



Loudness Logging System

개발기

글. 류덕형 SBS 미디어IT팀 매니저

안녕하세요. '방송과기술' 독자 여러분, 평소 즐겨보던 잡지에 제가 글을 쓴다는 게 신기하게 느껴집니다. 방송국에 다니다 보면 대단해 보이지는 않는 장비가 상상을 뛰어넘는 가격이어서 흠칫 놀랄 때가 종종 있습니다. 오늘은 이러한 장비를 정확히 이해해 보고자 우연치 않게 직접 개발하게 된, 그리고 이로 인해 결과적으로 수천만 원의 비용을 아낀 SBS의 Loudness Logging System 개발 스토리를 소개해 드리려고 합니다.

Loudness Logging System이 무엇인가?

오디오는 시청 경험에 지대한 영향을 미치는데, 우리는 본능적으로 채널을 돌리다 큰 소리가 나오면 해당 채널의 오디오가 좋다고 생각하게 됩니다. 그래서 방송사들이 경쟁적으로 소리 크기를 키우는 Loudness War 현상이 생기게 되는 이유가 되었죠. 이렇게 되면 결국 오디오의 전체적인 품질이 저하되는 문제가 발생하고, 채널마다 소리 크기가 다르면 시청자들이 채널을 돌릴 때마다 깜짝 놀라게 되는 문제가 있어 과학기술정보통신부는 국내 모든 방송 프로그램의 Loudness를 -24 LKFS(± 2 dB)로 제한하는 법을 발표하게 됩니다.

법을 자세히 살펴보면 SBS는 방송법 제70조의2제1항에 따라 방송프로그램 표준 음량기준(-24 LKFS ± 2 dB)을 준수하여야 하고 제2항 및 동법 시행령 제68조에 따라 송신된 방송프로그램의 측정, 기록, 관리해 중앙전파관리소에 음량 측정자료를 제출하여야 합니다. 전파관리과의 서면 조사 시 방송프로그램 목록(송출 편성자료), M-LKFS, I-LKFS 측정값을 요청할 수 있고 표준 음량기준 위반 시 방송법 제69조에 따라 과태료가 부과될 수 있기에, SBS는 주조정실 최종 송출 인코더 전단에서 Loudness Logger를 설치해 운용하고 있습니다.

어쩌다 직접 개발을 시작하게 되었나

해당 규제가 시작된 2016년부터(14년 11월 제정, 18개월의 유예기간으로 실제 16년 5월부터 시작) SBS는 외산제품인 Media Proxy 사의 제품을 사용해 로깅과 보고서 제출에 대응했으나 제품 지원 기간이 끝나가 작년에 새로 구매해야 할 상황이었습니다.

당시 저는 Loudness와는 전혀 상관없이, 회사 디지털T/F로 방송장비구매를 담당하는 선배와 함께 유튜브 제작을 돕기 위한 RTMP Multiview 제품을 알아보다 Stirlitz 사의 IP Multiviewer 제품을 구매하게 되었습니다. 그런데 마침 동사에 서 Media Logger라는 제품이 나오고 있었고 여기에 Loudness를 기록하는 기능도 있어 영업 담당자분들과 미팅을 잡고, 우리에게 필요한 사항들(편성표 기반 프로그램 I-LKFS값 계산, 기존 리포트 형태로 문서생성)을 반영해서 구매할 수 있는지 문의했습니다. 그런데 답변이 몇 주가 지나도 돌아오지 않아서 우리측의 요구사항들을 다시 읽어보다 보니 ‘이 게 그렇게 어려운 기술인가? 나도 만들 수 있을 것 같은데?’ 하는 근거 없는 건방진 생각이 생겨 일반 PC에서 코드를 짜 봤고, 가능성이 보여 제품 서칭을 멈추고, 본격적으로 개발에 착수하게 되었습니다.

Loudness란 무엇이며 어떻게 측정하는가?

