

# 다중 라우트 서버를 두는 확장된 가상랜 시스템

## (An Extended Virtual LAN System Deploying Multiple Route Servers)

서 주 연 <sup>†</sup> 이 미 정 <sup>\*\*</sup>  
(Juyeon Seo) (Meejeong Lee)

**요 약** 가상랜은 물리적 위치에 관계없이 마치 하나의 LAN에 연결되어 있는 것처럼 통신할 수 있는 구조로 브로드캐스트 도메인을 제한하여 대역폭 낭비를 감소시키고 전체 네트워크의 효율을 증가시킨다. Newbridge사는 IP 서브넷의 주소를 가상랜과 매핑하여 ATM LAN 스위치 망에서 3계층 가상랜을 구성하는 VIVID 시스템을 개발하였다. 이 시스템에서는 하나의 라우트 서버에서 주소 해석과 가상랜 구성 및 브로드캐스트 데이터 전송을 모두 담당하기 때문에 망의 규모가 커지게 되면 라우트 서버가 병목 지점이 될 수 있다. 이와 같은 문제점을 해결하기 위해 택할 수 있는 방법 중 한 가지는 다중의 라우트 서버를 두는 것이다. 본 논문은 VIVID 시스템에 여러 개의 라우트 서버를 두는 구조로서 유기적인 구조와 독립적인 구조 두 가지를 제시하고 시뮬레이션을 통하여 각 구조의 특성을 비교 분석하였다. 시뮬레이션 결과, 브로드캐스트 세션의 길이와 브로드캐스트 세션 내에서의 브로드캐스트 데이터 프레임 발생 간격 등에 의해 제시한 두 가지 모델의 성능이 변하게 되며, 확장성과 데이터 전송의 효율성 간에 서로 상쇄 효과가 있음을 볼 수 있었다.

**키워드** : 가상랜, 가상랜 브로드캐스트, 3계층 가상랜

**Abstract** Virtual LAN (VLAN) is an architecture to enable communication between end stations as if they were on the same LAN regardless of their physical locations. VLAN defines a limited broadcast domain to reduce the bandwidth waste. The Newbridge Inc. developed a layer 3 VLAN product called VIVID, which configures a VLAN based on IP subnet addresses. In a VIVID system, a single route server is deployed for address resolution, VLAN configuration, and data broadcasting to a VLAN. If the size of the network, over which the VLANs supported by the VIVID system spans, becomes larger, this single route server could become a bottleneck point of the system performance. One possible approach to cope with this problem is to deploy multiple route servers. We propose two architectures, organic and independent, to expand the original VIVID system to deploy multiple route servers. A course of simulations are done to analyze the performance of each architecture that we propose. The simulation results show that the performances of the proposed architectures depend on the lengths of VLAN broadcasting sessions and the number of broadcast data frames generated by a session. It has also been shown that there are tradeoffs between the scalability of the architecture and their efficiency in data transmissions.

**Key words** : Virtual LAN, virtual LAN broadcast, layer 3 virtual LAN

### 1. 서 론

가상랜이란 물리적으로 분리된 여러 개의 LAN 세그

먼트 상에 걸쳐 있는 일련의 종단 스테이션들이 그들의 물리적 위치에 구애받지 않고 마치 하나의 공통 LAN에 속해 있는 것처럼 통신할 수 있는 기술이다[1]. 이는 기존 LAN의 브로드캐스트와 멀티캐스트 트래픽이 전달되는 영역을 제한시켜 대역폭 낭비를 감소시키고 전체 네트워크의 효율을 증가시킨다. 또한 가상 경계를 만들어 원하는 그룹에만 데이터를 전송하게 하여 네트워크 보안을 강화시킨다. 특히, LAN 스위치로부터 시작된 가상랜 개념을 ATM 네트워크와 ATM에 연결된 디바이스

· 이 연구는 한국학술진흥재단의 2001년 BK21 특화사업지원으로 수행되었음.

<sup>†</sup> 비 회 원 : 이화여자대학교 컴퓨터학과  
982COG10@mm.ewha.ac.kr

<sup>\*\*</sup> 정 회 원 : 이화여자대학교 컴퓨터학과 교수  
lmj@mm.ewha.ac.kr

논문접수 : 2000년 7월 27일

심사완료 : 2001년 11월 28일

가 존재하는 환경에 확장하여 적용한다면, 가상랜의 장점은 그대로 수용하면서, 멀티미디어 데이터 전송이나 실시간 어플리케이션 지원과 같은 기존 LAN이 제공하지 못하는 요구사항을 만족시킬 수 있다[2].

그런데 현재 ATM-LAN 스위치 망에서의 가상랜을 정의하는 방법으로는 ATM 포럼에서 제안하여 표준화된 LANE(LAN Emulation)을 확장한 2계층 가상랜이 대부분이다. 2계층 가상랜을 구성하는 방법에는 포트그룹에 의한 것과 MAC 주소 그룹에 의한 것이 있다[3]. 스위치의 포트를 그룹화하여 가상랜을 정의하는 것은 구현이 용이한 반면 사용자의 포트가 변경될 때마다 네트워크 관리자가 가상랜을 재정의 해야한다는 단점이 있다. MAC 주소에 의한 가상랜은 초기 구성 시간이 오래 걸리지만 그 이후에는 MAC 주소가 하드웨어적으로 지정되어 있으므로 사용자의 위치가 변경되더라도 자동으로 구성할 수 있다는 장점이 있다.

그러나 LANE을 기반으로 하는 2계층 가상랜은 한 ELAN의 범위를 벗어나는 호스트들 간에는 가상랜 멤버가 될 수 없는 단점을 가지고 있다. 특히 물리적으로 다른 형태의 LAN(예 : 이더넷과 토큰링)에 속하는 호스트들 간에는 같은 가상랜의 멤버가 될 수 없다는 제약점이 있다. 그러나 3계층 주소를 기반으로 가상랜을 구현하게 되면 물리적인 위치에 관계없이 서로 다른 ELAN 혹은 다른 형태의 LAN에 있는 호스트들이 같은 가상랜 멤버가 되어 별도의 라우팅 없이 스위칭만으로도 효율적으로 멤버간의 통신이 가능하게 된다.

3계층 가상랜을 구성하는 방식은 크게 프로토콜 타입에 의한 것과 IP 서브넷에 의한 것 그리고 IP 멀티캐스트 그룹에 의한 것으로 나눌 수 있다[3]. 프로토콜 타입에 의한 방식은 프로토콜의 타입에 따라 가상랜을 정의하는 것으로 사용자가 물리적으로 이동하더라도 네트워크 주소를 재구성할 필요없이 가상랜을 유지할 수 있다. IP 서브넷에 의한 방식은 같은 IP 서브넷 주소를 가진 호스트를 같은 가상랜 멤버로 취급하는 방법으로 프로토콜 기반 가상랜과 마찬가지로 네트워크 주소의 재구성 없이도 가상랜을 유지할 수 있으며 스위치간에 가상랜 정보를 교환하기 위한 프레임 태깅이 필요없게 되고 자동으로 가상랜이 구성되도록 할 수 있는 장점이 있다. IP 멀티캐스트 그룹 방식은 멀티캐스트 그룹에 결합하는 모든 호스트들을 같은 가상랜 멤버로 취급하는 방법으로 높은 수준의 유동성을 갖는다.

ATM 상에서 3계층 가상랜을 지원할 수 있는 제품으로 현재 개발되어 있는 것은 3Com사의 FIRE ASIC과 Newbridge사의 VIVID 시스템 등이 있다. 두 제품 모

두 IP 서브넷 별로 가상랜을 구성하며, 초기에 관리자가 IP 서브넷과 가상랜 ID를 수동으로 매핑하고 그 후 호스트의 이동으로 인해 일어나는 변경에 대해서는 자동으로 가상랜을 재구성하는 반자동 방식을 사용한다. 또한 IP 서브넷으로 가상랜 정보를 알아낼 수 있으므로 특별히 가상랜 ID를 가상랜 프레임의 헤더에 추가하지 않아도 된다. 3COM사의 FIRE ASIC은 따로 서버를 두지 않고 가상랜을 구성하는 반면, Newbridge사의 VIVID는 라우트 서버를 두어 가상랜 정보를 관리하고 가상랜에 데이터를 전송하기 위해 서버에 주소해석을 요청해야하는 라우트 서버 방식을 택하고 있다[4][5].

