

A VPN Management Architecture for Supporting CNM Services in ATM Networks

J. T. Park¹, J. H. Lee¹, J. W. Hong², Y. M. Kim³, and S. B. Kim³

*¹School of Electronic and Electrical Engineering
Kyungpook National University, Taegu, Korea*

*²Dept. of Computer Science and Engineering
Pohang University of Science and Technology, Pohang, Korea*

*³Telecommunicatoin Network Research Lab.
Korea Telecom, Taejeon, Korea*

Abstract

As enterprises use ATM networks for their private networks and as these private networks use public ATM networks for wide area communication, the need for the customers to be able to manage both private and public networks is increasing. Currently, some standardization work is being done towards providing this capability to the customers. In this paper, we propose a virtual private network (VPN) management architecture for supporting integrated customer and public network management in ATM networks. The key component of this architecture includes an integrated customer network management (CNM) agent and public network manager. It also incorporates a CORBA-based shared management knowledge (SMK) system to provide a distributed processing environment for the exchange of management information between managers and agents.

Keywords

Integrated customer and public network management, VPN Management, CNM agent, public network manager, TMN, ATM network, CORBA-based SMK system

1 INTRODUCTION

As private networks need to communicate each other (due to enterprises having multiple private networks throughout the country or throughout the world), these private networks often use public networks in order to exchange information between private networks, thereby, forming virtual private network. In such situations, network administrators of private networks generally would like to be able to monitor and manage their corporate network as well as public networks which their private networks use. Currently, this is a very difficult task mainly due to a couple of reasons: 1) administrative and 2) technical. The first reason is that public network administrators do not desire their networks to be openly managed by private network administrators. The second reason is that there is no clean mechanism available for the private network administrators to be able to manage any portion of the public networks. Recently, however, the trend is to allow the private network administrators to manage a portion of the public network essential to the operation of virtual private networks. Our work focuses on the technical aspect of allowing the public networks to be managed by private network administrators.

Some work has been carried out by standardization bodies such as ATM Forum, ITU-T and ETSI for providing the solution [4, 7]. The work mostly focuses on defining the interfaces for ATM network management and on the specification of management information base (MIB). ATM Forum has specified five management interfaces for the exchange of management information for ATM networks. Among them, M3 is a management interface between customer network management system (CNMS) and public network management system (PNMS), and M4 is the interface supporting network element level and network level management of public ATM networks. The M3 service is provided by the public network provider via customer network management (CNM) agents in the provider's network. CNM is a concept providing capabilities for network providers and their customers to exchange management information [1]. CNM services provide the customer using the provider's network with the capabilities to manage the portion of the public ATM network, which is critical to the operation of their virtual private networks.

Earlier work on CNM service provisioning stated the necessity of interworking between the CNM agents and public network managers [1, 5]. However, they did not provide solutions for the efficient interworking. Our work presented in this paper attempts to provide that solution. In particular, we propose an architecture for public network management system supporting CNM services. We have developed a public network manager which integrates CNM agents based on the M3 and M4 standard

management interfaces. Our integrated network management system incorporates a CORBA-based Shared Management Knowledge (SMK) system [9, 12] which provides a distributed processing environment for the exchange of management information between managers and agents.

This paper is organized as follows. Section 2 discusses CNM and interworking requirements. Section 3 presents our proposed architecture for integrating CNM agent and public network manager. Section 4 presents a prototype implementation carried out as a proof of concept. Finally, a summary and possible future work are given in Section 5.

2 CNM AND INTERWORKING REQUIREMENTS

The ITU-T recommendation X.160 [4] defines an architecture for the Customer Network Management Services for public data networks. The CNM functional architecture (shown in Figure 1(a)) is composed of several function blocks that provide the functionality needed for CNM. The customer's management function provides the customer-related CNM functionality using the CNM functions provided by the service provider. The service boundary between the customer's management function and the CNM function is defined by the CNM reference point. Management information is exchanged between the customer's management function and the CNM function through the CNM reference point.

