

League Of Legend(LOL)가 SBS에서 중계되기까지

올해는 빅 이벤트가 유난히 많은 해이다. 북미정상회담, 2018 지방선거, 러시아월드컵, 아시안게임, 남북정상회담, 이산가족상봉을 거쳐 자카르타-팔렘방 아시안게임까지. 송출기술팀 회선업무(NQC)를 담당하는 필자로선 그 어느 해보다 다이내믹한 해가 아니었나 싶다. 상기 이벤트 중 예측 가능했던 것들이 있는 반면, 정상회담의 경우에는 누구나 그렇듯 예상치 못한 것들이고, e-Sports LOL 중계방송 또한 그러했다. 필자는 이 지면을 통해 기존의 광 회선을 이용한 방송 Feed가 아닌 Tencent Cloud를 통해 RTMP Feed를 Pulling 및 Baseband 화하여 중계방송을 준비한 경험을 공유하고자 한다.

E-Sports, 자카르타-팔렘방 아시안게임 시범 종목으로 채택되다



그림 1. Faker 이상혁 국가대표 선수

지난 5월, 인도네시아 e-Sports 협회는 6가지 게임을 자카르타-팔렘방 아시안게임의 시범종목을 선정했고, 게임들은 각각 클래시 로얄, PES, LOL, 스타크래프트 2, 하스스톤, 펜타스톰들이다. 이 중 LOL은 스타크래프트 1 다음으로 e-Sports의 중흥을 이끈 주역이라고 할 수 있다. 매년 열리는 국제대회 중 하나인 미드시즌 인비테이셔널(MSI)의 누적 시청자는 3억 6천만 명이었으며, 최고 동시 시청자 수는 430만 명이고 총상금 규모는 20억 원이었다고 하니, 그 규모가 매우 크다고 할 수 있다. 심지어, 세계 최고 선수인 'Faker' 이 상혁 선수는 국내 모든 스포츠 선수들보다 연봉이 높다고 한다.



그림 2. SBS x 아프리카TV 로고

그동안 OGN이나 SPOTV Games, 아니면 Twitch나 아프리카 tv로 방송되던 것을 지상파에서 중계방송을 한다니 매우 뜻깊은 일이 아닐 수 없고, 이번 중계방송을 급하게 준비했던 필자로서는 새로움의 연속이었다.

새로움, 긴장과 짜릿함의 경계선

이번 e-Sports 방송의 경우 지상파 3사에서는 중계 생방송 혹은 딜레이 중계방송을 했다. 앞서 기재한 대로 6종목의 E-Sports의 국제 신호(IS)는 기존 방식의 MSPP 장비를 통해 SDH 회선으로 G.703 등의 인터페이스를 사용, 통신사를 거쳐서 전달받는 것이 아닌, Cloud를 통해 RTMP로 소스를 전달받는다. 사실 방송 회선에 대해서도 지면에 할애할 수 있는 내용이 많지만 다음으로 미뤄두고 스포츠 빅 이벤트 회선에 대해서 얘기해보자면 다음과 같다. 기존 국제 회선의 경우 주/예비 회선이 각각 STM-4(622Mbps), STM-1(155Mbps)으로 이중화한 후 양단에 Codec을 이용하여 비교적 안정적으로 소스를 전달받을 수 있다. 이런 경우에는 보통 Protected/Managed 회선으로 광 구간 장애, 장비 고장의 경우가 드물다. 그렇다면 e-Sports 국제 신호는 어떠한가.