이미 방송과기술에 여러 번 소개되었기에(저 또한 옛날 방송과기술을 많이 참고하며 공부했습니다. 감사합니다!) 간략 하게 이야기하면 흔히 아는 dB가 소리(음압)의 절대적인 파워를 나타낸다면 Loudness는 인간이 주관적으로 느끼는 음 량의 정도를 나타내는 개념입니다. 인간의 주관적인 느낌을 계산에 반영하기 위해 특수한 계산을 해야 하는데 대한민국 에서는 방송법에 명시된 TTA 방송 음량 레벨 운용 기준에서 ITU-R BS.1770-3을 측정 방식으로 하고 있기에 해당 국제 표준을 따르면 됩니다.

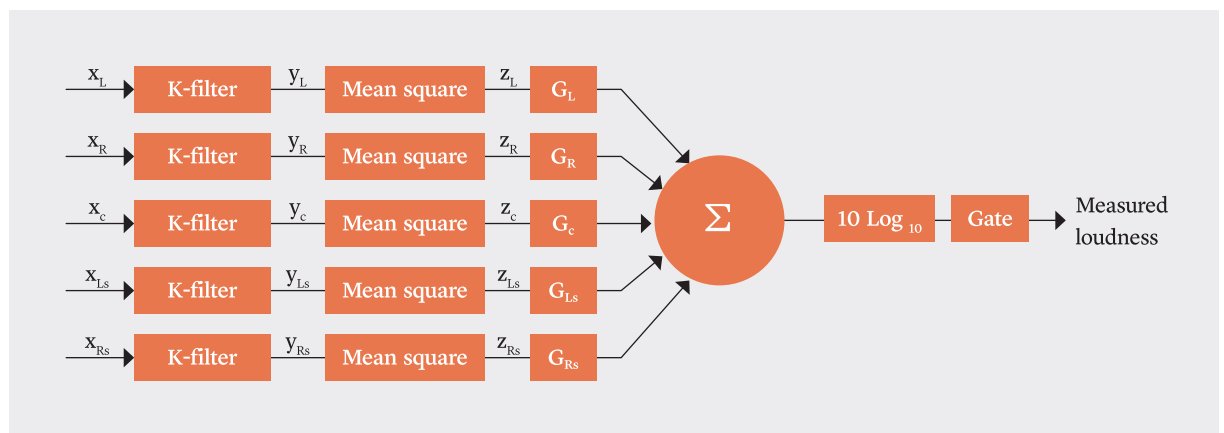


그림 1. LKFS 측정 과정에 대한 간단한 블록 다이어그램

BS.1770에서 Loudness는 크게 네 가지 과정을 거쳐 계산합니다. 400ms duration 음압값에 ① 지각적 주파수 가중치를 적용하는 K-frequency weighting ② 채널별 평균 제곱 계산 ③ 채널 가중치 합산(스테레오는 가중치가 1이라 단순 합), 이를 시간 축에서 75%가 겹치게 하는 다음 400ms 블록, 즉 100ms 다음 400ms 블록을 반복해서 측정한 후 ④ 게이팅이라는 과정을 거칩니다.

전파관리소에는 프로그램 I-LKFS와 M-LKFS값을 제출해야 하는데 M(momentary)값은 ①~③ 과정을 거친 값 즉 0.1 초마다 하나씩 나오는 값을 의미하고 I(Integrated)는 프로그램 시간 단위로 ④ 게이팅을 거친 값입니다. 게이팅은 Absolute Threshold Gating과 Relative Threshold Gating 단계로 나뉘어서 우선 -70 LKFS를 이하의 블록은 계산에서 제외하고, 여기서 1차 평균을 구한 뒤 이 평균보다 10dB 이상 작은 블록들은 또 제외해 2차 평균을 구해 최종 I-LKFS를 계산해 냅니다. 즉 M-LKFS값만 원본 그대로 가지고있으면 I-LKFS값은 언제나 프로그램 편성시간을 참고해서 계산해 낼 수 있는 것이죠.