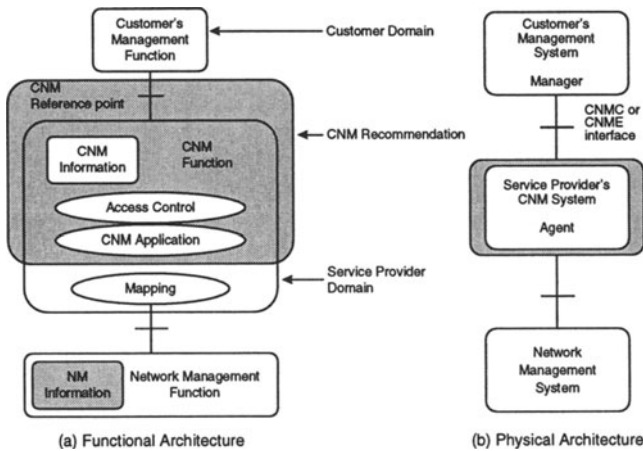


Figure 1 CNM Functional and Physical Architectures

As shown in Figure 1(a), the CNM function comprises several functional components: CNM information, access control, CNM application and mapping. The CNM functional architecture can be mapped onto the TMN functional architecture. The customer's management function and the CNM function correspond to the operations system function in TMN and the CNM reference point can be mapped onto the TMN X-reference point [1]. The CNM physical architecture (shown in Figure 1(b)) is composed of customer's network management system and a service provider's network management system. These two management systems are connected by the M3 interface.

CNM manager communicates with CNM agent within the public ATM network through the M3 interface. This communication takes place using SNMP over UDP. At the physical level, the communication may use an ATM UNI or dedicated circuit. The M3 specification is classified into two classes to allow public network providers to offer modular, incremental capabilities to meet different levels of customer's needs. The first class of M3 functions, Class I, requires that a public network provider to provide information on the configuration, fault and performance management of a specific customer's portion of a public ATM network. The second class of M3 functions, Class II, requires that a customer can request the addition, modification or deletion of virtual connections and subscription information in a public ATM network. In a nutshell, Class I provides monitoring capability only whereas Class II provides controlling capability as well.

Figure 2 shows an interworking structure between a CNMS and public NMS, in particular the interworking of a CNM agent and a public network manager. Recall that the public network management system is to provide either the Class I or Class II services to the customers. It is possible to provide the Class I service even if the CNM agent manages public network without interworking with the public network manager. However, it is absolutely necessary for the CNM agent to interwork with the public network manager if the Class II service is to be provided. This is mainly due to the control (i.e., add/modify/delete) capabilities associated with the Class II service provided to the customers who may potentially cause problems if they are not properly mediated by the public network manager.

Some work has been done for enabling the CNM agent to provide CNM services within the public networks. Most of these work thus far have proposed new information models or implementation methods to improve the performance of CNM manager and agent. However, the interworking mechanism between public network manager and CNM agent and the relationship between M3 MIB and M4 MIB in public

ATM network elements have not been studied. In the next section, we present our work on a framework which includes the interworking mechanism.

