# Management of Optical Networks: SNMP agents enabling TMN

Abilio Carvalho
CET / Telecom Portugal (acarv@smtpdos.cet.pt)

Fausto de Carvalho
CET / Telecom Portugal (cfausto@briol.cet.pt)

George Pavlou
University College London (gpavlou@cs.ucl.ac.uk)

**Abstract:** With the advent of Optical Networks, an important opportunity for the application of Telecommunication Management Network (TMN) framework emerges. At present, most such networks have small or no management capabilities and are characterised by lack of computational resources. Instead of introducing interim proprietary management facilities, the Internet SNMP model is particularly suited for providing low cost agents with minimal resources, based on personal computer technology; this is the approach adopted by RACE ICM and COBRA projects for managing the later's Passive Optical Network. Such Management interfaces can be automatically converted to standard TMN Qx/Q3 ones through suitable infrastructure as developed by the ICM project. This paper discusses the suitability of a phased approach towards full TMN, proposing an inexpensive SNMP based platform for element agents and explaining the issues behind the automatic conversion to standard TMN interfaces that will retain the initial investment.

**Keywords:** SNMP, TMN, Optical Networks, Windows Sockets, Q-Adaptor

## 1. Introduction

With the advent of Optical Networks (ONs), an important opportunity for the application of TMN concepts and models [TMN] is emerging as little work has been done yet for their management. This allows the introduction of the TMN framework as a means to guide work in this area and be used at the same time as a vehicle to assess and evaluate the practicality and benefits of the TMN approach. Aiming at these goals, the RACE projects ICM (R2059) and COBRA (R2065) established a collaboration to allow the use of the COBRA Coherent Multi-Channel (CMC) passive optical demonstrator network as an ICM testbed for one of its validation/demonstration phases [PON].

Centralised and global wavelength management and control is required by networks using Coherent Multi-Channel technology in order to explore the high degree of flexibility exhibited as their main advantage [Khoe]. The inherent flexibility displayed by the CMC technology needs to be matched by the capabilities provided by the management and control system or the gain in flexibility may be lost because of the lack of adequate management policies.

CMC networks have special requirements of security services since information conveyed in the network is distributed to all users and is potentially accessible by all. Authentication procedures for control of channel allocation at the receiver side are an essential element for the management of such networks.

Initially, external management was not planned for the COBRA demonstrator, therefore no TMN "hooks" existed. The option was to develop a simple proprietary agent that could be interconnected with the TMN management system through a Q-Adaptor. Based on previous ICM work on interworking between CMIS/P and SNMP and considering the computing resources of the experimental optical network, SNMP was chosen as the mechanism to provide access to network elements for monitoring and control. This approach allows an easy integration of network element agents in the existing COBRA system and makes full reuse of key ICM system components, such as the generic CMIS/P-SNMP gateway (Q-Adaptor) [Del5], enabling thus ICM developed TMN Operations Systems (OSs) to exercise the specified management policies.

In this paper we present our choice of hardware and software platform based on inexpensive personal computer technology that can be used to realise such SNMP element agents. The technical aspects of that platform are explained in section 2 while section 3 discusses aspects of the generic CMIS/P and SNMP interworking, encompassing the technical advantages and disadvantages when deployed in TMN systems. A discussion of the approach as a whole follows together with our conclusions on its feasibility in a real telecommunications environment.

