

MPLS의 GSMP 개방형 인터페이스를 위한 망 관리

차 영욱[†] · 김진년^{††} · 김춘희^{†††}

요 약

계층 3의 라우팅과 계층 2의 레이블 스위칭 기술을 통합한 MPLS는 고속의 전달, 트래픽 엔지니어링 그리고 가상 사설망 서비스의 지원을 가능하게 한다. MPLS의 레이블 스위치 및 제어기 사이에서 동작하는 GSMP 프로토콜은 연결, 구성, 장애, 성능관리 및 동기화 기능을 제공하는 개방형 인터페이스 프로토콜이다. GSMP가 적용된 개방형 인터페이스에서 망 관리를 위한 SNMP 에이전트는 레이블 스위치 또는 제어기에 탑재될 수 있다. 본 논문에서는 레이블 스위치의 단순화 및 자원 이용의 효율성을 높이기 위하여 SNMP 에이전트가 제어기에 탑재되어 GSMP와 연동하는 모델을 채택하였으며, MPLS 망 관리와 GSMP의 연동 시나리오를 제시하였다. 본 논문에서 제안된 연동 모델의 실현성을 확인하기 위하여 제어를 구현하였으며, 연동에 따른 구성연결의 연결설정 지연을 측정하였다.

Network Management for the GSMP Open Interface in the MPLS

Young Wook Cha[†] · Jin Nyun Kim^{††} · Choon Hee Kim^{†††}

ABSTRACT

MPLS which integrates routing of layer 3 and switching of layer 2, enables support for fast forwarding, traffic engineering and virtual private network services. GSMP is open interface protocol between a label switch and a controller, and it provides connection, configuration, event, performance management and synchronization. In the GSMP open interface, the functions of network management can be located either in the controller or in the label switch. To simplify the label switch and enhance the efficiency of resources, we adopt the network management model, in which the SNMP agent is located in the controller and is interworked with the GSMP. We presented the interworking scenarios between the GSMP and the network management of MPLS. We implemented the controller to verify the realization of our adopted network management model, and measured the connection setup delay of the provisioned connection.

키워드 : GSMP, MPLS, 개방형 인터페이스(Open Interface), 망 관리(Network Management)

1. 서 론

신뢰성의 상징으로 대변되는 텔레콤 네트워크는 정전이나 천재지변에도 전화 통화를 가능하게 하고, 네트워크 부하에 관계없이 동일한 음성의 품질을 지원하도록 한다. 그러나 텔레콤 네트워크는 비유연성으로 인하여 음성 호 이외의 새로운 응용을 지원하는 것이 어렵다. 이에 반해 인터넷은 자동적이고 단순한 네트워크 계층을 유지하면서 새로운 응용을 촉진시키는 개방형 환경의 지원을 목적으로 하고 있다. 인터넷에서 패킷들은 비연결형 기반의 IP 포워딩으로 처리되나 근래에는 MPLS(Multi-Protocol Label Switching)의 제어 기능을 이용하여 연결형 기반으로 처리할 수 있는 기술이 추가되고 있다[1].

계층 3의 라우팅과 계층 2의 레이블 스위칭 기술을 통합한 MPLS는 짧고 고정된 길이의 레이블을 기반으로 패킷을 전송하는 레이블 교환(swapping) 방식을 이용함으로써 고속의 데이터 전송이 이루어진다. 또한 각 레이블이 매핑되어 있는 경로의 특성에 따라서 트래픽 엔지니어링이나 가상 사설망과 같은 다양한 서비스를 제공하는데 있어서 적합한 기술로 인식되고 있다. MPLS의 망 관리를 위해서 IP 패킷과 레이블의 매핑 및 경로설정을 위한 MIB(Management Information Base)들의 표준화가 진행 중에 있다[2, 3].

제어기와 레이블 스위치 사이의 인터페이스로 GSMP(General Switch Management Protocol)의 표준화가 IETF에서 진행 중이다. GSMP는 연결, 구성, 장애, 성능관리 및 동기화 기능을 제공하는 개방형 인터페이스이다[4]. GSMP 버전 1.1 및 2에서는 ATM 스위치를 위한 제어 기능만이 정의되었으나, 버전 3에서는 프레임릴레이 및 MPLS 스위치를 지원하도록 확장되었다.

GSMP 기반의 개방형 인터페이스에서 망 관리를 위한 SNMP

※ 본 연구는 한국과학재단 우수연구센터사업의 연구결과임.
[†] 성희원 : 안동대학교 컴퓨터공학과 교수
^{††} 춘희원 : 경북대학교 대학원 컴퓨터공학과
^{†††} 정희원 : 새길디지털대학교 인터넷학과 교수
 논문접수 : 2001년 11월 17일, 심사완료 : 2002년 1월 7일

(Simple Network Management Protocol) 에이전트는 레이블 스위치나 제어기에 탑재될 수 있다. 본 논문에서는 GSMP가 적용된 MPLS 네트워크에서 SNMP 에이전트의 탑재 위치에 따른 자원 이용의 효율성 및 레이블 스위치의 구현 오버헤드를 분석하였다. 레이블 스위치의 단순화 및 자원 이용의 효율성을 높이기 위하여 본 논문에서는 SNMP 에이전트가 제어기에 탑재되는 모델을 채택하였다. 이러한 모델에서는 SNMP 에이전트와 GSMP의 연동이 요구되므로 본 논문에서는 연결, 구성, 성능, 장애관리에 대하여 MPLS 망 관리와 GSMP의 연동 시나리오를 제시하였다. 제어기의 구현을 통하여 제안된 연동 시나리오의 실현성을 확인하였으며, 연동에 따른 연결설정 지연의 오버헤드를 측정하였다.

