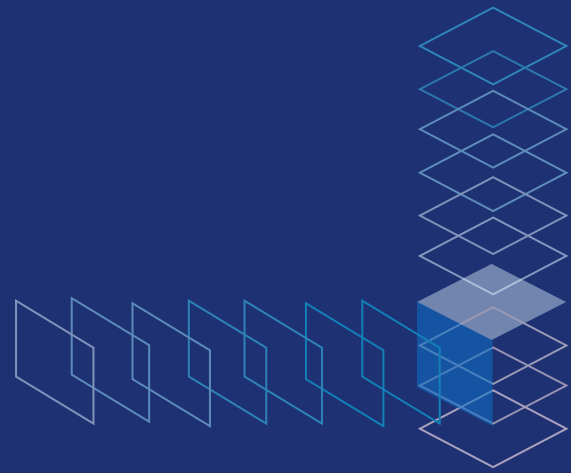
WEB3D 2020

The 25th International ACM Conference on 3D Web Technology
November 9-13, 2020, Virtual Conference, Seoul, Korea

X3D4 Sound model and Validation Examples

Eftychia Lakka, University of South Wales, Pontypridd Wales UK



ACMSIGGRAPH web3D CONSORTIUM Korea Computer Graphics Society

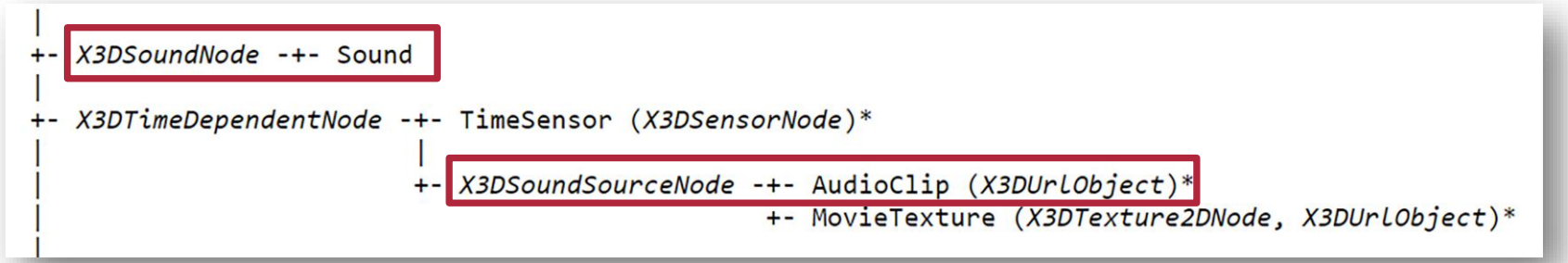
X3D4 Sound model and Validation Examples

Eftychia Lakka, University of South Wales, Pontypridd Wales UK

Overview

- X3D Sound Nodes
- Extend X3D with Spatial Sound Nodes (X3D4)
- New Nodes
  - Design and Naming
  - Abstract Nodes
  - Set of new Nodes(Sources/ Effects – Filters/ Visualisation/ Split & Merge/ Destinations)
- X3D Step by Step - Examples
- Conclusion – Next Steps

X3D**v3** Sound Nodes



X3DSoundNode

```
X3DSoundNode : X3DChildNode {
  SFNode [in,out] metadata NULL [X3DMetadataObject]
}
```

Sound

```
Sound : X3DSoundNode {
  SFVec3f [in,out] direction  0 0 1 (-∞,∞)
  SFFloat [in,out] intensity  1     [0,1]
  SFVec3f [in,out] location   0 0 0 (-∞,∞)
  SFFloat [in,out] maxBack    10    [0,∞)
  SFFloat [in,out] maxFront   10    [0,∞)
  SFNode  [in,out] metadata   NULL  [X3DMetadataObject]
  SFFloat [in,out] minBack    1     [0,∞)
  SFFloat [in,out] minFront   1     [0,∞)
  SFFloat [in,out] priority   0     [0,1]
  SFNode  [in,out] source     NULL  [X3DSoundSourceNode]
  SFBool  []       spatialize TRUE
}
```