시스템 설계의 세부사항

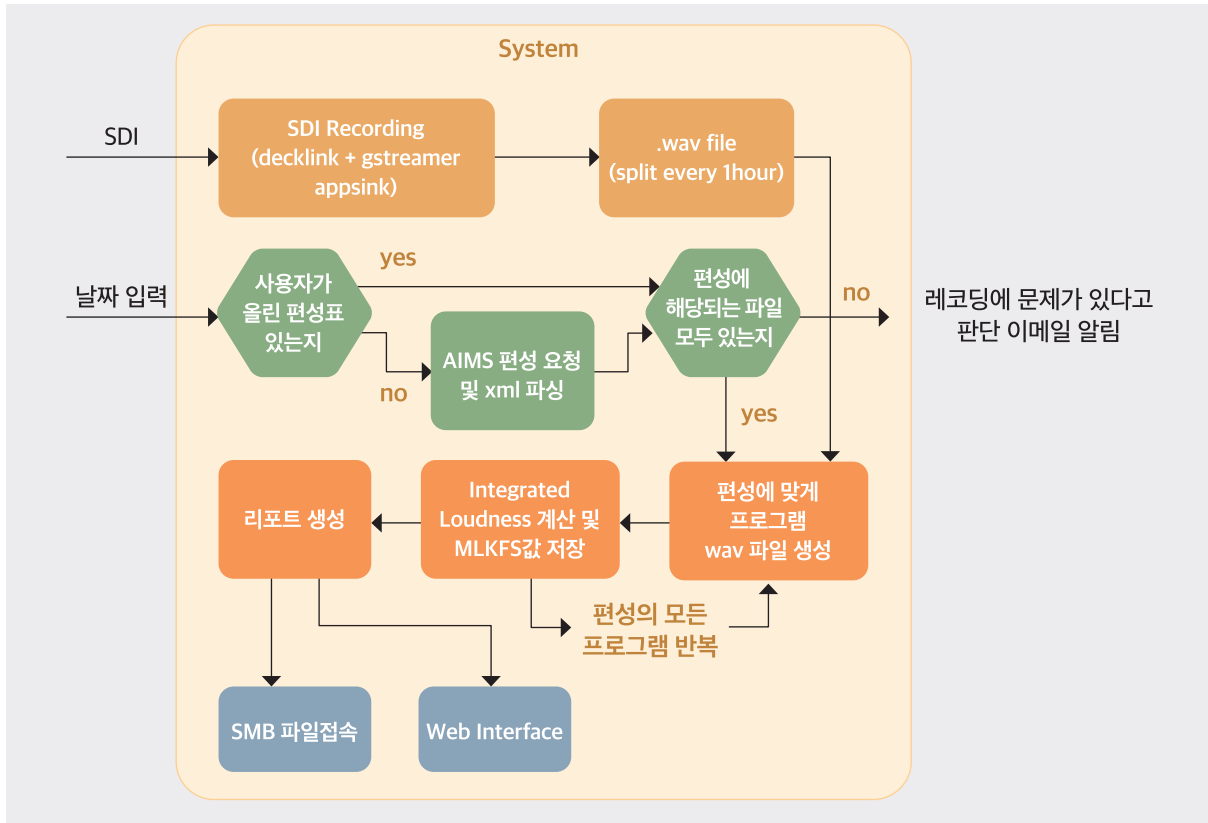


그림 2. Loudness Logging System의 블록 다이어그램

저는 전용 하드웨어를 만들 능력까지는 없어서 일반 PC에서 Decklink 카드를 꽂아 신호를 받고 처리하는 방식으로 접근했습니다. 소프트웨어적으로 Loudness Logging System은 크게 두 개의 시스템으로 이루어집니다. 하나는 최종 송출 소스인 주조정실 PGM 방송신호(SDI)를 24시간 무손실로 녹음하는 레코딩 시스템이고, 다른 하나는 녹음본과 방송 편성을 가지고 프로그램 통합 라우드니스(I-LKFS)를 계산하고 리포트를 생성하는 라우드니스 리포팅 시스템입니다.

