Wwise Unity Integration

Unity-Wise Wwise API

\Authoring\Help

* Unity
* Wwise Integration Package
* Wwise Unity Integration
* Build your Unity Game for a Target Platform
* WwiseUnityDLC
* Build the Native Integration Plug-in from Source
* API
* Licensing (Free & Commercial)
* Wwise
* Using Unity Wwise Spatial Audio
Contact Us

If you have questions about this integration, please post your questions to our Q&A community forum.
Wwise Unity Integration:

- **Wwise Unity Integration 2017.2.0.6500.947**
  - 2017.2
  - 2017.2
- **Wwise Unity Integration 2017.1.4.6407.845**
- **Wwise Unity Integration 2017.1.3.6377.812**
- **Wwise Unity Integration 2017.1.2.6361.791**
- **Wwise Unity Integration 2017.1.1.6340.770**
- **Wwise Unity Integration 2017.1.0.6302.726**
  - 2017.1
  - 2017.1
- **Wwise Unity Integration 2016.2.4.6098.531**
- **Wwise Unity Integration 2016.2.3.6077.504**
- **Wwise Unity Integration 2016.2.2.6022.430**
- **Wwise Unity Integration 2016.2.1.5995.409**
- **Wwise Unity Integration 2016.2.1**
- **Wwise Unity Integration 2016.2.0**
- **Wwise Unity Integration 2016.1.3**
- **Wwise Unity Integration 2016.1.2**
- **Wwise Unity Integration 2016.1.1**
- **Wwise Unity Integration 2016.1**
- **Wwise Unity Integration 2015.1.4**
- **Wwise Unity Integration 2015.1.3**
- **Wwise Unity Integration 2015.1.2**
- **Wwise Unity Integration 2015.1.1**
- **Wwise Unity Integration 2015.1**
- **Wwise Unity Integration 2014.1.6**
- **Wwise Unity Integration 2014.1.5**
- **Wwise Unity Integration 2014.1.4**
- **Wwise Unity Integration 2014.1.3**
- **Wwise Unity Integration 2014.1.2**
- **Wwise Unity Integration 2014.1.1**
- **Wwise Unity Integration 2014.1**
- **Wwise Unity Integration 2013.2.9**
- **Wwise Unity Integration 2013.2.8**
- Wwise Unity Integration 2013.2.5
- Wwise Unity Integration 2013.2.4
- Wwise Unity Integration 2013.1.1
- Wwise Unity Integration 2013.1
Wwise Unity Integration 2017.2.0.6500.947

Wwise 2017.2.0Wwise SDKUnity 2017.3

- Wwise SDK: 2017.2.0
- UnityUnity 5.55.62017.12017.22017.3 (PersonalPro)

Note: : Unity

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mac Windows</td>
<td>Unity 2017.3.0f3</td>
</tr>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td>Unity 2017.2.0f3</td>
</tr>
<tr>
<td>UWP (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Unity for Nintendo Switch 3.1.2 (Unity 5.6.4</td>
</tr>
<tr>
<td></td>
<td>with NintendoSDK 3.5.2)</td>
</tr>
</tbody>
</table>

2017.2

- breaking changes

- WG-25994: C# Using the Audio Input Source Plug-in in Unity.
- WG-27337: MIDI Sending MIDI to Wwise.
- **WG-28541**: AkAk
- **WG-33501**: SoundBank
- **WG-34446**: AkCallbackManager
Wwise Unity Integration 2017.1.4.6407.845

- Wwise SDK: 2017.1.4
- UnityUnity 5.55.62017.12017.2 (PersonalPro)
  
  **Note:** : Unity

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>UWP (Windows )</td>
<td>Unity 2017.2.0f3</td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Unity for Nintendo Switch 3.1.2 (Unity 5.6.4 with NintendoSDK 3.5.2)</td>
</tr>
</tbody>
</table>

- **WG-35168:** Switch
- **WG-35383:** WwiseTypesID
- **WG-35384:** AkAudioListener AkInitializer AkTerminator Awake() OnEnable()Editor
- **WG-35513:** AkGameObj
- **WG-35958:** Unity 2017.3Unity 2017.3
**Wwise Unity Integration 2017.1.3.6377.812**

Wwise 2017.1.3
Wwise SDK Unity 2017.1

- Wwise SDK: 2017.1.3
- Unity: 5.5.62

**Note:** Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Unity 2017.2.0f3</td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>UWP (Windows)</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Switch 1.0.1 Unity (Unity 5.5.0p1 with Nintendo SDK 1.4.0)</td>
</tr>
</tbody>
</table>

: 

- **WG-34855**: `AkSoundEngine.GetCurrentLanguage()`
- **WG-35075**: `Unity 2017.2 Wwise Launcher`
Wwise Unity Integration 2017.1.2.6361.791

Wwise 2017.1.2 Wwise SDK

- Wwise SDK: 2017.1.2
- Unity: Unity 2017.1 (PersonalPro)
  