X3DSoundSourceNode

```
X3DSoundSourceNode : X3DTimeDependentNode {
  SFString [in,out] description      ""
  SFBool   [in,out] loop             FALSE
  SFNode   [in,out] metadata         NULL [X3DMetadataObject]
  SFTime   [in,out] pauseTime        0    (-∞,∞)
  SFFloat  [in,out] pitch            1.0  (0,∞)
  SFTime   [in,out] resumeTime       0    (-∞,∞)
  SFTime   [in,out] startTime        0    (-∞,∞)
  SFTime   [in,out] stopTime         0    (-∞,∞)
  SFTime   [out]    duration_changed
  SFTime   [out]    elapsedTime
  SFBool   [out]    isActive
  SFBool   [out]    isPaused
}
```

AudioClip

```
AudioClip : X3DSoundSourceNode, X3DUrlObject {
  SFString [in,out] description      ""
  SFBool   [in,out] loop             FALSE
  SFNode   [in,out] metadata         NULL  [X3DMetadataObject]
  SFTime   [in,out] pauseTime        0     (-∞,∞)
  SFFloat  [in,out] pitch            1.0   (0,∞)
  SFTime   [in,out] resumeTime       0     (-∞,∞)
  SFTime   [in,out] startTime        0     (-∞,∞)
  SFTime   [in,out] stopTime         0     (-∞,∞)
  MFString [in,out] url              []    [URI]
  SFTime   [out]    duration_changed
  SFTime   [out]    elapsedTime
  SFBool   [out]    isActive
  SFBool   [out]    isPaused
}
```

Extend X3D with Spatial Sound Nodes (X3D**v4**) (1)

Extend X3D with Spatial Sound Nodes (X3D**v4**) (2)

Category	New X3D4 Node	Web Audio API
Sources	BufferAudioSource	AudioBuffer, AudioBufferSourceNode
	OscillatorSource	OscillatorNode
	StreamAudioSource	MediaStreamAudioSourceNode
	MicrophoneSource	-
	ListenerPointSource	AudioListener
Effects	SpatialSound	PannerNode
	BiquadFilter	BiquadFilterNode
	Convolver	ConvolverNode
	Delay	DelayNode
	DynamicsCompressor	DynamicsCompressorNode
	Gain	GainNode, gain field
	WaveShaper	WaveShaperNode
	PeriodicWave	PeriodicWave
	Analyser	AnalyserNode
	ChannelSplitter, ChannelSelector	ChannelSplitterNode
	ChannelMerger	ChannelMergerNode
	AcousticProperties	-
	Doppler	-
Destination	AudioDestination	AudioDestinationNode
	StreamAudioDestination	MediaStreamAudioDestinationNode

