

## [멀티미디어응용] glTF: 그래픽스 데이터 교환을 위한 새로운 표준

### 새로운 표준의 발표

2015년 10월 19일 미국 샌프란시스코에서 열린 HTML5 DevCon(개발자 대회)에서는 그래픽스 분야의 대표적인 Open Consortium인 Khronos Group이 glTF(OpenGL Transmission Format) 표준을 발표했다. 이날은 마침 미국 휴스턴에서 Khronos Group의 대표적 표준 심의 회의인 Khronos F2F meeting(face-to-face meeting)이 열리는 첫날이기도 해서, 양쪽 회의 모두에서 glTF 표준의 발표가 가장 큰 화제에 올랐다.

디지털 자료를 교환해야 하는 분야에서는 전체적으로 실질적 표준으로 사용되는 데이터 포맷(data format)이 굳건하게 확보된 경우가 많다. 오디오 분야에서는 mp3, 비디오 분야에서는 h.264, 이미지 분야에서는 jpeg, 이런 식으로 확실한 파일 포맷이 널리 사용되고 있다. 반면에, 3D 그래픽스 분야에서는 아직까지 확실한 파일 포맷이 확보되지 않았다. 물론, Wavefront OBJ 포맷이나, STL(STereoLithography) 포맷 등이 비교적 널리 사용되는 편이지만, 이들은 제한된 용도로 일부 응용에 적합한 정도이다.

3D 그래픽스 분야의 파일 포맷이 확립되지 못하다 보니, 시장에서는 여러 가지 포맷들이 혼용되고 있고, 그 결과로 응용 프로그램 또는 앱 개발자들은 어떠한 포맷을 사용할지 결정하는 것도 문제였고, 결정한 후에는 이를 지원하는 파일 로딩 툴이나, 라이브러리를 확보하는 것도 문제였다.

3D 그래픽스 분야에 뚜렷한 포맷을 정하기 어려웠던 가장 큰 이유는 필요로 하는 데이터가 다양했다는 점을 꼽을 수 있다. 3D 그래픽스 분야의 대표적인 응용 분야인 실시간 컴퓨터 애니메이션 재생 프로그램을 본다면, 프로그램의 입장에서는 일단 재생할 물체를 정의하는 3차원 기하학 모형과 이들의 결합 관계를 표현한, 3차원 기하학 자료들이 필요하다. 이 3차원 물체를 화면에 재생할 때는 다시 물체의 물질 특성(material property)과 텍스처(texture) 정보들이 흔히 2차원 이미지 형태로 제공되어야 한다. 이제 텍스처를 입혀서 입체적으로 잘 표현된 3차원 물체는 시간이 흐름에 따라, 적절한 동작을 하도록, 애니메이션 자료나, 스킨 제어 자료들이 필요하다. 이들은 3차원 커브(curve)나, 쿼터니언(quaternion), 스킨 행렬(skin matrix) 등의

독특한 자료로 제공되어야 한다. 최근에는 좀 더 사실적인 색상 표현을 위해서 실제 사용되는 셰이더 프로그램(shader program)까지 제공하는 경우도 있어서, 필요로 하는 자료들이 더욱 다양해졌다.

이에 따라, 현재의 3D 그래픽스 분야에서는 기하학 자료, 텍스처, 커브 등의 특정 종류의 자료에 대한 파일 포맷은 나와있고, 나름대로 각 종류별로 널리 쓰이는 표준도 있는 상황이지만, 전반적으로 필요한 자료 전체를 하나로 통합해서 관리하는 표준은 아직까지 확립된 것이 없었다. glTF(OpenGL Transmission Format)은 이러한 다양한 목적의 3D 그래픽스 자산들을 통합해서 관리하도록 해주고, 특히 OpenGL, OpenGL ES, WebGL 등의 OpenGL 패밀리의 그래픽스 라이브러리들에서 쉽게 임포트(import)/엑스포트(export)가 가능하도록 표준화된 툴들과 함수들까지 제공하는 것을 목표로 시작된 표준 명세이다.

