

KBS 청주방송총국 기술국 자생적 프로젝트팀, TIMS를 소개합니다 - 2

웹기술이 가져올 방송기술의 새로운 패러다임

글. 권영부 KBS 청주방송총국 기술국 기술감독

지난 호에서 소개해 드렸던 KBS 청주방송총국 기술국의 자생적인 프로젝트팀, TIMS의 구체적인 성과에 대해서 이번 호부터 몇 차례에 걸쳐 상세히 소개해드리고자 합니다.

2021년은 다양한 신기술의 등장으로 새로운 패러다임을 기대하는 사람들로 하여금 그 열기가 아주 뜨거웠던 한해다. 그 주인공들은 블록체인 기반의 NFT, 메타버스, AI 등으로 SNS, 뉴스에서 한 번쯤은 들어보았을 것이다. 본고 역시 미래 혁신을 이끌어갈 새로운 기술을 소개하는 글이라 생각했다면 그건 아니다. 오히려 등장한 지는 오래되고 아주 무르익을 대로 묵은, 그럼에도 꾸준히 발전하고 있는 웹기술, 바로 HTML5를 소개한다.

HTML5는 2014년 표준화되고 꾸준히 버전업되면서 현재는 광의적인 의미로 최신 웹 기술(HTML Living Standard, CSS3+, ECMAScript 6+ 등) 등을 통틀어 칭하기도 한다. HTML5 웹기술을 설명함에 있어 블록체인, 메타버스를 언급한 이유는 방송사가 최신 기술 트렌드에 꾸준히 관심을 가지는 것도 좋지만, 이미 BBC를 포함한 유럽방송사에서 웹기술에 근간한 조직 구조로 변화하기 위해 전사 차원에서 관련 분야의 인력확충, 기술개발에 많은 투자를 하고 있기 때문이다. 즉 이미 패러다임은 시작했으며 국내 방송사 역시 혁신을 이루기 위한 과감한 결정과 투자가 필요한 때가 아닐까 싶다. 이 글은 2021년 한 해 동안 웹기술과 함께했던 필자의 경험담과 웹기술을 활용한 방송제작기술 발전 방향을 소개한다.

웹기반 CG 제작시스템과의 첫 조우

KBS에서는 모바일 퀴즈쇼 잼라이브(JAM Live)를 서비스하는 회사(네이버 자회사 스노우)와 협업하여 예능 프로그램인 꿀잼퀴즈방을 제작한 바 있다. 이는 모바일에서만 즐겼던 퀴즈쇼를 지상파에서도 동시 시청할 수 있도록 한 것이다. TV-모바일 동시 방송 플랫폼이라는 방송기술에는 전례가 없는 시도였고 구축과정에 여러 기술적 난관에 부딪히게 된다. 이때 구원투수로 vMix를 이용한 웹기반 CG 제작시스템이 등장하게 된다.



그림 1. 알파값이 들어간 채팅창

제작 당시 프로그램 제작진은 모바일 플랫폼에서 경험했던 CG가 방송화면에 똑같이 구현될 것을 요구했다. 이는 기존 방송 CG 시스템으로는 구현이 불가능한 점이 많았다. 먼저 시청자와의 소통수단인 채팅창이다. 채팅은 짧은 시간 안에 서버로부터 소켓 프로토콜을 통해 수많은 텍스트 데이터를 실시간으로 주고받는데 이를 CG로 구현하는 과정은 전통적인 CG 장비로는 불가능했다.

또 다른 요구사항으로는 문제의 정답이 발표되면 시청자들의 정답/오답 선택률을 막대 그래프로 표시되도록 하는 것이다. 기존 CG 시스템에서는 성능상의 이슈로 CG 파일이 사전에 프리렌더 되어있어야 하는데 사용자 데이터가 없는 상황에서 막대 그래프 CG로 구현한다는 것은 애초에 불가능한 것이었다. 또 다른 문제로 정답발표 CG를 미리 생성해놓아야 하기에 제작진들 사이에서 정답이 유출되지 않도록 보안에도 신경 써야 했다. 종합적으로 보면 실시간으로 CG를 생성해야 하는 요건을 기존 CG 시스템으로 충족하기가 어려웠다.

그러나 vMix 기반의 웹기반 CG 제작시스템에서는 모든 게 해결된다. 당연하지만 모바일에서 즐겼던 웹기반의 CG를 방송화면에 그대로 적용하면 되니까 말이다. (당시에는 vMix와 기존 CG 시스템을 같이 쓰는 하이브리드 방식으로 제작함). 당시의 경험은 3년 뒤 HTML 기반의 CG 생성 시스템을 직접 개발하게 되는 결과로 이어진다.