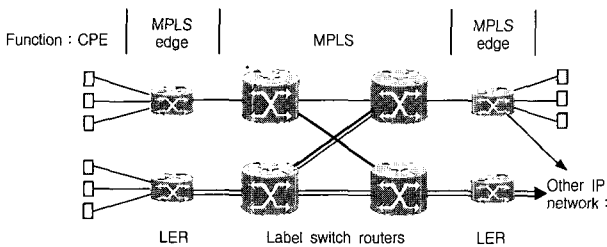
본 논문의 2장에서는 MPLS 기술과 GSMP 프로토콜의 표준화 동향을 기술한다. 3장에서는 MPLS 망 관리와 GSMP의 연동 모델 및 연동 시나리오를 제안하며 4장에서는 제안된 연동 모델의 실현성을 확인하기 위하여 구현한 제어기에 대하여 기술한다. 마지막으로 5장에서는 본 논문의 결론 및 향후 연구 과제에 대하여 기술한다.

2. MPLS와 GSMP

본 장에서는 IETF에서 진행되고 있는 MPLS 망 관리와 GSMP의 표준화 동향에 대하여 기술한다.

2.1 MPLS

계층 3의 라우팅과 계층 2의 레이블 스위칭 기술이 통합된 MPLS 네트워크는 (그림 1)과 같이 LSR(Label Switching Router)이 가운데 위치하고 입구 및 출구에 LER(Label Edge Router)이 위치하게 된다. MPLS 네트워크에 접속된 CPE(Customer Premises Equipment)들은 기존의 IP 전달 기능을 수행하는 라우터이다.

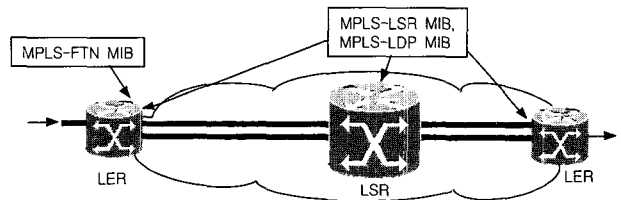


(그림 1) MPLS 네트워크의 구조

MPLS에서는 짧고 고정된 길이의 레이블을 기반으로 패킷을 전송하는 레이블 교환 방식을 이용한다. IP 패킷을 목적지까지 전송하기 위해 필요한 IP 헤더 처리 과정이 모든 홉에서 수행될 필요 없이 입구 LER에서만 수행된다. 입구 LER에서 IP 패킷이 하나의 레이블로 매핑됨으로써 MPLS 네트워크 내의 LSR에서는 레이블 스위칭 기술을 이용한 고속의

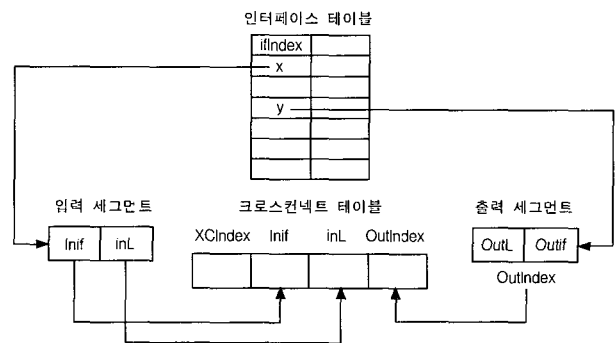
데이터 전송이 이루어지며, 각 레이블에 매핑되어 있는 경로의 특성에 따라서 트래픽 엔지니어링이나 가상 사설망과 같은 다양한 서비스를 제공할 수 있다. 이와 같은 MPLS 기술은 빠른 속도로 증가하고 있는 인터넷 사용자 수와 다양한 서비스의 요구에 적절히 대응할 수 있는 고속의 인터넷 백본망을 구성하기에 적합한 방안으로 대두되고 있다[2, 3].

MPLS 망 관리를 위하여 정의되고 있는 MIB에는 (그림 2)와 같이 MPLS-FTN(FEC To Next hop forwarding label entry) MIB[5], MPLS-LSR MIB[6] 및 MPLS-LDP MIB[7]가 있다. MPLS-FTN MIB는 입구의 LER에서만 적용되며, MPLS-LSR MIB는 MPLS 네트워크의 모든 LER 및 LSR에 적용된다. MPLS-LDP MIB는 레이블 할당 및 분배를 위한 LDP(Label Distribution Protocol) 신호 프로토콜[8]을 위하여 정의되었다.



