

3D Cyber Museum Service의 Multicast performance

Monitoring 구조 설계

조종근^o, 박인수, 박용진, 권용우
한양대학교 전자전기컴퓨터공학부

{jkcho^o, ispark, park}@hyuee.hanyang.ac.kr, ymk@kist.re.kr

Design of Multicast Performance Monitoring Architecture on 3D Cyber Museum Service

Joong-Keun Cho^o, In-Soo Park, Yong-Jin Park, Yong-Moo Kwon

Division of Electrical and Computer Engineering, Hanyang University

요 약

본 논문에서는 3D Cyber Museum Service에서의 효과적인 Multicast performance monitoring을 위한 receiver 기반의 분산 모델을 제안한다. 멀티캐스트 네트워크를 통해 전송되는 멀티미디어 서비스의 효과적인 performance monitoring을 위해서 사용자들이 경험하는 QoS를 사용자의 호스트상에서 보다 직접적으로 측정하고, 멀티캐스트를 사용함으로써 얻어지는 대역폭의 효율적인 사용에 대한 측정을 위한 모델을 설계한다.

1. 서 론

인터넷의 급격한 발달로 인해 사용자 수의 급증과 더불어 사용자의 요구가 기존의 텍스트 위주의 서비스에서 대용량, 실시간 멀티미디어 서비스에 집중되고 있다. 3D Cyber Museum Service는 한국과 유럽을 연결하는 KOREN-TEIN간 네트워크 상에서 3D를 이용하여 가상의 박물관 서비스를 제공하는 Service로서 네트워크로 대량의 데이터를 전송하기 위해 큰 대역폭을 요구하며, 이러한 사용자의 요구에 부응하기 위해, 일대-다, 다-대-다 통신을 지원하는 멀티캐스트는 필수적이라 할 수 있다.

멀티캐스트는 하나의 호스트에서 그룹으로 구별되는 다수의 사용자에게 패킷을 전송한다. 여기서 패킷을 송신하는 호스트는 누가 자신의 서비스를 받고 있는지, 서비스가 사용자에게 제대로 전달이 되는지 확인할 수 없다. 따라서, 네트워크 관리자 또는 서비스 제공자는 자신의 서비스가 제대로 전달되는지 확인하고 관리할 필요성이 있다. 멀티캐스트 모니터링 툴은 네트워크 계층의 정보를 분석하여 망 자체에 대한 정보를 제공하는 툴과, 어플리케이션 계층에서 사용자들이 경험하게 되는 QoS에 대한 정보를 제공하는 툴이 있다. 네트워크 계층의 정보를 제공하는 툴에는 Mantra, mrinfo, MultiMon, Mview 등이 있다. 어플리케이션 계층에서의 모니터링을 수행하는 툴에는 RTPMon, Mhealth, MListen 등이 있다. [1] 어플리케이션 계층에서 사용자에 대한 QoS를 분석하기 위해 사용되는 정보는 RTCP 메시지를 사용한다.

본 논문에서 제안하는 Multicast Performance Monitoring 구조는 사용자의 호스트에서 RTCP 메시지를 QoS 파라미터를

분석하여 서버에 전송하는 클라이언트 에이전트와 이 결과들을 모아서 전체적인 monitoring performance를 분석하는 server로 구성된다.

2. 본 론

2.1 클라이언트 에이전트 모델

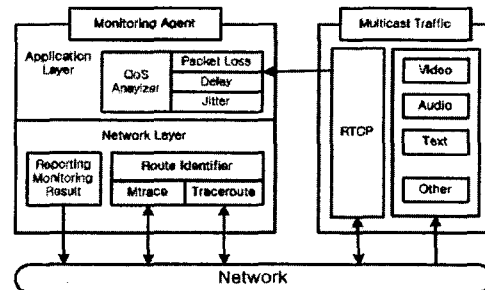


그림 1 Monitoring Agent at Client

본 논문에서 제안하는 모니터링 시스템은 기존의 시스템들이 모니터링 서버에서 각종 데이터를 받아 분석하는 방식을 벗어나서, End-User의 호스트에서 자신이 경험하는 QoS를 직접 분석하여 이 결과를 모니터링 서버로 전송하고, 모니터링 서버는 이 결과를 종합하여 실시간으로 전체 네트워크에 대한 정보를 보여 주고, 또한 이 결과를 DB화하여 통계정보를 제공하는 역할을 수행한다.