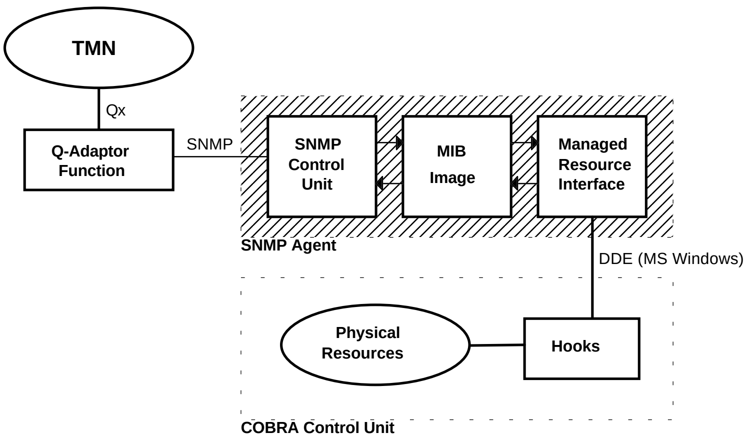


**Fig. 1.** The SNMP Agent and its environment

## 2. The SNMP Agent Platform

The COBRA network is accessed through its Control Unit, where all the management actions can be performed [COBRA]. This unit is built on an inexpensive PC, running MS Windows, with a TCP/IP protocol stack underneath and the SNMP agent has been implemented as a set of tasks running concurrently with the already existing ON control software.

This SNMP Agent Platform comprises three blocks [ACarv93] [ACarv94], as shown in figure 1: the SNMP Control Unit, the MIB Image and the Managed Resource Interface.

### 2.1. The SNMP Control Unit

The SNMP Control Unit is generic and embodies all the SNMP protocol intrinsic aspects. This unit comprises a generic module for sending and receiving SNMP messages using the User Datagram Protocol/Internet Protocol (UDP/IP) [UDP]. For this purpose, the TCP/IP Trumpet protocol stack and sockets interface for Microsoft Windows is used [Trumpet], following the Windows Sockets specification [WinSock].

Included in this unit there are also an ASN.1 module composed by several functions that are used for decoding SNMP messages into internal structures and the encoding (serialisation) of these internal structures into octet sequences using the Basic Encoding Rules (BER) of ASN.1 [ASN1].

The SNMP Control Unit has a main loop - the processing function. For each managed object identified in the SNMP message, the processing function looks for the corresponding object representation in the MIB Image and triggers the associated access method located in the Managed Resource Interface. The processing function is responsible for the implementation of all mechanisms associated with the SNMP protocol. Therefore, the arrived SNMP Message is correctly formatted by the processing function with the information provided by the access method.

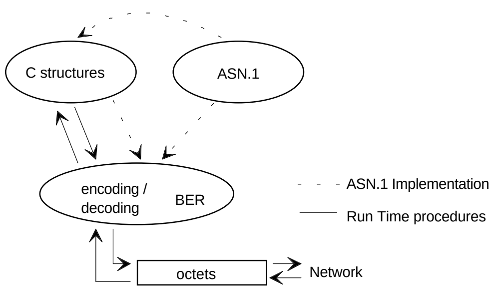


**Fig.2.** Encoding and decoding SNMP messages

## 2.2. MIB Image and Managed Resource Interface

The MIB Image is a set of dynamic C structures, implementing an object-oriented view of the physical resources [MIB]. It constitutes a complete independent module to which new managed objects can be added and its access is provided by a set of MIB access methods. These routines can be used for the search of managed objects or for name conversion (textual name to OID and vice-versa). An object can be found using its textual name, its Object Identifier (OID) and for the Get-Next-Request there is a routine that returns the next object lexicographically located in the present MIB.

The MIB Image is built based on an object's file in mosy format. The mosy format is the output of the Managed Object Syntax Compiler Yacc-based (mosy) included in the ISODE environment [ISODE]. It is an ASCII format and thus can be used in any system, such as a PC running MS Windows. This mosy format file is one of the techniques that the programmer uses to add new managed objects to the current SNMP agent.

The MIB Image is an implementation of the Management Information Tree (MIT) in which each managed object is represented (name, syntax, OID) and has two method pointers (get and set) pointing to the associated access methods located in the Managed Resource Interface.

The Managed Resource Interface is a set of access methods and is responsible for the communication between the MIB Image and the physical resources and makes the SNMP Agent Platform independent of the Network Element (NE) where it will be installed.

This interface is based on MS Windows Dynamic Data Exchange (DDE) mechanisms and uses a very simple protocol in which each object is identified using a DDE item as its ASCII name and the values are exchanged through DDE transactions in ASCII format. Due to the mutual knowledge about each object (by the SNMP Agent and the DDE Server included in the NE), the syntax of the object value is converted to and from the ASCII syntax as needed.

