reporting (see next section). For instance, in some devices, topology information received on one port may pass through the device and be seen by other downstream devices. Going back to the diagram, Device C might report that both Device B and Device A are connected to port **11. the following information would allow a network management** station to accurately determine the physical topology of the network:

```
device_id = A
    Port 1 has seen the following device_ids: B, mac2, mac4, C,
        mac10, mac11
    Port 3 has seen the following device_ids: mac1
    Port 5 has seen the following device_ids: mac3
device_id = B
    Port 8 has seen the following device_ids: C, mac10, mac11
    Port 2 has seen the following device_ids: mac2
    Port 6 has seen the following device_ids: mac4
device_id = C
    Port 11 has seen the following device_ids: B, mac2, mac4, A,
        mac1, mac3
    Port 12 has seen the following device_ids: mac10
    Port 13 has seen the following device_ids: mac11
device_id = mac1
    Port 1 has seen the following device_ids: any one device_id
device_id = mac2
    Port 1 has seen the following device_ids: any one device_id
device_id = mac3
    Port 1 has seen the following device_ids: any one device_id
device_id = mac4
    Port 1 has seen the following device_ids: any one device_id
device_id = mac10
    Port 1 has seen the following device_ids: any one device_id
device_id = mac11
    Port 1 has seen the following device_ids: any one device_id
```

>From this discussion, it should be clear that agents must be given the flexibility to represent topology data a number of different ways. Specifically, each device reports on its local view of topology information based on the topology mechanism(s) employed by that agent and its surrounding agents. In general, agents report on as much 'downstream data' as they can for each port on the device.

In addition to topology data, it is important that agents also report on other downstream agents they have detected. This allows the management workstation to query each agent for its understanding of topology and assemble the total representation of the physical topology. For this reason, it is recommended that device_ids include the address of the agent for the device. This accomplishes both goals of identifying devices and locating agents in the network.

6.4.4 Degenerate Cases

The example above assumed that each device contained an agent that supported the PTOPO MIB and all agents were discovered and interrogated by the NMS application. There are a number of degenerate topology cases that need to be considered to determine how they impact the ability of the manager to determine the physical topology. These cases are described below:

6.4.4.1 Non-managed (or non-PTOPO) devices

If a device in the network does not support a PTOPO agent, it may still be possible to infer its existence from the other agents. For instance, in the example above, if Device B did not report any topology data, but transparently passed any topology mechanism protocols through it, then Device A would report that its port 1 was connected to Device C and Device C would report that its port 11 was connected to Device A. However, both Device A and Device C would report seeing mac2 and mac4 on ports 1 and 11 respectively. From this, the manager would infer that one or more non-managed devices must exist between A and B, and that mac2 and mac4 are connected to the device. The existence of an unmanaged device is referred to as a 'cloud', since it is often depicted that way on a physical topology diagram. An accurate view of the devices connected to the cloud can be determined, but the interconnections within the cloud are unknown.

The above case would be different if device B had no other devices connected to it and did not itself generate any traffic (from which topology information can be derived). A two-port unmanaged repeater is just such a device. In this case, neither Device A or B would be aware of it - it would be totally transparent to the physical topology.

A third example of the impact of non-managed devices is if a leaf device, such as mac4 in the example above, did not report topology information. In this case the only information missing would be the port that mac4 is using to connect to the network. For typical leaf devices such as workstations, there may be only one port of attachment so the impact of not supporting the topology MIB is minimal.

6.4.4.2 Two or more agents in the same device

The Entity MIB allows for multiple agents to exist in the same managed system, with possibly overlapping management responsibilities. In this situation, all agents should use the same device ID if possible.

6.4.4.3 Multiple links between two devices

In this case, the device ID may not be sufficient to determine which pairs of ports are actually connected. It may be required for the

devices to exchange device and port information in a store and forward manner or to use unique device IDs for each port.

