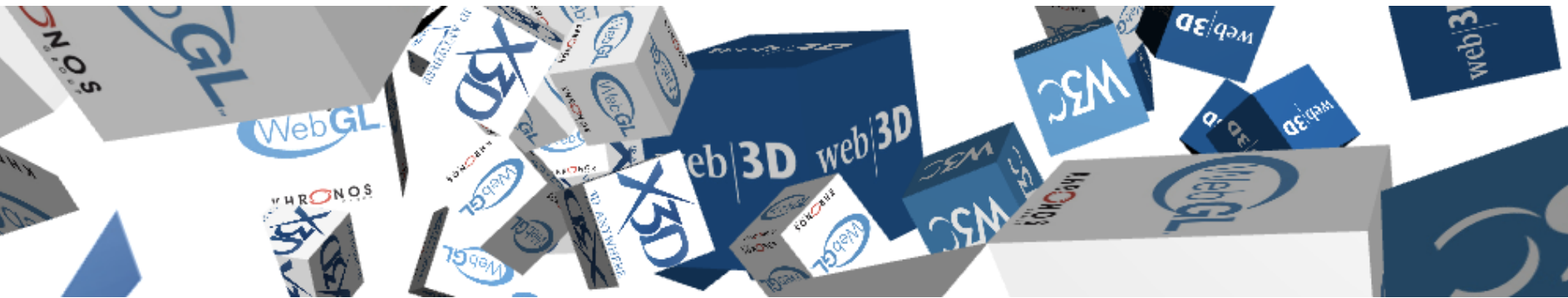


X3DOM

Fast content delivery for declarative 3D



WebGL BOF, Siggraph 2011

Johannes Behr

W3C: Chair of „Declarative 3D“
Community Group

johannes.behr@igd.fraunhofer.de



W3C “Declarative 3D” CG

Fraunhofer IGD, DFKI and Web3D Consortium



2D
(HTML5 spec)



Declarative

Scenegraph

Part of HTML-document

DOM Integration

CSS/ Events

3D
(No W3C spec yet)



