

碩士學位論文

# 품질보장 VoD 서비스를 위한 오버레이 멀티캐스트

濟州大學校 大學院

컴퓨터工學科

金成修

2007年 12月

# 품질보장 VoD 서비스를 위한 오버레이 멀티캐스트

지도교수 안 기 중

김 성 수

이 논문을 공학 석사학위 논문으로 제출함

2007 年 12 月

김성수의 공학 석사학위 논문은 인준함

審査 委員長 \_\_\_\_\_ 印

委 員 \_\_\_\_\_ 印

委 員 \_\_\_\_\_ 印

濟州大學校 大學院

2007 年 12 月

Overlay Multicast for QoS guaranteed  
VoD Service

Sung-su Kim

(Supervised by professor Khi-jung Ahn)

A thesis submitted in partial fulfillment of the requirement for  
the degree of Master of Computer Engineering

2007. 12.

This thesis has been examined and approved.

Thesis director, \_\_\_\_\_

Thesis director, \_\_\_\_\_

Thesis director, \_\_\_\_\_

December 2007

Department of Computer Engineering

Graduate School

Cheju National University

## 감사의 글

제가 98년도에 처음 제주 대학교 컴퓨터 공학과를 입학하면서 학부생을 거쳐 대학원 까지 컴퓨터 공학과와 함께 지낸 날일 10년이 되었네요.

그동안 컴퓨터에 대해서 아무것도 모르던 제가 이제는 다른 이가 컴퓨터에 대해서 물어 보게 된다면 어느 정도 도움을 줄 수 있을 정도로 지식을 갖게 되었습니다. 이런 제가 있기까지 학부생에서부터 가르침을 주신 김장형 교수님, 안기중 교수님, 곽호영 교수님, 이상준 교수님, 변상용 교수님, 변영철 교수님, 김도현 교수님과 지금 캐나다에 계신 송왕철 교수님께 깊이 감사드립니다.

처음에 대학원을 들어오면서 많은 어려움이 있었는데 그때 마다 저에게 부족한 점과 나아갈 길을 지도 해주신 송왕철 교수님께 다시 한 번 감사 하다는 말씀드립니다.

그리고 같은 연구실에 김강석 선배, 허지완 선배, 양덕성 선배, 정우철 선배, 이경진 선배, 이정윤 선배 등 다른 연구실에 많은 선배님들과 현석, 명성, 창준 등 후배들이 있었기에 학업 증진에 많은 도움이 되었습니다. 그리고 경복이형, 연미누나, 은경이 누나, 정아 누나, 혜선이, 부림이 등 여러 도움을 저에게 준 점 정말 감사드립니다.

지금도 많이 부족한 저이지만 앞으로도 열심히 학업에 충실하여 앞으로 점점 나아지는 모습을 보이도록 노력 하겠습니다.

2008년 1월  
김 성 수 올림

# 목차

그림목차 .....	i
국문 초록 .....	ii
영문 초록 .....	iii
I. 서론 .....	1
II. 연구 이론적 배경 .....	4
1. Vod 서비스 .....	4
1) VOD 시스템의 등장배경 .....	4
2) VOD 시스템의 정의 .....	5
3) VOD 시스템의 특징 .....	5
2. VoD 데이터 전송 방식 .....	7
1) 유니캐스트(Unicast) .....	7
2) 브로드캐스트(Broadcast) .....	7
3) 멀티캐스트(Multicast) .....	7
3. IP 기반 멀티캐스트 기법 .....	8
1) Batching .....	9
2) Adaptive Piggy backing .....	10
3) Pyramid Broadcasting .....	10
4. 응용 계층 멀티 캐스트 .....	10
1) Overlay Multicast .....	10
2) Patching 기법 .....	12
III. Peer-to-Peer 구조 VoD 서비스 .....	16
1. 시스템 구조 .....	16
2. Peer-to-Peer 구조 VoD 서비스 기본동작 .....	20
1) 비디오 스트림의 블록단위 .....	21
2) 클라이언트의 가입 .....	22
3) 세션 .....	23
4) 장애복구 .....	24
IV. Peer-to-Peer 구조 VoD 서비스 시뮬레이션 .....	26
1. 시뮬레이션 환경 및 설정 .....	26
2. 시뮬레이션 결과 및 고찰 .....	27

V. 결론 및 향후 연구 ..... 31  
    참고 문헌 ..... 33



## 그림 목차

그림 1. Unicast, Broadcast, Multicast .....	7
그림 2. IP Multicast .....	8
그림 3. Overlay Multicast .....	10
그림 4. 오버레이 멀티캐스트 기술 분류 .....	11
그림 5. P2Cast 시스템 구조 .....	13
그림 6. Client 1, 2의 스트림 비교 .....	13
그림 7. P2VoD 시스템 구조 .....	15
그림 8. P2P Vod 시스템 전체 구조 .....	17
그림 9 각 비디오 프레임 관리 블록 단위 .....	21
그림 10 Client 5의 기존 세션 Tree 조인 및 .....	22
그림 11 기존에 서비스 받고 있는 Client4 와 새로 조인한 Client 5의 .....	24
그림 12 시뮬레이션 네트워크 구성 .....	26
그림 13. Net 1 Peer의 데이터 전송량 .....	28
그림 14. Net 2 Peer의 데이터 전송량 .....	29
그림 15. Net 3 Peer의 데이터 전송량 .....	29

## 품질보장 VoD 서비스를 위한 오버레이 멀티캐스트

컴퓨터공학과 김성수  
지도교수 안기중

오늘날 인터넷을 통한 UCC(User Created Contents), IPTV, 화상 회의 등 많은 멀티미디어 서비스가 증가되어 가고 있다. 더불어 사용자들은 보다 나은 품질의 멀티미디어 서비스를 원하게 되었고 이에 멀티미디어 데이터는 점점 대용량화 되어 가고 있다. 현재 가장 많이 사용하고 있는 VoD 서비스에서 고품질 대용량 멀티미디어 서비스를 하기 위해서는 기존의 Unicast, Multicast, IP Multicast 방식은 많은 무리가 있고 또한 오버레이 멀티캐스트 기법인 P2Cast는 서버 병목현상이나 P2VoD는 트리 구성 메카니즘의 복잡한 단점이 있다.

본 논문에서는 고품질의 QoS가 보장된 VoD 서비스를 위한 P2P 방식의 오버레이 멀티캐스트 기법을 제안하고 있다. 이 기법을 이용하게 된다면 VoD 서비스를 받기 위한 Client는 자신의 갖고 있는 네트워크를 자원을 최대한 이용하여 현재 플레이되고 있는 프레임 이후의 프레임을 단기간에 최대한 많은 양의 스트림 데이터를 받을 수 있다. 그 결과 고품질의 QoS가 보장된 VoD 서비스를 사용자에게 제공할 수 있을 것이다.



# Summary

## Overlay multicast for QoS guaranteed VoD service

Sung-su kim

Dept. of Computer Engineering

The Graduate School

Cheju National University

Nowadays, demand for high QoS (Quality of Service) provisioning is increasing in multimedia because of increasing usage of multimedia contents, services etc. User Created Contents (UCC), IPTV, video conference use multimedia data of high quality or large size, but traditional Unicast, Multicast, IP Multicast are limitations for VoD services. Also P2Cast can become a bottleneck for heavy load at the peers and P2VoD has a complex mechanism for building tree of groups.

In this paper, The author propose a overlay multicast technique that extends P2Patching for QoS guaranteed VoD service. Clients can receive many data streams from other peers in short time. As a result, The author will expect high QoS guaranteed VoD service

## I. 서 론

오늘날 인터넷을 통하여 IPTV, 화상 회의, 인터넷 영화 서비스 등 많은 멀티미디어 서비스가 이루어지고 있다. 또한 인터넷 비디오 광고나 UCC (User Created Contents)와 같이 사용자 자신들만의 비디오 파일을 제작하여 동영상 게시판에 올려 많은 사용자들이 이를 이용하는 등 멀티미디어 사용자들은 더욱 증가하고 있다. 이에 사용자들은 좀 더 나은 품질의 멀티미디어 서비스를 원하게 되었고 그 결과 멀티미디어 데이터 파일이 점점 고품질·대용량화 되어가고 있다. 하지만 네트워크의 자원은 한정 되어 있기 때문에 멀티미디어 데이터가 대용량화 되어 갈수록 서버 자원의 고갈, 지연과 지연 등 많은 문제점이 나타나게 된다. 다행히 네트워크 전송 기술의 발전과 사용자 PC의 향상으로 네트워크 자원이 풍부해 지고 있어 네트워크 자원을 최대한 효율적으로 이용하여 단기간에 많은 비디오 데이터를 전송 받을 수 있게 된다면 이런 문제점을 해결 할 수 있어 대용량 멀티미디어 서비스도 가능 하게 될 것 이다.

초기에 인터넷을 통한 비디오 서비스는 비디오 서버로부터 비디오 데이터를 전송 받아 플레이 하는 방식을 사용했었다. 이 방식은 서버로부터 데이터를 모두 전송 받아야만 사용자가 비디오 플레이를 실행 시킬 수 있었으며 전송 받는 동안 사용자에게 기다림을 요구하여 사용자들로 하여금 많은 불편함을 느끼게 했다. 이에 현재 가장 많이 이용하고 있는 VoD(Video-on-Demand) 서비스가 등장 하였다. VoD 서비스는 인터넷을 통하여 비디오 데이터 스트림을 사용자에게 전송함으로써 사용자로 하여금 비디오 데이터를 전송 받는 동안에도 비디오 플레이를 가능 하도록 하였다. VoD 서비스는 비디오를 플레이 하는 동안에 현재 플레이 되고 있는 스트림의 나머지 스트림을 전송 받아 플레이 하는 방식을 사용함으로써 사용자로 하여금 비디오 스트림을 모두 전송 받을 때까지의 기다리는 불편함을 없앴다. 하지만 VoD 서비스가 나올 당시에는 많은 각광을 받았지만 비디오 데이터 파일이 점점 커짐에 따라 많은 문제점이 나타나기 시작하였다. 많은

수의 클라이언트가 서버에 연결되어 서비스를 받을 때 서버 측의 대역폭과 같은 자원 문제와 이에 따르는 확장성 문제 등 많은 제약을 가지고 있었다. 또한 서버와 클라이언트를 유니캐스트로 연결한 전형적인 VoD 서비스 방식은 인기가 높은 콘텐츠에 대해서는 필연적으로 서버 측의 부하를 가져오는 등 많은 문제점을 발생 시켰다.

