

nCUBE RTSP 기반 MPEG-2 TS 스트리밍 시스템 개발

조창식*, 배수영, 마평수, 강지훈*
전자통신연구원, 충남대학교*

MPEG-2 TS Streaming System based on nCUBE RTSP Protocol

ChangSik Cho, SuYoung Bae, PyeongSoo Mah, JiHoon Gang*
ETRI, ChungNam Nat'l Univ.*

요 약

사용자의 고품질, 요구와 사업자의 차별화된 서비스 제공 노력의 결과로 기존의 MPEG-4 기반이 아닌 고품질 전용의 MPEG-2 화질을 사용하는 VOD 서비스가 새로운 대안으로 제시되고 있다. MPEG-2 비디오는 높은 네트워크 대역폭을 요구하는 단점이 있는 반면, 사용자에게 양질의 화질을 제공할 수 있으며 표준의 사용으로 콘텐츠 유지, 보수에 유리하다. 본 논문에서는 상용 스트리밍 서버인 nCUBE 서버와 연동하여 MPEG-2 TS 데이터를 스트리밍 하는 VOD 시스템에 대하여 설명한다. VOD 제어 프로토콜로 RTSP(Real Time Streaming Protocol)를 사용하였으며, 스트림 전송 프로토콜로 UDP/IP 방식을 사용하였다. 지원하는 VCR 기능으로는 FF, RW, STOP, Pause가 있다.

1. 서론

현재의 주문형 비디오 시스템은 인터넷을 기준으로 서비스되며 낮은 대역폭에서도 작동하기 위하여 MPEG-4[1]를 기반으로 하여 서비스되고 있다[2]. 이는 사용자에게 비디오의 QoS에 대한 만족도를 희생할 수 밖에 없다. 최근 VDSL로 대표되는 초고속 인터넷의 발전과 광 접속망의 빠른 보급에 따라 최종 사용자에게 제공되는 대역폭은 늘어나고 있다. 또한 사업자들은 화질이나 음질에 대한 고급화를 통하여 기존의 서비스와 차별화를 시도하고 있다[3,4,5].

사용자의 고품질 요구와 사업자의 차별화된 서비스 제공 노력의 결과로 저속의 MPEG-4 기반이 아닌 고품질 전용의 MPEG-2 화질을 사용하는 VOD 서버

가 새로운 대안으로 제시되고 있다. MPEG-2 TS 데이터는 디지털 방송 데이터와의 호환성이나 네트워크 전송 표준의 호환성의 조건을 만족함으로써 고품질 비디오를 위한 표준 포맷으로 자리잡고 있다.

MPEG-2[6] 비디오는 높은 네트워크 대역폭을 요구하는 단점이 있는 반면, 사용자에게 양질의 화질을 제공할 수 있으며 표준의 사용으로 콘텐츠 유지, 보수에 유리하다. 특히 DVD 영화에서 사용되는 콘텐츠나 디지털 방송에서 사용되는 콘텐츠를 VOD 서비스에 쉽게 사용함으로써 콘텐츠의 유통 비용을 줄이는 효과가 있다.

본 논문에서는 상용 스트리밍 서버인 nCUBE[7] 서버와 연동하여 MPEG-2 TS 데이터를 스트리밍 하는 VOD 시스템에 대하여 설명한다. 4.5Mbps 혹은

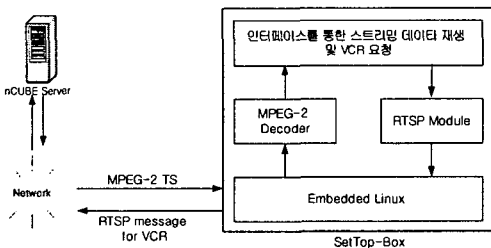
19Mbps로 인코딩된 MPEG2 TS 스트림을 이용하여 고품질 VOD 서비스를 제공한다. VOD 제어 프로토콜로 RTSP(Real Time Streaming Protocol)[8,9]를 사용하였으며, 스트림 전송 프로토콜로 UDP/IP 방식을 사용하였다. 지원하는 VCR 기능으로는 FF, RW, STOP, Pause가 있다.