레코더 파트

정확한 Loudness 값 측정을 위해서는 송출단에 나가는 24bit 48kHz 오디오를 손실 없이 안정적으로 녹음하는 것이 가장 중요합니다. 초당 48,000개의 오디오 값을 놓치지 않고 기록해야 한다는 의미인데요, 이를 위해 최종 송출 인코더 전단인 주조정실 PGM SDI 신호를 Decklink 카드로 받았고 G-streamer라는 멀티미디어 프레임워크의 Appsink 플러그인을 통해 분리된 오디오에 접근해 SDI의 8채널 오디오를 무손실로 녹음했습니다. 또 나중에 이 파일을 다룰 때 메모리가 필요 이상으로 무거워지지 않도록 1시간 단위로 나누어 wav 파일로 저장하는데, 이것이 이 시스템의 기본 녹음파일 형식이 됩니다.

Loudness 측정 및 리포팅 파트

우선 라우드니스 계산에는 PyLoudNorm이라는 오픈소스를 수정해서 사용했습니다. 해당 라이브러리는 ITU-R BS.1770-4를 구현했는데 이를 사용해도 되는 것인지 알아보니, 방송법에 명시된 TTA 방송 음량 레벨 운용기준 측정방

식인 1770-3과 Stereo 채널 LKFS 계산 시 차이가 없기 때문에 문제될 것이 없었습니다. 이는 ITU의 문서를 보면 알 수 있었는데 1770-4는 3에서 Annex 3가 추가되어 많은 입력 채널에서의 계산법을 안 내할 뿐, 스테레오(뿐 아니라 주요 5개 채널까지) 측정에 있어서는 동일한 Annex 1을 따르게 되어있어 3과 4의 차이가 없음을 알 수 있었습니다. 또한 실제로 ITU-R BS.2217 자료를 통해 PyLoudNorm 알고리즘의 Compliance 여부를 확인하니 문제가 없음을 테스트할 수 있었습니다. 다만 기본 PyLoudNorm 라이브러리는 전파관리소 제출에 필요한 계산근거 M-LKFS값, 즉 게이팅 전 모든 블록의 M-LKFS값을 따로 저장할 수 있는 기능이 없어 이 부분이 가능하도록 수정해서 사용했습니다.

File	Target	Implementation							
		pyloudnorm		loudness.py	ffmpeg	libebur128	Essentia	Audition	youlean
		Default	De Man						
FrequencySweep	-18.0	-18.03	-17.99	-17.99	-18.00	-18.00	-18.18	-18.03	-18.02
25Hz_2ch	-23.0	-23.00	-22.99	-22.99	-23.10	-23.00	-26.37	-23.04	-23.02
100Hz_2ch	-23.0	-23.03	-22.99	-22.99	-23.10	-23.00	-22.86	-23.04	-23.02
500Hz_2ch	-23.0	-23.04	-22.99	-22.99	-23.10	-23.00	-22.99	-23.04	-23.02
1000Hz_2ch	-23.0	-23.03	-22.99	-22.99	-23.10	-23.00	-23.00	-23.04	-23.02
2000Hz_2ch	-23.0	-23.03	-22.99	-22.99	-23.10	-23.00	-23.00	-23.04	-23.02
10000Hz_2ch	-23.0	-23.04	-22.99	-22.99	-23.10	-23.00	-23.00	-23.04	-23.02
25Hz_2ch	-24.0	-24.00	-23.99	-23.99	-24.10	-24.00	-27.21	-24.04	-24.02
100Hz_2ch	-24.0	-24.03	-23.99	-23.99	-24.10	-24.00	-23.92	-24.04	-24.02
500Hz_2ch	-24.0	-24.04	-23.99	-23.99	-24.10	-24.00	-23.99	-24.04	-24.02
1000Hz_2ch	-24.0	-24.04	-23.99	-23.99	-24.10	-24.00	-24.00	-24.04	-24.02
2000Hz_2ch	-24.0	-24.04	-23.99	-23.99	-24.10	-24.00	-24.00	-24.04	-24.02
10000Hz_2ch	-24.0	-24.04	-23.99	-23.99	-24.10	-24.00	-24.00	-24.04	-24.02
RelGateTest	-10.0	-10.07	-10.03	-10.03	-9.60	-10.00	-10.03	-10.07	-10.15
AbsGateTest	-69.5	-69.49	-69.45	-71.46	-69.50	-69.50	-69.45	-69.49	-69.55
Mono.Voice+Music	-23.0	-23.03	-22.99	-22.99	-23.10	-23.00	-22.97	-23.03	-22.98
Mono.Voice+Music	-24.0	-24.03	-23.99	-23.99	-24.10	-24.00	-23.97	-24.04	-23.98
Stereo.VinL+R	-23.0	-23.03	-22.98	-22.98	-23.10	-23.00	-22.97	-23.02	-22.99
Stereo.VinL+R	-24.0	-24.02	-23.98	-23.98	-24.10	-24.00	-23.97	-24.02	-23.98

