확장성 높은 웹 기반 실시간 비디오 스트리밍을 위한 플랫폼 구조 설계

윤동식*, 김성환*, 최규범*, 윤찬현*
*한국과학기술원 전기 및 전자 공학과
e-mail : {dongsik.yoon, s.h_kim, mosfet1kg, chyoun}@kaist.ac.kr

An Architectural Design for Scalable Web based Real-time Video Streaming Platform

Dong-Sik Yoon*, Seong-Hwan Kim*, Gyu-beom Choi*, Chan-Hyun Youn*
*Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology

요 약

Web Real-Time Communication(WebRTC) 기술은 웹 기반으로 실시간 스트리밍 서비스를 제공하기 위해 쓰이는 기술이다. WebRTC 기술은 인터넷 상의 두 컴퓨팅 노드 사이의 Peer-to-Peer RTP 연결을 가능케 함으로써 상호간 실시간 스트리밍을 가능케 하지만 컨퍼런스 시스템과 같이 다수의 노드가 통신에 참여하는 경우 네트워크 연결은 메쉬 토폴로지의 형태로 구성되어 대역폭의 한계로 확장성에 제한이 존재한다. 따라서 본 논문에서는 인코딩과 먹싱을 통해 다중 노드간의 통신을 중계를 지원하는 Multipoint Control Unit(MCU)과 MCU 사이의 클러스터링, 그리고 클라우드 플랫폼을 통해 확장성 높은 실시간 스트리밍 서비스를 지원하는 플랫폼 구조를 설계하고 그 예제를 보인다.

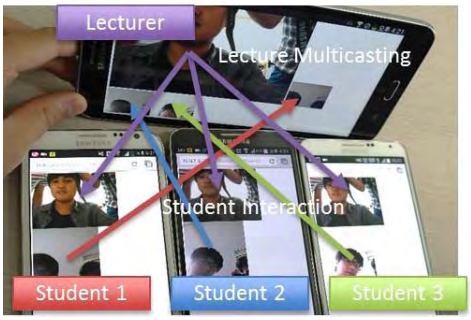
1. 서론

WebRTC 는 웹 브라우저를 통해 실시간 비디오/데이터 스트리밍을 제공하는 기술로서 높은 호환성을 바탕으로 모바일 컨퍼런스 시스템 등에서 활용된다. 웹 표준 API 를 이용함으로써 네이티브 어플리케이션에 비해 다소 낮은 성능을 보이지만 높은 접근성과 실제 개발자가 고려해야 할 부분들을 줄임으로써 다양한 스트리밍 웹 어플리케이션의 제작을 가능케 한다. 그러나 WebRTC 는 Peer-to-Peer 연결을 전제로 함으로써 통신하고자 하는 노드가 늘어날수록 Mesh-network 를 구성하기 위한 스트림 연결의 수가 기하급수적으로 늘어나므로 성능이 급격히 저하된다. 이를 해결하기 위한 방법으로 MCU 는 전용 장비를 통해 세션으로 들어오는 미디어 스트림을 실시간으로 하나의 프레임 안에 취합함으로써 노드 추가에 따른 성능 제약을 줄인다. 또한 사용자 수용 증가에 따른 단일 MCU 수용 한계 문제는 다중 MCU 간의 부하 분산 기법을 통해 해결할 수 있으며 클라우드 자원 연계로 탄력적 대응이 가능하다. 본 논문에서는 확장성 높은 클라우드 플랫폼을 기반으로 MCU 와 다중 노드의 접속 관리를 가능케 하는 기반 기술 연구, 플랫폼 구조 설계 및 프로토타입 테스트를 수행하였다.

