



glTF 2.0 - the new transmission format

Norbert Nopper, Co-founder UX3D

Khronos Munich Chapter Meeting 04/28/2017





Content

- About me (One slide only)
- What is glTF 2.0?
- *Killer* features of glTF 2.0
- glTF 2.0 demos
- Summary, Q & A



About me (it is really only one slide)

- Studied computer science
- Graphics/engine programmer
- Open source enthusiast
 - E.g. OpenGL and Vulkan samples on GitHub
- Some contributions to glTF 2.0
 - 3D assets, specification tweaks
- Co-founded UX3D with Thomas Kress
 - Real-time 3D user experiences

The screenshot shows the GitHub profile of Norbert Nopper (McNopper). The profile includes a profile picture of a person wearing a white helmet and goggles. The bio states "Norbert Nopper McNopper" and "Block or report user". The location is "Germany" and there are links to "Sign in to view email" and "http://nopper.tv". The profile shows 12 repositories, 8 stars, 275 followers, and 4 following. The "Popular repositories" section lists: OpenGL (OpenGL 3 and 4 with GLSL, 720 stars, 247 forks), Vulkan (Vulkan examples using Vulkan Tools (VKTS), 158 stars, 32 forks), OpenGL_ES (OpenGL ES 2.0, 3.0 and 3.1 with GLSL, 111 stars, 42 forks), GraphicsEngine (OpenGL 4 Graphics Engine, 96 stars, 25 forks), GLUS (GLUS for OpenGL, OpenGL ES and OpenVG, 87 stars, 33 forks), and EGL (EGL 1.5 implementation for Windows and X11 supporting OpenGL, 21 stars, 13 forks). The "669 contributions in the last year" section shows a calendar heatmap with a legend for "Less" and "More" contributions. The "Contribution activity" section shows a timeline from April 2017 to 2016.



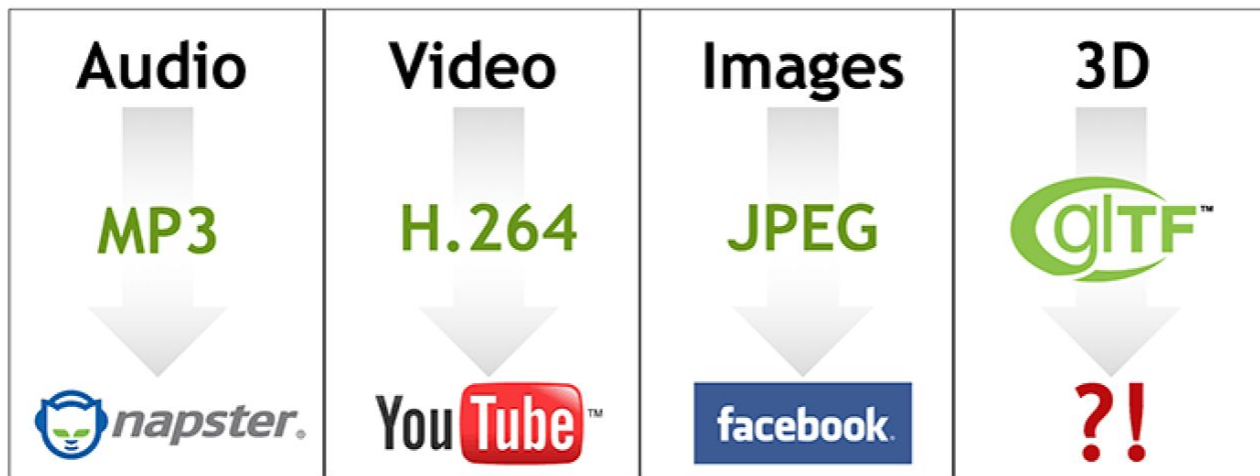


What is glTF 2.0?