기존의 VIVID 시스템 모델에서는 하나의 라우트 서버가 가상랜 구성을 위한 정보 수집과 데이터 전송을 위한 주소해석, 브로드캐스트 전송을 담당한다. 따라서, 망의 규모가 커지게 되면 일반적으로 단일 서버 중앙집중형 구조에서 발생하게 되는 서버에서의 병목현상이 VIVID 시스템에서도 일어날 가능성이 있다. 그러나, 현재 업체들에 의하여 제안된 VIVID 시스템을 포함한 3계층 가상랜 구성 방안에서는 이와 같은 확장성에 대한 해결책이 언급된 바 없다. 이에 본 논문에서는 다중의 라우트 서버를 두어 VIVID 시스템을 확장함으로써 확장성을 지원할 수 있는 고유한 방안을 두 가지 제시하였다. 첫 번째 방안인 유기적 구조에서는 다중의 라우트 서버가 각각 네트워크 전체에 대한 동기화된 가상랜 정보를 유지하여 가상적으로 마치 하나의 라우트 서버처럼 작동하며, 두 번째 방안인 독립적 구조에서는 각 라우트 서버가 자신이 담당하는 네트워크 영역의 가상랜 정보만을 유지하고 다른 영역에 대한 자세한 가상랜 정보 및 가상랜 브로드캐스트에 대해서는 그 영역을 담당하는 라우트 서버를 의뢰한다.

두 모델의 성능을 비교하기 위하여 시뮬레이션을 실행하였다. 시뮬레이션 결과 유기적 방법이 제어 메시지 오버헤드나 라우트 서버의 메모리 요구량 등의 측면에서 본 확장성 면에서는 항상 더 우수하며, 성능면에서도 가상랜 브로드캐스트 세션의 길이 및 데이터 양이 어느 수준 이하인 경우에는 독립적인 구조에 비하여 우수함을 알 수 있었다.

본 논문은 다음과 같은 순서로 구성되어 있다. 먼저 2장에서는 기존의 VIVID 시스템을 설명하고 3장에서는 다중의 라우트 서버를 두기 위해 VIVID 시스템을 확장하는 두 가지 방안에 대하여 자세히 설명한다. 4장에서는 두 방안의 성능을 비교하기 위한 시뮬레이션과 그 결과를 제시하고 마지막으로 5장에서 결론을 맺는다.

## 2. VIVID 시스템

Newbridge사의 VIVID 시스템은 고속 백본 네트워크 상에서 LAN 인터넷워킹을 제공하는 엔터프라이즈 네트워크 솔루션으로써 Ethernet, FDDI, 그리고 ATM 호스트에 대하여 스위치된 상호 연결 서비스를 제공한다 [5]. 또한 Ethernet, FDDI, 그리고 ATM 호스트들 사이에서 네트워크 계층 IP 라우팅과 MAC 계층 브릿징을 지원하는 기능을 가지고 있다.

VIVID 시스템은 사용자 정의 가상 네트워크를 지원하고, 자동적으로 호스트들의 추가, 이동 그리고 변경을 탐지하여 물리적인 위치와는 관계없이 논리적 워크그룹을 구성할 수 있도록 해 준다. 가상랜 워크 그룹은 물리적인 위치에 관계없이 논리적으로 구성되어 있기 때문에 기존의 라우터 방식과 같이 각 라우터 포트마다 분리된 IP 서브넷을 만들 필요가 없는 것이 특징이고, 서로 다른 위치에 있는 호스트들이 같은 서브넷에 속할 수 있다. 여기서 특정 IP 기반 가상랜에 속하기 위해서는 장치의 IP 주소 서브넷 부분과 가상랜이 정의한 서브넷 부분이 일치해야 한다. 또한, VIVID 시스템에서는 각 LAN 스위치에 있는 포트를 라우트된 가상 네트워크에 바인딩함으로써 그 포트가 특정 서브넷으로부터의 트래픽을 지원할 수 있게 한다. 즉 ATM 스위치 상에 연결되어 있는 서로 다른 LAN 스위치의 이더넷 포트들이 동일한 라우트된 가상 네트워크에 할당될 수 있다. VIVID 시스템에서는 3계층 가상랜을 라우트된 가상 네트워크 (Routed Virtual Network)라고 부른다. 본 논문에서는 앞으로 이를 간단하게 (3계층) 가상랜이라 부르기로 한다.

VIVID 시스템은 그림 1에 보여준 것처럼 데이터 포워딩을 담당하는 스위칭 장치들(ATM/LAN 스위치)과 라우팅 기능을 담당하는 VIVID 라우트 서버로 구성된다 [5]. VIVID 라우트 서버는 네트워크 번호와 포트 멤버십을 포함한 가상랜을 정의하는 관리 정보를 저장하고, 확장된 비연결성 네트워크의 토폴로지를 알기 위해 OSPF와 RIP를 사용하여 외부 라우터와 통신하는 기능을 가지고 있다. VIVID 라우트 서버는 VIVID 네트워크에 대해서 브릿지된 LANE을 제공하고, 비연결형 LAN 프로토콜을 연결 지향 ATM 망에 매핑하기 위해서 주소 해석, 토폴로지 관리, 가상랜 브로드캐스트 서비스 등의 기능을 제공한다. VIVID 시스템의 동작 방식은 MPOA (MultiProtocol Over ATM)와 비슷한 개념을 가지고 있는데 VIVID 라우트 서버가 MPS (MPOA Server)와 유사한 기능을 제공하고 있다[5][6].

LAN 스위치가 등록되지 않은 근원지 MAC 주소로

부터 처음으로 패킷을 받게 되면, 그 패킷을 VIVID 라우트 서버에게 보내게 되고, VIVID 라우트 서버는 근원지의 IP 주소를 검사한다[5]. 근원지의 IP 주소가 VIVID 라우트 서버가 담당하는 가상랜에 속하면 그 근원지를 해당 가상랜에 포함시키면서 자신의 테이블 내에 그 내용을 저장한다. 그리고 LAN 스위치에게 해당 가상랜에 속하는 모든 다른 멤버들에 대한 정보인 ATM 주소와 MAC 주소, IP 주소 등을 함께 전달한다. 가상랜 멤버가 가상랜 브로드캐스트 전송을 하고자 할 경우 VIVID 라우트 서버는 LANE의 LECS(LAN Emulation Configuration Server)의 역할을 대신해 준다[5][7]. 즉, 그림 1의 예를 보면 B.1이 가상랜 멤버인 B.2, B.3에게 데이터를 전송하고자 하는 경우로, B.1은 우선 이 사실을 LAN 스위치 1을 통해 VIVID 라우트 서버에게 알린다. 그러면 VIVID 라우트 서버는 자신의 테이블 정보를 이용하여 브로드캐스트하고자 하는 데이터를 해당 가상랜에 속하는 모든 스테이션들인 B.2, B.3에게 포워드 해 준다.