  **Note:** Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td>Unity 5.6.1p1</td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>UWP (Windows)</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Switch 1.0.1 Unity (Unity 5.5.0p1 with Nintendo SDK 1.4.0)</td>
</tr>
</tbody>
</table>
**Wwise Unity Integration 2017.1.1.6340.770**

Wwise 2017.1.1 Wwise SDK Unity 2017.1

- Wwise SDK: 2017.1.1
- Unity: Unity 2017.1 (PersonalPro)

**Note:** Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mac Windows</td>
<td>Unity 2017.1.0f3</td>
</tr>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td>Unity 5.6.1p1</td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>UWP (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Switch 1.0.1 Unity (Unity 5.5.0p1 with Nintendo SDK 1.4.0)</td>
</tr>
</tbody>
</table>

: 

- **WG-33018:** Decode Banks
- **WG-33818:** iOS Android
- **WG-34090:** WSA Unity PluginTLS Allocator Error
- **WG-34188:** System.EventHandler fully qualified name
- **WG-34205:** UnityEditor.Menu UnityEditor.MenuItem fully qualified name
Wwise Unity Integration 2017.1.0.6302.726

Wwise 2017.1.0 Wwise SDK
Unity 2017.1

- Wwise SDK: 2017.1.0
- Unity: Unity 2017.1 (PersonalPro)
  
  **Note:** : Unity

<table>
<thead>
<tr>
<th></th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mac Windows</td>
<td>Unity 2017.1.0f3</td>
</tr>
<tr>
<td>Android iOS Linux</td>
<td></td>
</tr>
<tr>
<td>PS4 PS Vita tvOS UWP (Windows)</td>
<td>Unity 5.6.1p1</td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Switch 1.0.1 Unity (Unity 5.5.0p1 with Nintendo SDK 1.4.0)</td>
</tr>
</tbody>
</table>

- **2017.1**

- Wii U
- GameObject Unity Integration Extensions
- AkCallbackManager 2017.1
- WG-27479: \texttt{AkInitializer}
- \textbf{WG-30791}: \texttt{WwiseTypes} \texttt{WwiseTypes}
- \textbf{WG-31155} \quad \texttt{AkMemSettings} \quad \texttt{AkChannelConfig} \quad \texttt{SoundEngine}
- \textbf{WG-31735} \quad \texttt{GameObject} \quad \textbf{Unity Integration}
- \textbf{WG-32348} \quad \texttt{SoundBanksInfoXML}
- \textbf{WG-32657} \quad \texttt{foreach}
- \textbf{WG-33303} \quad \texttt{AK_MusicPlaylistSelect}
- \textbf{WG-34003}:
**Wwise Unity Integration 2016.2.4.6098.531**

Wwise 2016.2.4

- Wwise SDK: 2016.2.4
- Unity: Unity 5.6 (PersonalPro)

**Note:** Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Unity 5.6.1p1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
</tbody>
</table>

Switch: Switch 1.0.1 Unity (Unity 5.5.0p1 with Nintendo SDK 1.4.0)

- **WG-33395:**
**Wwise Unity Integration 2016.2.3.6077.504**

Wwise 2016.2.3 Wwise SDK

- Wwise SDK: 2016.2.3
- Unity: Unity 5.6 (PersonalPro)

**Note:** : Unity

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Unity 5.6.0f3</td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Switch</td>
<td>Unity for Switch 0.6.1</td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.4.2f2</td>
</tr>
</tbody>
</table>

- WG-32536: Nintendo Switch
- WG-32623: Android
**Wwise Unity Integration 2016.2.2.6022.430**

Wwise 2016.2.2Wwise SDK

- **Wwise SDK**: 2016.2.2
- **Unity**: Unity 5.5 (PersonalPro)

  **Note**: Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Unity 5.5.0p4</td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows )</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.4.2f2</td>
</tr>
</tbody>
</table>

- **WG-31862**: enum
Wwise Unity Integration 2016.2.1.5995.409

Wwise 2016.2.1 Wwise SDK

- Wwise SDK: 2016.2.1
- Unity: Unity 5.5 (PersonalPro)

Note: : Unity

<table>
<thead>
<tr>
<th></th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Unity 5.5.0f3</td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.4.2f2</td>
</tr>
</tbody>
</table>

- WG-32006: GameObject"Unknown Game Object ID"
Wwise Unity Integration 2016.2.1

Wwise 2016.2.1 Wwise SDK

- Wwise SDK: 2016.2.1
- Unity: Unity 5.5 (PersonalPro)

Note: Unity 4

<table>
<thead>
<tr>
<th>Wwise Unity Integration</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>Unity 5.5.0f3</td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows)</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.4.2f2</td>
</tr>
</tbody>
</table>

- WG-27085: AuxSends
- WG-31127: AK_MusicSyncUserCue
- WG-31650: SoundEngineAkGameObjFixedNullReferenceException
- WG-31651: SetObjectPositionAkGameObj
- WG-31862: enum

- Unity case #861189: Unity Windows StoreLauncher
Wwise Unity Integration 2016.2.0

Wwise 2016.2.0 Wwise SDK

- Wwise SDK: 2016.2.0
- Unity: Unity 5.4 (PersonalPro)

