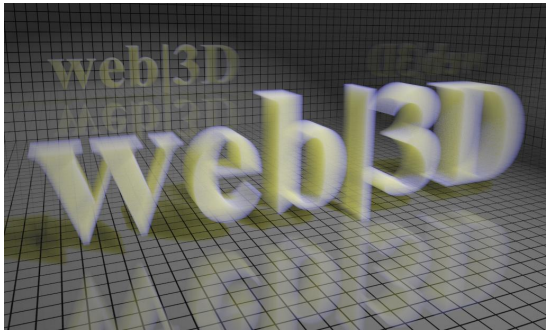
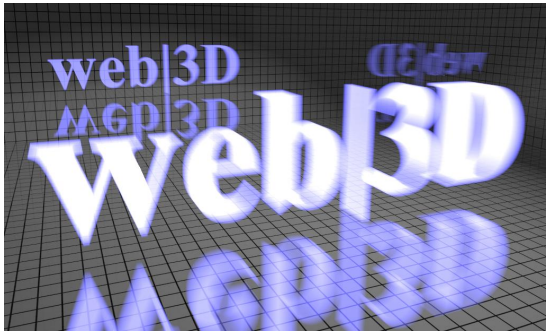
CommonVolumeShader: Simple and Portable Specification of Volumetric Light Transport in X3D

Karsten Schwenk, Johannes Behr, Dieter W. Fellner

Fraunhofer IGD, Germany

karsten.schwenk@fraunhofer.de





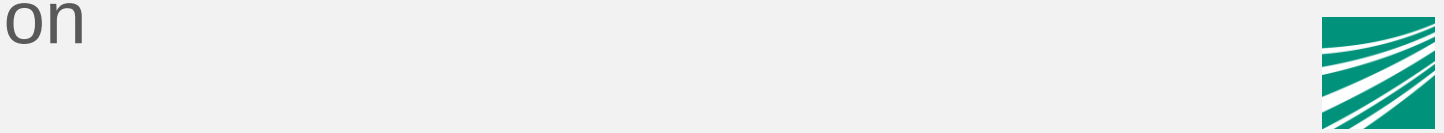
Fraunhofer IGD

instantreality

Web3D 2011

Introduction

Motivation



- Why model volumetric light transport?
  - Gives many materials their characteristic look

Images: Scattering. SIGGRAPH ASIA Course Notes, 2008.

Introduction

The Problem

- X3D has little support for local volumetric phenomena
- Fog
  - Not local
  - Not terribly accurate
- Volume Rendering Component
  - No arbitrary shapes
  - Intended for stylized rendering
- Shaders
  - Low portability (even worse than for surfaces)

Introduction
Our Suggestion

- Want a VolumeShader with these properties
  - Captures most common effects well
  - Attach to polygonal geometry (Shape/Appearance)
  - Complementary to (and compatible with) CommonSurfaceShader
  - Portable
  - Compact specification
  - Easy to implement
  - Physically-based

Our Proposal

Overview

- CommonVolumeShader
  - Inspired by CommonSurfaceShader
  - Satisfies aforementioned requirements
    - (well, some more than others...)
    - Maybe it’s not that ‘easy to implement’

Our Proposal

VolumeShader Node

```
VolumeShader : X3DShaderNode {
  [...]
}
```

- Group all declarative volume shader nodes
- Why derive from X3DShaderNode?
  - Can be assigned to Appearance
  - Can combine with CommonSurfaceShader
  - Selection mechanism

Our Proposal

CommonVolumeShader Node

- VolumeShader impl. for common materials
- Composed of three components:
  - Emission
  - Absorption
  - Scattering
- Enough to capture most materials