2절에서는 VOD 시스템의 구성에 대하여 설명하고, 3절에서는 VOD 시스템을 연동하기 위한 절차에 대하여 설명한다. 4절에서는 스트리밍 환경하에서의 네트워크 지연 및 손실에 대한 실험 결과를 설명한다. 마지막으로 결론이 있다.

2. VOD 시스템 구성

본 논문에서 제시하는 VOD 시스템의 구성도는 (그림 1)과 같다. 스트리밍 서버로 nCUBE사의 n4x 서버를 사용하며, 클라이언트로는 임베디드 리눅스가 탑재된 셋탑박스를 사용한다. n4x 서버는 제어 프로토콜로써 RTSP를 사용하고 실제 스트림 전송은 MP2TS/UDP/IP 형태로 전송한다. 제어의 편리성이나 메시지 교환을 단순화시키기 위하여 nCUBE 서버의 경우 IETF 표준 RTSP[8] 프로토콜 이외에 확장된 RTSP[9] 메시지를 사용하는데, 본 시스템은 이를 적용하여 서버 성능을 최적화시켰다.

현재 SETUP, PLAY, PAUSE, TEARDOWN 메시지에 대하여 처리 가능하며, 각각의 메시지 전달 단계에서 상태의 변화를 관리하게 된다. 빨리보기 및 빨리 되감기를 위한 VCR 연산은 SCALE 속성 값을 가지는 PLAY 메시지를 사용하여 처리하였다.



(그림 1) VOD 스트리밍 시스템 구성도

n4x 서버를 이용해서 트릭 모드를 구현하는 방법

은 single asset 방식과 multi asset 방식으로 나누어진다 [8]. single asset 방식은 하나의 콘텐츠에서 사용자의 요청에 따라 실시간으로 트릭 모드 스트림을 생성해서 클라이언트로 전송하는 방식으로 만약 사용자가 2배속 빨리보기를 요청하면 서버는 콘텐츠에서 1 프레임만을 추출해서 클라이언트로 전송한다. multi asset 방식은 콘텐츠를 서버에 등록할 때 트릭 모드용 스트림을 동시에 생성해두는 방식으로, 특정 파일을 서버에 등록하면 서버에는 빨리감기용 파일과 되감기용 파일이 동시에 등록된다. 트릭 모드용 파일들은 I, P, B 프레임을 모두 가지고 있는 정상적인 MPEG 파일이 된다.

single asset 방식을 사용하면 서버에 부하가 걸린다는 단점이 있고, multi asset 방식은 트릭 모드 지원서버에 걸리는 부하를 줄일 수 있지만, 저장공간을 많이 차지한다는 단점이 있다. 본 스트리밍 시스템은 두개의 모드에 모두 작동하는 구조로 설계되었다

본 VOD 시스템에서 지원하는 포맷은 두 가지 종류가 있다. MPEG-2 비디오, MPEG-2 오디오가 4-5Mbps로 다중화된 SD급 MPEG-2 TS 데이터와 MPEG-2 비디오, AC3 오디오가 19Mbps로 다중화된 HD급 방송 데이터가 있다.

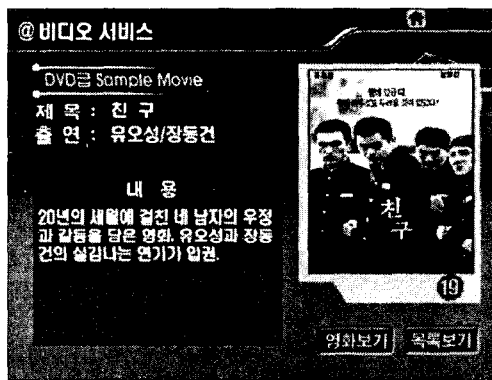
클라이언트 셋탑박스로는 저전력과 저소음의 요구 사항을 만족하는 인텔 Celeron CPU를 사용하였으며, SD급 데이터는 물론 HD급 방송 데이터의 실시간 디코딩을 가능하게 하기 위하여 ATI사의 Xilleon 하드웨어를 온보드화시켰다.