그림 3. 주요 알고리즘의 BS.2217 LKFS 측정치 비교

방송 스케줄은 SBS의 주조 송출시스템인 AIMS에 API로 요청해 하루치 전일 편성정보를 받아오도록 했습니다. 일일보고서를 만들려면 하루 간의 정확한 편성정보를 알아야 하는데 이걸 그 하루 방송이 다 끝나야 정확하게 확정되기 때문입니다. 일일 편성이 끝나는 시간은 정확히 정해져 있지 않고, 올림픽중계와 같은 방송을 할 때는 새벽 7시까지도 가능경우가 있어서 확실하게 전일 편성이 끝나있을 매일 아침 8시에 전일 편성에 대해 I-LKFS를 일괄적으로 계산하고 전파관리소 보고서 양식에 맞추어 문서를 자동으로 생성하도록 했습니다.

▲	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
22	1024:12	1024:22	00:00:10	-22.2749	5. 이어서													
23	1024:22	1028:15	00:03:53	-22.8801	[예고] 금토드라마 (우리 영화) 하이라이트(평일)(6.15~7.19)-203초[[예고] 보석이네 건강수다(당일)(6.16(월))-30초													
24	1028:15	1028:30	00:00:15	-22.8474	6. ID 생활의달인													
25	1028:30	1053:24	00:24:54	-23.1519	예니겔리(25년06월16일(월))	P20160429002												
26	1053:24	1054:40	00:01:16	-23.003	6. 이어서													
27	1054:40	1058:06	00:03:26	-23.044	[예고] 금토드라마 (우리 영화)(3회)(평일)(6.15~6.18)-36초[[예고] 토크나면(평일)(6.12~6.16)-58초[[예고] 신발 벗고 돌싱포맨(평일)(6.13~6.16)-75초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
28	1058:06	1058:21	00:00:15	-23.5067	7. ID 사계의봄													
29	1058:21	1149:52	00:51:31	-23.1541	보석이네 건강수다(25년06월16일(월))(UHD)	P20250430001												
30	1149:52	1151:23	00:01:31	-23.0798	7. 이어서													
31	1151:23	1153:13	00:01:50	-23.0016	[예고] 금토드라마 (우리 영화)(3회)(평일)(6.15~6.18)-36초[[예고] 생활의 달인(당일)(6.16(월))-37초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
32	1153:13	1153:28	00:00:15	-23.8084	8. ID 토크나면													
33	1153:28	1238:49	00:45:21	-22.7883	SBS 12뉴스(생방VCR)(25년06월16일(월))	P20130910005												
34	1238:49	1238:59	00:00:10	-22.3018	8. 이어서													
35	1238:59	1243:02	00:04:03	-23.0455	[합창캠페인] 우유자조금관리위원회-맛있다 우리우유(4.1~)40초[[예고] 금토드라마 (우리 영화) 하이라이트(평일)(6.15~7.19)-203초													
36	1243:02	1243:17	00:00:15	-21.6995	9. ID CH SUMMER													
37	1243:17	1255:35	00:12:18	-22.8894	미니미니 미니나(25년06월16일(월))(UHD)	P20250527003												
38	1255:35	1256:51	00:01:16	-23.041	9. 이어서													
39	1256:51	1257:58	00:01:07	-22.9396	[예고] 토크나면(평일)(6.12~6.16)-30초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