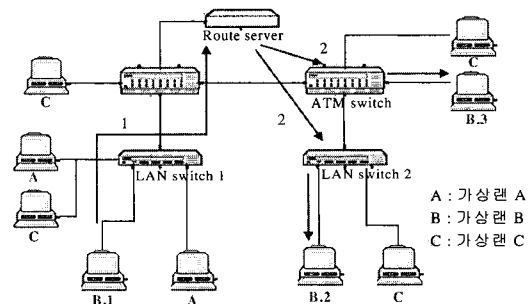


그림 1 VIVID 시스템 구조

## 3. 다중 라우트 서버를 두는 VIVID 시스템 구성 방안

앞서 설명했듯이 VIVID 시스템은 라우트 서버가 라우팅 기능뿐만 아니라 브로드캐스트 데이터 전송까지 담당하고 있어 망이 확장됐을 경우 라우트 서버에서 병목현상이 발생할 수 있다. 본 논문은 이러한 병목현상을 해결할 수 있는 방안으로 여러 개의 라우트 서버를 두도록 VIVID 시스템을 확장하는 방안을 제시하였다. 즉 라우트 서버끼리 협력 혹은 정보의 동기화를 위해 정보를 교환하는 방법을 정의하고 서로 다른 라우트 서버가 관리하는 여러 구역에 퍼져 있는 가상랜 멤버들에게 브로드캐스트 데이터를 어떻게 전송해야하는지를 정의한다. 본 논문에서는 두 가지의 가상랜 브로드캐스트 전송

방식을 제안하는데, 브로드캐스트 전송 방식에 따라 라우트 서버간의 정보 교환 방법이 달라진다. 첫 번째 브로드캐스트 방식은 먼저 근원지 라우트 서버가 브로드캐스트의 목적지인 가상랜을 담당하는 각 라우트 서버들로 브로드캐스트 데이터를 전달하고 이들 라우트 서버들이 다시 자신이 담당하는 구역의 해당 ATM/LAN 스위치로 데이터를 릴레이하는 방식이다. 앞으로 이를 유기적 구조라고 명칭한다. 두 번째 방식은 근원지 라우트 서버가 가상랜 멤버를 가지고 있는 목적지 ATM/LAN 스위치들과 직접 VC를 설정하여 브로드캐스트하는 방식으로 독립적 구조라 부르기로 한다. 두 구조에서 모두 라우트 서버들끼리는 영구적 VC로 연결되어 있다. 그러나 라우트 서버와 ATM/LAN 스위치들 간에는 기존의 VIVID 시스템 구조에서의 마찬가지로 필요시에만 VC를 설정한다. 3.1과 3.2절에서는 이들 방안 각각에 대해 구체적으로 VIVID 시스템의 각 엔티티에서 유지하는 데이터 구조와 가상랜에서의 데이터 브로드캐스트를 위한 동작방식을 설명한다.

3.1 유기적 구조

이 구성방안에서는 근원지 라우트 서버가 브로드캐스트 데이터의 목적지 가상랜을 담당하고 있는 라우트 서버들에게 먼저 데이터를 전달하고 이들 라우트 서버가 다시 각각 자신이 담당하는 구역의 해당 ATM/LAN 스위치로 데이터를 릴레이한다. 이 방식은 근원지 라우트 서버가 목적지 ATM/LAN 스위치들과 직접 VC를 맺어 브로드캐스트 하는 방식에 비해 라우트 서버끼리만 VC를 맺고 있으면 되므로 매번 브로드캐스트 할 때마다 VC를 설정하기 위한 시간을 줄일 수 있고 라우트 서버가 가상랜 별로 담당 ATM/LAN 스위치의 정보를 유지하지 않아도 되므로 라우트 서버 테이블 정보 양을 줄일 수 있어 확장성 면에서 유리한 방법이다.

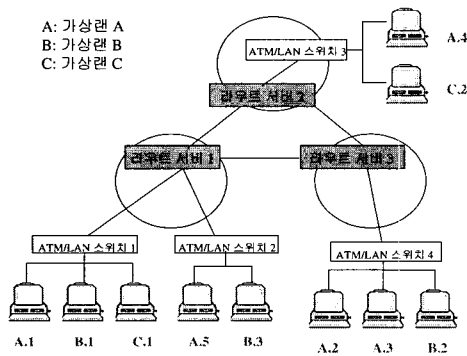


그림 2 유기적 구조의 예

그림 2는 세 개의 가상랜을 세 개의 라우트 서버가 관리하는 구조의 예를 보여주고 있다. 라우트 서버간에는 테이블 정보 교환을 위한 업데이트 메시지가 전달된다. 라우트 서버와 스위치 사이에서는 사용자 데이터와 호스트의 가상랜 등록을 위한 주소해석 메시지가 전달되고 스위치와 호스트 사이에서는 가상랜에 브로드캐스트하기 위한 데이터 프레임이 전송된다.

3.1.1 유기적 구조에서의 라우트 서버 테이블 구조

라우트 서버 테이블은 자신이 관리하는 구역에 속한 호스트들이 가입해 있는 모든 가상랜(IP 서브넷 주소)에 대하여 그 가상랜을 관리하는 서버들의 ATM 주소 리스트를 유지한다. 또한 자신이 담당하는 구역에 대해서는 각 가상랜에 관련된 ATM/LAN 스위치에 관한 정보(ATM 주소)도 유지한다. 서버는 이와 같은 정보를 유지하면서 호스트의 가상랜 가입을 처리하고 데이터 전송을 위한 주소 해석을 한다. 서버는 가상랜 멤버 정보의 동기화를 유지하기 위해 다른 구역의 서버들과 서로 테이블 정보를 교환하며, 스위치와는 가상랜 가입과 데이터 전송을 위해 정보를 교환한다. 그림 3은 그림 2의 라우트 서버 1이 유지하는 테이블 구조를 보여주고 있다. 라우트 서버 1이 관리하는 구역에 가상랜 A, B, C가 존재하므로 서버의 테이블에는 가상랜 A, B, C의 IP 서브넷 주소 리스트가 있고 가상랜 각각에 대해서는 담당하는 서버들의 ATM 주소 리스트를 두며, 각 가상랜에 대해 자신(라우트 서버 1)이 담당하는 ATM/LAN 스위치의 ATM 주소를 유지한다.

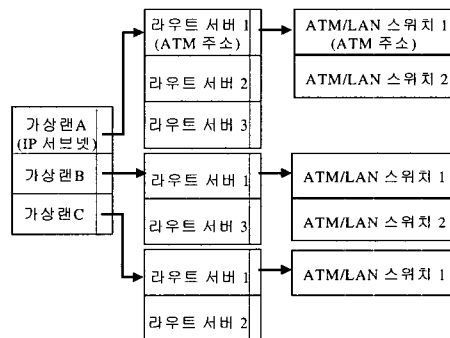


그림 3 그림 2의 유기적 구조에서 라우트 서버 1의 테이블 구조

3.1.2 유기적 구조에서의 ATM/LAN 스위치 테이블 구조

ATM/LAN 스위치는 자신에게 접속되어 있는 LAN 및 호스트들이 가입해 있는 모든 가상랜에 대해 각 가

상랜의 멤버 리스트를 유지한다. 스위치는 가상랜에 등록되지 않은 호스트의 가입을 위한 주소해석 요청 프레임과 이미 등록된 호스트가 데이터를 가상랜에 전송하려고 할 때의 데이터 프레임을 서버에게 전송한다. 또한 목적지까지의 데이터 전송을 담당한다. 그림 4는 그림 2의 ATM/LAN 스위치 4가 유지하는 테이블 구조를 보여준다. ATM/LAN 스위치 4에 연결된 호스트들이 가입한 가상랜 A, B의 주소(IP 서브넷 주소)와 자신에게 접속되어 있는 호스트들 중 각 가상랜에 해당하는 호스트들의 주소 리스트(ATM 주소, MAC 주소)를 유지한다.

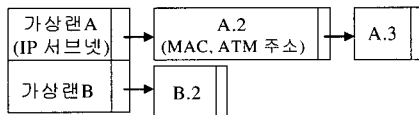


그림 4 그림 2의 유기적 구조에서 ATM/LAN 스위치 4의 테이블 구조

3.1.3 유기적 구조의 동작 방식

■ 가상랜에 등록되지 않은 호스트가 가상랜에 브로드캐스트 하는 과정

ATM/LAN 스위치는 자신에게 등록되지 않은 새로운 MAC 주소로부터 가상랜에 브로드캐스트 되는 패킷을 받았을 때 라우트 서버에게 이 프레임과 함께 주소해석 요청을 보낸다. 즉 호스트는 별도의 가상랜 등록요청을 하지 않지만 새로운 호스트를 발견한 경우 스위치가 자동적으로 호스트의 IP 서브넷 주소를 기반으로 하여 가상랜에 등록시킨다. 호스트가 가상랜에 브로드캐스트하는 데이터는 기존의 VIVID 시스템에서 제안된대로 802.2 LLC/SNAP 포맷으로 인캡슐레이션된다(그림 5, 6)[5][8]. 호스트는 가상랜에 브로드캐스트하려는 802.2 LLC/SNAP 프레임의 근원지 IP 주소 필드와 MAC 주소 필드에 자신의 IP와 MAC 주소를 삽입하고 목적지 IP 주소 필드에는 가상랜 브로드캐스트 주소(예 : 203. 255.176.111)를 삽입하여 스위치에게 전송한다.