클라이언트 에이전트는 어플리케이션 계층에서 QoS를 분석하는 모듈과 네트워크 계층에서 Mtrace[2]와 traceroute를 수행하여 경로를 추적하는 모듈과, 모니터링 결과를 모니터링 서버로

전송하는 모듈로 구성된다.

QoS 분석 모듈은 호스트가 멀티캐스트 그룹에서 멀티미디어 데이터를 받고, 이에 대한 제어정보를 보내는 RTCP 메시지를 캡처하여, QoS 파라미터를 분석한다.

QoS 파라미터는 다음과 같은 것이 있다.

- packet loss[3][4]

RTP Spec에서 Packet Loss의 계산은 예상되는 패킷 수와 실제 받은 패킷 수와의 차이로 규정한다.

- One-Way Delay[5]

One-Way Delay는 source로부터 destination간 packet 전송 시 발생하는 delay를 말한다.

이 파라미터는 source측에서 패킷전송시 표시하는 timestamp의 값을 destination측에서 검사하여 network delay를 측정할 수 있다.

- Interarrival Jitter[6]

Jitter는 Network delay의 변화를 나타내는 parameter로 real-time stream전송에 매우 중요한 요소이다. RTP spec에서 jitter는 연속되어 들어오는 packet pair의 delay변화에 대한 평균 편차로 나타낸다.

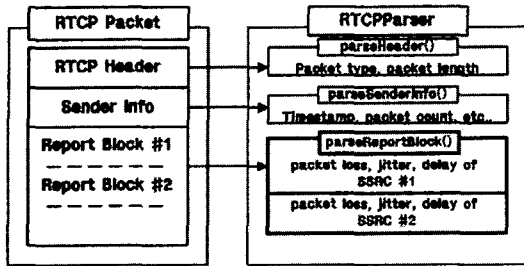


그림 2 QoS Analyzer

경로 추적 모듈은 두 가지로 나뉜다.

- Mtrace

Mtrace는 멀티캐스트 세션 내에서 source와 destination간 경로를 추적하는 툴이다.

Mtrace는 그 기능상 유니캐스트의 traceroute와 유사한데 그 방법은 전혀 다르다. traceroute는 TTL expiration을 이용하여 source에서 destination의 방향으로 추적하지만, 멀티캐스트에서 TTL을 사용하지 못하므로 역으로 destination에서 source로 경로를 추적하는 reverse-path lookup 방식을 사용한다.

- traceroute

traceroute는 대역폭의 효율을 측정하기 위해서 행해진다.

멀티캐스트는 다수의 사용자에게 데이터를 전송할 때, 공유하는 경로를 지날 때는 하나의 데이터를 보내다가 분기점에서 이 데이터를 복사해 각 사용자에게 보냄으로써 대역폭을 효과적으로 사용한다는 장점을 지닌다.

본 논문에서는 대역폭의 효율을 나타내는 파라미터로써 Chalmers와 Almeroth가 제안한 Hop을 사용하고[7], 에이전트

모델을 사용하여 보다 정확한 값을 계산해 낼 수 있음을 보인다.

Hop은 멀티캐스트 홉 수와 유니캐스트 홉 수를 비교해 전체적인 대역폭 사용에 대한 효율을 측정한다.

$$\delta = 1 - [\text{multicast hop}] / [\text{unicast hop}]$$

δ 이 1에 가까울수록 효율이 높다.

그림 3을 보면 멀티캐스트 토폴로지의 예가 있다. (a)의 경우 멀티캐스트 홉수는 6이고, 유니캐스트 홉수는 10이 된다. 따라서 $\delta = 1 - (6/10) = 0.40$ 이 된다. (b)는 멀티캐스트 홉은 9가 되고, 유니캐스트 홉은 19가 되고 따라서 $\delta = 1 - (9/19) = 0.53$ 이 된다.

Destination과 source간에 공유하는 링크가 많을수록 효율은 높아진다.

홉을 정확히 계산하기 위해서는 멀티캐스트 홉수와 유니캐스트 홉수를 알아야 한다.

