Simple Network Management Protocol (SNMP)

In this report:

Editor's Note

Simple Network Management Protocol (SNMP) originated in the Internet community as a means for managing TCP/IP networks and Ethernet networks. During 1989, SNMP's appeal broadened rapidly beyond the Internet, attracting waves of users searching for a proven, available, multivendor-network monitoring method.

Report Highlights

The Simple Network Management Protocol (SNMP) is a viable alternative to the ISO CMIP over TCP/IP (CMOT) protocol. Originally defined to manage TCP/IP networks, SNMP can also be used to manage OSI networks. "Agents", "managers", and "Management Information Bases" combine to control network devices. Non-SNMP devices can be managed with proxy agents. SNMP's designers created a successful vehicle for multivendor network management; however, the protocol itself is less important than what users can do with the data. This report explains SNMP's architecture, details its history and implementation, and discusses its advantages and disadvantages.

—By L. Michael Sabo
Communications Architect SSDS, Inc.

SNMP History

SNMP evolved from the Simple Gateway Management Protocol (SGMP), formalized in November 1987 by Chuck Davin of MIT (formerly with Proteon); Jeffrey Case of the University of Tennessee/SNMP Research, Inc.; Mark Fedor of NYSERNet; and Martin Schoffstall of NYSERNet. SGMP was an early attempt to address the issue of network router management under TCP/IP. While SNMP is similar to SGMP in architecture and design philosophy, the syntax is different and the two protocols are incompatible.

In August 1988, the same four SGMP authors formalized SNMP as an Internet Draft Standard. In April 1989, SNMP became an Internet Recommended Standard (RFC 1098). Table 1 lists the current SNMP RFCs.

The Internet Activities Board (IAB) is currently examining both SNMP and OSI's Common Management Information Services and Protocol over TCP/IP (CMOT) as potential solutions for TCP/IP network management. Although many analysts view CMIP over the OSI stack as the preferred long-term solution for network management, TCP/IP implementations are widely available today and will continue in use for some time. Furthermore, SNMP is eclipsing CMOT as the interim TCP/IP solution.

Using SNMP

Using SNMP, a network administrator can address queries and commands to network nodes and devices. It can be used to monitor network performance and status; control operational parameters; and report, analyze, and isolate faults. The protocol performs these functions by carrying management information between *managers* and *agents*.

SNMP Architecture

SNMP operates on three basic concepts: manager, agent, and the Management Information Base (MIB) (see Figure 1).

A **manager** is a software program housed within a Network Management Station. The manager has the ability to query agents, receive agent responses, and set specific variables using various SNMP commands.

An **agent** is a software program housed within a managed network device (such as a host, gateway, terminal server, etc.). An agent stores management data and responds to the manager's data requests.

The **Management Information Base (MIB)** is a database of managed objects, accessible to agents and manipulated via SNMP to provide network management information.

The MIB

The MIB conforms to the Structure of Management Information (SMI) for TCP/IP-based Internets, as described in RFC 1155. This SMI, in turn, is modeled after OSI's SMI, as defined in Draft Proposal (DP) 2684. While the SMI is similar for both SNMP and OSI environments, the actual objects defined in the MIBs are different.

SMI conformance is important, since it means that the MIB is capable of functioning in both current and future SNMP environments. In fact, the Internet SMI and the MIB are completely independent of any specific network management protocol, including SNMP.

The Internet-Standard MIB

Each SNMP agent contains instrumentation that, at minimum, must be capable of gathering "Internet-standard MIB" objects specified in RFC 1156 (May 1990). Such objects include network addresses, interface types, counters, thresholds, and similar data for all network devices and NMSs involved. (Nonstandard MIB objects are manageable under SNMP, provided they are defined using SMI conventions specified in RFC 1155.)

Objects are defined using a subset of Abstract Syntax Notation One (ASN.1), the ISO SMI specification language. Also, SNMP's designers chose the ASN.1 basic encoding rules to align the protocol with the OSI environment.