Our Proposal

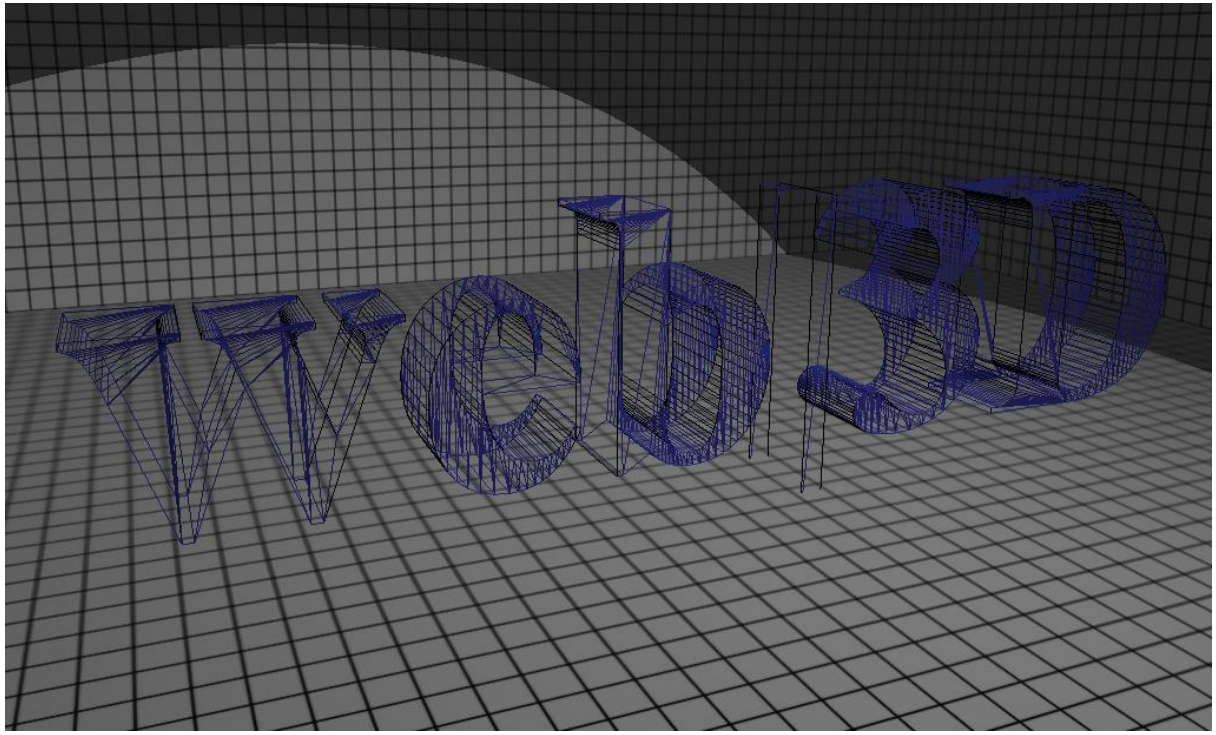
CommonVolumeShader Fields

```
CommonVolumeShader : VolumeShader {
  [...]
  SFVec3f   [in,out] emissionFactor        0.0 0.0 0.0
  SFNode    [in,out] emissionTexture             NULL

  SFVec3f   [in,out] absorptionFactor      0.0 0.0 0.0
  SFNode    [in,out] absorptionTexture           NULL

  SFVec3f   [in,out] scatteringFactor      0.0 0.0 0.0
  SFNode    [in,out] scatteringTexture           NULL
  SFNode    [in,out] phaseFunction               NULL
}
```

- See paper on how these map to physical properties
- PhaseFunction node in paper, too

Our Proposal

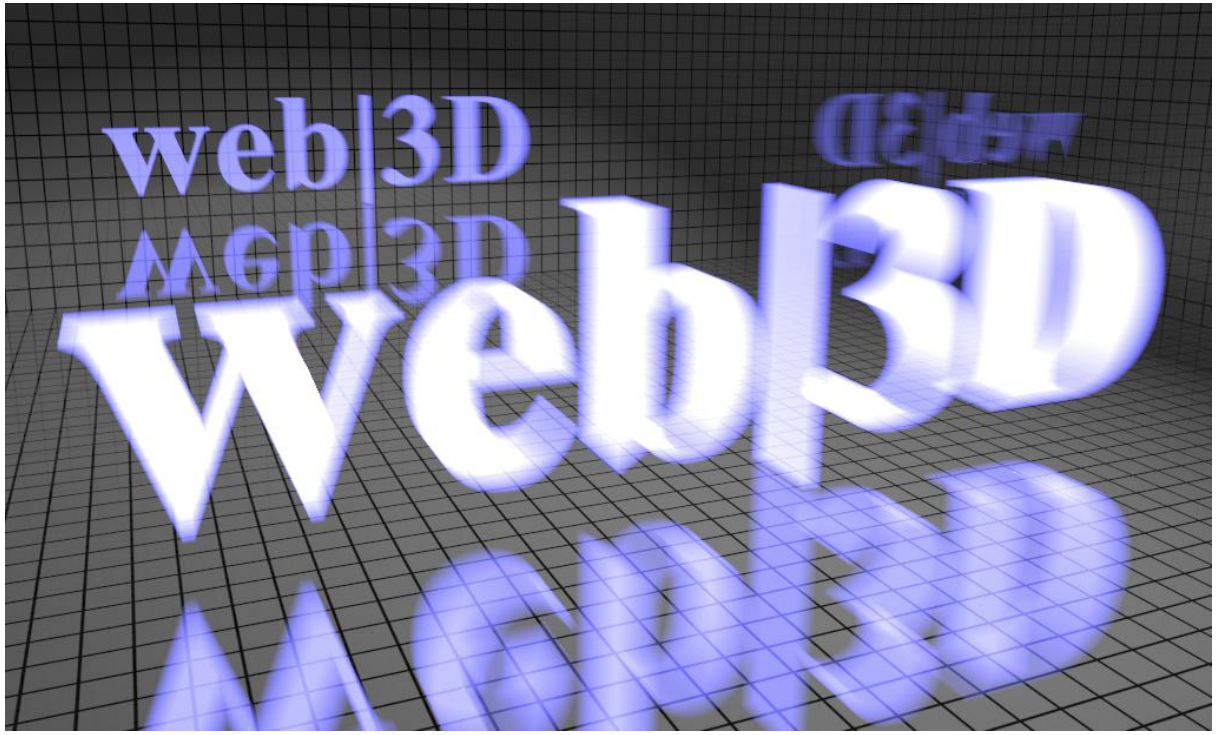
CommonVolumeShader Geometry




```
<Shape>
  <Appearance>
    [...]
  </Appearance>
  <Text string='web|3D' depth='.25' solid='true' resolution='2.'>
    <FontStyle style='BOLD' justify='MIDDLE' family='SERIF'/>
  </Text>
</Shape>
```