ATM/LAN 스위치와 라우트 서버가 주고받는 주소해

석 요청과 응답은 ATMARP 프로토콜의 주소해석 요청/응답 포맷의 프레임을 사용한다(표1)[5][9]. ATM/LAN 스위치에서 라우트 서버로 보내는 주소 해석 요청은 ATMARP의 동작 코드에 요청 프레임임을 나타내고 근원지 ATM 주소 필드에 ATM/LAN 스위치의 ATM 주소를 채운다. 또한 라우트 서버가 ATM/LAN 스위치에게 주소 해석 응답을 하기 위한 프레임은 ATMARP의 동작 코드에 응답 프레임임을 표시하고 가상랜 ID 필드 부분에 가상랜의 IP 서브넷 주소를 채운다.

표 1 ATMARP 제어 프레임 포맷

필드명	설 명
ar\$afn	Address Family (0x0013)
ar\$pro	프로토콜 타입 (0x0800)
ar\$shl	근원지 ATM number의 타입과 길이 (r)
ar\$ssl	근원지 ATM subaddress의 타입과 길이 (q)
ar\$op	동작 코드
ar\$spln	근원지 프로토콜 주소의 길이 (s)
ar\$sthl	목적지 ATM number의 타입과 길이 (x)
ar\$stsl	목적지 ATM subaddress의 타입과 길이 (y)
ar\$stpln	목적지 주소의 길이 (s)
ar\$sha	근원지 ATM number
ar\$ssa	근원지 ATM subaddress
ar\$spa	근원지 프로토콜 주소
ar\$tpa	가상랜 ID
ar\$tha	목적지 ATM number.
ar\$tsa	목적지 ATM subaddress.

ATM/LAN 스위치로부터 주소 해석 요청을 받은 라우트 서버는 호스트에서부터 전송된 802.2 LLC/ SNAP 프레임의 헤더로부터 근원지 IP 주소를 조사하여 서브넷 주소와 관련된 가상랜 정보를 가지고 있다면 해당 가상랜 주소와 근원지 IP 주소, MAC 주소 리스트를 ATMARP 주소해석 응답 프레임에 실어 스위치에게 전송한다. 이를 전송 받은 스위치는 해당 가상랜 리스트에 호스트의 MAC주소와 IP 주소를 등록하고 호스트와 연결된 포트와 MAC 주소를 바인딩한다. 만약 스위치의 테이블에 존재하지 않는 가상랜이라면 가상랜 리스트에

MAC LLC/SNAP Header	Vers	Hlen	Serv_type	Tot_len	ID	Flag	TTL	Protocol	h_ch sum	Src. IP	Dst. IP	IP opt	Pad	Data	FCS
---------------------	------	------	-----------	---------	----	------	-----	----------	----------	---------	---------	--------	-----	------	-----

그림 5 IP 인캡슐레이션 포맷 (802.2 LLC/SNAP)

Preamble	SFD	Dst_MAC	Src_MAC	Length	DSAP	SSAP	Control
----------	-----	---------	---------	--------	------	------	---------

그림 6 MAC LLC/SNAP Header

새로운 가상랜 엔트리를 추가하고 서버로부터 받은 정보를 등록한다(그림 7 참조).

그런데 만약 라우트 서버가 ATM/LAN 스위치로부터 주소 해석 요청을 받았을 때, 802.2LLC/SNAP 프레임의 근원지 IP 서브넷 주소에 해당하는 가상랜을 관리하고 있지 않다면, 서버는 자신과 VC로 연결된 모든 라우트 서버에게 업데이트를 요청한다(그림 8 참조). 라우트 서버간에 주고받는 업데이트 요청/응답 프레임은 단일 라우트 서버만을 두는 기존 VIVID 시스템에는 정의되어 있지 않다. 본 연구에서는 라우트 서버와 스위치간에 주소해석을 위해 사용된 ATMARP 메시지를 이용하여 라우트 서버간에 주고받는 업데이트 메시지를 정의하였다. 업데이트 요청 프레임은 ATMARP의 동작 코드로 업데이트 요청 프레임임을 나타내고 근원지 주소 필드에 라우트 서버의 ATM 주소를 목적지 주소 필드에는 요청 프레임 보낼 라우트 서버의 ATM 주소를 넣고 가상랜 주소 필드에는 호스트의 IP 서브넷 주소를 실어 보낸다. 이 요청을 받은 라우트 서버들은 자신의 테이블을 조사하여 해당 가상랜 주소가 존재하면 그 가상랜의 라우트 서버 리스트에 업데이트를 요청한 라우트 서버의 주소를 추가하고 자신의 ATM 주소를 업데이트 응답 프레임의 근원지 주소에 실어 업데이트를 요청했던 서버에게 전송한다. 이 응답을 받은 서버는 자신의 테이블에 새로운 가상랜 엔트리를 추가하고 이 엔트리에 응답한 서버의 주소와 주소해석 요청을 보냈던 ATM/LAN 스위치의 주소를 삽입한다. 그리고 새로운 가상랜 주소와 호스트의 IP 주소, MAC 주소를 ATMARP의 주소해석 응답 프레임에 실어 ATM/LAN 스위치에게 보낸다. 이를 받은 ATM/LAN 스위치 역시 새로운 가상랜 엔트리를 만들고 그 엔트리에 응답으로 받은 호스트 정보를 삽입한다.

이와 같이 호스트의 가상랜 가입이 진행되는 동안 그 호스트로부터 ATM/LAN 스위치로 전송된 가상랜 브로드캐스트 프레임은 ATM/LAN 스위치에 버퍼링 되었다가 ATM/LAN 스위치에서 가상랜 엔트리가 완성되고 나면 라우트 서버에게 전달한다. 이 단계부터는 가상랜에 등록된 호스트가 브로드캐스트하는 과정과 같다.

그림 7과 8은 IP 주소가 203.255.176.58인 호스트가 새로 등록하는 과정을 나타낸 것이다. 그림 7은 라우트 서버에 ATM/LAN 스위치가 새로 발견한 호스트의 가상랜 (IP 서브넷)에 대한 정보가 있을 경우에 대한 예를 보여주고 있고, 그림 8은 호스트가 자신이 속한 라우트 서버의 테이블에 현재 존재하지 않는 가상랜에 등록하려는 경우의 예를 보여주고 있다.

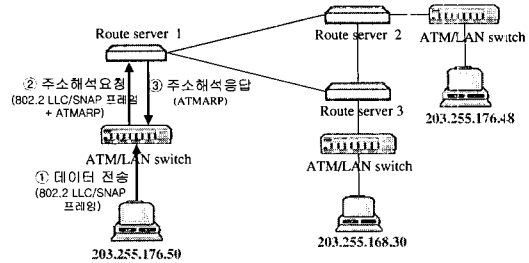


그림 7 그림 2의 유기적 구조에서 라우트 서버 1이 관리하는 구역 내에 현재 존재하는 가상랜에 호스트가 가입하는 경우

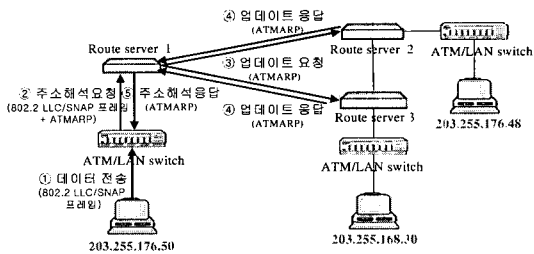


그림 8 그림 2의 유기적 구조에서 라우트 서버 1이 관리하는 구역 내에 현재 존재하지 않는 가상랜에 호스트가 가입하는 경우

■ 가상랜에 등록된 호스트가 가상랜에 브로드캐스트하는 과정

호스트가 ATM/LAN 스위치로 브로드캐스트 패킷을 전송하면 (그림 9의 ①) 스위치는 이 패킷을 바로 라우트 서버에게 전달하고 (그림 9의 ②) 이를 받은 라우트 서버는 해당 가상랜 멤버를 관리하는 다른 라우트 서버에게 브로드캐스트한다(그림 9의 ③). 각 서버는 해당 가상랜을 담당하는 ATM/LAN 스위치에게 ATM/LAN 스위치는 해당 가상랜 멤버에게 브로드캐스트한다. (그림 9의 ④)