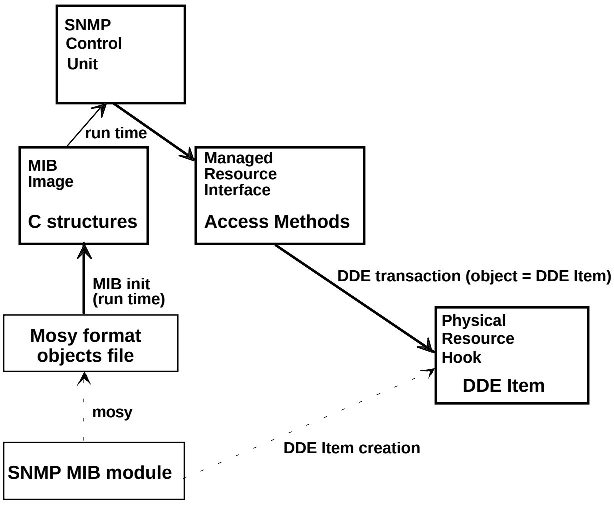


**Fig.3.** Internal mechanisms of the SNMP agent

Adding new managed objects to this SNMP Agent Platform is very simple: from the new MIB module, objects file is updated using mosy facilities, so next time the agent runs, it will have in memory an updated MIB Image. The next step consists of writing the access routines (get/get-next/set) in Managed Resource Interface module. These routines are created based on generic skeletons, having only slight differences to reflect the particularities of each new managed object.

The use of DDE exploits the potential of the Windows environment and its networking capabilities: with NetDDE, the new network-wide dynamic data exchange mechanisms, available in Windows for Workgroups and Windows NT, it is possible to place the SNMP agent in another PC in the network.

Finally, a reference to alarm handling: system alarms are forwarded to the Q-Adaptor using the Trap mechanisms of SNMP. The SNMP Control Unit receives these alarms from managed resources, constructs an SNMP Trap Message and then uses the appropriate ASN.1 code to apply BERs, before sending it by using Windows Sockets. The SNMP Trap uses the unreliable connectionless transport mechanism offered by the Internet UDP. This weakness is overridden by connecting the SNMP agent and the Q-Adaptor to a common physical network with high level reliability, such as an Ethernet.

## 3. The CMIS/P to SNMP Generic Gateway

### 3.1 Concept and Generic Translation

SNMP was conceived as an interim solution to the management of TCP/IP-based internets, to be eventually replaced by OSI management over TCP/IP, the approach known then as CMOT. As such, its information model has been a scaled down version of the OSI one, taking a simpler object-based than the fully fledged object-oriented OSI approach. The potential for translation between the two models has been obvious from the beginning and the TCP/IP MIB-II was translated "by hand" to the equivalent OSI GDMO one. An early implementation of the latter as an example application of the OSIMIS platform [Pav93a] showed the feasibility of the approach.

This dual agent approach, though possible, requires substantial investment to re-implement agents for which there already exists an SNMP implementation. A proxy agent approach is more suitable and can become a really interesting proposition if automatic translation procedures can be specified in which case the whole process can be automated.

Work towards that direction was undertaken within the RACE ICM project which investigated the possibility of a fully automatic translation between CMIS/P and SNMP, through suitable tools i.e. translators and GDMO compiler and a generic proxy agent [Del5]. At the same time, the Network Management Forum (NMF) recognising the importance of co-existence and interworking between different management technologies, initiated an activity to define standards in this area. This work is by now complete [NMF1][NMF2]; ICM has contributed substantially to this work (the two approaches were harmonised) and the ICM prototype will constitute the first openly available implementation of such a gateway to the research community, verifying the feasibility of such an approach and instigating product development.

It must be noted that automatic translation is possible only from CMIS/P to SNMP. This is the most important direction, given the proliferation of network equipment with SNMP interfaces, all of which can be made instantly manageable through CMIS/P by proxy agents, acquiring all the additional advantages of OSI management as discussed in section 3.2.