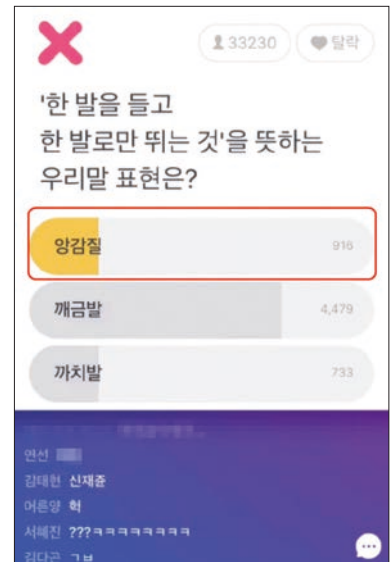


그림 2. 모바일 앱라이브에서 선택지에 정답/오답 선택률 표시

HTML 기반 CG 시스템을 직접 개발해보기

2021년 'NodeCG를 활용하여 HTML 기반 CG 그래픽 제작'이라는 프로젝트로 KBS 사내 기술공모전에서 입상한다. 프로젝트 내용은 HTML 웹페이지를 방송에 사용할 수 있는 방송 CG로 구현한다는 것이다. 이를 구현하기 위해 방송영상송출 S/W인 vMix와 Nodejs 기반의 NodeCG 프레임워크를 사용한다. NodeCG는 간단히 설명하면 웹서버이다. 일반적인 웹서버와의 차이점은 웹페이지의 그래픽 요소를 쉽게 조작할 수 있는 소켓 통신 기반의 API를 제공한다는 것이다.

요약하면 NodeCG는 그래픽 요소가 있는 웹페이지를 서비스하는 역할을 하고, vMix¹⁾에서 해당 웹페이지를 불러와서 웹페이지의 그래픽 요소를 방송 CG 장비처럼 Key/Fill SDI 신호를 출력하도록 한다.

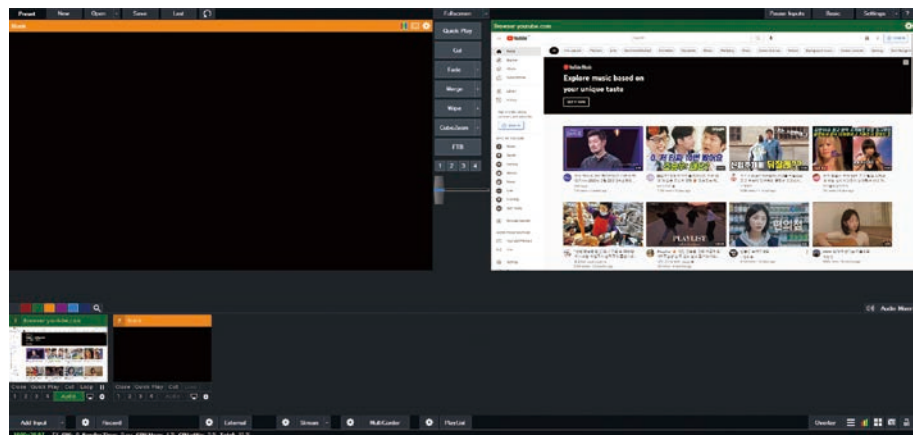


그림 3. vMix에서 Youtube 사이트를 입력소스로 추가한 모습

여기까지는 어렵지 않은데 그렇다면 HTML을 어떻게 CG로 구현할 수 있을까. 트위치나 아프리카TV 스트리밍 방송을 본 적이 있다면 화면상에 움직이는 CG가 합성된 것을 보았을 텐데 이와 동일한 원리이다. 바로 CEF의 HTML off-screen rendering 기능이다. 웹페이지에 스타일과 레이아웃을 정의하는 CSS에서 <body> 엘리먼트의 background-color를 명시하지 않으면 <body>의 백그라운드 영역은 렌더가 되지 않는다. 마치 PNG나 TARGA 이미지의 배경에 alpha 값을 0으로 준 것과 같다. 예로 트위치에서 제공하는 TwitchAlerts와 같은 오버레이 도구를 사용하면 방송 스트리밍 중에 채팅창이나 기부, 응원, 팔로우, 구독을 하면 CG로 표현되어 화면에 오버레이 할 수 있다.



그림 4. 트위치의 후원 / 채팅창 오버레이

1. vMix는 HTML 웹페이지를 렌더링할 수 있는 CEF(Chromium Embedded Framework) 브라우저(= headless browser)를 탑재한다.

NodeCG + vMix = 방송 CG 시스템



그림 5. NodeCG를 활용한 온라인 게임 스트리밍 중계

자동으로 그래픽이 변경되도록 한다. 예로 게임 승패에 따라 게임플레이어 이름, 캐릭터 사진, 진행 상황, 점수표를 쉽게 바꿀 수 있도록 NodeCG 대시보드에서 제어를 하면 된다.

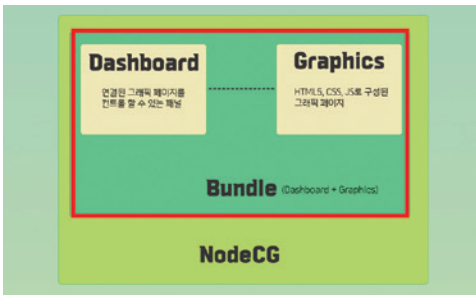


그림 6. NodeCG 구조

하는데 개발자가 사용하기 쉽도록 API 구조화가 잘 되어있어 개발자는 백엔드 개발에 드는 시간을 줄이고 디자인에만 집중하면 된다. 현재는 NodeCG 프레임워크를 사용하지 않고 Next.js로 NodeCG와 같은 프레임워크를 개발·연구 중이다.

