論文 98-23-5-22

Implementation of TMN Agent for ATM switch :Considering Integration of Agent into ATM switch

Hee-San Hwang*, Byung-Yoon Lee*, Gil-Heang Lee*, Wang-Don Woo* *Regular Members*

ATM 교환기를 위한 TMN 관리 대행 시스템의 구현

正會員 황 희 산*, 이 병 윤*, 이 길 행*, 우 왕 돈*

ABSTRACT

There are many implementation methods according to models integrating TMN Agent into an ATM switch. In this paper, we evaluate the integrating models for integrating the Agent into an ATM switch in the aspects of the size of MIB(Management Information Base), the internal protocol profiles and the facility of implementation. Based on the evaluation, we choose an integrating model and implement the Agent. To ensure merit of the model, we propose an interface for exchanging management information between the Agent and an ATM switch. We also show the feasibility of our Agent system through some filed testes for the average processing time.

요 약

ATM 교환기 관리를 위한 TMN 시스템의 개발에 있어, ATM 교환기와 관리 대행자간의 구조적 결합 방법에 따라 여러 가지 형태의 구현 방법이 존재할 수 있다. Embedded Agent와 Gateway Agent와 같은 분류가 그 대표적인 것이라 하겠다. 지금까지의 TMN 시스템의 개발에 있어, 이들 구현 방법들에 대한 객관적 평가를 통한 적절한 구현 모델의 선택이 거의 이루어지지 않았다.

본 논문에서는 ATM 교환기를 위한 관리 대행자 시스템의 개발에 있어, 가장 먼저 선행되어져야 할 구현 모델 선택을 위해, 5가지의 구현 모델을 설정하고, 각 구현 모델에 대해 MIB 구현 측면, ATM 교환기 내부 프로토콜 측면, 구현의 용이성 측면을 평가한 후, 가장 적절한 구현 모델 하나를 선택하였다. 또한 이 선택된 모델을 기초로 관리 대행 시스템을 구현하였고, 구현 모델의 이점을 극대화하기 위한 관리 대행자와 교환기간의 인터페이스 모델을 제안, 구현하였다. 또한 구현된 관리 대행 시스템을 실제 TMN 관리 시스템에 적용하여 그 성능을 평가하였다.

*한국전자통신연구원 초고속TMN연구실
論文番號:97337-0920
接受日字:1997年 9月 20日

Ⅰ. Introduction

With the introduction of various networks such as PSTN, ISDN, B-ISDN and etc., there is an urgent need to efficiently manage them. To meet the need, the concept of Telecommunication Management Network (TMN) has attracted vendor and standards bodies attention. The basic concept of TMN is to provide standardized protocols and interfaces which can achieve interoperability and interconnection between various network management systems or telecommunication network elements.

Since ATM technology became to be applied in the commercialized networking equipment, in the near future, ATM switch will be operated in public network. With the work of ATM switch development, vendors have required applying TMN to management system of ATM switch. To develop TMN management system for ATM switch, a real implementation methodology is needed. But TMN standard bodies are generally confined to information modeling, essential management application function specification as a kind of studying principles of TMN. They will not be able to provide the general and recommended implementation models or methods.

To realize TMN, developer has to work to solve the following implementation issues.

(1) Defining the management behavior of managed resource using GDMO(Guidelines for Definition of Managed Objects).
(2) Integrating Agent into ATM switch
(3) Building of management scenario
(4) Construction and handling of API(Application Program Interface) for CMIP(Common Management Information Protocol) and MOs(Managed Objects) as scenario

Up to date, we have researched topics associated above (1), (3)and (4)[1, 2, 3].

In this paper, we present methods for integrating TMN Agent into an ATM switch, and the evaluations of it. Though there are various implementation methods according to managed resource and developers intention, we implement a TMN Agent system with consideration for the structure of integrating Agent into ATM switch.

The rest of this paper is organized as follows: In section 2, the integration methods of Agent into an ATM switch are presented. In section 3, the implementation details of the proposed Agent is presented. Finally, in section 4 we conclude the paper with future work.