New Nodes - Design and Naming

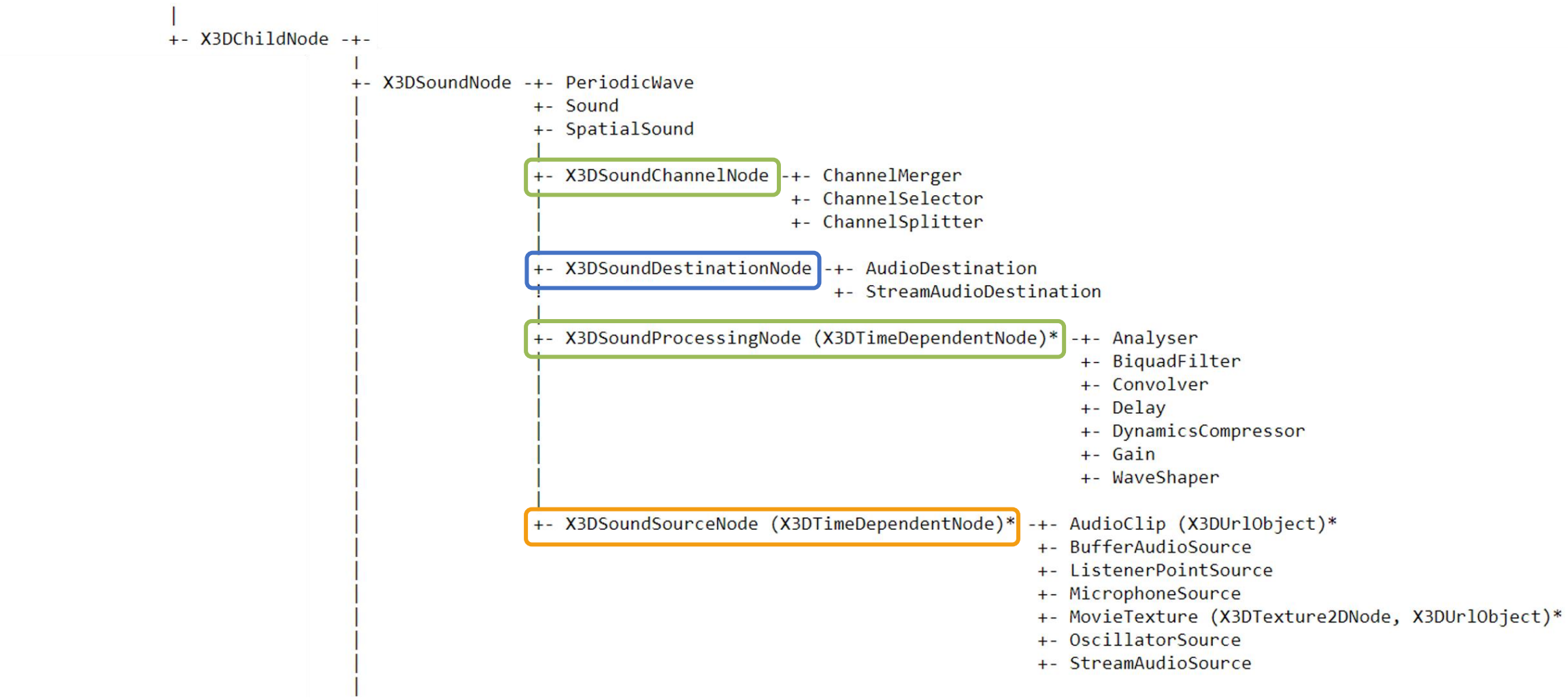
- X3D nodes are defined in an object-oriented fashion
- X3D naming conventions follow a specific design pattern
- names for concrete instantiable nodes are kept simple and functional
- abstract node types
  - begin with “X3D” and
  - end with “Node” (e.g., X3DTexture-Node)

- since many W3C Audio API interfaces end with “Node” as well, distinct names are needed for clarity even when semantics are identical

New Nodes - Abstract Nodes (1)

- **X3DSoundSourceNode**: derive node types that can emit audio data
- **X3DSoundProcessingNode**: for all sound processing nodes
- **X3DSoundChannelNode**: for nodes that handle of channels in an audio stream, allowing them to be split or merged
- **X3DSoundDestinationNode**: for all sound destination nodes, which represent the final destination of an audio signal

New Nodes - Abstract Nodes (2)



New Nodes - Set of new Nodes - Sources

- BufferAudioSource a memory-resident audio asset that can contain one or more channels
- OscillatorSource a virtual audio source generating a periodic waveform, providing a constant tone
- StreamAudioSource operates as an audio source whose media is received from a MediaStream obtained using the WebRTC
- MicrophoneSource captures input from a physical microphone
- ListenerPointSource represents the position and orientation of a person listening to virtual sound in the audio scene and provides single or multiple sound channels as output. Multiple ListenerPointSource nodes can be active for sound processing

New Nodes - Set of new Nodes – Effects/Filters (1)

- SpatialSound positions, emits and spatializes an audio stream in three-dimensional (3D) space
- BiquadFilter represents different kinds of filters, tone control devices, and graphic equalizers
- Convolver performs a linear convolution on a given AudioBuffer, often used to achieve a reverberation effect
- Delay causes a time delay between the arrival of input data and subsequent propagation to the output

New Nodes - Set of new Nodes – Effects/Filters (2)

- DynamicsCompressor implements a dynamics compression effect
- Gain amplifies or de-amplifies the input signal
- WaveShaper represents a nonlinear distorter that applies a wave-shaping distortion curve to the signal
- PeriodicWave defines a periodic waveform that can be used to shape the output of an Oscillator