Runs
on

Imperative

Procedural API

Drawing context

Flexible

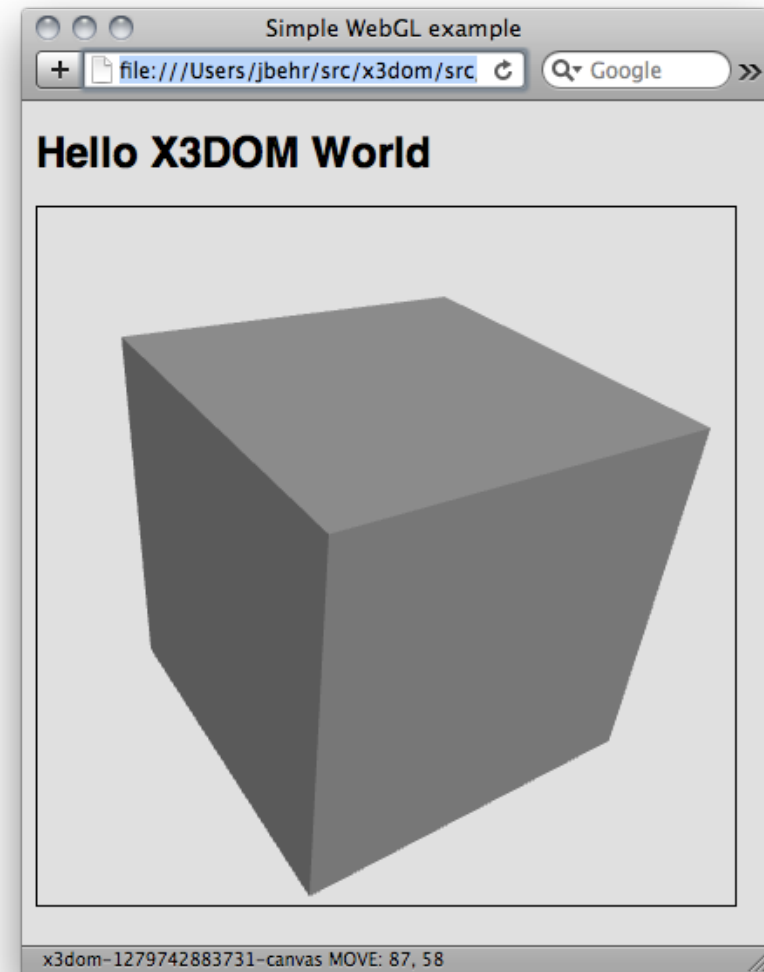


Declarative (X)3D in HTML

Embed a live scenegraph in the DOM



```
<!DOCTYPE html >
<html >
  <body>
    <h1>Hello X3DOM World</h1>
    <x3d xmlns='...' profile='HTML' >
      <scene>
        <shape>
          <box></box>
        </shape>
      </scene>
    </x3d>
  </body>
</html>
```



Efficient processing of large content sets



Browser are not build to hold GByte of DOM attribute data (e.g. multiple data copies)

Reference external sub-trees

X3D "Inline" node

black/white-box interface?

Binary XML decomposition

x3z: (ISO) Decoding on JS-Level

x3db: (ISO) Fast Infoset: No UA or JS-lib

EXI: (W3C) Even worse

```

# Import the keras module
import keras

# Import the Sequential model class from the keras.models module
from keras.models import Sequential

# Import the Dense layer class from the keras.layers module
from keras.layers import Dense

# Create a Sequential model
model = Sequential()

# Add a Dense layer with 10 units
model.add(Dense(10))

# Compile the model with the Adam optimizer, categorical_crossentropy loss function, and accuracy metric
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Load the training data (X_train, y_train) and validation data (X_val, y_val)
X_train = ...
y_train = ...
X_val = ...
y_val = ...

# Train the model for 10 epochs
model.fit(X_train, y_train, validation_data=(X_val, y_val), epochs=10)

# Evaluate the model on the validation data
loss, accuracy = model.evaluate(X_val, y_val)

# Print the loss and accuracy
print('Validation loss: ', loss)
print('Validation accuracy: ', accuracy)

```

DOM holds structure and data



```
<!DOCTYPE html>
<html>
  <head>
    <link rel='stylesheet' type='text/css' href='http://www.x3dom.org/x3dom/release/x3dom.css'></link>
    <script type='text/javascript' src='http://www.x3dom.org/x3dom/release/x3dom.js'></script>
  </head>
  <body>
    <x3d id='3dstuff' width='400px' height='400px'>
      <scene DEF='scene'>
        <shape>
          <appearance>
            <material diffuseColor='#FF0000'></material>
          </appearance>
          <indexedTriangleSet solid='false' index='0 1 2 1 3 2 1 4 3 5 4 1 0 5 1 0 6 5 6 7 5 5 7 4 7 8 4 7 9 8 7 6 9 6 10 9 10 11
9 10 2 11 10 0 2 6 0 10 11 2 3 8 11 3 4 8 3 11 8 9'>
            <coordinate point='0.447214 0 -0.894427 0.447214 0.850651 -0.276393 1 0 -0 0.447214 0.525731 0.723607 -0.447214
0.850651 0.276393 -0.447214 0.525731 -0.723607 -0.447214 -0.525731 -0.723607 -1 0 0 -0.447214 0 0.894427 -0.447214 -0.850651
0.276393 0.447214 -0.850651 -0.276393 0.447214 -0.525731 0.723607'></coordinate>
            <normal vector='0.447214 0 -0.894427 0.447214 0.850651 -0.276393 1 0 -0 0.447214 0.525731 0.723607 -0.447214 0.850651
0.276393 -0.447214 0.525731 -0.723607 -0.447214 -0.525731 -0.723607 -1 0 0 -0.447214 0 0.894427 -0.447214 -0.850651 0.276393
0.447214 -0.850651 -0.276393 0.447214 -0.525731 0.723607'></normal>
          </indexedTriangleSet>
        </shape>
      </scene>
    </x3d>
  </body>
</html>
```