이에 대한 해결책으로 VoD 서비스를 위한 기술로서 IP 멀티캐스트가 적절한 메커니즘으로 여겨져 왔으나 IP 멀티캐스트[1] 역시 다수의 구성원들에게 데이터를 전송할 때 제어신호의 과다 발생과 라우터에서 경로 계산의 복잡도 증가로 인한 문제 등 많은 문제점을 발생 시켰다. 또한 완벽한 IP 멀티캐스트를 구현하기 위해서는 인터넷상에 존재하는 대부분의 라우터들이 멀티캐스트 기능을 지원하도록 수정·교체해야만 하는 등 많은 문제점을 안고 있어 현재까지도 실제 상업망에서 제한되어 있다. 이러한 이유로 IP 멀티캐스트는 현실적으로 사용하기 어렵다.

이처럼 현실성이 부족한 IP 멀티캐스트의 대안으로 IP계층 대신에 응용계층에서의 멀티캐스트 기능을 구현하는 오버레이 멀티캐스트가 최근 제안되어 활발한 연구가 이루어지고 있다. 비록 IP멀티캐스트와 비교하여 오버레이 멀티캐스트는 물리적 링크 상에서 중복된 트래픽을 더 많이 발생 시키지만 오버레이 멀티캐스트에서는 멀티캐스트에 참여하는 몇몇 호스트가 멀티캐스트 라우터의 기능을 대신할 수 있으며 현존하는 라우터에 새로운 부가기능을 추가하지 않기 때문에 오버레이 멀티캐스트를 현재 인터넷에 바로 적용할 수 있는 등 많은 이점을 가지고 있다. 하지만 오버레이 멀티캐스트의 가장 큰 문제점인 적절 하지 않은 트리 구성과 VoD 특성인 잦은 호스트의 이탈, 조인, 호스트의 Failure 발생 시 지터와 지연에 의해 QoS를 보장하지 못하고 있어 이에 적절한 트리의 설계 및 failure 발생 시에 이를 최소화 할 수 있는 트리구성에 대한 P2Cast[2], P2VoD[3]등 다양한 연구들이 있어왔다. 이런 연구들 모두 적절한 트리의 설계와 호스트 Failure 발생 시 처리 방안에 대한 연구들로서 현재 사용자들이 요구 하고 있는 고품질 대용량의 멀티미디어 서비스를 제공하는 VoD 서비스를 위한 충분한 QoS를 제공 하지 못하고 있다. 따라서 충분한 QoS가 보장되고 고품질 대용량의 멀티미디어 서비스가 가능한 VoD 서비스 시스템이 절실히 요구 되고 있다.

본 연구의 목적은 현재 요구 되고 있는 충분한 QoS가 보장된 고품질 대용량의 멀티미디어 서비스를 제공 할 수 있는 VoD 서비스 시스템 구축을 목적으로 한다.

현재 네트워크 전송 기술과 사용자 PC의 향상으로 자원이 풍족해진 현실을 감안 한다면 사용자 PC의 기능을 더욱 높여 호스트로서의 역할을 충분히 수행 할 수 있을 것이다. 이에 VoD 서비스를 비롯한 멀티미디어 데이터들의 실시간성이라는 특성을 Peer-to-Peer 서비스 특성에 더욱 주목 한다면 이를 해결 할 수 있을 것이다. 따라서 Peer의 네트워크 자원을 최대한 이용 할 수 있을 것이고 Peer의 현재 플레이 되고 있는 스트림 이후의 스트림을 VoD 서버가 아닌 중간 Peer들로부터 단기간 안에 최대한 많은 양의 스트림을 미리 전송 받을 수 있어 시간의 제약으로 인한 지연과 지터를 해결 할 수 있을 것이다. 결과적으로 사용자에게 고품질이 멀티미디어 VoD 서비스를 제공할 수 있는 시스템을 사용자에게 제공 할 수 있을 것이다.

본 논문의 구성은 제 2장에서 인터넷 동영상 서비스에서 현재 가장 많이 이용되고 있는 VoD 서비스와 기존 연구 현황에 대해서 간단히 알아보고 제 3장에서는 본 논문에서 제안 하고 있는 P2P 구조의 VoD 서비스 시스템을 기술한다. 그리고 제 4장에서는 제안한 P2P 구조의 VoD 서비스 시스템 시뮬레이션 결과와 P2Cast를 비교 분석 한다. 마지막으로 제 5장에서는 본 연구의 결론과 향후 연구 방향을 제시한다.

## II. 연구 이론적 배경

### 1. Vod 서비스

VoD 서비스란 'Video On Demand' 주문형 비디오 서비스 또는 요구 형 비디오 서비스라고도 불리는데 통신망으로 연결된 단말기를 통해 원하는 콘텐츠를 언제든지 실시간으로 받아볼 수 있는 영상 서비스를 말한다. 현재의 단일 방향 TV를 한 차원 더 발전시킨 대화형 TV (Interactive TV)의 하나로서 일반 전화를 통해 연화와 같은 각종 멀티미디어 정보를 실시간적으로 제공 하는 서비스이다. 여기서 On Demand란 '사용자가 다양한 멀티미디어 정보 중에서 원하는 내용(Contents)을 원하는 시간(Realtime)에 선택적으로 활용 가능 하다'라는 말이며 개인이 소유하고 있는 VOD 단말기를 이용하여 가입자 선로나 광대역 통신망을 통해 VOD 서버로 접속하여 원하는 콘텐츠를 수신함으로써 쉽게 이를 이용할 수 있다. TV나 케이블TV(유선방송)처럼 일방적으로 송출되는 콘텐츠만을 보는 것이 아니라 원하는 때에 원하는 콘텐츠를 선택할 수 있다는 점에서 대화형 TV의 원형으로도 볼 수 있다. 또한 프로그램을 시청하는 도중에 일시 정지시키거나 느린 속도로, 혹은 반복해서 볼 수도 있다. VOD 시스템을 갖추려면 방대한 멀티미디어 데이터베이스에 접속 가능한 VOD 서버와 고속의 전송률을 보장하는 광대역 통신망이 필요하다.

#### 1) VOD 시스템의 등장배경

통신망 기술의 광대역화 및 고속화의 진행과 멀티미디어 정보 요구의 확대에 따른 전송 압축 기술 및 멀티미디어 처리 기반 기술, 디지털 영상처리 기술 등 VOD 서비스 제반 기술이 발전이 이루어지고 있다. 이로 인하여 사용자의 요구에 따른 동영상을 서비스하는 방송과 통신의 통합된 형태로 발전하게 되었다.

VOD 서비스 개발은 미국 벨 아틀란틱스사가 양방향 대화형 전송 프로토콜인

ADSL의 개발에 의해 매우 활발해졌다. ADSL은 기존의 전화망에서 사용자가 요구 데이터와 비디오 서비스를 서로 주고받을 수 있는 전송기술로서 구체적이고도 현실적인 VOD 실행의 원동력이 되었다. 또한 국제 표준화 기구인 ISO/IEC JTC1/SC29/WG11에서 디지털 동영상 압축 및 복원 부호화 기술인 MPEG-1과 MPEG-2의 표준이 완성되어 이를 이용한 가입자 장치로의 응용이 가능함에 따라 VOD 시스템 연구에 활력이 되었다.

그 결과 VOD 시스템에 따른 장치 및 서비스의 국제 표준이 절실히 요구되어 왔으며, 1994년 6월 미국 산호세에서 VOD 관련 표준화 그룹인 DAVIC(Digital Audio Visual Council)이 결성되었다. DAVIC의 목적은 인터페이스와 프로토콜, 디지털 Audio-visual 응용 프로그램과 서비스 구조에 대한 규격의 공식적 표준화를 실현하는데 있다.

## 2) VOD 시스템의 정의

VOD 시스템은 대화형 맞춤형영상정보서비스 또는 주문형 비디오 조회 시스템이라고도 한다. 기존 공중망 방송이나 케이블 TV에서 프로그램을 일방적으로 수신하는 것이 아니라 가입자의 요구에 따라 원하는 콘텐츠를 선택하여 수신 받는 영상 서비스이다. VOD 시스템은 비디오 서버에 영상을 저장하여 다수의 사용자가 시간적 제약 없이 자신이 원하는 영상을 보고 싶을 때 볼 수 있는 실시간 시스템이며, 쌍방향 서비스를 말한다. 여기서 On Demand란 사용자가 다양한 멀티미디어 정보 중에서 원하는 내용(contents)을 원하는 시간(real time)에 선택적으로 활용 가능한 것을 뜻한다. 통신망을 통하여 영상정보를 주문한다는 것 이외에는 일반 비디오 기기와 마찬가지로 방법으로 재생(play) 및 정지(stop), 멈춤(pause), 빨리 감기(fast forward), 빨리 되감기(fast rewind), 임의의 위치에서의 재생 기능을 모두 제공한다.

## 3) VOD 시스템의 특징

기존의 방송은 아날로그 신호를 일방적으로 송신하였으나 VOD는 디지털 신호

로서 압축이 가능하며 사용자의 요구에 따른 영상을 송신한다. VOD는 압축된 신호를 복원하기 위한 디코더 장비를 필요로 하며, 기존 방송망 방식과 다른 유선 방송망을 이용한다. 또한 기존의 수치 및 텍스트 데이터와 비교하여 VOD 서비스 데이터들이 갖는 일반적인 특성으로는 대용량성과 비정형성, 시간성을 들 수 있다.

VOD 데이터는 기존의 문자와 수치 데이터 저장과는 비교가 안 될 정도로 큰 용량의 기억장소가 필요하다. 이러한 이유로 멀티미디어 데이터를 저장할 때에는 데이터 압축 및 복원 기법의 사용이 필수적이다. VOD 서비스 데이터는 특별한 형식이 없는 비정형의 구조를 갖는다. 예를 들면, 이미지 데이터의 경우 일정한 구조를 갖지 않는 연속적인 스트림(stream)이라 볼 수 있다.