**Note:** Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td>Unity 5.4.1p2</td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store (Windows)</td>
<td>Unity 5.2.4f1</td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td></td>
</tr>
</tbody>
</table>

- Windows Phone 8.1
- Windows Store 8.08.13
- Windows Store Universal Windows Platform (UWP)
- Xbox 360
- PS3

WG-30571: Library Wwise Unity Integration
- WG-30960: AkGameObjUnity
- WG-31507: DurationMediaIDbStreaming
### Wwise Unity Integration 2016.1.3

Wwise 2016.2.3 Wwise SDK

- Wwise SDK: 2016.1.3
- Unity: Unity 5.4 (PersonalPro)

**Note:** Integration Unity 4 Unity 4Wwise Integration Unity

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.4.0p2</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.2.4f1</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>
Wwise Unity Integration 2016.1.2

Wwise 2016.1.2 Wwise SDK Wwise Unity

- Wwise SDK: 2016.1.2
- Unity: Unity 5.4 (PersonalPro)

Note: Unity 4

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.4.0p2</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.2.4f1</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

WG-30567 Fixed: WAV

- SWIGzipzipSWIG SWIG
- Wwise 2014.1.42016.12015.1.6
2016.1
• UnityUnity 5.4Wwise 2016.1.2 Unity 5.4Wwise 2016.1.2
Wwise Unity Integration 2016.1.1

Wwise 2016.1.1 Wwise SDK

- Wwise SDK: 2016.1.1
- Unity: Unity 5.3 (PersonalPro)

**Note:**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td>Unity 5.3.5p6</td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>tvOS</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.2.4f1</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

WG-30021: SoundBankSoundBank
WG-30228: Linux
WG-30231: tvOS
WG-30259: GC.Collect AkGameObj

WG-30128: VitaSoundBank
• WG-30139: Wii USoundBank
**Wwise Unity Integration 2016.1**

Wwise 2016.1 Wwise SDK

- **Wwise SDK: 2016.1**
- **Unity: Unity 5.3 (PersonalPro)**

  **Note:** Unity

<table>
<thead>
<tr>
<th></th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td>Unity 5.3.4p3</td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 5.2.4f1</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

- **WG-25675 : WwiseSoundBank** Wwise Picker
  "Generate SoundBanks"
- **WG-27583 Fixed: UnitySoundBank** Unity SoundBank
- **WG-28175 : WwiseGlobal**
- **WG-26011 : SceneAK Audio Listener** Default Unity Audio Listener
- SWIGzipzipSWIG
  SWIG
- Wwise 2014.1.42016.12015.1.6
  2016.1

: 

- WG-30021: EditorDecodeBanks
- WG-30021: DecodeBanksSoundBankSoundBank
  DecodedBanks
- WG-30128: VitaSoundBank
- WG-30139: WiiUSoundBank
## Wwise Unity Integration 2015.1.4

Wwise 2016.1 Wwise SDK

- Wwise SDK: 2015.1.4
- Unity: Unity 4.6 Pro Unity 5.2 (Personal Pro)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td>Unity 5.2.2p1</td>
</tr>
<tr>
<td>Metro</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Phone</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 4.3.7f1 Wii U2.2.5</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

: 

- WG-28412 : AkGameObj Prefab
- WG-28723 : PS4
Wwise Unity Integration 2015.1.3

Wwise 2015.1.3 Wwise SDK Universal Windows Platform

- Wwise SDK: 2015.1.3
- Unity: Unity 4.6 Pro Unity 5.2 (Personal Pro)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.2.2p1</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Store Apps</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 4.3.7f1 Wii U2.2.5</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

- Universal Windows Platform
  - Windows Store Apps
  - Windows Store Apps SDK 8.1
  - Windows Phone 8.0
  - Windows Store Apps SDK 8.0

- WG-25945 : WwiseUnity WwiseGlobal Wwise Unity
- WG-26011 : SceneAK Audio Listener Default Unity Audio Listener
- WG-28108 : Editor Editor
- WG-28175 : WwiseGlobal
- WG-28479 : SetupMain Camera Setup
- WG-28526 : Unity EditorGameObjectsSoundEngine
## Wwise Unity Integration 2015.1.2

Wwise 2016.1 Wwise SDK

- Wwise SDK: 2015.1.2
- Unity: Unity 4.6.5 Pro
- Unity 5.1.2p2 (Personal  Pro)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td>Unity 5.2.0f3</td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>Metro</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td></td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Phone</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 4.3.7f1Wii U2.2.5</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

- WG-27029 : Unity GameObject
- WG-28200 : Everything(0, 0, 0)GameObject3D L0

AkGameObj
Wwise Unity Integration 2015.1.1

Wwise 2015.1.1 Wwise SDK

- Wwise SDK: 2015.1.1
- Unity: Unity 4.6.5 Pro Unity 5.1.2p2 (Personal Pro)

<table>
<thead>
<tr>
<th></th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>Metro</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.1.2p2</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PS Vita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Phone</td>
<td></td>
</tr>
<tr>
<td>Xbox One</td>
<td></td>
</tr>
<tr>
<td>Wii U</td>
<td>Unity 4.3.7f1 Wii U 2.2.5</td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
</tbody>
</table>