2. WebRTC 기술

WebRTC[1]는 웹 표준 API 를 통해 획득한 미디어 장치(e.g. 웹캠)로의 접근 권한을 이용하여, 해당 미디어 스트림을 RTCPeerConnection API 를 통해 원격 노드에 스트림 세션을 구성 및 전송 가능케 하는 기술 집합으로, 이를 통해 웹 브라우저 주도의 음성 및 화상 통신이 가능해진다. 복잡/다양한 글로벌 네트워크 환경에서 WebRTC 세션 구성은 웹 브라우저만으로도 항상 유효해야 하므로 이를 지원하기 위해 SIP, JSEP, Signaling 서버, ICE 기술이 제공되며 결과적으로 상호간의 Real-Time Transport Protocol(RTP) 세션이 구성된다. 이러한 RTP 는 UDP 기반의 time stamp 정렬 방식으로 동작하는 프로토콜로서 실시간 전송에 적합한 송수신 패킷 구조를 정의하고 있으며 SIP 를 통한 세션 협상과정을 통해 합의된 요구사항(e.g. 허용대역폭, 비트레이트)을 기준으로 미디어 스트림을 전송한다.

Peer-to-Peer 연결 방식이므로 통신에 참여하는 노드 수가 증가할수록 요구 대역폭 또한 늘어난다. Mesh topology 에서는 n 개의 노드가 서로 연결할 경우 최대 n(n-1)/2 의 연결이 요구된다. 이러한 연결 방식은 다수의 노드가 참여하는 컨퍼런스 환경에 적합하지 않다.

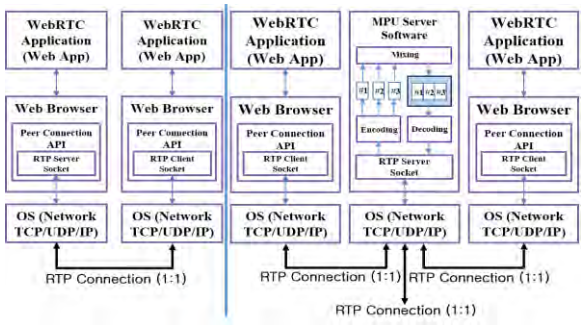


(그림 1) WebRTC 모바일 강의제공 환경 테스트

해당 기술을 모바일 환경에서 테스트해 본 결과 학생 수가 늘어날수록 허브인 강사 노드에 CPU, 메모리, 대역폭 점유율이 증가되어 송수신 딜레이가 발생하였으며 특히 모바일 디바이스의 낮은 성능으로는 사용에 제한이 있음을 확인하였다. 따라서 사용자 단말의 성능에 무관하게 서비스 품질을 보장하기 위한 차선책이 요구된다.

3. Multipoint Control Unit (MCU)

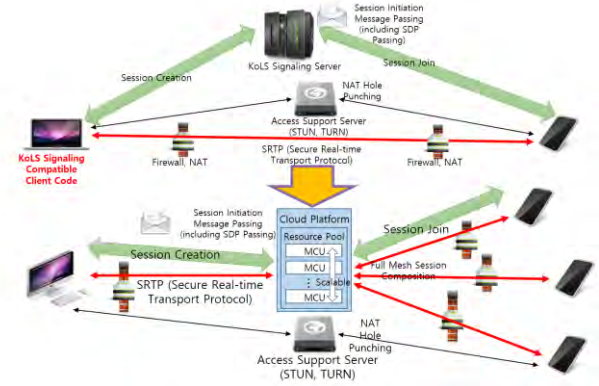
MCU 는 비디오 컨퍼런싱에 사용하는 전용 장비로서 다수의 노드들을 수용할 수 있다. 일대일 통신에서의 노드 수 한계를 효과적으로 보완하기 위해 다수의 RTP 스트림을 디코딩, 믹싱, 인코딩하여 하나의 프레임으로 합쳐 각 노드에게 전달한다. 선택적 믹싱 기법은 각 노드로부터 받은 비디오의 중요도에 따라 가중치를 부여함으로써 화질 및 대역폭을 차별적으로 믹싱하며, 중요도가 낮은 비디오 대역폭을 줄임으로써 효율적으로 자원을 활용한다[2]. 그러나 단일 MCU 는 수용할 수 있는 노드 수에 한계가 있기에 다중 MCU 클러스터링 기법을 통해 이를 해결한다.