VOD 데이터는 시간 제약성을 갖는다. 오디오와 비디오, 애니메이션 데이터는 본질적으로 시간에 의존하는 특성을 가지며 여러 가지 다양한 포맷의 데이터가 복합적으로 시간적인 순서에 따라 표현(present)되는 동기화(synchronization) 특성이 존재한다. 따라서 연관된 정보를 포함하는 데이터간에 정해진 시간과 순서에 따라 표현되어야만 사용자에게 정확한 정보를 전달할 수 있다. 예를 들면, 동영상과 오디오 데이터가 결합된 데이터의 경우 동기화 정보를 이용하여 정해진 표현 순서대로 표현해야 의미 있는 정보를 전달할 수 있다.

일반적인 VOD 서비스가 갖는 특징은 다음과 같다.

- 서비스 시간이 길다. 일반적인 동영상 서비스는 90~120분 동안 지속되기 때문에 사용자에게 대한 긴 세션을 유지해야 한다.
- 높은 대역폭이 요구된다. MPEG-1 형식의 스트림을 1.5Mbps의 서버 디스크 I/O와 네트워크 대역폭이 요구된다. MPEG-2인 경우 3~10 Mbps가 요구된다.
- VCR과 같은 상호작용을 지원한다. 사용자는 VOD 서비스에서 VCR에서 지원하는 재생 기능뿐만 아니라 빨리 감기, 되감기, 일시정지와 같은 기능을 요구한다.
- QoS(Quality of Service)에 민감하다. VOD 서비스 제공자들은 사용자에게 서비스를 제공할 때 서비스 대기시간 및 상호 작용성, 비디오 재생 효과와 같은

기능 등을 고려해야 한다.

## 2. VoD 데이터 전송 방식

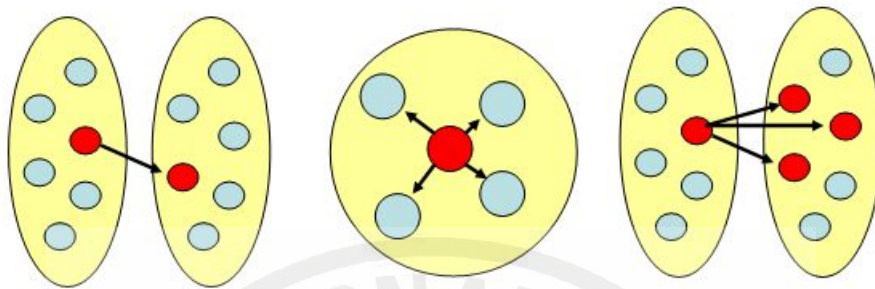


그림 1. Unicast, Broadcast, Multicast

### 1) 유니캐스트(Unicast)

유니캐스트 전송 방식은 1:1 통식방식으로 하나의 송신자가 다른 하나의 수신자로 데이터를 전송하는 방식이다. 그룹 통신을 하기 위해서 다중 수신자들에게 동일한 데이터를 전송하고자 할 경우 전송하고자 하는 데이터 패킷을 다수의 수신자에게 각각 여러 번 전송해야 하며, 이러한 동일한 패킷의 중복전송으로 인해 네트워크 효율이 저하된다. 또한 수신자 수가 증가할 경우 이러한 문제점은 더 커지게 된다.

### 2) 브로드캐스트(Broadcast)

브로드캐스트 전송 방식은 1:N 통신 방식으로 하나의 송신자가 다수의 수신자로 데이터를 전송하는 방식이다. 예를 들어 라디오나 텔레비전과 같은 공중파방송을 생각하면 쉽다. (LanCard)라고 하는 이더넷(Ethernet)은 48bit로 된 고유의 MAC Address라고 하는 하드웨어주소를 가지고 있으며, 자신에게 도착하는 패킷중 이 주소부분이 자신의 MAC Address로된 패킷과 모두 1로 셋팅된 것을 받아들여지게 된다. 브로드캐스트 통신방식을 사용하여 통신을 하게 되면 불필요한 네트워크 트래픽을 유발하게 될 수도 있다.



### 3) 멀티캐스트(Multicast)

멀티캐스트 역시 N:N 통신 방식으로 다수의 송신자가 다수의 수신자로 데이터를 전송하는 방식이다. 브로드캐스트가 통신이 가능한 모든 곳으로 패킷을 보낸다면 멀티캐스트는 수신가능한 수신지에서만 패킷을 받아들이게 되는 통신방식이다. 즉, 멀티캐스트는 하나이상의 송신지에서 다수의(멀티캐스트로 전송되는 패킷을 받아들이도록 Lan카드를 설정했다면)수신지로 패킷을 보내게 되는 통신방식이라고 할 수 있다. 브로드캐스트보다는 네트워크 트래픽을 최소화하여 네트워크 트래픽부하를 줄일 수 있다는 점에서 인터넷 방송등에 많이 응용이 되고 있다. 하지만 제어신호의 과다 발생과 라우터에서 경로 계산의 복잡도 증가로 인한 문제, 완벽한 IP 멀티캐스트를 구현하기 위해서는 인터넷상에 존재하는 대부분의 라우터들이 멀티캐스트 기능을 지원하도록 수정 · 교체해야만 하는 등 많은 문제점 가지고 있어 현재까지도 실제 상업망에서 제한되어 있다.

#### 3. IP 기반 멀티캐스트 기법

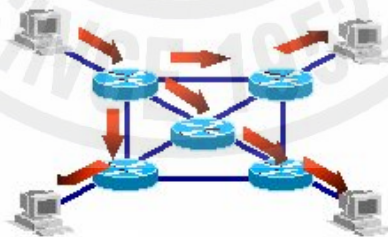


그림 2. IP Multicast

현재 인터넷 환경에서 다중의 사용자들에게 중복된 패킷을 줄이기 위해 가장 많이 사용하는 전달 방법이 Multicast이다. 그림 2 에 전형적인 IP Multicast에 대해서 나타내고 있다. 즉 같은 데이터를 요구하는 사용자가 많은 경우 그 데이터를 하나하나

사용자에게 보내주는 것이 아니라 한번의 데이터 전송으로 다중 사용자가 받아 볼 수 있게 한다. Multicast 프로토콜을 사용하여 같은 데이터의 재전송을 미연에 방지함으로써 대역폭의 낭비를 줄일 수 있고, Network의 혼잡을 줄일 수 있다. 현재 Multicast는 인터넷 전화, 화상회의, VoD 서비스 그리고 network 온라인 게임등 많은 응용 분야에 사용되고 있으며. 이를 지원하기 위한 라우팅 기술들에 대한 연구가 활발히 이루어지고 있다.

한번의 전송으로 다중 사용자가 데이터를 수신하기 위해서 Multicast는 Group을 이용한다. 같은 데이터를 원하는 사용자들을 하나의 Group으로 묶어서 관리를 하게 되고, 이에 필요한 것이 IP중에서 IGMP (Internet Group Management Protocol)이다. Multicast Group을 알게 된 라우터는 자신의 Subnet에 대한 사용자들을 관리하게 되고 이때 이 라우터는 주기적으로 자신의 Subnet에 해당 Group에 대한 IGMP\_Query를 보내게 되고 이를 받은 호스트는 계속 그 Multicast Group을 유지하고 싶으면 IGMP\_Report로 응답을 하고 탈퇴하고 싶으면 응답을 하지 않음으로써 Multicast Group을 유지하게 된다. 또한 새로 가입 하고자 하는 호스트는 라우터에서의 주기적인 IGMP\_Query를 기다릴 필요 없이 바로 IGMP\_Report를 보냄으로서 Group에 Join하게 된다. 데이터를 동시에 다수의 상대에게 송신하는 멀티캐스트, 멀티캐스트를 실행하기 위해서는 클래스 D라고 하는 IP 주소 체계를 사용한다. 클래스 D의 주소 체계에서는 32비트 중에서 최초의 4비트로 멀티캐스트라는 것을 식별하고, 나머지 28비트로 특정의 멀티캐스트 집단을 지정한다. IP 기반의 멀티캐스트 기법은 동일한 요청 비디오 데이터에 대하여 인접한 시간에 발생한 요청들을 하나로 묶어 서버 채널을 할당 하는 기법으로 서버의 자원과 대역폭을 공유 한다.

IP 기반의 멀티 캐스트 기법은 제어신호의 과다 발생과 라우터에서 경로 계산의 복잡도 증가로 인한 문제, 완벽한 IP 멀티캐스트를 구현하기 위해서는 인터넷상에 존재하는 대부분의 라우터들이 멀티캐스트 기능을 지원하도록 수정 · 교체해야만 하는 등 많은 문제점을 안고 있어 현재까지도 실제 상업망에서 제한되어 있다

다음은 VoD 서비스에서 가장 많이 사용되고 있는 기법 들이다.

### 1) Batching

서버가 짧은 시간대에 발생하는 요청들에 대해서 하나의 Batch로 묶어 서버채널을 할당하여 서비스함으로써 서버의 자원을 줄이는 정적 멀티캐스트 기법, 하지만 일정 batch 시간보다 먼저 도착한 요청이나 늦게 도착한 요청에 대해서는 다른 batch로 할당하기 때문에 유동적이지 못하다.

### 2) Adaptive Piggy backing

Batching을 확장한 유동적인 멀티캐스트 기법으로 일정한 시간 동안의 요청에 대해서 Batch 로 묶어 서비스하는 것은 Batching과 똑같다. 하지만 늦게 도착하는 요청에 대해서 기존에 Batch로 함께 묶어 서비스 하기 때문에 Batching 보다 유동적이다.

### 3) Pyramid Broadcasting

사용자가 서버에 비디오 스트림에 관한 요청을 하지 않고 서버는 같은 비디오를 일정시간마다 주기적으로 방송하는 기법이다.

## 4. 응용 계층 멀티 캐스트

응용계층 멀티 캐스트 기법은 중단 시스템에서 P2P 기법을 이용하여 멀티캐스트 트리를 구성하고 사용자들에게 데이터를 전송 하는 방식이다.

### 1) Overlay Multicast

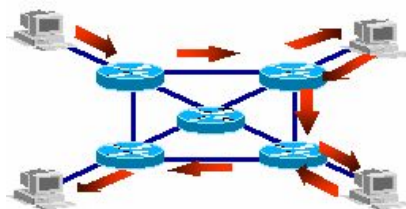


그림 3. Overlay Multicast

Overlay Multicast의 기본 아이디어는 그림 3과 같다. Overlay Multicast에서는 멀티캐스트가 도입되지 않은 기존의 인터넷 망에서는 유니캐스트 전송을 사용하고, 각 지역에 중계기 서버를 설치하여 원격 송신자의 멀티캐스트 트래픽을 저장 혹은 caching 한 후에 해당 지역 사용자에게 멀티 캐스트 혹은 유니 캐스트로 중계하는 기술이다. 기존에 IP 멀티캐스트는 네트워크상의 라우터에서 패킷 복제 및 중계가 수행되는 반면에 Overlay Multicast에서는 호스트 단말에서 패킷 복제 및 중계가 이루어진다.