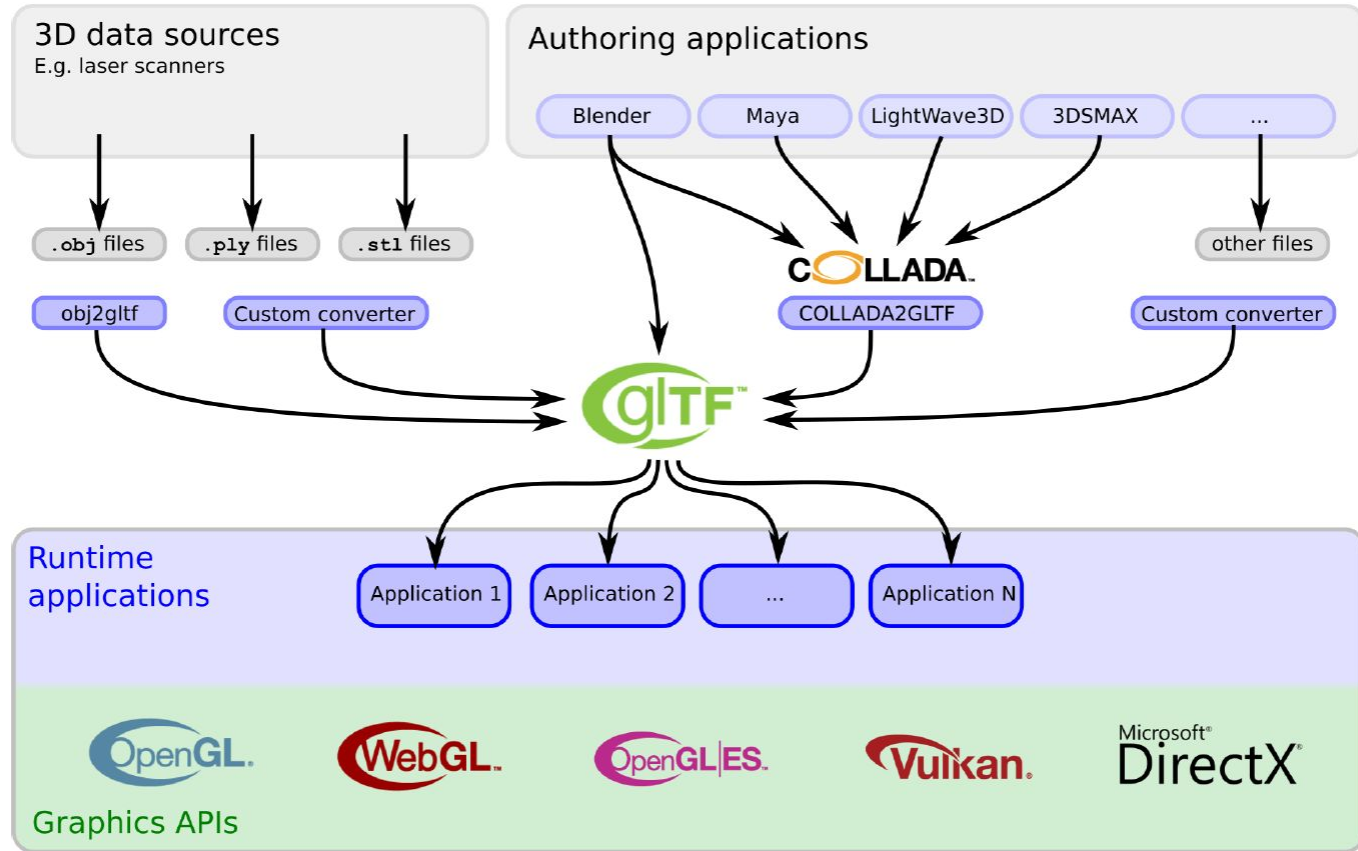
- *Cherry picking* from Khronos glTF Webinar, February 2017
<https://www.khronos.org/news/events/webinar-khronos-gltf>
- Added some latest changes/additions

What is glTF?