## glTF의 내부 구조

glTF는 표준의 제정에 착수할 때, “JPEG for 3D”라는 슬로건을 내걸었고, 최대한의 사용자 및 프로그래머 편의성을 확보하려고 노력하고 있다. 3D 자산(asset)들을 다양한 응용 프로그램들과 앱에서 사용할 수 있도록 하고, 특히 WebGL, OpenGL ES, OpenGL을 사용하는 앱들에 최적화하는 것을 목표로 한다.

glTF의 내부 구성으로는 우선 상위의 노드 구조(node hierarchy)를 설정할 수 있도록, 기존의 JSON(JavaScript Object Notation)을 사용해서 카메라, 기하학 구조, 애니메이션, 스킨, 셰이더, 텍스처, 광원, 물질 특성 등의 다양한 노드들을 제공하고 있다. 셰이더 구조는 OpenGL 등에서 직접 사용 가능한 셰이더 프로그램은 물론이고, 셰이더에서 즉각적으로 필요로 하는 물질 특성(material property)을 확장 명세(extension) 형태로 추가할 수 있도록 했다. 텍스처 이미지는 JPEG 등의 압축 파일 포맷을 주로 사용해서, 전송 효율을 높이려고 했다. 물체의 기하학 구조와 애니메이션 구조는 모두 이진 자료를 사용하도록 해서, 역시 전송 효율을 극대화했다.

내부 설계를 보면, 자료의 다운로드나 전송 효율을 위해서, 메쉬 구조, 애니메이션 데이터 등에서 이진 표현(binary representation)을 대폭 도입했고, OpenGL에의 최적화를 위해서는 OpenGL의 고유 자료형들을 그대로 사용해서 자료의 파싱(parsing)이나 변환에 필요한 시간을 최소화했다.

표현력에서는 가능한 한 많은 3차원 애니메이션을 커버할 수 있도록, 노드 구조(node hierarchy), 물질 특성, 애니메이션, 카메라, 광원 등의 설정이 가능하게 했다. 또한, 벤더(vendor)들이 자유롭게 자신의 확장 명세(extension)를 추가할 수 있도록 해서, 채택율을 높이려고 노력하고 있다.

glTF는 제정 과정이 완전 공개되어 있었고, 활발한 토론을 거쳐, 10월 19일에 1.0 버전이 최종 공표되었다. 현재 glTF를 채택한 유명 시스템들로는 three.js, babylon.js 등의 로더(loader)가 있다. 또한 크롬 브라우저에서 작동하는 WebGL 기반 Virtual Globe and Map Engine인 Cesium 엔진에서도 glTF를 채택하고 있다. 이와는 별도로, 기존의 COLLADA 내용들을 glTF로 변환해주는 collada2glTF 컨버터를 비롯하여, COLLADA to glTF 컨버터, FBX to glTF 컨버터 등이 제공된다.

glTF에는 다양한 확장 명세(extension)들이 이미 정의되어 있는데, 현재 가장 중요한 것으로는 Open3DGC(Open 3D Graphics Compression) 라이브러리를 들 수 있다. 이 라이브러리는 MPEG 3D 표준에 대한 그래픽스 코덱(graphics codecs)을 매우 효과적으로 제공한다. 현재 Open3DGC 인코더는 C++로 구현되어 제공되고 있고, 이에 대응되는 디코더는 C++와 JavaScript 형태로 제공되어, OpenGL은 물론, WebGL에서도 효과적으로 사용할 수 있다. 3차원 자료들에 대한 이 라이브러리의 압축률은 매우 우수해서, 일반적인 압축 방법으로 사용되는 gzip 등의 기법들에 비해 확실한 우위를 보여준다. 이외에도 다양한 확장 명세들이 속속 개발되고 있다.

## 결언

이제까지 설명한 바와 같이, glTF는 3D 그래픽스 분야의 모든 자산(asset)들을 하나로 통합해서 관리하기 위한 표준 파일 포맷으로 제시된 국제 표준이다. Khronos Group에서 새로이 제안한 표준이지만, 제정 과정에서 상당히 오랫동안 공개 토론을 거쳤고, 여러 시스템들에서 입출력 방법으로 이미 채택하기도 했으므로, 급속하게 보급될 것이 분명해 보인다. 기존의 OpenGL 사용자라면, 새로 개발하는 프로그램이나 앱에 새로운 표준을 적용해 보는 것은 확실한 이득을 가져올 것은 분명하다. 다만, 그 확산 속도가 과연 다른 분야의 파일 포맷들, 즉 mp3, jpeg, h.264 등에 필적할 만큼 빠를지에 대해서는 아직까지는 확실하지 않다. 성공할 것은 분명하되,

완전한 성공이 될지, 절반의 성공이 될지는 지켜보아야 할 것으로 보인다.

백낙훈 (주)모바일그래픽스 대표, [oceancru@gmail.com](mailto:oceancru@gmail.com))