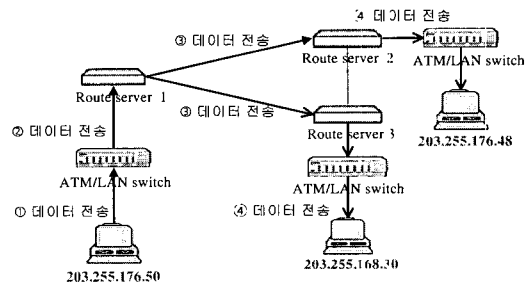


그림 9 유기적 구조에서의 가상랜 브로드캐스트 과정

3.2 독립적 구조

이 방식에서는 브로드캐스트의 전송 속도를 빠르게 하기 위해 라우트 서버가 브로드캐스트 되어야 할 가상랜을 담당하는 각 ATM/LAN 스위치들과 직접 VC를 설정하고 데이터를 전달한다. 라우트 서버에서 유지해야 하는 정보가 많아지고 VC 설정이 많아지기 때문에 확장성이 좋지 않다. 그러나 일단 VC가 설정되면 빠르게 데이터를 전송할 수 있다는 장점이 있다. 그림 10은 독립적 구조의 예를 보여준다. 호스트 A.1이 브로드캐스트 전송을 요청했을 때 라우트 서버와 1과 가상랜 A에 해당하는 호스트가 접속되어 있는 ATM/LAN 스위치 3, ATM/LAN 스위치 4가 VC로 직접 연결된다.

라우트 서버간이나 라우트 서버와 ATM/LAN 스위치 사이 그리고 ATM/LAN 스위치와 호스트 사이에서 전송되는 메시지 종류는 유기적 구조의 경우와 같다.

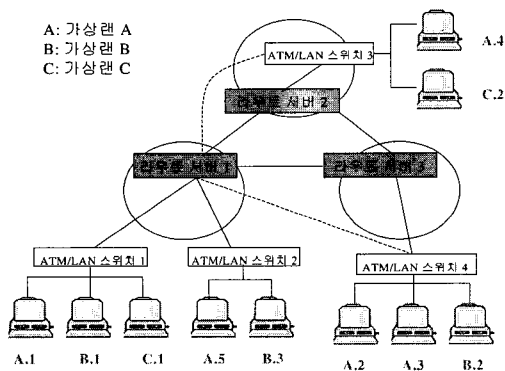


그림 10 독립적 구조의 예

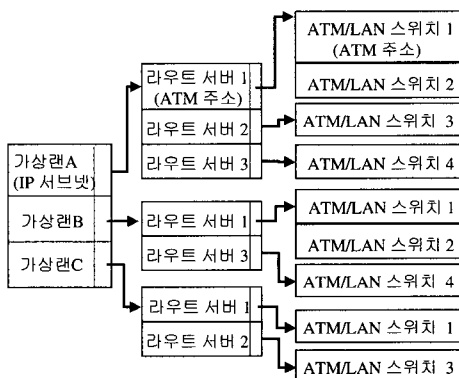


그림 11 그림 10의 독립적 구조에서 라우트 서버 1의 테이블 구조

3.2.1 독립적 구조에서의 라우트 서버 테이블 구조

라우트 서버 테이블은 그림 11과 같이 라우트 서버가 담당한 구역의 호스트가 가입하고 있는 가상랜의 리스트를 유지하고 있고 각 가상랜에 대해서는 이를 관리하는 서버들과, 각각의 서버와 관련된 ATM/LAN 스위치 정보를 유지한다. 유기적 구조에서 라우트 서버가 자신이 담당하는 구역의 ATM/LAN 스위치 정보만을 유지하는 것과 비교해 볼 때 독립적 구조에서 라우트 서버가 유지해야 하는 정보 양이 더 많음을 알 수 있다.

그림 11은 그림 10에서의 라우트 서버 1의 라우트 서버 테이블이다.

3.2.2 독립적 구조에서의 ATM/LAN 스위치 테이블 구조

유기적 구조의 경우와 마찬가지로 ATM/LAN 스위치는 접속되어 있는 LAN 혹은 호스트가 가입해 있는 모든 가상랜에 대해 그 가상랜 멤버들의 리스트를 유지한다. 즉 ATM/LAN 스위치가 유지하는 테이블의 내용은 유기적 구조의 경우와 동일하다.

3.2.3 독립적 구조의 동작 방식

- 가상랜에 등록되지 않은 호스트가 가상랜에 브로드캐스트하는 과정

독립적 구조에서 ATM/LAN 스위치가 새로운 근원지 MAC 주소의 호스트로부터 브로드캐스트 프레임을 받아 라우트 서버에게 주소해석 요청 프레임을 보내는 과정까지는 유기적 구조에서의 과정과 동일하다. 그리고 브로드캐스트 프레임의 포맷과 ATMARP 주소해석 요청/응답 포맷에 포함되는 정보도 같다. 독립적 구조가 유기적 구조와 다른 점은 라우트 서버끼리 주고받는 메시지의 내용이다. 따라서 업데이트 요청/응답 프레임의 형식은 달라진다. 독립적 구조에서는 각 라우트 서버가 자신이 담당하는 모든 가상랜들(자신이 담당하는 구역의 호스트들이 가입해 있는 모든 가상랜들)에 대해 다른 구역의 라우트 서버와 접속해 있는 ATM/LAN 스위치들의 정보까지 유지해야하므로 업데이트 응답 프레임에 여러 개의 ATM 주소리스트를 실을 수 있도록 ATMARP 프레임을 확장시켰다.

ATM/LAN 스위치는 새로운 MAC 주소를 가지는 호스트로부터 브로드캐스트 프레임을 받으면 라우트 서버에게 주소해석 요청 프레임과 함께 호스트가 보낸 브로드캐스트 프레임(그림 12의 ①, ②), 이를 받은 라우트 서버는 브로드캐스트 프레임(802.2LLC/SNAP형식)의 근원지 필드로부터 호스트의 IP 주소를 조사하여 이 IP 서브넷과 관련된 가상랜이 자신의 테이블에 존재하는 경우 해당 가상랜 리스트에 주소 해석

요청을 보낸 ATM/LAN 스위치의 ATM 주소를 등록하고 다른 라우트 서버들 중 이 가상랜을 관리하는 라우트 서버에게만 업데이트 요청 프레임(그림 12의 ④). 그림 12에서 라우트 서버3은 해당 가상랜을 담당하지 않으므로 라우트 서버 1은 라우트 서버 2에게만 업데이트 요청 프레임을 보낸다. 이때, 업데이트 요청 프레임의 근원지 ATM 주소 필드에는 자신의 ATM 주소를 입력하고 ATM/LAN 주소 필드에는 새로운 호스트의 가상랜 등록을 요청한 ATM/LAN 스위치의 주소를 실어 전송한다. 또한 주소 해석 요청한 ATM/LAN 스위치에게 가상랜 주소와 호스트의 IP 주소 및 MAC 주소를 주소해석 응답 프레임에 실어 전송한다(그림 12의 ③). 업데이트 요청 프레임을 받은 라우트 서버는 해당 가상랜 엔트리에 ATM/LAN 주소를 삽입하고 업데이트 응답 프레임에 서버 자신의 ATM 주소와 자신이 담당하는 구역에서 해당 가상랜을 담당하는 ATM/LAN 스위치의 주소 엔트리를 실어 업데이트 요청한 라우트 서버에게 전송한다(그림 12의 ⑤). 이 업데이트 응답 프레임을 받은 라우트 서버는 자신이 유지하고 있던 ATM/LAN 스위치 주소 엔트리와 응답 프레임에 포함된 ATM/LAN 주소 엔트리를 비교하여 새로운 주소가 발견되면 주소 리스트에 주소를 추가한다.

