

Integrated Network Management

with TINA Manager and TMN & SNMP Agent

Ho-Cheal Kim*, Young-Tak Kim**

Dept. of Multimedia & Communication Eng., Graduate School, Yeungnam Univ.*

Dept. of Information & Communication Eng., Graduate School, Yeungnam Univ.**

Abstract

The next generation highspeed networks will be composed of the ATM/SDH-based highspeed wired transport network, IMT-2000-based wireless transport network and the IP-based internetworking systems. Since the provisioning of broadband mobile multimedia services is the most important issue in the next generation highspeed networks, the efficient network management architecture that can manage efficiently the network resources is essential. The management functions of these network element systems are currently implemented with various heterogeneous technologies, such as TMN (Telecommunications Management Network) and SNMP (Simple Network Management Protocol). In order to integrate the heterogeneous management functions efficiently, the distributed network management architecture such as TINA (Telecommunications Information Networking Architecture) is most appropriate. In this paper, we propose an integrated network management architecture for TINA manager and TMN&SNMP agent, where the TINA-based EML (Element Management Layer) subnetwork management operations are mapped into the related management operations of the TMN/GDMO-based or SNMP-based NE (Network Element) agent. The proposed interworking architecture is simpler than JIDM (Joint Inter-Domain Management), and the processing overhead is minimized.

1. Integration of TINA, TMN and SNMP

Currently, the most popular management technologies are CMIP&GDMO-based TMN and SNMP, which are implemented on network element systems such as SONET/SDH-based transmission systems, ATM switches, IP routers, and workstations. These management technologies by themselves cannot support efficiently next generation telecommunication services, such as broadband mobile communication services, which require more flexible resource management. We need distributed network management architecture such as TINA to integrate efficiently the heterogeneous

management functions for NE (Network Element) agents, while providing flexible network-view resource management. But, TMN and SNMP-based network systems cannot communicate directly with the TINA management components because they do not support the distributed environment of TINA. To integrate the management functions of TINA and TMN&SNMP, we must consider an efficient interworking architecture.

The integration of heterogeneous management functions with the TINA manager can be explained as two viewpoints: the implementation and the support of next generation telecommunication services.

The creation of a new telecommunication service

with now-a-day's technologies spend lots of costs and times because of the underlying dependency and architectural complexity[1]. To overcome this problem, the TINA-C has started. Since the TINA's intention is to use the recent advances in distributed computing, distributed communication environment, object oriented analysis and design, improvement of interoperability, reuse of software and specifications, and flexible placement of software on nodes, the TINA-based technology can reduce the costs and times of traditional service implementation technologies[2].

The TINA has been designed to satisfy the needs of traditional voice services, future interactive multimedia services, information services, and operations and management services, and will provide the flexibility to operate services over a wide variety of technologies. For this reason, the TINA manager has advantages both commercially and technologically to the next generation telecommunication services.[2]

Especially in TINA, the connection management is defined as a separated management function, while it is defined as part of configuration management in TMN. This connection management function is suitable to the management of mobile and multimedia communication services.

In this paper, we first analyze the integration scenarios between TINA manager and TMN agents or SNMP agents, and then propose an integrating architecture.

2. Integration Scenarios

2.1 Horizontal vs Vertical Interworking

In order to integrate the heterogeneous management functions of NE agents with TINA-based manager, we can consider two types interworking scenarios. The first one is horizontal interworking between the same management layer as shown in ①,②,③ and ④ of Figure 1.

This scenario is useful when we have already installed CMIP/GDMO-based TMN or SNMP-based management domains separately and a new TINA based management domain is added.

The other scenario is a vertical interworking between the different management layers. This scenario is useful when different telecommunication management technologies coexist within one management system. The vertical interworking scenario can be implemented at three interface points. The first interface point exists between the TINA SML components and the TMN NMS or SNMP as shown in ⑤ and ⑧ of Figure 1. The second interface point exists between the TINA NML components and the TMN EMS as shown in ⑥ of Figure 1. The last interface point exists between the TINA EML components and the TMN NE agent or the SNMP NE agent as shown in ⑦ and ⑨ of Figure 1.