ATI Xilleon 225H[10] 드라이버는 기본적으로 포트에서 들어오는 MPEG-2 데이터를 처리하도록 설계되어 있다. 스트리밍 서비스를 위해서는 하드웨어 포트에 데이터가 전송되는 것이 아니라, 네트워크 소켓을 통한 데이터 입력을 처리할 수 있어야 한다. 즉 메모리에 임시 저장 버퍼가 존재하고, 카드에서 버퍼를 접근하여 필요한 만큼의 데이터를 가져가는 구조가 되어야 한다. 이를 처리하기 위하여 드라이버의 데이터 입력 부분을 수정하였으며, 메모리 버퍼에 저장된 데이터가 카드에 대한 입력이 되도록 드라이버를 수정하였다.

또한 ATI 카드는 포트상의 데이터를 처리하는 구조이기 때문에 데이터의 지연이나 손실에 대해서는 드라이버에서 관여하지 않는다. 이를 응용 레벨에서 처리할 필요가 있기 때문에, 카드에 대한 버퍼를 시스템, 비디오, 오디오 디코딩 버퍼의 세 단계로 나누고 각 버퍼에 지연이나 손실에 대해서는 드라이버의 작동을 일시 중지시키는 방법을 사용하였다.

3. VOD 작동 시나리오

VOD 시스템은 웹 페이지에서 해당 콘텐츠를 클릭하게 되면 웹 브라우저의 플러그인을 호출하는 구조로 이루어진다. VOD 서비스를 받기 위해서는 [비디오 서비스]를 선택하고, [영화 장르]를 선택 한 다음 현재 서비스 중인 영화의 목록을 볼 수 있다. 목록에 있는 영화 중 하나를 선택하면 해당 영화에 대한 정보를 볼 수 있으며, [영화보기]를 클릭하면 영화가 시작된다. 비디오가 상영되는 동안 시청을 중단하려면 리모컨의 [STOP] 버튼을 누르면 된다. (그림 2)는 영화를 선택하는 웹 페이지의 한 장면이다.



(그림 2) 영화 선택 웹 페이지

다음은 VOD 시스템의 대략적인 작동 시나리오이다.

- 사용자가 선택한 비디오 프로그램의 정보를 해당 VOD 서버로 전송한다.
- VOD 서버와 세션을 맺고 사용자의 VOD 관련 제어정보를 전송한다.
- VOD 서버로부터 전송되는 UDP/IP 패킷 형태의

MPEG2 TS 스트림을 재조합한다.

- 재조합된 MPEG2 TS 스트림을 DMA를 이용해 디코더로 전송한다.

웹 페이지에서 영화 보기를 클릭하면 해당되는 비디오 데이터베이스를 검색하고, 검색된 비디오 정보를 VOD 응용으로 넘겨준다. 전달되는 인자들은 콘텐츠에 대한 url 주소(nCUBE 서버 주소 및 파일 경로), 비디오, 오디오에 대한 PID 정보, 오디오 코덱의 종류(MPEG-2 오디오 혹은 AC3로 구분), 비트율 정보가 있다. URL 주소 정보는 실제 nCUBE 서버에 접속하기 위해 필요한 정보이다. PID 정보나 코덱 정보, 비트율 정보는 재생기의 자원 관리를 위해 사용된다. 즉 PID 정보를 가지고서 다중화 필터를 구동시키며, 코덱 정보에 따라 필요한 오디오 코덱을 구동시킨다. 또한 비트율에 따라 가변적으로 네트워크 수신 버퍼를 할당하게 되어 있다.

VOD 응용과 nCUBE 서버는 RTSP(Real Time Streaming Protocol)를 이용하여 통신한다. 클라이언트가 서버와 세션을 맺고 PLAY 메시지를 보내면 VOD 서버는 MPEG-2 TS 스트림을 클라이언트에게 전송한다. 스트림은 UDP/IP 패킷 형태로 클라이언트에 전송되고, 클라이언트 모듈은 전송받은 패킷들을 MPEG-2 TS 스트림으로 재조합한 후 재생 모듈로 넘겨준다. 네트워크 버퍼에서 ATI 카드의 재생 모듈로 MPEG-2 TS를 전송할 때는 DMA(Direct Memory Access)방식을 이용한다.