(그림 2) (좌)P2P 기반 스트리밍 (우)MCU 기반 중계

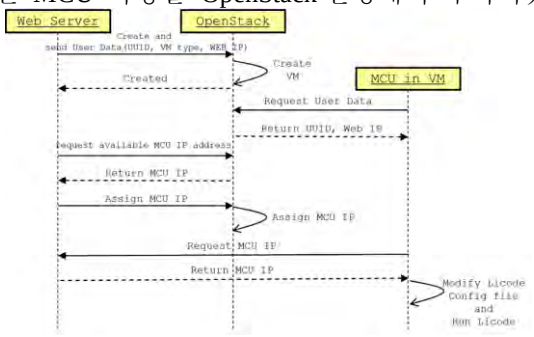
4. 클라우드 플랫폼 기반 MCU

분산 MCU 기법은 노드 수 증가로 단일 MCU 가 처리할 수 있는 작업 부하량을 초과할 경우 RTP 세션을 다중 MCU 에 중계하여 분산 처리함으로써 허용할 수 있는 동시 접속자 수를 증가시킨다[3]. 이러한 분산 MCU 를 효율적으로 활용하기 위해 작업 부하 변화에 따라 MCU 자원을 동적 할당하는 탄력적인 클라우드 모델을 이용할 수 있다. 이에 클라우드 자원 연계형 MCU 관리 플랫폼 구조를 그림 3 과 같이 제안한다.



(그림 3) 클라우드 자원 연계형 MCU 관리 플랫폼

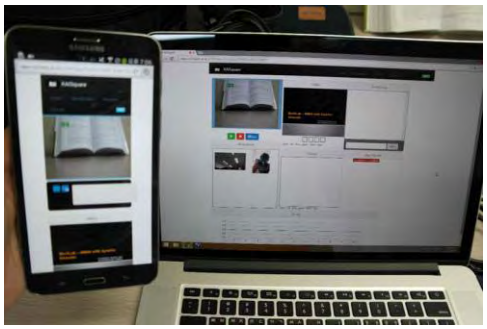
해당 구조는 MCU 의 사용량 분석을 기반으로 클라우드 연계를 통한 추가 MCU 가상 머신 프로비저닝을 수행한다. 그림 4 의 흐름에 따라 동작하는 실시간 MCU 확장을 OpenStack 환경에서 구축하였다.



(그림 4) 실시간 MCU 확장 Procedure

5. 프로토타입 테스트 수행 결과

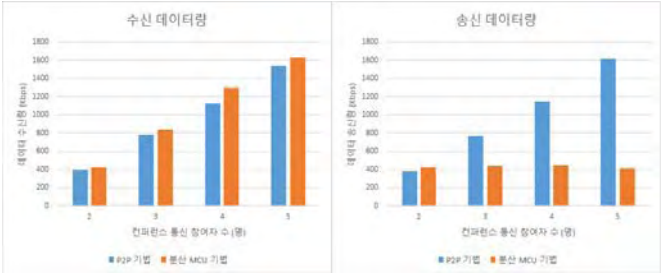
위에서 기술한 플랫폼을 일반 서버장비에서 MCU 서버 구축을 지원하는 오픈소스 플랫폼 Licode[3]를 통해 구현하였다. 그림 5 는 교육플랫폼 환경 구축 결과이며 클라우드 가상머신을 통해 배포된 MCU 간의 부하 분산 수행 내역을 네트워크 패킷 분석 툴인 Wireshark 를 통해 확인하였다.