New Nodes - Set of new Nodes – Visualisation

- Analyser node provides real-time frequency and time-domain analysis information, without any change to the input other than gain amplification

New Nodes - Set of new Nodes – Split & Merge

- ChannelSplitter separates the different channels of a single audio source into a set of monophonic output channels
- ChannelSelector selects a single channel output from all input channels
- ChannelMerger unites different input channels into a single output channel

New Nodes - Set of new Nodes – Acoustic Properties

- AcousticProperties describes coefficients related to Acoustic effects including surface reflection (specular, diffuse), wave phenomena (refraction, diffraction) and absorption

```
AcousticProperties : X3DAppearanceChildNode  {
  SFFloat    [in,out]     absorption   0      [0,1]
  SFString   [in,out]     description  ""
  SFFloat    [in,out]     diffraction  0      [0,1]
  SFFloat    [in,out]     diffuse      0      [0,1]
  SFBool     [in,out]     enabled      TRUE
  SFNode     [in,out]     metadata     NULL   [X3DMetadataObject]
  SFFloat    [in,out]     refraction   0      [0,1]
  SFFloat    [in,out]     specular     0      [0,1]}
```

New Nodes - Set of new Nodes – Destinations

- AudioDestination represents the final audio destination and is what user ultimately hears, typically from the speakers of user device

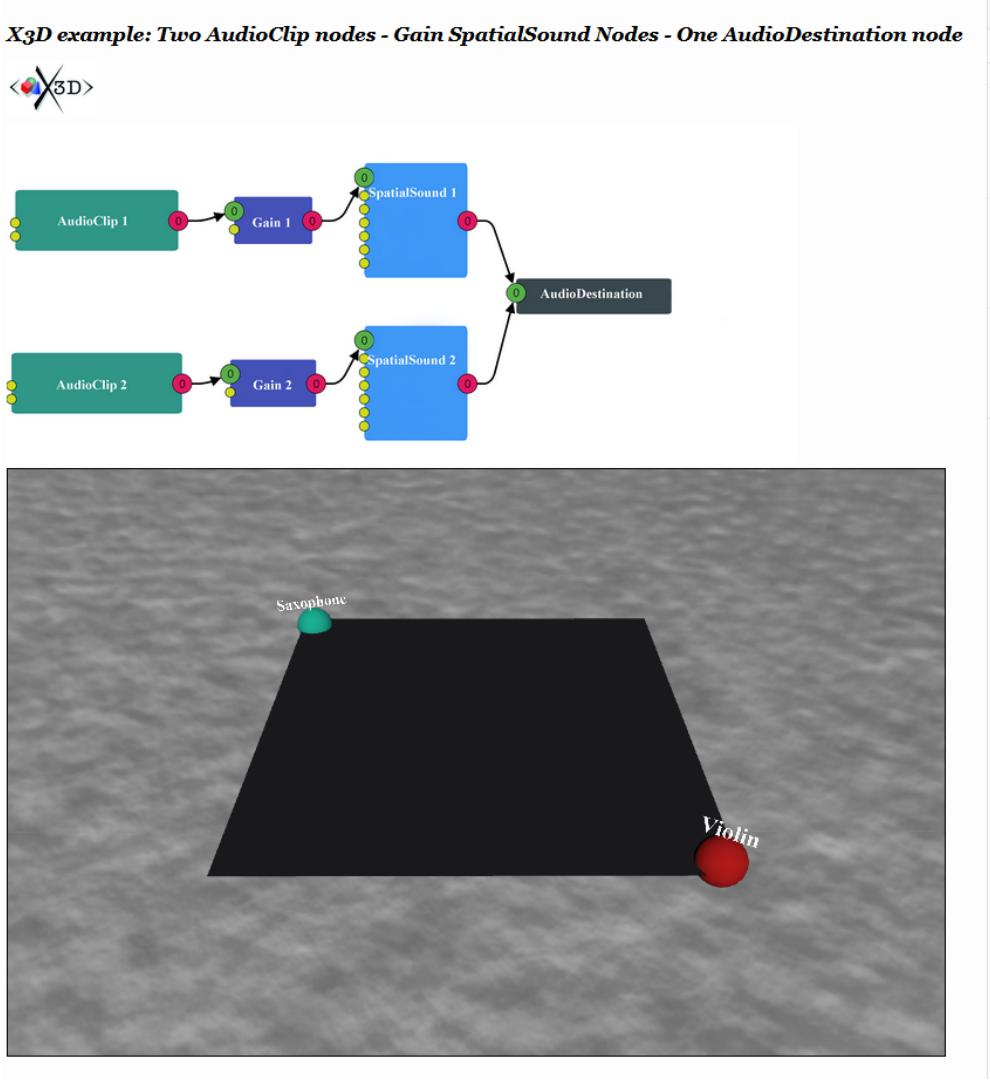
- StreamAudioDestination is an audio destination representing a MediaStream with a single MediaStreamTrack whose kind is "audio"