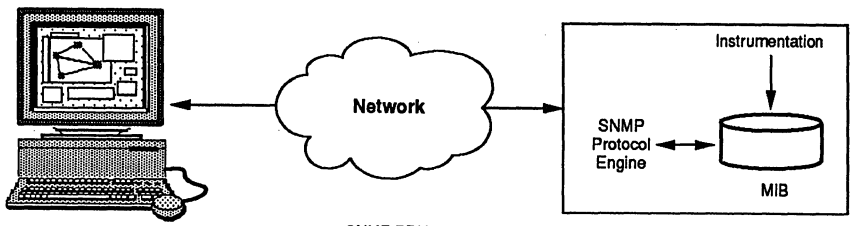
The standard MIB's structure is logically represented by a tree. The root (which is unlabeled) divides into three main branches: ISO, CCITT, and Joint ISO/CCITT (see Figure 2). Within the Internet subtree, which is several levels down the ISO branch, exist four subtrees: Directory, Management, Experimental, and Private. The Experimental subtree is reserved for Internet research purposes. The Internet-standard MIB, now at revision level MIB-I, finds its root under the Management subtree. Under the Private subtree is a very important branch called Enterprises.

The Private Enterprise

The Enterprise subtree, with its root under Private, is reserved for organizations wishing to develop extensions to the Internet-standard MIB. Organizations may apply for a specific Enterprise number, which uniquely identifies the organization's management tree, and is essential if the device is to manage objects other than those defined in the standard MIB.

Figure 1.
SNMP Architecture

SNMP has three architectural components: manager, agent, and MIB. Agents collect management information through instrumentation and store the information in a database called the MIB. The agent will provide management information to an SNMP manager upon request.



Organizations may obtain a Private Enterprise number (free of charge) by sending an electronic mail message on the Internet to jkrey@isi.edu. While the IAB controls the contents of the Internet-standard MIB, Private Enterprise MIBs are controlled by vendors or other special-interest groups. As of July 1991, over 250 Private Enterprise numbers had been assigned.

Agent Responsibilities

Each agent possesses its own MIB view, which includes the Internet-standard MIB and, typically, other extensions. The agent's MIB need *not* implement *every* group of defined variables in the standard MIB specification RFC 1156. For example, gateways need not support objects applicable only to hosts. This eliminates unnecessary overhead, facilitating SNMP implementation in smaller LAN components with little excess memory capacity. If a device supports a specific protocol (such as UDP), however, *all* objects from that particular group (i.e., the UDP group) within the MIB *must be supported.*

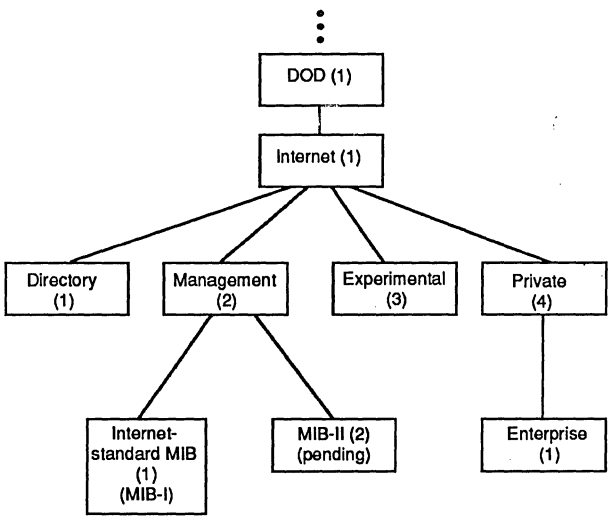
An agent performs two basic functions:

- Inspects variables in its MIB
- Alters variables in its MIB

Inspecting variables usually means examining the values of counters, thresholds, states, and other parameters. Altering variables may mean resetting these counters, thresholds, etc.

It would be possible to actually reboot a node, for example, by setting a specially defined variable (assuming one exists) to reboot=1. Most "SetRequest" commands accomplish tasks such as modifying routes or interface types, however. (For more information on SetRequest and other SNMP commands, see How SNMP Works, following.)

Figure 2.
The Management Information Base (MIB)



This figure depicts the location of the Internet-standard MIB within the Internet tree.