(그림 2) MPLS 네트워크의 MIB

MPLS-LSR MIB에는 LSP(Label Switched Path)의 설정 및 관리를 위하여 인터페이스의 구성 테이블, 입력 및 출력 세그먼트 테이블 세그먼트들은 여과시키 크로스컨넥트 테이블, 트래픽 파라미터 테이블, 레이블 스택 테이블 그리고 인터페이스 및 세그먼트를 위한 성능 테이블로 구성된다. (그림 3)은 크로스컨넥트 엔트리에서 입력 세그먼트와 출력 세그먼트 그리고 인터페이스 엔트리들과의 상호 관계를 나타낸다. 입력 및 출력 세그먼트들은 인터페이스 인덱스로 인터페이스 테이블과 연관이 된다. 크로스컨넥트 엔트리는 입력 세그먼트의 인터페이스 인덱스와 입력 레이블, 출력 세그먼트의 출력 인덱스를 이용하여 입력 및 출력 세그먼트들을 스위칭하게 한다.



if : interface
L : Label
XC : cross-connect

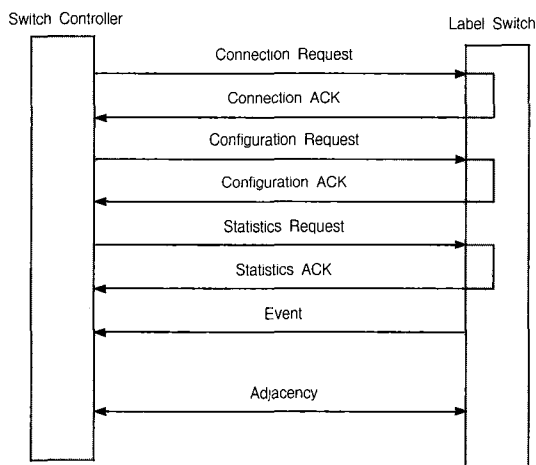
(그림 3) 크로스컨넥트와 세그먼트의 관계

입구의 LER을 위한 MPLS-FTN MIB는 FTN 테이블, FTN 매핑 테이블 그리고 FTN 성능 테이블로 구성된다. FTN 테이블의 각 엔트리들은 동일한 전달 클래스에 속하는 패킷들에 적용될 규칙과 출력 세그먼트 엔트리를 정의한다. FTN 매핑 테이블은 FTN 테이블에서 정의된 FTN 엔트리들을 시스템의 해당 인터페이스에 연관시켜 활성화 시키는 테이블이며, FTN 성능 테이블은 FTN 엔트리를 위한 성능 카운터를 제공한다.

2.2 GSMP 프로토콜

개방형 네트워크의 제어 이슈들에 관한 연구를 활성화하기 위하여 1996년에 개최된 OpenSig 워크샵을 시작으로 개방형 인터페이스에 대한 연구가 활발히 진행되고 있다[9]. IETF에서는 개방형 인터페이스를 위하여 많은 프로토콜을 정의하고 있는데 대표적인 것으로는 MEGACO(Media Gateway Control)[10], COPS(Common Open Policy Service)[11], GSMP 프로토콜 등이 있다. 또한 최근에는 ForCES(Forwarding and Control Element Separation)[12] 워킹 그룹이 결성되어 전달요소와 제어요소의 분리에 따른 개방형 구조 및 프로토콜 요구 사항들에 대하여 검토가 진행되고 있다.

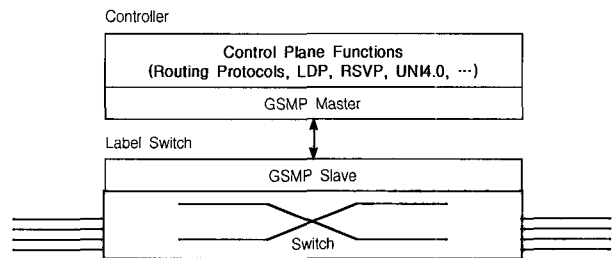
IETF에서 표준화가 진행 중인 GSMP는 (그림 4)와 같이 레이블 스위치와 제어기 사이에서 연결, 구성, 성능, 장애관리 및 동기화 기능을 제공하는 개방형 인터페이스이다. 연결, 구성 및 성능관리를 위한 요구 메시지는 제어기에 의해 생성되며, 스위치는 제어기가 보낸 메시지에 대한 응답을 수행한다. 장애관리를 위한 이벤트 메시지는 스위치에 의해 생성되며, 제어기는 장애 메시지에 대하여 응답 메시지를 보내지 않는다. 프로토콜 버전의 합의, 상태 동기화 등의 정보를 교환하는 GSMP의 Adjacency 기능은 제어기나 스위치 어느 곳에서든 먼저 활성화 될 수 있다[4].



(그림 4) GSMP 기능

GSMP로 인터페이스되는 제어기와 스위치는 (그림 5)와 같

이 마스터-슬레이브 관계를 형성하게 된다. GSMP 모델에서 하나의 제어기는 여러 개의 스위치들을 제어할 수 있으며 또한 다수의 제어기가 하나의 스위치를 분할 기술을 이용하여 제어할 수도 있다. GSMP 버전 1.1 및 2에서는 ATM 스위치를 위한 제어 기능만이 정의되었으나, 버전 3에서는 프레임 릴레이 레이블, MPLS 일반 레이블 및 FEC 레이블이 지원되도록 확장되었다.



(그림 5) GSMP 모델

3. MPLS 망 관리와 GSMP 연동

본 장에서는 GSMP가 적용된 MPLS 네트워크에서 SNMP 에이전트의 탑재 위치에 따른 자원 이용의 효율성 및 구현 오버헤드를 분석하며, MPLS 망 관리와 GSMP의 연동 시나리오를 제시한다.

