X3D C++/C# Language Binding

SC24 WG9 and Web3D Meetings

2016.1.11 - 14

Myeong Won Lee
Java language binding

• X3D scene access using Java language
• 19777-2 scope
  – provides a browser implementation independent way of accessing the browser capabilities through the Java language.
  – provides a set of implementation independent base classes and interfaces that represent the possible interactions with the X3D scene through the SAI
  – provides Java API format for X3D scene access
Concepts of C++/C# language binding

• What is C++ / C# language binding?
  – The same as Java or not?

• If the same
  – X3D scene access using C++ and C# languages
  – scope
    • provides a browser implementation independent way of accessing the browser capabilities through the Java language.
    • provides a set of implementation independent base classes and interfaces that represent the possible interactions with the X3D scene through the SAI
    • provides Java API format for X3D scene access

• If not the same
  – Additionally, X3D encoding in C++/C# programs is necessary.
Scope of C++/C# language binding

- X3D scene access API
- X3D encoding in C++/C# programs

what is the scope?
A sample of X3D scene access API

```xml
<X3D>
  <Scene>
    <Background skyColor='1 1 1'/>
    <Viewpoint description='Book View'
      orientation='-0.747 -0.624 -0.231 1.05' position='-
      1.81 3.12 2.59'/>
    <Shape>
      <Box size='1 2 3'/>
      <Appearance>
        <Material/>
      </Appearance>
    </Shape>
  </Scene>
</X3D>
```

getX3D
getScene
getBackground
getViewpoint
getShape
getBox
getApperance
getMaterial

setX3D
setScene
setBackground
setViewpoint
setShape
setBox
setApperance
setMaterial
A sample of X3D scene access API (C++)

**getX3D** (&pX3D)
**getScene**(&pScene)
**getBackground**(&pBackground)
**getViewpoint**(&pViewpoint)
**getShape**(&pShape)
**getBox**(&pBox)
**getApperance**(&pAppearance)
**getMaterial**(&pMaterial)

**setX3D** (pX3D)
**setScene**(pScene)
**setBackground**(pBackground)
**setViewpoint**(pViewpoint)
**setShape**(pShape)
**setBox**(pBox)
**setApperance**(pAppearance)
**setMaterial**(pMaterial)
## A sample of X3D scene access API (C#)

<table>
<thead>
<tr>
<th>Getter Method</th>
<th>Setter Method</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>getX3D</code> (pX3D)</td>
<td><code>setX3D</code> (pX3D)</td>
</tr>
<tr>
<td><code>getScene</code> (pScene)</td>
<td><code>setScene</code> (pScene)</td>
</tr>
<tr>
<td><code>getBackground</code></td>
<td><code>setBackground</code></td>
</tr>
<tr>
<td><code>getViewpoint</code></td>
<td><code>setViewpoint</code></td>
</tr>
<tr>
<td><code>getShape</code></td>
<td><code>setShape</code></td>
</tr>
<tr>
<td><code>getBox</code></td>
<td><code>setBox</code></td>
</tr>
<tr>
<td><code>getApperance</code></td>
<td><code>setApperance</code></td>
</tr>
<tr>
<td><code>getMaterial</code></td>
<td><code>setMaterial</code></td>
</tr>
</tbody>
</table>

### Example Usage

```csharp
X3DScene x3dScene = new X3DScene();

// Getters
x3dScene.GetX3D();
x3dScene.GetScene();
x3dScene.GetBackground();
x3dScene.getViewpoint();
x3dScene.GetShape();
x3dScene.GetBox();
x3dScene.GetApperance();
x3dScene.GetMaterial();

// Setters
x3dScene.SetX3D(x3d); // x3d is of type X3D
x3dScene.setScene(scene); // scene is of type X3DScene
x3dScene.setBackground(background); // background is of type X3DBackground
x3dScene.setViewpoint(viewpoint); // viewpoint is of type X3DViewpoint
x3dScene.setShape(shape); // shape is of type X3DShape
x3dScene.setBox(box); // box is of type X3DBox
x3dScene.setApperance(apperance); // apperance is of type X3DApperance
x3dScene.setMaterial(material); // material is of type X3DMaterial
```