40	1257:58	1258:13	00:00:15	-23.7121	10. ID 돌싱포맨													
41	1258:13	1351:58	00:53:45	-23.0243	김원희의 원더랜드(재)(25년06월16일(월))(해설)(UHD)	P20241118001												
42	1351:58	1353:39	00:01:41	-22.8888	10. 이어서													
43	1353:39	1354:19	00:00:40	-23.3861	[합창캠페인] 우유자조금관리위원회-맛있다 우리우유(4.1~)4(S20080612003													
44	1354:19	1354:34	00:00:15	-22.7584	11. ID 동상이몽2													
45	1354:34	1548:05	01:53:31	-23.4391	뉴스브리핑(생방VCR)(25년06월16일(월))(UHD)	P20170317009												
46	1548:05	1549:36	00:01:31	-23.013	11. 이어서													
47	1549:36	1552:20	00:02:44	-22.9735	[예고] 금토드라마 (우리 영화)(3회)(평일)(6.15~6.18)-36초[[예고] 신발 벗고 돌싱포맨(평일)(6.13~6.16)-54초[[예고] 생활의 달인(당일)(6.16(월))-37초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
48	1552:20	1552:35	00:00:15	-23.646	12. ID 사계의봄													
49	1552:35	1650:59	00:58:24	-23.0393	출연자침 스페셜(25년06월16일(월))(UHD)	P20200702002												
50	1650:59	1651:40	00:00:41	-22.6803	12. 이어서													
51	1651:40	1653:33	00:01:53	-22.9665	[합창캠페인] 우유자조금관리위원회-맛있다 우리우유(4.1~)40초[[예고] 금토드라마 (우리 영화)(3회)(평일)(6.15~6.18)-36초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
52	1653:33	1653:48	00:00:15	-23.9554	13. ID 사계의 봄													
53	1653:48	1744:26	00:50:38	-22.9804	SBS 오뉴스(생방VCR)(25년06월16일(월))(UHD)	P20161230003												
54	1744:26	1745:07	00:00:41	-23.0284	13. 이어서													
55	1745:07	1747:23	00:02:16	-22.9582	[예고] 사계의 봄(평일)(6.12~6.16)-32초[[예고] 토크나면(평일)(6.12~6.16)-30초[[예고] 생활의 달인(당일)(6.16(월))-37초[[예고] 동상이몽2 너는 내 운명(당일)(6.16(월))-37초													
56	1747:23	1747:38	00:00:15	-23.6439	14. ID 골때리는너를													
57	1747:38	1841:46	00:54:08	-22.9472	열린 TV 시청자 세상(25년06월16일(월))	P20180111002												

그림 4. 생성된 보고서

접근성 개선

새롭게 시스템을 만드는 만큼 단순 기존 제품 대체가 아닌 사용자 편의성 면을 개선하려고 노력했습니다. 기존에는 보고서 요청이 있을 때마다 담당자가 직접 장비에 가서 보고서를 가져오는 불편함이 있어, 사내망에서 장비의 IP를 알면 일일보고서, 녹음파일, M-LKFS값 등 주요 파일들에 대해 SMB로 접속해 원격으로 가져올 수 있게 했습니다. 또한 flask를 이용해 웹 인터페이스도 만들어 같은 작업을 웹 브라우저에서도 할 수 있게 했습니다. 자유도가 높은 웹 인터페이스에서는 추가로 편성정보를 브라우저상에서 직접 수정하고 라우드니스 재계산 및 보고서 재생성 등의 작업을 할 수 있게 해 사용자가 기계실에 가지 않아도 되게끔 만들었습니다.