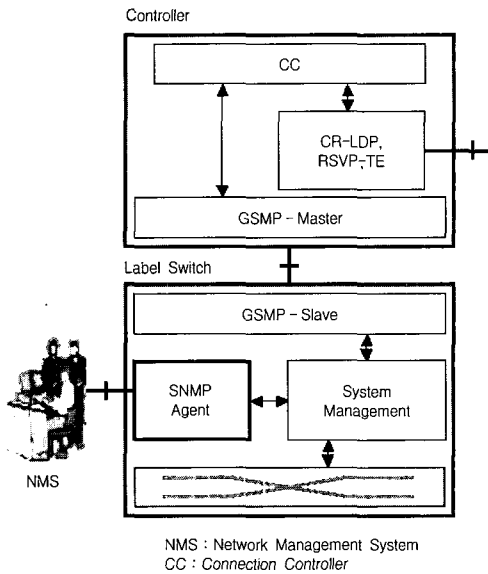
3.1 GSMP 인터페이스의 망 관리 위치

GSMP 기반의 개방형 인터페이스에서 제어기는 레이블 스위치의 모든 자원 상태를 알고 있다는 전제 하에서 동작한다. 즉, 제어기는 연결수락제어 기능에 의하여 연결의 설정 여부를 결정하며, 레이블 스위치는 제어기의 명령에 대하여 단순히 응답을 수행하게 된다[13, 14]. GSMP가 적용된 개방형 인터페이스에서 동적연결은 제어기에 탑재된 신호 프로토콜의 결정에 의하여 설정이 되며, 구성연결(provisioned connection)은 망 관리 기능에 의하여 설정이 된다. 구성연결의 관리를 위한 SNMP 에이전트는 레이블 스위치 또는 제어기에 위치할 수 있다.

3.1.1 레이블 스위치에 탑재된 망 관리

(그림 6)은 SNMP 에이전트가 레이블 스위치에 탑재되어 시스템 관리 기능과의 상호 작용을 통하여 구성연결을 설정 및 해제하는 모델이다. 이러한 모델에서 제어기는 레이블 스위치에서 설정 및 해제된 구성연결과 관련한 자원의 상태를 알 수 없게 된다. GSMP 인터페이스에서 제어기가 레이블 스위치의 모든 자원 상태를 알도록 하기 위하여서는 레이블 스위치가 제어기에게 구성연결과 관련한 자원의 변화 상태를 통보하여야 한다. 이를 위하여서는 GSMP 프로토콜에 새로운 이벤트 메시지가 정의되어야 한다. 다른 대안은 망 관리를 통하여 설정 및 해제되는 구성연결과 신호 프로토콜을 이

용하여 설정 및 해제되는 동적연결을 위한 자원들을 분리하여 사용하는 방안이 있다. 자원을 분리하여 사용하는 방안은 관리면에서는 단순하나 자원의 이용면에서는 비효율적인 방법이다.

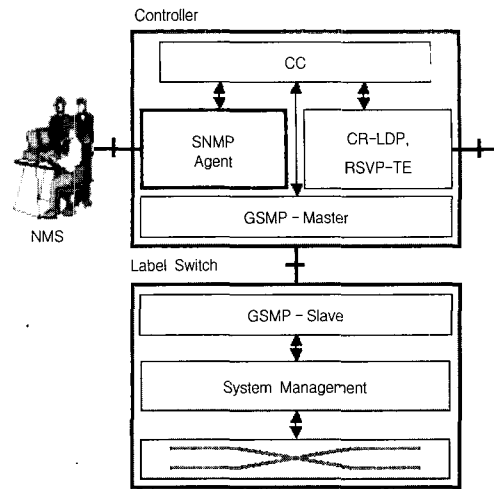


(그림 6) 레이블 스위치에 탑재된 망 관리 모델

3.1.2 제어기에 탑재된 망 관리

GSMP에는 연결관리 뿐만 아니라, 장애관리, 성능관리 및 구성관리를 위한 절차들이 정의되어 있다. GSMP 기반의 개방형 인터페이스에서는 제어기가 이러한 관리 기능들을 주도적으로 수행하므로 레이블 스위치는 디바이스 드라이버 및 기본적인 시스템 관리 기능만으로 구현이 가능하게 된다. 그러나 SNMP 에이전트가 레이블 스위치에 위치하게 되면 망 관리 매니저와의 인터페이스 및 제어기와의 인터페이스를 위한 관리 기능들을 레이블 스위치에 모두 구현하여야 한다. 이는 GSMP가 적용된 개방형 인터페이스에서 레이블 스위치의 기능을 단순화시키려는 의도에 위배되는 접근 방법이다.

본 논문에서는 (그림 7)과 같이 제어기에 SNMP 에이전트와 GSMP의 마스터 기능이 위치하여 상호 연동하는 모델을 채택한다. 즉, SNMP 에이전트가 망 관리 매니저로부터 구성연결의 요구를 수신하여, GSMP와의 연동을 통하여 레이블 스위치에게 구성연결을 설정하는 모델이다. 이와 같은 연동 모델에서는 레이블 스위치를 GSMP의 슬레이브 및 기본적인 시스템 관리 기능만으로 구현이 가능하므로 레이블 스위치를 단순화시킬 수 있으며, 제어기가 구성연결과 동적연결을 위한 모든 자원을 관리하므로 자원을 효율적으로 이용할 수 있다. 또한 하나의 제어기가 여러 개의 레이블 스위치를 관리하는 경우에도 SNMP 에이전트 기능을 모든 레이블 스위치에 구현할 필요 없이 제어기에만 SNMP 에이전트를 구현하면 된다.



