

부분 캐싱을 지원하는 트랜스코딩 프록시에서의 이벤트 모델 설계

최민석^o 박유현 유영호 김경석

부산대학교 컴퓨터공학과 한국전자통신연구원 부산대학교 정보전산원 부산대학교 정보컴퓨터공학부
mschoi@asadal.pusan.ac.kr bakyh@etri.re.kr yhyu@pusan.ac.kr gimgs0@asadal.pusan.ac.kr

Design of Event Model in the Transcoding Proxy Supporting Partial Caching

Min-suk Choi^o Young-ho Yu Kyong-sok Kim

Dept of Computer Engineering, Pusan national university
Electronics and Telecommunications Research Institute

Information Technology Center, Pusan national University

Dept of Computer Science and Engineering, Pusan national University

1. 서론

프록시 서버는 서버와 클라이언트 사이에 위치하는 캐쉬 서버로 시간적, 공간적 지역성이 높은 데이터를 사용자 측에 가까운 곳에 위치한 프록시 서버에 저장하여 서비스한다. 프록시 서버는 사용자의 요청이 서버에 이르기 전에 먼저 받아서 요청된 데이터의 저장 여부를 검사하여 요청된 데이터가 프록시 서버에 존재하면(cache hit) 직접 데이터를 전송한다. 만일 프록시 서버에서 이미 저장되어 있지 않은 데이터로 판단되면(cache miss), 프록시 서버는 이에 대한 요청을 서버에게 전달한다. 요청을 받은 서버는 프록시에게 데이터를 전송하고 프록시 서버는 이 데이터를 먼저 저장한 후 사용자에게 전달한다.

트랜스코딩 프록시는 이러한 경우 외에 사용자는 낮은 품질을 가지는 버전을 요청했을 때, 프록시에는 해당 버전은 저장 하고 있지 않으나, 높은 품질의 버전을 저장하고 있어서 이를 통하여 요청한 버전을 만들 수 있는 경우에 기존의 프록시와는 다른 동작을 하게 된다. 이 경우에 cache system은 트랜스코더에게 트랜스코딩 작업을 요청하고 이의 결과를 사용자에게 전송하여 서비스를 제공하게 된다. 이러한 경우를 기존의 프록시에서 사용하는 hit, miss와는 구별되도록 transcoding hit(또는 transcode hit)라고 정의한다.

만일 트랜스코딩 프록시가 기존의 웹 프록시와 같이 객체 단위로 캐싱 하게 되면 hit, miss, transcoding hit의 세 가지 이벤트로 모든 경우를 표현할 수 있다. 하지만, 트랜스코딩 프록시에서 대상으로 하는 미디어의 종류가 기존의 웹 프록시에서 주로 다루는 작은 크기의 미디어가 아닌 대용량이 필요한 비디오, 오디오와 같은 미디어 이기 때문에, 객체 단위로 캐싱할 경우 프록시의 저장공간의 급격한 소모로 인해 캐싱 효율이 떨어진다. 따라서 트랜스코딩 프록시에 부분 캐싱을 적용하면 좀더 효율적으로 캐싱할 수 있다. 하지만, 기존의 이벤트로는 모든 이벤트들을 표현하는데 문제가 발생하게 된다. 본 논문에서는 부분 캐싱을 지원하는 트랜스코딩 프록시에서 발생하는 이벤트들을 정의한다. 이러한 이벤트들은 시스템 설계 시 발생할 수 있는 여러 가지의 현상들을 명확하게 표현할 수 있고, 각 이벤트에 대한 프록시 서버의 후속작업을 정확히 규정할 수 있다[1], [2].

2. 본론

넓은 의미의 트랜스코딩은 코딩된 신호를 다른 신호로 변환하는 것을 의미한다. 동영상에서의 트랜스코딩은 압축된 동영상을 다른 규격의 동영상으로 변환하거나 비트율(bitrate), 공간 해상도, 프레임율, 오류 탄력성(error-resilience) 등을 조절하는 기술을 의미한다. 또한, 트랜스코딩이 수행되는 장치 또는 그 구조를 트랜스코더(transcoder)라고 한다.

일반적인 프록시 시스템에서는 cache hit와 cache miss라는 두 가지 이벤트들이 사용된다. 이러한 이벤트들은 프록시 시스템이 어떤 동작을 수행해야 하는지를 결정한다. 즉, 프록시 서버는 cache hit 이벤트가 발생하면 즉시 사용자에게 요청한 데이터를 전송하고, cache miss 이벤트가 발생한 경우에는 먼저 서버에게 사용자 요청을 전달하여 데이터를 전송 받아 프록시 서버에 저장하고 이를 사용자에게 다시 전달한다.

트랜스코딩 프록시는 이러한 경우 외에 사용자는 낮은 품질을 가지는 버전을 요청했을 때, 프록시에는 해당 버전은 저장 하고 있지 않으나, 높은 품질의 버전을 저장하고 있어서 이를 통하여 요청한 버전을

만들 수 있는 경우에 기존의 프록시와는 다른 동작을 하게 된다. 이 경우에 cache system은 트랜스코더에게 트랜스코딩 작업을 요청하고 이의 결과를 사용자에게 전송하여 서비스를 제공하게 된다. 이러한 경우를 기존의 프록시에서 사용하는 hit, miss와는 구별되도록 transcoding hit(또는 transcode hit)라고 정의한다[3], [4].