Ⅱ. Integrating Agent into an ATM switch

1. Management Using Manger and Agent Model

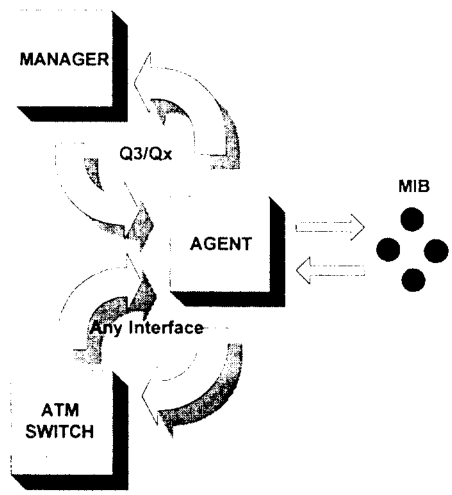


Fig. 1 Overview of TMN System using Manager/Agent Model

The TMN system of ATM switch is composed of Managers and Agents, and the Manager and the Agent are connected by Q interface as shown in figure 1. The Agent has MIB(Management Information Base). The MIB consists of MOs(Managed Objects) which represent components of the ATM switch. The Agent maintains management information of ATM switch and provides the information to Manager by using MIB. The Manager collects the information and manages ATM switches by using the information. For example, the Manager sends the management command for creating a call connection using Q interface to the Agent. Then the Agent sends the command to the ATM switch and updates or creates MOs representing call connection after receives the result of the command from the ATM switch. Finally, the Agent sends result to Manager. If the Manager needs connection information, the Manager is able to get the information from the Agent without request to the ATM switch. That is, because the Agent is sever which exchanges management information between NMS(Network Management System) and managed resource by using MIB, which is a key to realize TMN concept is in the Agent.

In this section, we survey the integration methods of Agent into an ATM switch and choose a method after considering system performance and facility of implementation. We also present implementation of interface between Agent to ATM switch according to the chosen method.

2. Configuration of a Managed ATM switch

Because there is close correlation between Agent and managed resource in TMN, it is important to consider the managed ATM switch.

Our ATM switch consists of one ACS(ATM Central Switch Subsystem) and a number of ALS(ATM Local Switch Subsystem)as shown in figure 2. The ALS is composed of a single-stage switch module and SIM (Subscriber Interface Module)/TIM(Trunk Interface Module) for UNI/NNI interface functions. The ACS is a switching network for the interconnection between ALSs. It also has operations and maintenance functions within the overall system and has HMI (Human Machine Interface) for local operator[4].

Each subsystem has a processor that communicate with other processor by IPC(Inter-Processor Communication) mechanism. The distributed fashion facilitates the expansion of system without performance degradation.

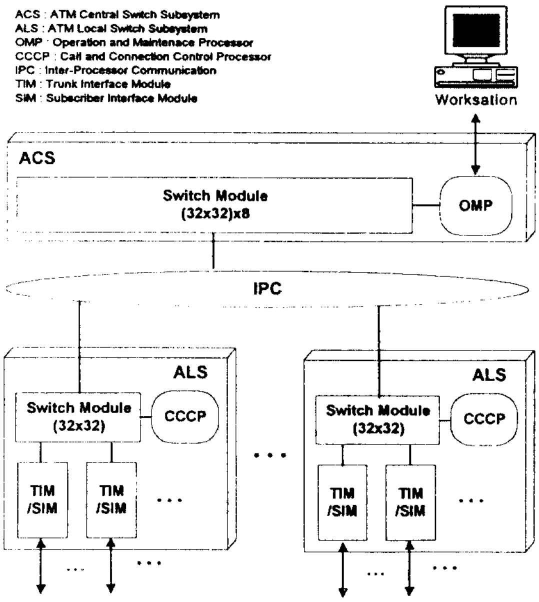


Fig. 2 Distributed Configuration of an ATM switch System

3. Considerations for Integrating

There are many model to integrate Agent into ATM switch. In this section, we evaluate five integrating models with several factors, and choose a model.

3.1 Integrating Models

Agent has information models for components of the ATM switch and maintains the information model. According to relationship between Agent and the ATM switch, each information model is modified and the processing mechanism of that is changed. The ATM switch is also influenced by the relationship. Because the interrelation has influence on Agent or the ATM switch, we must consider the influence when we implement Agent or ATM switch. So, we choose a integrating method after considering several integrating models.

Though there are various models for integration, we consider five models as shown in figure 3. The Model (a) is that an Agent is separated from an ATM switch. The Model (b) is that an Agent is located in the ACS of the ATM switch. In (a) and