### 1-1) 오버레이 멀티캐스트 기술

지금까지 제안된 대표적인 오버레이 멀티캐스트 기술로는 ALM, Narada, Scatter Cast, TBCP, NICE, Yoid등이 있다. 그림 4는 제안된 기술들을 중앙 집중형(Centralized)과 분산형(Distributed)으로 분류하여 도식화한 것이다.

각각의 기법들을 특정 응용 서비스를 목적으로 개발 되었다. 즉 Narada나 ALMI 등은 오디오/비디오 컨퍼런싱과 같은 실시간 멀티미디어 서비스를 위하여 개발 되었으며, RMX와 같이 신뢰적 전송이 요구되는 서비스 제공을 위하여 개발된 기술이 있다. 현재까지는 표준화된 인터페이스가 정의되어 있지 않아서 응용 서비스의 요구사항에 따라 해당 기법을 선택해서 사용해야 한다는 어려움이 있다.

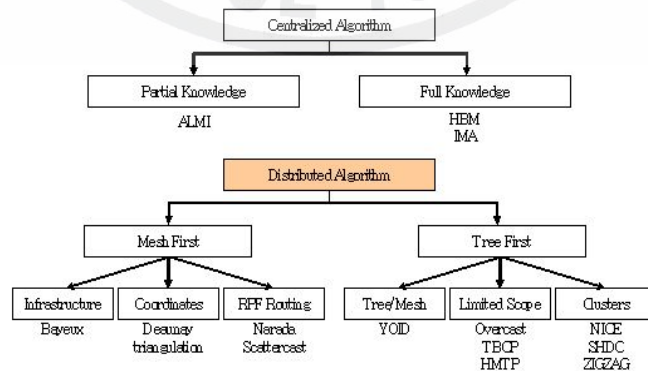


그림 4. 오버레이 멀티캐스트 기술 분류

## 2) Patching 기법

VoD서비스 오버레이 멀티캐스트 기법을 이용하여 제공하기 위한 연구들은, 기본적인 ALM(Application Layer Multicast)에 대한 연구가 지연과 지터를 고려한 효율적인 트리구성에 주로 집중되는 데 반해, 패칭기법은 다른 측면에서의 접근 방식들이 보인다. 기본적으로 구성된 네트워크 장비들을 교체하지 않고 VoD 서비스를 제공하기 위해서는 오버레이 멀티캐스트 기법과 패칭기법의 조합이 필요하다. 멀티캐스트 에서 새로운 클라이언트가 접속 하였을 때, 실시간 라이브 방송에서는 오직 접속한 시간의 미디어 스트림만이 필요한 반면 On-Demand 서비스에서는 미디어의 시작부분부터 접속한 시간까지의 모든 스트림이 필요하다. 따라서 미디어의 시작 부분을 전송해줄 수 있는 특별한 기법이 필요하며, 이를 패칭기법 이라고 한다[4]. 오버레이 멀티캐스트는 라우터의 구현, 혼잡 제어, 신뢰성 있는 전송 등의 문제점을 지니는 IP 멀티캐스트의 대안으로 제안된 기법이며 전형적인 패칭기법 또한 이러한 IP 멀티캐스트 상에서 연구되어온 기법이다. 이러한 패칭기법을 오버레이 멀티캐스트 상에 적용하여 제안된 대표적인 관련 연구들을 살펴보면 다음과 같다.

### 2-1) Coopnet

Coopnet[5]은 실시간 비디오 스트리밍을 위한 시스템으로서 미디어 콘텐츠를 여러 부분으로 나누어 배분시키는 MDC(Multiple Description Coding)을 적용하여 peer의 장애가 발생하더라도 데이터의 손실을 최소화한다. 이는 VoD를 위한 기능은 지원하지 않지만, 재생 점이 다른 사용자들을 관리하는 기법은 제공하지 않는다.

### 2-2) NICE, Zigzag

NICE[6]와 Zigzag[7]는 계층 구조의 클러스터 기반 기법으로서 모든 멤버들을 확장성 있는 계층형 구조로 형성하고, 동적 특성들을 소화할 뿐만 아니라 멀티캐

스트 트리의 제어트래픽을 감소시킬 수 있으나 네트워크 유동적인 환경에서 개념적인 대역폭을 다루는 이러한 기법들이 현 인터넷에서 적용하기엔 무리가 따른다. 특히 VoD 스트리밍 서비스에는 이런 구조로는 해결될 수 없다.

### 2-3) P2Cast

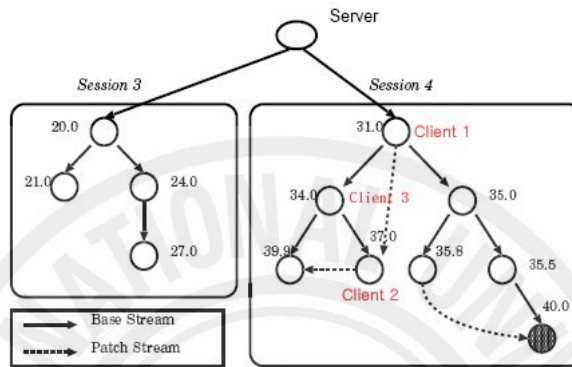


그림 5. P2Cast 시스템 구조

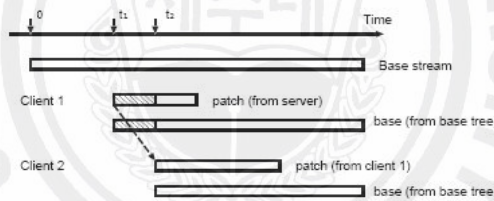


그림 6. Client 1, 2의 스트림 비교

P2Cast는 P2P네트워크와 IP 멀티캐스트의 패칭 채널, 그리고 ALM(Application Layer Multicast)을 조합하여 고안된 기법으로 그림 5 에서 P2Cast의 동작 구조를 볼 수 있다. 서비스를 제공 받기위한 클라이언트들은 서버로 접속 하여 서버로부터 Video Stream 서비스를 받는 것이 아니라 몇몇 클라이언트만이 서버로부터 직접 Video Stream을 받고 나머지 클라이언트들은 서버가 아닌 기존에 서비스 받고 있는 클라이언트들에게 Video Stream을 전송 받는 방식으로 그림 5에 보는 바와 같이 세션 트리를 구성하여 서비스 받게 된다. 이에 Video Stream의 시작 부분을 다른 클라이언트에서 받고 동시에 base Stream을 받기 때문에 서버의 부하를 줄일 수 있고 보다 많은 수의 Client에게 서비스를 제공

할 수 있다는 장점을 가지고 있다.

새로운 클라이언트(Client 1)가 Video Stream 서비스를 받기 위해 서버로 접속하게 되면 서버는 먼저 기존의 서비스되고 있는 조인 가능한 세션이 존재하는지를 검사하게 된다. 기존 세션이 존재하지 않는다면 새로운 세션을 만들고 조인한 클라이언트는 세션 트리의 최상의 클라이언트가 되어 서버로부터 Video Stream을 제공 받게 된다. 조인 가능한 세션이 존재할 때에는 기존에 Video 서비스가 되고 있는 트리에서 맨 하위 노드에 조인하게 된다. 조인한 클라이언트(Client2)는 상위 클라이언트(Client3)에서 Base Stream을 받게 되고 미디어의 첫 부분을 VoD 서버 또는 미디어의 시작 부분을 캐쉬에 저장하고 있는 다른 상위 클라이언트(Client1)로부터 Patch stream을 수신 받게 된다.

그림 6은 기존에 서비스 되고 있는 Client 1과 새로 조인한 Client 2의 수신 스트림 양을 보여 주고 있다. 보는 바와 같이 Base Stream으로부터 다른 클라이언트들과 같은 일정량의 Stream수신 하면서 Client 1 으로부터 Patch Stream을 받아 많은 양의 Video Stream을 수신 받기 때문에 VoD 서비스의 지연과 지터를 줄일 수 있다. 하지만 현재 요구 고품질·대용량의 멀티미디어 서비스를 제공하기에는 충분한 QoS를 보장 하지 않는다.

또한 P2Cast에서는 서비스 받고 있는 중간 노드의 장애·이탈시 Best-Bit(BF) algorithm, BF-dely algorithm, BF-delay-approx algorithm을 이용하여 해결하고 있다. 이는 장애가 발생 할 때 마다 서버가 개입하여 해결하는 방식을 사용하기 때문에 서버의 병목현상이 발생 될 수 있고 VoD 서비스를 받는 중간 클라이언트들이 이탈이 활발한 경우에는 상위 클라이언트가 다수의 클라이언트에게 스트림을 전송해 주어야만 서비스가 가능하다는 단점을 가지고 있다.

본 논문에서는 P2Cast는 세션내의 클라이언트들은 수신된 스트림을 모두 캐쉬에 저장하고 있지만 본 논문에서 제안된 기법에서는 캐쉬가 아닌 특정크기의 공유버퍼를 사용함으로써 여러 클라이언트들에게서 Patch Stream 수신 받아 단기간에 많은 양의 Video Stream을 저장 할 수 있어 지터와 지연을 해결할 수 있고 나아가 고품질·대용량의 멀티미디어 서비스도 가능 하다. 또한 중간 노드의 장애·이탈시 현재 서비스 되고 있는 클라이언트 중에서 필요한 Video Stream 블록을 가지고 있는 가장 네트워크 자원이 풍부한 클라이언트를 다시 할당하기 때

문에 유동성 있는 세션 트리 구성이 가능하다.

#### 2-4) P2VoD

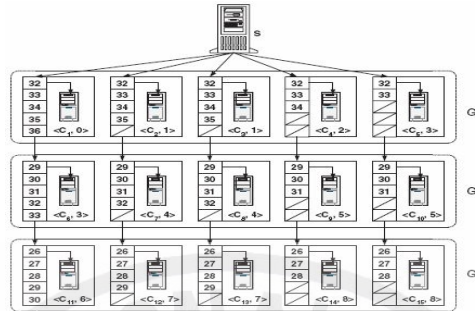


그림 7. P2VoD 시스템 구조

P2VoD는 IP 주소의 부족, 보안 및 보급상의 문제로 실현 불가능한 IP 멀티캐스트에 대한 대안으로 제안된 오버레이 멀티캐스트를 이용하여 VoD서비스를 수행하는 P2Cast 기반의 또 다른 기법이다.