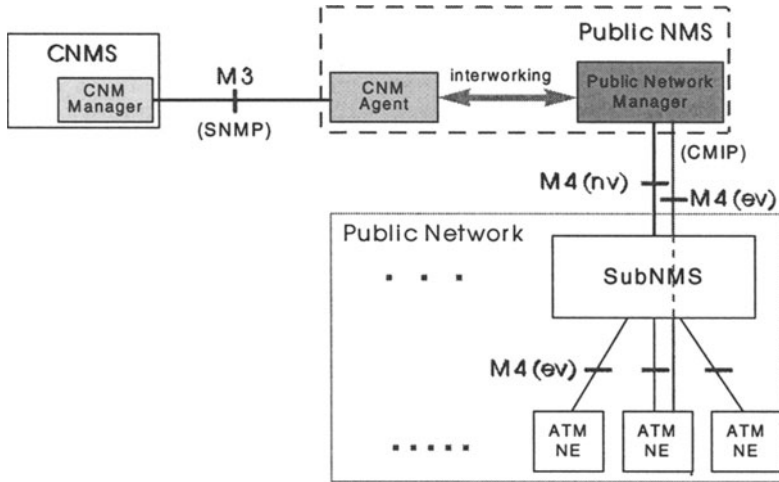


Figure 2 Interworking between CNM agent and public network manager

3 INTEGRATED PUBLIC NMS FOR SUPPORTING CNM SERVICES

Figure 3 illustrates the proposed architecture of a public NMS which integrates a public NMS and CNM agent. The integrated public NMS provides CNM services to CNM managers. The unique feature of the architecture is that it incorporates shared management knowledge (SMK) system [9, 12] to enable distributed management of CNM and public NMS.

In Korea, the government is trying to deregulate the communication service environment in a way that several regional public ATM network service providers are allowed to provide multimedia communication services. In this case, a customer may need to access several carrier management systems, i.e., public NM systems to manage its own VPN. Each public NM system may require access to several subordinate network management systems (i.e., subNMSs) to accomplish end-to-end management tasks. Consequently, VPN management requires a multi-level hierarchical manager-agent structure based on TMN where there are many-to-many relationships between

managers and agents. This requires a distributed management architecture and an efficient interaction mechanism for managers and agents which are distributed across the network and interconnected in a hierarchical way. This distributed management functionality is accomplished by CORBA-based SMK system as shown in Figure 3.

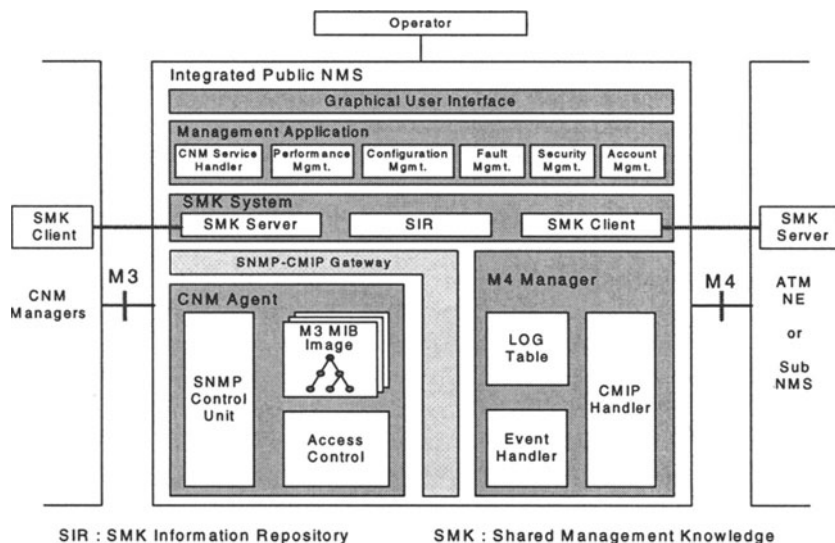


Figure 3 Integrated Public NMS Architecture

This architecture is composed of management applications, CNM agent, M4 manager, SNMP-CMIP gateway and SMK system. As shown in Figure 3, a CNM agent consists of M3 MIB image, Access Control, and SNMP Control Unit. This CNM agent realizes the CNM function modules described in Section 2. These major components of the integrated architecture are described in detail below.

3.1 CNM Agent

The CNM agent provides CNM services to CNM managers. As shown in Figure 2, the interaction between the CNM manager and CNM agent is defined by the M3 interface using SNMP as the management protocol. The SNMP control unit receives requests from a CNM manager and passes them to the M4 manager through the SNMP-CMIP gateway. It also handles replies and event reports from the M4 manager and forwards them to appropriate CNM managers. The access control is used to check the

permission of management requests and controls the access of management information. There may be multiple CNM managers accessing a single CNM agent, information on these CNM managers is also stored here. Since the actual MIB is located in the managed objects of the public network, M3 MIB image is used to reflect the actual MIB and is maintained by the CNM agent.