The use of the interface point between TINA SML and TMN NMS can take advantage of the efficient telecommunication management capability of TMN and the flexible telecommunication service creation capability of TINA.

The use of the interface point between TINA NML components and TMN EMS is not acceptable because the TINA NML components are not boundary interfaces of TINA resources layer.

The use of the interface point between TINA EML components and TMN or SNMP NE agent is useful if there is no network manager functions based on TMN or SNMP and when more flexible connection management is required such as mobile and multimedia services.

We think that the last vertical interworking scenario is the most acceptable one to integrate heterogeneous network elements having different management function with TINA manager. So, we propose an interworking architecture to integrate the TINA-based manager and network element based on CMIP/GDMO-based TMN or SNMP.

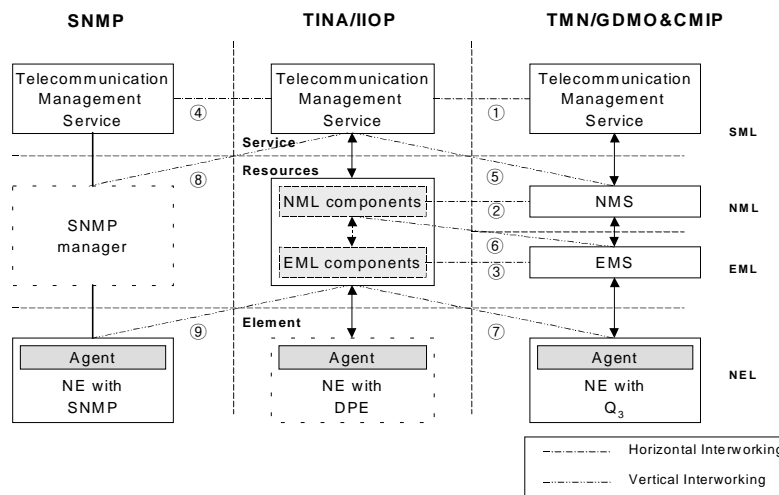


Figure 1 : Interworking scenarios between TINA and TMN/SNMP

2.2 Interworking between TINA Manager and TMN Agent

The interworking function between TINA EML components and NE agent based on TMN can be implemented as i)an individual TINA computational component, ii)an embedded type in the TINA EML components, and iii)a modified embedded type. Figure 2 shows these three types interworking function architectures.

The use of individual computational component type to interwork each NE agent has software architectural merit that the NE agent can be treated as a distributed TINA computational component by the interworking component. But, the interworking function must supports all requests of distributed TINA EML components and provides inter-protocol translation between IOP and Q_3 CMIP interface. This type can increase the operation complexity of and the processing overhead of the interworking component.

The use of embedded type to interwork each NE agent is an approach to simplify the architecture of the interworking function by including functions separately to each EML component. Since, this type interworking supports only one TINA EML computational

component's requests, the architecture can be relatively simple. And, this type interworking can take an additional merit that reduces the overhead of distributed environment by combining the TINA EML component and interworking function. But, there are two shortcomings: the growth of TINA EML component size because of the CMISE protocol stack, and the growth of the association count for Q_3 interface.

In order to overcome these shortcomings of the embedded type, we modified the function structure. The modified embedded type can take both advantage of individual type and embedded type by splitting protocol translation from EML computational components. The split protocol translator has no responsibility of management functions. This modified embedded type look like JIDM but the management function mechanism is basically different each other.

2.3 Interworking between TINA Manager and SNMP Agent

The interworking architecture to integrate SNMP agent is apparently similar to the case of TMN agent. But, there are some kinds of differences that must be considered.