P2Cast는 클라이언트들이 세션 트리를 구성을 하여 Video Stream 서비스를 제공 받지만 P2VoD에서는 클라이언트의 그룹이 세션 트리를 구성하여 Video Stream 서비스를 제공 받는다. 그림 7은 P2VoD 시스템 구조를 나타내고 있다.

서버는 클라이언트의 요청에 대해서 비디오 스트림을 시간단위의 데이터로 블록화 하여 서비스를 수행한다. P2VoD 세션에 가입한 모든 클라이언트들은 이러한 데이터 블록 중 가장 최근의 블록을 저장 하기위한 특정 크기의 버퍼를 구성 하게 된다. 같은 데이터 블록을 보유하는 클라이언트들을 동일 차수로 그룹화 하여, 새로운 요청이 있는 클라이언트는 자신보다 상위 차수에 있는 모든 클라이언트들을 검사하고, 요청한 스트림 블록을 보유하고 있는 클라이언트에게서 서비스를 받게 된다. P2VoD에서는 P2Cast에서 중간 노드의 장애 • 이탈시 서버가 직접 관여하여 해결 하였던 문제를 그룹화 메카니즘을 통해 장애가 발생 했을 때 빠른 복구가 가능하도록 하였다. 하지만 그룹화 메카니즘은 장애 발생시 빠른 복구에는 효과적일 수 있지만 그룹화 메커니즘의 구현이 복잡해질 수 있는 단점이 있다.



### Ⅲ. Peer-to-Peer 구조 VoD 서비스

#### 1. 시스템 구조

본 논문에서 제안하는 시스템의 목적은 현재 요구 되고 있는 고품질 · 대용량의 멀티미디어 서비스를 제공 할 수 있는 QoS가 보장된 VoD 서비스 시스템 구축을 목적으로 한다.

제안시스템의 기반은 VoD 서비스를 제공 받는 사용자 단말기(Peer)들은 성능이 향상됨에 따라 호스트로서의 역할을 충분히 수행할 수 있음을 감안하였다. 이에 VoD 서비스를 비롯한 멀티미디어 데이터들의 실시간성이라는 특성을 Peer-to-Peer 서비스 특성에 더욱 주목한다면 이를 해결 할 수 있을 것이다. 따라서 Peer의 네트워크 자원을 최대한 이용 할 수 있을 것이고 Peer의 현재 플레이 되고 있는 스트림 이후의 스트림을 VoD 서버가 아닌 중간 Peer 들로부터 단기간 안에 최대한 많은 양의 스트림을 미리 전송 받을 수 있어 시간의 제약으로 인한 지연과 지터를 해결 할 수 있을 것이다. 결과적으로 사용자에게 고품질이 멀티미디어 VoD 서비스를 제공할 수 있는 시스템을 사용자에게 제공할 수 있을 것이다. 기본적으로 Peer가 호스트로서의 기능을 수행하기 위해서는 각각의 Peer들은 상당량의 자원을 가지고 있기 때문에 자신이 받은 스트림 데이터를 바로 폐기하지 않아도 충분한 자원을 사용할 수 있는 여지가 있다는 전제를 만족해야 하며, 이러한 자원에 저장된 콘텐츠 데이터는 다른 피어들에게 서비스 될 수 있어야 한다. 이 전제 조건을 만족했을 때 Peer는 중간 호스트로서의 기능을 수행할 수 있으며 위 전제 조건을 만족하지 않는 Peer들은 단지 VoD서비스만을 제공 받는 클라이언트로 분류 한다.

따라서 VoD 서비스를 제공 받기를 원하는 새로운 Peer는 VoD 서버 또는 중간 Peer로부터 비디오 스트림 앞부분을 전송 받아 충분한 스트림을 확보하게 되면 비디오 플레이하는 동안에 현재 플레이되고 있는 스트림 이후의 데이터를 서

버 또는 서버의 조정자가 넘겨준 정보를 참조 하여 현재 서비스 되고 있는 상위 Peer들로부터 현재 자신의 가진 네트워크 대역폭을 최대한 이용하여 스트림 데이터를 단기간 안에 최대한 많은 양의 스트림을 미리 전송 받아 플레이함으로서 지터와 지연을 없애수 있을 것이다.

본 논문에서 제안하는 Peer-to-Peer 구조 VoD 시스템의 전체적인 구조는 그림 8과 같다

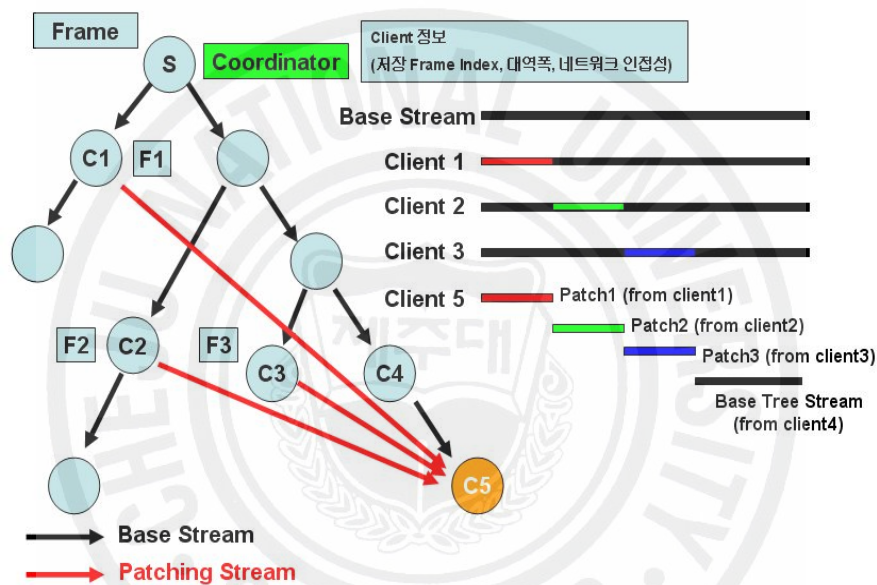


그림 8. P2P Vod 시스템 전체 구조

먼저 시스템 구성원들을 간략하게 살펴보면 기본적인 스트림 서비스를 제공하는 중앙 VoD 서버와 서버에 접속 하는 Peer들을 관리 하는 조정자 (Coordinator), VoD 서비스를 제공 받기 위한 Peer로 나눌 수 있다. 또한 Peer는 두 가지의 형태로 분류 할 수 있는데 상당량의 자원을 가지고 있어 자신이 받은 스트림 데이터를 바로 폐기하지 않아도 저장가능 하며 다른 요청 Peer들에 대해서도 스트림 데이터를 제공할 수 있는 충분한 자원을 가지고 있는 중간 호스트 Peer와 충분한 자원을 보유 하고 있지 않고 단지 스트림 서비스만을 받기를 원하는 일반 Peer로 나눌 수 있다. 이 분류 기준은 나중에 다시 설명을 하겠지만

Peer가 서버 또는 중간 Peer들로 서비스를 제공 받을 때 현재 자신의 자원 현황을 조정자로 보고 할 때 조정자가 이를 분류를 한다.

각각의 구성원들의 역할을 살펴보면 다음과 같다.

- 중앙 VoD 서버: 일반적인 VoD 서버로서의 역할을 수행한다. 비디오의 전체 프레임을 저장 하고 있으며 각 비디오 스트림은 블록 단위로 나뉘 index를 붙여 관리하게 되는데 Peer의 요청에 따라 해당 비디오 스트림 블록을 Peer에게 제공 하게 된다.
- 조정자(coordinator) : 중앙 VoD 서버에 위치하고 있으며 서버에 접속하는 Peer들에 대한 관리 역할을 수행한다. 서버에 접속한 Peer들에 대하여 새로운 VoD 서비스 세션을 생성하고 할당하며, 기존에 생성된 세션에 대한 조인, 중간 Peer의 이탈에 따른 Peer의 스트림 요청에 대하여 어느 피어로부터 비디오 스트림 서비스를 받을지에 대한 정보 제공 역할을 수행한다. 조정자는 각 세션에 대한 정보와 세션에서 제공 받고 있는 Peer들에 대한 정보(어떤 스트림 데이터 수신단위가 어떤 피어에 있는지에 대한 정보)를 가지고 있다. 이 정보는 Peer가 스트림 블록 단위를 수신을 할 때 마다 Peer가 조정자에게 보고를 하는데 이때 업데이트가 이루어진다.
- 피어: VoD 서비스를 제공 받기를 원하는 클라이언트가 될 수 있으며, 동시에 다른 Peer에게 비디오를 제공하는 중간 호스트 Peer의 역할도 수행 할 수 있다. Peer는 자신의 네트워크 자원에 따라서 중간 호스트 Peer와 단지 스트림 서비스만을 받는 Peer로 분류 할 수 있다. 각 Peer는 서비스 받는 중에 특정 블록단위를 수신하게 된다면 자신의 네트워크 자원정보 및 저장 데이터에 대한 정보를 조정자에게 제공한다.

Peer-to-Peer 구조 VoD 시스템의 전체적인 동작 구성은 각 Peer들은 스트림 데이터를 연속적으로만 받는 것을 가정하는 것이 아니라, 스트림 데이터를 일정

부분씩 끊어서, 그림 8에서 보듯이 이를 여러 피어들로부터 동시에 받으려 한다.

VoD 서버는 비디오 데이터를 블록단위로 나누어 저장하고 있으며 이를 Client들에게 제공한다. 조정자는 현재 서비스 되고 있는 세션의 정보와 Client들에 대한 정보를 가지고 있다.