: 

- WG-27977 : Unity
- WG-28030 : Wii U/Unity: Release RPLTRCNDEBUG
- WG-28042 : Unity: Wwise
- WG-28044 : Unity: Wwise
- WG-28046 : AkInitializer
- WG-28048 : Wwise ProfilerGameObject
Wwise Unity Integration 2015.1

Wwise 2015.1 Wwise SDK

- Wwise SDK: 2015.1

- WG-25669: Wwise PickerAuto PopulateMac Editor

- WG-27079 Fixed: WwiseSettings.xml
Wwise Unity Integration 2014.1.6

Wwise 2014.1.6 Wwise SDK

- Wwise SDK: 2014.1.6
- Unity: Unity 4.6.5 Pro Unity 5.1.2p2 (Personal Pro)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>Metro</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.0.2p3</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PSVita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Phone</td>
<td></td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
<tr>
<td>Xbox One</td>
<td>Unity 5.0.2p4</td>
</tr>
</tbody>
</table>

WG-27585 Fixed: Wwise Picker
WG-27624 Fixed: (PS3) SetListenerPostionPS3

Unity:

- Xbox One: UnitySoundEngine
- Unity 4 Windows Store Apps: DllNotFoundException
  unity_troubleshooting

WG-27585 Fixed: Wwise Picker
# Wwise Unity Integration 2014.1.5

Wwise 2014.1.5 Wwise SDK

- Wwise SDK: 2014.1.5
- Unity: Unity 4.6.5 ProUnity 5.1.2p2 (Personal Pro)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Wwise Unity Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
</tr>
<tr>
<td>Mac</td>
<td></td>
</tr>
<tr>
<td>Metro</td>
<td></td>
</tr>
<tr>
<td>PS3</td>
<td>Unity 5.0.2p3</td>
</tr>
<tr>
<td>PS4</td>
<td></td>
</tr>
<tr>
<td>PSVita</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td>Windows Phone</td>
<td></td>
</tr>
<tr>
<td>Xbox 360</td>
<td>Unity 5.0.1f1</td>
</tr>
<tr>
<td>Xbox One</td>
<td>Unity 5.0.2p4</td>
</tr>
</tbody>
</table>

- WG-25669: Wwise PickerAuto PopulateMac Editor

**Unity:**

- Xbox One: UnitySoundEngine
- Unity 4Windows Store Apps: DllNotFoundException
  unity_troubleshooting

- Android x86
- Windows Store AppsScripting Define Symbols
- WG-27108 Fixed: DestroyUnity
- WG-25733 Fixed: Windows/iOS/Mac
- WG-26875 Fixed: AkMemBankLoader
Wwise Unity Integration 2014.1.4

Wwise 2017.1.1Wwise SDKUnity 2017.1

- Wwise SDK: 2014.1.4
  - Unity: Unity 4 ProUnity 5 PersonalPro

: 

- WG-26780 Fixed: WwiseUnity
  - WG-26837 Fixed: AkBankManager

: 

- WG-25669: Wwise PickerAuto PopulateMac Editor
  - WG-25733: WindowsiOS/Mac

: 

- Windows Phone 8.1Metro
  - Unity 5Wwise Unity IntegrationPlugin Importer API
    `<UNITY_PROJECT_ROOT>/Assets/Plugins`

-- : Unity 4Unity 5:

1. 
2. Unity 4Unity 5Unity -- -# Unity 52014.1.4
   Unity
3. "Start"
4. 
5. Unity 5Wwise Unity Integration
Wwise Unity Integration 2014.1.3

Wwise 2014.1.3 Wwise SDK

- Wwise SDK: 2014.1.3

- WG-25669: Wwise PickerAuto PopulateMac Editor
- WG-25733: Windows iOS/Mac
Wwise Unity Integration 2014.1.2

Wwise 2014.1.2 Wwise SDK

- Wwise SDK: 2014.1.2
- Unity Editor (October XDK) Wwise SDK (November XDK) XDK
  Xbox One

: 

- WG-26305 Fixed: Null 64-bit
- WG-26337 Fixed: Mac 64-bit
- WG-26385 Fixed: Android
- WG-26395 Fixed: MSBUILD
- WG-26430 Fixed: iOS SetBasePath() 

: 

- WG-25669: Wwise PickerAuto PopulateMac Editor
- WG-25733: Windows iOS/Mac
Wwise Unity Integration 2014.1.1

Wwise 2014.1.1 Wwise SDK

- Wwise SDK: 2014.1.1
- Unity: 4.x 5.0

- Wii U
  - Plug-in Registration
  - Unity "Wwise Settings"Wwise
  - GeneratedSoundBanksPopulate

- WG-26201: C# System:
  - WG-25669: Wwise PickerAuto PopulateMac Editor
  - WG-25733: WindowsiOS/Mac

- WG-25669: Wwise PickerAuto PopulateMac Editor
- WG-25733: WindowsiOS/Mac
Wwise Unity Integration 2014.1