어떤 호스트가 라우트 서버가 관리하는 구역에 현재 존재하지 않는 가상랜에 가입하는 경우의 가상랜 가입 과정은 유기적 구조의 경우(그림 8)와 같다. 단, 주고받는 메시지의 내용과 테이블에 저장하는 내용이 달라지게 된다. 주소해석 요청을 받은 라우트 서버의 테이블에 호스트의 IP서브넷과 관련된 가상랜 정보가 없다면 가상랜 리스트에 새로운 가상랜을 추가하고 담당 ATM/LAN 스위치의 주소를 삽입한다. 또한 라우트 서버는 업데이트 요청 프레임에 주소 해석을 요청한 ATM/LAN 스위치의 ATM 주소와 호스트의 IP 주소를 실어 모든 라우트 서버에게 브로드캐스트한다. 이 프레임을 받은 라우터 서버들은 해당 가상랜이 있는지 테이블을 조사하여 존재하면 가상랜 리스트에서 해당 가상랜 엔트리에 업데이트를 요청한 라우트 서버의 ATM 주소와 ATM/LAN 스위치의 주소를 새로 삽입한다. 또한 응답하는 서버 자신의 ATM 주소와 자신이 담당하는 구역에서 해당 가상랜을 담당하는 ATM/LAN 스위치의 주소 리스트를 업데이트 응답 프레임에 실어 테이블 업데이트 요청에 대해 응답한다. 이 응답 프레임을 받은 라우트 서버는 응답한 라우트 서버의 ATM 주소와 그 라우트 서버에 속한 해당 가상랜 담당 ATM/LAN 스위치의 주소 리스트를 해당 가상랜 엔트리에

추가한다.

이와 같이 호스트의 가상랜 등록이 끝나면 호스트 등록이 진행되는 동안 ATM/LAN 스위치에 버퍼링 되었던 브로드캐스트 프레임은 라우트 서버에게 전달된다. 그 이후 과정은 다음 가상랜에 등록된 호스트가 가상랜 멤버들에게 브로드캐스트하는 과정과 같다.

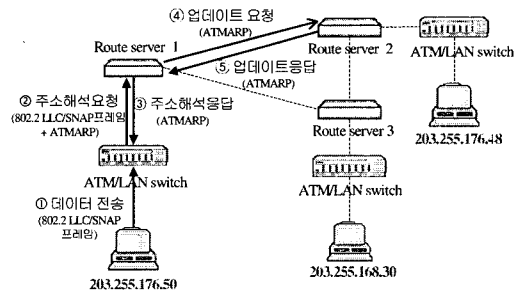


그림 12 그림 10의 독립적 구조에서 라우트 서버 1이 관리하는 구역 내에 이미 존재하는 가상랜에 호스트가 가입하는 경우

■ 가상랜에 등록된 호스트가 가상랜에 브로드캐스트하는 과정

호스트가 ATM/LAN 스위치로 브로드캐스트 패킷을 전송하면 (그림 13의 ①) 스위치는 이 패킷을 바로 라우트 서버에게 전달한다. (그림 13의 ②) 이를 받은 라우트 서버는 해당 가상랜 멤버를 담당하는 ATM/LAN 스위치와 VC를 맺어 ATM/LAN 스위치에게 브로드캐스트하고 (그림 13의 ③) 스위치는 다시 해당 가상랜 멤버에게 브로드캐스트한다(그림 13의 ④). 라우트 서버와 ATM/LAN 스위치 간에 설정된 VC는 일정 시간 동안 전송되는 데이터가 없으면 자동적으로 해제하게 된다.

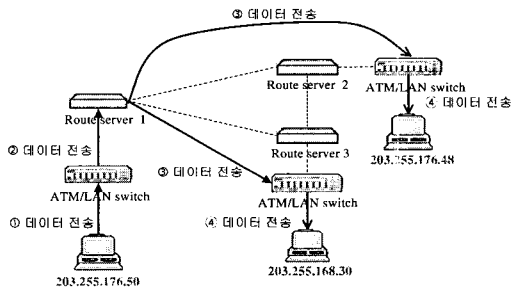


그림 13 독립적 구조에서의 가상랜 브로드캐스트 과정

4. 시뮬레이션



본 장에서는 유기적 구조와 독립적 구조의 성능을 비교하기 위한 가상랜 가입 및 브로드캐스트 데이터 전송 시뮬레이션을 설명하고 그 결과를 분석한다. 성능분석을 위한 시뮬레이션은 SDL (Specification and Description Language)을 사용하여 구현하였다. SDL은 시스템을 명세화하고 기술하는 표준화된 언어로서 통신분야에서 여러 동시 작업과 상호동작이 중요시되는 사건 중심의 실시간 시스템을 기술하는데 적합하다[10]. SDL을 이용하여 프로토콜을 구현할 경우 보다 편리하고 빠른 시간 안에 구현이 가능하며, 명확하게 구현할 수 있고, 명시한 사실이 바뀌더라도 간편하게 변경 내역을 수정할 수 있다. SDL을 사용하면 전체 시스템을 블록들과 그 연결로 표현할 수 있고 각 블록은 다시 프로세스들로 구성되도록 정의할 수 있어 계층적인 시스템 표현이 가능하다. 이를 그래픽으로 표현함으로써 타 언어로 구현하는 것보다 쉽게 이해되며, 구현된 시스템에 대한 단계적인 분석이 가능하고 기능에 따른 시뮬레이션 및 검증이 용이하다[11]. SDT(SDL Tool)는 SDL 사용자를 위한 툴이다. SDT에서 제공하는 SDL 에디터는 Z.100에서 표준화한 그래픽적인 SDL의 표기방식을 사용하여 시스템을 명세화하고 기술할 수 있게 하며, SDL 시뮬레이터는 사용자가 자신이 명세화한 시스템의 동작을 이해하고 디버깅할 수 있게 한다[12]. 본 시뮬레이션은 Ultrasparc Sun OS 5.6 버전의 운영체제를 사용하고 CPU가 167MHz, RAM이 132M인 워크스테이션에서 SDL로 구현하였다. 4.1절에서 먼저 유기적 방식과 독립적 방식의 성능 비교를 위해 시뮬레이션에서 사용한 네트워크 모델 및 트래픽 생성 모델을 제시한다. 4.2절에서는 시뮬레이션 결과에 따라 두 방식의 성능을 비교 분석한다.

4.1 시뮬레이션 네트워크 모델

라우터 서버를 이용한 3계층 가상랜을 구현하는 VIVID 시스템은 ATM/LAN 스위치 정보를 유지하는 라우트 서버와 가상랜 멤버를 관리하는 ATM/LAN 스위치 그리고 호스트로 구성된다. 본 시뮬레이션을 위한 네트워크 모델은 그림 14와 같다. 세 개의 라우트 서버가 VC로 연결되어 있으며 각 라우트 서버에 두 개의 ATM/LAN 스위치가 연결되어 있고 각 스위치에는 20개의 호스트가 연결되어 있다.

이와 같은 네트워크 모델을 SDL의 가장 상위 레벨에서 구현한 것이 그림 15이다. 그림 15의 라우트 서버와 ATM/LAN 스위치, 호스트를 나타내는 각 블록은 3개의 서버와 6개의 ATM/LAN 스위치, 120개의 호스트 프로세스로 구성된다. 이들 프로세스들은 각각 FIFO

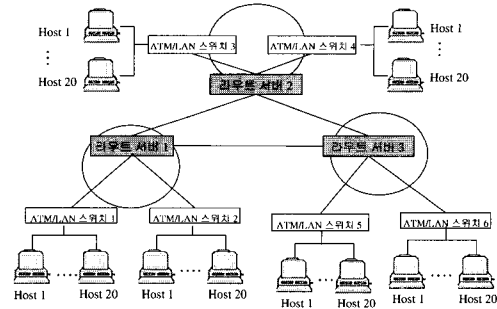


그림 14 시뮬레이션 네트워크 모델

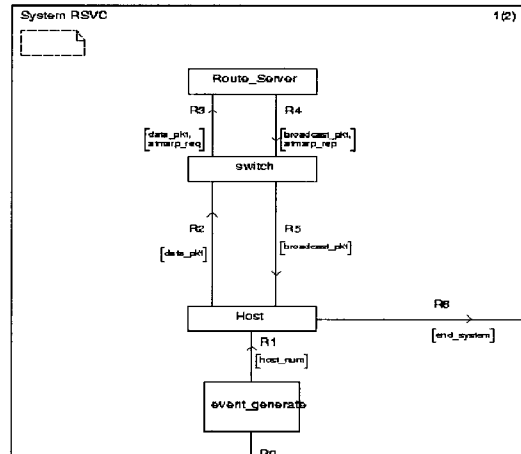


그림 15 SDT에 의한 시뮬레이션 시스템 구성도

(First In First Out)큐와 서버로 해당 엔터티를 모델링한다. 즉, 이들 프로세스에 도착하게 되는 각 패킷은 일단 그 프로세스의 큐의 끝에 저장되며, 프로세스의 서버는 큐의 머리에 있는 패킷에 대해 해당 엔터티의 프로세싱에 소요되는 시간 및 채널 용량에 따른 전송 지연에 해당하는 시간이 경과한 후 이를 다음 엔터티에 해당하는 프로세스에게로 보내고 다음 패킷을 서버한다. 어떤 프로세스의 서버를 떠난 패킷은 채널 전파지연에 해당하는 시간이 추가적으로 경과한 후 그 프로세스가 지정한 다음 프로세스에 도착하게 된다.