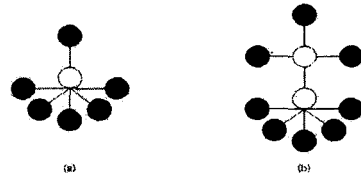


그림 3 Example tree topology

그림 3의 예제는 네트워크상의 모든 라우터가 멀티캐스트를 지원하는 즉, 터널링이 구성되어 있지 않았을 때를 가정한 것이다. 하지만 실제 네트워크는 멀티캐스트 라우터간 터널링이 구성되어 있기 때문에 Mtrace로 추적한 경로상에 실제 경로를 추적하기 위해 유니캐스트 traceroute가 필요하다. 하지만, 각 호스트간 traceroute를 수행하려면, 네트워크 관리자 또는 멀티캐스트 그룹의 active sender 즉 source가 모니터링을 수행하는 경우가 아니라면 불가능하다. 만약, 그룹의 source가 아닌 다른 곳에서 모니터링을 수행한다면, 그룹의 트리 구성에 관한 정보는 알 수 있겠지만, source가 아닌 다른 곳에서 수행한 traceroute와 source에서 수행한 결과는 다르기 때문에 위의 식에 적용할 수가 없다.

2.2 모니터링 시스템 구조

그림 3에서 모니터링 시스템은 모니터링 서버와 사용자의 호스트에서 동작하는 Agent로 구성되어 있다.

모니터링 서버는 모니터링하고자 하는 세션에 가입해서 그룹의 멤버간 교환되는 RTCP 메시지를 받는다.

모니터링 서버는 세션에 가입한 후, 멀티캐스트 그룹의 주소로 모니터링 에이전트를 전송한다. 에이전트를 전송 받은 그룹 멤버는 자신이 전송하는 RTCP 메시지를 분석하여 QoS 파라미터를 분석하는 동시에 그룹의 active sender로 Mtrace와 unicast traceroute를 수행하여, 경로를 파악한 뒤 이 결과를 모니터링 서버로 전송한다.

모니터링 서버는 이 결과를 받아서 전체적인 세션의 트리를 구성하고, 각 호스트별 QoS정보를 표시한다. 또한 이 결과를

DB에 저장하여, 시간에 따른 세션의 통계정보를 제공한다.

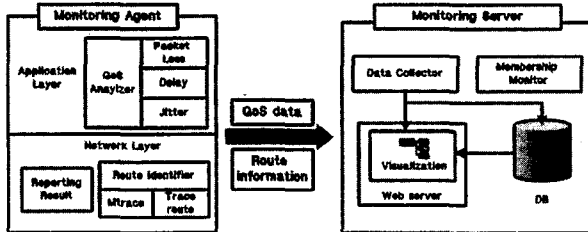


그림 4 Monitoring System Architecture

에이전트 모델은 기존 모니터링 시스템에서 발생하는 몇 가지 문제점을 개선할 수 있다.

첫째, RTCP 데이터에 대한 신뢰성을 확보할 수 있다.

RTCP 데이터는 RTP 프로토콜 내에 포함된 제어정보를 전달하는 프로토콜로서 UDP를 사용해서 전송된다. 기존의 시스템들은 주기적으로 전송되는 RTCP정보를 분석하여 사용자들의 QoS 파라미터를 분석하는데 UDP를 사용함으로써 RTCP 데이터 자체가 유실될 가능성이 있다. 에이전트 모델은 호스트에 다른 멤버에게로 보내는 RTCP 메시지를 직접 가져와서 분석한 뒤 이를 모니터링 서버로 HTTP protocol을 사용하여 유니캐스트로 전송함으로써 모니터링 데이터에 대한 신뢰성을 높일 수 있다.

둘째, 네트워크 부하를 줄일 수 있다.

에이전트 모델은 멀티캐스트 세션 모니터링에 두 가지 형태의 데이터를 제공한다. 첫째는 QoS에 대한 정보이고, 두 번째는 멀티캐스트 그룹의 트리 구성에 대한 정보를 제공한다. 이는 앞서 언급한 것과 같이 Mtrace를 수행함으로써 제공된다.

