

Web Audio API Nodes - Categories

General audio graph definition	Defining audio sources	Defining audio effects filters	Defining audio destinations	Data analysis and visualization	Splitting and merging audio channels	Audio spatialization
AudioContext	OscillatorNode	BiquadFilterNode	AudioDestinationNode	AnalyserNode	ChannelSplitterNode	AudioListener
AudioNode	AudioBuffer	ConvolverNode	MediaStreamAudioDestinationNode		ChannelMergerNode	PannerNode
AudioParam	AudioBufferSourceNode	DelayNode				
	MediaElementAudioSourceNode	DynamicsCompressorNode				
	MediaStreamAudioSourceNode	GainNode				
		WaveShaperNode				
		PeriodicWave				

Audio Spatialization Nodes

1. **AudioListener**: represents the position and orientation of the person listening to the audio scene.

The positionX, positionY, positionZ parameters represent the location of the listener in 3D Cartesian coordinate space. PannerNode objects use this position relative to individual audio sources for spatialization.

The forwardX, forwardY, forwardZ parameters represent a direction vector in 3D space.

Both a forward vector and an up vector are used to determine the orientation of the listener.

In simple human terms,

- the **forward** vector represents which direction the person's nose is pointing.
- the **up** vector represents the direction the top of a person's head is pointing.

These two vectors are expected to be linearly independent.

```
interface AudioListener {
  readonly attribute AudioParam positionX;
  readonly attribute AudioParam positionY;
  readonly attribute AudioParam positionZ;
  readonly attribute AudioParam forwardX;
  readonly attribute AudioParam forwardY;
  readonly attribute AudioParam forwardZ;
  readonly attribute AudioParam upX;
  readonly attribute AudioParam upY;
  readonly attribute AudioParam upZ;
  void setPosition (float x, float y, float z); DEPRECATED
  void setOrientation (float x, float y, float z, float xUp, float yUp, float zUp
); DEPRECATED
};
```

Attributes:

forwardX	Sets the x coordinate component of the forward direction the listener is pointing in 3D Cartesian coordinate space.
forwardY	Sets the y coordinate component of the forward direction the listener is pointing in 3D Cartesian coordinate space.
forwardZ	Sets the z coordinate component of the forward direction the listener is pointing in 3D Cartesian coordinate space.
positionX	Sets the x coordinate position of the audio listener in a 3D Cartesian coordinate space.
positionY	Sets the y coordinate position of the audio listener in a 3D Cartesian coordinate space.
positionZ	Sets the z coordinate position of the audio listener in a 3D Cartesian coordinate space.

upX	Sets the x coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.
upY	Sets the y coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.
upZ	Sets the z coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.

- PannerNode:** represents a processing node which positions / spatializes an incoming audio stream in three-dimensional space. The spatialization is in relation to the AudioContext's AudioListener (listener attribute).

```
[Exposed=Window,
  Constructor (BaseAudioContext context, optional PannerOptions options)]
interface PannerNode : AudioNode {
  attribute PanningModelType panningModel;
  readonly attribute AudioParam positionX;
  readonly attribute AudioParam positionY;
  readonly attribute AudioParam positionZ;
  readonly attribute AudioParam orientationX;
  readonly attribute AudioParam orientationY;
  readonly attribute AudioParam orientationZ;
  attribute DistanceModelType distanceModel;
  attribute double refDistance;
  attribute double maxDistance;
  attribute double rolloffFactor;
  attribute double coneInnerAngle;
  attribute double coneOuterAngle;
  attribute double coneOuterGain;
  void setPosition (float x, float y, float z); DEPRECATED
  void setOrientation (float x, float y, float z); DEPRECATED
};
```

Attributes:

coneInnerAngle	A parameter for directional audio sources, this is an angle, in degrees, inside of which there will be no volume reduction. The default value is 360. The behavior is undefined if the angle is outside the interval [0, 360].
coneOuterAngle	A parameter for directional audio sources, this is an angle, in degrees, outside of which the volume will be reduced to a constant value of coneOuterGain. The default value is 360. The behavior is undefined if the angle is outside the interval [0, 360].

coneOuterGain	A parameter for directional audio sources, this is the gain outside of the coneOuterAngle. The default value is 0. It is a linear value (not dB) in the range [0, 1]. An InvalidStateError MUST be thrown if the parameter is outside this range.
distanceModel	Specifies the distance model used by this PannerNode. Defaults to "inverse".
maxDistance	The maximum distance between source and listener, after which the volume will not be reduced any further. The default value is 10000. A RangeError exception MUST be thrown if this is set to a non-positive value.
orientationX	Describes the x component of the vector of the direction the audio source is pointing in 3D Cartesian coordinate space. Depending on how directional the sound is (controlled by the cone attributes), a sound pointing away from the listener can be very quiet or completely silent.
orientationY	Describes the y component of the vector of the direction the audio source is pointing in 3D cartesian coordinate space.
orientationZ	Describes the Z component of the vector of the direction the audio source is pointing in 3D cartesian coordinate space.
panningModel	Specifies the panning model used by this PannerNode. Defaults to "equalpower".
positionX	Sets the x coordinate position of the audio source in a 3D Cartesian system.
positionY	Sets the y coordinate position of the audio source in a 3D Cartesian system.
positionZ	Sets the z coordinate position of the audio source in a 3D Cartesian system.
upX	Sets the x coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.
upY	Sets the y coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.
upZ	Sets the z coordinate component of the up direction the listener is pointing in 3D Cartesian coordinate space.
refDistance	A reference distance for reducing volume as source moves further from the listener. The default value is 1. A RangeError exception MUST be thrown if this is set to a negative value.
rolloffFactor	Describes how quickly the volume is reduced as source moves away from listener. The default value is 1. A RangeError exception MUST be thrown if this is set to a negative value.