Expires September 1997

[Page 12]

6.4.4.4 Agent knows the entire topology

If the topology mechanism is extremely robust, it is possible that agents in the network could know more than their own local topology. For instance, the agent in Device B could know the port connections between Device A and mac1 and mac3. In the extreme, the manager might only be required to gather data from a single agent in order to ascertain the complete physical topology for all devices employing the same topology mechanism.

[6.4.5](#) Ring topologies

to be supplied

[6.4.6](#) MIB Requirements

The following information must be provided by the PTOPO MIB

- uniquely identify the local device using the device ID
- for each port on this device, identify the type of topology mechanism(s) in use and the instance of that mechanism.
- uniquely identify each of the physical attachment points or ports of the device (use entity MIB)
by slot and port number
- for each port, provide the topology information available and indicate the topology mechanism used to obtain that data. This information may include device IDs identified as downstream from this port and port IDs associated with those device IDs.
- device IDs should identify downstream agents that may contain additional topology information
- timestamp as to when the topology knowledge at this agent last changed

[6.5](#) Manager Process

A manager uses the following process to determine the physical topology of a portion of the network using a common topology mechanism. Limiting the manager process to those devices using a common mechanism is not required, but it does contain the effort and allows the topology to be built piece by piece in a methodical way. The process described assumes that a single topology mechanism is in use across the set of devices over which physical topology is being determined. Manager processing for overlapping topology mechanisms is described subsequently.

- a manager first identifies a managed device on the network to begin the physical topology collection process
- the manager chooses a port to begin the topology determination and obtains the topology mechanism and instance for that port
- for each port on the device which use the same instance of this topology mechanism, the manager obtains the topology data for that port and any downstream agent identities.
- the manager then queries all downstream agents identified by this device and repeats the previous step.
- the manager continues walking through the topology until it has no other new agents identified and has collected topology data from all agents.
- the manager then performs topology calculations as required based on topology data returned to determine the actual physical topology of this collection of devices.
- once this portion of the network has been mapped, the manager should identify other ports on devices that are running a different instance of the topology mechanism or a different topology mechanism altogether and repeat the process to map that topology.
- following this iterative procedure, the physical topology of an arbitrarily large network can be calculated.

6.5.1 Processing for Overlapping topology mechanisms

If multiple topology mechanisms are being used across two overlapping sets of devices, the manager has two choices. the first choice is to treat each mechanism independently and follow the above algorithm separately for those each mechanism. when the topology calculations are completed, the manager can review the final topologies and determine how to collapse the two physical topologies into one physical topology, based on redundant information in both calculations. Alternatively, the manager can combine the information obtained from each mechanism by the local agent and use this combined information to calculate a single physical topology across both overlapping sets of devices. while this approach probably provides the most robust solution, it will require more computation by the manager since a larger set of devices is being processed at one time. If all devices are using both mechanisms, then the calculation should definitely be done on the combined data.

7. Security Considerations

The ultimate goal of this framework is to allow the creation of standards to provide consistent and interoperable multi-vendor support for physical topology. The gathering of physical topology information is a read-only function, and is therefore limited in its impact on security. In some cases, even the determination of physical connectivity by unwarranted persons could indeed constitute a breach of security. This issue will be investigated more fully as part of the effort to define the physical topology MIB.

8. Open Issues

- May want to add concrete examples of topology mechanisms
- need to add section on FDDI and Token Ring topology

9. Acknowledgments

I would like to thank my fellow employees at Bay Networks for help and guidance in preparing this document and would also like to thank my wife, who delayed delivering our fourth child, so that I could finish this document on time.

10. References

[1] McCloghrie, K, and Bierman, A., "entity MIB using SMIV2", RFC 2037, October, 1996

11. Author's Address

Kendall S. Jones
Bay Networks
4401 Great America Parkway
Santa Clara, CA
phone: 408-495-7356
email: kjones@baynetworks.com

Expires September 1997

[Page 15]