Obtaining RFCs on the Internet

RFCs are available through file transfer protocol (FTP) from Internet host NIC.DDN-.MIL. Log in using the username **anonymous** and the password **guest**. Once logged in, type in **get RFC:RFCnnnn.txt**, where nnnn is the RCF number. For example, **get RFC:RFC1157** will retrieve *A Simple Network Management Protocol (SNMP).*

RFCs can also be obtained through electronic mail. Send a message to SERVICE@NIC.DDN.MIL and place the RFC number in the subject field.

FTP to NIC.DDN.MIL with the **anonymous guest** login to obtain a current index of all RFCs. Once the session is established, type **dir RFC:RFC-INDEX**. A document name, such as RFC-INDEX.TXT.nnnn, will be returned. The nnnn represents the latest RFC number. Type **get RFC:RFC-INDEX.TXT.nnnn** to fetch the index for review. Type **quit** to log out.

Manager Responsibilities

Managers execute network manager station (NMS) applications and often provide a graphical user interface depicting a network agents map. The manager also typically archives MIB data for trend analysis.

How SNMP Works

SNMP Protocol Data Units (PDUs)

To carry out these duties, SNMP specifies five types of commands, or verbs, called *Protocol Data Units*:

1. GetRequest
2. GetNextRequest
3. SetRequest
4. GetResponse
5. Trap

GetRequest and GetNextRequest

An agent will inspect the value of MIB variables after receiving either a GetRequest or GetNextRequest command (PDU) from a manager.

SetRequest

The agent will alter MIB variables after receiving a SetRequest command. An NMS, for example, could instruct an agent to modify an IP route using SetRequest. It is a powerful command and could corrupt configuration parameters and seriously impair network service if used improperly. Due to SNMP's lack of inherent security measures (see SNMP's Disadvantages), some component vendors have not implemented or enabled SetRequest within their SNMP agent implementations. Many vendors

Table 1. RFCs Applicable to SNMP

RFC Reference	Title	Date
RFC 1155	Structure and Identification of Management Information for TCP/IP-based Internets	May 1990
RFC 1156	Management Information Base for Network Management of TCP/IP-based Internets	May 1990
RFC 1157	A Simple Network Management Protocol (SNMP)	May 1990
RFC 1158	Management Information Base for Network Management of TCP/IP-based internet: MIB-II	May 1990
RFC 1161	SNMP over OSI	June 1990
RFC 1187	Bulk Table Retrieval with the SNMP	October 1990
RFC 1215	A Convention for Defining Traps for use with the SNMP	March 1991
RFC 1227	SNMP MUX Protocol and MIB	May 1991
RFC 1228	SNMP-DPI—Simple Network Management Protocol Distributed Program Interface	May 1991
RFC 1229	Extensions to the Generic-Interface MIB	May 1991
RFC 1230	IEEE 802.4 Token Bus MIB	May 1991
RFC 1231	IEEE 802.5 Token Ring MIB	May 1991
RFC 1232	Definitions of Managed Objects for the DS1 Interface Type	May 1991
RFC 1233	Definitions of Managed Objects for the DS3 Interface Type	May 1991
RFC 1238	CLNS MIB—for use with connectionless Network Protocol (ISO 8473) and End System to Intermediate System (ISO 9542)	June 1991
RFC 1239	Reassignment of Experimental MIBs to Standard MIBs	June 1991
RFC 1243	AppleTalk Management Information Base	July 1991

are working to enhance security features within their products in order to offer a more secure SetRequest implementation.

GetResponse

An SNMP agent responds to an SNMP manager's GetRequest, GetNextRequest, and SetRequest PDUs with a GetResponse PDU. The GetResponse includes the original request followed by the requested information. Returning the original request with the response implements a stateless protocol where the manager need neither track outstanding requests nor correlate replies.

Traps

Trap is a special, unsolicited command type that agents send to a manager after sensing a prespecified condition, such as ColdStart, WarmStart, LinkDown, LinkUp, AuthenticationFailure, EGPneighborLoss, or other enterprise-specific events. Traps are used to guide the polling timing and focus, which SNMP employs to monitor the network's state.