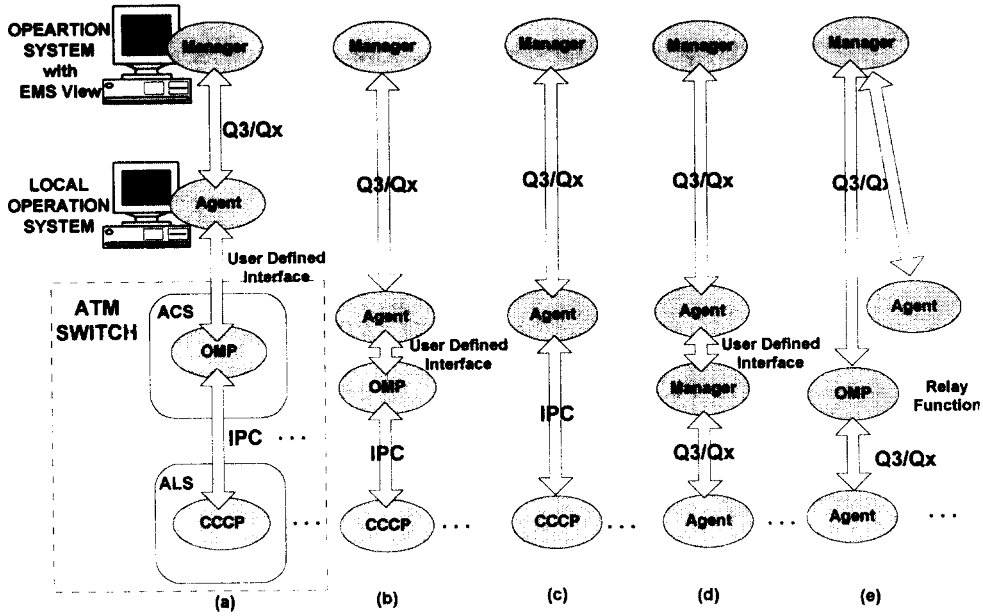


Fig. 3 Models of integrating Agent into an ATM switch

(b), both Models send management command to the ATM switch and supervise the status of the ATM switch through existing S/W blocks of ALS. The Model (c) is that an Agent directly sends command to and receives the result of the command from each ALS with new defined interfaces. The Model (d) and (e) are that Agents are located in each subsystem (ACS, ALSs) of the ATM switch. The Model (d) is that each ALSs is managed by using Manager and Agent model between the ACS and the ALSs. The Model (e) is that EMS Manger is directly access to each Agent in the subsystem. In this case, the ACS has also gateway or relay function.

3.2 Evaluation

It is very difficult to evaluate five models after implementing all of them. So, after we consider the three main factors, which influence the system, we compare and evaluate five models in SUN SPARC ULTRA 1 workstation. The target ATM switch is HANbit ACE64.

The evaluation factors are as following:

- Number of Managed Objects Comprising MIB (Management Information Base)
- Internal Protocol Profile
- The facility of implementation

3.2.1 Number of Managed Objects Comprising MIB

The building of MIB, such as number of MO, MIB implementation technique and processing capacity of MIB, influence on performance of TMN system. For example, Shimizu[5] evaluated the average processing time for the implementation techniques using the same basic MIB. On the contrary, if the MIB run on the same processor and has the same store technique, the number of MO comprising the MIB is a key factor of performance.

The Model (a), (b) and (c) have a MIB, which consists of MOs for whole configuration of the ATM switch. The Model (d) and (e) must have MIBs as many as subsystem. However, the number of MO for

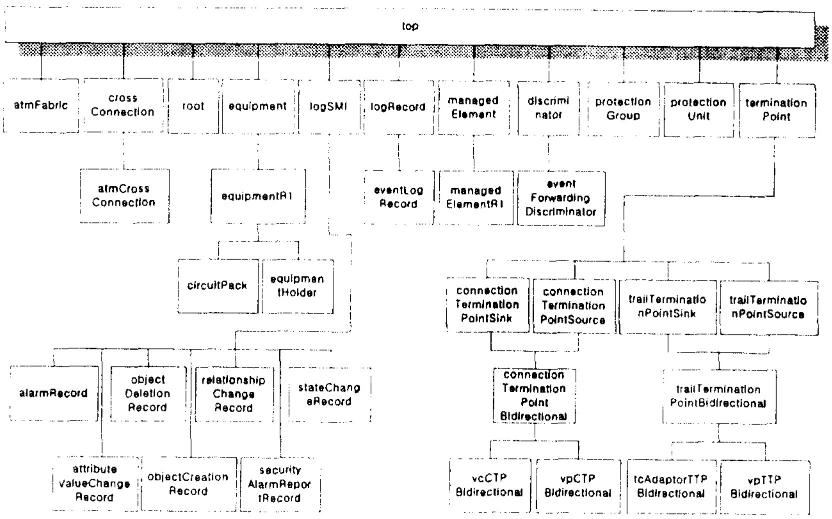


Fig. 3 Object Class Hierarchy for Managed Object

each MIB is much smaller than one of the model (a), (b) and (c). In order to compare the performance of each the model, the relationship of Agent and number of MOs is worthy of consideration. In order to evaluate, we build MIB of ATM switch using MO shown figure 3.