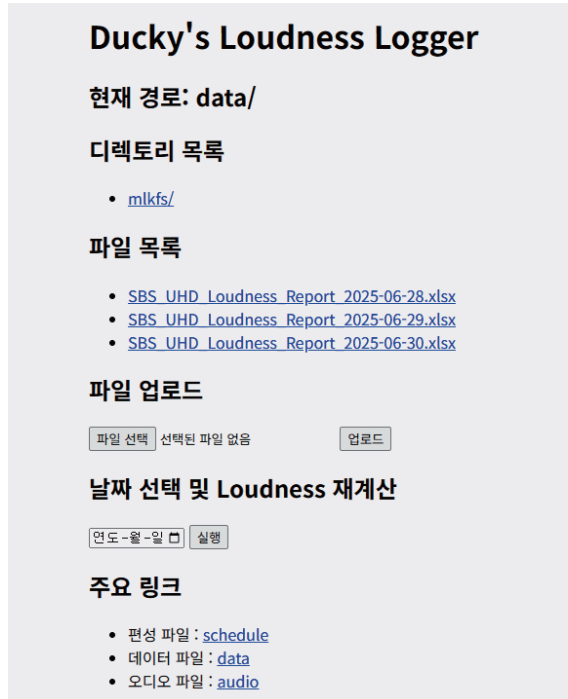


그림 5. 웹 인터페이스 기본페이지

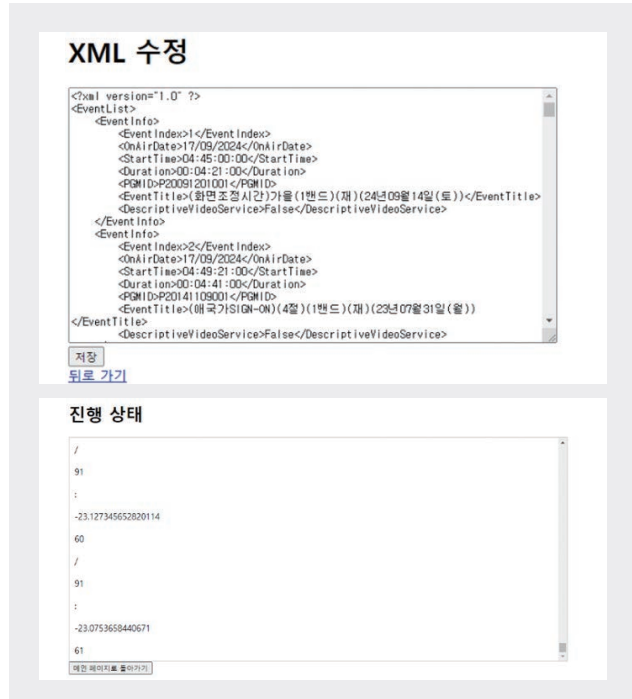


그림 6.1, 6.2. 편성수정과 라우드니스 재계산 웹 인터페이스

하드웨어 구성과 OS

하드웨어는 비용 절감을 위해 서버형 조립컴퓨터를 맞췄습니다. GPU 연산작업이 없으니 그래픽카드는 전혀 중요하지 않았고, 24시간 365일 문제없이 안정적으로 켜져있을 수 있도록 파워나 쿨러에 가장 신경을 썼으며 프로그램 I-LKFS를 계산할 때 스포츠 편성 같은 경우는 4~5시간 이상 긴 편성이 나올 수도 있어 대량의 음압값이 충분히 메모리에 올라갈 수 있게 램 용량에 힘을 썼습니다. 현행법상 최소 90일 치 이상은 원본을 유지할 수 있어야 하기에 필요한 파일들의 크기를 러프하게 계산해 OS 디스크 외에 16TB의 하드를 추가했습니다. CPU는 Decklink와의 호환성을 위해 인텔을 선택했고 계산속도를 위해 클럭수를 포기하고 싶지 않다 보니 말도 많고, 탈도 많은 14세대 i9을 쓰게 되었습니다.