Transport Mechanisms

As mentioned previously, managers and agents exchange commands via messages. SNMP's monitoring and control transactions are not actually TCP/IP dependent—SNMP only requires the datagram transport mechanism to operate. It can therefore be implemented over any network media or protocol suite, including OSI. There are currently two standard SNMP transport mechanisms: User Datagram Protocol (UDP) and within Ethernet frames (as defined in RFC 1083). Currently, there are no commercial implementations of SNMP directly over Ethernet. All commercially available SNMP NMSs use UDP to exchange SNMP PDUs. Figure 3 diagrams SNMP's relationship with its transport mechanisms in terms of the OSI model.

UDP

Each SNMP message is represented entirely within a single UDP datagram. This lessens the message processing burden and helps to minimize the agent's complexity. The SNMP message consists of:

—version identifier—SNMP community name—PDU

The *version identifier* refers to the RFC version (currently at 1). An SNMP *community* consists of an agent and its associated applications. As mentioned before, a PDU is one of five command types. The SNMP protocol entity receives most messages at UDP port 161 on its associated host. Traps are received on UDP port 162.

GNMP:
New Kid on the Block

On May 31, 1991, the National Institute of Standards and Technology (NIST) issued its first version of the *Proposed Government Network Management Profile (GNMP)*, a document to provide "the standard reference for all Federal Government agencies to use when acquiring Network Management (NM) functions and services for computer and communications systems and networks." The proposal discusses:

- the scope of the GNMP
- its applicability
- its development
- its specification sources
- its relationship to the Government Open Systems Interconnection Profile (GOSIP)

Scope
The GNMP mandates CMIS and CMIP as the management information exchange protocol. Managed Objects (MOs) are included from DMI, NMSIG 90/197, IEEE 802.3 HUB Management, and other international standard publications. The proposal also details five systems management functions:

1. Object Management Function
2. State Management Function
3. Attributes for Representing Relationships
4. Alarm Reporting Management Function
5. Event Reporting Function

Applicability
GNMP is mandatory for federal agencies. This presents problems since some of the standards it adopts (CMIP, for example), and the GNMP itself, are still under development. The intent, however, is to provide guidance in selecting from current NM tools while evolving tighter specifications.

Development
NIST conducted a survey of federal agencies in the summer of 1990. The results indicated that the management of local area networks and their interconnecting bridges was a prime concern. GNMP was proposed specifically to address that concern. The Phase I implementation, proposed as GNMP Version 1.0, focuses on management of OSI model layers 1 and 2.

Specification Sources
NIST cites part eighteen of the *OIW Stable Implementation Agreements, December 1990* as the primary specification source for GNMP Version 1.0. Other sources, however, provided the additional specifications needed to provide the minimum-required management capabilities.

Relationship to GOSIP
GNMP is the management information specification of the networks defined by GOSIP. GOSIP specifies the protocol stacks and system services that convey the management information between managed objects and their managing systems. GNMP cites GOSIP heavily, and later versions of each document will continue to cross-reference each other.

Conclusion
The GNMP will have a significant impact on the network management marketplace in the United States for two primary reasons. First, it is a guideline for implementing accepted international standards for network management. Second, the use of GNMP is mandated for federal agencies and encouraged for companies that do business with federal agencies.

The second reason is causing some trouble in Corporate America. GNMP specifically omits Simple Network Management Protocol (SNMP), widely used for network management. It is conceivable, though unlikely, that use of GOSIP and GNMP could become mandatory for firms wishing to conduct business with the federal government and for colleges and universities that receive federal aid, causing a costly retooling of entrenched network systems.

Ethernet Frames
SNMP over Ethernet frames, while implementable, is not recommended by the SNMP specification authors. While the SNMP message looks the same, an SNMP NMS must be configured to accept SNMP PDUs directly from the Ethernet driver.

SNMP Proxy Agents

Proxy agent software permits an SNMP manager to monitor and control network elements that are otherwise not SNMP addressable. For example, a vendor wishes to migrate its network management scheme to SNMP but has devices on the network that use a proprietary network management scheme. An SNMP proxy can manage those devices in their native mode. The SNMP proxy acts as a protocol converter, translating the SNMP manager's commands into the proprietary scheme. This strategy facilitates migration from the current proprietary environment to the open SNMP environment (see Figure 4).