(그림 5) 교육플랫폼 환경 구축 테스트 결과

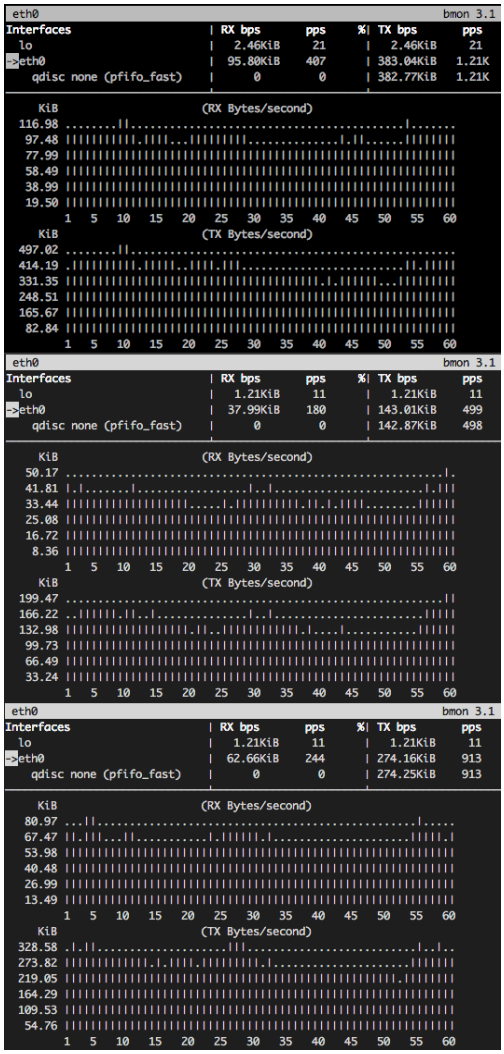
6. 기존 P2P 기법과의 효율 비교

본 논문에서 제안한 분산 MCU 기법의 실제 효율성 증대를 확인하기 위해 데스크탑 1 대, 노트북 4 대가 연결된 컨퍼런스 환경 하에 기존 P2P 기법과 분산 MCU 기법에서의 송수신 데이터량을 각각 측정 및 비교하였다. 각 영상의 최대 대역폭은 300Kbps 수준으로 제한하였으며 해상도는 640*480 화소이다. 컨퍼런스 통신 참여자 수는 2-5 명으로 변화시켰으며 송수신 데이터량 측정은 컨퍼런스에 참여하는 기기 중 특정 노트북 1 대에서 수행하였다.



(그림 6) 각 기법에서의 송수신 데이터량 비교

분산 MCU 기법의 경우 기존 P2P 기법과 달리 컨퍼런스 통신 참여자 수가 증가하더라도 송신 데이터량 변화가 거의 없음을 그림 6 을 통해 확인하였으며 이는 다수가 참여하는 컨퍼런스 통신 환경에서의 데이터 전송량을 MCU 를 통하여 획기적으로 감소시킬 수 있다. 이는 동시 참여 노드 수의 증가와 연계된다.



(그림 7) 다중 MCU 에서의 작업 부하량 분산 처리

본 테스트에서는 클러스터링된 세 개의 머신에 각각 MCU 를 실행시킴으로써 분산 MCU 기법을 수행하였다. 그림 7 은 위에서부터 차례로 각 MCU 노드당 작업 부하량을 나타내며 이로써 실제 MCU 분산이 수행되었음을 확인하였다.

7. 결론

본 논문에서는 웹 기반 실시간 비디오 스트리밍을 제공하는 WebRTC 및 한계와 이를 보완하는 MCU 의 활용을 소개하였다. 통신 노드의 수 증가에 대처하는 유연한 MCU 클러스터링 기법을 더했으며 확장성 높은 클라우드 플랫폼 기반 기술 연구와 테스트를 수행하였다. 별도의 제약 없이 저비용 고효율로 다자간 통신을 가능케 함으로써 모바일 컨퍼런스 환경 구축 등 여러 방면에 기여할 것으로 본다.

Acknowledgement

본 연구는 미래창조과학부 '범부처 Giga KOREA 사업'의 일환으로 수행하였음. [GK13P0100, Giga Media 기반 Tele-experience 서비스 SW 플랫폼 기술 개발]

참고문헌

[1] WebRTC. http://www.webrtc.org/
[2] Eleftheriadis Alexandros 외 2 인, "*Multipoint videoconferencing with scalable video coding*".
[3] Licode. http://lynckia.com/licode/