

MaterialX and MaterialX Web

Bernard Kwok & Sebastian Dunkel

MaterialX : Core and VizX



MaterialX

- Industry standard shader and material representation
- Reusable format to support data interoperability
- Node networks for procedural and texture patterns and shading models
- Accepted into the [ASWF](#) (Academy Software Foundation).
 - Autodesk part of TSC
- [Khronos](#) adoption under discussion.

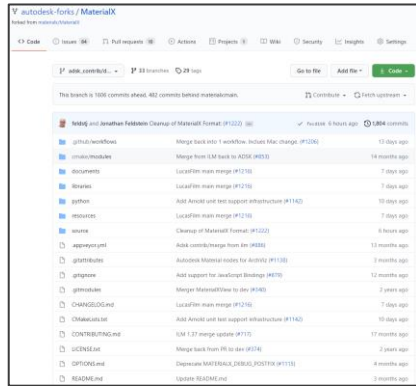
(See: [MaterialX home page](#) and [Open Source Repo](#))



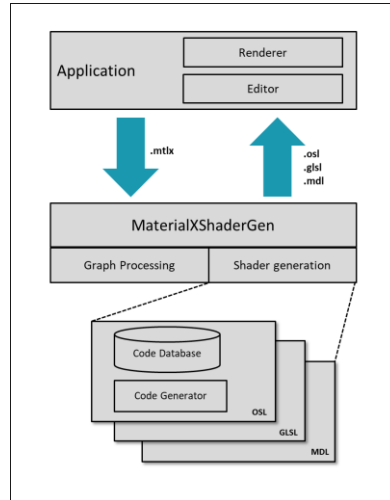
© & ™ Lucasfilm LTD. Used with permission.

Contributions

Collaborate on specification and implementation



Shader Generation



Physically Based Shading nodes

MaterialX Physically Based Shading Nodes

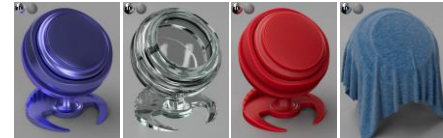
Version 1.38

Niklas Harrysson - Autodesk
Doug Smythe - Industrial Light & Magic
Jonathan Stone - Lucasfilm Advanced Development Group
February 2, 2021

Introduction

The MaterialX Specification describes a number of standard nodes that may be used to construct node graphs for the processing of images, procedurally-generated values, coordinates and other data. With the addition of user-defined custom nodes, it is possible to describe complete rendering shaders using node graphs. Up to this point, there has been no standardization of the specific shader-semantic nodes used in these node graph shaders, although with the widespread shift toward physically-based shading, it appears that the industry is settling upon a number of specific BSDF and other functions with standardized parameters and functionality.

This document describes a number of shader-semantic nodes implementing widely-used surface scattering, emission and volume distribution functions and utility nodes useful in constructing complex layered rendering shaders using node graphs. These nodes in combination with other nodes may be used with the MaterialX shader generation (ShaderGen) system.