(1) Handling Scenario of MO

In order to test Agent for evaluation, we set up handling scenarios of MO. Figure 4 is an object diagram for M-SET of CMISE(Common Management Information Service Entity). To activate a deactivated interface board, Manager requests activation of board. Before Agent changes an attribute, Agent sends the request to ATM switch. Agent receives the result from ATM switch, changes administrative status of a circuitPack MO.

(2) Response time of an Agent

We measure average response time of an Agent as changing size of MIB. Figure 5 shows the average response time of an Agent for M-GET and M-SET of CMISE(Common Management Information Service Element) service element.

Beyond our expectation, increasing the average response time is very insensible to number of the subsystem, as shown in figure 5. The difference of between 1 and 10 is about 0.012 sec for M-SET. In the case of maximum equipment, the number of subsystem is 64 and the difference between 1 and 64 is about 0.0125 sec for M-SET. This result is much smaller than that of our expectation. Though an Agent of model (d) and (e) manage fewer number MO than the model (a), (b) and (c), the difference of the response time is little.

We omit the measure for M-DELETE and M-CREATE, because these services are much dependent on response time of the ATM switch. However, we can expect the relationship for response time and the number of MO through the result for M-GET and M-SET.

From the result of the measure, we can conclude that the changing the number of MO little influence on the response time of Agent.

(3) CPU utilization of an Agent

In the case of model (a), (b) and (c), because processing of management information is centralized at

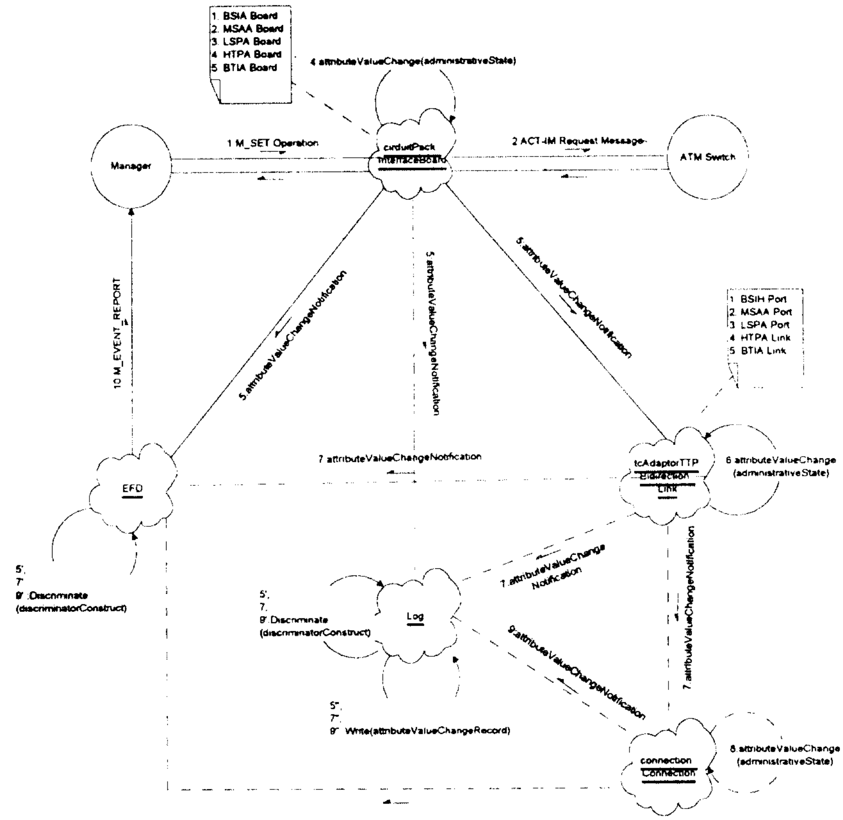


Fig. 4 Object Diagram for M-SET

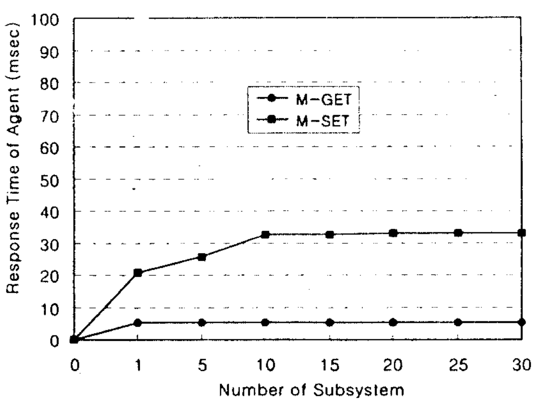


Fig. 5 The Average Response Time of an Agent

an Agent, the processing load is concentrated at an Agent. The concentration of load declines the performance of an Agent. In particular, the model (b), (c), which are located within the ATM switch, influence the performance of the ATM switch. Therefor, with changing the frequency of TMN event, we measure CPU utilization of an Agent for different number of the subsystem.