X3D Step by Step – 1st Example (1)

- Description: evaluates the attenuation of two different sound sources, while the camera (the user) is moving in the 3D scene. Through the immersion in the X3D scene the user could attend a rational navigation. Whenever the camera moves in the direction of an existing sound source, the sound strength of this source increases, while the sound strength of the other (the second one) decreases and vice versa
- X3D scene – use of new sound nodes
- X3DOM – registration of new nodes
- jQuery – parse the xml nodes
- Web Audio API – control the audio

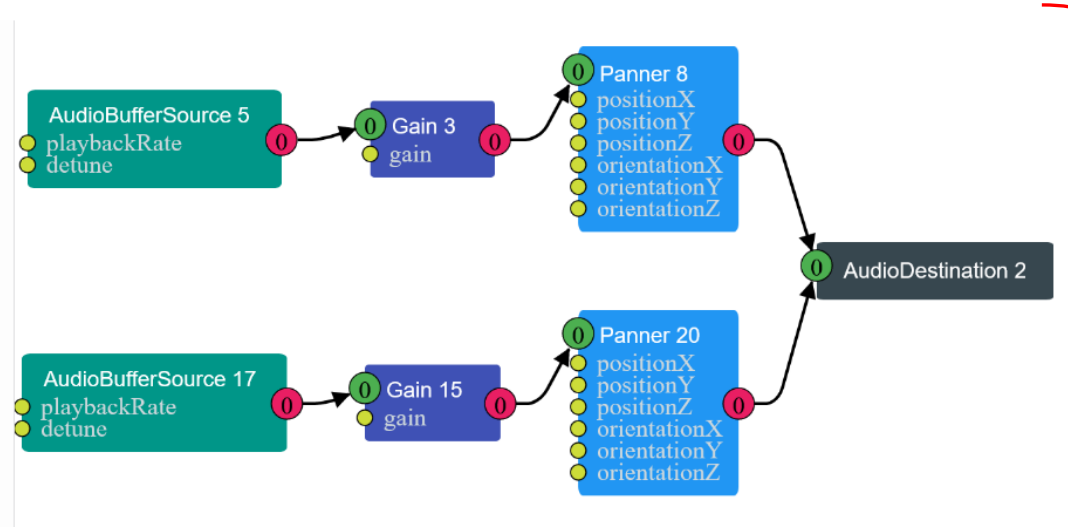
X3D Step by Step – 1st Example (2)