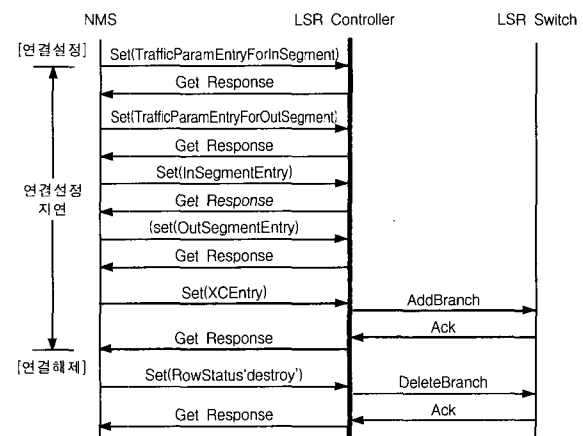
(그림 7) 제어기에 탑재된 망 관리 모델

3.2 연동 시나리오

본 절에서는 SNMP 에이전트가 제어기에 탑재되는 모델에서 MPLS 망 관리와 GSMP의 관리 기능에 대한 연동 시나리오를 기술한다.

3.2.1 연결관리

(그림 8)은 망 관리를 통하여 LSR에 구성연결을 설정하고 해제하기 위한 SNMP와 GSMP의 연동 절차를 나타낸다. 망 관리 매니저는 트래픽 파라미터 및 세그먼트 엔트리를 설정한 후에 크로스컨넥트 엔트리의 설정을 요구한다. 크로스컨넥트 엔트리의 설정을 요구 받은 제어기는 GSMP의 Add Branch 메시지를 이용하여 레이블 스위치에게 구성연결의 설정을 요구하게 된다.



(그림 8) 연결관리의 연동 절차

연결 해제 과정은 망 관리 매니저가 크로스컨넥트 엔트리의 상태 칼럼에 해제(destroy) 값을 설정하여 연결의 해제를 요구하게 된다. 연결 해제의 요구를 수신한 제어기는 GSMP의 Delete Branch 메시지를 이용하여 구성연결의 해제를 스

위치에게 요구하게 된다.

3.2.2 구성관리

GSMP에는 스위치, 포트 그리고 서비스와 관련된 구성관리 기능을 제공한다[4]. GSMP의 구성관리 기능과 MPLS 망 관리의 연동을 지원하기 위해서는 <표 1>과 같이 GSMP에서 제공되는 구성관리 메시지에 기반하여 MPLS-LSR MIB의 확장이 요구된다.

<표 1> 구성관리를 위한 MIB의 확장

GSMP 메시지	확장된 MPLS-LSR MIB
Switch Configuration	anuMplsSwitchConfTable
Port Configuration	anuMplsPortConfTable
Service Configuration	anuMplsServiceConfTable

다음은 스위치의 구성관리를 위한 GSMP의 Switch Configuration 메시지와 망 관리의 연동을 위하여 정의한 anuMplsSwitchConfTable이다. 스위치 구성 테이블의 각 엔트리들은 스위치의 분할 식별자를 인덱스로 사용하며, 엔트리의 오브젝트들은 GSMP의 Switch Configuration 메시지에 정의된 정보 요소들을 기반으로 정의하였다.

```

anuMplsSwitchConfTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF AnuMplsSwitchConfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table specifies switch configuration."
    ::= { anuMplsLsrObjects 1 }

anuMplsSwitchConfEntry OBJECT-TYPE
    SYNTAX      AnuMplsSwitchConfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This entry specifies the global (not port specific) configuration for the switch partition."
    INDEX       { anuMplsSwitchPartitionIndex }
    ::= { anuMplsSwitchConfTable 1 }

AnuMplsSwitchConfEntry ::= SEQUENCE {
    MplsSwitchPartitionIndex      Integer32,
    MplsSwitchConfMtype1          Integer,
    MplsSwitchConfMtype2          Integer,
    MplsSwitchConfMtype3          Integer,
    MplsSwitchConfMtype4          Integer,
    MplsSwitchConfFirmwareVerNo   Integer,
    MplsSwitchConfWindowSize      Integer,
    MplsSwitchConfSwitchType      OCTET STRING,
    MplsSwitchConfSwitchName      OCTET STRING,
    MplsSwitchConfMaxReservation   Integer
}
    
```

3.2.3 성능관리

MPLS-LSR의 MIB에는 인터페이스, 입력 세그먼트 그리고 출력 세그먼트에 대한 성능관리를 위하여 세 개의 테이블을 정의하고 있다[5]. GSMP에서는 Port Statistics 및 Con-

nection Statistics 메시지를 이용하여 포트 및 연결에 대한 성능관리를 수행한다[4].

MPLS 망 관리에서 인터페이스의 성능관리를 위하여 정의된 mplsInterfacePerfTable의 각 엔트리들은 GSMP의 Port Statistic 메시지와 연동이 가능하다. GSMP의 Connection Statistics 메시지에 있는 입력 및 출력 카운터 정보 요소들은 MPLS 망 관리의 세그먼트 성능관리를 위하여 정의된 mplsInSegmentPerfTable 및 mplsOutSegmentPerfTable의 각 엔트리들과 연동이 된다.