NodeCG 운용법

하나의 예를 보자. Dashboard[그림 7]에서 볼 수 있듯이 그래픽으로 표현하고자 하는 문자를 입력한다. 그 입력된 데이터 값은 NodeCG API를 통해 Graphics[그림 8]로 전달되어 Lower Third 효과의 애니메이션으로 표출된다. 응용하면 텍스트 뿐만 아니라 외부로부터 데이터를 받아 다양한 시각화 라이브러리를 활용하면 그래프, 2D 그림, 3D 모델링, 어떤 형태로든 실시간으로 보여줄 수 있다.

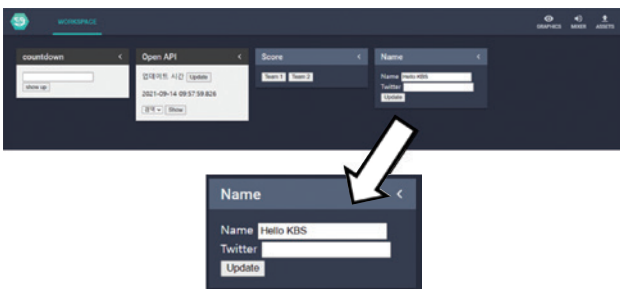


그림 7. Dashboard 예시



그림 8. Graphics 페이지 예시

웹기반 CG 제작시스템의 특징

방대한 NodeJS 생태계는 웹기반 CG 제작시스템의 가장 큰 장점이다. 유명한 애니메이션 자바스크립트 라이브러리로는 Anime.js, Pixi.js, Three.js, Gsap.js²⁾ 등이 있다. 이를 이용하면 3D 모델링, 폰트 및 캐릭터 애니메이션, 영상 필터 효과 등이 웹에서 실시간으로 구현 가능하다. 차트 라이브러리로는 Nivo, React-Vis, Victory, Recharts, ChartJS 등이 있다. 본 프로젝트에서는 Anime.js 로 움직이는 애니메이션을 구현하였고, Nivo 차트 라이브러리를 이용하여 실시간 투표결과를 파이차트로 구현하는 프로젝트를 수행하였다.

또한 웹API와 연동이 쉬워 CG 자동화 워크플로우 구현이 용이하다. 공공기관에서 다양한 오픈 API를 제공하고 있다. 이러한 API를 이용하면 실시간 자료를 재가공하여 원하는 목적의 CG를 바로 표출할 수 있다. 특히 정부에서 제공하는 공공데이터 오픈 API를 잘 활용한다면 시청자들이 알고 싶어 하는 공공데이터를 이해하기 쉽게 시각화하여 실시간으로 전달할 수 있다. 네트워크 기반이기 때문에 같은 네트워크에 있는 어느 사용자가 쉽게 접속하여 CG를 구현할 수도 있다. 즉, 제작시스템과 제작인력을 효율적으로 구성할 수 있다. 또한 본사에서 잘 만든 CG 웹페이지를 지역에서도 템플릿화하여 쓸 수 있다면 매우 큰 메리트이지 않은가. 여기에 같은 CG 템플릿 안에서 제작인력이 해당 지역국 상황에 맞게 웹CG를 변경하여 적용한다면 업무 효율에 도움이 될 것이다. 부가설명을 하자면 벡터그래픽을 우선하는 웹기반 CG는 2K, 4K, 8K 등 모든 해상도에 자동으로 대응할 유연성을 가진다. 본사와 지역 간의 시스템 차이로 발생하는 CG 호환성 문제는 웹기반 CG를 도입하여 문제를 해결할 수 있다. 예로 UHD 기준으로 제작된 CG를 지역에서 HD로 쉽게 변환하여 사용할 수 있어 리소스 재사용성을 높일 수 있다.

마지막으로 라이브에 특화되어 있어 소프트웨어로 실시간에 원하는 영상을 별도의 파일출력 없이 바로 렌더링하여 화면으로 표현할 수 있다. 서버/클라이언트 모델을 기반으로 하는 웹CG는 별도로 CG를 저장하는 스토리지가 필요가 없다. Chrome 엔진의 Hardware Acceleration 기능이 발전하면서 실시간 렌더 성능도 좋아졌다. 2022년 하반기 Chrome 브라우저에 WebGPU 기능이 정식으로 탑재될 계획이다. 그렇게 되면 WebGL 보다 훨씬 뛰어난 실시간 3D 렌더 성능을 보여줄 것으로 기대하고 있다. 언리얼 엔진, Unity, Vizart 같은 무거운 소프트웨어를 설치할 필요가 없어 오퍼레이터 입장에서는 운용하기가 훨씬 쉬워질 것으로 생각된다.



그림 9. Chrome Canary 브라우저에서 GLTF 모델을 WebGPU API로 렌더한 모습

영상송출 S/W

방송화면에 적용하려면 웹페이지를 인풋소스로 받아 방송신호로 변환할 수 있는 소프트웨어가 필요하다. 대표적으로 vMix가 있으나, 정리 차원에서 비슷한 기능들을 가진 소프트웨어 간의 차이점을 비교 분석해보았다.

	OBS Studio	xSplit	vMix	CasparCG
스트리밍	O	O	O	X
SDI Video Playout	X	X	O	O
HTML 렌더링	O	O	O	O
가격	무료(오픈소스)	유료	유료	무료(오픈소스)

표 1. 영상 믹서 소프트웨어 비교 분석표

유튜버들이 가장 많이 사용하고 있는 스트리밍 프로그램으로 OBS Studio가 있다. HTML 페이지를 인풋소스로 받을 수 있지만, 스트리밍 전용이어서 방송에는 사용할 수 없다. 공식적으로는 SDI Video 출력이 불가능하지만 decklink 카드 tweak을 통해 가능하다고 한다.

