



# Ecosystem Update
## June 2018

Saurabh Bhatia, Microsoft
@iamSBTron

# glTF - Cross-Platform 3D Asset Transmission

| Audio | Video | Images | 3D |
|---|---|---|---|
| MP3 | H.264 | JPEG | glTF |
| napster. | YouTube | facebook | New market opportunities for 3D content creation and deployment! |

All glTF spec development
on open GitHub:
https://github.com/KhronosGroup/glTF

glTF

- Compact to Transmit ✓
- Fast to Load ✓
- Describes Full Scenes ✓
- Runtime Neutral ✓
- Open and Extensible ✓

OpenGL Transmission Format
Efficient transmission of 3D scenes and assets

glTF

WebGL

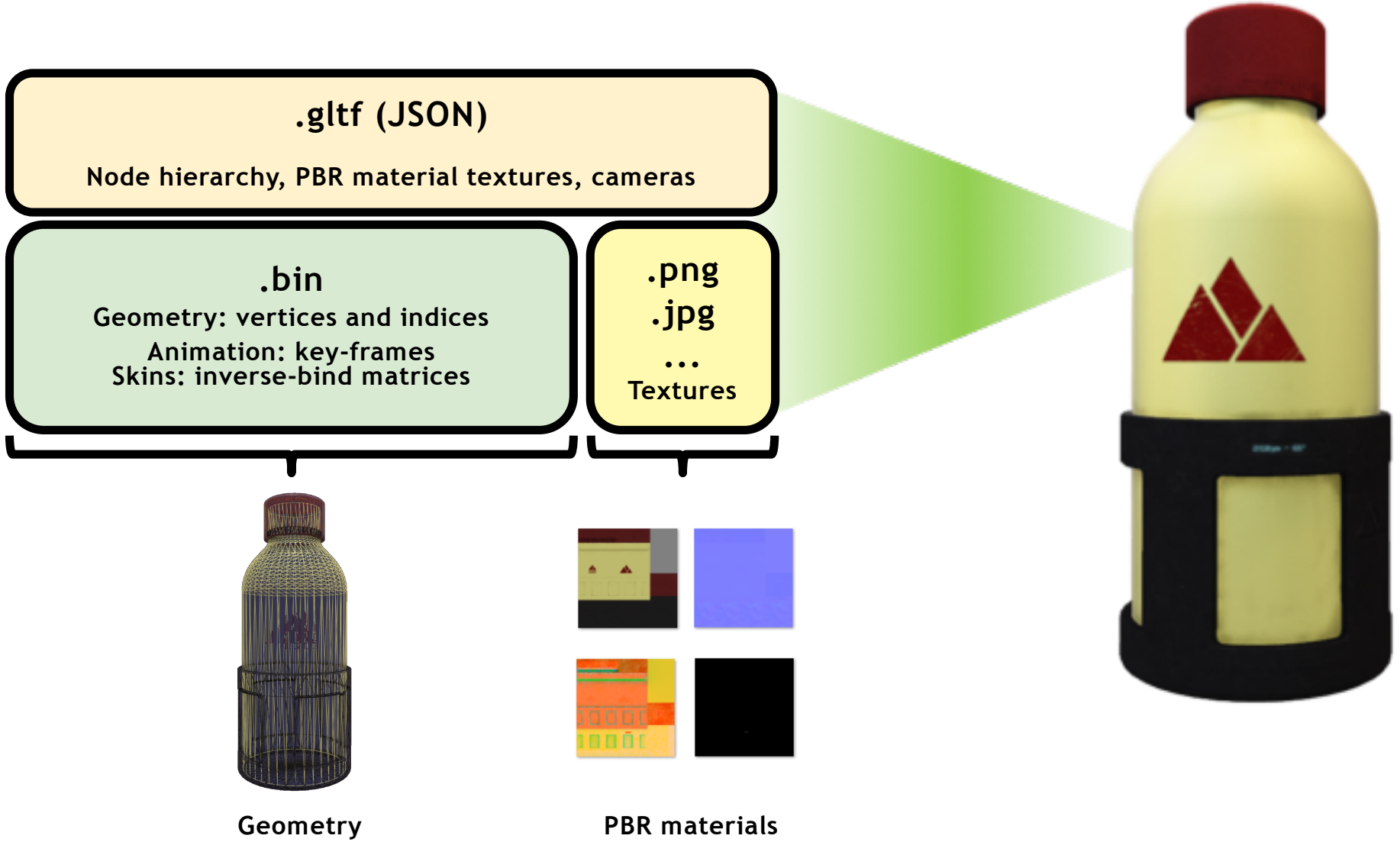
**glTF 1.0 - Primarily for WebGL**
Uses GLSL for materials
Released December 2015

WebGL Vulkan

**glTF 2.0 - Physically Based Rendering!**
Metallic-Roughness and Specular-Glossiness Materials
Rendering API independence
Released June 2017