기존 모니터링 시스템에서 그룹의 멤버십에 대한 정보를 갱신하고, 트리 정보를 실시간으로 갱신하기 위해 주기적으로 경로 정보를 추적해야 했다. Mtrace는 각 source-destination 쌍에 대해 각각 수행해야 한다. 따라서 트리 정보를 유지하기 위해 모든 source-destination 쌍에 대해 순차적으로 반복적으로 수행해야 한다. 이는 source의 first-hop router에 반복적인 Query를 요청함으로써 라우터에 과부하를 줄 수 있는 요인이 된다.

에이전트 모델에서는 경로 추적에 대한 수행을 호스트 자체에서 수행하게 한다. 기존 모델에서 반복적으로 트리 추적을 한 이유는 멤버의 가입, 탈퇴에 따른 세션의 트리 구성이 변할 수 있기 때문에 이를 최대한 반영하기 위해서 이다. 하지만, 하나의 destination에서 source로의 경로는 쉽게 변하지 않기 때문에 에이전트 모델에서는 경로 추적을 모니터링 초기에 한번만 수행하면 된다. 호스트가 그룹에서 탈퇴할 경우에는, 에이전트는 이를 모니터링 서버에게 알림으로써 그 결과를 트리 정보에 반영할 수 있다. 네트워크상의 문제로 호스트의 에이전트가 탈퇴정보를 보고하지 못하는 경우가 발생할 수 있다. 모니터링 서버는

그물에 가입한 자신에게로 오는 RTCP 정보를 분석하여, source-destination 리스트를 구성하여 그룹의 멤버십을 모니터링하고 있으므로 리스트상의 호스트가 일정시간 동안 아무런 정보도 보내오지 않으면 탈퇴한 것으로 가정한다.

또한 QoS 파라미터의 전송에 있어서도, 각 파라미터의 최대 한도를 정한 뒤에 그 한도를 넘을 경우에는 즉시 보고를 하고, 보고 주기를 비교적 길게(10분~1시간) 설정하여 네트워크의 부하를 줄일 수 있다. 보고 주기는 어플리케이션의 종류 또는 네트워크의 상황에 따라 결정될 수 있다.

3. 결론

본 논문에서 제안한 멀티캐스트 Performance 모니터링 시스템은 기존 모니터링 서버에서 이루어지던 세션 모니터링과 QoS 파라미터의 분석을 End-user의 호스트에서 수행하도록 함으로써 보다 직접적이고 신뢰성 있는 QoS 정보를 제공할 수 있고 멀티캐스트의 대역폭 활용에 대한 파라미터를 보다 정확한 분석이 가능하며, 모니터링으로 인해 추가적으로 발생하는 트래픽을 감소하여 네트워크의 부하를 줄일 수 있는 모델을 제시하는데 그 의의가 있다.

4. 참고자료

- [1]K. Sarac and K. Almeroth, "Supporting Multicast Deployment Efforts: A Survey of Tools for Multicast Monitoring", Journal of High Speed Networking - Special Issue on Management of Multimedia Networking, March 2001
- [2]W. Fenner and S. Casner, "A 'traceroute' facility for IP multicast." Internet Engineering Task Force(IETF), draft-ietf-idmr-traceroute-ipm-*.txt, August 1998.
- [3]H. Schulzrinne, S. Casner, R. Frederick, and J. V., "RTP: A transport protocol for real-time applications." Internet Engineering Task Force(IETF), RFC 1889, January 1996.
- [4]G. Almes, S. Kalidindi, M. Zekauskas, "A One-way Packet Loss Metric for IPPM", Internet Engineering Task Force(IETF), RFC2680, September 1999.
- [5]G. Almes, S. Kalidindi, M. Zekauskas, "A One-way Delay Metric for IPPM", Internet Engineering Task Force(IETF), RFC 2679, September 1999
- [6]C. Demichelis, P. Chimento, "IP Packet Delay Variation Metric for IPPM", Internet Engineering Task Force(IETF), draft-ietf-ippm-ipdv-08.txt, November 2001
- [7]Chalmers, R.C., Almeroth, K.C. "Developing a multicast metric", Global Telecommunications Conference, 2000. GLOBECOM '00. IEEE, Volume: 1, 2000.