3.2.4 장애관리

MPLS 망 관리에는 크로스컨넥트의 생성 및 해제와 연관된 NOTIFICATION 만이 정의되어 있다[5]. GSMP에는 레이블 스위치에서 발생한 장애나 이벤트를 제어기에게 통보하기 위하여 Port Up, Port Down, Invalid Label, New Port, Dead Port 및 Adjacency Update와 같은 이벤트 메시지를 정의하고 있다[4]. 장애관리에 대한 GSMP와 MPLS 망 관리의 연동을 위하여서 레이블 스위치로부터 GSMP 이벤트가 통보되면 제어기는 수신한 이벤트를 SNMP Trap 메시지로 망 관리 매니저에게 통보한다. <표 2>는 GSMP 이벤트 메시지와 MPLS 망 관리와의 연동을 위하여 새로이 정의한 NOTIFICATION의 유형 및 관련 정보 요소들이다.

<표 2> GSMP 이벤트와 NOTIFICATION

이벤트	NOTIFICATION
Port Up	AnuMplsPortUp {partition ID, port no, port session no}
Port Down	AnuMplsPortDown {partition ID, port no, port session no}
Invalid Label	AnuMplsInvalidLabel {partition ID, port no, port session no, label}
New Port	AnuMplsNewPort {partition ID, port, port session no}
Dead Port	AnuMplsDeadPort {partition ID, port, port session no}
Adjacency Update	AnuMplsAdjacencyUpdate {partition ID, code}

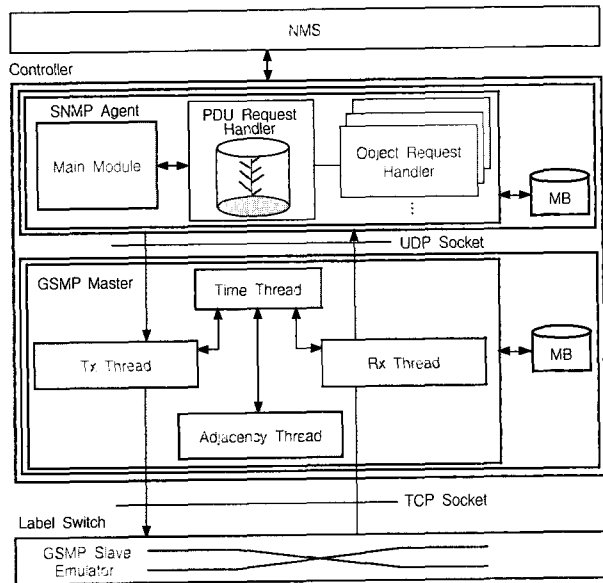
4. 제어기의 구현

본 장에서는 제안된 연동 시나리오의 실현성을 확인하기 위하여 구현한 제어기와 연동에 따른 연결설정 지연의 측정 결과를 기술한다.

4.1 구현 구조

제어기의 전체 구현 구조는 (그림 9)와 같다. MPLS 망 관리를 위한 에이전트와 매니저는 AdventNet 사의 Agent Toolkit [15]과 Management Builder[16]를 이용하여 구축하였다. Agent Toolkit에 의하여 생성되는 에이전트 코드는 Java이다. 에이

전트는 MM(Main Module), PRH(PDU Request Handler) 그리고 ORH(Object Request Handler) 처리기로 이루어진다. 주 모듈인 MM은 NMS로부터 받은 명령의 타당성을 검사하여 오류가 없을 때 메시지 처리기인 PRH에게 명령을 보낸다. PRH의 등록 트리에는 각 오브젝트들의 처리기인 ORH가 Java의 이벤트 처리 기술인 리스너(listener)로 등록되어 있어서 NMS로부터의 명령을 처리한다.

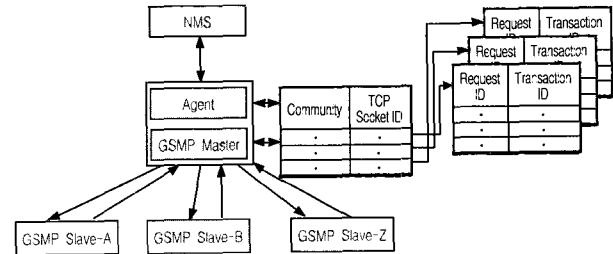


(그림 9) 제어기의 구현 구조

GSMP 마스터는 4개의 Java 쓰레드로 구성된다. Tx 쓰레드는 SNMP 에이전트에서 전달 받은 메시지를 스위치로 전달하는 역할을 수행하며 Rx 쓰레드는 스위치로부터의 응답 메시지를 처리하는 기능을 수행한다. Adjacency 쓰레드는 제어기와 스위치 사이의 동기를 설정하고 유지하는 기능을 수행하며, Time 쓰레드는 GSMP 프로토콜의 타이머 기능을 지원한다. GSMP 마스터는 스위치와의 상호작용에 의하여 생성 및 수정된 연결, 구성, 이벤트 정보 등을 JDBC(Java Data Base Connectivity)와 연계하여 데이터베이스에서 관리한다. 제어기내에 탑재되는 GSMP 마스터와 SNMP 에이전트의 인터페이스는 UDP 소켓으로 구현하였다. 레이블 스위치에 탑재되는 GSMP의 슬레이브 기능은 제어기와의 상호 운용성 시험을 위하여 이모레이터로 구현하였으며, 일반 PC에서 동작하도록 하였다. 마스터와 슬레이브 사이의 인터페이스는 GSMP의 표준 인캡슐레이션 방식 중에서 TCP 소켓으로 구현하였다.