입력 트래픽은 브로드캐스트 데이터 전송 요구인테이들이 발생하는 간격을 조절해보면서 시뮬레이션의 결과를 수집하였다. 특정 가상랜에게 전송되는 일련의 브로드캐스트 프레임의 흐름을 세션이라고 정의할 때, 세션의 길이와 세션간의 도착 시간 간격 그리고 세션 안에서 발생하는 브로드캐스트 프레임간의 도착시간 간격

은 지수 분포를 따른다고 가정하였다. 브로드캐스트 세션의 데이터 전송 요구는 그림 15의 Event 블록에서 발생하여 호스트로 전달된다. Event 블록은 세션을 발생시키는 모듈과 발생된 각 세션에 대해 데이터 전송요구를 발생시키는 모듈로 구성된다. 세션 발생 모듈에서는 세션 발생 시간과 세션 길이를 결정하고 균일 분포에 따라 무작위로 브로드캐스트 데이터를 전송할 근원지 호스트를 선택한다. 세션 발생 모듈이 결정한 세션 시작 시간이 되면 그 세션에 대한 브로드캐스트 데이터 전송요구를 발생시키는 모듈이 작동하여 해당 세션의 근원지 호스트 블록에게 데이터 발생 신호를 보낸다. 브로드캐스트 데이터 프레임의 크기는 512 바이트라고 가정하였고, 제어 메시지의 크기는 ATMARP 권고안에서 제시한 형식에 따라 메시지 크기를 구한 결과 경우에 따라 52 바이트~92 바이트 사이의 값을 가지게 된다. 또한, 각 엔터티 간의 전파지연은 표 2와 같이 가정하였고, 각 엔터티를 연결하는 채널용량은 VIVID 시스템의 규정에서 제시한 값을 따라 표 3와 같이 가정하였다. 그리고, 독립적 구조에서 가상랜 브로드캐스트시 라우트 서버와 해당 가상랜을 담당하는 ATM/LAN 스위치간의 VC 셋업을 위해 각 엔터티에서 소요되는 프로세싱 타임은 5msec라고 가정하였다.

표 2 각 엔터티 간의 전파지연

엔터티	전파지연시간
라우트 서버 - 라우트 서버	150 $\mu$ s
라우트 서버 - ATM/LAN 스위치	50 $\mu$ s
ATM/LAN 스위치 - 호스트	10 $\mu$ s

표 3 각 엔터티 간의 채널 용량

엔터티	대역폭
라우트 서버 - 라우트 서버	150 Mbps
라우트 서버 - ATM/LAN 스위치	150 Mbps
ATM/LAN 스위치 - 호스트	10 Mbps

4.2 시뮬레이션 결과 분석

이 절에서는 본 논문에서 제안한 두 가지 3계층 가상랜 구성 모델의 성능을 평가하기 위해 시뮬레이션 결과를 제시하고 이를 분석한다. 시뮬레이션 파라미터로 평균 세션 길이와 세션간의 도착 시간 간격 그리고 한 세

션 내에서의 프레임 발생 간격을 변화시키면서 평균 데이터 전송시간과 데이터 메시지당 발생하는 총 제어 메시지 양이 어떻게 변화하는지 알아보았다.

그림 16은 평균 세션 길이가 20초이고 세션간의 평균 도착 시간 간격이 3초인 경우 한 세션 내에서의 프레임 발생 간격을 변화시켰을 때 유기적 모델과 독립적 모델의 평균 데이터 전송 시간을 보여주는 그래프이다. 그래프에서 RS-RS에 해당하는 선은 유기적 구조의 결과이고 RS-SW에 해당하는 선은 독립적 구조의 결과이다. 데이터 전송 시간 측면에서는 프레임 발생 간격이 짧을수록 독립적 구조가 유리해지고 프레임 발생 간격이 길수록 유기적 구조가 유리함을 볼 수 있다. 독립적 구조의 경우 브로드캐스트 세션이 시작될 때 근원지 라우트 서버와 해당 가상랜을 서브하는 원격 ATM/LAN 스위치간에 VC를 설정해야 하는 오버헤드가 있지만 일단 VC가 설정되고 나면 효율적으로 데이터 전송이 이루어진다. 반면, 유기적 구조의 경우는 세션 초기의 VC 설정 오버헤드는 없지만 데이터가 매번 해당 가상랜에 속하는 호스트가 속해 있는 라우트 서버를 경유해서 전달되므로 데이터 양이 늘어나게 되면 라우트 서버에서 병목 현상이 발생할 수 있다. 따라서 세션 내에서의 평균 데이터 프레임 발생 간격이 짧아져 발생하는 데이터 프레임 수가 많아지게 되면 독립적 구조의 세션 초기 VC 설정 오버헤드가 유기적 구조의 매 데이터 프레임 당 발생하는 라우트 서버 경유 오버헤드에 상쇄되어 독립적 구조의 성능이 유기적 구조의 성능을 능가하게 된다.

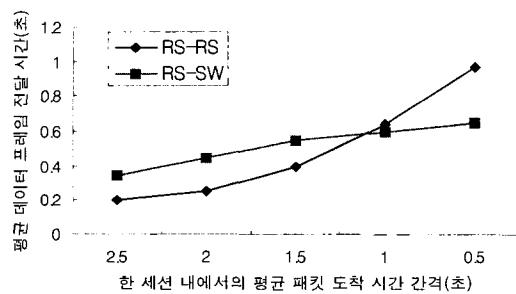


그림 16 프레임 발생 간격 변화에 대한 평균 데이터 프레임 전달시간

그림 17은 한 세션 내에서의 평균 프레임 발생 간격을 0.5초로 고정시키고 평균 세션 길이를 변화시켰을 때 평균 데이터 전송 시간을 측정한 것이다. 세션의 길이가 길어질수록 독립적 모델이 유기적 모델보다 전송 시간에 있어 유리해짐을 볼 수 있다. 이와 같은 결과는 앞에서

설명했듯이 독립적 모델은 새로운 세션이 시작되는 경우 VC를 설정하기 위한 오버헤드가 크지만 일단 지속적으로 특정 근원지 호스트로부터의 브로드캐스트가 이루어지는 경우 이미 설정된 VC를 통해 근원지 라우트 서버와 ATM/LAN 스위치간에 데이터 전송이 직접적으로 이루어지기 때문이다. 세션의 길이가 길어지면 데이터 발생이 많아져 두 모델 모두 평균 데이터 전송 시간이 증가하지만 세션 초기의 VC 설정 오버헤드 비중이 적어져 상대적으로는 독립적 구조의 성능이 유기적 구조보다 나아지게 된다.