Web3D Extension Proposal – Focus only on spatialization

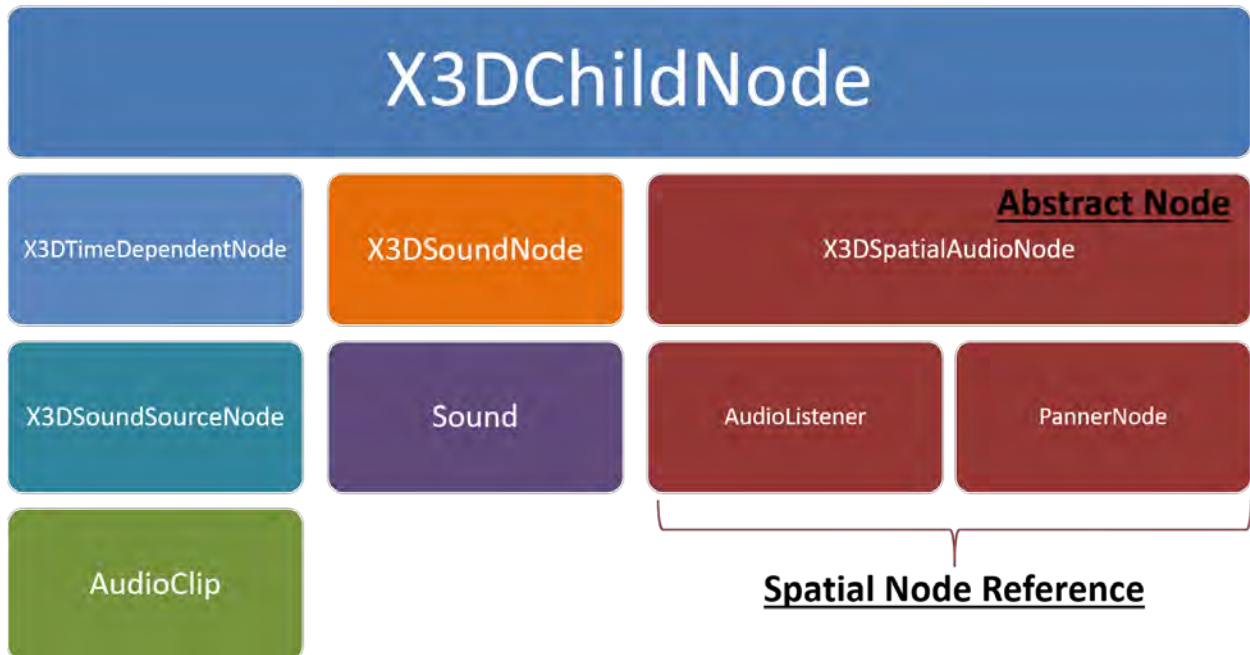
```
<pre class="node">AudioListener : AudioNode {
  SFFloat [in,out] positionX      0      (-&#8734;,&#8734;)
  SFFloat [in,out] positionY      0      (-&#8734;,&#8734;)
  SFFloat [in,out] positionZ      0      (-&#8734;,&#8734;)
  SFFloat [in,out] forwardX       0      (-&#8734;,&#8734;)
  SFFloat [in,out] forwardY       0      (-&#8734;,&#8734;)
  SFFloat [in,out] forwardZ      -1     (-&#8734;,&#8734;)
  SFFloat [in,out] upX            0      (-&#8734;,&#8734;)
  SFFloat [in,out] upY            1      (-&#8734;,&#8734;)
  SFFloat [in,out] upZ            0      (-&#8734;,&#8734;)
  <!-- Extra attributes -->
  SFInt32[in,out] gain            1      [0,&#8734;)
  SFBool [in,out] isViewpoint    TRUE
  <!-- if isViewpoint == TRUE
        newAudioNode.AudioListener = x3dom.canvases[0].doc._scene.getViewpoint();
        context.listener.setPosition(x3dom.canvases[0].doc._viewarea._movement.x-
newAudioNode.AudioListener._vf.position.x,
        x3dom.canvases[0].doc._viewarea._movement.y-
newAudioNode.AudioListener._vf.position.y,
        x3dom.canvases[0].doc._viewarea._movement.z-
newAudioNode.AudioListener._vf.position.z);
  -->
}
```

```
<pre class="node">PannerNode : AudioNode {
  SFFloat [in,out] positionX      0      (-&#8734;,&#8734;)
  SFFloat [in,out] positionY      0      (-&#8734;,&#8734;)
  SFFloat [in,out] positionZ      0      (-&#8734;,&#8734;)
  SFFloat [in,out] orientationX   1      (-&#8734;,&#8734;)
  SFFloat [in,out] orientationY   0      (-&#8734;,&#8734;)
  SFFloat [in,out] orientationZ   0      (-&#8734;,&#8734;)
  SFFloat [in,out] coneInnerAngle 360    [0,&#8734;)
  SFFloat [in,out] coneOuterAngle 360    [0,&#8734;)
  SFFloat [in,out] coneOuterGain  0      (-&#8734;,&#8734;)
  SFString [in,out] distanceModel  "inverse"
  SFFloat [in,out] maxDistance    10000  [0,&#8734;)
  SFString [in,out] panningModel   "HRTF"
  SFFloat [in,out] refDistance     1      [0,&#8734;)
  SFFloat [in,out] rolloffFactor   1      [0,&#8734;)
  <!-- Extra attributes -->
  SFInt32 [in,out] gain            1      [0,&#8734;)
}
```

```
</pre>
```