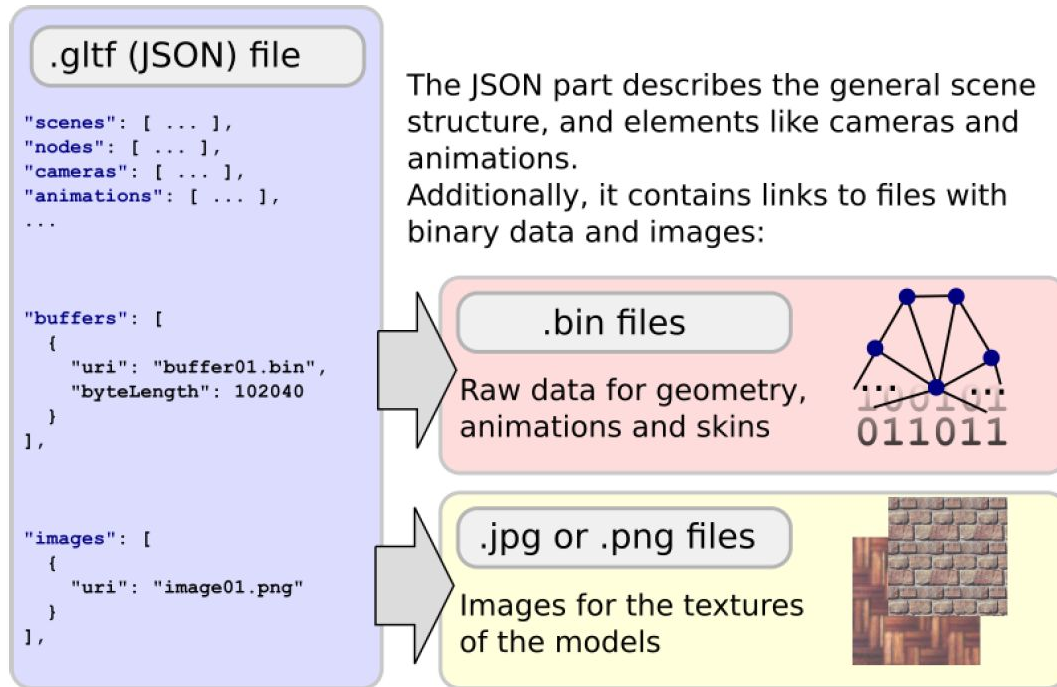
- glTF is the *GL Transmission Format*
 - An open standard, developed by Khronos: [khronos.org/gltf](https://www.khronos.org/gltf)
- Designed for the efficient transfer of 3D assets
 - Versatile, compact, and easy to process by the client