Our Proposal

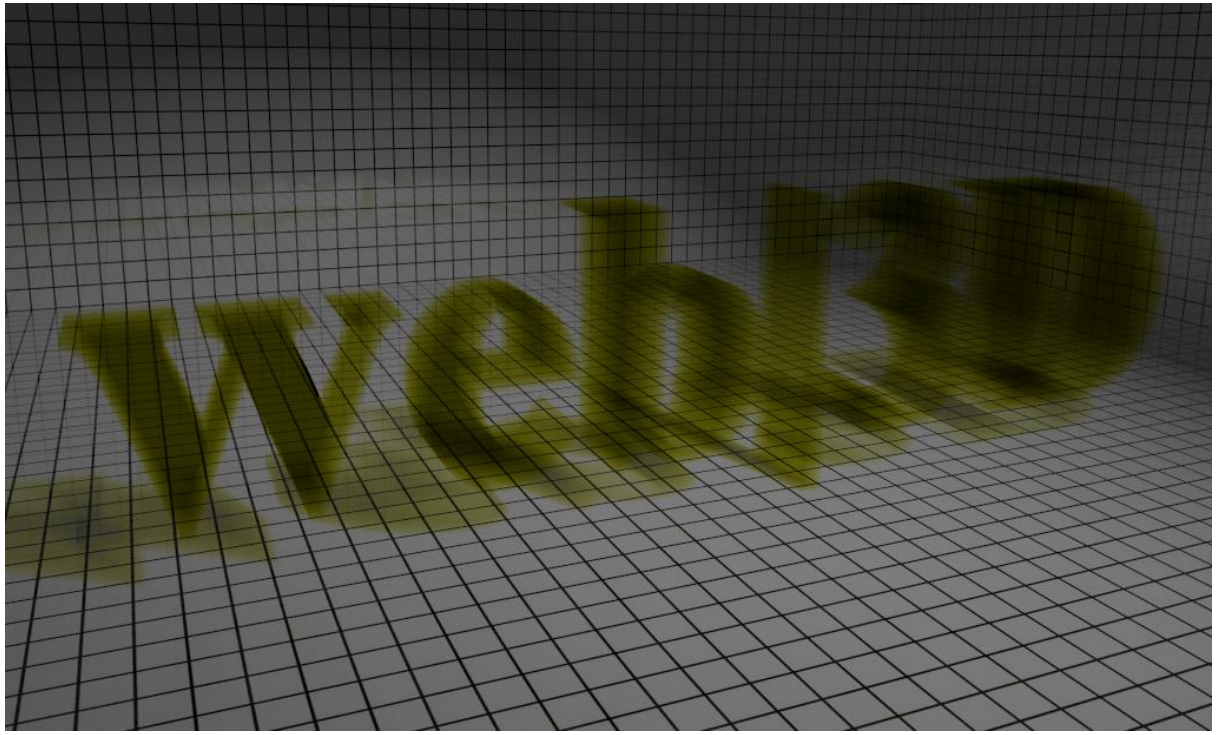
CommonVolumeShader Emission




```
<Appearance>
  <CommonSurfaceShader diffuseFactor='0 0 0' transmissionFactor='1 1 1' />
  <CommonVolumeShader emissionFactor='4 4 24' />
</Appearance>
```

