glTF Briefing
September 2016

Efficiently describe and transmit your 3D scenes!
Background and Motivation

• OpenGL ES and WebGL have led to a proliferation of Web 3D...
  ... but no standard way to deliver data into applications
  - Efficient transmission
  - Full scene information
  - Vendor- and runtime-neutral

  **OBJ - too simple | COLLADA - too bulky | FBX - vendor-specific**

• As a result, content and app creators developing new pipelines per project
  - Huge inefficiencies
  - Limited opportunities for sharing data among applications
Compact to Transmit
Fast to Load
Runtime Neutral
Extensible
glTF - The “JPEG of 3D”
Timeline

2012 - 2014
Development

Design iteration
Multiple implementations

2015
Standardization

V 1.0 Ratified
Dec 2015

2016
Conformance, Adoption,
New Features

V 1.0.1 Update
Syntax Validator
New features kickoff
glTF Structure

.glTF
JSON describes node hierarchy, materials, cameras

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

.glsl
Shaders

.png
.jpg
Textures

Describes full scenes— not just meshes

NORAD’s Santa Tracker
glTF Extensions

• Syntax and name registry for extending the base specification
  - KHR_binary_glTF - binary container format, single payload - ratified
  - KHR_materials_common - common fixed function materials and lights e.g.
  - Vendor extensions e.g. CESIUM_RTC, WEB3D_quantized_attributes

• Keeps the base spec small while allow for experimentation and domain-specific use cases

• Popular extension can potentially be promoted to the base spec

File declares extensions used up front
"extensionsUsed" : [
  "KHR_binary_glTF"
]

“extensions” property contains the data
"a_shader" : {
  "extensions" : {
    "binary_glTF" : {
      "bufferView" : // ...
    }
  }
}
Adoption

Oculus Executive Calls For 3D Equivalent Of JPEG To Build The Metaverse

A new standard for 3D scenes is gaining momentum with support from graphics industry leader, potentially laying the groundwork for science fiction’s “metaverse” to be realized.

The GL Transmission Format (gltf) from The Khronos Group, a computer graphics industry standards body, could also put magnitudes more 3D content on the Internet. The Khronos Group is responsible for a variety of technologies critical to the development of modern graphics, including OpenGL and OpenCL.

Publicly Stated Support for glTF
glTF Ecosystem

Drag and Drop FBX -> glTF
(coming soon)
http://gltf.autodesk.io/

AUTODESK
Autodesk FBX -> glTF
AssImp
OBJ2GLTF
glTF Pipeline
COLLADA2GLTF
Cesium converter

Drag and drop COLLADA -> glTF
http://cesiumjs.org/convertmodel.html

Tools

Blender DIRECT export

Translate

Convert | Optimize

Validator

Validate

Export

glTF 1.0.1 Spec in Review and
glTF Validator in open source!
http://github.khronos.org/glTF-Validator/

model/gltf+json MIME type
Approved by IANA!

Apps & Engines

Import

A-FRAME
three.js
js

xenoEngine
bablyon

NVidia
PEX
Cesium

glTF Ecosystem Page
https://github.com/KhronosGroup/glTF#glTF-tools
glTF 1.0.1 Validator

• glTF 1.0.1 tightens specification
  - For robust validation and interoperability
    https://github.com/KhronosGroup/glTF/issues/605

• Validator in open source on GitHub
  - Khronos Validator project RFQ awarded to Alexey Knyazev - doing awesome work!
  - Rigorous checking for correctly formed glTF files
  - Checks JSON syntax, all property details, GL parameter combinations etc. etc.
  - Built using Dart (easy API level integration)
  - Shipping today as client-side drag-n-drop and command-line wrapper
  - Client-side JavaScript library coming soon
  - Extensible - validation plugins for extensions - output can be integrated into the validation report

Please give us feedback on GitHub!

http://github.khronos.org/glTF-Validator/
**Roadmap Discussion Topics**

- Physically Based Rendering
  - Modern, compact, scalable
  - Fraunhofer, NVIDIA MDL?

- Streaming and Mesh Compression
  - MPEG 3DGC (royalty-free), Fraunhofer SRC?

- Enhanced API Support
  - Make efficient use of WebGL 2.0 & Vulkan

- Advanced Surfaces
  - Pixar’s OpenSubdiv?

- Streaming and Mesh Compression
  - MPEG 3DGC (royalty-free), Fraunhofer SRC?

- Increased Efficiency
  - Improved parsing, arrays, bounding boxes, spatial constructs

- Enhanced Metadata
  - For mixing advanced experiences

- Enhanced Animation
  - Morph Targets

**Must avoid the complexity trap!**

Core glTF must remain efficient and straightforward to use

‘Feature Sets’ for domain specific functionality

**glTF Community on GitHub**

https://github.com/KhronosGroup/gltf

**Or join Khronos to get directly involved!**
Khronos AR/VR Standards

WITHOUT Standards

WITH Standards

Tools import/export custom 3D formats and so do not interoperate

Mix and match tool pipelines through common 3D asset import/export

Every service/app stores 3D assets in a custom format -> Silo’d content

3D assets are easily understood and used by any application and device

Long download times and proprietary code to unpack received 3D assets

3D assets packed into efficient formats with streaming and compression

Apps have to be ported to each device and often don’t use acceleration

APIs provide consistent access to graphics, compute and vision acceleration

Khronos standards useful for AR

COLLADA™ Many other authoring formats

GLTF

GLTF™

WebGL

Vulkan™

OpenVX™

OpenCL™

OpenGL™

OpenGL ES™

AR/VR Will Need Many, Many Standards
Khronos standards can help..

© Copyright Khronos Group 2016 - Page 12