2. 1년 전 기준으로 현재는 수많은 애니메이션 라이브러리가 존재한다.

vMix는 스트리밍도 가능하고 SDI Video 출력이 가능하다. 유료이지만 유료라서 사용 친화적 인터페이스와 잘 정리된 메뉴얼, 그리고 활성화된 커뮤니티 포럼을 통해 발생하는 문제들도 빠른 업데이트를 통해 해결이 잘 되는 편이다.

여기서 주목할 점은 오픈소스 기반의 CG 전용 프로그램인 CasparCG다. BBC를 포함한 대부분의 유럽 방송국에서 CasparCG를 사용하고 있다. 이미 수많은 레퍼런스를 가지고 있으며 다양한 방송 장비와 시스템으로 엮여 많이들 활용하고 있다. 그래서 유럽방송사들은 SDI Video 출력으로는 CasparCG를, 스트리밍으로는 OBS와 혼용해서 사용한다. CasparCG 커뮤니티에 들어가 보면 CasparCG를 활용한 다양한 프로젝트를 확인할 수 있는데, 최근에는 리액트 UI 프레임워크를 사용

하여 텍스트 입력, 애니메이션, 위치 조정 등 모든 작업을 웹에서 할 수 있도록 한 CasparCG React client 프로젝트도 있다. 이와 유사한 프로젝트가 최근 들어 많이 개발되고 있는데, 코로나 발생 이후 비대면 방송시스템의 요구가 맞아떨어지면서 생긴 현상이라고 볼 수 있다.

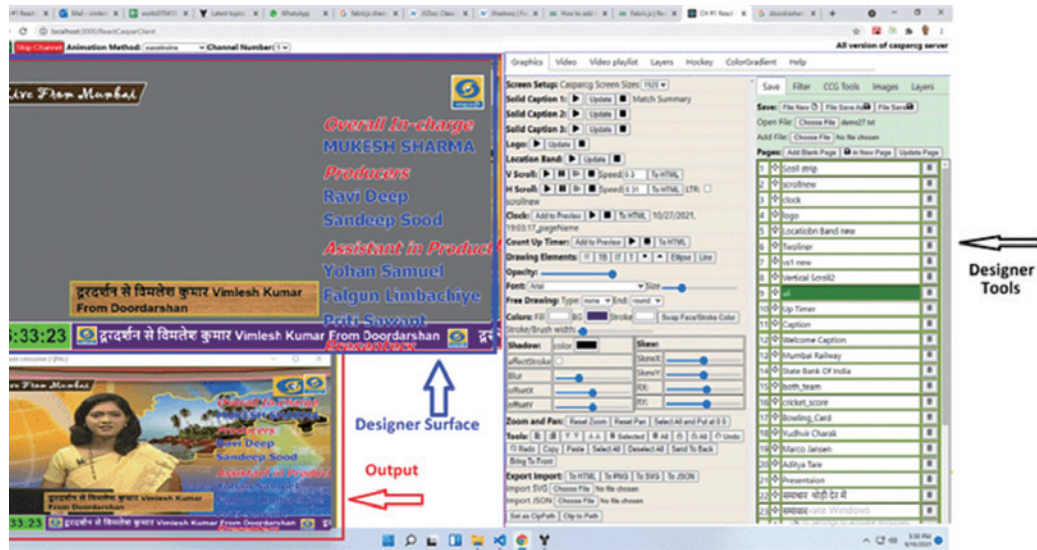


그림 10. CasparCG React client

NodeCG 개발 번들 설명

본 프로젝트에서는 4개의 번들을 개발하였다.

개발 번들 #1 : 채팅창 뉴스티커로 표출

번들의 컨셉은 프로그램 참여용 채팅앱에서 사용자들이 대화한 채팅 내용 중에 의미 있는 대화, 혹은 프로그램에 대한 질문을 뉴스티커 CG로 표출하는 것이다. 이미 방송에서 비슷한 컨셉으로 사용하고 있지만, 기존방식과의 차이점은 문자참여방식이 아니라 채팅웹앱을 통한 참여방식이라 할 수 있다. 채팅앱은 실제 사용자가 접근할 수 있도록 개발에 사용되는 모든 서비스는 퍼블릭 클라우드 기반으로 한다. 즉 DB와 웹앱 호스팅은 클라우드 서비스를 사용한다. 또한 SNS 인증을 통해 로그인한 사용자만 채팅앱을 이용할 수 있도록 SNS 로그인 인증 API를 이용해야 했다. 이로써 Public API와 Private API를 구분하는 웹앱을 만들어야 했다.

이 개발과정에서 다소 생소할 수 있는 스택을 사용하였다. 먼저 클라우드 DB로는 Heroku에서 호스팅하는 PostgreSQL을 이용하였다. 그리고 백엔드 서버는 GraphQL 언어를 처리할 수 있는 Hasura 서비스를 이용하였다. Hasura는 AWS 기반의 BaaS(Backend as a Service)로 제공한다. 클라이언트 웹앱은 Microsoft Azure에서 호스팅하고, GraphQL 언어로 백엔드와 통신하기 위해 Apollo Client(for React)를 이용하였다.