하나의 제어기가 여러 개의 레이블 스위치를 관리하는 경우에는 (그림 10)과 같이 NMS와 제어기 그리고 스위치들이 1:1:n의 관계를 갖는 구성이다. 이러한 구성에서 NMS는 각 스위치 별로 할당된 SNMP의 커뮤니티 필드 정보를 이용하여 SNMP 에이전트에게 명령을 전달한다. 제어기는 수신한 SNMP

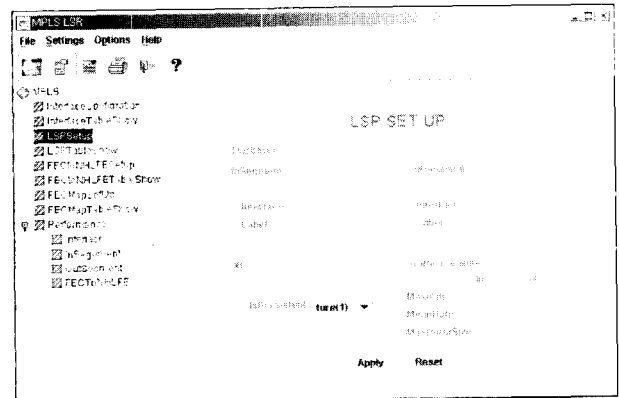
메시지에 포함된 커뮤니티 필드 정보와 매핑되는 TCP소켓 식별자를 이용하여 해당 스위치로GSMP 메시지를 전달한다. 또한 SNMP 메시지의 요구 식별자(Request ID)는 GSMP 메시지의 트랜잭션 식별자와 매핑이 된다.



(그림 10) 연동 모델의 1:1:N 구성

4.2 연결설정 절차

연결설정을 위하여 사용자는 (그림 11)의 인터페이스를 통하여 세그먼트들을 위한 트래픽 파라미터와 레이블 정보를 입력하고 크로스컨넥트를 위한 정보를 입력한다. 적용 버튼을 클릭하면 매니저는 제어기의 SNMP 에이전트에게 연결 설정을 위한 SNMP의 Set 명령들을 전달하게 된다.



(그림 11) 연결설정을 위한 사용자 인터페이스

SNMP 에이전트의 메인 모듈은 매니저로부터 수신한 명령을 메시지 처리기에게 보낸다. 메시지 처리기는 등록 트리에서 해당 오브젝트 처리기를 찾아서 Set 명령에 포함된 각 오브젝트들의 처리를 수행하게 한다. 에이전트는 트래픽 파라미터 엔트리와 입력 및 출력 세그먼트 엔트리들이 생성되면 망 관리 매니저로 Get Response로 응답하게 된다. 크로스컨넥트 엔트리의 생성을 요구 받은 에이전트는 UDP 소켓을 통하여 GSMP의 Tx 쓰레드에게 연결의 설정을 요구하게 된다. 에이전트로부터 연결설정을 요구 받은 Tx 쓰레드는 Add Branch 메시지를 생성하여 레이블 스위치에게 전달한다. GSMP의 Rx 쓰레드로부터 Add Branch의 응답을 수신한 에이전트는 크로스컨넥트 엔트리를 생성하고 매니저에게 크로스컨넥트 엔트리가 생성되었음을 Get Response로 통보하게 된다.

4.3 연결설정 지연

연결설정 지연을 측정하기 위하여 망 관리 매니저와 제어기 그리고 GSMP 이블레이터가 모두 800 MHz 급의 펜티엄-III PC에서 동작하도록 하였다. (그림 8)의 연결관리 연동절차에서 연결설정 지연은 매니저에서 입력 세그먼트를 위한 트래픽 파라미터 엔트리의 생성을 요구한 시점에서 크로스컨넥트 엔트리의 생성에 대한 응답을 받을 때까지의 시간이다.

GSMP와 SNMP 에이전트의 연동을 통하여 연결을 설정하는 경우에 실험실 환경에서 측정된 연결설정 지연은 957ms이었다. 망 관리 매니저로 부터의 연결설정 요구를 에이전트에서 바로 응답하는 경우에 측정된 연결설정 지연은 897ms이었다. 두 가지 경우의 연결설정 지연에 대한 실험 결과 GSMP 프로토콜의 처리 오버헤드가 약 60ms 임을 알 수 있었다. 이러한 오버헤드는 전체 연결설정 지연의 약 6% 정도이므로 전체 연결설정 지연에 큰 영향을 미치지 않는 범위이다. 망 관리를 통한 연결설정 지연이 실 시간성을 요구하지는 않으므로 레이블 스위치의 단순화 및 자원의 효율적인 이용을 가능하게 하는 MPLS 망 관리와 GSMP의 연동에 따른 오버헤드는 무시할 만한 것으로 판단된다.