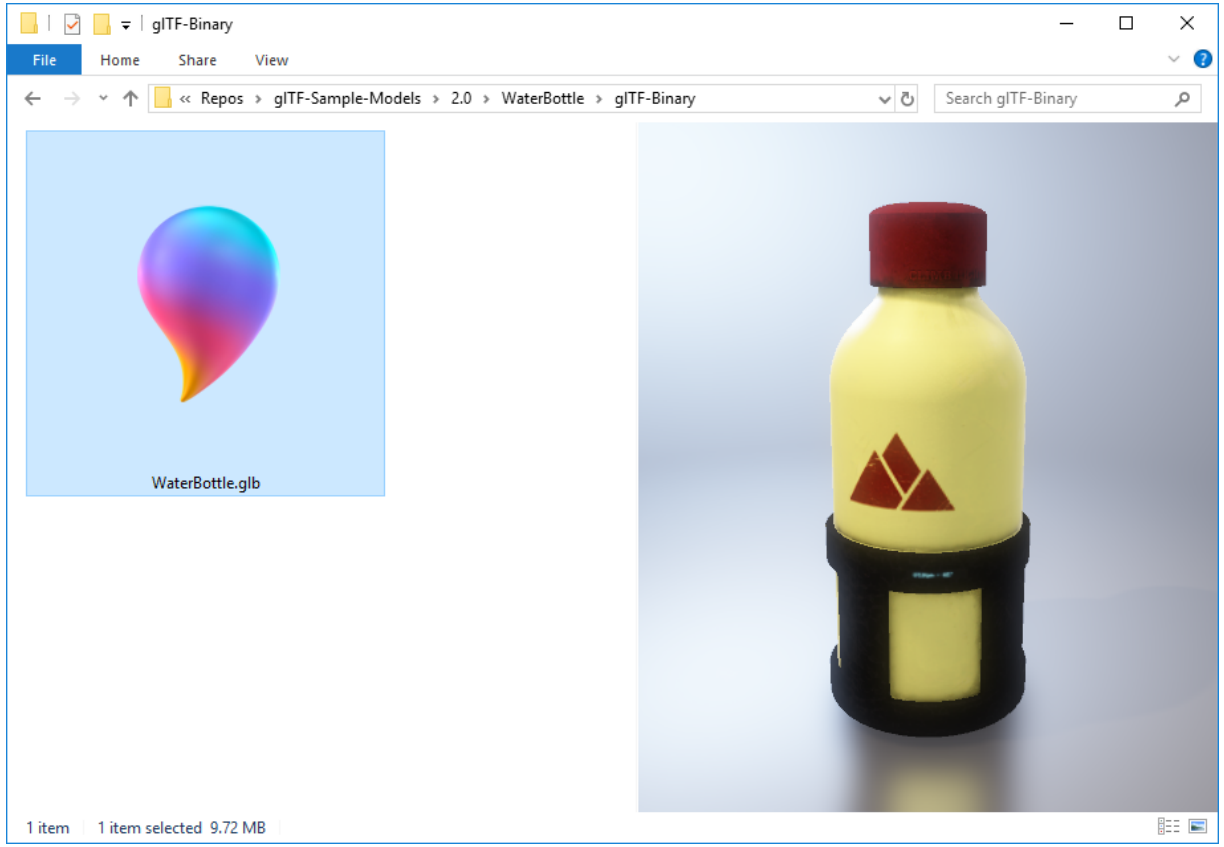
KHRONOS GROUP

# glTF 2.0 Scene Description Structure

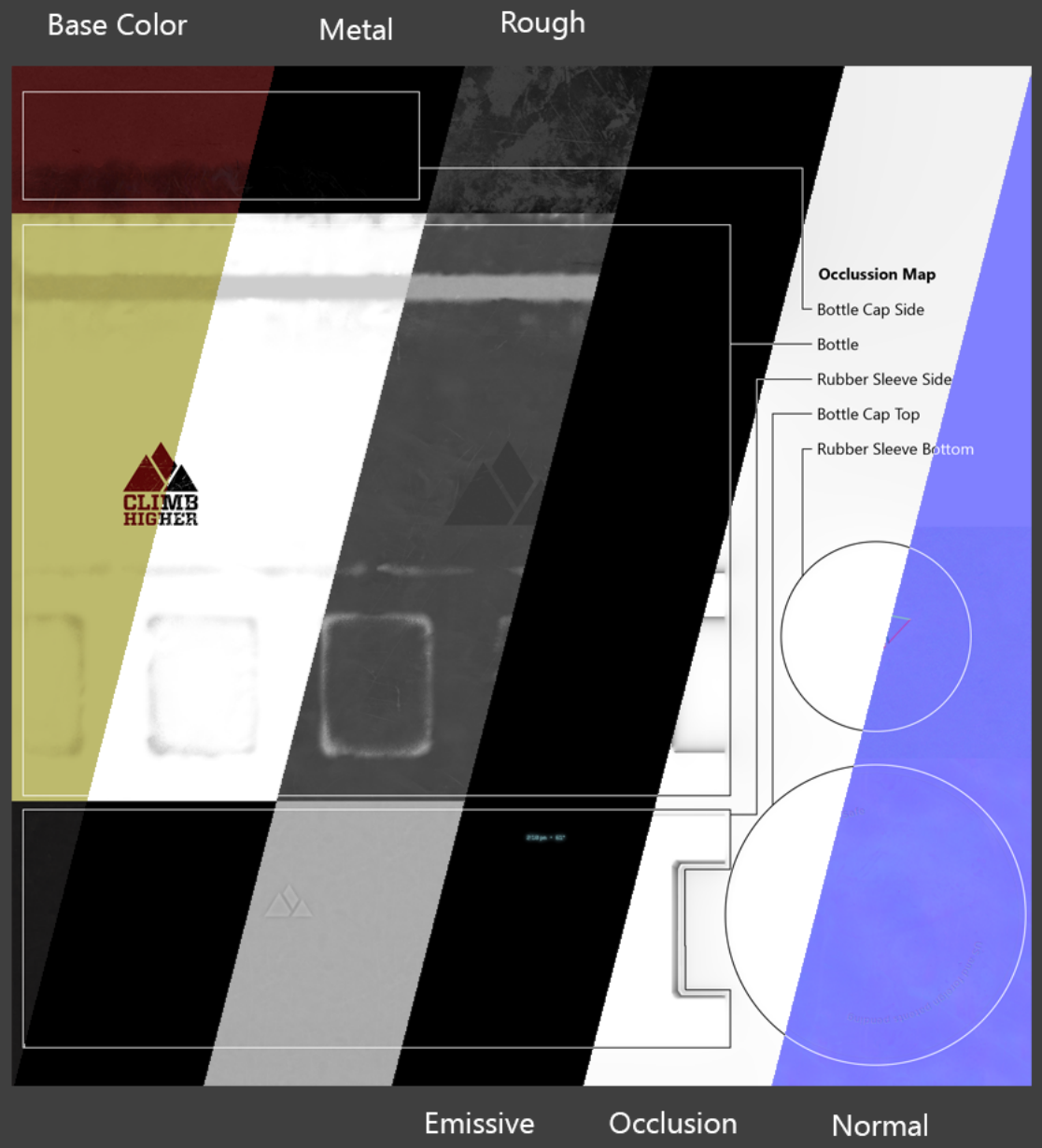


# glTF Binary - .glb