3.2 SNMP-CMIP Gateway

All the management operation requests of CNM agent for public NM applications are performed through this SNMP-CMIP functional module. This module translates SNMP requests of CNM agent into understandable forms of CMIP and forwards it to the M4 manager. Also, when the M4 manager receives CMIP replies and event reports, those intended for CNM managers must pass through this module, converting them into those that can be handled by the CNM agent. Our earlier work on the CMIP/SNMP gateway function [13] has been applied here.

3.3 SMK System

The exchange of management information between two function blocks in TMN requires a common view about the protocol knowledge, management functions, managed object classes and their instances, and authorized capabilities. Such information is collectively referred to as the shared management knowledge (SMK) and is defined in ITU-T M.3010 [8]. In order to perform management functions, the knowledge on the specific role as a manager or an agent is necessary in addition to the specific options for each function. It is necessary to discover the managed object classes and their instances for each management communication interface pair.

In Figure 3, the SMK server in the integrated public NMS performs the functionality of providing the information on CNM agent which is related to M3 MIB image, access control and SNMP control unit to the SMK client. SMK Information Repository (SIR) exists in the integrated public NMS, and it is the place for the storage of SMK information. The SMK server can, with direct access to the M3 MIB image, obtain the management information without accessing the SIR. In case where the contents of M3 MIB change very often, and the size of M3 MIB is relatively small, it may be more efficient to directly access the M3 MIB to get the SMK. However, if the contents of M3 MIB are mostly static, and the size of MIB is large, it may be more efficient to store them in SIR in advance. This can be achieved by making the SMK server access the SMK information and store them in SIR when the agent process is activated. As the SMK client requests the SMK information, the SIR is accessed for the provision of

required SMK information. In this case, M3 MIB is only accessed once each time the agent process is activated. The management of SMK information in a sense is similar to the view maintenance problem in distributed database management system. The suitable SMK management strategy may depend on the update frequency and size of M3 MIB, the complexity of M3 MIB access functionality, and of course the semantic correctness of SMK update operations [9]. We have designed and implemented this SMK management system which are described in more detail in Section 4.

3.4 M4 Manager

The M4 manager performs management interactions with public network management (M4) agents on behalf of the public network management applications as well as the CNM agent. Management requests from the public NMS come from the public NM applications directly whereas the CNM requests comes from the CNM agent via the SNMP-CMIP gateway. These requests must be distinguished by the M4 manager so that the replies and event reports from M4 agents can be delivered to appropriate managers. The LOG table is used to log all the interactions that take place within the integrated manager.

One of the important subcomponents within the M4 manager is the event handler. Event notification capability provides a customer with the ability to receive unsolicited events from public networks upon detection of abnormal conditions related to the customer's access. Event notifications could include changes in the status of ATM UNI configuration parameters, PVC configuration parameters, or access authorization failures. Event notifications are helpful to the customer for isolating failures in the customer's network. M4 agents report status changes and faults of specific customer-related NEs to M4 manager. When NMS receives an event report from an M4 agent, it determines whether the received event report relates to a specific customer's portion of public ATM network. If so, the event report is transmitted to the public network management application and the CNM agent. Thereafter, the CNM agent delivers it to the appropriate CNM manager by using an SNMP Trap. Figure 4 shows the procedure of event report handling.

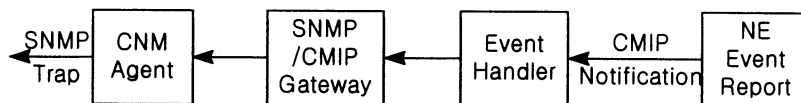


Figure 4 Procedure of event-report handling

Here, we describe the concept of M3 MIB Image which has been incorporated into the integrated public NMS in order to increase performance. Consider the case where a public network provider provides the M3 Class I service. When a CNM agent receives a SNMP GET request, a normal processing would require that this request be passed to the M4 manager via the SNMP-CMIP gateway. Then, the M4 manager would get appropriate information from an M4 agent. This process may take a long time for a simple GET request. Instead, in order to reduce the response time, we have installed the M3 MIB image within the CNM agent and it is updated periodically by the M4 manager. Basically, the M4 manager periodically polls the customer-related M4 MIB from the M4 agent and updates the M3 MIB image. When a SNMP request is received, the value of M3 MIB image can be returned immediately. When a specific event occurs which requires modification to the M3 MIB image, the M4 manager transfers the event to CNM agent through event handler directly. The procedure of the Class I service with the use of M3 MIB Image is illustrated in Figure 5.