Introduction and motivation

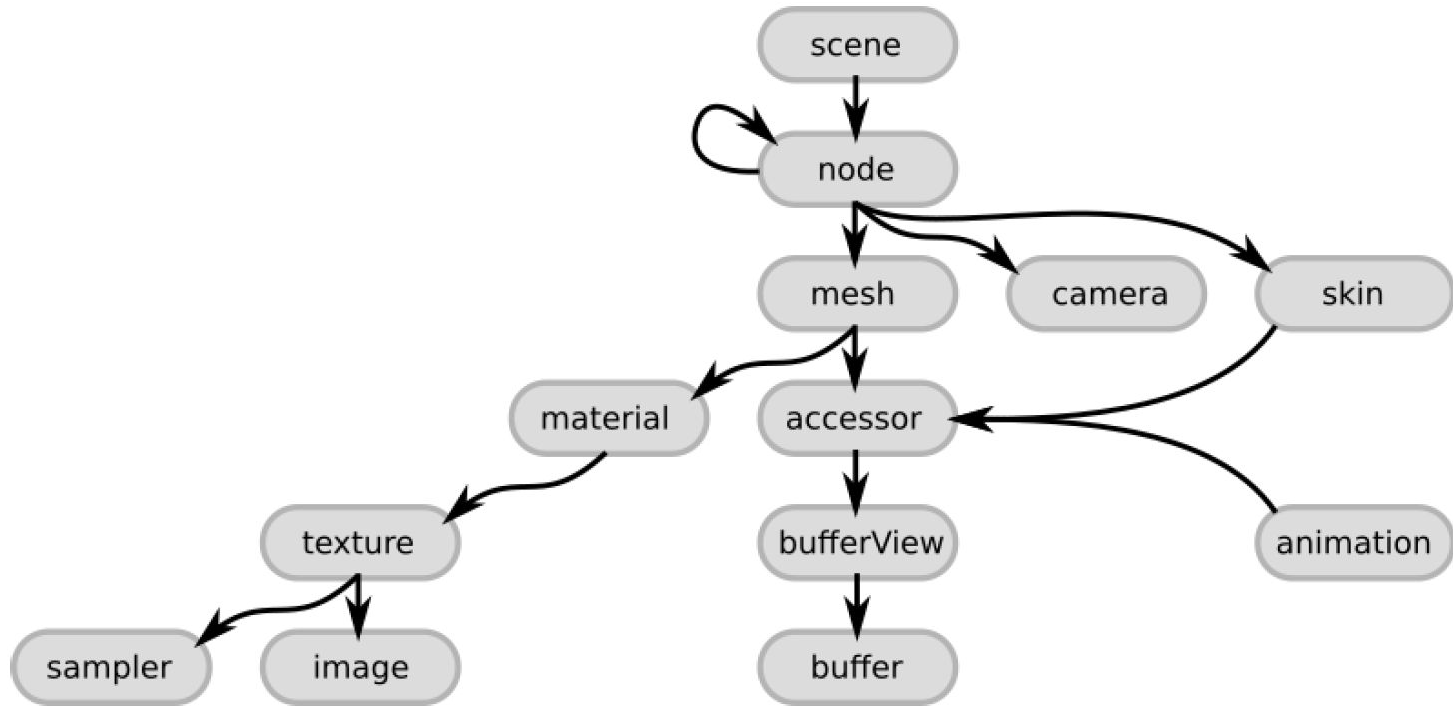


Basic file structure



- External resources *can* be embedded into JSON, as data URIs

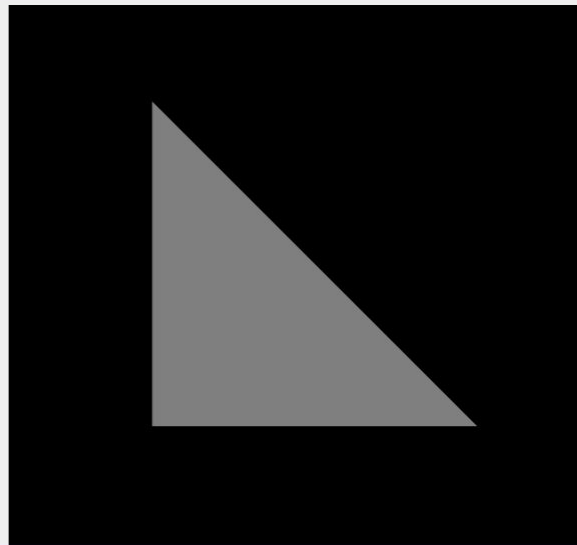
Basic JSON structure




```
{
  "scenes" : [ { "nodes" : [ 0 ] } ],
  "nodes" : [ { "mesh" : 0 } ],
  "meshes" : [ {
    "primitives" : [ { "attributes" : { "POSITION" : 0 } } ]
  }
],
"buffers" : [
  {
    "uri" : "data:application/octet-stream;base64,AAAAAAAAAAAAAAAAACAPwAAAAAAAAAAAAAAAAAgD8AAAAA",
    "byteLength" : 36
  }
],
"bufferViews" : [
  {
    "buffer" : 0,
    "byteOffset" : 0,
    "byteLength" : 36,
    "target" : 34962
  }
],
"accessors" : [
  {
    "bufferView" : 0,
    "byteOffset" : 0,
    "componentType" : 5126,
    "count" : 3,
    "type" : "VEC3",
    "max" : [ 1.0, 1.0, 0.0 ],
    "min" : [ 0.0, 0.0, 0.0 ]
  }
],
"asset" : { "version" : "2.0" }
}
```

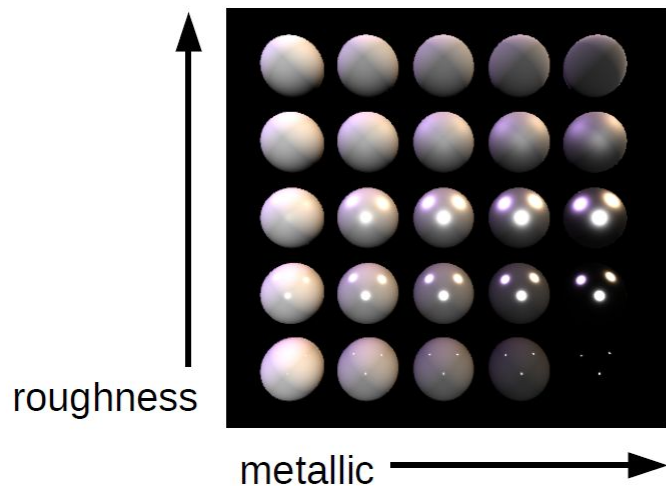
This is a complete
glTF asset
with an embedded buffer

