

# Overview and Comparison of Scene Description Formats for Immersive Applications

Thomas Alder

thomas.alder@otoy.com



*Based on the white paper "An Overview of Scene Description Formats" published by IDEA*

# Outline

- Purpose
- Scene descriptions & typical workflow
- Quick overview of FBX, ABC, glTF, USD, ITMF
- Conclusion

# Purpose

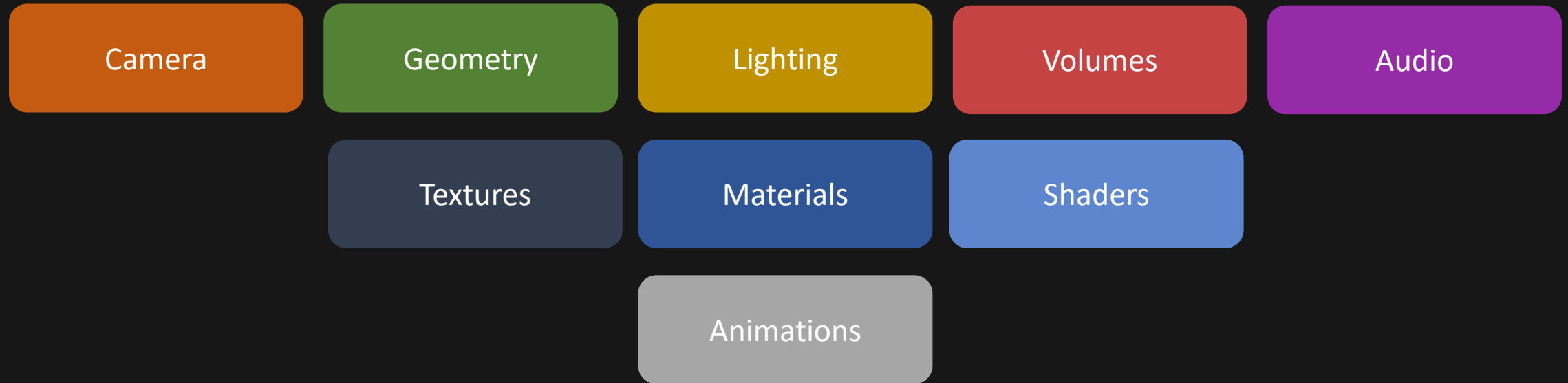


# Immersive Applications

- Immersive media embraces more than 2D experiences
- Immersive devices based on rays or waves rather than raster-based
- Displays will dictate application implementations
- 3D scenes are integral to the immersive experience

# Scene Descriptions

An ideal scene may contain:



A format may also support streaming, compression, or extensions

# High-Level Workflow

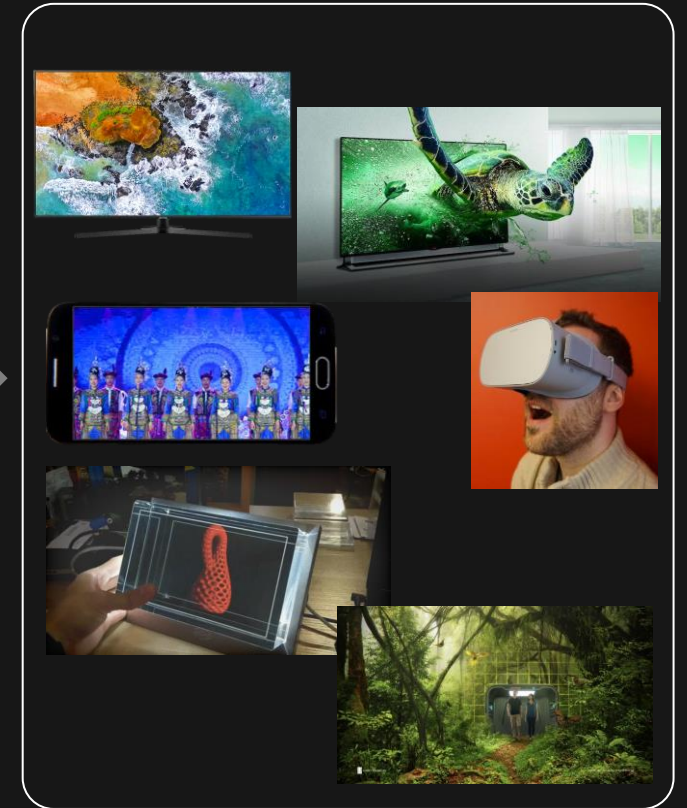
DCC Tools



Renderers/Game Engines



End Displays



# Filmbox (FBX)

3D data interchange scene graph format

Interoperability between DCCs, asset storage, animation processing, general packaging