Proxy agents are well suited for vendors with an existing base of non-SNMP devices communicating efficiently under a proprietary scheme. By using a proxy agent, the vendor can reduce the investment risk of putting SNMP equipment into the field.

SNMP's Advantages

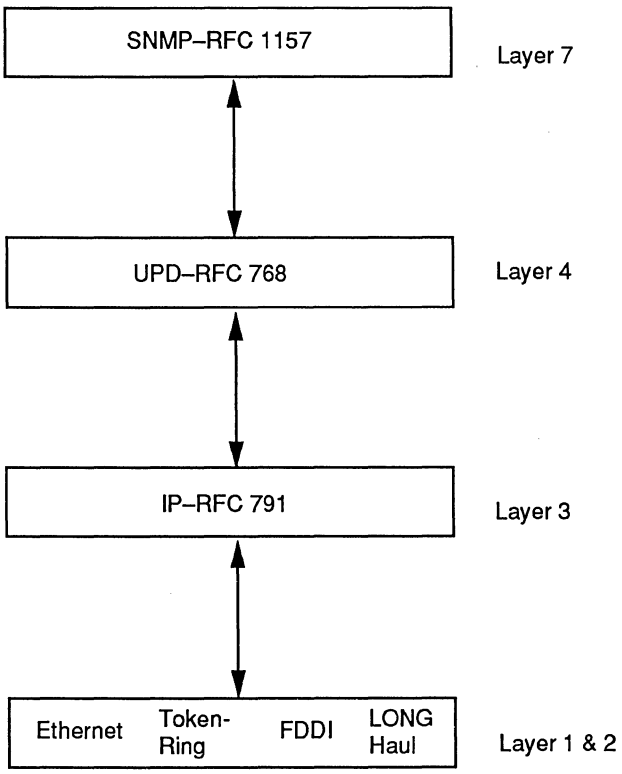
SNMP's major advantages are the following:

- Its simplicity eases vendor implementation effort
- Its memory and CPU cycle requirements are lower than CMIP's
- Its protocol has been used and tested on the Internet
- Its products are available

Simplicity

SNMP's designers successfully kept the protocol simple, easing vendor implementation and thereby encouraging widespread implementation.

Figure 3.
SNMP Protocol Suite



This figures details the relationship between SNMP RFCs and the OSI Seven Layer Model.

Memory and CPU Use

By using only a subset of ASN.1 to define the MIB and implementing only five command types, agent implementations require far less memory and fewer CPU cycles than most network management protocols, including CMIP.

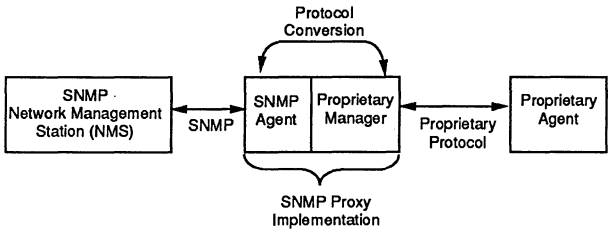
Tested and Used

SNMP has a distinct advantage over CMIP in that it was tested and actually used by the Internet community *before* it became a standard. The RFC process of standardization, in fact, requires that a critical mass of users employ and then comment on a particular protocol or other specification before the authors submit it for approval. In contrast, ISO's method is to develop a preliminary standard after much commenting, with implementation and testing as a poststandard process.

Availability

Finally, SNMP products are available now. While SNMP is not sophisticated, its availability will give many network managers the opportunity to try out multivendor network management and possibly discover what they need to manage their networks. SNMP developers and proponents know that the SNMP tools available now fall far short of satisfying user needs. Yet, the best way to clarify those needs is to get experience and redefine requirements on an ongoing basis.

Figure 4.
SNMP Proxy Implementations



SNMP proxies help a vendor migrate a proprietary network management scheme into the SNMP environment.