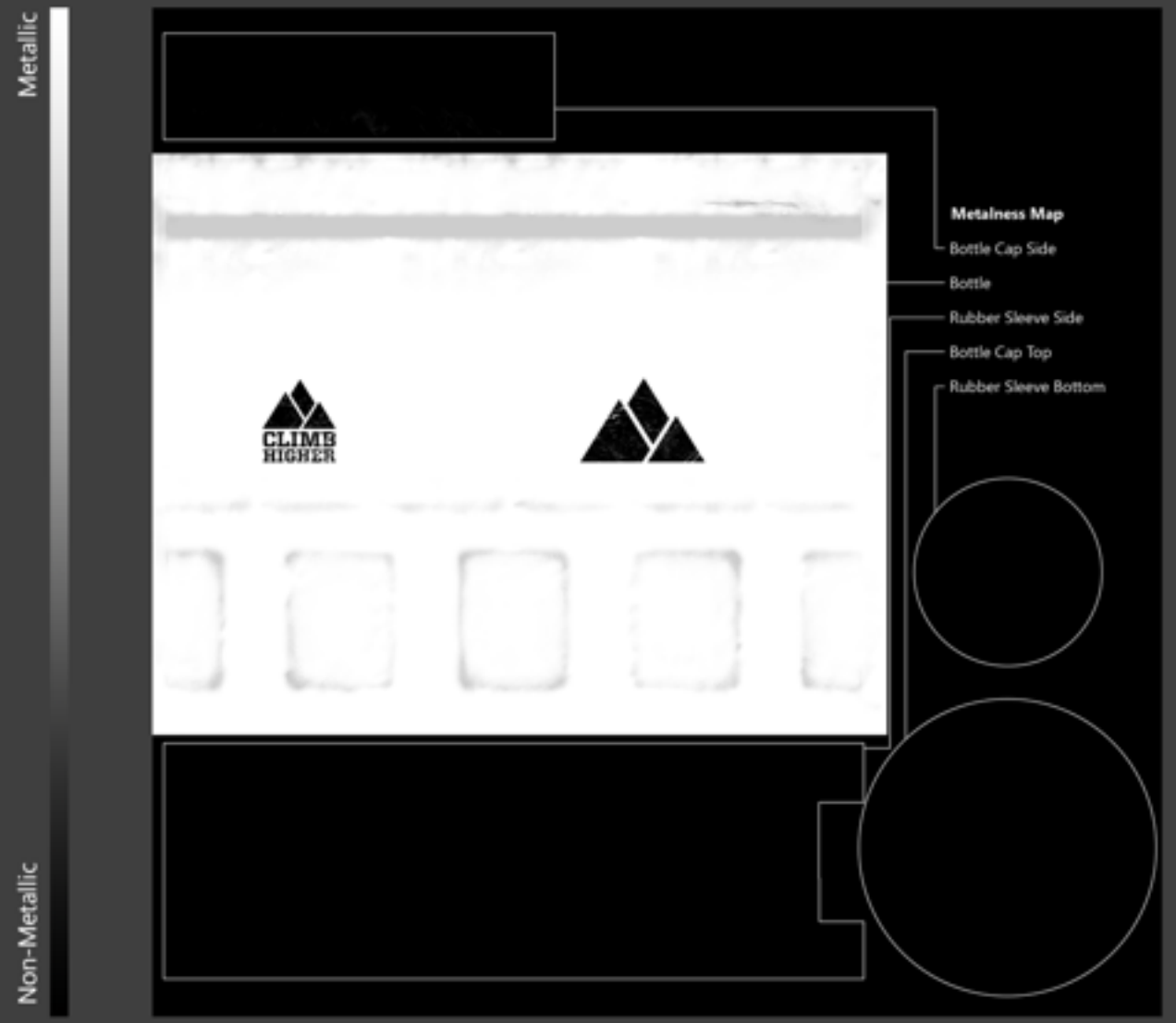
- Binary version of glTF
- Single file that encapsulates all geometry and textures
- Ideal for end users to interact with a single file