Wwise 2014.1 Wwise SDK 2013.2 Wwise Unity Integration

- 2013.2 Wwise Unity Integration

- Wwise SDK: 2014.1
- Unity: 4.x

--

- Android
- iOS
- Linux*
  - 32-bit
  - 64-bit
- Mac OS X (10.6 )
- PS3
- PS4
- PS Vita
- Windows
  - 32-bit
  - 64-bit
- Windows 8 (Metro)
  - IntelARM Windows Store App
- Windows Phone 8.0
- Xbox360
- Xbox One

:
- **Wwise**
- 2013.2 Wwise Unity Integration
- Wwise/Wwise Unity IntegrationUUIDWwiseUnity

- **Adding New Triggers for Wwise Events**
- Wwise
- AkSoundEngine
- WwiseGlobalAkListener

- WG-25783: Callback Manager
- WG-25677: Mac

- WG-25669: Wwise PickerAuto PopulateMac Editor
- WG-25733: Windows/iOS/Mac
Wwise Unity Integration 2013.2.9

Wwise 2013.2.9 Wwise SDK

--

- Wwise SDK: 2013.2.9
- Unity: 4.x

--

- Android
- iOS
- Linux*
  - 32-bit
  - 64-bit
- Mac OS X (10.6)
- PS3
- PS4
- PS Vita
- Windows
  - 32-bit
  - 64-bit
- Windows 8 (Metro)
  - IntelARM
- Windows Phone 8.0
- Xbox360
- Xbox One

:

- Windows Phone 8.0
Wwise Unity Integration 2013.2.8

Wwise 2013.2.8 Wwise SDK Wwise-Unity

---

- **Unity**
- Wwise Picker
- Wwise Unity Integration 2015.1
- Wwise SDK

- Wwise SDK: 2013.2.8
- Unity: 4.x

---

- Android
- iOS
- Linux*
  - 32-bit
  - 64-bit
- Mac OS X (10.6)
- PS3
- PS4
- PS Vita
- Windows
  - 32-bit
  - 64-bit
- Windows 8 (Metro)
  - IntelARM
- Windows Phone 8.0
- Xbox360
- Xbox One
- Xbox One
- PS4
- Linux
  * Unity Integration for Linux Wwise 2013.2.8 Linux BETA
- Windows Phone 8.0

- WindowsMetro
- UnityScripting Define Symbol
- Integration
- Unity Editor
- Integration
- Windows IDE Visual Studio 2008 2010
- WindowsMetro
Wwise Unity Integration 2013.2.5

Wwise 2013.2.5 Wwise SDK

- Wwise SDK: 2013.2, 2013.2.x
- Unity: 4.x

--

- Android
- iOS
- Mac OS X (10.6)
- PS3
- PS4
- PS Vita
- Windows
  - 32-bit
  - 64-bit
- Windows 8 (Metro)
  - IntelARM Windows Store App
- Xbox 360
- Xbox One

: PS4
: PS Vita
: Xbox One
: Android: API AddBasePath() I/O POSIX

: WG-24351: iOS:
Wwise Unity Integration 2013.2.4

Wwise 2013.2.4 Wwise SDK

- Wwise SDK: 2013.2, 2013.2.x
- Unity: 4.x

--

- Windows 8 (Metro)
  - IntelARM
- Windows Store App
- Windows
  - 32-bit
  - 64-bit
- Mac OS X (10.6)
- iOS
- Android
- Xbox360
- PS3

: Integration  Wwise > Help
- Wwise SDKVersion.txt

: WG-24080: Windows APIMac APIAndroid

- Android armeabi
Wwise Unity Integration 2013.2.1

Wwise 2013.2.1Wwise SDK

- Wwise SDK: 2013.2, 2013.2.1
- Unity: 4.x.x

--

- Windows 8 (Metro)
  - IntelARM Windows Store App
- Windows
  - 32-bit
  - 64-bit
- Mac OS X (10.6)
- iOS
- Android
- Xbox360
- PS3

Unity

- Unity Integration
- Unity Editor
- WwiseUnity EditorUI:
  - SoundBank ID C++ C#
- Unity 4 Scripting Define Symbols

IntegrationDemo

Android: SoundBank Android Expansion Files (OBB)
- WG-23781: Bank
- WG-23734: WindowsMarker
- WG-23345: Unity Editor
- WG-23436: Unity:
- WG-23423: Unity GameObject
- WG-22533: Unity: API

- iOS

- Unity

- iOS Xcode API

- Build for iOS

- UI

- Apple

- SWIG 2.0.11 Mac SWIG pg_compileswig
Wwise Unity Integration 2013.1.1

Wwise 2016.1 Wwise SDK Integration
Wwise Unity Integration 2013.1

Wwise 2013.1 Wwise SDK Android

- Wwise SDK: 2012.2.x, 2013.1
- Unity: 3.4.x, 3.5.x, 4.x

--

- Windows 8
  - IntelARMWindows Store App
- Windows (32bit)
- Windows (64bit)
- Mac OS X (10.6)
- iOS
- Android
- Xbox360
- PS3