SNS 로그인 API를 사용하기 위해서는 Auth0를 이용하였다. SNS 인증 프로바이더로는 카카오톡 인증을 사용하였다. 대부분의 대한민국 국민이라면 카카오톡계정을 가지고 있기 때문이다.



그림 11. 뉴스티커 번들 개발 구성도

개인적으로는 정말 어려운 도전이었지만 그만큼 성취감도 컸다. 아래는 결과물이다.



그림 12. 웹앱 랜딩페이지

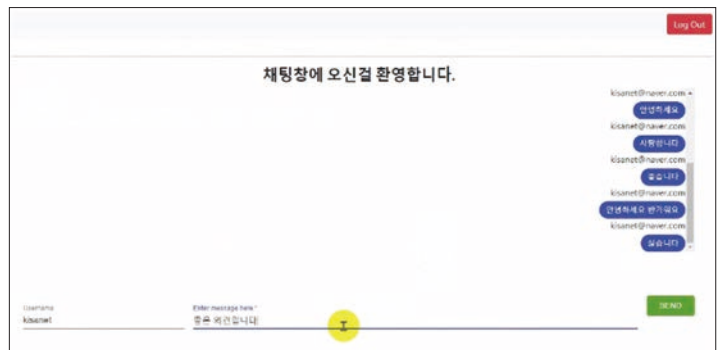


그림 13. 채팅 페이지(Material UI로 구현)



그림 14. 채팅글 관리용 대시보드

NodeCG 대시보드에서 업데이트 버튼을 눌러 최근 채팅 내용을 불러와 선택한 채팅 글만 뉴스티커로 송출되도록 한다. 그래픽 페이지 하단에 채팅 글이 차례로 표시되는 뉴스티커 CG가 표출된다.

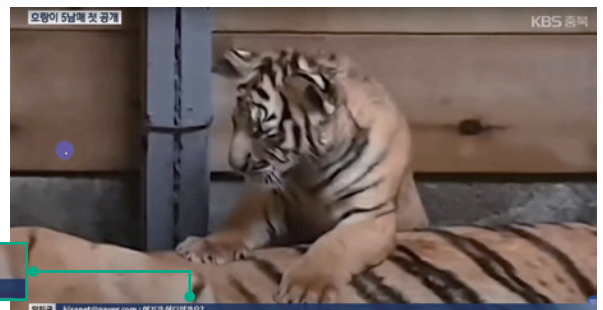
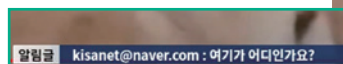


그림 15. 영상 하단 채팅 글이 뉴스티커로 표출

SNS 로그인 후 채팅창에 접속한다. 유저 인터페이스 라이브러리로는 React를 사용하였다.

개발 번들 #2 : 공공데이터 연동

공공데이터포털은 공공기관이 생성 또는 취득하여 관리하고 있는 공공데이터를 한 곳에서 제공하는 통합 창구이다. 본 연구에서는 보건복지부에서 관리하는 코로나19 시·도 발생 현황 API를 활용하여 실시간으로 현황데이터를 시각화해서 보여주는 CG를 구현하였다. 먼저 공공데이터 포털에서 코로나19 시·도 발생 현황 API를 신청한다. 신청 후 마이페이지에서 신청이 승인된 Open API를 확인할 수 있다.



그림 16. 공공데이터포털의 신청한 오픈 API 목록

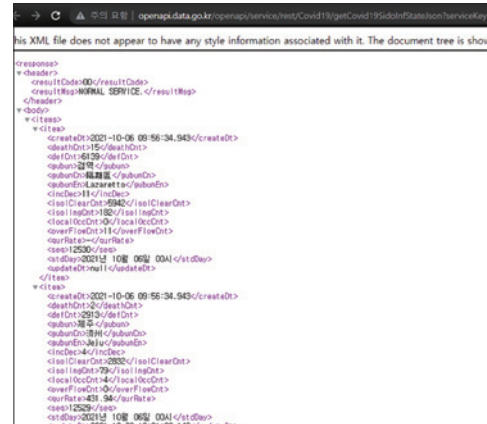


그림 17. 코로나19 발병현황 API 리턴 결과 (XML 포맷)

발급받은 코로나 현황 API에 Query String Parameter로 인증키와 요청날짜를 입력하면 해당 날짜의 전국 코로나 발병현황 값을 받아올 수 있다. 이 리턴값을 NodeCG에서 파싱하여 지역별로 발병현황을 보여주기 위해 했다. API fetch 라이브러리로는 Axios를 사용하였다. 그리고 Object manipulation 유틸리티로는 자바스크립트에서 가장 유명한 라이브러리 중 하나인 Lodash를 사용하였다.

대시보드[그림 18]에서 업데이트를 눌러 최근 날짜로 변경하고, 원하는 지역을 선택하여 Show 버튼을 누르면 그래픽 페이지에서는 썸네일 CG에 해당 지역의 코로나 발병현황 데이터가 입력되어 보여준다.