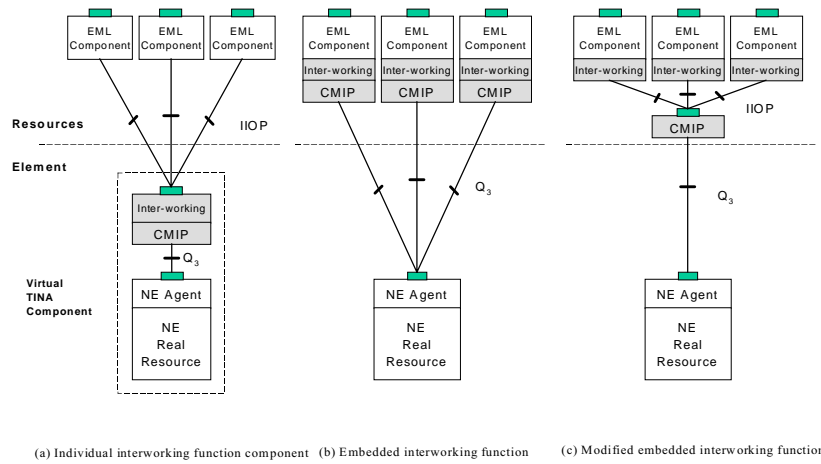


Figure 2 : Three types of interworking function

The first one is the difference of the managed object within NE agent. In the TMN agent, the managed objects are exist as object instance of GDMO, which is base on object orientation concept, but in the SNMP NE agent, the managed objects are exist as pair of object ID and scalar value.

The second consideration is the difference of communication characteristic, connection oriented or connectionless. The SNMP is connectionless while the CMIP, which is the protocol of TMN, is connection oriented. The third one is the difference of the number of management message type. The CMIP has more management messages than the SNMP. This means that the CMIP/GDMO-based TMN can support more management functions than SNMP.

These differences can be matched in embedded function of our proposed interworking architecture by implementing managed object mapping and management functional mapping.

2.4 Comparison with JIDM architecture

The major differences of JIDM(Joint Inter-Domain Management), which is a standard for the CORBA (Common Object Resource Broker Architecture) - based manager, and proposed interworking architecture is the difference of manager concept between TINA-based

manager and CORBA-based manager and difference of managed object treatment. The JIDM is well defined to interwork distributed TMN manager and TMN agent or distributed SNMP manager and SNMP agent. But, to apply JIDM to TINA-based management system some modification required. Because the TINA management has hierarchical component structure and specific managed object.

3. The Structure of the EML Component

In this section, we explain the functional requirement of interworking architecture. The EML computational component must be implemented as two models, TMN and SNMP. In TINA management architecture, EML computational component can manage only a NE. This means that the specific EML computational component is needed to manage the specific NE.

3.1 TINA EML Component with Q_3 Interface

The architecture of TINA EML component with Q_3 interface using the modified embedded type can be divided into two functional sub-layers as figure 3. The EML interface sub-layer provides the original TINA EML component interfaces. This sub-layer supports

management functions to upper network management layer computational components. Another sub-layer of the EML component performs abstract translations that provides following function:

- Manipulates managed objects of EML and performs the mapping of managed objects and management functions with the related TMN NE agent.
- Translates management functions from network-view functions to network-element-view functions when a certain EML management function is needed to perform NEL management functions.
- Sends notification messages to NML computational components through the notification reference in EML interfaces sub-layer.

The translated management messages will be delivered to the TMN NE agent across the protocol translator. The protocol translator provides association with TMN NE agent and provides CMISE interfaces to abstract translation sub-layer in TINA EML computational components.

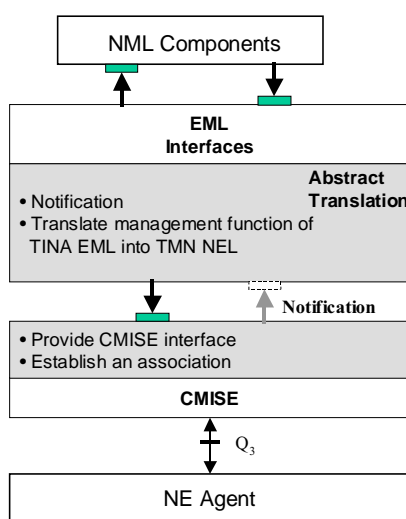


Figure 3 : Functional architecture of modified embedded type

3.2 TINA EML Component with SNMP Interface

Most of the internet systems are managed by SNMP[3]. In order to interwork the TINA manager and the SNMP agent, we can consider two integration scenarios: interworking via TMN Q_3 adaptor or direct interworking.