먼저 VoD 서비스를 제공 받기를 원하는 Peer(Client 5)는 VoD 서버로 접속을 하게 된다. VoD서버의 조정자는 새로운 Client 5가 조인 가능한 VoD세션이 존재하는지를 조사한다. 기존 세션에 조인 가능하다면 서비스 되고 있는 Peer들에 대한 정보를 자신의 가지고 있는 Client 정보 (저장 Frame index 정보, 대역폭, 네트워크 인접성)를 참고 하여 Client 5에게 제공 하게 된다. Client 5는 이를 이용하여 기존 세션에 조인하게 되고 그림 8과 같이 기존 세션 트리 하단에 조인하여 Client 1 으로 부터는 Frame 1을 제공 받고 Client 2로 부터는 Frame 2를, Client 3로부터는 Frame3, Client 4로 부터는 정규 스트림을 제공 받는다. 스트림을 제공 받는 순서는 Frame1을 최우선으로 제공 받아 비디오 스트림을 플레이 가능 하도록 하고 그 다음 다른 중간 Client 들로부터 나머지 스트림을 동시에 제공 받는다.

비디오 스트림 서비스를 제공 받는 Client5는 자신의 특정 블록을 모두 수신하게 된다면 이를 VoD 서버의 조정자에게 현재 자신이 수신한 블록의 index와 현재 자신의 네트워크 자원의 정보를 알려 주게 된다. VoD서버의 조정자는 이를 이용하여 Client5 가 중간 호스트 Peer로서 역할을 수행 할 수 있는지를 판단을 하고 자신의 가지고 있는 Client정보를 업데이트함으로써 가장 최신의 Client 정보를 유지하게 된다. 이 작업은 Client가 특정 블록을 모두 수신 받을 때 마다 이루어진다. 기존 VoD 세션이 존재하지 않는다면 새로운 세션을 생성하고 VoD 서버로부터 스트림 데이터를 전송 받는다.

Client가 스트림 서비스를 종료 할 때는 조정자에게 종료를 알리고 조정자는 자신의 저장 Client정보에서 해당 Client 정보를 제거한다. Client 가 스트림 서비스를 제공 받는 도중에 중간 피어의 장애·이탈시 Client는 해당 피어의 장애를 조정자에게 알리고 조정자로부터 새로운 중간 호스트 Peer의 정보를 얻어 장애 스트림 블록을 중간 피어 또는 VoD 서버로부터 전송 받게 된다. 장애·이탈 Client 역시 조정자의 저장 Client 정보에서 제거된다.

이와 같이 VoD 서비스를 받으려는 피어는 현재 비디오 스트림을 플레이 하는 동시에 그 이후의 여러 스트림 데이터를 동시에 다른 피어들로 받을 수 있어서 자신이 현재 디스플레이하는 것보다 훨씬 앞서있는 데이터를 확보할 수 있다. 결과 적으로 하나의 프레임을 실시간으로 받으면서 이를 디스플레이 하는 동시에 이후의 프레임들을 각각 다른 피어들로부터 전송 받아 미리 확보하기 때문에 하나의 프레임을 디스플레이 하는 시간 동안에 그 이후 여러 프레임을 디스플레이 하기 위한 데이터를 자기 자신에게 축적하게 되어 실시간으로 스트리밍 데이터를 받아야 하는 제약으로부터 벗어날 수 있다. 또한 Peer의 네트워크 자원을 최대한 이용할 수 있기 때문에 고용량의 멀티미디어 데이터도 단기간에 최대한 많은 양이 데이터를 전송 받을 수 있어 고품질의 멀티미디어 VoD 서비스도 가능하게 된다.

## 2. Peer-to-Peer 구조 VoD 서비스 기본동작

오버레이 멀티캐스트 기법을 이용하여 스트림 데이터를 서비스 받기 위해서는 기본적으로 세션의 생성을 위해 일정 시간에 조인한 클라이언트들을 모아 그룹으로 관리하기 위한 Batching과 기존 세션에 조인하기 위한 Piggybacking, 다른 여러 클라이언트들에게 스트림 서비스를 받기위한 Patching[4] 기법을 필요로 한다. 하나의 클라이언트 피어가 스트림 데이터를 받기 위해서는 다음과 같은 순서로 접속하게 된다.

- 서버의 조정자에 접속
- 새로운 세션의 생성 및 기존 세션으로의 조인
- 새로운 세션이 생성 되었다면 서버로 접속해서 스트림 서비스의 수신
- 해당 세션에 조인이 되었다면 기존 세션에서의 상위 피어로부터 기본 스트림을 받고 다른 여러 상위 피어들로부터 패치 스트림을 받게 된다.
- 중간 피어의 장애 시 서버에 접속하여 다른 중간 피어의 정보를 요청 하고 그 정보를 바탕으로 스트림 데이터를 수신하거나 재요청 중간 피어의 장애

시 해당 스트림 데이터를 서버로부터 수신 한다.

이를 위해서 클라이언트인 피어는 일정 시간 내에 있는 VoD 세션에 할당되고, 시작 시점에 최초의 비디오스트림 데이터는 서버에서 받으며, 동시에 그 이후의 비디오 스트림 데이터는 조정자에 의해 배당된 피어들로부터 받는다. 피어는 자신이 비디오를 받는 클라이언트이면서 동시에 다른 클라이언트에게 비디오 데이터를 주는 비디오 서버의 역할을 같이하기 때문에, 자신이 비디오 저장을 위해 가지는 자원의 양과 자신의 네트워크 인터페이스의 통신용량을 최초 접속 시 조정자에게 등록하는 절차를 거친다. 또한 이후에 지정된 비디오 스트림 블록을 모두 수신 하였다면 그 해당 비디오 스트림 번호와 자신의 현재 네트워크 자원의 정보를 서버의 조정자에게 보낸다. 따라서 조정자는 현재 서비스 되고 있는 피어의 자원 정보를 실시간으로 저장 할 수 있어 다른 클라이언트인 피어가 포함될 때 가장 작은 용량을 가진 피어를 기준으로 해당 세션의 허용 시간범위를 결정 할 수 있게 된다.

#### 1) 비디오 스트림의 블록단위

##### 비디오 전체 프레임

Index 1	Index 2	Index 3	Index 4	Index 5
---------	---------	---------	---------	---------

그림 9 각 비디오 프레임 관리 블록 단위

본 논문에서는 클라이언트 피어가 다른 다수의 피어들로부터 동시에 스트림을 수신하는 메커니즘을 제안하고 있다. 이를 위해서 스트림은 블록단위로 취급되어 순서번호를 붙이게 되며, 조정자는 어떤 번호의 블록이 어떤 피어에 저장되어 서비스될 수 있는지에 대한 정보를 가지고 있다. 다만 본 논문에서 피어가 충분히 비디오 데이터의 소스로서 역할하기위해서 근래 하드웨어의 발전에 힘입어 충분한 저장 공간을 가지고 있다고 가정한다. 이는 클라이언트 역할을 하는 피어가 자신의 디스플레이 시점이 지나가도 바로 데이터를 버리는 것이 아니라 자신의

저장용량이 허용하는 한 과거의 스트림 데이터도 충분히 가지고 있으며, 이 데이터가 다른 피어들에게 제공되는 비디오데이터로서 사용될 수 있도록 하려는 것이다.

## 2) 클라이언트의 가입

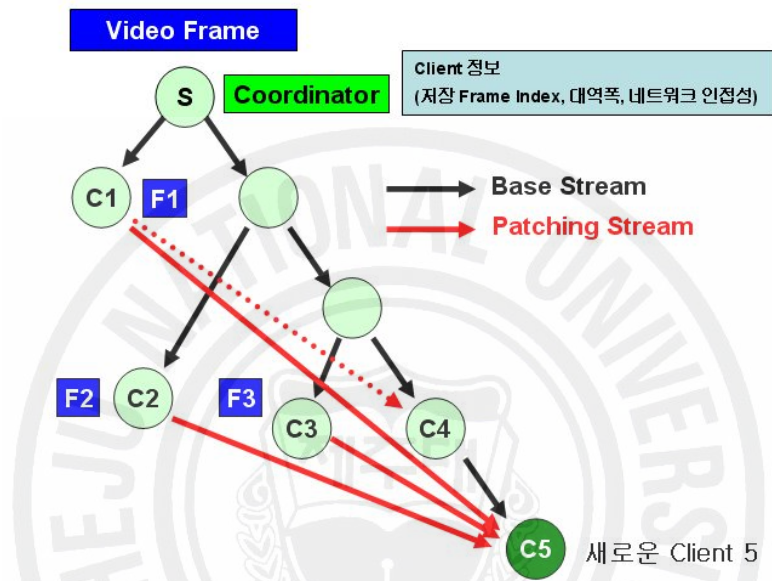


그림 10 Client 5의 기존 세션 Tree 조인 및 스트림 서비스 받는 모습

VoD 서비스를 받기 위해 가입한 클라이언트는 먼저 조정자에 접속하게 된다. 조정자는 접속한 클라이언트가 현재 조인 가능한 세션을 찾고 조인 가능한 세션이 있으면 해당 세션에 조인 되도록 하고 조인 가능한 세션이 없으면 새로운 세션을 생성하고 비디오 스트림을 서버로부터 수신 받는다. 기존 세션에 새 클라이언트 피어의 조인 시 조정자로부터 스트림 데이터를 받을 수 있는 피어들을 할당받게 된다. 이때 조정자는 비디오 스트림을 공급하게 될 피어들에 대한 네트워크 인터페이스의 통신용량과 비디오 스트림 현재 통신량을 알고 있기 때문에, 충분한 대역폭을 가지고 있다고 평가되며 또한 해당 비디오 데이터를 가지고 있는 피어들을 클라이언트에게 비디오스트림을 공급하도록 할당한다.

해당 세션 트리로부터 스트림 서비스를 받는 동작 화면은 그림 10에 보면 알 수 있다. 조정자는 스트림데이터를 블록 별로 구분해서 해당 데이터를 가진 다수의 피어들을 클라이언트 피어에게 할당해주기 때문에 비디오의 시작부분에 해당하는 데이터 블록은 서버에서 받아 스트림 플레이 서비스를 시작하는 동시에 그 이후의 데이터 블록들을 다른 피어들로부터 동시에 받게 된다.