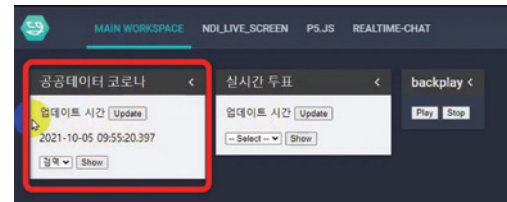


그림 18. 대시보드에서 코로나19 발병 현황 CG 그래픽 제어



썸네일 CG

썸네일 CG에 데이터가 입력된 모습 & 교차 출력 애니메이션 적용

한정된 CG 레이아웃에서 다양한 데이터를 보여주기 위해 글자를 교차하여 보여주는 애니메이션을 사용하였다. 전국대비 발생률이라는 긴 글자와 발생률 실제 데이터는 레이아웃에서 큰 공간을 차지하기 때문에 동일한 위치에서 교차로 보여주어 공간활용도를 높였다. 또한 실제로는 데이터값에 변동이 없더라도 최신 정보로 업데이트가 되고 있다는 느낌을 줄 수 있도록 데이터별로 loop staggering animation을 적용하였다. 이를 구현하기 위해 Anime.js의 타임라인 기능을 활용하였다. 이 부분도 웹기반 CG만의 장점이라 할 수 있다. css의 z-index 속성을 사용하면 고정위치에 레이아웃을 무한하게 얹을 수가 있어 데이터 표현에 대한 제약이 사라진다.

개발 번들 #3 : 실시간 투표 결과를 차트 CG로 표출

이번 번들의 컨셉은 프로그램 참여용 웹에서 설문지에 투표한 결과값을 실시간으로 차트그래프로 표현하는 것이다. 이전 뉴스티커 CG 번들과 개발프로세스에는 큰 차이는 없으나 다만 표출되는 CG가 텍스트에서 그래프로 바뀐 차이가 있다. 기존 미르CG에서도 소켓 통신으로 텍스트 데이터값을 연동하는 것은 어렵지 않으나 실시간으로 움직이는 그래프 애니메이션을

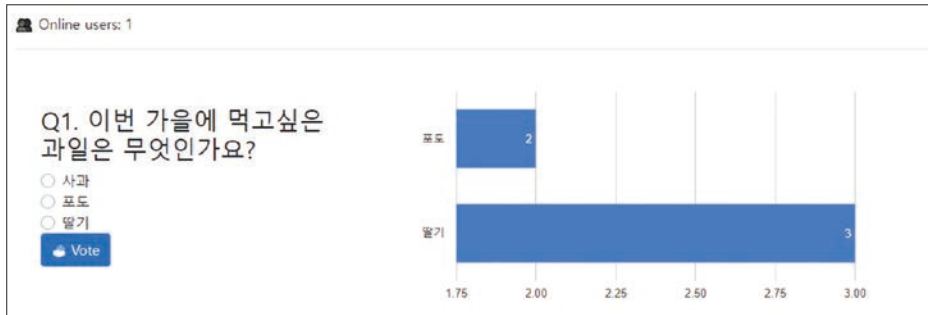


그림 19. 설문조사 페이지

로그인 후 설문지 페이지에 접속한다. 사용자가 투표한 결과값이 실시간으로 그래프로 표현되는 것을 확인할 수 있다. 여기서 실시간 통신을 가능케 하는 GraphQL subscription 오퍼레이션을 사용한 덕분이다.



그림 20. 화면 세이프티존 가이드

NodeCG에서 생성한 그래픽 페이지를 vMix에 합성한 화면이다. CG의 오른쪽에 보이는 파이차트는 Nivo의 Pie 라이브러리를 사용하였다.

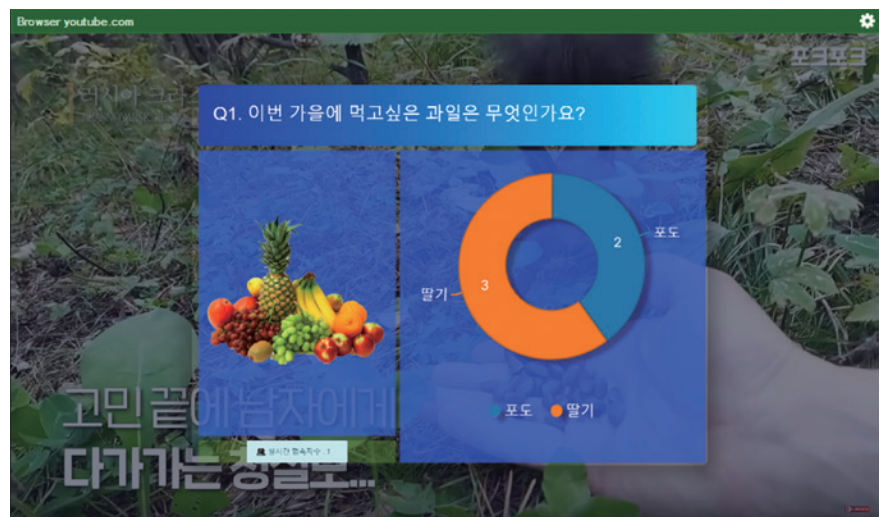


그림 21. 투표결과를 파이차트로 구현하여 화면에 오버레이

개발 번들 #4 : 우상단 프로그램 예고

- 그래픽 크기 : 144px by 144px
- 상측 여유 픽셀 : 144px
- 우측 여유 픽셀 : 100px

청주충국 다큐프로그램인 '다큐공작소'의 초단위 카운트다운 프로그램 예고 그래픽을 제작하였다. 기존에 사용하고 있는 화면 세이프티 존 규격인 '편성전략국-1181 (2020.07.15.) 「화면 세이프티존 확장 변경 시행」'에 따라 진행하였다. 먼저 KBS 1TV 채널 로고 하단에 프로그램 예고 그래픽 위치와 픽셀 크기를 좌측과 같이 측정하였다.