As shown in figure 6, an Agent has delicate difference of CPU utilization for varying number of the subsystem. However, because the model (b), (c) are located at the ACS only in practically, the ACS has maximum 64 times as the interarrival time in com-

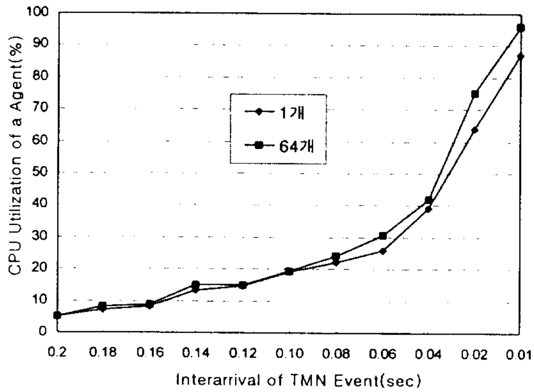


Fig. 6 CPU Utilization of an Agent

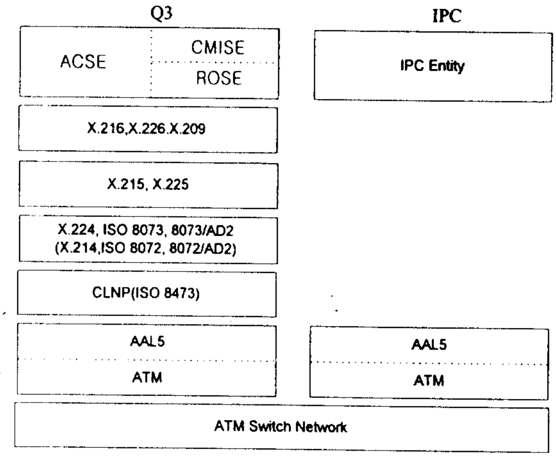


Figure 7. Q3 and IPC Protocol Profiles

parison with a subsystem. For example, the interarrival time of a subsystem which CPU utilization is more 90% is 0.64(0.01 * 64)sec. Therefore, these models heavily load on OMP and decrease performance of the ATM switch. In the case of model (a), though management information is centralized in an Agent, because the Agent is separated from the ATM switch, the load of the Agent dose not influence on the performance of an ATM switch. The model (d), (e) distribute an Agent to each the subsystem and the load is distributed to each the subsystem. We expect that these models have high performance for short interarrival time, and that the influence of the Agent on the ATM switch is insignificant. However, though the Agents of model (d) have low load for large scaled ATM switch, this model brings increasing of load at the Manager. Because of this reason, model (d), which has additional Manager on the ACS, increases the load of OMP. The model (e) has the lowest CPU Utilization of an Agent.

3.2.2 Internal Protocol Profiles

As shown in figure 7, there is two protocol profiles as internal protocol. In the case of Q_3, it is possible for developer to modify protocol. But we base on standard of Q_3 interface that is recommended in Q. 811, Q.812 of ITU-T[6, 7]. IPC protocol is used as internal protocol of the ATM switch[4].

The ATM switch use ATM network as internal control network. Therefor, the lower layer of Q_3 interface uses the CLNS(Connectionless Network Service) protocol, which supported by AAL5 over ATM over ATM physical layers. The IPC protocol uses only 1 and 2 layer of ATM as lower layer and has structure which middle layers are omitted. IPC protocol is simpler than Q_3. We can easily estimate that Q_3 increase CPU utilization and delay time due to complication of the protocol. UEDA[8] presented indirectly evaluation data in his paper. In accordance with his paper, when switch is busy, complicated protocol structure bring increasing of wait time at an Agent. In the complicated aspect of protocol, the model (a), (b) and (c) are suitable. However, because the model (b) and (c) use with M&A application program of ATM switch, two IPC entities must exist in internal network of the ATM switch for an event. In other words, when an element of ATM switch is faulted, M&A application program of ATM switch must send two notifications to local operation system and an Agent. In the case of model (a), only one IPC entity is needed in the ATM switch and message is distributed at workstation. In the result, model (a) has simple internal protocol and can minimize decreasing of performance by protocol.