The TINA management architecture adapts and generalizes the concept of the TMN[4]. So, the TINA EML components can be constructed with unique interworking architecture if the SNMP agent will be interworked via the CMIP/GDMO-based TMN NE agent as shown in case 1 of Figure 4. This can be implemented by the mapping of the CMIP/GDMO-based NE managed object and the related SNMP NE MIB.

But, the interworking of the SNMP NE agent through the CMIP/GDMO-based TMN NE agent increases the translation overhead and reduces the performance of management system. So, to interwork efficiently the SNMP NE agent, the TINA EML computational component is needed to be interworked directly as shown in case 2 of Figure 4.

If we implement the SNMP interworking function, the TINA EML components architecture is similar to the TMN interworking architecture as Figure 3. The Q_3 interface will be replaced to UDP for SNMP and the protocol translator interface will be replaced to SNMP management message support. Since the managed objects and management messages are different from TMN, the abstract translation sub-layer must be modified to support SNMP.

4. Implementation experience of Integrated Management Functions

4.1 Connection Management Integration

The connection management of the TINA is divided into four parts: communication session, connectivity

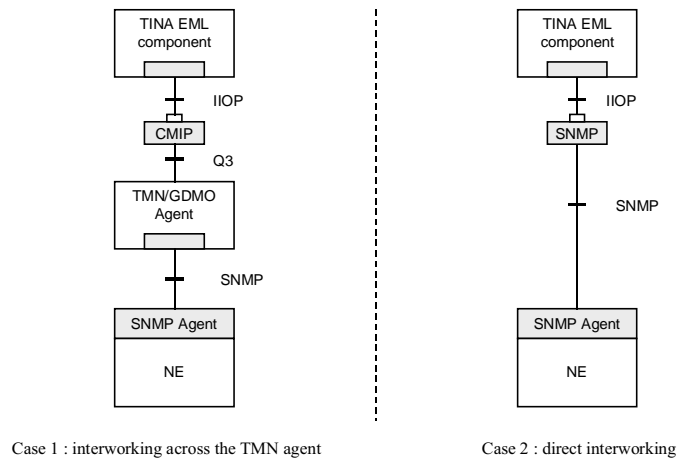


Figure 4 : Integration scenarios of SNMP

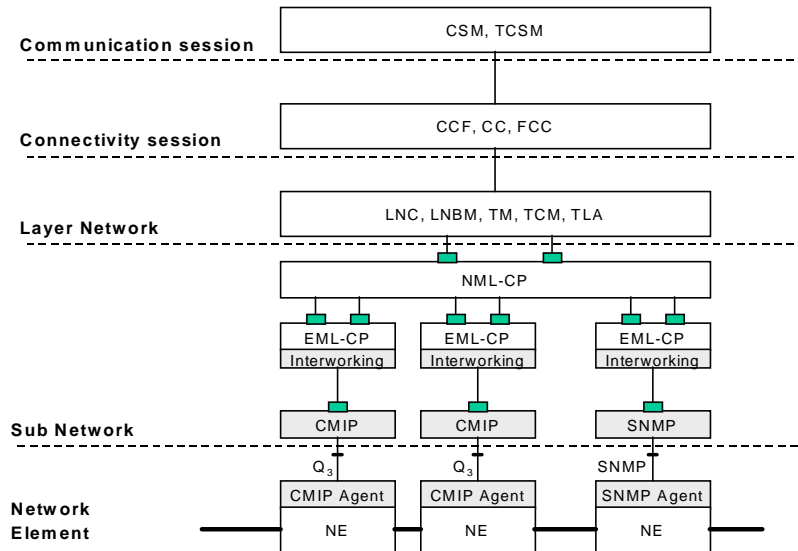


Figure 5 : Connection management integration with TINA interworking functions

session, layer network, and subnetwork[5]. Our concern of the integration is the subnetwork with EML components. The EML-CP is the lowest subnetwork of the TINA connection management. And, the network elements coexist with the CMIP/DGMO-based TMN NE and the SNMP based NE. Because each EML-CP manages individual NE, we can regard an NE is a subnetwork.