피어는 해당 스트림 블록 별로 데이터를 모두 수신 하게 된다면 현재 자신이 저장하고 있는 스트림 블록 번호와 현재 사용하고 있는 자신의 네트워크 자원 정보를 서버의 조정자로 보내게 된다. 조정자는 이 정보를 이용하여 저장된 피어의 정보를 업데이트함으로써 가장 최신의 피어 정보를 유지 할 수 있게 된다. 피어는 스트림 데이터를 수신 하는 도중에 중간 피어의 이탈 • 장애 시 서버의 조정자로 접속 하여 장애 발생 중간 피어의 정보를 알리고 다른 해당 블록이 존재하는 중간 피어의 정보를 요청하게 된다. 피어는 다른 중간 피어로부터 해당 블록 스트림 데이터를 수신 하고 만약에 요청 중간 피어도 장애를 발생 하게 된다면 조정자에게 장애 피어의 정보를 알리고 서버로부터 해당 블록 스트림 수신 받게 된다. 충분한 스트림 데이터가 클라이언트 피어에 축적되면 이제는 상당부분 데이터가 축적되어 있기 때문에, 클라이언트는 실시간성의 제약성에서 벗어나서 비디오를 플레이할 수 있게 된다. 이때부터는 이미 축적된 비디오 데이터의 양이 충분하기 때문에 비디오 소스 피어의 개수가 작어 진다. 이는 클라이언트 피어가 소모하는 자원의 양을 최소화하기 위한 것이다. 이는 기존 ALM 방식의 비디오 스트림 방식이 가입과 패칭을 구분지어 기술되는 것을 동시에 이뤄지도록 한 것이다.



### 3) 세션



그림 11 기존에 서비스 받고 있는 Client4 와 새로 조인한 Client 5의 스트림 데이터 수신율 비교

새로운 클라이언트 피어가 VoD 서비스 스트림을 조정자에게 요청하면, 각 스트림 데이터에 대한 번호를 단위로 조정자가 가지고 있는 정보에 기초하여 새로 생성되거나 기존의 세션이 이용된다. 조정자는 새로운 클라이언트가 생기게 되면, 먼저 서버로부터 최초의 스트림데이터를 받게 하고, 초기의 스트림 데이터 블록들을 보유하고 있는 피어들에 대해 큰 유효대역폭 및 네트워크의 인접성 등에 대한 파라미터를 고려하여, 세션을 형성시킨다. 형성된 세션으로부터, 먼저 클라이언트 피어는 다중의 피어들로부터 스트림 데이터를 수신하게 된다.

그림 11은 기존에 서비스 받고 있는 Client4 와 새로 조인한 Client 5의 스트림 데이터 수신율을 비교 한 것이다. 보는 바와 같이 Client 1으로 부터 Video Stream에 처음 프레임을 받고 그 이후에 Video Stream에 대해서 다른 Client 2, 3 부터는 Patch Stream를 Base tree에는 정규 스트림을 동시에 받기 때문에 단 기간에 현재 플레이 되고 있는 스트림 이후의 프레임을 빠르게 받는 것을 알 수 있다.

이후, 조정 작업은 실시간의 제약에서 벗어난 상황이므로, 망 상황 및 시스템의 상황을 충분히 고려하여 스트림 데이터를 줄 수 있는 피어를 선정하여 비디오 스트림 데이터의 원천으로 배정한다.

#### 4) 장애복구

P2P 모델은 원천적으로 장애의 발생 원인을 가지고 있다. 피어의 장애는 물론이고, 피어가 인위적으로 동작을 중단, 이탈시 스트림 데이터를 수신 받던 클라이언트는 어쩔 수 없이 장애를 경험하게 된다.

본 논문에서는 피어의 장애, 이탈시 모두 장애가 발생한 것으로 보고 장애 복구 절차가 이루어진다. 장애복구 절차는 다음과 같다.

먼저 중간피어의 장애 발생 시 피어는 조정자로 접속을 하여 장애 피어의 존재를 알리고 비디오 블록이 존재하는 다른 중간 피어의 정보를 요청 하게 된다. 다른 중간 피어로부터 비디오 블록을 수신하게 된다면 장애 복구가 이루어진 것으로 본다. 만약 다른 중간 피어도 비디오 블록을 수신 받지 못하게 된다면 다시 조정자로 접속 하여 장애 피어의 존재를 알리고 VoD 서버로부터 필요 비디오 블록을 수신 받게 된다.

장애의 복구 문제는 본 시스템이 실시간성을 뛰어넘기 위한 것이기 때문에, 클라이언트가 이미 실시간 제약에 비해 충분한 데이터를 축적하고 있거나 주위의 다른 피어들이 비슷한 데이터를 가지고 있을 것이기 때문에 인근 피어에 연결하여 데이터를 받아오도록 할 수 있어서 큰 문제가 되지는 않는다. 다만 초기에 있어서 장애를 경험하게 되면, 실시간 제약에 묶여있을 수 있기 때문에 조정자가 비디오 데이터를 공급할 수 있는 피어를 빨리 할당해주는 것이 필요하지만, 이것 역시 하나의 피어로부터 받는 데이터가 블록단위의 데이터로 작기 때문에 여러 처리가 되어도 실제 전체 비디오 화질에는 큰 장애를 미치지 않게 된다. 기본적으로 장애에 큰 영향을 받지 않는 구조이지만 장애 발생 시 피어는 서버로 접속을 하여 바로 해당 블록을 서버로부터 직접 받게 하는 구조를 통하여 조정자가 계산하는 시간을 없애도록 하였다.

## IV. Peer-to-Peer 구조 VoD 서비스 시뮬레이션

### 1. 시뮬레이션 환경 및 설정

실험 시뮬레이션을 위해서 범용 네트워크 시뮬레이션 툴인 SSFNet[8] 을 사용하였다. SSFNet은 JAVA 와 C++ 를 지원하는 범용 네트워크 시뮬레이션 툴로서 본 논문에서 제안하는 시스템을 시뮬레이션 하기에는 충분하다.

실험을 위한 시뮬레이션 네트워크 환경은 기본적으로 17개의 Router와 16개의 Host을 서로 연결을 하였다. 각 라우터는 1GB로 연결을 하였으며 각각의 호스트는 각 라우터 와 100Mb, 10Mb로 연결 하여 4개의 소규모의 Network 망으로 연결된 Test Network망을 구성 하였다.

전체적인 실험 시뮬레이션 환경은 그림 12와 같다.

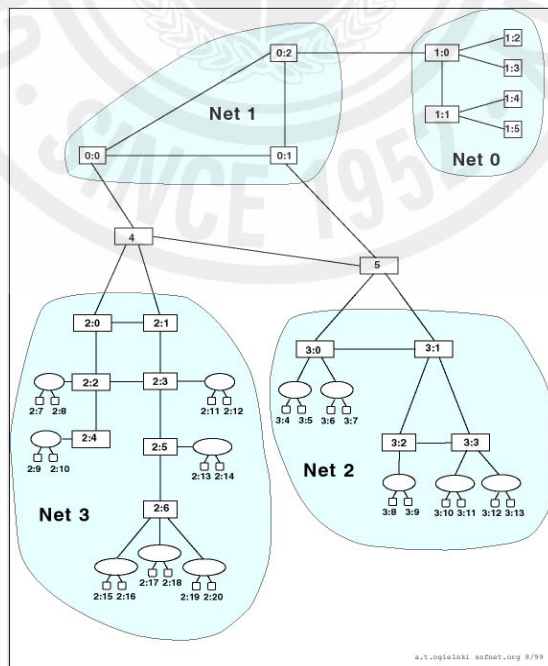


그림 12 시뮬레이션 네트워크 구성

먼저 실험을 위하여 Network 0의 1:2 호스트를 VoD 서비스를 위한 VoD 서버를 지정을 하였고 다른 각 네트워크의 호스트를 Peer로서 동작 하도록 하여 10초 마다 VoD 서버로 차례로 접속하도록 하였다.

실험이 시작 되면 각 네트워크의 호스트들은 VoD 서버인 1:2 호스트에 접속 하게 되고 서버의 조정자는 접속하는 호스트들을 접속하는 시간에 맞춰 스트림 서비스를 위한 트리를 구성하게 된다. 트리 구성이 끝나면 각 피어들은 서버 또는 상위 피어로부터 스트림 데이터를 전송 받게 된다.

실험 비교 분석을 위해서 P2 Cast인 2 Patching stream 인 경우와 본 논문에서 제안하는 다중 스트림 형태인 3 Patching, 4 Patching, 5 Patching 인 경우에 대해서 데이터 전송 속도를 측정 하였으며 스트림 데이터의 용량에 따른 제안 시스템의 향상을 측정하기 위해서 각 네트워크 Network 1, 2, 3 에서 Peer의 전체 비디오 데이터 전송이 완료 되는 시간을 측정 하였다. 또한 거리에 대한 제안 시스템의 향상을 측정하기 위해서 각 네트워크 Network 1, 2, 3 에서 Peer의 전체 비디오 데이터 전송이 완료 되는 시간을 측정 하였다.

## 2. 시뮬레이션 결과 및 고찰

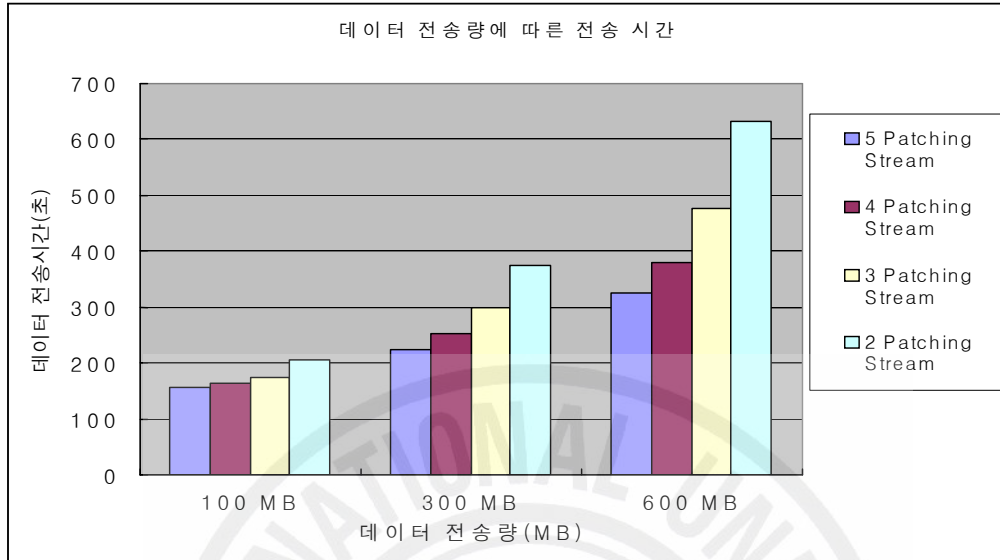


그림 13. Net 1 Peer의 데이터 전송량

그림 13 은 Network 1 에서 Peer의 수신 스트림 데이터양을 비교한 그래프이다.