3.2.3 The facility of implementation

In practical management of the ATM switch, TMN concept can not cover all of OA&M(Operation, Administration and Management) functions and call processing functions of the ATM switch. Therefore TMN function and local operation function is must coexist for the ATM switch. If Agent can use functions of the ATM switch as it is, the implementation of Agent is easy and the configuration of the ATM switch is simple. In this point, the model (a), (b) and (c) that can use S/W blocks of the ACS and the ALS as it is, are the most proper models. On the other hand, in the case of model (d) and (e), The ATM switch and Agent need redundant configurations such as Q interface. The model (d) that must implement an additional Manager is the most complicated.

We estimate that the model (a) is the easiest model for implement. Because an Agent is separated from the ATM switch, developer can implement the Agent with independence for the ATM switch as compared with the others. In building TMN platform, because developer can use general and commercial system environments, he can reduce work as using commercial TMN platform and can work in familiar environment. The mapping Agent software to OAM software of ATM switch, is simple, because the model use internal software block and interface of the ATM switch as it is.

3.2.4 Result of Evaluation

Table 1. Result of Evaluation

Model / Factor	a	b	c	d	e
Processing Speed & Load	B	C	C	C	A
Internal Protocol Complication	A	B	C	D	D
Implementation Facility	A	B	B	D	C

(A : excellent > B > C > D : bad)

From the result of evaluation as shown table 1, we choose two integrating models, the model (a) and (e). The model (a) has merit that ATM switch has simple internal protocol and facility of implementation. On the other hand, when the scale of ATM switch is increase, the increasing of CPU load is defect of model (a). However, If Agent operated in workstation having high performance, this problem will be solved. The model (e) has merits, shortest response time and lower CPU load. On the other hand, the defect of the model (e) is increasing of response time due to complication of internal protocol. However, because the model (e) does not have to use a complete seven layered OSI protocol, using lighter Q interface[6] solves this defect.

In conclusion, all of two models is worthy of an integrating model, and has merits and defects. Therefor, we think what to choose is no problem. In this paper, we choose the model (a) and implement it. The facts that the model (a) is easy to implementation and can reuse functions of the ATM switch are point of advantage in comparison with model (e).

Ⅲ. Implementation Details of the Agent

1. Structure of the Proposed Agent

Our Agent system using the model (a) consists of NEMS(Network Element Management Subsystem) and CASS(Connector Agent to ATM switch System) as shown figure 8.

The NEMS, which has multithread structure, consists of the Agent Worker and several MO Workers. The multithread is essential concept for implementation of Agent. The Agent Work consists of SAP(Service Access Point) Handler, CMIP Handler, MO Worker Handler and MIB Handler. The SAP Handler works communication function with NEML(Network Element Management Layer) Manager. The CMIP Handler translates and makes CMISE service. The MO Worker Handler creates MO Workers, requests carrying of TMN command to MO Workers and receives the result or notification from MO Worker.

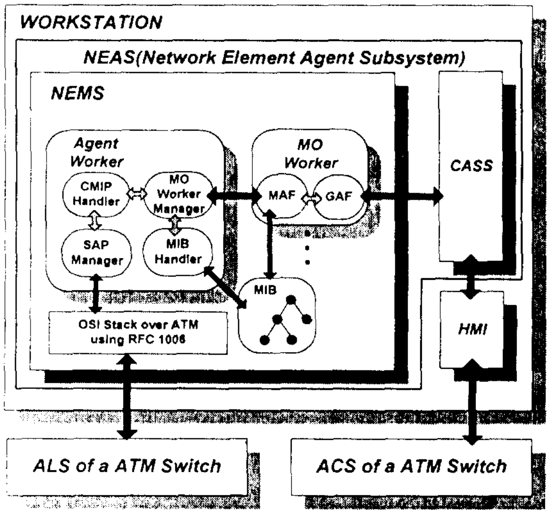


Figure 8. The Structure of Proposed Agent

The MIB Handler controls MIB such as building of MIB, scope, filtering and logging. The MO Worker has two function blocks, MAF (Management Application Function) and GAF(Gateway Application Function). The MAF carries TMN command and result of the ATM switch. It updates related MOs according to result of carry, and sends the result to the MO Worker Handler. If the Agent must communicate with the ATM switch, the GAF carries the role. The CASS takes charge of interface with the ATM switch.