# PBR Materials

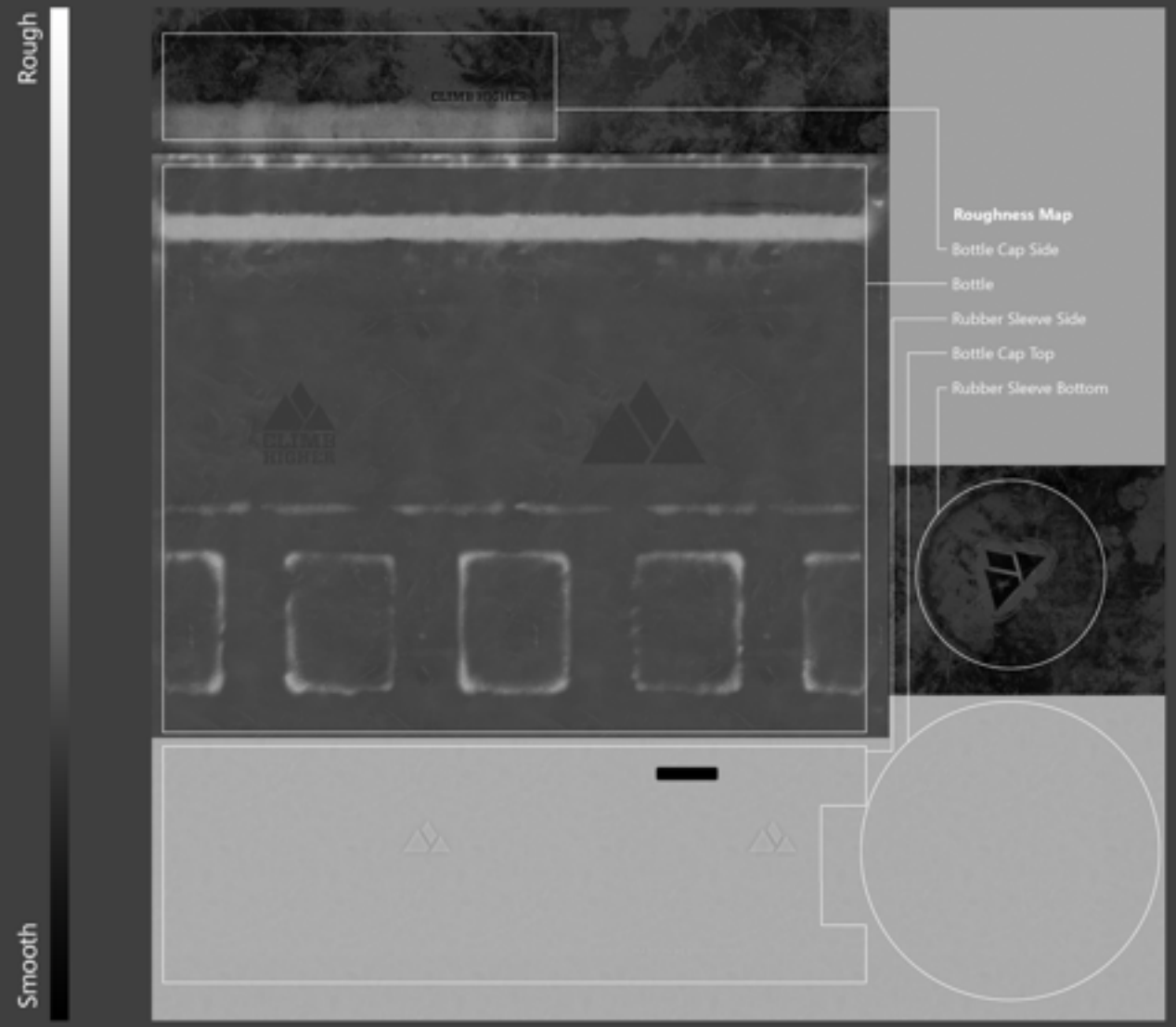


# PBR Materials



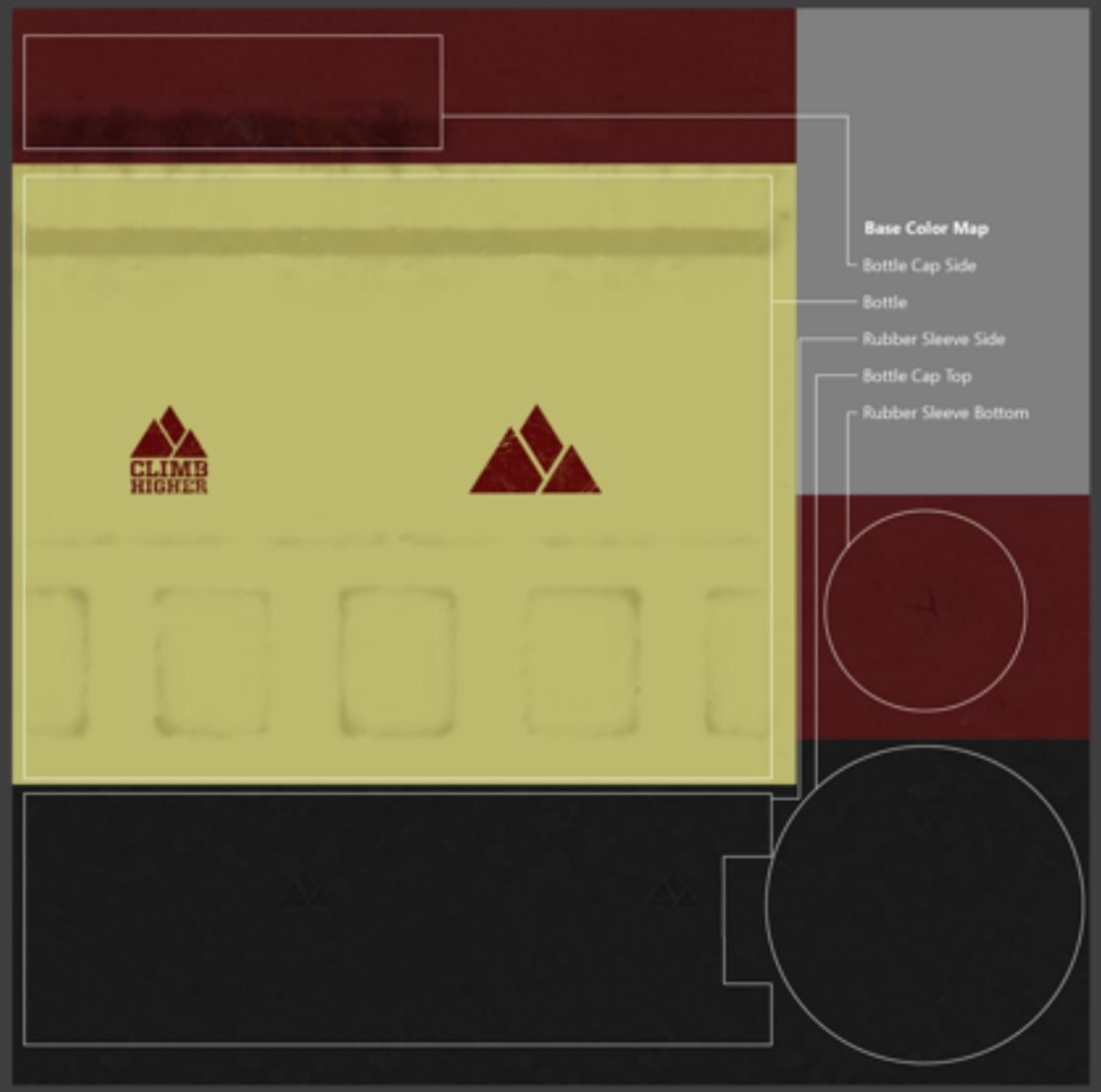

Metalness Map

# PBR Materials



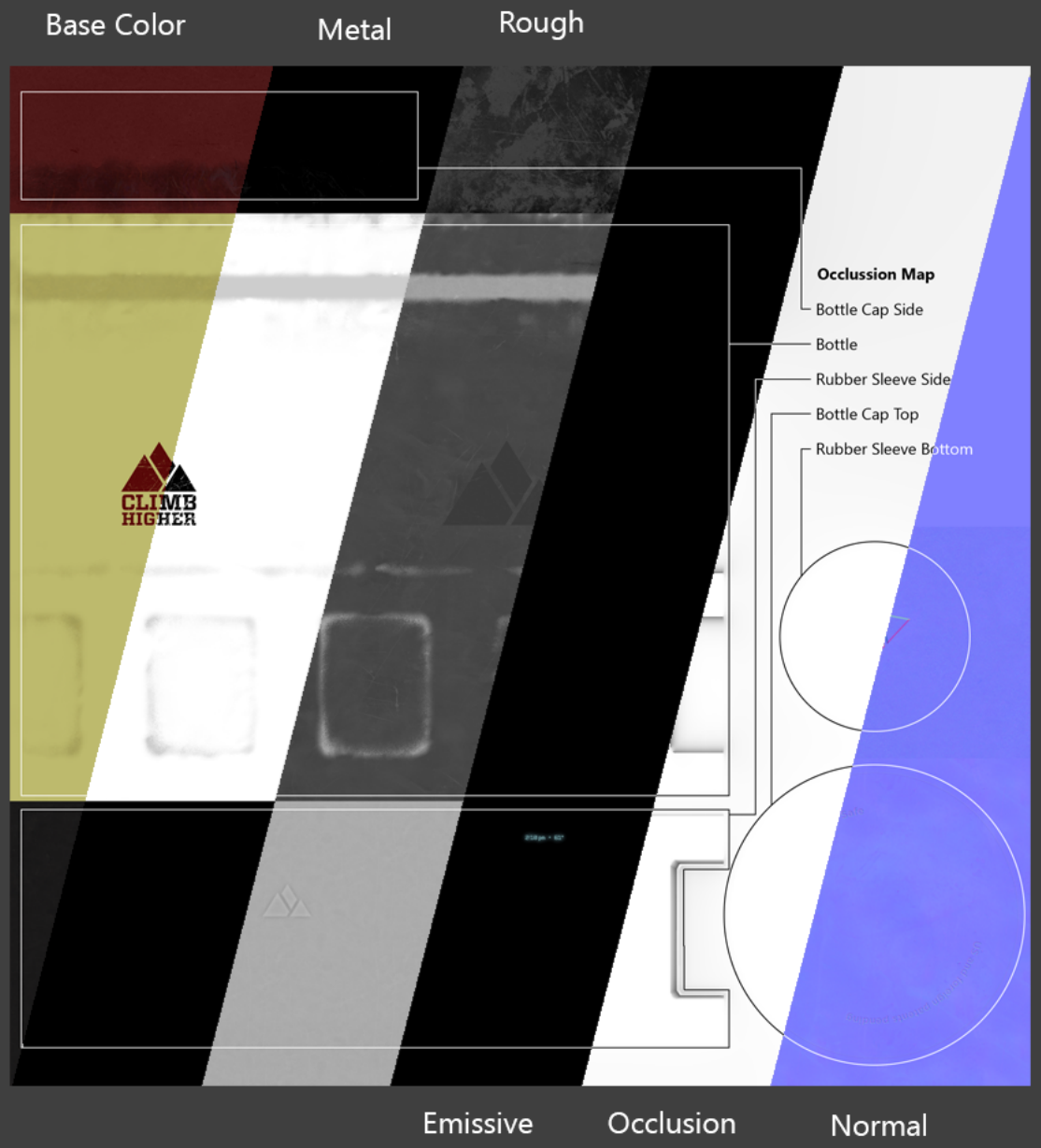

Roughness Map

# PBR Materials

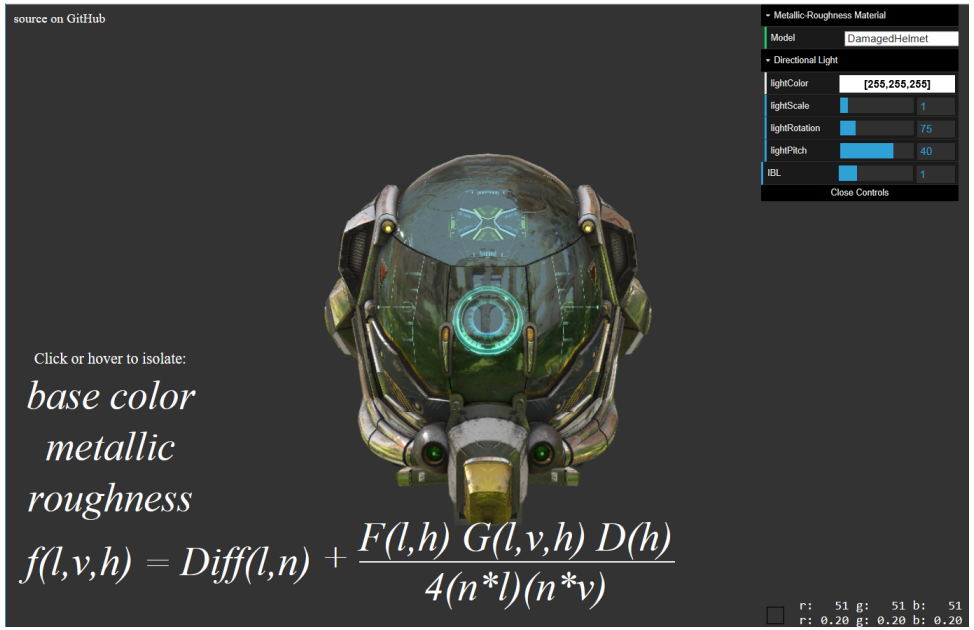

Base Color Map

# PBR Materials

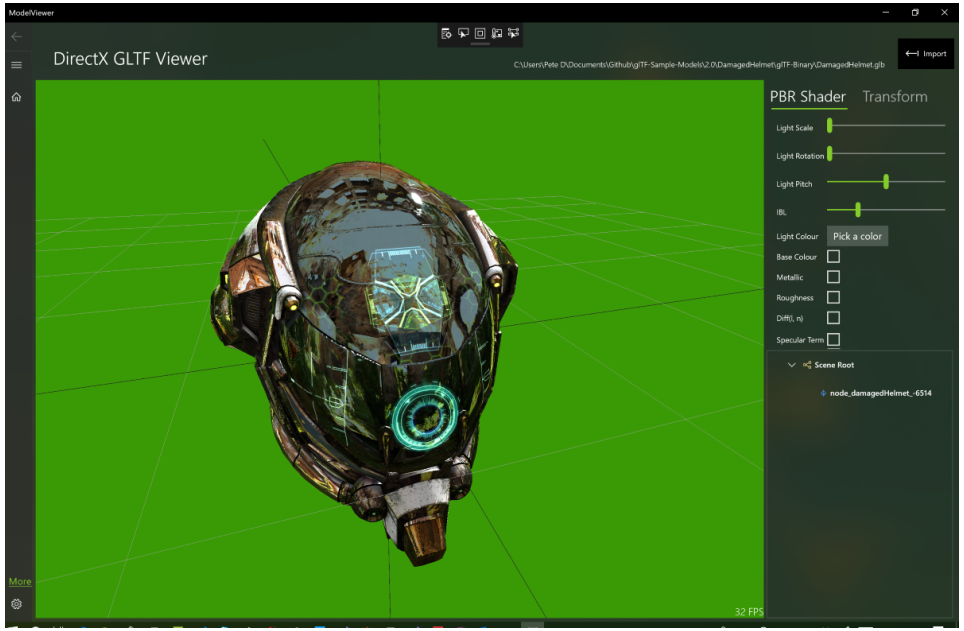


# PBR sample apps

https://github.com/KhronosGroup/glTF-WebGL-PBR




https://github.com/Microsoft/glTF-DXViewer



# glTF 2.0 Reference Guide

- Graphics API neutral
  - Proven by engine implementations using WebGL, Vulkan and Direct3D
- Physically Based Rendering (PBR) material definitions
  - Material information stored in textures
- Improvements
  - Binary glTF (.glb) in core spec
- Morph Targets
  - Enhanced animation system

- Download the glTF 2.0 Reference Guide!