: 

- Android: SoundBanksAndroid (apk) LoadBank() API
- Windows 64bit
- Windows 8

: WG-22948: PS3

- WG-22938: Mac OSXUnity Integration
- WG-22334: UnityMusic userCue
- WG-22329: Unity IntegrationPostEvent()
- WG-22255: Unity Android:
- WG-22165: 3DiOSAndroid
- WG-21933: AkCallbackManager.csfloatUserCue
- WG-21365: Unity iOS: iOS
• WG-22533: API

• 1UnityWindowsMac

• quietverbose
• Python
• IntegrationUnity

• Integration for PS3Visual Studio 2010
• Python 2.6Python 2.7.x3.x
• IntegrationUNITY_PROJECT_ROOTIntegration
  UNITY_PROJECT_ROOT
  Plug-in from Source
• AndroidPostprocessBuildPlayerUnity
• IntegrationDemoUnity4
2017.2

Wwise 2017.2Unity Integration:

- **Edit Mode**
  - Edit
- **WwiseGlobal**
- **Wwise Audio Input**
- **MIDI**
- **Automatic SoundBank Management**
Edit Mode

WwiseUnityEditPlayWwise

AkEvent (AkAmbient) The Play/Stop button plays and stops individual Events. When multiple objects that have AkEvents are selected, the Play Multiple and Stop Multiple buttons play or stop all currently selected Events. The Stop All button stops all currently playing Events.

Edit

EditAkBankLoad On:AwakeUnload On:Never
WwiseGlobal

WwiseGlobalAkEditorEventPlayerEdit
AkEventWwiseEventsWwiseGlobal
AkGameObj AkAudioListenerEdit AkEnvironment AkGameObj AkRoom

pg_installprojectchange sect_audio_i

Audio Input Wwise Audio InputC#

Source Plug-in in Unity.
MIDI

WwiseMIDIC# Sending MIDI to Wwise.
Automatic SoundBank Management

SoundBank
Deploying SoundBanks in single-platform projects
Deploying SoundBanks in multi-platform projects
Spatial Audio

UnityAPI
Wwise Unity Integration »
### 2017.2

<table>
<thead>
<tr>
<th>Pre-2017.2:</th>
<th>New in 2017.2:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ErrorLevel</td>
<td>AK.Monitor.ErrorLevel</td>
</tr>
<tr>
<td>DynamicSequenceType</td>
<td>AK.SoundEngine.DynamicSequence.DynamicSequenceType</td>
</tr>
<tr>
<td>MultiPositionType</td>
<td>AK.SoundEngine.MultiPositionType</td>
</tr>
<tr>
<td>PreparationType</td>
<td>AK.SoundEngine.PreparationType</td>
</tr>
<tr>
<td>RTPCValue_type</td>
<td>AK.SoundEngine.Query.RTPCValue_type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pre-2017.2:</th>
<th>New in 2017.2:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iterator</td>
<td>AkIterator</td>
</tr>
<tr>
<td>Playlist</td>
<td>AkPlaylist</td>
</tr>
<tr>
<td>PlaylistItem</td>
<td>AkPlaylistItem</td>
</tr>
</tbody>
</table>
Wwise Unity Integration »
2017.1

Wwise 2017.1 Unity Integration:

- WwiseTypes
- Unity Integration
- 3D Busses
AkEvent

AkEnvironment AkEnvironmentPortal Rigidbody

Rigidbody AkEnvironmentAkEnvironmentPortal

Rigidbody AkGameObj"Environment aware" RigidbodyEditor

AkGameObj-AkEnvironment interactions require a Rigidbody component on the object or the environment. AkGameObj-AkEnvironment Rigidbody

WwiseSettings Show Warning for Missing Rigidity
public class Footsteps : MonoBehaviour
{
    [Header("Wwise Types")]  
    public AK.Wwise.Event FootStepEvent = null;  
    public AK.Wwise.RTPC SpeedRTPC = null;  
    public AK.Wwise.Switch UnderFootMaterialSwitch = null;  
    public AK.Wwise.Bank FootStepBank = null;  

    void InitializeSound() { FootStepBank.Load(); }  
    void FinalizeSound() { FootStepBank.Unload(); }  

    void PlayFootStepSound(float speed)  
    {  
        SpeedRTPC.SetValue(gameObject, speed);  
        UnderFootMaterialSwitch.SetValue(gameObject);  
        FootStepEvent.Post(gameObject);  
    }  
}
Unity Integration

AkSoundEngine Launcher
3D Busses

Wwise3D
AkAudioListener  AkGameObj  AkAudioListener

AkAudioListener
Apply Position Offset

Environment Aware:

Initial Listener List  Use Default Listeners
Listener 0  None (Ak Audio Listener)

Add Listener

Wwise Unity Integration Mon Jan 8 10:46:17 2018  doxygen 1.6.3
2017.1

**AkCallbackManager**

- `AkCallbackManager.AudioInterruptionCallback()` has
  ```csharp
  public delegate AKRESULT AudioInterruptionCallback(bool in_bEnterInterruption, object in_Cookie);
  ```