First, let's define and clarify the terms proxy agent, application level gateway and Q-Adaptor (QA). The term proxy agent is used in both the OSI and Internet management models to describe an agent whose managed objects are mapped onto managed objects of another model with a different protocol used as the access method. A proxy system could be proxying for more than one network element of a whole subnetwork, in which case it could be considered as an application level gateway, being described in terms of the OSI layered model. In TMN parlance, a Q-Adaptor is a converter between any proprietary M management interface e.g. SNMP to the standard TMN ones, Qx or Q3 i.e. CMIS/P - the Qx/Q3 relationship and its relevance to this automatic translation is discussed later.

By the term "generic" proxy, gateway or Q-Adaptor, one refers to a system which is realised once and can be used subsequently to translate between the two interfaces for any information model without the need of additional logic.

An off-line procedure is only necessary to inform such a system of any information model for which it may proxy; this involves the generation of run-time support starting from the formal definition of that model and using suitable translators/compilers.

As already stated, the SNMP information model is an object-based, limited version of the OSI one, with no inheritance and containment, a different naming scheme resulting in a linear object name space and only table entries being multiple instanced objects. A translation is possible [NMF1][Pav93b], resulting in a shallow inheritance tree and with a fairly shallow containment tree of at most two levels of depth. There are two approaches for the construction of proxy systems, stateless [NMF2] and stateful as realised by ICM [Sou93][Del5]. Powerful CMIS services such as scoping and filtering can be emulated, and in fact in the ICM system they are provided by the underlying OSIMIS platform [Pav93a]. The stateful approach requires more memory as local managed object copies are kept but may result in faster access times if cacheing mechanisms are used.

Such a proxy system is implemented once and supporting tools are used to generate run-time information describing the structure of the proxied information models. The exact procedure is to start with the SNMP information model, either in version 1 or 2. In the latter case, the SNMPv2 model is converted first to the SNMPv1 equivalent through a translator. Then, the SNMPv1 information model is translated to the equivalent OSI GDMO one through another translator implemented by ICM and known as Enhanced SNMP Management Information Compiler (ESMIC). Then a GDMO compiler is used to compile the resulting model and generate run-time support in the form of flat files - the GDMO compiler implemented by ICM is an integral part of the OSIMIS platform. Those files are read in the proxy when it starts-up and suitable internal representations are formed to guide the translation. The whole cycle is shown in Figure 4.

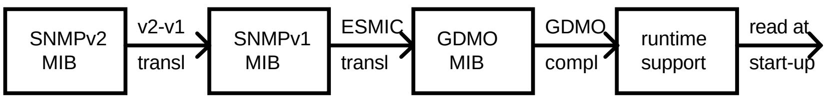


**Fig.4.** Proxy information translation cycle

From then on, such a proxy system can act as a translator for requests addressed to the proxied elements with native SNMP agents, its operation being totally transparent. Managers or TMN Network Element Management Operations Systems (NEM OSs) can access those without being aware of the proxy relationship involved but seeing every device as having a native OSI agent. Access control information translation is also specified to make intrusive management possible as far as the managed devices support it. The proxy system is configured to support a number of initial proxied elements. Others may be added or existing ones can be removed on-the-fly by creating and deleting managed objects representing a whole proxied system; of course, the transparency of operation is then lost as the manager must know it talks to a proxy system. The latter can translate to either SNMPv1 or SNMPv2, depending of the type of the underlying SNMP agent. The ICM system translates at present only to SNMPv1 as SNMPv2 is not yet widely adopted. If

SNMPv2 gains acceptance in the future, the system can be easily enhanced to translate also to SNMPv2.

### 3.2 Advantages, Problems and TMN Applicability

The advantages of the generic proxy or Q-Adaptor approach should be obvious, as it allows CMIS/P-based management of the large and increasing number of devices with SNMP management capabilities. This enables compliant TMN realisation using the OSI model as suggested by the relevant ITU recommendation [TMN] by talking to SNMP-capable network elements through Q-Adaptors. The advantages of using OSI management as opposed to SNMP are manyfold and are discussed here.