X3D Nodes



X3D-X3DOM Scene

Console Parse the X3D Scene



Web Audio API Graph Nodes

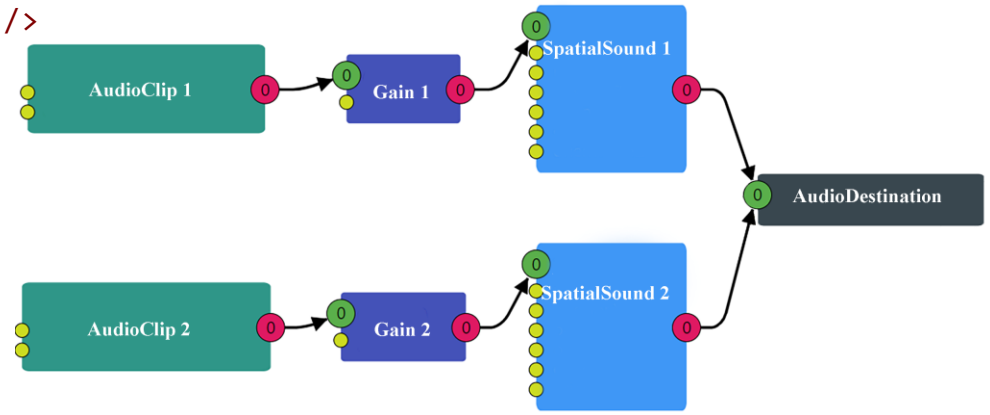
X3D Step by Step – 1st Example (3)

X3D CODE

```
<AudioDestination id='AudioDestination' channelInterpretation= 'speakers’>

<SpatialSound USE='Audio1' location='0 0 0' direction='1 0 0' coneInnerAngle='360' coneOuterAngle='0' coneOuterGain='0’
                distanceModel='linear' maxDistance='3500' enableHRTF='true' referenceDistance='1' rolloffFactor=‘1’>
       <Gain id='Gain1' gain= '1.0’>
          <AudioClip loop='true' url='sound/violin.mp3' pauseTime = '-1' resumeTime= '-1' stopTime = '-1' pitch = '1'/>
       </Gain>
   </SpatialSound>


<SpatialSound USE='Audio2' location='0 0 0' direction='1 0 0' coneInnerAngle='360' coneOuterAngle='0' coneOuterGain='0’
                 distanceModel='linear' maxDistance='3500' enableHRTF='true' referenceDistance='1' rolloffFactor='1'>
      <Gain id='Gain2' gain= '1.0' >
         <AudioClip loop='true' url='sound/saxophone.mp3' pauseTime = '-1' resumeTime= '-1' stopTime = '-1' pitch = '1' />
      </Gain>
   </SpatialSound>
</AudioDestination>
<ListenerPointSource id = 'listenerPoint' trackCurrentView = 'true' />
```



X3D Step by Step – 1st Example (4)

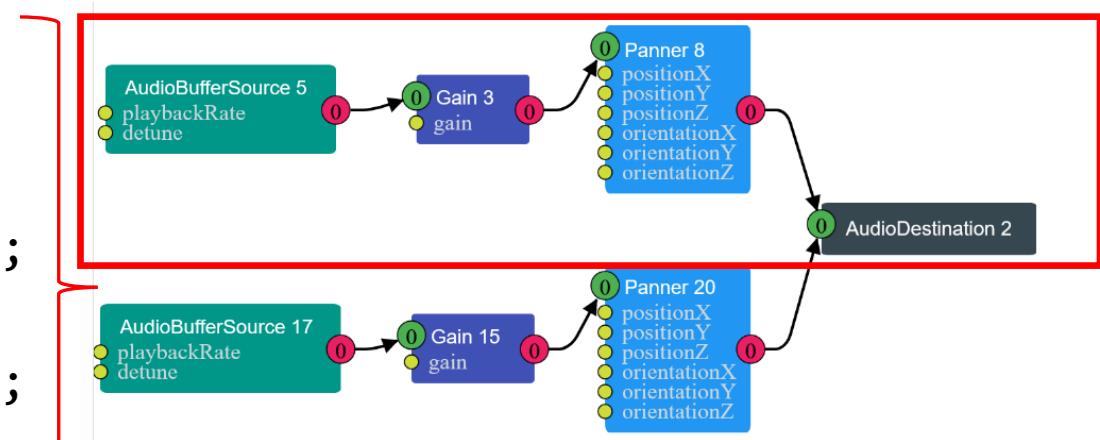
Javascript CODE

```
//Parse the xml nodes - find the "AudioDestination"
var numberSources = $("AudioDestination").children().length;
//Parse the xml nodes - find the n-th SpatialSound
newAudioNode.SpatialSound = $("SpatialSound:nth-of-type("+j+")");
//create Web Audio API PannerNode
newAudioNode.panner = context.createPanner();
//if find Gain-->create Web Audio API GainNode
if(newAudioNode.SpatialSound.children()[i].localName == "Gain")
    newAudioNode.volume = context.createGain();
//if find AudioClip-->create Web Audio API AudioBufferSourceNode
if(newAudioNode.Gain.children[0].localName == "AudioClip")
    newAudioNode.source = context.createBufferSource();
...
```