Compression, extensibility, and streaming not defined



Camera: Basic, stereo

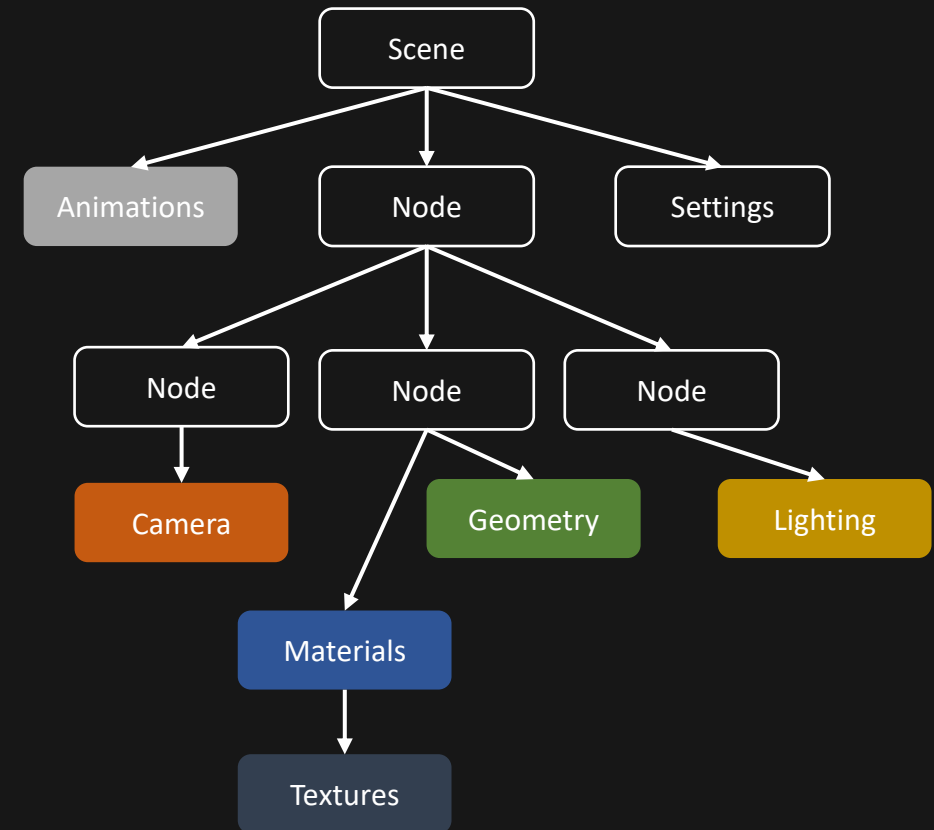
Geometry: meshes,  
NURBS, patches, lines