SNMP's Disadvantages

SNMP has several disadvantages, including the following:

- Lack of global vision
- Weak security features
- Problems with the Trap command

Lack of Global Vision

While SNMP's widespread deployment before standardization is an advantage in one respect (see SNMP's Advantages), the protocol's definition is more or less cast in stone—essentially without giving vendors and users outside the U.S. the opportunity to voice their needs, concerns, and suggestions. The Internet Activities Board (IAB) is, of course, under no obligation to do so. Yet European vendors, commercial users, and academicians are now embracing SNMP, since they share our same need for an open, multivendor network management protocol. SNMP may spread throughout the world and leave some non-U.S. users at a disadvantage.

In contrast, ISO/OSI standards developers have a truly global vision and put a high priority on accommodating the viewpoints expressed by all nation representatives. In the OSI community, developing nations are heard on an equal par. The price of emphasizing universal applicability within OSI standards is a much slower pace of standards development, as compared to rapidly developed standards such as SNMP.

Security Issues

There are very few security mechanisms defined as part of the SNMP protocol specification. For example, there is no capability defined to ensure that SNMP PDUs received by an agent actually originated from an actual manager—and not from an unauthorized interloper.

Thus, vendors are reluctant to support the SetRequest verb on their agent implementations. SNMP does, however, support access modes of read-only/read-write classifications for MIB variables. To employ this capability, the user may configure a variable such as RouteTable as read only. This prevents the agent from setting RouteTable to a potentially harmful value, if an unauthorized perpetrator tries to fool the agent with a SetRequest command. However, using read-only when the variable should be defined read-write effectively disables the SetRequest verb on those variables and reduces SNMP's functionality.

Trap Problems

SNMP does not define the mechanism for where a Trap should be sent, nor what the agent should provide as part of a Trap (even for standard Traps). The specification

merely notes that a Trap should include "interesting information." Thus, Trap is implementation specific.

Conclusion: Meeting the Goals

SNMP's authors adhered to several design goals during the protocol's comparatively brief development time:

1. **Keep the agent simple**—Minimize the number and complexity of the agent's management functions.
2. **Make SNMP monitoring and control extensible**—Accommodate unanticipated aspects of network operation and management.
3. **Make SNMP architecture independent**—Do not code to any particular host/gateway architectures.

SNMP's designers achieved the first goal by limiting functions to five and by requiring only an unreliable datagram service such as UDP. Simplifying the agent reduces vendor development costs—making it more attractive for component vendors to support SNMP. Widespread availability of SNMP agents is a prerequisite both for user acceptance of SNMP *and* for stimulating NMS vendors to integrate SNMP manager implementations.

The second goal is realized by employing the MIB. The current Internet-standard MIB will evolve and expand to include more items—thus giving network managers more control over their networks. New objects are added by integrating new subtrees into the MIB. SNMP can easily traverse the new structure by using the GetRequest or GetNextRequest command.

The third goal is realized via the open communications architecture above which SNMP operates and by SNMP's capability to operate over many different transport mechanisms. In addition, through the use of ASN.1 basic encoding rules, SNMP is not tied to any specific machine architecture.

SNMP's designers successfully created a new vehicle for multivendor network management, a first step toward solving an increasingly critical problem. SNMP is merely a vehicle. The protocol itself is less important than the tools and applications that must be developed; how the data is gathered is less important than what users can do with the data. Most of the work lies ahead, as users gain experience with managing multivendor nets and determine what works and what does not. ■

This report was prepared and updated exclusively for Datapro by L. Michael Sabo, a communications architect with SSDS, Inc., Littleton, CO. Mr. Sabo is currently consulting on various networking and internetworking projects. Previously, he participated in porting TCP/IP to the emerging ANSI High-Performance Parallel Interface (HIPPI) Gigabit/sec LAN standard. Mr. Sabo has been active in integrated network management. He participated in developing an object-oriented and SNMP-based network management architecture for Lockheed Integration Services. This effort included defining numerous private enterprise management information base (MIB) objects to support system management functions.

Mr. Sabo is a member of the SNMP working group and has been active in the Internet for six years. He is a member of the board of advisors for Datapro *Network Management*. He holds a master's degree in data processing management from the University of Denver and a bachelor's degree in Computer Science from Wright State University.