X3D Step by Step – 1st Example (5)

Javascript CODE

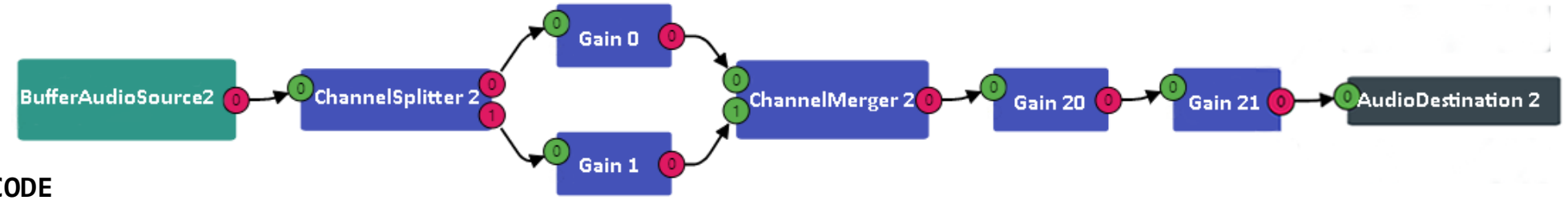
```
//connect source with GainNode
newAudioNode.source.connect(newAudioNode.volume);
//connect GainNode with PannerNode
newAudioNode.volume.connect(newAudioNode.panner);
//connect PannerNode with the destination
newAudioNode.panner.connect(context.destination);
...
//check if the ListenerPoint will be the camera
if($("#listenerPoint")[0].getAttribute("trackCurrentView") == "true")
    newAudioNode.ListenerPoint = x3dom.canvases[0].doc._scene.getViewpoint();
```



X3D Step by Step – 2nd Example (1)

- Description: assesses the capability of the audio channels split. It includes a simple sound source which can be moved right and left. Depending on the position of the sound source, the user can hear the produced sound from the corresponding output speaker
- New X3D nodes:
  - BufferAudioSource
  - ChannelSplitter
  - ChannelSelector
  - ChannelMerger
  - Gain
  - AudioDestination

X3D Step by Step – 2nd Example (2)



X3D CODE

```
<ChannelSplitter DEF='ChannelSplitter2' channelCountMode='explicit'>
    <BufferAudioSource DEF='BufferAudioSource2'/>
    <Gain DEF='Gain0' gain='-0.5'>
        <ChannelSelector DEF='ChannelSelector1' channelNumber='0'/>
    </Gain>

    <Gain DEF='Gain1' gain='0.2'>
        <ChannelSelector DEF='ChannelSelector2' channelNumber='1'/>
    </Gain>
</ChannelSplitter>

<AudioDestination DEF='AudioDestination2' channelInterpretation ='speakers'>
    <Gain DEF='Gain21' gain='0.5'>
        <Gain DEF='Gain20' gain='0.2'>
            <ChannelMerger DEF='ChannelMerger2'>
                  <ChannelSplitter USE='ChannelSplitter2'/>
             </ChannelMerger
        </Gain>
    </Gain>
</AudioDestination>
```

Conclusion – Next Steps

- Evaluation: more examples with the use of new X3D registered nodes
- Extension of the current examples using X_ITE Javascript library
- Development and demonstration of Acoustic Properties algorithms
- Developed more sophisticated 3D scenes using the X3D4 sound nodes

Contact

Eftychia Lakka

efilakka@gmail.com

University of South Wales, Pontypridd Wales UK

WEB3D 2020

3D for a Hyperconnected World

The 25th International ACM Conference on 3D Web Technology
November 9-13, 2020, Virtual Conference, Seoul, Korea

ACMSIGGRAPH web3D CONSORTIUM Korea Computer Graphics Society