We explain NEMS with M-SET scenario as shown figure 4. TMN Manager requests disable of a board to Agent. The request of Manager is converted to CMIP(Common Management Information Protocol) and is sent through PVC(Permanent Virtual Channel) connection of ATM switch to Agent. The SAP Manager of Agent is received request from Manager and activates CMIP Handler. CMIP Handler translates CMIP request and sends kind of CMISE service element, Invoke identifier, Object Class and Object Instance to MO Handler Manager. MO Handler Manager recognizes M-SET related circuitPack from received information and activates circuitPack MO Handler. The MAF of circuitPack MO Handler commands disable of a board to ATM switch and receives the result from ATM switch through GAF and CASS. From the result of ATM switch, MO Handler disables administrativeState, an attribute of circuitPack in MIB and sends the result to MO Hander Manager. MO Handler Manager deactivates circuitPack MO Handler, logging the result and activates CMIP Hander. CMIP Handler formats the result to CMIP protocol and sends Manager through SAP Manager. If Manager request is M-GET service, MO Handler Manager activates MIB Handler instead of MO Handler.

In the case of the model (a), in order to increase merit of the model, role of CASS is very importance. In the next section, we will present the implementation detail of the CASS

2. Design of CASS[9]

The key of the CASS is that it uses HMI(Human Machine Interface), local operation system of the ATM switch, as interface between the Agent and the ATM switch. The configuration of the CASS is shown in figure 9. The CASS is composed of Interface Module, Connection Module and Automatic Message Analyzer. The Interface Modules are the mapping blocks that convert TMN commands to HMI commands, and output message of the ATM switch to input message of the GAF. In order to enhance the performance of CASS and reduce the delay time, we divide Interface Module according to the TMN management functions; fault management, connection and configuration management, charging management and performance management. The Connection Module is the filtering block that selects the needed messages among output messages of the ATM switch and distributes those messages to the related the Interface Module. The Automatic Message Analyzer is a common library that converts from TMN commands to HMI commands and from operation messages of the ATM switch. to information understood by the MO worker. Because this is the same library that HMI uses in the ATM switch, we do not need novelty library.

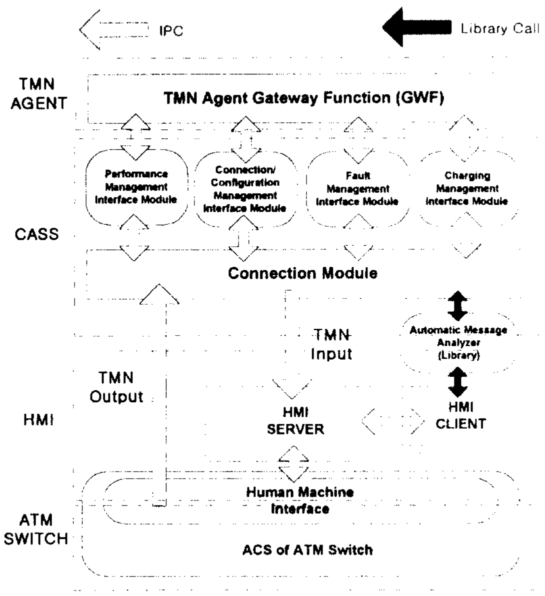


Fig. 9 The Configuration of CASS

A main merit of the model (a) is using the functions of the ATM switch as it is. In this point, the CASS, which reuses interface between ACS and local operator, makes more increase the merit.

3. Processing Time of the Proposed Agent

Table 2. TMN Function for Evaluation

CMISE Services	TMN Function	Description
M-GET	DIS-IM-STS	Display of Status for Subscriber Board
M-SET	ACT-IM-STS DACT-IM-STS	Activation or Deactivation of Service Status for Subscriber Board
M-DELETE	DEL-IM-CONF	Removing of Configuration for Subscriber Board
M-CREATE	ADD-IM-CONF	Adding of Configuration for Subscriber Board

In order to find performance of our Agent, we use SUN SPARC ULTRA 1 workstation for Agent. HMI system, local operation system of ATM switch, is co-operated same workstation. we measure the average processing time when workstation CPU usage is 90%. We measure the average processing time for TMN command as shown table 2.

As shown in figure 10, we also measure the processing time in each section and compare each time. The processing time of CASS is nearly constant and short enough to have no influence on the total performance. In order to measure the processing time for M_CREATE, we execute command, which add a subscriber board into Configuration of ATM switch. In this case, the processing time of the NEMS is longer. The reason is that the NEMS create not only MO of a subscriber board but MOs of subscriber links. If we add a board that has no link, the time is about 120 msec.

We estimate that the proposed Agent has performance that can be applied to TMN system for the ATM switch.