Lighting: Point,  
directional, spot, area,  
volume

Materials: Phong,  
Lambertian Surface  
Shading Models

Animations: stacks, layers,  
curve modes, curves,  
curve keys

Textures: File, layered





# Alembic (ABC)

Interchangeable framework and geometry caching format

Distills skeletal rigs/networks of computations

Not a “live” scene graph

Compression and streaming not defined

Extensible by open source nature



Camera: Basic

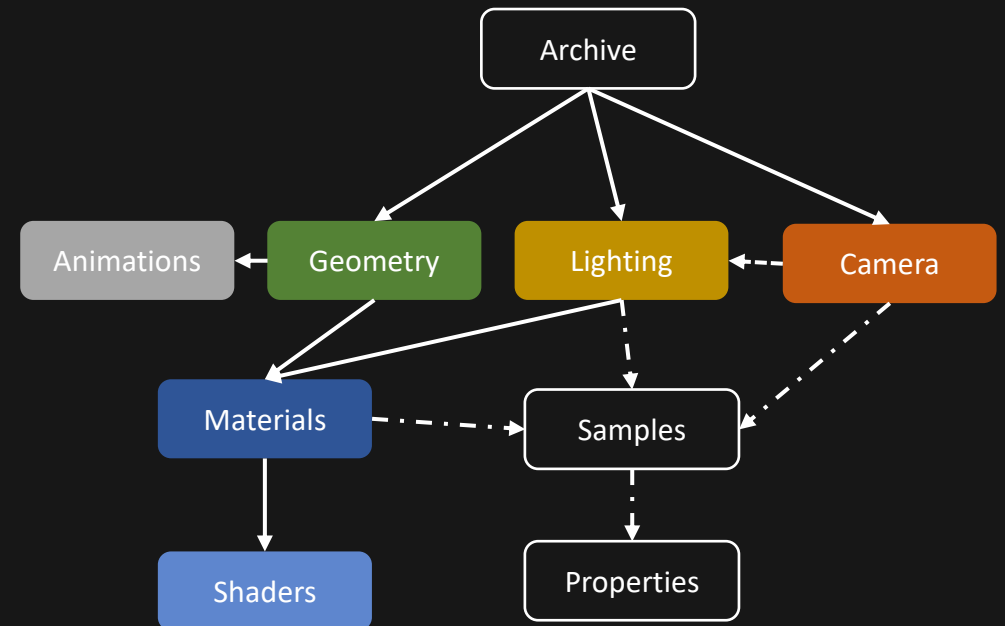
Geometry: meshes,  
NURBS, parametric  
curves, particles

Shaders:  
application/renderer  
targeted

Lighting: Expressed by  
material input

Animations: distilled,  
time-sampled vertex pos.  
and transforms

Materials: Container  
defining props for specific  
target



# Universal Scene Description (USD)

Composition engine for consistent interchangeability

Nondestructive layering and referencing of files

Loose usd files are archived in usdz

Draco mesh compression, custom plugins and individual schemas, “possibly streamable”



Camera: Basic

Geometry: meshes,  
NURBS, patches, curves,  
points, shapes

Lighting: Disk, distant,  
dome, geometry, rect,  
sphere

Materials: Preview  
surface material

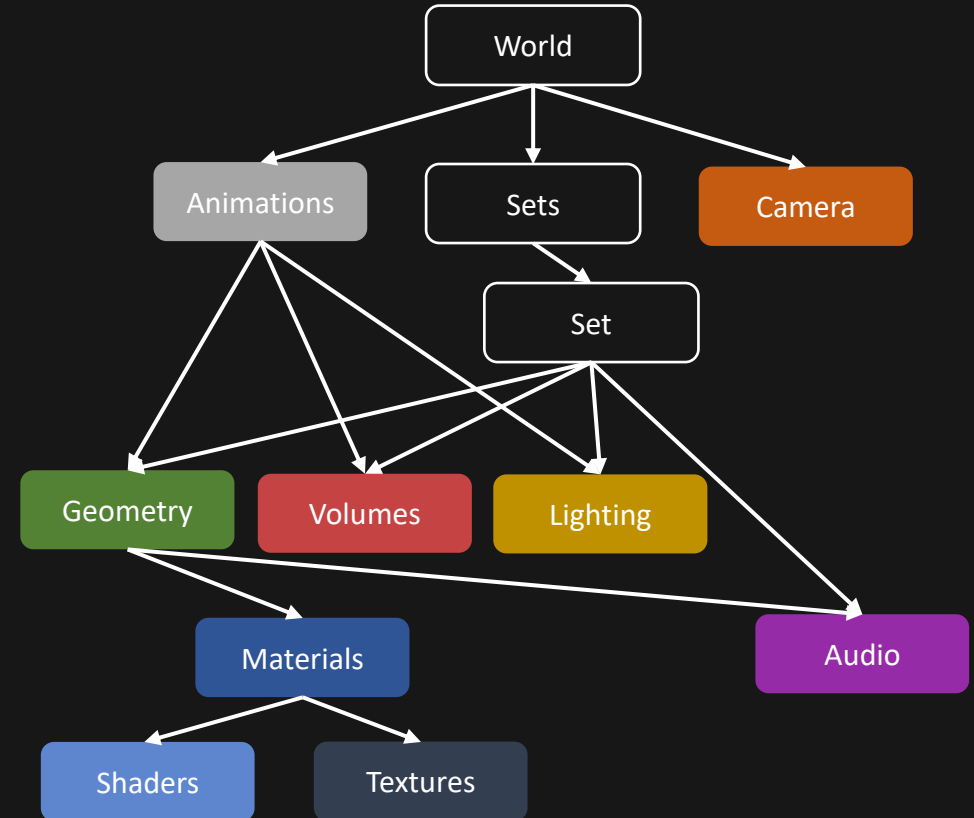
Textures: Image-based

Shaders: referenced per  
application

Volumes: OpenVDB,  
Field3D

Audio: Spatial (proposed)