4. 네트워크 지연 및 손실 제어

본 시스템에서는 초기 버퍼링을 위해 비트율에 따라 5초 가량의 데이터 수신을 요구하며, 트릭모드 전환시에는 1초의 버퍼링을 요구한다. 5초 가량의 데이터 수신은 ATI 카드의 초기화를 위해서 필요한 데이터량을 기준으로 잡았다. 실제 초기 버퍼링 단계에서는 서버가 비트율보다 빠른 속도로 데이터를 클라이언트로 전송하게 되므로 5초 이내에 버퍼링이 완료된다. ATI Xilleon 드라이버를

초기화시키는데 1초 정도의 시간이 소용되므로 초기 재생 지연시간은 4.5초 정도가 된다.

트릭 모드 수행에 필요한 시간은 1초의 버퍼링만을 요구한다. 이의 근거로는 카드 자체의 버퍼가 1초에서 4초 정도의 지연을 상쇄시키는 기능을 가지고 있기 때문에 실제로는 2초에서 6초 정도의 지연이 있기까지는 시스템이 제대로 작동하게 된다. 실제 실험 결과로 1초 정도의 데이터 버퍼링으로 트릭 전환이 순조롭게 진행되었다.

또한 nCUBE 서버는 정해진 시간 간격마다 클라이언트가 살아 있는지를 검사하기 위하여 heartbeat 메시지를 주기적으로 검사한다. 만약 서버가 클라이언트로부터 heartbeat 메시지를 수신하지 못하면 자동으로 서비스를 종료하게 된다. 이를 반영하여 클라이언트는 90초의 네트워크 지연에 대하여 서비스를 종료시킨다. 이의 근거로는 특정 시간 이후 네트워크가 재개되더라도 이미 서버와의 접속이 끊긴 상황이기 때문에 서버와의 세션을 담당하고 있는 TCP 메시지 소켓이 깨어지면 서비스를 종료시키게 된다.

스트리밍 데이터 손실에 대한 실험으로 네트워크 에러에 의한 데이터 손실이 0.5% 이하로 나타날 경우에 VOD 서비스 자체에 대한 영향은 미미하였다. 손실율이 10% 이상의 경우 아주 심한 잡음과 화면 찌그러짐으로 인하여 영화 상영이 거의 불가능한 것으로 나타났다.

ATI 카드의 특성상 3.5초 정도의 딜레이에 대하여 오디오가 나오지 않게 된다. 이는 비디오, 오디오의 동기화가 어긋날 경우에 드라이버에서 오디오 필터를 없애 버리고, 비디오 데이터만을 재생하도록 구현되어 있기 때문이다. 오디오 데이터가 중지되면 네트워크 상황이 좋아진 이후에도 계속적으로 오디오 데이터가 복원되지 않는 문제점이 있기 때문에 본 시스템에서는 네트워크 지연에 대비하여 자동으로 오디오 디코더를 중지시키고, 네트워크 상황이 좋아지면 다시 오디오를 재개시키는 방법을 채택하였다.

이를 위해서 오디오 버퍼에 대한 lag를 원천적으로 차단시킨다. 오디오 버퍼가 비워지게 되는 순간에는 카드에 오디오를 off 시키는 것이 필요하다. 스트림 버퍼와 오디오 버퍼의 양을 계산하여 버퍼의 데이터가 일정 이하로 나타날 경우 카드를 일시정지시키고, 네트워크에서 데이터를 수신한 후 다시 재생 재개시키게 된다. 이때 오디오 버퍼의 데이터 양에 따라 제어하며, 네트워크 입력에서 받은 시스템 버퍼에 대한 입력의 양을 보조적으로 사용한다. 카드가 자체에서 처리할 수 있는 지연 시간을 계산해 보면 (표 1)과 같다. 이는 카드에서 할당된 버퍼의 크기를 비트율로 나누어보면 각각의 버퍼가 흡수할 수 있는 시간 지연을 나타낸다. 이것은 카드의 버퍼가 full 상태에서의 계산값인데, 비디오, 오디오 버퍼를 채우는 것은 카드의 필터가 자동으로 처리하기 때문에 응용 프로그램에서 제어할 수 있는 방법은 없다.