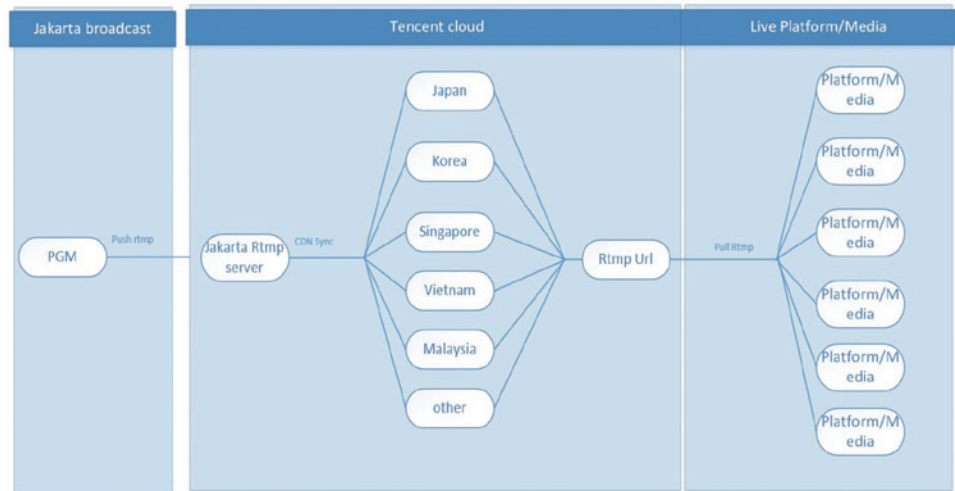


그림 3. E-Sports 국제 신호 FEED Distribution

AESF(Asian Electronic Sports Federation)로 부터 전달받은 e-Sports Summary Draft 중 Distribution 부분만 발췌한 그림이다. 위의 그림에서 알 수 있듯이, 프로그램을 자카르타 현지 RTMP Server로 밀어주고, 각 방송사들(RHB)은 Tencent Cloud에 RTMP Pulling 요청을 하여 수신하는 것이다. (참고로 Tencent는 중국의 인터넷 서비스 기업으로 포털(QQ.com)이나 SNS(WeChat) 등의 플랫폼을 보유하고 있으며 2011년 LOL의 개발사인 라이엇 게임즈를 인수함) Pulling URL의 경우에는 사별로 받았으며 백업의 경우 공통 URL을 받았다. 과거 필자는 다음 팟플레이어로 2015년 진도 팽목항 부감 장면의 RTMP Feed를 수신하여 Baseband로 변환해 뉴스 소스로 만든 경험 이 있다. 당시에도 화질 열화와 끊김이 발생하여 매우 애먹었던 기억이 있어 긴장도 되고 한 경기당 1시간 내외의 게임을 그대로 방송하 는 컨셉은 신선하게 다가왔다. 아마도 방송국에서 익숙하지 않은 수신 방식 RTMP 때문일 것이다. RTMP, 도대체 무엇인가

RTMP, 도대체 무엇?

RTMP는 Real Time Messaging Protocol로 Adobe가 만든 네트워크 계층의 프로토콜이며 TCP 등의 상위 계층에 올려 사용할 수 있다. Client는 Server에 Handshaking을 요청함으로써 통신이 시작된다. Client는 C0, C1, C2 3개의 메시지를, Server는 S0, S1, S2 3개의 메 시지를 가지고 Initialization을 한다. C0와 S0의 메시지는 RTMP 버전만 들어 있다. 현재 Version은 3이고 이전의 0, 1, 2는 예전 버전으 로 현재는 사용하지 않는다. Client가 먼저 C0를 보내면 Server가 S0를 던지고 다시 Client가 C1을 보낸다. C1에는 총 192Byte로 4Byte 가 Time 정보, 다시 4Byte는 Zeros, 나머지 Byte는 임의 값으로 이뤄져 있다. 이를 받은 S1 또한 동일한 메시지를 Client에 보내면 다시 C2를 수신한다. C2와 S2에는 특별한 Data는 없고 시간 정보만 들어 있다. Handshaking이 끝나면 Client는 Command Message를 날 리는데, 이 메시지들에는 Play, Receive Audio, Video, Seek, Pause와 같은 Control Message가 있다. 이는 마치 RTP를 사용하여 미디 어를 전송할 때 RTP나 RTCP와 같은 프로토콜을 사용하여 Control 하는 것과 비슷하다. Client에서 Play Message를 Server에 보내면 Server에서 Fragmented Message(Chunk) Size를 정하고, 몇 개의 Control Message(StreamIsRecorded, StreamBegin)와 Command Message(Onstatus-Play-Reset/Start)를 보낸 후 Audio/Video를 보낸다.