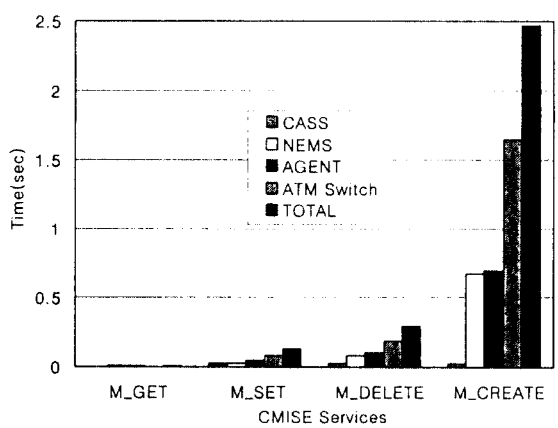


Fig. 10 The Average Processing Time of Agent

Ⅳ. Conclusion

We have implemented the Agent for the ATM switch. Before the implementation, we chose an integrating model of Agent into the ATM switch. The integrating model supports the facility of implementation, low load of the ATM switch and simple internal protocol. We presented the implementation details in based on the integrating model. Then we developed the interface for exchanging management information between the Agent and the ATM switch. Finally, through we measure the average processing time of the Agent, we show that our Agent is feasible.

We will evaluate about NEML Manager and evaluate performance of TMN system later on.

Reference

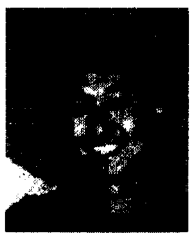
1. H.S. HWANG, Y.W. NAM, S.J. KIM, B.Y. Lee, G.H. Lee, "The Design of TMN-Based Fault Management Function for ATM switching System in Korea," IEEE ICCS/ISPACS96, Vol. 2 of 3, pp. 860~863, NOV., 1996.
2. S.H. Lee, W.D. WOO, "An Implementation of TMN Management Application Framework for ATM switches and Subnetwork," IEEE ICCS/ ISPACS96, Vol. 2 of 3, pp. 542~546, NOV., 1996.
3. Y.W. NAM, H.S. HWANG, "Design of Connection Management Function in TMN Agent System for ATM switching System," Summer Conference of KICS, No. 2, pp. 1069~1072, JUN., 1996.
4. Y.B. KIM, S.S. Lee, C.H. OH, Y.S. KIM, C. HAN, C.H. Yim, "An Architecture of Scalable ATM switching System and Its Call Processing Capacity Estimation," ETRI Journal, vol. 18, NO. 3, Oct., 1996.
5. T. Shimizu, I. Yoda and N. Fujii, "Implementing and deploying MIB in ATM transport Network," Proceedings of the Fourth International Symposium on Integrated Network Management, 1995.
6. ITU-T Recommendation Q.811, Lower Layer Protocol Profiles for The Q3 Interface, March, 1993.
7. ITU-T Recommendation Q.812, Upper Layer Protocol Profiles for The Q3 Interface, March, 1993.
8. Kiyoshi Ueda, Hisayoshi Inamori, Hiroshi Sunaga and Keiichi Koyanagi, "Applying TMN to a Highly Reliable Distributed Switching Node," IEICE Trans., Commun., Vol. E78-B, NO. 1, JAN., 1995.
9. H.S. Hwang, "Implementation of Additional Interface for Exchanging Fault Management Information between TMN Agent and Legacy ATM switch," the 7th Joint Conference on Com. & Inf., No. 2, pp. 1437~1440, April, 1997.
10. ATM Forum, CMIP Specification for the M4 Interface, September, 1995.

황 희 산(Hee-San Hwang) 정회원

1992년 : 한국항공대학교 항공전자공학과(공학사)

1994년 : 한국항공대학교 항공전자공학과(공학석사)

1994년 3월~현재 : 한국전자통신연구원 교환기술연구단 초고속 TMN 연구실 연구원



이 병 윤(Byung-Yoon Lee) 정회원

1990년 : 홍익대학교 전자계산학과(학사)

1992년 : 서강대학교 전산학(석사)

1992년 3월~현재 : 한국전자통신연구원 교환기술연구단 초고속TMN연구실 선임연구원

이 길 행(Gil-Heang Lee) 정회원

1984년 2월 : 전남대학교 계산통계학과(학사)

1986년 2월 : 한국과학기술원 전산학과(석사)

1996년 2월 : 한국과학기술원 전산학과(박사)

1986년 2월~현재 : 한국전자통신연구원 교환기술연구단 초고속TMN연구실, 책임연구원

우 왕 돈(Wang-Don Woo) 정회원
1981년 : 고려대학교 수학과(학사)
1985년 : 숭전대학교 전자계산학과(석사)
1982년 12월~현재 : 한국전자통신연구원 교환기술연구단 초고속TMN 연구실장, 책임연구원