만일 트랜스코딩 프록시가 기존의 웹 프록시와 같이 객체 단위로 캐싱 하게 되면 hit, miss, transcoding hit의 세 가지 이벤트로 모든 경우를 표현할 수 있다. 하지만, 트랜스코딩 프록시에서 대상으로 하는 미디어의 종류가 기존의 웹 프록시에서 주로 다루는 작은 크기의 미디어가 아닌 대용량이 필요한 비디오, 오디오와 같은 미디어 이기 때문에, 객체 단위로 캐싱할 경우 프록시의 저장공간의 급격한 소모로 인해 캐싱 효율이 떨어진다. 따라서 본 논문에서는 부분 캐싱을 적용하게 되었고, 트랜스코딩 프록시에 부분 캐싱을 적용하게 되면 기존의 이벤트로 표현되지 못하는 경우가 발생한다. 이러한 이벤트들은 시스템 설계 시 발생할 수 있는 여러가지의 현상들을 명확하게 표현할 수 있고, 각 이벤트에 대한 프록시 서버의 후속작업을 정확히 규정할 수 있다.

먼저 기존의 웹 프록시에서 사용하는 cache hit와 cache miss는 각각 H(Hit)와 M(miss-Fetch)로 정의한다. 또한 기존의 트랜스코딩 프록시에서 정의한 transcoding hit은 T(Transcoding hit)로 정의한다. 본 논문에서는 위에서 언급한 3개의 이벤트 외에 추가적으로 m(miss-Fetch&Transcoding)를 정의한다. 연속적인 캐싱은 하나의 버전을 캐싱할 때 이 버전이 모두 연속적으로 저장됨을 의미한다. 특정 버전은 앞부분(prefix)의 캐싱된 부분과 뒷부분(suffix)의 캐싱되지 않은 영역으로 나누어진다. 이러한 경우들을 정의한 기본 이벤트를 이용하여 표현하면 HT, HM, Hm, HTm, Tm과 같은 5개의 복합 이벤트들이 존재하게 된다. 연속적인 캐싱에서의 복합 이벤트들은 시간 단위로 보면 H+, T+, M+, m+, H+T+, H+M+, H+m+, H+T+m+, T+m+이 되며, 편의상 H, T, M, m, HT, HM, Hm, HTm, Tm 표현한다. 불연속적인 캐싱은 하나의 버전을 캐싱할 때 이 버전이 불연속적으로 저장될 수 있음을 의미한다. 이러한 방법은 VCR 기능을 제공할 때 유용한 캐싱 방법이다. 하지만, 이러한 방법의 캐싱은 메타데이터의 관리에 보다 많은 비용이 요구될 가능성이 높다. 불연속적인 캐싱의 경우들을 정의한 기본 이벤트를 이용하여 표현하면 H, T, M, m, HT, HM, Hm, HTm, Tm 이벤트에 TH, mT, mH, MH, MHT, MHm, ... , mTHM 등의 많은 경우가 나타난다.

3. 결 론

본 논문에서는 사용자 단말에 적응적인 스트리밍 서비스를 제공하기 위한 트랜스코딩 프록시에서 부분 캐싱을 지원할 경우의 이벤트들을 정의하였다. 정의하는 이벤트들은 기존의 연구에서 정의된 H(hit), M(miss), T(transcoding hit) 이외에 m 라는 새로운 이벤트로 기본 이벤트를 정의하고 이들을 결합하여 복합 이벤트를 정의하였다. 복합 이벤트들은 트랜스코딩 이벤트 전이 그래프 상에서 기본 이벤트들의 흐름으로 구성된다. 특히 연속적으로 캐싱하는 트랜스코딩 프록시에서는 H, T, M, m, HT, HM, Hm, HTm, Tm의 9개의 이벤트로 정의되며 불연속 캐싱에서는 그 이상의 이벤트가 발생한다.

이러한 이벤트들을 정의함으로써 트랜스코딩 프록시는 연속적인 캐싱, 불연속적인 캐싱 방법 등의 연구에 도움이 되는 모델이 된다.

참 고 문 헌

- [1] Yoohyun Park, Yong-Ju Lee, Hag-Young Kim, Kyongsok Kim, "Segment Based Caching Replacement Algorithm in Streaming Media Transcoding Proxy" 10th Asia-Pacific Network Operations and Management Symposium(APNOMS2007), LNCS4773, Sapporo, Japan, 10-12 Oct. 2007
- [2] Yoohyun Park, Yong-Ju Lee, Hag-Young Kim, Kyongsok Kim, "Efficient Segment Based Streaming Media Transcoding Proxy for Various Types of Mobile Devices" Pacific Rim Conference on Multimedia 2007(PCM2007), LNCS4810, Hong Kong, China, 11-14 Dec. 2007
- [3] Bo Shen, Sung-Ju Lee, Sujoy Basu, "Caching Strategies in Transcoding-Enabled Proxy Systems for Streaming Media Distribution Networks", IEEE Transactions on Multimedia, Vol.6, No.2, Apr. 2004
- [4] Chi-Feng Kao, Chung-Nam Lee, "Aggregate Profit-Based Caching Replacement Algorithms for Streaming Media Transcoding Proxy Systems", IEEE Transactions on Multimedia, Vol.9, No.2, Feb. 2007