이 RTMP 방식은 현재도 많이 사용된다. Youtube에서 Live 방송할 때나, Twitch에서는 HLS(Http Live Streaming)로 스트리밍 서 비스를 하는데 HLS로 보내기 전에 RTMP로 받아 Video Ingest를 하고 이를 다시 트랜스 코딩하여 서비스한다고 한다. 그렇다면 ABR(Adaptive Bitrate)도 지원이 될까? 답은 '그렇다'이다. 이번 국제 신호의 경우에는 5000kbps, CBR로 제작하여 Feed를 분배하였 지만 RTMP에서는 ABR도 지원한다. Command Message Play2를 보내면 Server는 New Rate로 전송을 하게 되어 있다. 위에도 언급 했듯이 현재 대부분의 실시간 미디어 스트리밍 방식은 Adaptive Streaming을 지원한다. ATSC 3.0 ROUTE 방식 또한 그러하다. 바야흐 로 Progressive Download, Pseudo Streaming에 이어 Real-Time Streaming이 안착하였다.

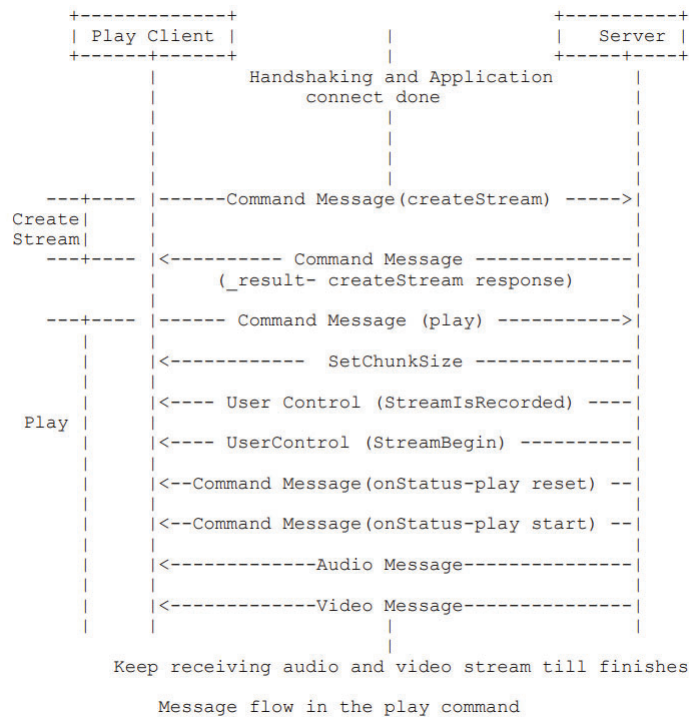


그림 4. RTMP Play 과정

방송사고, 방지 또 방지

Adobe에서 배포한 RTMP Specification 52page의 내용 중 미디어 끊김 없는 전송을 보장한다는 부분은 그 어디에도 없다. 당연하다. 규약에 없는 방송사고 방지는 엔지니어가 마땅히 할 일이다. 하지만 필자에게는 빈약한 지식과 경험만 존재할 뿐이었다. 이를 불쌍히 여긴 기라성 같은 선배들의 도움이 아니었다면 이렇게 독자와 만날 수 있는 기회가 없었을 것이다.

모든 PC 화면이 그러하듯이 원치 않은 메시지를 접할 때가 많다. 백신 프로그램에서 알람이 뜰 수도 있고, 갑자기 프로그램이 다운돼 바탕화면이 보이는 순간도 있고 하물며 마우스라도 살짝 건드리다가는 커서를 모든 시청자에게 보이게 된다. 방송에서는 이런 화면 노출이 바람직하지 못하다. 이에 자카르타 IBC에 계신 선배께서 친히 파라미터와 함께 MPlayer 사용법을 전달하사, 필자의 플레이어 선택

```

Playing rtmp://bc-source.liveplay.
libavformat version 57.58.101 (internal)
libavformat file format detected.
[flv @ 0196c160]video stream discovered after head already parsed
[flv @ 0196c160]audio stream discovered after head already parsed
[lavf] stream 0: video (h264), -vid 0
[lavf] stream 1: audio (aac), -aid 0
VIDEO: [h264] 1920x1080 0bpp 25.000 fps 5000.0 kbps (610.4 kbyte/s)
<vo_direct>:ERROR>Your card doesn't support overlay
=====
Opening video decoder: [ffmpeg] FFmpeg's libavcodec codec family
libavcodec version 57.66.109 (internal)
Selected video codec: [ffh264] vfm: ffmpeg (FFmpeg H.264)
=====
Clip info:
author:
copyright:
description:
keywords:
rating:
title:
presetname: Custom
creationdate: Fri Aug 31 12:04:45 2018
videodevice: VhVideoCustom
avclevel: 40
avcprofile: 100
audiodevice: VhAudioCustom
audiochannels: 2
audioinputvolume: 100
aid: 3vb/10EzZ57XT0tUhxwog==