First, an event driven model of operation becomes immediately available at the proxy through the use of the OSI systems management functions such as event reporting, logging, metric and summarisation objects. ICM has also been researching in additional objects such as intelligent management summarisers, object creation, deletion and attribute change detectors, policy objects etc. Such objects enhance the "raw" information model that results from the SNMP translation, offering all the event driven sophistication the OSI model is capable of. Using that approach, polling is restricted in the local domain between the proxy system and the proxied agents.

In addition, a connection-oriented model of operation becomes immediately available to higher level managers, relieving them from the burden of handling retransmissions and allowing them to focus on their management policies. Bulk data transfer facilities for tables, logs etc. are made available through CMIS scoping while sophisticated querying is also possible through CMIS filtering. In general, the model of operation becomes that of OSI management, integrating seamlessly all those SNMP capable network elements. It should be finally stated that the centralised control implied by the use of such a system is not a problem as replication of a proxy system is possible to cater for that. ICM also looks into global naming schemes to support the replication, migration and other transparencies in addition to location transparency [Del5].

The first problem with that approach is the fact that two hops are required to access management information residing in an SNMP device. As far as the proxy system is relatively close to the proxied one (e.g. on the same LAN/MAN), this will result to a slight only performance degradation. Early performance measurements of this approach show a smaller than expected latency. In fact, a formal comparison is under way in ICM to look at the differences between native CMIS/P and SNMPv1/v2 and also the proxy approach. It is reminded here that the OMG CORBA approach which emerges as the standard model for distributed processing always requires a two hop mechanism through an object broker[CORBA].

A more serious limitation is the fact that the resulting information model from the translation is actually Qx rather than Q3 in TMN terms. This means that though it may be semantically similar to the equivalent GDMO model specified by ISO/ITU for that device e.g. a X.25 or ATM switch, the syntax is going to be different in terms of class and attribute names, containment relationships etc. As such, a mediation function is needed to translate from Qx to Q3 and this could be physically located in the same box, which will become a combination of a QA/MD, translating M (SNMP) to qx (non-standard OSI) and then to Q3 (fully standard OSI).

The task of automating this translation is addressed in ICM through the possibility of a formal language describing the inter-dependencies of two semantically similar but syntactically different OSI information models. Such a language will be supported by a suitable compiler to yield run-time support that will realise automatically that mediation function through information conversion. This relationship for the information models, reference points and interfaces involved is shown in Figure 5.

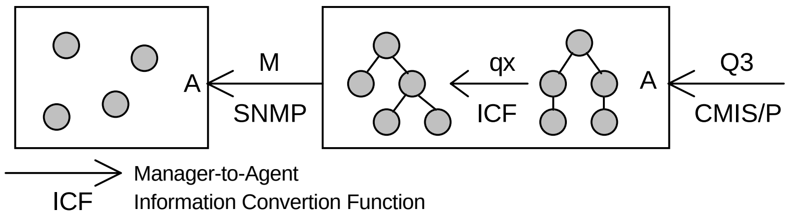


**Fig.5.** Information models, reference points and interfaces

## 4. Discussion and Conclusions

SNMP is the native management language for most of the communications devices related to data networks, like routers, bridges and gateways, through which the computing world is merging with telecommunications. On the other hand, telecommunications networks nowadays consist of a large number of systems with little or no management hooks included and deployed over large geographic areas. In particular, Optical Networks typically have few associated computational resources, thus reinforcing the advantage of using light-weight management protocols to interface to their network elements.

The approach presented in this paper, based on an inexpensive hardware and software platform environment, can easily be deployed, developed and enhanced to cover other telecommunications equipment that need to be managed by the TMN environment being set-up by Public Network Operators (PNOs). This will certainly bring down the cost of moving to global OSI management environments, contributing to a phased approach in the sense that a PNO will not need to make all the investment at once, with Qx/Q3 interfaces available in all his network elements, neither to have as many intermediate hybrid management systems that will become eventually obsolete by a complete TMN solution. Instead, the TMN can be deployed, explored and tuned in a phased approach, with SNMP enabling the early management of network elements with non-standard management interfaces, as well as the use of Management Applications that will not be greatly affected by the expected evolution of interfaces at the lowest TMN layer.