  - https://www.khronos.org/files/gltf20-reference-guide.pdf

# Happy Birthday glTF 2.0!

KHRONOS GROUP
CONNECTING SOFTWARE TO SILICON

Developers ▾ Conformance ▾ Membership ▾ News & Even About ▾

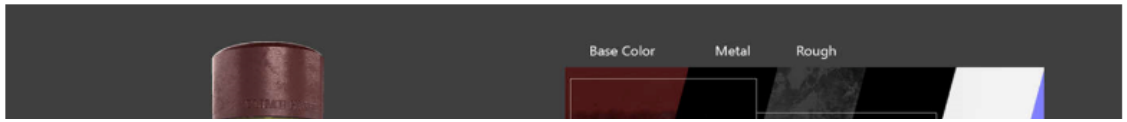
## Khronos Releases glTF 2.0 Specification

*Runtime 3D Asset Delivery Format Enhanced with*
*Platform Independent Physically Based Rendering*

**June 5, 2017 – 6:00 AM Pacific Time -- Brisbane, Australia – The Khronos™ Group,** an open consortium of leading hardware and software companies, announces from the Web3D 2017 Conference the immediate availability of the finalized glTF 2.0 specification incorporating industry feedback received from developers through the provisional specification that was made available for review on GitHub.

The release of glTF 2.0 delivers a significant upgrade to glTF 1.0, an extensible, runtime neutral, open standard format for real-time delivery of 3D assets, which describes full scenes with compact transmission and fast load time. In response to major functionality requests from the developer community using glTF 1.0, the release of glTF 2.0 adds Physically Based Rendering (PBR) for portable, consistent description of materials. In glTF 1.0, a material was defined with a GLSL shader, which suited WebGL, but was problematic when importing a glTF model into a Direct3D or Metal application. Through using PBR, visually arresting glTF 2.0 models are now consistently portable to any rendering API. A PBR material is defined by a few concise parameters that can be used to generate shaders for any rendering API. glTF 2.0 defines a simple to implement, but powerful, PBR model that provides high-quality materials, and yet, is scalable to suit the capabilities of different classes of platform and device.

"glTF's momentum continues to grow, with increasing adoption from tools, players and applications throughout the industry," said Neil Trevett, Khronos president and glTF Chair. "In February we released the glTF 2.0 developer preview and made an open call for feedback. Since then we have had enthusiastic community input that has significantly influenced our preparation for the final spec release. We now look forward to a continued industry engagement to expands glTF's capabilities - for example with advanced texture and geometry compression extensions. We believe that glTF 2.0 will help the industry move towards PBR-based materials in many application areas."