<https://github.com/autodesk-forks/MaterialX>

Ecosystem



P I X A R



Open Shading Language



ASWF



MATERIALX

MATERIALX CORE



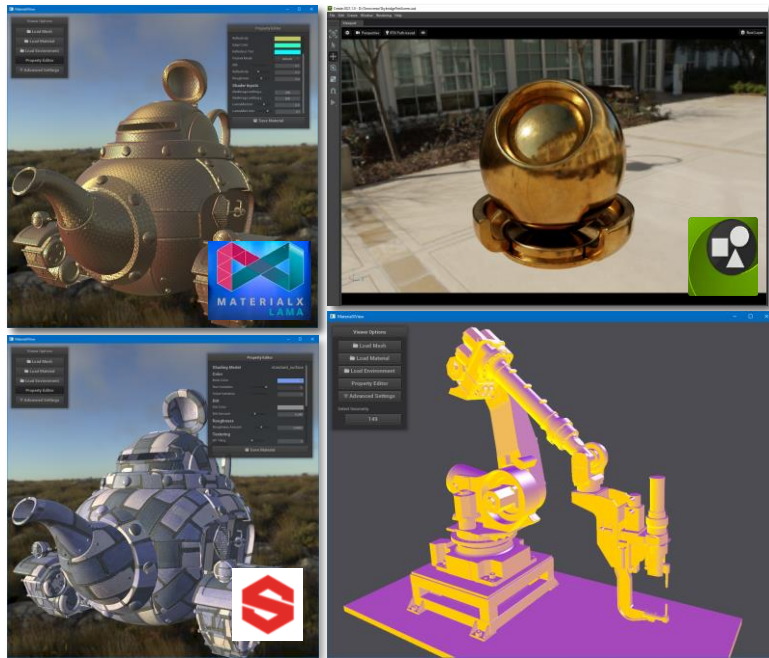
EMS : Open Standards Group
PSET Forge : Graphics, VizX

KHRONOS GROUP



Core Updates

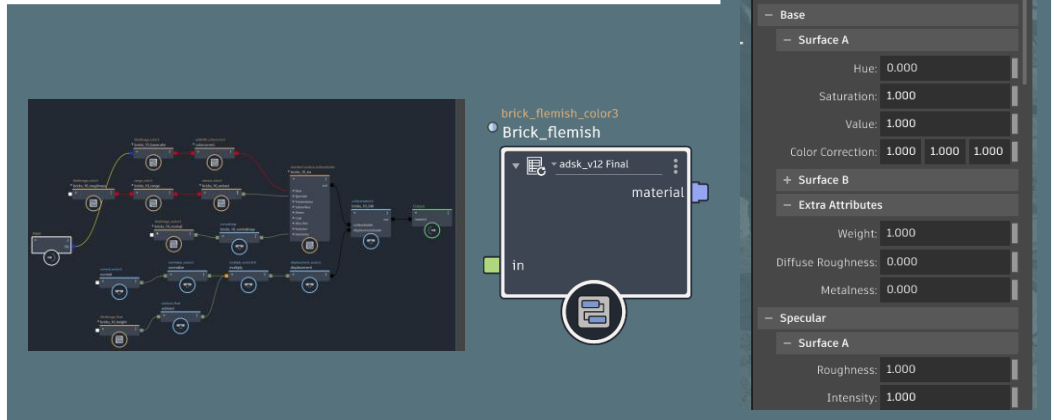
- Common [OSL](#) Implementations
- [OCIO](#) Color Management
- [MDL](#) updates for Omniverse.
- [SPIR-V](#) for OGS Modernization
- PBR node changes + [Pixar LAMA support](#)
- Procedurals
 - Unlit shader and modeling / NPR materials
 - Adobe Substance interchange



MaterialX Library Update

- “Standard material” representation
- ADSK definition library
- Prism materials (PrismX)
- Unified Real-World Unit and Color Management
- 1.38 Support
 - Arnold, 3ds Max, LookdevX, PrismX, Substance Designer

(See: [MaterialX Library page](#))