If heterogeneous NE agents are mixed in equal management domain, the TINA connection management can easily integrate with TMN oriented EML-CP and

SNMP oriented EML-CP as Figure 5. Another consideration is to integrate the isolated subnetworks where the CMIP/GDMO-based TMN NEs and the SNMP-based NEs are separately managed. In this case, each NML subnetwork component, which manages number of EML-CPs of subnetworks can be integrated by upper NML component.

When the integration is completed, the CSM (Communication Session Manager) can receive requests from service layer components. For each network connection establish requirement, the CSM interacts with

Table 1 : Connection management object mapping

TINA Quasi-GDMO		TMN GDMO		SNMP MIB	
Object	Attribute	MO	Attribute	MIB	MIB Group
Subnetwork	characteristicInfo	atmFabric	atmFabricId	pathStatus chanrStatus	pathRouteTable chanRouteTable
Subnetwork Connection	connectionTopology	atmCrossConnection	crossConnectionId administrativeState operationalState directionality fromTermination toTermination	pathrStatus chanrStatus pathrInputPort pathrInputVPI chanrInputPort chanrInputVPI chanrInputVCI pathrOutputPort pathrOutputVPI chanrOutputPort chanrOutputVPI chanrOutputVCI	pathRouteTable chanRouteTable
Network Connection Termination Point	directionality	vcCTPBidirectional vpCTPBidirectional	vcCTPId vpCTPId segmentEndPoint	pathVPI chanVCI	pathTable channelTable
Link Termination Point	directionality	-	-	-	-
Edge	directionality	-	-	-	-

the CC(Connection Coordinator) to establish connections between network access points[6]. And, the CC requests the LNC(Layer Network Coordinator) to setup the connection in that layer network[6]. An LNC requests CPs(Connection Performer) to establish subnetwork connections[6]. A CP is responsible for establishing connections across subnetwork. The lowest CP is EML-CP, witch has interworking function we proposed. The EML-CP will interact with the NE agent through Q₃ or SNMP interface.

The EML-CP manages several managed objects of network view to create subnetwork connections[7]. These managed objects are related functionally with NE agent's managed objects. For the integration of heterogeneous management environment these related managed objects must have a management function translation mechanism from higher layer to lower layer. There are five managed objects related to EML-CP component as shown in Table 1. Three managed objects have related managed object in CMIP/GDMO-based TMN NE agent or the SNMP NE agent but others have not. Especially, Edge is a unique managed object only defined in TINA connection management for intra-

subnetwork connection.

The TMN NE agent and the SNMP NE agent do not have dynamic connection managed object like the TINA Edge managed object. So, they do not satisfy the requirement of the mobile and the interactive multimedia services. For the next generation telecommunication services, we need the capability to support the mobile and the interactive multimedia services. The capability to support the next generation telecommunication services of the TMN&SNMP-based network system can be emulated restrictively through integration the TMN and SNMP NE agent with TINA manager.

The subnetwork(SNW) managed object of the EML-CP represents a switch or a cross connect, while the subnetwork connection(SNC) managed object represents a connection of the intra-subnetwork. A Network Connection Termination Point (NWCTP) is an end point of subnetwork connection. An Edge represents an end point of the intra-subnetwork.

The connection establish sequence of TINA EML computational component is listed below:

Step 1 : Creates a SNC and Root Edge

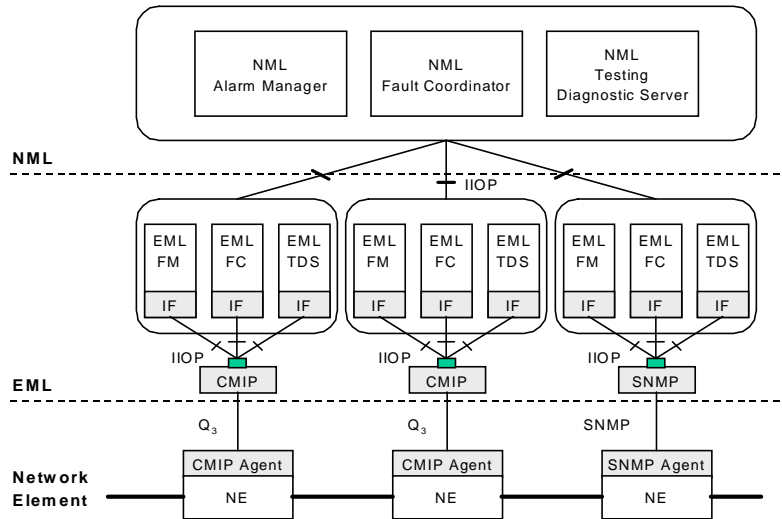


Figure 6 : Fault management architecture with interaction function

Step 2 : Binds Root Edge to source NWCTP

Step 3 : Creates Leaf Edge

Step 4 : Binds Leaf Edge to sink NWCTP

The connection establishment of TMN is performed by a M-ACTION connect message transfer to AtmFabric object for ATM switch. In case of SNMP, the connection establishment is performed by SET-Request messages to SNMP NE agent.

In EML-CP abstract translation sub-layer, the sequence of connection establishment is monitored. When the step 4 is reached abstract translation sub-layer send M-ACTION message or SET-Request message to NE agent.

4.2 Fault Management Integration

The fault management of TINA consist of alarm surveillance, fault localization, fault correction, testing/diagnostic, and trouble administration[6]. And the fault management service of network resources is provided by the interaction of the computational objects such as alarm manager, fault coordinator, and test/diagnostic server.

The difference of fault management components

from connection management components of the TINA is the direction of messages. The integration of the coexisted NE agent for fault management is same as the integration of the connection management. But, management message flow is in reverse direction compared to the connection management; the fault notification message is generated by NE agent and sent to EML component. To solve this reverse management message flow in integrated environment, the EML-FM must have notification server mechanism suitable to each management protocol.

Figure 6 shows the interworking of EML components, NML components, and NE agents with Q_3 interface or SNMP. If the EML-AM receives alarm from NE agent through interworking function for the TMN/GDMO or SNMP, it records log and performs alarm filtering and alarm log recording for alarm analysis. It then analyzes correlation of each alarm with the EML-FC and forward alarms to the NML-AM. The EML-FC interacts with the EML-AM to find the cause of fault, and if possible, performs fault correction according to a pre-determined fault restoration algorithm and invokes the EML-TDS. Processing procedure of NML components is same as EML components.

Table 2 : Fault management object mapping

TINA Quasi-GDMO		TMN GDMO		SNMP MIB	
Object	Attribute	MO	Attribute	MIB	MIB Group
AlarmRecord	probableCause perceivedSeverity specificproblems	AlarmRecord	probableCause perceivedSeverity specificproblems(c) backedUpStatus	swAlarmType atmfM4Alarm- Severity	swAlarmTable atmfM4Alarm- SeverityTaxtual Convention
AlarmSeverity- AssignProfile	alarmSeverityAssign- mentList	AlarmSeverity Assignment Profile	alarmSeverityAssign- mentProfileId alarmSeverityAssign- mentList	atmfM4Alarm- SevProfileIndex atmfM4Alarm- SevProfileEntry	atmfM4Alarm SeverityProfile Table
CurrentAlarm- SummaryControl	alarmStatusList perceivedSeverityList probableCauseList	CurrentAlarm- SummaryControl	alarmStatusList perceivedSeverityList probableCauseList currentAlarmSummary- ControlId objectList		
FaultManageable	alarmStatus currentProblemList communicationsAlarm(n) qualityofserviceAlarm(n)	Managed Element	alarmStatus currentProblemList(c) communicationsAlarm(n) qualityofserviceAlarm(n)	swAlarmStatus swAlarmType	swAlarmTable

The AlarmRecord managed object of the EML-FM (Fault Manager) represents the alarm information stored in a log, and the AlarmSeverityAssignmentProfile specifies the assignment of alarm severity to each type of alarms; each profile object may specify different severity assignment. The FaultManageable managed object represents the management information that a network resource has to provide so that it can be subject to fault management.

Each managed object of TINA is mapped into GDMO of TMN NE agent and MIB of SNMP NE agent. Table 2 shows object mapping for the managed objects of the TINA manager and TMN or SNMP agent managed objects. Most managed objects in TINA are mapped into GDMO in TMN gracefully and vice versa, but it is impossible to map SNMP MIBs exactly.