개발했던 소프트웨어는 파이썬 기반 프로그램이기 때문에 OS는 리눅스나 윈도우나 전혀 상관이 없는데(윈도우에서 g-streamer를 쓰려면 msys-utrc를 써야하는 약간의 불편이 있기는 합니다), 개발 당시에는 14세대 i9이 윈도우 11에서 만 그나마 안정화된 모습을 보여 윈도우로 구성하게 되었습니다. 이렇게 구성된 하드웨어에서 24시간 3개 이상의 SDI 소스 녹음을 진행하면서 일일편성 프로그램별 I-LKFS를 한 번에 계산하고 보고서를 생성해 보니 채널별로 총 10분이 넘는 시간이 걸렸습니다.

안정적인 운영의 성공 및 소감

이 시스템은 작년 말에 설치해서 한 달 정도 기존 장비와 병행운영하며 안정성 테스트를 거치고 올해부터 SBS의 메인 Loudness Logger로써 성공적으로 운용되고 있습니다. 이 글을 쓰고 있는 7월까지도 문제없이 잘 사용되고 있는 모습을 보니 마음에 깊은 안도감과 뿌듯함이 듭니다.

개발을 시작할 때만 해도 코드 몇 줄 짜보며 소프트웨어적으로 구현 가능성만 대충 테스트해 보고 약간의 객기로 쉽게 생각하고 덤벼들었습니다. 그런데 막상 개발에 본격적으로 들어간 뒤 정말 많은 공수를 쏟은 건 오히려 사소하다고 여겼던 일들이었습니다. 현행법과 국제 표준을 확인하며 적합성을 따지고, 사용자 인터페이스를 만들고 수정하며, 서버형 조립컴퓨터 견적내는 모든 작은 디테일 하나하나에 깊은 고민과 노력이 필요했습니다. 제가 쉽게 생각하고 무시했던 제품들도 실사용 수준의 제품이 되려면 이런 작은 디테일들을 굉장히 많이 신경 썼다는 것을 몸소 느낄 수 있었습니다.

개인적으로 저는 과거에 오디오 컴퓨팅을 좋아하는 학생이었는데, 학부를 졸업하고 관련 대학원 과정을 찾아보다 유학을 가야겠다고 결심했었습니다. 최종 합격을 하고 비자까지 나온 상황에서 당시 커리어에 대한 막연한 두려움으로 마지막에 비행기를 타지 않았었는데, 이번 개발을 하며 PyLoudNorm에 대해 조사하다 보니 해당 라이브러리를 만든 사람이 해당 학교 해당 과정을 나온 사람이었다는 것을 알고 정말 깜짝 놀랐습니다. 취직을 하면서 제 관심 분야와 이제는 멀어진다고 생각했는데 계속 해당 분야와 접점이 생기는 게 개인적으로 느낌이 참 묘하고 흥미로웠습니다.

업무와 개발을 병행하면서 중간중간 솔직한 마음으로 ‘아 그냥 기존 제품 구매하게 둘 걸...’하는 생각(후회...?)도 많이 했는데, 뭔가 막히고 어려울 때마다 같이 머리 싸매며 고민해 주고 위로도 해주던 선배님들 덕분에 잘 마무리할 수 있었던 것 같습니다. 이런 선배들에게 정말 감사하고 앞으로 저 또한 회사의 크고 작은 일들의 기술적 해결을 같이 머리 싸매고 고민하는 동료가 되겠습니다. 🙏