SNMP offers a simple, light-weight solution in terms of both processing power and memory requirements which allows the implementation of "standard" interfaces to network elements with minimal resources through the inexpensive software and hardware platform described. Its information model, though not as rich and expressive as the equivalent OSI one, allows to model most of the required management interactions, including imperative actions. Problems such as its polling nature due to the unreliable/limited trap capability and its limited support for bulk data transfer are eliminated in the local environment, while a full enhancement of its "raw" information model can be made available at the QA/MD. The technology for an automatic QA facility has been standardised by the Network Management Forum and related products are about to appear in the market place. As far as the SNMP network element information model is semantically equivalent to the ISO/ITU OSI one with respect to management capabilities, this approach can yield an inexpensive Qx/Q3 interface and ensure a painless TMN migration for PNOs without any loss of investment.

## 5. References


[ACarv93] A.P.Carvalho, "Integrated Platform for Development of Network Management Applications based on the SNMP", M.Sc. Thesis, University of Coimbra, November 1993

[ACarv94] A.P.Carvalho, N.P.Rocha, "SNMP Platform to develop managed objects", Revista do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro, Aveiro, January 1994, Vol.1 Nº1

[ASN1] D.Steedman, "Abstract Syntax Notation One, The Tutorial and Reference", Tecnology Appraisals, 1990

[COBRA] COBRA - RACE Project R2065, "Flexible Broadband Networks", WG 'system', August 1993

[CORBA] "The Common Object Request Broker: Architecture and Specification", OMG Document Number 91.12.1, Revision 1.1

[Del5] ICM - RACE Project R2059, "Revised TMN Architecture, Funtions and Case Studies", Deliverable 5, R2059/ICS/DPG/DS/P/007/b1, September 1993

[ISODE] M.T.Rose, J.P.Onions, C.J.Robbins, "The ISO Development Environment User's Manual Version 8.0", PSI, July 1991

[Khoe] G.Khoe, "Coherent Multicarrier Lightwave Tecnology for Flexible Capacity Networks", IEEE Communications Magazine, March 1994

[MIB] M.T.Rose, K.McCloghrie, "Consise MIB Definitions", RFC1212, PSI, March 1991

[NMF1] Network Management Forum, Forum 026: "Translation of Internet MIBs to ISO/CCITT GDMO", March 1994

[NMF2] Network Management Forum, Forum 028: "ISO/CCITT to Internet Management Proxy", March 1994

[Pav93a] G.Pavlou, "The OSIMIS TMN Platform: Support for Multiple Technology Integrated Management Systems", Proceedings of the 1st RACE IS&N Conference, November 1993, Paris

[Pav93b] G.Pavlou, S.N.Bhatti, G.Knight, "Automating the OSI to Internet Management Conversion through the use of an Object-Oriented Platform", In Advanced Information Processing Techniques for LAN and MAN Management: Proc. IFIP TC6/WG6.4, Elsevier Science Pubs, Amsterdam, 1993

[PON] J.Schmidt, T.Almeida, F.Carvalho, A.Carvalho, P.Legand, "ICM Phase 3 PON Case Study Description - draft version 1.2", R2059, April 1994

[SNMP] J. Case, M. Fedor, M. Schoffstall, J. Davin, "A Simple Network Management Protocol (SNMP)", RFC1157, May 1990

[Sou93] J.N.DeSouza, K.McCarthy, G.Pavlou, N.Agoulmine, "CMIP to SNMPv1 Translation Through Application Level Gateways using the OSIMIS/ISODE Platform", Proceedings of the 1st RACE IS&N Conference, November 1993, Paris

[TMN] CCITT M.3010, "Principals for a Telecomunications Management Network", Working Party IV, Report 28, December 1991

[Trumpet] P.R.Tattam, Trumpet Winsock version 1.00 Alpha #18

[UDP] J.Postel, "User Datagram Protocol", RFC 768, November 1980

[WinSock] M.Hall, M.Towfiq, G.Arnold, D.Treadwell, H.Sanders, "Windows Sockets - An Open Interface for Network Programming under Microsoft Windows - version 1.1", January 1993