For example, the CurrentAlarmSummaryControl which specifies the criteria for the generation of a current alarm summary report and FaultManagementDomain which represents a set of FaultManageable objects that is controlled by a fault management function can not be mapped to SNMP objects.

In this paper we used several SNMP-MIBs, TMN/GDMO and TINA/Quasi-GDMO: af-nm-0095 MIB[8], FORE-SWITCH MIB[9], rfc1695 MIB[10], X.721[11], I.751[12], M.3100[13] for TMN and NRM

(Network Resource Information Management) specification version 2.2[5] for TINA.

Each mapping is performed not in NML (Network Management Layer) but in EML (Element Management Layer), because there is no SNMP MIB for Network Management.

5. Conclusion

In this paper, we analyzed the integration scenarios between TINA manager and TMN&SNMP agents. From the functional analysis, we concluded that the most appropriate interworking level to integrate the TINA manager and TMN or ANMP agent for the management of multimedia or mobile communication services is TINA EML component that has a one-to-one correspondence with NE. According to the analysis, we proposed an interworking function, which is located in between the TINA EML components and TMN or SNMP NE agent. The proposed interworking architecture is simple, and the processing overhead of TINA EML components is minimized.

As an example of the interworking, we explained the TINA fault and connection management architecture based on the proposed interworking architecture. The proposed interworking architecture can be efficiently

used in the development of integrated network management functions for next generation internet and highspeed transport networks.

[References]

- [1] Steve Vinoski, "CORBA: Integrating Diverse Applications within Distributed Heterogeneous Environments", IEEE Communications Magazine, Feb. 1997, pp. 46-55.
- [2] TINA-C TB_MDC.018_1.0_94, "Overall Concepts and Principles of TINA", Feb. 1995.
- [3] X/Open P509, "Inter-domain Management: Specification Translation", May. 1997.
- [4] TINA-C TB_GN.010_2.0_94, "Management Architecture", Dec. 1994.
- [5] TINA-C NCS_DRAFT_v2.2_97_12_20, "Network Components Specification Version 2.2 Final Draft", Dec. 1997.
- [6] TINA-C NRA_v3.0_97_02_10, "Network Resource Architecture", Feb. 1997.
- [7] TINA-C NRIM_v3.0_97_12_17, "Network Management Information Model Specification version 3.0", Dec. 1997.
- [8] ATM Forum AF-NM-0095.001, "SNMP M4 Network Element View MIB", Jul. 1998.
- [9] FORE system, "ASX-200 ATM switch MIB", Jun. 1996.
- [10] IETF RFC-1695, "Definition of Managed Object for ATM Management Version 8.0 using SMIV2", Aug. 1994.
- [11] ITU-T Rec. I.721, "Information Technology-Open Systems Interconnection-Structure of Management Information : Definition of Management Information", Feb. 1992.
- [12] ITU-T Rec. I.751, "Asynchronous Transfer Mode Management of The Network Element View", Mar. 1996.
- [13] ITU-T Rec. M.3100, "Generic Network Information

Model", Jul. 1996.



Young-Tak Kim

1984. 2 : received B.S. in Electronic Eng. from Yeungnam university

1986. 2 : received M.D. in Electric & Electronic Eng. from KAIST

1990. 2 : received Ph.D. in Electric & Electronic Eng. from KAIST

1990.3-1994.8 : worked as a Member of Technical Staff & Section Leader of Telecommunication Network Research Lab. in Korea Telecom.

1994.9- : Assistant professor and associate professor in the Dept. of Information & Communication Eng., collage of Eng., Yeungnam Univ.

Research Area : ATM/B-ISDN, IN, TINA, TMN, and NGI



Ho-Cheal Kim

1989. 2 : received B.S. in Electronic Eng. from Kyungpook National University.

1999. 2 : received M.D. in Multimedia & Communication Eng.

from Yeungnam Univ.

1999- : under Ph.D. course in Multimedia & Communication Eng. of Yeungnam Univ.

1989-1995 : worked as a Junior Research worker in SAMSUNG Electron Devices Research Center.

1995-1996 : worked in SAMSUNG Commercial Motors Tech. Support Team.

Research Area : TMN, Distributed management, NGI, and ATM/B-ISDN