5. 결 론

MPLS 네트워크는 계층 3의 라우팅과 계층 2의 레이블 스위칭 기술의 결합을 통하여 고속의 패킷 전달 및 다양한 QoS 보장 서비스를 제공할 수 있다. MPLS에서 전달평면의 레이블 스위치와 제어평면의 제어기 사이에 적용되는 GSMP는 연결, 구성, 장애, 성능관리 및 동기화 기능을 제공하는 개방형 인터페이스이다. GSMP가 적용된 개방형 인터페이스에서 망 관리를 위한 SNMP 에이전트는 레이블 스위치 또는 제어기에 탑재될 수 있다.

본 논문에서는 레이블 스위치의 단순화 및 자원 이용의 효율성을 높이기 위하여 GSMP 마스터와 SNMP 에이전트가 제어기에 탑재되는 모델을 채택하였다. 제어기에 SNMP 에이전트와 GSMP 마스터가 탑재되므로 MPLS 망 관리와 GSMP의 연동이 요구되며, 본 논문에서는 연결, 성능, 장애, 구성관리 별로 연동 시나리오를 제시하였다. 제어기의 구현을 통하여 제안된 연동 모델 및 시나리오의 실현성을 확인하였으며, 연동에 따른 연결설정 지연을 측정하였다. 전체 연결설정 지연에 있어서 GSMP의 연동에 따른 오버헤드는 약 6% 정도이며, 이러한 오버헤드는 전체 연결설정 지연에 큰 영향을 미치지 않는 범위이다.

추후 연구사항으로는 입구의 LER에서 MPLS 망 관리와 GSMP의 연동을 위하여 MPLS-FTN MIB에 대한 연동 시나리오를 정의하고자 한다.

참 고 문 헌

[1] Ori Gerstel, "Optical Layer Signaling : How Much Is Really

Needed?," IEEE Communications Magazine, October, 2000.

- [2] Jeremy Lawrence, "Designing Multiprotocol Label Switching Networks," IEEE Communications Magazine, July, 2001.
- [3] E. Rosen *et al.*, "Multiprotocol Label Switching Architecture," RFC 3031, January, 2001.
- [4] A. Doria, *et al.*, "General Switch Management Protocol," Internet draft, <draft-ietf-gsmp-09.txt>, May, 2001.
- [5] Cheenu Srinivasan *et al.*, "MPLS Label Switch Router Management Information Base Using SMIPv2," Internet draft, <draft-ietf-mpls-lsr-mib-07.txt>, January, 2001.
- [6] Thomas D. Nadeau *et al.*, "Multiprotocol Label Switching (MPLS) FEC-ToNHLFE (FTN) Management Information Base Using SMIPv2," Internet draft, <draft-ietf-mpls-ftn-mib-01.txt>, April, 2001.
- [7] J. Cucchiara *et al.*, "Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)," Internet draft, <draft-ietf-mpls-ldp-mib-08.txt>, August, 2001.
- [8] L. Andersson *et al.*, "LDP Specification," RFC 3036, January, 2001.
- [9] Nils Bjorkman *et al.*, "The Movement from Monoliths to Component-Based Network Elements," IEEE Communications Magazine, January, 2001.
- [10] Greene, N. *et al.*, "Media Gateway control protocol architecture and requirements," RFC 2805, April, 2000.
- [11] J. Boyle, "The COPS (Common Open Policy Service) Protocol," RFC 2748, January, 2000.
- [12] T. Anderson *et al.*, "Requirements for Separation of IP Control and Forwarding." <draft-anderson-forces-req-02.txt>, September, 2001.
- [13] Tom Worster, *et al.*, "A QoS Model for GSMP," <draft-worster-gsmp-qos-00.txt>, March, 1999.
- [14] 백현규, 김춘희, 차영욱, "광 인터넷에서의 개방형 인터페이스", 제6회 통신 소프트웨어 학술대회(COMSW2001), 2001.
- [15] AdventNet, "AdventNet Agent Toolkit (java Edition) 4.2 Documentation," August, 2001.
- [16] AdventNet, "AdventNet Management Builder Release 4.2 Product Documentation," 2001.



차 영 욱

e-mail : ywcha@andong.ac.kr

1987년 경북대학교 전자공학과 졸업(학사)

1992년 충남대학교 계산통계학과 졸업(석사)

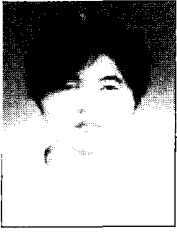
1998년 경북대학교 컴퓨터공학과 졸업(박사)

1987년~1999년 한국전자통신연구원 선임연

구원

1999년~현재 안동대학교 컴퓨터공학과 조교수

관심분야 : 광 인터넷, 개방형 통신망, 망 관리



김진년

e-mail : edie@comeng.andong.ac.kr

1999년 안동대학교 컴퓨터공학과 졸업
(학사)

2002년 안동대학교 대학원 컴퓨터공학과
(석사)

2002년~현재 경북대학교 대학원 컴퓨터공
학과(박사과정)

관심분야 : 광 인터넷, 개방형 인터페이스, 망 관리



김춘희

e-mail : chkim@sgdu.ac.kr

1988년 전남대학교 전산통계학과(학사)

1992년 충남대학교 전자계산학과(석사)

2000년 경북대학교 컴퓨터공학과(박사)

1988년~1995년 한국전자통신연구원 연구원

2002년~현재 새길디지털 대학교 인터넷학
과 조교수

관심분야 : 네트워크 프로토콜, 액세스 망, 트래픽 제어