Our Proposal

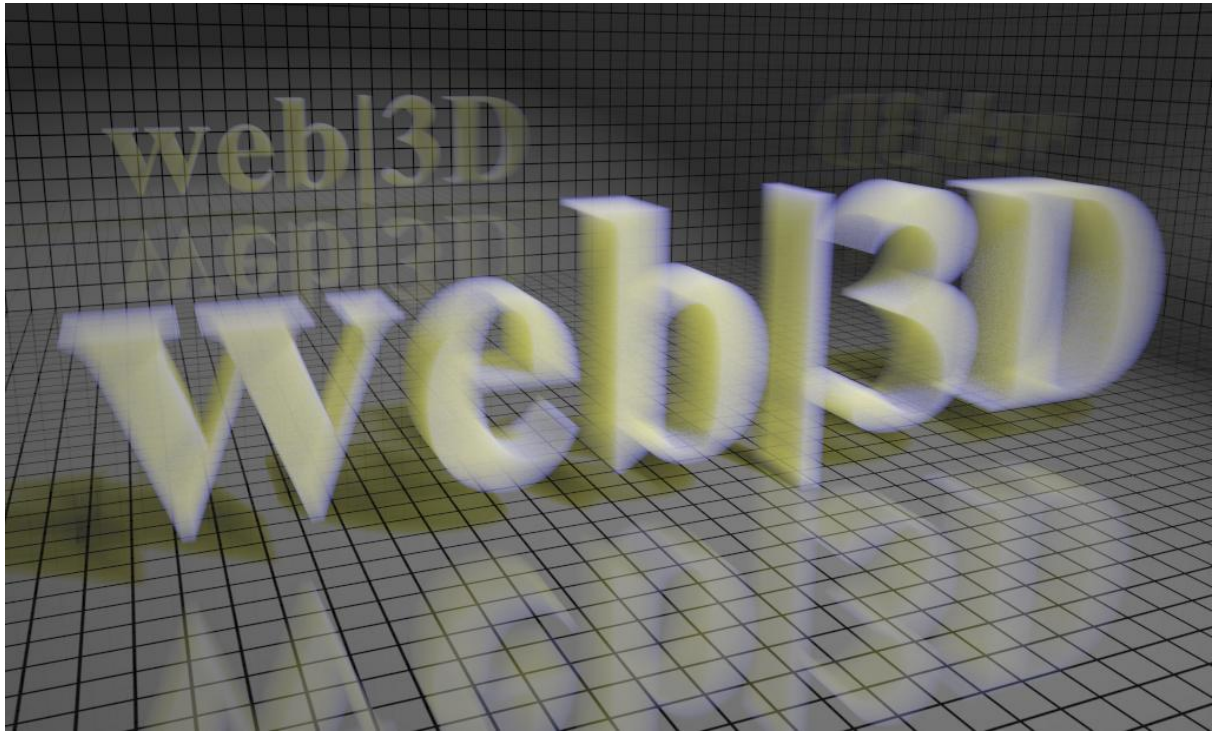
CommonVolumeShader Absorption




```
<Appearance>
  <CommonSurfaceShader diffuseFactor='0 0 0' transmissionFactor='1 1 1' />
  <CommonVolumeShader absorptionFactor='4 4 24' />
</Appearance>
```

Our Proposal

CommonVolumeShader Scattering




```
<Appearance>
  <CommonSurfaceShader diffuseFactor='0 0 0' transmissionFactor='1 1 1' />
  <CommonVolumeShader scatteringFactor='4 4 24' />
</Appearance>
```

Our Proposal

VolumeShaderFog Node

- Binds to Fog bindable stack
- Single Scene-wide VolumeShader
- Primarily for more accurate (lit) fog

Our Proposal

Implementation

- Ray tracer and rasterization pipeline
- Both use ray marching
  - Can be expensive
- Both use only single scattering
  - Wrong for high-albedo media
- Both have problems with nested volumes
- Rasterization has additional problems with
  - Alpha-blended stuff
  - Non-convex volumes

Our Proposal

Limitations of the Node

- Only isotropic emission / absorption
- No fluorescence / phosphorescence
- Only physically plausible media (no effect shaders)

Conclusion

- CommonVolumeShader
  - Compact & portable representation
  - Captures many common materials accurately
  - Basic support not that hard
  - Full implementation tricky

Future Work

- Address limitations
  - Especially of rasterization implementation
- Implementation for path tracer
  - Multiple scattering
- Flesh out behavior for problematic cases
  - Two shaders selected
  - Nested volumes

CommonVolumeShader: Simple and Portable Specification of Volumetric Light Transport in X3D

Karsten Schwenk, Johannes Behr, Dieter W. Fellner
Fraunhofer IGD, Germany
karsten.schwenk@fraunhofer.de

- Acknowledgements
  - Buddha, Dragon: Stanford 3D Scanning Rep.
  - Sponza: Marko Dabrovic / Crytek
  - Church Window: Wikimedia / Chartres Cathedral

Fraunhofer IGD

instantreality

Web 3D 2011