- `AkCallbackManager.BGMCallback()` has a new signature:
  ```csharp
  public delegate AKRESULT BGMCallback(bool in_bOtherAudioPlaying, object in_Cookie);
  ```

- `AkCallbackManager.EventCallback()` has a new signature:
  ```csharp
  public delegate void EventCallback(object in_cookie, AkCallbackType in_type, AkCallbackInfo in_info);
  ```

- `AkCallbackManager AkCallbackInfoC++

- `gameObjID` ulong IntPtr
- `AkCallbackManager.AkMidiEventCallbackInfo AkMIDIEventCallbackInfo"MIDI"
- `AkCallbackManager.AkMusicSyncCallbackInfoBase AkMusicSyncCallbackInfo`
- **AkCallbackManager.AkMusicSyncCallbackInfo.segmentInfo**
  ```csharp
  AkMusicSyncCallbackInfo.segmentInfo_*
  ```
Wwise Unity Integration
Unity

WwiseUnityIntegration UnityPackageUnityWwiseUnity
Editor

- Wwise Integration Package
- Unity
- SoundBank
- Wwise Integration Package
Wwise Unity Integration » Unity
Wwise:

Unity:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Required components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unity 5 PersonalPro</td>
<td></td>
</tr>
<tr>
<td>iOS</td>
<td>Xcode 4.2 or above with compatible iOS SDK</td>
</tr>
<tr>
<td>Linux</td>
<td>libSDL2 (See Linux)</td>
</tr>
<tr>
<td>Windows</td>
<td>DirectX End-User runtime</td>
</tr>
</tbody>
</table>

The x64 Visual Studio 2013 redistributable

Windows 32-bit Debug

The x86 Visual Studio 2013 redistributable
Wwise Integration Package

Wwise Launcher

**Warning:**
- LauncherUnity
- WwiseUnity(Warning)
- UnityUnity
Unity Integration

Launcher

Wwise Unity Integration Mon Jan 8 10:46:17 2018
Wwise Unity Integration » Unity
Unity

- Edit > Project Settings > AudioDisable Audio

- "Create WwiseGlobal GameObject""Wwise Global"Wwise
  The WwiseGlobal object is also given
  AkEditorEventPlayer, AkAudioListener and AkGameObj scripts
  which are used for previewing AkEvent objects in edit mode. A
  Rigidbody component is added such that interactions between
  AkEnvironment, AkRoom and AkGameObj can also be previewed.

- Script Execution Order Wwise (AkInitializer, AkGameObj,
  AkAudioListener, AkBank, AkTerminator)

- "Add Ak Listener to Main Camera"Audio Listener

- Play in Background Wwise Profiler
Wwise

1. WAVWindows Explorer1Project ExplorerAudio Sound1
2. EventSound New Event
3. SoundBankSoundBanksF7New
4. EventSoundBankEventProject ExplorerSoundBank
5. Generate SoundBanks

Unity

1. **Wwise Picker** (Window > Wwise Picker) Event SoundBank
2. SoundBank
3. Event
4.
UnityWwiseUnity

- AkAmbient
- AkBank SoundBank
- AkEnvironment Collider

-- Wwise API2

- WWISE
- AK::SOUNDENGINE
To avoid packaging the SoundBanks for all platforms with your game, you should enable the automatic generation of SoundBanks or use a `BuildPlayerPipeline` script.

Build your Unity Game for a Target Platform.
Wwise Unity Integration » Unity
WwiseUnitySoundBank
:

- Wwise.
- SoundBank
- Unity
- SoundBank StreamingAssetsEditor
- Editor

**Note:** Unity

- Wwise Compressor
- Wwise Delay
- Wwise Expander
- Wwise Gain
- Wwise Matrix Reverb
- Wwise Meter
- Wwise Parametric EQ
- Wwise Peak Limiter
- Wwise RoomVerb
- Wwise Silence
- Wwise Sine
- Wwise Tone Generator

WwiseUnity Creating New Plug-ins

Wwise (McDSP, iZotope, Auro, SoundSeed, Crankcase REV, Convolution Reverb, Motion)
Wwise Integration Package

Warning:

- UnityUnity
- UnityUnity
- WwiseUnity(Warning)
- C++ AkSoundEngineC++ Wwise SDK C++

**Warning:** Wwise IntegrationUnity

IntegrationWwise LauncherWwise Integration

**Warning:** Wwise IntegrationWwise

2WwiseUnity
1. UnityWwise IntegrationWwise
2. Wwise Project
3. Wwise 'Yes'
4. Wwise SoundBankUnityWwiseOnce completed, make sure to regenerate your Wwise SoundBanks to be ready to carry on with your work after updating the Unity project to the new Wwise version.
UnityWwise

UnityWwise Wwise Launcher
Wwise Unity Integration
Wwise Unity Integration

- **AkAmbient** Use this component to attach a Wwise Event to any object in a scene. The sound can be started at various moments, dependent on the selected Unity trigger. This component is more useful for ambient sounds (sounds related to scene-bound objects) but could also be used for other purposes. Since AkAmbient has AkEvent as its base class, it features the play/stop, play multiple, stop multiple and stop all buttons for previewing the associated Wwise event. AkAmbientInspector

- **AkAudioListener**
  Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. isDefaultListener determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their AkGameObjListenerList's.