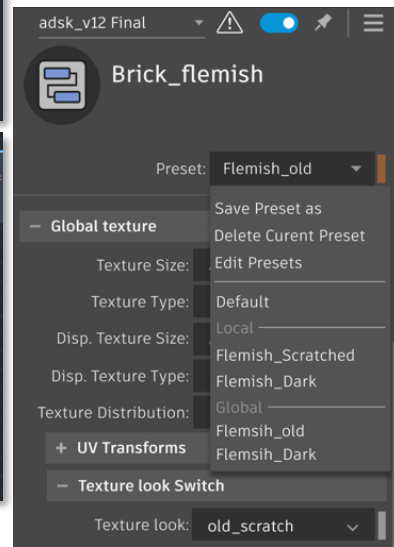
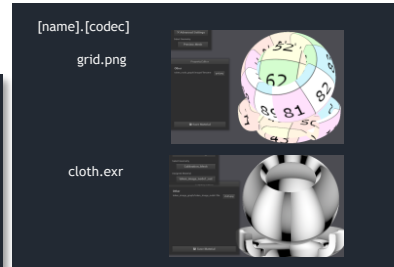
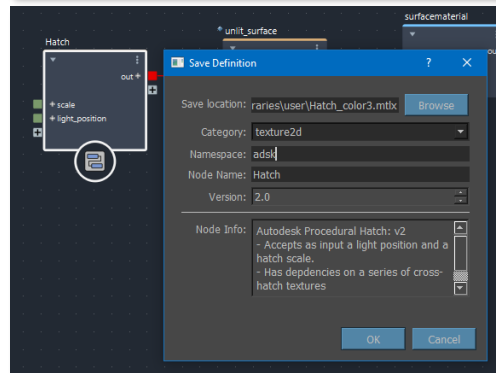
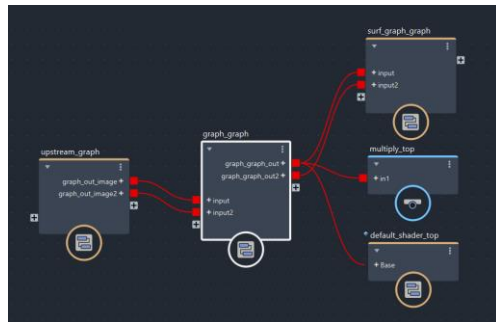


LookdevX Update

Assets and Runtime Execution

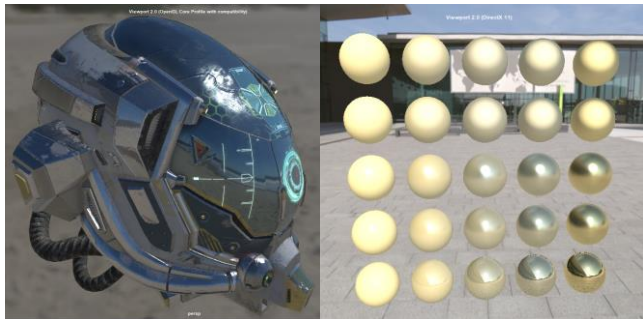
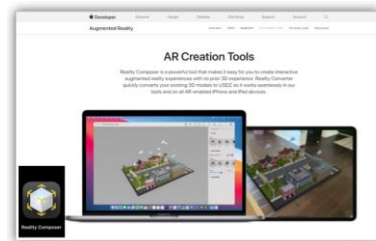
- Asset Requirements
 - Tokenization and Presets
 - Grouping flexibility
 - Definition publishing
 - Library management
- “Runtime” interaction
 - Export resolve for rendering
 - MaterialX Runtime module

(See: [TechX 2020 Presentation](#) for asset details.)



Integrations

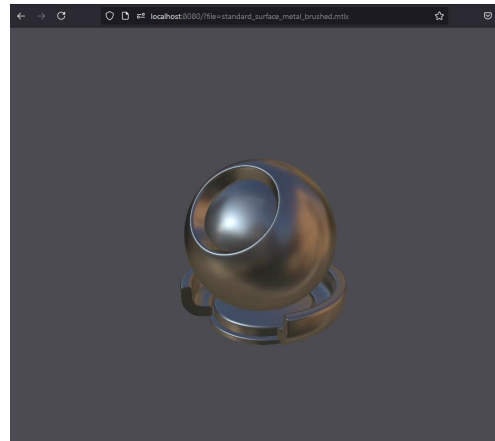
- Material Translation and Converter
 - ProteinX (Protein to Standard Surface as MaterialX)
- Fusion360
 - MaterialX in ATF
 - Baking + Translation
 - PrismX
- Maya USD/MaterialX
 - Light loop + prim-vars
 - Classic OGS fragments
 - Code gen performance



MaterialX Web Update

Why MaterialX Web?