```

그림 5. MPlayer 동작 화면

에 대한 고민을 덜었다. 동영상 플레이어 오픈소스인 MPlayer에 옵션을 달아 키보드의 윈도우 키를 제외한 모든 키 조작과 마우스 조작이 불가하게 만들었다. 또한 연결이 끊어지거나 프로그램이 다운돼도 계속 연결을 시도하는 Batch 파일을 만들었다.

네트워크 또한 매우 중요한 이슈였다. Distribution[그림 1]에서 볼 수 있듯이, 애초에 RTMP Server로 전달하는 Feed 자체가 하나이므로 이쪽 네트워크가 무너지면 끝나는 일이지만 이 경우를 제외한 모든 네트워크 문제가 일어나는 걸 방지하기 위해 많은 걸 준비했다.

Live Stream Configure	
Source Signal	1080i 50
Resolution	1920×1080
Frame rate	25fps
Video bitrate	5000kbps
Video encoding codec	H.264
Mode	CBR
Audio encoding codec	AAC LC
Audio sample rate	48KHz
Audio bitrate	192kbps
Format	Stereo
De-interlace or Interlace	De-interlace

표 1. AESF 제공 e-Sports Summary Draft

먼저 전용회선의 서비스 용량을 늘려 안정적인 수신을 도모했고, 선배들의 아이디어로 사내 OA망을 이용한 수신, LTE Decoder를 이용한 수신 등 RTMP Feed를 받을 방법들과 모든 인프라를 동원했다. 그런데도 Feed 수신 시, Frame Drop 현상이나 연결이 아예 되지 않는 경우가 매우 많아 Live 가능 여부를 판단하기가 쉽지 않았다.

자세하게 나와 있지 않았지만 AESF가 제공한 영상 스펙은 Video의 Bitrate는 5000kbps에 Codec은 H.264, Resolution은 1920×1080(50i) 등의 정보였다. Codec의 세부 정보(Profile, Level 등)가 없었고, 동일 Codec을 사용하는 SBS SNG의 경우 이보다 높은 Bitrate로 송출을 하기 때문에 우려가 있었다. 검색을 해보니 Youtube의 Live의 경우 1080P 60fps로 방송을 하려면 H.264 Level 4.2로 인코더 설정할 것을 권장하며 이때 Bitrate는 4500~9000kbps라고 하니 화질에 대해 큰 문제는 없다고 판단했다. 아니, 문제가 꼭 없었어야만 했다.

협업이라 쓰고 신선함이라 부른다



그림 6. SBS 중계방송 화면

이번 협업을 진행했던 아프리카TV 엔지니어들은 비슷한 중계방송 경험이 풍부하기 때문에 도움을 얻을 수 있었다. Frame Drop 등의 이슈가 없을 만한 적절한 사양의 워크스테이션, 멀티 기능 Scan Converter(Analog Way 社, BHD930)를 지원받았다. 게임 내 중요 정보가 화면 상단, 측면 등 곳곳에 배치되어있기 때문에 화면 Resizing이 필요했다. 기 경험이 있는 아프리카TV 측으로부터 적정 값을 전달 받고, Scan Converter에서 설정했다.