Animations: time-  
sampled attribute values



# GL Transmission Format 2.0 (glTF)

Runtime asset delivery format

“Last-mile” of content transmission

Work being done in MPEG-I Part 14 to extend glTF 2.0 to include video and audio support

JSON scene representation, binary data

GLB container

Draco mesh compression, extremely extensible, streamable through its buffer design



Camera: Perspective,  
orthographic

Geometry: Meshes, skins,  
morphs

Lighting: directional,  
point, spotlights, image-  
based

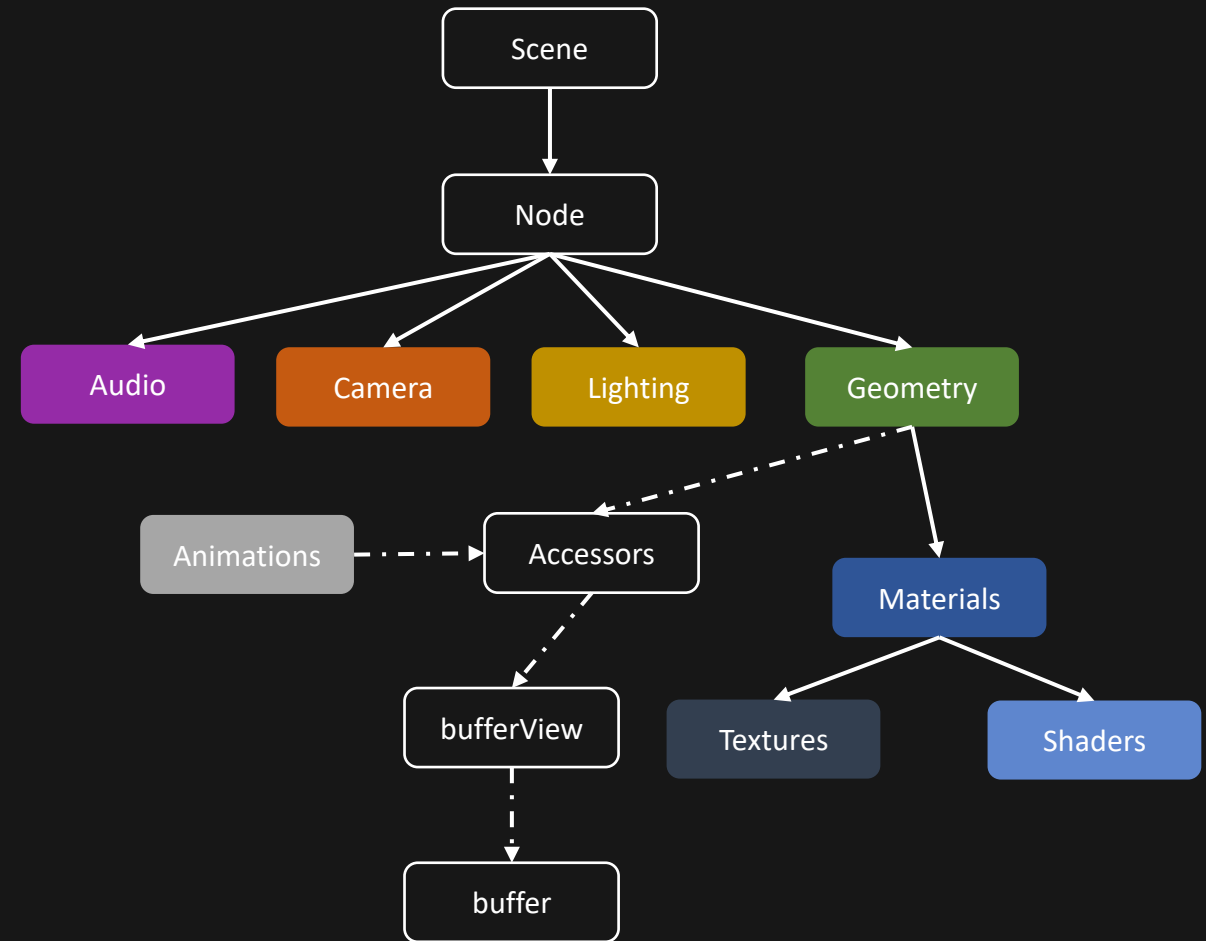
Materials: PBR, PBR Next

Textures: image-based,  
samplers

Shaders: GLSL in  
KHR\_techniques\_webgl  
extension

Audio: spatial in progress

Animations: node  
transforms, morph target  
weights



# Immersive Technologies Media Format (ITMF)

Ray traced, immersive media focus

Packages 3D scene description and mesh formats for usage in content creation and network transmission pipelines