그림 22. 우상단 예고 세이프티존 가이드

기존의 자막발생기 또는 어도비 편집 프로그램을 사용해 직관적으로 정사각형 모양의 프로그램 예고 그래픽을 만드는 작업과는 전혀 다르다. JavaScript 기반으로 [그림 23]과 같이 각기 다른 선의 굵기와 위치, 텍스트를 만들기에 오랜 시간이 걸린다. 이를 해결하기 위해서 Sketch, Figma, Zeplin과 같은 다양한 디자이너 툴이 있다. 그림판처럼 만들고자 하는 그래픽을 직관적으로 만들 수 있는데, SVG, JPG 등의 파일

형태로 export 할 수 있어 HTML 기반인 NodeCG에서 활용 가능하다. 본 연구에서는 Figma를 통해 기존 '다큐공작소' 프로그램 로고를 모방하여 만들었다.

다만, Figma와 같은 디자이너 툴에 한계가 있다면 원하는 값을 타임라인에 따라 변경하는 것은 불가능하다. 즉, 스틸 이미지에서 애니메이션을 주기 위해서는 제작한 SVG 파일을 JavaScript로 가져와 효과를 주어야 한다.



그림 23. 기존 다큐공작소 로고



그림 24. 새로 제작한 우상단 프로그램 로고

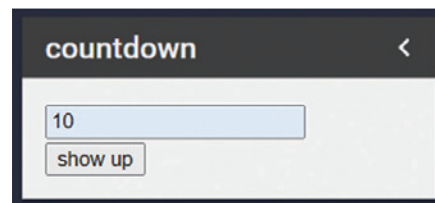


그림 25. NodeCG Dashboard 화면으로 카운트다운 숫자 입력

NodeCG에서 구현하고자 하는 목표는 다음과 같다.

- ① 원하는 초 단위 카운트다운 숫자를 Dashboard에서 입력 [그림 25]
- ② show up 버튼을 누르면 프로그램 로고 카운트다운
- ③ 카운트다운이 끝남과 동시에 프로그램 로고는 사라짐
- ④ 언제든지 초 단위 카운트다운 숫자를 설정해 show up 버튼을 클릭하면 로고 표출



그림 26. 세이프티존 규격에 맞춘 우상단 프로그램 로고

이제 vMix를 통해 [그림 24]를 화면 오버레이 하면 [그림 27]과 같이 송출된다. 이로써 NodeCG에서 운용했을 때의 장점은 렌더링을 거치지 않고도 초 단위 카운트다운 숫자를 바로 재설정하여 표출할 수 있다는 것이다. 또한 어떠한 장르의 프로그램 홍보에도 쓰일 수 있기 때문에 다양하게 활용될 수 있을 것으로 예상된다.



그림 27. vMix 인풋으로 받은 우상단 프로그램 로고

국내의 방송 CG 제작기술의 변천사

방송에서 CG는 시청자에게 논리적인 해석이나 정확한 정보전달이 요구될 때 보조적으로 활용된다. 그런 점에서 방송 CG는 메시지를 효과적으로 전달하기 위한 중요한 수단이다. 문자 중심이었던 방송 CG는 그래픽 소프트웨어의 발전과 프로그램 장르의 다변화로 시각 기호, 그래프, 캐릭터 등 표현기법이 다양해졌다. 재난방송에는 인포그래픽을 활용하여 복잡한 정보를 간결하고 명료하게 표현할 수 있고, 스포츠 경기에서는 실시간 스코어뿐만 아니라 게임의 전반적인 모든 상황을 표출하여 시청 중인 경기의 이해도를 높여준다. 그 외 시사·다큐 프로그램에서는 여운이 남는 메시지를, 예능 프로그램에는 다양한 시각 효과를 주어 방송의 재미를 더하기도 한다.

문제는 기존의 방송 CG 시스템으로는 그 기능이 활용한계점에 도달하여 새로운 변화가 없다는 것이다. 프로그램 장르별 CG 포맷은 전형적인 틀을 따라가다 보니 제작워크플로우의 변화가 없고 CG를 활용하는 제작방식에서 창의적인 발상을 기대하기 어려워졌다. 현재로서는 CG는 나타났다 금세 사라지는 Fade in/out 효과가 대부분이며 영상화면의 순간순간을 텍스트화하여 상황적 의미를 강조하는 성격이 짙어졌다. 이는 CG가 질보다 양적으로 프로그램의 가치를 승부하는 방향으로 발전하게 된다.