Press Details

Short Link: https://khr.io/ha

Chinese Translati

Japane Translation

Korean Translation

Media Cont

Alex Crabb,
Communi
alex@caste
401-792-708

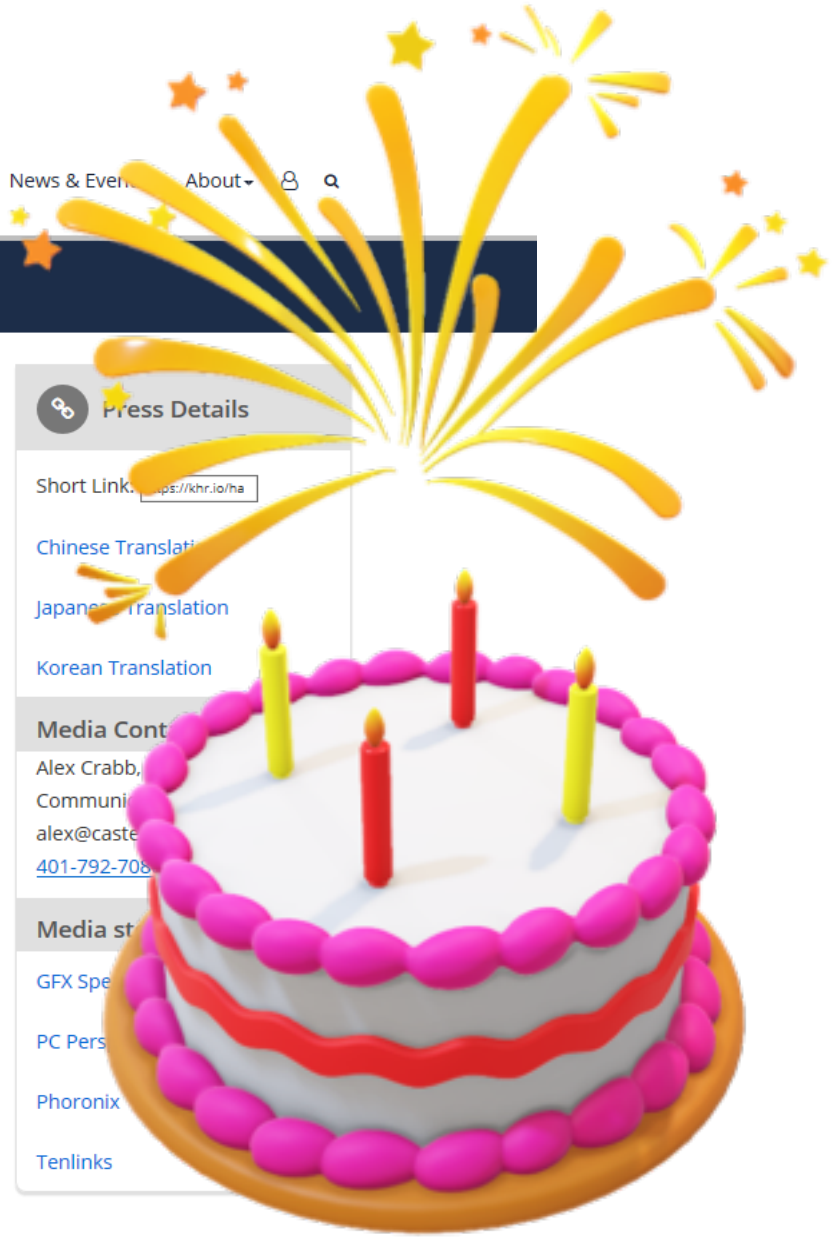
Media st

GFX Spe

PC Pers

Phoronix

Tenlinks



# glTF Momentum

## Creation Tools

blender
Archilogic
PiXYZ SOFTWARE
SketchUp
MINECRAFT
Adobe Dn
SUBSTANCE PAINTER
MARMOSET TOOLBAG
Sony 3D Creator
UNBOUND
Modo
Oculus
medium
Paint 3D
instant3Dhub
SIMPLYGON
Assimp Open Asset Import Library
gltf-vscode
Collada2gltf
FBX2glTF
glTF-validator
glTF-asset-generator

## 3D Content

Sketchfab
150K glTF Models in Creative Commons!
Poly poly.google.com
Remix 3D

Discover

Create

glTF™

Experience

Drive Demand

## Users

Bentley
otoy
Continental
OGC
NVIDIA.

## Apps and Engines

UNREAL ENGINE
unity
GODOT Game engine
PLAYCANVAS
three.js
babylon.JS
A-FRAME
React VR
CESIUM
AUTODESK FORGE
UX3D ENGINE
xeogl
Office
Windows Mixed Reality Home
JANUSVR

KHRONOS GROUP

# Focus on glTF ecosystem robustness

- **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/bghgary/glTF-Asset-Generator

# glTF Roadmap discussions

Balance building a widely supportable and reliable ecosystem vs. adding more features

## Ratified extensions ready for implementors

- Google Draco extension for compressed geometry
- Unlit Material

## Ongoing work

- KHR_texture_transform – offset, scale, rotation
- KHR_lights
- KHR_technique_webgl
- KHR_environments
- Universal Texture Compression – Optimized transmission format with efficient local expansion to any GPU format

# Promoting extensions

- **Who will use your extension?**
  - My application only
    - Do a vendor extension. Register your PREFIX by submitting an Issue.
  - Multiple applications from multiple vendors
    - Do a multi-vendor “EXT” extension
  - Broadly applicable across all apps/platforms
    - Propose a “KHR” extension
    - Best practice: “KHR” extensions need at least two implementations to ensure the proposal works correctly
    - “KHR” extensions are discussed and agreed by glTF working group and covered by Khronos Intellectual Property Framework

- **When designing your extension always have a fall back to core spec**
  - Avoid breaking compatibility with broader ecosystem
  - If you choose to not have a fall back then list your extension in *extensionsRequired*


# See you on GitHub!

https://github.com/khronosgroup/gltf