XML Scene Graph, BML Container

Compression, streaming worked on within IDEA

Extensible through OSL shading, Lua Scripted Node Graphs

Camera: Thin Lens,  
panoramic, baking, OSL

Geometry: OBJ, FBX, ABC,  
glTF, USD

Lighting: Emission,  
environments, spectron

Materials: PBR, non-PBR  
materials, layered

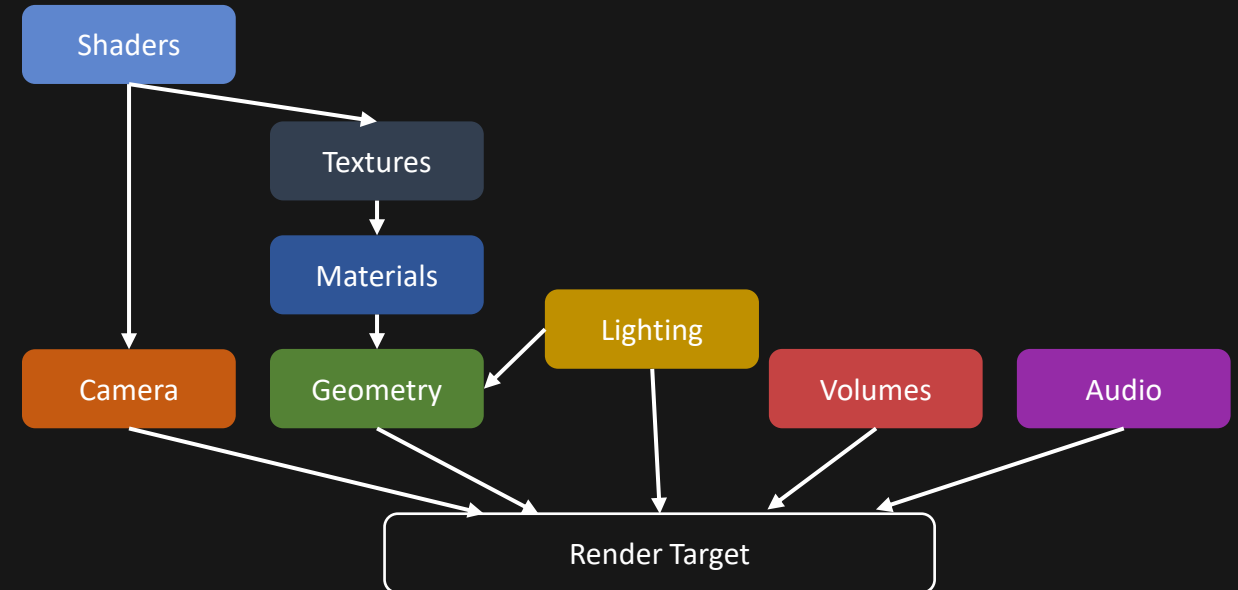
Textures: Image-based,  
procedural, geometric,  
operators

Volumes: Mediums,  
OpenVDB, vectron

Audio: Spatial (in  
progress)

Shaders: Open Shading  
Language

Animations: supported  
formats, time-sampled  
animators





# Review of Purposes

FBX	Interoperability between DCCs, asset storage, animation processing, and general packaging.
Alembic	Interchangeability and geometry caching for distilled scenes.
USD	Nondestructive composition of files for collaboration earlier throughout the content creation pipeline.
glTF	Minimize asset size and runtime processing in the last mile of 3D mesh publication and delivery.
ITMF	End-to-end support for immersive media content creation, packaging, and network transmission.

# Conclusion

Each scene graph has advantages and disadvantages

Intended to target different workflows, content types, and end use cases

Two scopes

- Interoperability and interchange
- Transmission

Immersive applications require a master and distribution format rich in 3D scene information

Questions? Feedback appreciated!  
[comments@ImmersiveAlliance.org](mailto:comments@ImmersiveAlliance.org)  
[thomas.alder@otoy.com](mailto:thomas.alder@otoy.com)