해외는 국내와 사뭇 다르다. 얼마나 많은 폰트를 보유하느냐를 중요하게 여기는 국내와는 다르게 해외는 애니메이션 디자인을 중심으로 발전해왔다. 그 이유는 자막을 잘 보지 않는 문화차이도 있지만 그 근간에는 플래시 애니메이션에 있다. 플래시 애니메이션은 gif 애니메이션의 한계를 극복하기 위해 개발되었으며, Key/Fill 기능을 지원하기에 유럽방송사에서는 플래시를 지원하는 CasparCG와 같이 쓰고 있다. 액션스크립트 기반의 플래시는 역동적이고 복잡한 애니메이션을 구현할 수 있는데, 스크립트 작성이 다소 어렵기 때문에 디자이너의 영역인지 개발자의 영역인지 모호한 부분이 있다. 해외방송사에서는 스크립트 기반의 플래시를 계속 사용해오다 최근에는 플래시에서 HTML 기반으로 바뀌 가는 추세다. 반면에 국내는 비주얼리서치와 컴덱스에서 만든 애니메이션 기능이 탑재된 CG 장비를 도입하게 된다. 그래픽은 포토샵이나 CG 장비에서 하고 애니메이션 효과는 CG 장비가 맡는 것으로 이원화가 되어있다.

[그림 28]은 Space X 공식 유튜브 채널에 있는 우주선 발사 영상이다. CasparCG 커뮤니티에서 확인한 바로는 영상 하단의 CG가 웹기반으로 만들어졌다고 한다. 우주선 발사 과정의 모든 상태정보를 실시간 CG로 보여주는데 어떻게 구현했을까를 고민해보며 영상을 한번 보시길 추천한다.

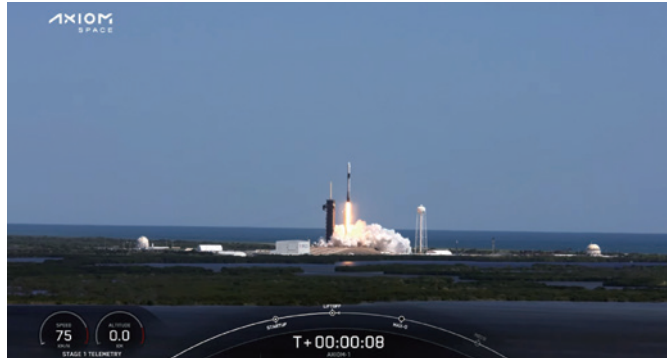


그림 28. Space X 공식 유튜브의 영상

웹개발 테크 트리

테크트리란 실시간 전략 게임에서 나온 용어로, 게임상에서 레벨업을 하는 과정에 연구, 업그레이드 절차의 계층도를 뜻한다. 웹개발을 목표로 무엇을 구현할 수 있고, 어떤 테크트리를 타야할 지에 간략히 정리해보았다.

1. Data visualization : D3.js + 빅데이터 + UI 프레임워크(React / Svelte)

D3.js의 개발자인 Mike Bostock인 New York Times에서 다양한 분야의 통계데이터를 시각화하여 웹사이트에 게재하였는데 이를 참고하면 좋다.

2. Realtime Video Processing : p5.js + NDI + AI 기술

실시간으로 영상 일부를 Blur 처리, 2D 이미지를 3D depth Image로 실시간 변환하기(Google Photo 신기능)

3. 시청자 참여데이터 전달 : AWS / Azure 등 클라우드 서비스 + GraphQL + Oauth

4. SVG 애니메이터 : Adobe AE + Lottie 등

웹이미지인 SVG를 Adobe Effects에서 Lottie 플러그인을 통해 Export 할 수 있다. SVG는 벡터그래픽이기에 확대해도 깨지지 않는다. Lottie를 사용하면 어려운 웹코딩 없이 쉽게 SVG 생성할 수 있어 웹애니메이터로 전향하는 분들이 많이 사용하는 툴이다.

5. 3D 모델링 : Blender + GLTF + webGPU + three.js + depth Camera(intel realsense)

개인적으로 연구하고픈 프로젝트 주제 중 하나다. 엔트리급 Depth camera가 속속히 등장하면서 3D Camera Tracking 예전보다 훨씬 쉽게 구현이 가능할 것으로 보인다. 버추얼 세트까진 아니더라도 텍스트 모델을 실시간 모션 트래킹은 가능할 것으로 생각된다.

마지막으로

BBC에서는 Rowzed라는 웹기반 CG 시스템을 개발하여 스포츠 중계에 사용하고 있다고 한다. 추측이지만 본 프로젝트의 개발 스택은 2019년 BBC 채용공고 사이트에서 확인 가능하다. 해당 잡의 채용 자격요건으로 HTML5와 Responsive Web Design 그리고 클라우드 서비스 등 다양한 웹기술을 다룰 수 있는 풀스택 엔지니어를 요구한다.

또한 BBC에서 어떤 기술에 관심이 많은지 알고 싶다면 Medium 사이트에서 BBC publication을 구독해보길 바란다. 빠르게 변화는 웹기술에 얼마나 민첩하게 대응하고 연구/개발하는지 확인할 수가 있다.

다소 모호했던 CG라는 영역이 미래에는 방송기술의 당위성을 넓히고 발전 가능성이 있는 먹거리가 되길 바란다. 미래의 방송시스템은 웹(WEB)과 더 친해져야 하지 않을까 하는 고민을 끝으로 글을 마무리한다. 📺