먼저 2 Patching Stream 그래프는 P2Cast를 나타내며 본 논문에서 제안한 다중 스트림 서비스 형태인 3 Patching Stream, 4 Patching Stream, 5 Patching Stream 일 경우에 대해서 각각의 스트림 데이터 수신 시간을 비교 하였다. 각 용량에 따라 기존 P2Cast 보다 성능이 향상된 것을 볼 수 있으며 수신 스트림 데이터 용량이 적은 경우 100MB 에는 약 20%로 향상을 보였고 600MB인 경우에는 약 50% 정도의 향상을 볼 수 있었다. 따라서 본 논문에서 제안하는 다중 패칭 스트림을 이용한다면 피어의 수신 비디오 데이터양이 크면 클수록 더욱 많은 향상을 가져오는 것을 알 수 있다.

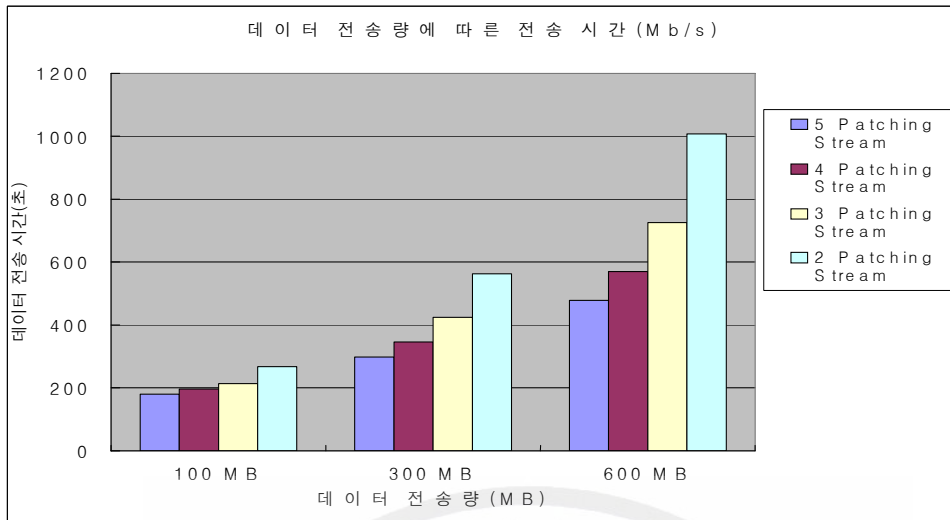


그림 14. Net 2 Peer의 데이터 전송량

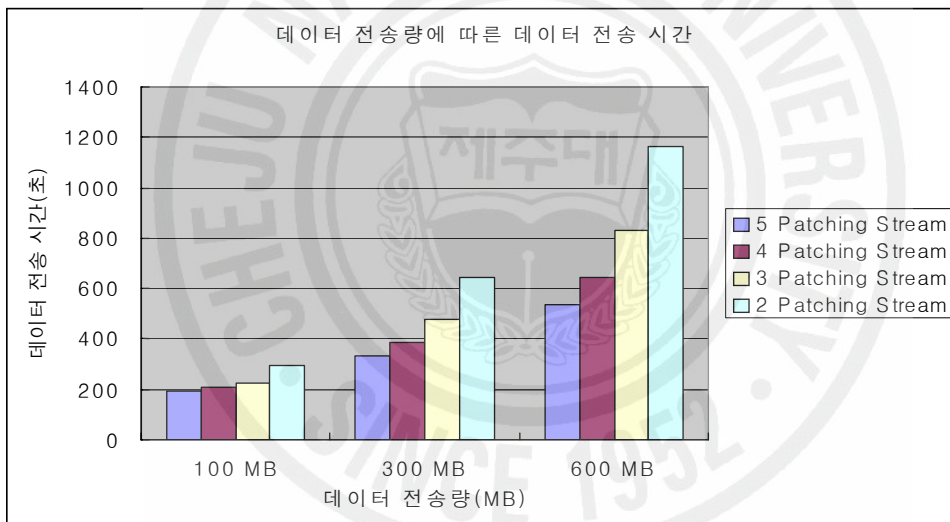


그림 15. Net 3 Peer의 데이터 전송량

그림 14와 그림 15는 Network 2 와 Network 3에서 Peer의 수신 스트림 데이터양을 나타내고 있다. Network 3의 Peer의 수신 데이터양을 비교 한다면 100MB인 경우에는 약 25%의 향상을, 600MB인 경우에는 약 55%의 향상을 보이는 것을 알 수 있다. 이것으로 VoD서버로 부터의 같은 거리에 있는 Peer들을 비교 했을 때 VoD 스트림 서버에서 근접한 Peer 들 보다 더 멀리 떨어진 Peer 들에게 더욱 효과 적인 것을 알 수 있다.

## . 결론 및 향후 연구

현재 인터넷을 통한 멀티미디어 서비스가 다양화 되어가고 있으며 사용자와 그에 따른 요구 또한 증가되고 있다. 그 결과 멀티미디어 데이터가 점점 고품질·대용량화 되어가고 있지만 인터넷 자원은 한정되어 있기 때문에 필연적으로 지터와 지연이 발생 하여 충분한 QoS를 사용자들에게 제공되지 않아 사용자의 요구를 만족시키지 못하고 있다.

따라서 본 논문에서는 현재 가장 많이 이용하고 있는 인터넷을 통한 멀티미디어 서비스인 VoD 서비스를 기반으로 인터넷 네트워크 전송 기술의 발전과 VoD 서비스를 제공 받는 단말기의 성능이 향상됨에 따라 단말기가 VoD 서비스의 중간 Peer로서의 기능을 수행 할 수 있다는 점을 착안 하여 다중 멀티 스트림 VoD 시스템을 제안하였다. 이 다중 멀티 스트림 VoD 시스템을 이용한다면 VoD 서비스를 제공 받는 Peer의 네트워크 자원을 최대한 이용할 수 있으며 현재 플레이 되고 있는 스트림 이후의 스트림을 서버가 아닌 중간 호스트 Peer로부터 전송 받을 수 있기 때문에 단기간 안에 최대한 많은 양의 데이터를 전송 받을 수 있다. 그 결과 서버의 부하를 줄일 수 있고 고화질·대용량의 멀티미디어 서비스도 지터와 지연이 없는 고품질의 QoS 가 보장된 VoD 서비스를 사용자들에게 제공 할 수 있을 것으로 사료 된다.

이 시스템은 오버레이 멀티캐스트인 ALM(Application layer multicast)자체 보다는 VoD 시스템에 대해 한정되어있으며 실시간으로 제공되는 스트리밍 데이터가 아닌 저장된 데이터에 대해서 사용자들마다 다른 시간에 데이터가 요구될 수 있는 상황만을 고려하고 있다. 본 시스템은 현재 사용자의 단말기가 갈수록 풍부한 자원을 보유하고 있다는 점에 근거하였고, 이를 근거로 사용자가 일정 프레임 단위로 데이터를 받은 것을 피어으로써 상당부분 서버로서의 기능성에 초점을 두었다. 따라서 단기간 에 현재 플레이되고 있는 스트림 이후의 스트림을 Peer 자신이 가진 네트워크의 대역폭을 최대한 이용하여 가능한 많은 양의 스트림 데이터를 수신함으로써 Vod QoS의 최대 관심사인 지터와 지연을 줄일 수 있다. 이

에 현재 VoD 서비스에 스트림 서비스의 품질을 향상 시킬 수 있으며 사용자들에게 충분한 QoS를 제공 할 수 있고 성능 면에서도 좋은 향상을 가져다 줄 수 있을 것이다.

향후 네트워크 전송 기술과 사용자 PC 성능이 좋아 질수록 본 시스템 구조는 더욱 성능을 발휘 할 수 있게 될 것이며 멀티미디어 스트림 플레이 속도에 비해 네트워크 속도가 빨라질수록 데이터의 실시간성에서 오는 지연과 지터 문제로부터 해방될 수 있을 것이다. 하지만 VoD 스트림 서비스의 고전 적인 문제점인 중간 Peer의 장애 발생 시 해결 방안은 본 시스템에서 중간 Peer의 의존도가 큰 만큼 이에 대한 효과적인 장애 복구 알고리즘에 대한 연구가 필요 하다.





## 참고문헌

- [1] S. Deering, "Host Extensions for IP Multicasting," RFC1112, August 1989.
- [2] Yang Guo, Kyoungwon Suh, Jim Kurose, Don Towsley, "P2Cast: P2P Patching Scheme for VoD Service", in WWW 12th, 2003.
- [3] Tai Do, Kien A. Hua, and Mounir Tantaoui, "P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment", in Proc. of the IEEE International Conference on Communications (ICC 2004), June 20-24 2004, Paris, France
- [4] K. Hua, Y. Cai, and S. Sheu, "Patching: A multicast technique for true video-on-demand services," in Proc. ACM Multimedia, September 1998.
- [5] V. N. Padmanabhan, H.J. Wang, P.A. Chou, "Resilient peer-to-peer streaming", in IEEE ICNP, Atlanta, GA, USA, November 19-22 2003.
- [6] S. Banerjee, B. Bhattacharjee, C. Kommareddy, "Scalable Application Layer Multicast," in ACM SIGCOMM, Pittsburgh, PA, USA, 2002.
- [7] D.A. Tran, K.A. Hua and T.T. Do, "A Peer-to-Peer Architecture for Media Streaming", in IEEE journal on Selected Areas in Communications, vol.22, no. 1, Jan 2004.
- [8] [HTTP://www.ssfnet.org/homePage.html](http://www.ssfnet.org/homePage.html)
- [9] 김종경, 이재혁, 박승규 "P2Patching : 주문형 P2P 서비스를 위한 효율적인 패칭 기법" 한국통신학회 논문지, 제31권2B호, pp.137-145, 2006
- [10] 이남일 "차등화 서비스에 IP Multicast를 적용시킨 IP Qos Multicast에 관한 개선 방안" 중앙 대학교 대학원 학위논문

- [11] 문종배 분산 VoD 서버 환경에서 히스토리 기반의 동적 부하 분산 스케줄러에 관한 연구“ 숭실 대학교 대학원 학위 논문
- [12] K. Hua, Y. Cai, and S. Sheu, "Patching: A multicast technique for true video-on-demand services," in Proc. ACM Multimedia, September 1998