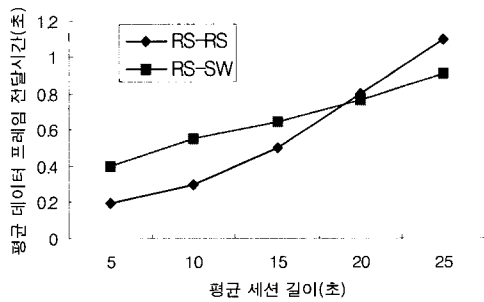


그림 17 세션 길이 변화에 대한 평균 데이터 프레임 전달시간

그림 18은 한 세션 내에서의 평균 프레임 발생 간격을 0.5초로 고정시키고 평균 세션 길이를 변화시켰을 때 유기적 모델과 독립적 모델에서 데이터 메시지당 발생하는 제어 메시지의 비율을 나타낸 것이다. 독립적 구조가 유기적 구조에 비해 발생하는 제어 메시지 양이 많은데 그 이유는 다음과 같다. 어떤 ATM/LAN 스위치가 자신이 담당하지 않았던 새로운 가상랜을 담당하게 되었을 때의 상황을 생각해보면 유기적 모델에서는 그 ATM/LAN 스위치가 접속해 있는 라우트 서버만 새로운 ATM/LAN 스위치의 정보를 유지하고(그림 7) 있으면 되지만, 독립적 모델에서는 그 ATM/LAN 스위치가 접속해 있는 라우트 서버뿐만 아니라 해당 가상랜을 담당하고 있는 다른 라우트 서버에게도 새로운 ATM/LAN 스위치가 추가되었음을 알리기 위해 업데이트 요청 프레임을 발생시켜야 한다(그림 12). 또한 이때 응답으로 오는 업데이트 응답 프레임에는 해당 가상랜과 관련된 모든 ATM/LAN 스위치의 주소 리스트가 포함되기 때문에 그 크기가 다른 제어 메시지 보다 크다. 각 방식에 대해 발생한 총 제어 메시지의 양은 일정하지만 세션이 길수록 데이터 양이 많아지므로 데이터

메시지당 발생된 총 제어 메시지의 비율은 두 모델 모두 세션이 길어질수록 감소하게 된다. 그리고 독립적 모델의 제어 메시지 양이 유기적 모델에 비해 많으므로 모든 세션 길이에 대해 총 제어 메시지 비율은 독립적 구조가 유기적 구조보다 높다.

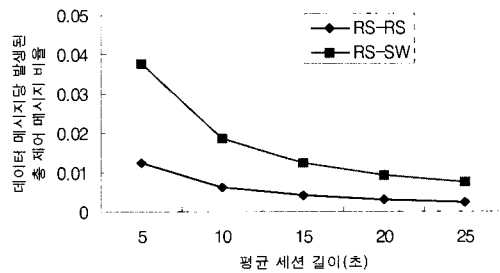


그림 18 데이터 메시지 당 발생하는 총 제어 메시지의 비율

그림 19는 ATM/LAN 스위치의 개수에 따른 라우트 서버의 테이블 사이즈를 나타낸다. 스위치의 개수가 증가함에 따라 유기적 구조의 테이블 사이즈는 거의 변화가 없는 반면 독립적 구조는 많은 메모리를 필요로 하게 된다. 그 이유는 유기적 구조의 라우트 서버는 자신이 관리하는 구역의 스위치의 정보만을 유지하고 독립적 구조의 라우트 서버는 자신이 관리하는 모든 가상랜에 대해 모든 스위치의 정보를 유지해야 하기 때문이다.

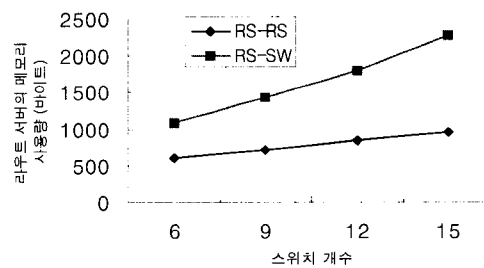


그림 19 ATM/LAN 스위치 개수에 따른 라우트 서버의 테이블 사이즈

5. 결론

본 논문에서는 ATM-LAN 스위치 망에서 3 계층 가상랜을 구현하는 VIVID 시스템을 확장하는 방안으로 다중의 라우트 서버를 두는 두 가지의 모델을 제시하고, 시뮬레이션을 통하여 그 성능을 비교하였다. 유기적 구

조라 명칭한 모델에서는 브로드캐스트 데이터가 항상 목적지 가상랜을 담당하는 라우트 서버들을 경유하여 가상랜 멤버 호스트들에게 전달된다. 독립적 구조라 명칭한 모델에서는 근원지 라우트 서버가 가상랜 멤버가 연결되어 있는 ATM/LAN 스위치들과 직접 VC를 설정하여 가상랜 브로드캐스트 데이터를 전송한다. 독립적 구조의 경우에는 어느 특정 가상랜에 대한 브로드캐스트가 시작되면 근원지 라우트 서버와 해당 가상랜을 담당하는 ATM/LAN 스위치 간에 브로드캐스트 데이터 전송을 위한 VC를 설정해야 하므로 오버헤드가 크지만 세션이 진행되는 동안에는 라우트 서버들을 경유할 필요 없이 설정된 VC를 통해 직접 브로드캐스팅이 가능하므로 신속한 데이터 전송이 가능하다. 독립적 구조의 경우 가상랜 브로드캐스트 세션의 길이가 길어지고, 한 브로드캐스트 세션에서 발생하는 프레임간 간격이 짧아 질수록 세션 초기 VC 설정을 위한 추가 오버헤드 비중이 적어져 세션 길이나 세션에서 발생하는 프레임 수가 어느 수준 이상에 이르면 유기적 구조 보다 더 나은 성능을 보이게 된다. 그러나 독립적 구조의 경우는 각 라우트 서버가 다른 라우트 서버들이 담당하는 구역의 가상랜 정보까지 모두 유지하고 있어야 하기 때문에 라우트 서버간에 교환해야 하는 정보가 많고 라우트 서버에서의 메모리 요구량도 네트워크 규모에 비례하여 커지게 된다. 또한 가상랜 멤버들이 접속되어 있는 ATM/LAN 스위치의 수에 비례하여 VC 설정 오버헤드도 커지게 된다. 따라서 확장성 면에서는 유기적 모델이 더 우수하며, 브로드캐스트 세션의 길이와 세션 내에 발생하는 브로드캐스트 프레임의 수가 어느 수준 이하인 경우에는 성능면에서도 유기적 구조가 더 우수하다.

### 참 고 문 헌

- [1] Draft Standard P802.1Q/D11 IEEE Standards for Local and Metropolitan Area Networks: "Virtual Bridged Area Networks," July 30, 1998.
- [2] 김희정, 서주연, 채기준, 이미정, 강훈, 최길영, 김성해, "SDL을 이용한 ATM-LAN 스위치 망에서의 정책 기반 가상랜 구현", 한국정보과학회 가을 학술 발표논문집 Vol. 26 No. 2, 1999.
- [3] Marina Smith, "Virtual LANs A Guide to Constructions, Operation, and Utilization," McGraw-Hill, 1997.
- [4] 3Com, "3Com Transcend VLANs," [http://www.3com.com/technology/tech\\_net/white\\_papers/537VLAN.html](http://www.3com.com/technology/tech_net/white_papers/537VLAN.html).
- [5] "VIVID Route Server Release 1.1," <http://www.newbridge.com/techinfo/Descriptions/RDs/InterNetwork/index.html>.

- [6] The ATM Forum, "Multiprotocol Over ATM Version 1.0 (Letter Ballot)," May. 1997.
- [7] The ATM Forum. "LAN Emulation over ATM Version 2, LUNI Specification (Straw Ballot)," Feb. 1997.
- [8] Mark A. Miller, P.E., "LAN PROTOCOL HANDBOOK," M&T Publishing, Inc. 1990.
- [9] M. Laubach, J. Halpern, "Classical IP and ARP over ATM," RFC 2225, April. 1998.
- [10] SDT 3.2 Getting Started. SDL 매뉴얼.
- [11] SDT Methodology Guidelines.
- [12] Rolv Braek, "SDL Basic," Computer Networks and ISDN System 1996.



서 주 연

1998년 이화여자대학교 컴퓨터학과 졸업 (학사). 2001년 이화여자대학교 과학기술대학원 컴퓨터학과(공학석사). 2001년 ~ 현재 코리아링크 기술 연구소. 관심분야는 가상랜, 고속 통신 프로토콜, 멀티캐스트 프로토콜

이 미 정

정보과학회논문지: 정보통신  
제 29 권 제 1 호 참조