DOM holds structure and data

More than 95% are usually unstructured data



```
<!DOCTYPE html>
<html>
  <head>
    <link rel='stylesheet' type='text/css' href='http://www.x3dom.org/x3dom/release/x3dom.css'></link>
    <script type='text/javascript' src='http://www.x3dom.org/x3dom/release/x3dom.js'></script>
  </head>
  <body>
    <x3d id='3dstuff' width='400px' height='400px'>
      <scene DEF='scene'>
        <shape>
          <appearance>
            <material diffuseColor='#FF0000'></material>
          </appearance>
          <indexedTriangleSet solid='false' index='0 1 2 1 3 2 1 4 3 5 4 1 0 5 1 0 6 5 6 7 5 5 7 4 7 8 4 7 9 8 7 6 9 6 10 9 10 11
9 10 2 11 10 0 2 6 0 10 11 2 3 8 11 3 4 8 3 11 8 9'>
            <coordinate point='0.447214 0 -0.894427 0.447214 0.850651 -0.276393 1 0 -0 0.447214 0.525731 0.723607 -0.447214
0.850651 0.276393 -0.447214 0.525731 -0.723607 -0.447214 -0.525731 -0.723607 -1 0 0 -0.447214 0 0.894427 -0.447214 -0.850651
0.276393 0.447214 -0.850651 -0.276393 0.447214 -0.525731 0.723607'></coordinate>
            <normal vector='0.447214 0 -0.894427 0.447214 0.850651 -0.276393 1 0 -0 0.447214 0.525731 0.723607 -0.447214 0.850651
0.276393 -0.447214 0.525731 -0.723607 -0.447214 -0.525731 -0.723607 -1 0 0 -0.447214 0 0.894427 -0.447214 -0.850651 0.276393
0.447214 -0.850651 -0.276393 0.447214 -0.525731 0.723607'></normal>
          </indexedTriangleSet>
        </shape>
      </scene>
    </x3d>
  </body>
</html>
```

Separate structure and data

HTML element reference external binary data element



DOM / HTML Document

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
    charset=utf-8" />
    <title>Single Mesh Model with a flat hierarchy (small model)
  </title>
    <link rel="stylesheet" type="text/css" href="x3dom.css" />
  </head>
  <body>
    <h1>Single Mesh Model with a flat hierarchy (small model)</h1>
    <p class="case">
      <!-- Use the same "x", "y", "width" and "height" attributes as SVG.
      Optionally support the "viewbox" and "preserveAspectRatio" just
      like SVG -->
      <X3D xmlns="http://www.web3d.org/specifications/x3d-namespace"
      id="spheres" showStat="true" x="0px" y="0px" width="400px" height="400px">
        <Scene DEF="scene">
          <Viewpoint position="0 0 25" orientation="0 0 1" />
          <Background skyColor="1 1 1"/>
          <Shape>
            <Appearance DEF="App">
              <Material ambientIntensity="0.0243902" diffuseColor="0.41 0.39 0.03"
              shininess="0.12" specularColor="0.94 0.72 0"/>
            </Appearance>
            <IndexedFaceSet DEF="_0" solid="false" creaseAngle="3.14"
            0.0489382 />
          </IndexedFaceSet>
        </Shape>
      </Scene>
    </X3D>
    <p>
      <p class="case"> &nbap; </p>
      <script type="text/javascript" src="x3dom.js"></script>
    </body>
  </html>
```

Binary asset resources

Generic asset dictionary

- Directly loaded to TypedArrays
- Data assignment in JS
- Multiple arrays per file
- Multiple files per scene

Images and Videos

- Encodes int/float arrays (e.g. coordinate, normal, texCoords, generic-attributes) in RGBA-images
- Multiple images per array
- Multiple images per scene

/<video> as generic binary container

Powerful abstraction for efficient data encoding for Web-apps



Decompression for free (only lossless png is useful right now)

Streaming updates for free: WebGL/X3DOM support <video>

Quantization and encoding supports LOD & streaming of precision

Flexible and efficient visualization

Single VBO: Extremely fast visualization with Vertex Textures Units,
precision grows until vertex texture limit is reached

Multiple VBO: WebGL without Vertex Texture Unit / Flash 11

Encoding is simple, fast and works with any mesh type

Browser/Server well optimized to handle large number of images
and parallel downloads of image => Great user experience

User, Developer, Browser, Server and W3C love images and video:
Content is HTML + image/video-resource data

Demo

http://x3dom.org/x3dom/example/x3dom_imageGeometry.html