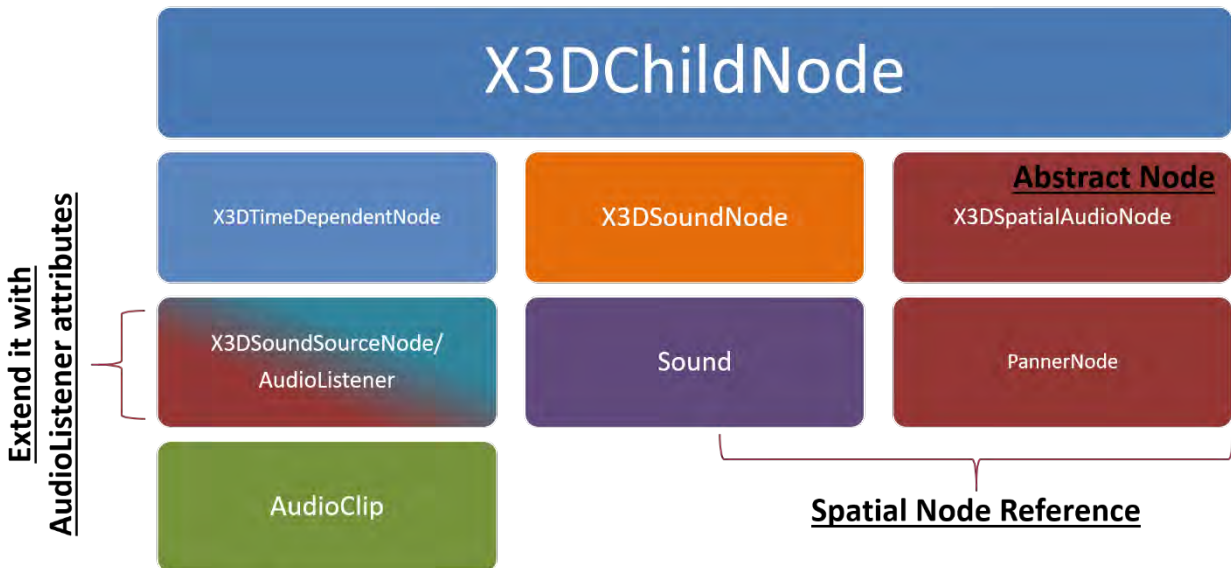
X3D scene example

1st Approach



```
<Transform USE='Audio1' />
  <X3DSpatialAudioNode>
    <PannerNode position='0 0 0' orientation='1 0 0' velocity='0 0 0'
      coneInnerAngle='360' coneOuterAngle='360'
      coneOuterGain='0' distanceModel='inverse'
      maxDistance='1' panningModel='HRTF'
      refDistance='1' rolloffFactor='1' gain =1 ></PannerNode>
    <AudioListener gain = 0.5 isViewpoint = true ></AudioListener>
    <AudioClip loop='true' url='\"sound/africa/beat.mp3\" />
  </X3DSpatialAudioNode>
```

2nd Approach



```
<Transform USE='Audio1' />
  <X3DSpatialAudioNode>
    <PannerNode position='0 0 0' orientation='1 0 0' velocity='0 0 0'
      coneInnerAngle='360' coneOuterAngle='360'
      coneOuterGain='0' distanceModel='inverse'
      maxDistance='1' panningModel='HRTF'
      refDistance='1' rolloffFactor='1' gain =1 >>/PannerNode>
    <AudioClip loop='true' url='"sound/africa/beat.mp3" gain = 0.5
      isViewpoint = true />
  </X3DSpatialAudioNode>
```

Pros: No need graph properties (connection between nodes)