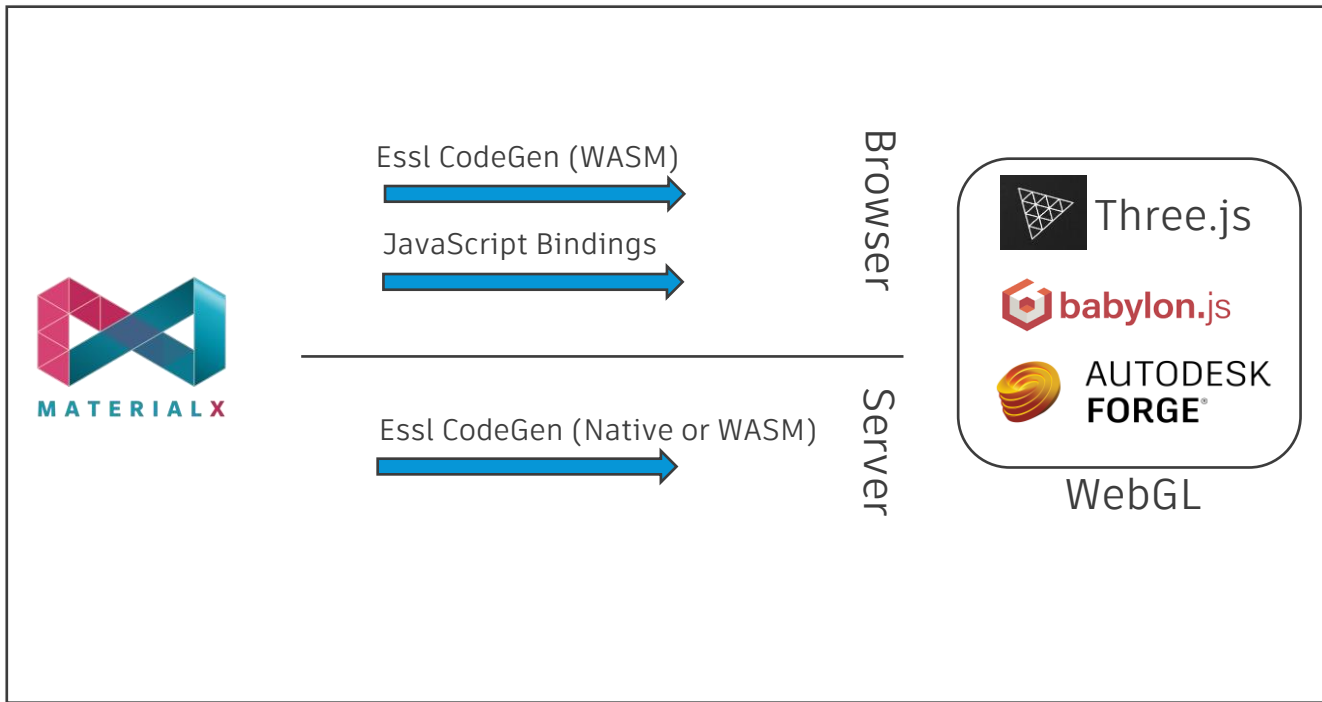
- Increase reach of MaterialX to Web based workflows
- Support Khronos (GLTF) discussions on MaterialX adoption
- MaterialX support in Forge Viewer to foster adoption in ADSK



MaterialX Web Update

- [Open-source project](#) for upcoming release
- Components:
 - Javascript Bindings + Web Assembly
 - Core, Format, GenShader
 - Essl CodeGen: WebGL, Open GL ES 3.0 (untested)
 - Web Viewer Sample Application
- Shader snippets fully compatible with current GLSL implementation
- Supported Browsers Chrome, Firefox, Edge, Safari*
- Framework agnostic

MaterialX Web Update



MaterialX Web Update

Demo