(표 1) 카드의 버퍼에서 처리할 수 있는 지연시간

버퍼 종류	버퍼수*버퍼 크기	전체버퍼 크기	버퍼 형
스트림 버퍼	20*99264	2MB	원형큐작동
비디오 버퍼	1*1310720	1.25MB	fifo 작동
오디오 버퍼	1*65536	64KB	fifo 작동

구분	비트율	버퍼 크기	버퍼링시간
MPEG-2 SD급 비디오	4Mbps	1.25MB	2.5sec
MPEG-2 HD급 비디오	19Mps	1.25MB	0.5sec
MPEG-2 오디오	128kbps	64KB	4 sec
AC-3 오디오	384kbps	64KB	1.3 sec

따라서 네트워크 지연을 제어하기 위해서는 네트워크 버퍼에서 카드의 입력으로 데이터를 채우는 스트림 버퍼 레벨에서 이를 제어하게 된다. VOD 프로그램으로 제어가능한 버퍼인 스트림 버퍼를 살펴보면 SD급(4-5Mbps)의 경우 3.8 sec, HD급(19.2Mps)의 경우 약 0.8 sec가 된다. 스트림 버퍼의 경우 정상 상태의 경우 17-19개의 버퍼가

채워져 있는 상태가 되며, 만약 스트림 버퍼가 완전히 lag되면 곧이어 오디오 버퍼가 lag 되게 된다. 따라서 카드가 자체적으로 버퍼링할 수 있는 시간인 최대 4 sec, 1 sec 정도의 시간을 가지고 카드의 pause/resume을 제어해야 한다.

5. 결론

본 논문에서는 상용 스트리밍 서버인 nCUBE n4x 와 연동하는 VOD 시스템에 대하여 설명하였다. SD급 혹은 HD급으로 인코딩된 MPEG2 TS 스트림을 이용하여 고품질 VOD 서비스를 제공하였다. VOD 제어 프로토콜로 RTSP(Real Time Streaming Protocol)를 사용하였으며, 스트림 전송 프로토콜로 UDP/IP 방식을 사용하였다.

본 논문에서는 5초의 초기 지연 시간과 트릭모드 전환시 1초의 지연을 기준으로 설계되었으며, 오디오 데이터가 lag된 상태에서 오디오 필터가 내려가는 것을 방지하기 위하여 오디오 버퍼 및 스트림 버퍼의 양을 항상 감시하도록 처리하였다.

본 시스템을 사이버 아파트에 실제 서비스를 함으로써 기능성 및 안정성을 검증하였다. 앞으로의 연구 사항으로는 DRM 기술을 적용하여 서비스 사업자가 콘텐츠 유통에 대한 관리를 좀 더 효율적으로 수행할 수 있도록 한다.

[참고문헌]

[1] ISO/IEC, Information Technology-Coding of Audio-Visual Objects, IS 14496, July 2000

[2] ISMA, Internet Streaming Media Alliance Implementation Specification Version 1.0, August 2001.

[3] D.J. Gemmel, H.M. Vin, D.D. Kandlur, P.V. Rangan, and L.A. Rowe, "Multimedia Storage Servers: A Tutorial", IEEE Computer, Vol. 28, No. 5, pp.40-49, May 1995.

[4] D.D. Kandlur, M.S. Chen, and Z.Y. Shae, "Design of a Multimedia Storage Server", IS&T/SPIE Int. Symposium on Electronic Imaging: Science and Technology, San Jose, pp. 164-178, Feb. 1994.

[5] E. Chang and H. Garcia-Molina, "Effective Memory Use in a Media Server", 23rd VLDB Conference, pp.496-505, Aug. 1997.

[6] ISO/IEC, generic coding of moving pictures and associated audio information, IS 13818, April 1996

[7] <http://www.ncube.com/pressroom/downloads/n41.pdf>, n4 Streaming Media System.

[8] IETF, RFC 2346, Real Time Streaming Protocol (RTSP), April 1998.

[9] nCUBE Corporation, nCUBE RTSP Implementation and Extensions, April 2000.

[10] <http://www.ati.com/products/dtv.html>, XILLEON 220.