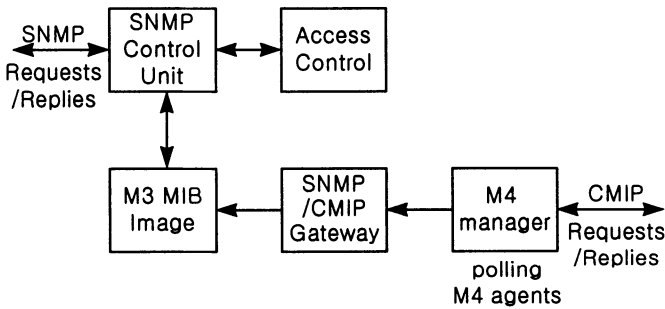


Figure 5 Procedure of Class I services

When a CNM agent receives a SNMP SET request for the update of management information, in the case that network provider supports the M3 class II service, it invokes the M4 manager for the provision of update services. The M4 manager, in response to the request from the CNM agent, stores service profiles in LOG table and requests M4 agents to update the related management information in ATM NE directly or via subNMS indirectly. The M4 manager is responsible for modification, addition and deletion of connections and subscription information in public ATM networks according to customer's requests. The response of M4 agent is transferred to a M4 manager and then to CNM agent based on the information stored in LOG table. The procedure of providing the Class II service is illustrated in Figure 6.

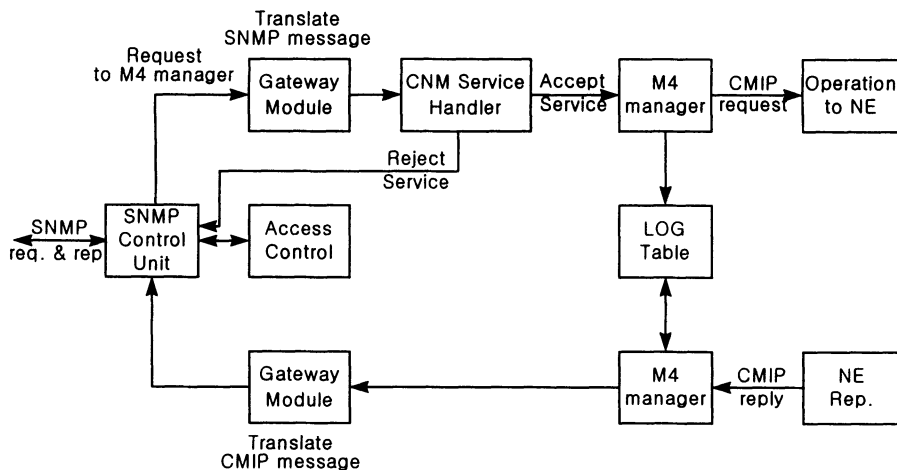


Figure 6 Procedure of Class II services

4 PROTOTYPE IMPLEMENTATION

Figure 7 illustrates the architecture of a prototype implementation of the integrated CNM agent and M4 manager, M4 agent and SMK system. We used the OSIMIS platform which was developed at UCL [10] for implementing manager-agent interactions. It provides an environment for the development for management applications which hides the details of the underlying management service through the object-oriented application program interfaces (APIs) and allows implementers to concentrate on the intelligence to be built into management applications rather than the mechanics of management service/protocol access. It also provides an implementation of the Internet SNMP over UDP using the ISODE ASN.1 tools and a generic CMIS to SNMP application gateway driven by a translator between SNMP and OSI GDMO MIBs.