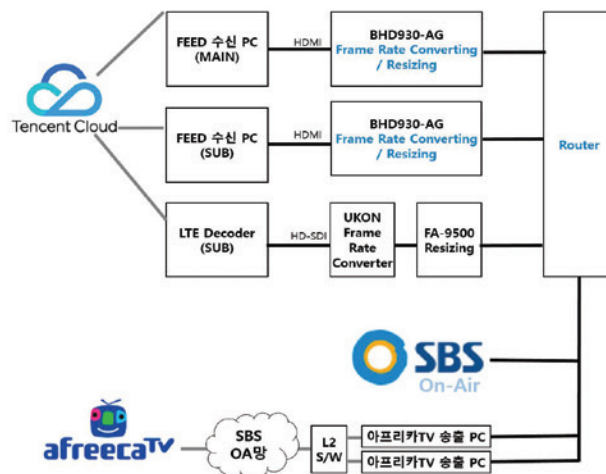


그림 7. From Tencent Cloud to Transmission 계통도



그림 8. 아프리카tv 앱 내 화면

보통 아프리카TV의 경우 중계방송 진행 시, WOWZA 스트리밍 솔루션을 사용한다고 한다. 이번의 경우에는 제공 피드를 직접 당겨 받기 때문에 팟플레이어를 사용했다. [그림 2]에서 Main PC는 MPlayer, 예비는 팟플레이어를, 2nd backup은 LTE Decoder를 활용했다. Feed를 수신하는 서버, 중계 부조와 통신할 수 있는 Intercom 등 간이 중계시스템을 필자가 근무하는 NQC 복도 한 칸에 구축했다. 한편 아프리카TV에서는 중계 부조에서 범퍼 역할을 하는 LSM 클립들 외에 자체 브릿지 영상을 준비하여 방송사고에 대비했다.

애초 29일 결승전 중계방송 예정이었는데 예선전 방송 편성이 추가되면서 27일 생방송을 진행했다. 이전까지 Feed가 제대로 들어오지 않아서 걱정했는데, 예상보다는 문제가 덜 발생해서 무사히 마칠 수 있었다. 아프리카TV 공식방송 접속자수가 약 3만 명이었고 결승전은 5만 명에 육박했다. 앱 내 공식방송의 중계방송까지 합치면 10만 명 가까이 시청한 거로 추측된다.

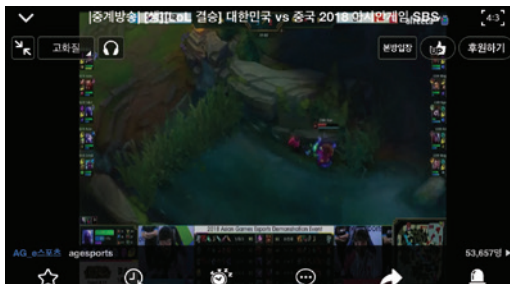


그림 9. 아프리카tv 앱 내 화면

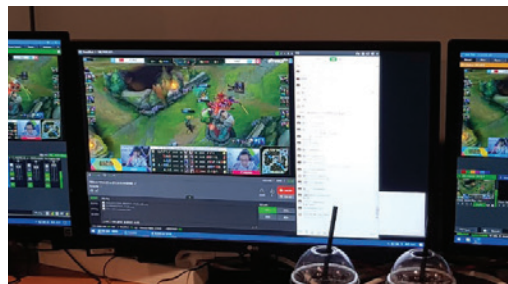


그림 10. 아프리카tv 송출 화면



그림 11. 중계 시스템



그림 12. SBS 방송 송출 화면



마치며

지상파 SBS와 개인 방송 플랫폼 대표주자라 할 수 있는 아프리카TV의 합작 방송을 준비하며 필자는 루이비통과 슈프림의 콜라보레이션과 같은 신선함을 느꼈다. 기존 중계방송의 익숙함에 새로움을 얻으니 신선도 100% 과즙 주스가 된 기분이다. 물론 준비 과정이 호락호락하지 않았지만 늘 그렇듯이 끝나고 보니 소중한 경험이었음을 느낀다. 4년 뒤 다음 항저우 아시안게임에서는 정식 종목으로서 e-Sports를 만나게 된다. Distribution 방식이 변경될 수 있겠지만 다음에는 더 나은 품질을 시청자에게 전달할 수 있도록 노력하겠다.