glTF Overview
July 2019

Presenter: Ed Mackey, AGI

Slides by Neil Trevett & the 3D Formats Working Group
glTF - The JPEG of 3D!

- **Audio**: MP3
- **Video**: H.264
- **Images**: JPEG
- **3D**: glTF

**Efficient, reliable transmission**
- Bring 3D assets into 1000s of apps and engines
- Compact to Transmit
- Simple and Fast to Load
- Describes Full Scenes
- Runtime Neutral
- Open and Extensible

**glTF 1.0 - December 2015**
- Primarily for WebGL
- Uses GLSL for materials

**glTF 2.0 - June 2017**
- Native AND Web APIs
- Physically Based Rendering
- Metallic-Roughness and Specular-Glossiness

**glTF spec development**
on open GitHub - get involved!
[https://github.com/KhronosGroup/glTF](https://github.com/KhronosGroup/glTF)
Wait... **Minecraft** ???

- Operator of “Creative” mode game
- `/give @s structure_block`
- Place the Structure Block into the world
- Right-click to activate
- Export up to 32 x 32 x 32 area to .glb
glTF 2.0 Scene Description Structure

- `.gltf (JSON)`
  - Node hierarchy, PBR material textures, cameras

- `.bin`
  - Geometry: vertices and indices
  - Animation: key-frames
  - Skins: inverse-bind matrices

- `.png`  `.jpg`  `.ktx2`
  - Textures

Mandatory Metallic-Roughness Materials

- Texture based PBR materials

Optional Specular-Glossiness Materials

- Geometry
Draco glTF Mesh Compression Extension

- Library for compressing and decompressing 3D geometric meshes and point clouds
  - Draco designed and built for compression efficiency and speed - great fit with glTF!
    - https://github.com/google/draco

- Draco glTF extension launched in February 2018
  - https://github.com/KhronosGroup/glTF/blob/master/extensions/2.0/Khronos/KHR_draco_mesh_compression/README.md

- Google has released Draco encoders and decoders in open source
  - C++ source code encoder to compress 3D data
  - C++ and JavaScript decoders for the encoded data
    - https://github.com/google/draco/tree/gltf_2.0_draco_extension

- glTF/Draco compression already in use
  - Blender, three.js, BABYLON.JS, Adobe Dimension, glTF pipeline, FBX2glTF, AMD Compressonator and glTF sample models

![Mesh Compression Ratios](image_url)
Universal Textures for glTF

- Fragmentation of GPU texture formats is significant issue for developers
  - Binomial’s ‘Basis Universal’ technology enables JPEG-sized texture assets
  - Transcodable on-the-fly to natively supported compressed GPU formats

- glTF Universal Texture extension uses KTX2 as a flexible container
  - Precisely defined specification for consistent, cross-vendor generation and validation
  - Can contain wide range of texture formats used in Vulkan/DirectX/Metal
  - Supports streaming and full random access to MIP levels
  - Subset of full KTX2 - mandating supercompressed textures using Basis Universal technology

*ASTC support in development
Compare File Sizes: 512 x 768 RGB Photo

- Uncompressed PNG: 1,179,648
- Compressed PNG: 602,078
- JPEG: 65,397
- ETC1S: 196,608
- Basis Universal: 55,872

File Size vs GPU Size
Universal Textures - Get Involved!

- Design discussions
  - [https://github.com/KhronosGroup/glTF/pull/1612](https://github.com/KhronosGroup/glTF/pull/1612)

- Khronos open source tools
  - [https://github.com/KhronosGroup/KTX-Software/tree/ktx2](https://github.com/KhronosGroup/KTX-Software/tree/ktx2)
  - toktx - create a KTX2 file from a set of .png images
  - ktxsc - convert images in KTX2 file to supercompressed images using Basis transcoder

- Ecosystem forming around KTX2
  - Khronos glTF texture tool with GUI for generating supercompressed textures
  - Increasing number of run-times integrating prototype KTX2 support

Applications and engines with prototype KTX2 support

Blender 2.80 Supports Full glTF Import/Export

Project driven by Mozilla, Khronos and the glTF community

Blender’s Principled BSDF Shader node maps to glTF’s PBR materials

https://docs.blender.org/manual/en/2.80/addons/io_scene_gltf2.html
Roadmap Discussions

- Many of these topics are being discussed on GitHub
  - [https://github.com/KhronosGroup/gltf](https://github.com/KhronosGroup/gltf)
  - Come and give your views!

- Next Generation PBR Materials

- Animation 2.0
  - Advanced Avatars and Face emoji, with compression

- LOD and Streaming

- Cross-asset linking

- Enhanced Metadata

NVIDIA MDL Physically Based Rendering
Focus on glTF Ecosystem Robustness

- Khronos constantly working on improving ecosystem's consistency
  - Rendering (reference viewer, reference environment)
  - Technical low-level issues (validator & asset generator)

- If you are CREATING glTF Files
  - Ensure generated files are validator clean
    - https://github.com/KhronosGroup/glTF-Validator
  - Help the community understand what your exporter supports
    - https://github.com/KhronosGroup/glTF/issues/1271

- If you are LOADING glTF files
  - Ensure loader can correctly load all sample models (integration tests)
    - https://github.com/KhronosGroup/glTF-Sample-Models
  - Ensure loader can correctly load all asset generator models (unit tests)
    - https://github.com/KhronosGroup/glTF-Asset-Generator

Users of glTF can help to keep glTF reliable and consistent!
Resources

• glTF Home Page
  - https://www.khronos.org/gltf/

• glTF GitHub
  - https://github.com/KhronosGroup/glTF

• PBR 2.0 - advanced materials
  - https://github.com/KhronosGroup/glTF/issues/1442

• Khronos 3D Commerce Exploratory Group
  - https://www.khronos.org/exploratory/3d-commerce/

• More Information
  - www.khronos.org
  - ntrevett@nvidia.com
  - @neilt3d
Up Next

- Facebook, Renee Rashid
- Cesium, Omar Shehata
  - 3D Tiles & KTX2 / Basis Universal
- DGG, Max Limper
- Uber, Georgios Karnas
- UX3D, Fabian Wahlster and Moritz Becher
  - glTF editor and tools
- Esri, David Körner
  - glTF with Esri JS API
- Sketchfab, Alban Denoyel
- Google, Adrian Perez
  - AR Search at Google
- Wayfair, Shrenik Sadalgi
  - Khronos 3D Commerce