In the SMK system, the CMIS scoping in the M4 interface and SNMP in the M3 interface is not a suitable technique for the identification of complete set of MO classes, and a more generic identification mechanism was needed [9]. Thus, we used the CORBA-based approach. The SMK client object of a management application is associated with the SMK server object at the managed system through the communication facility provided by the underlying ORB. The SMK information can be

accessed by using the interface which is described in Interface Description Language (IDL). CORBA IDL and ORB implementation embedded in ORBeline [11] from PostModern Computing Technologies was used as a CORBA implementation. The communication mechanism of the CORBA ORB enables the location transparency between distributed objects to be supported, and this created the dynamic, efficient distributed processing environment.

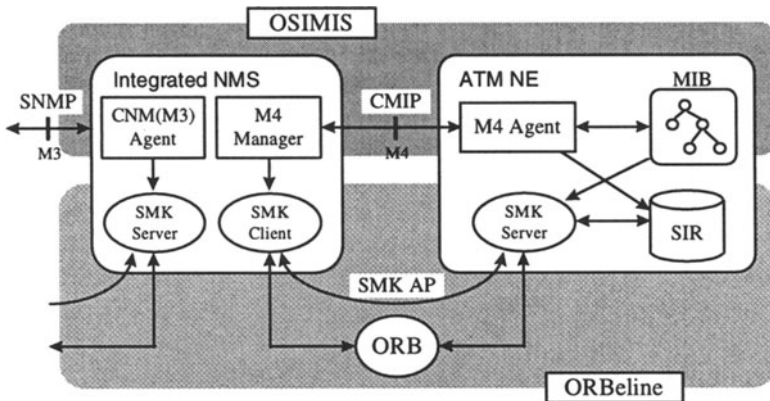


Figure 7 Prototype implementation using OSIMIS and ORBeline

The definition of SMK interface for a manager is shown in below. This SMK interface is one that a manager invokes to acquire SMK. A management agent implements SMK server and provides SMK information. A manager accesses information on SMK through this interface. SMK interface is also described in IDL and corresponds to a CORBA object. IDL interface has a following structure.

```
Interface <identifier> [<inheritance_spec>]
{
    <interface_definition;>*
};
```

<interface_definition:> includes the declaration of type, constant, exception, operation, and attribute. In the declaration of operations, the operation names and parameter types are specified below. The operation `Get_ProtocolInfo` identifies the type of management information protocol: CMIP or SNMP. The operation `Get_AccessPolicy` checks the access policy upon reception of request from the SMK client. The meaning of most

operational primitives are self-explanatory. The details of these specifications can be found in [9].

The SMK server process can be generated by forking the agent process. An agent generates a child process after initializing the MIB when running the program. The child process creates the SMK server object, and brings in the reference point to the MIB. The design algorithm for creation of SMK server object process at an agent is described below.

Algorithm Server-Process (environment_variable)

Begin

```
initializeSyntaxes(syntaxes);
agent.initialize(environment_variable, service);
MO::initializeMIB(MIB_INIT_FILE);

// create SMK server process
if (fork() == 0) {
    // create SMK server object
    SMKObject *smkObj = new SMKObject(object_name);
    // get reference pointer to MIB
    MO **root = MO::getRoot();
    smkObj->GetMO(root->getWholeSubtree());
    CORBA::BOA::impl_is_ready();
}
coordinator.listen();
```

End

5 CONCLUSION

ATM network management system is absolutely necessary for providing the future communication services in ATM networks. The research in ATM NM is being carried out by many researchers as well as by standardization organizations. In this paper, we focused on the M3 and M4 interfaces for the purpose of CNM service provisioning and public network management.

As a solution to the problem of managing public ATM networks by the CNM systems, we proposed a VPN management architecture of an integrated public NMS supporting CNM services. The proposed architecture has functional modules for supporting both CNM functions and public network management applications. The prototype implementation of our design was made by using the OSIMIS platform and ORBeline.

For future work, we plan to investigate the mechanism of information exchange between public NMSs through the TMN X interface and interworking with CNM agents in the case where the customer's PVCs traverse multiple public networks. Another future work is to do a performance analysis of our prototype implementation. We plan to deploy our prototype in an operational ATM testbed of Korea Telecom.