(Supposed to be *the*
minimal glTF asset)



Materials

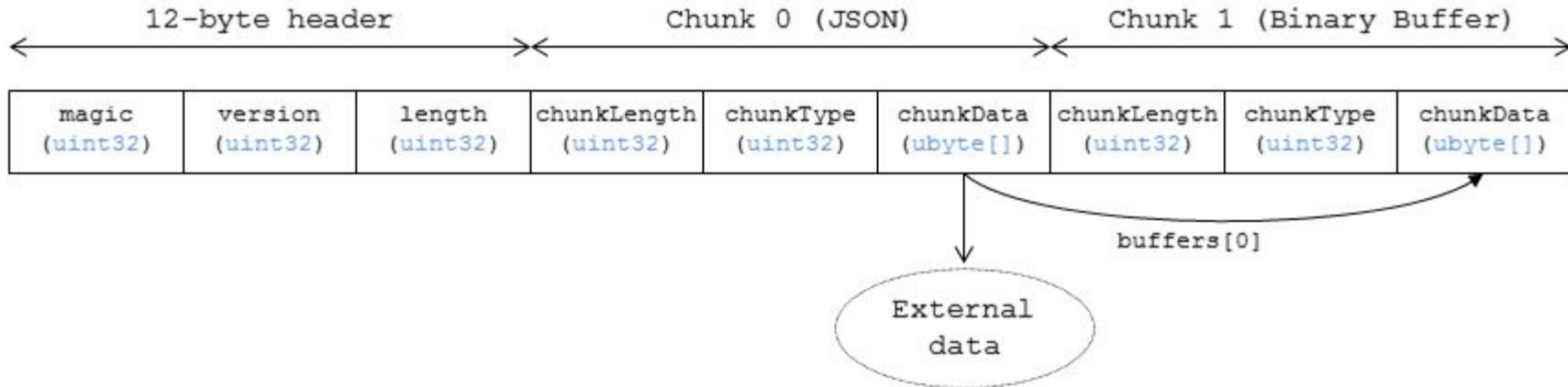
- A `material` stores material parameters
 - For example: Metallic-ness and roughness
 - Can also be given as textures
- Physically based rendering (PBR) part of glTF 2.0
 - Coordinated effort to define a standard for PBR!





What is glTF 2.0?

- GLB: glTF JSON, images and buffers stored in one binary file






What is glTF 2.0?

- Even not final, all accessible now!
- Specification:
<https://github.com/KhronosGroup/glTF/tree/2.0/specification/2.0>
- Examples:
<https://github.com/KhronosGroup/glTF-Sample-Models/tree/master/2.0>
<https://sketchfab.com/features/glTF>
- Exporter (just a few):
Unity <https://github.com/sketchfab/Unity-glTF-Exporter>
Blender (soon) <https://github.com/KhronosGroup/glTF-Blender-Exporter>



Killer features of glTF 2.0 (my top five)

- GLB
 - One file with all assets embedded, so easy to deploy.
- Physically based material(s)
 - PBR metal roughness in core glTF 2.0.
- Extensions
 - New features does not break compatibility.
- Graphics API independent
 - E.g. works with WebGL, OpenGL (ES), Vulkan and DirectX.
- Industry and open source community driven
 - Many tools/exporters right from start.
 - E.g.  **Sketchfab** supports glTF 2.0.

glTF™ glTF 2.0 demos

- *BoomBox* in three.js



glTF™ glTF 2.0 demos

- *BoomBox* in VKTS





Summary

- glTF 2.0 is open, standard 3D transmission format
- Specification is close to final
- Graphics API independent
- Many tools and assets from industry and open source developers

<https://github.com/KhronosGroup/glTF/issues/867>



Questions?