6 REFERENCES

- [1] Jacqueline Aronsheim-Grotsch, "Customer Network Management CNM," Proceedings of NOMS'96, April 1996, pp. 339-348.
- [2] Open Management Group, "The Common Object Request Broker: Architecture and Specification," December 1993.
- [3] Henry J. Fowler, "TMN-Based Broadband ATM Network Management," IEEE Communication Magazine, 33(3):74-79, March 1995.
- [4] ITU-T Recommendation X.160, "Architecture for Customer Network Management Service for Public Data Network", 1994.
- [5] Michael Hinchliffe and Nigel Cook, "Customer Network Management and ATM Networks," Proceedings of NOMS'96, April 1996, pp. 155-164.
- [6] Miyoshi Hanaki, "LAN/WAN Management Integration using ATM CNM Interface," Proceedings of NOMS'96, April 1996, pp. 12-21.
- [7] ATM Forum M3 Specification Revision 1.03, "Customer Network Management for ATM Public Network Service," 1994.
- [8] ITU-T Draft Recommendation M.3010, "Principles of Telecommunication Management Network," 1995.
- [9] J. T. Park, S. H. Ha and J. W. Hong, "Design and Implementation of a CORBA-Based TMN SMK System," Proc. of NOMS'96, April 1996, pp. 64-74.
- [10] G. Pavlou, "The OSIMIS TMN Platform: Support for Multiple Technology Integrated Management Systems," Proc. of RACE IS&N'93, November 1993.
- [11] PostModern Computing Technologies, ORBeline 2.0 User's Guide, 1996.
- [12] J. T. Park and J. W. Hong, "Implementation and Performance of a TMN SMK System", Proc. of DSOM'96, L'Aquila, Italy, October 1996.
- [13] J. T. Park, Y. W. Choi and J. D. Kim, "The Integration of OSI Network Management and TCP/IP Internet Management using SNMP," Proc. of IEEE First International Workshop on Systems Management, April 1993, pp. 145-154.

7 BIOGRAPHIES

Jong-Tae Park received the BS degree from Kyungpook National University, Korea and the MS degree from Seoul National University, Korea. He received PhD from Computer Science and Engineering, University of Michigan, in 1987. From 1987 to 1988, he was at AT&T Bell Labs, working on network management and service provisioning. Since 1989, he has been working at the School of Electronic and Electrical Engineering at Kyungpook National University, Korea, where he is now an associate professor. His research interests include TMN, distributed DBMS, network and resource management in ATM networks, multimedia communication, and personal communication services.

Jae-Hong Lee received the BS and the MS degrees in Electronic Engineering from Kyungpook National University, Korea, in 1995 and 1997, respectively. He is currently working as a researcher at LG Electronics, An-Yang, Korea. His research interests include ATM networks and network management.

James Won-Ki Hong is an assistant professor in the Dept. of Computer Science and Engineering, POSTECH, Pohang, Korea. He has been with POSTECH since May 1995. Prior to joining POSTECH, he was a research professor in the Dept. of Computer Science, University of Western Ontario, London, Canada, where he worked on the CORDS project and MANDAS project. Dr. Hong received the BSc and MSc degrees from the University of Western Ontario in 1983 and 1985, respectively, and the PhD degree from the University of Waterloo, Waterloo, Canada in 1991. His research interests include network and systems management, distributed computing and multimedia systems. He is a member of IEEE and ACM.

Young-Myoung Kim is a Member of Technical Staff in Telecommunication Network Research Lab (TNRL) of Korea Telecom, Korea. He has been with Korea Telecom since 1989. He received the MS degree from KAIST in 1989. His research interests include Telecommunications Network Management. He is an associate member of IEEE Communications and Computer Society.

Seong-Beom Kim is a Principal Member of Technical Staff in Telecommunications Network Research Lab (TNRL) of Korea Telecom, Korea. He has been with Korea Telecom since 1984. Prior to joining Korea Telecom, he was a member of research staff in ETRI. He received the BSc and MSc degrees from Hanyang University, Seoul, Korea, in 1980 and 1986, respectively. His research interests include telecommunication network and system management, concurrency control in DBMS