- **AkBank**
  Loads and unloads a SoundBank at a specified moment. Vorbis sounds can be decompressed at a specified moment using the decode compressed data option. In that case, the SoundBank will be prepared.

- **AkEmitterObstructionOcclusion**
  Obstructs/Occludes the emitter of the current game object from its listeners if at least one object is between them.

- **AkEnvironment**
  Use this component to define a reverb zone. This needs to be added to a collider object to work properly. AkEnvironment AkEnvironmentPortalInspector (Reverb Zones)

- **AkEnvironmentPortal**
  Use this component to define an area that straddles two different AkEnvironments zones and allow mixing between both zones. AkEnvironmentAkEnvironmentPortalInspector (Reverb Zones)

- **AkEvent**
  Wwise EventUnityHelper
AkGameObj
This component represents a sound object in your scene tracking its position and other game syncs such as Switches, RTPC and environment values. You can add this to any object that will emit sound, and it will be added to any object that an AkAudioListener is attached to. Note that if it is not present, Wwise will add it automatically, with the default values, to any Unity Game Object that is passed to Wwise.

AkRoom
An AkRoom is an enclosed environment that can only communicate to the outside/other rooms with AkRoomPortals.

AkRoomPortal
An AkRoomPortal can connect two AkRoom components together.

AkRoomPortalObstruction
Obstructs/Occludes the spatial audio portal of the current game object from the spatial audio listener if at least one object is between them.

AkSpatialAudioEmitter
Add this script on the GameObject which represents an emitter that uses the Spatial Audio API.

AkSpatialAudioListener
Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. isDefaultListener determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their AkGameObjListenerList's.

AkState
This will call AkSoundEngine.SetState() whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it.

AkSurfaceReflector
This component will convert the triangles of the GameObject's geometry into sound reflective surfaces.

AkSwitch
This will call AkSoundEngine.SetSwitch() whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it.
Wwise Picker

- **AK.Wwise.AuxBus**
  This type represents an auxiliary send in the Master-Mixer Hierarchy.
- **AK.Wwise.Bank**
  This type can be used to load/unload SoundBanks.
- **AK.Wwise.CallbackFlags**
  This type represents the values of the flags used when posting an Event with a callback.
- **AK.Wwise.Event**
  This type can be used to post Events to the sound engine.
- **AK.Wwise.RTPC**
  This type can be used to set game parameter values to the sound engine.
- **AK.Wwise.State**
  This type can be used to set Wwise States.
- **AK.Wwise.Switch**
  This type can be used to set Switch values on gameobjects.
- **AK.Wwise.Trigger**
  This type can be used to post triggers to the sound engine.

Wwise Types
Wwise

3

- **Wwise Picker** Event Wwise Picker Unity Viewer Inspector Ak Ambient Game Object
- **Add Component** Ak Ambient Ak Event Unity Game Object
- **Wwise Types** `AK.Wwise.Event.Post()` C#
- `AKSoundEngine::SetGameObjectAuxSendValues()` C#
AkAmbientInspector

- **AkAmbient:**
  - **Trigger On:**
    Provides a list of Unity events that can trigger your event.
    AkSoundEngine.PostEvent

**Adding New Triggers for Wwise Events**

- **Event Name:**
  Specifies the name of the current event.
  *Ok* Wwise Picker

- **Action On Event:**
  Enables users to override some event parameters defined in Wwise directly from Unity.
  - **Action On Event Type:**
    Overrides the event type.
  - **Curve Interpolation:**
    Overrides the interpolation curve.
  - **Fade Time:**
    Overrides the sound's fade time.

- **Use Callback:**
  Provides an easy way to make a game object react to an event callback.
  - **Game Object:**
    The game object that will receive the callback.

  - **Callback Function:**
    This is the function that will get called by *Game Object* when the callback happens.
    *Function*
    *For this to work, Game Object must define Callback Function in one of its components.*
    *FunctionName(AkEventCallbackMsg in_info).*

- **Callback Flags:**
  Select a flag which specifies when *Callback Function* will be called.

- **Play / Stop:**
Can be used to preview the Wwise Event when in Edit mode.

- **Stop All:**
  Stops all currently playing Wwise events.

- **Position Type:**
  Defines the way the event's position will be sent to the audio engine.
  - **Simple_Mode:**
    The event's position will be the same as the game object to which it's attached.
  - **Large_Mode:**
    The event can have multiple positions that are defined by a set of points.

- **MultiPosition_Mode:**
  This mode enables us to have only one instance of a sound for all instances of **AkAmbient** using the same event **in order to save memory**.
  All **AkAmbient** instances that are using this mode and that have the same event will automatically get detected and the same sound instance will be used for all of them instead of loading the same sound multiple times.
  Note that all **AkAmbient** instances in this mode and with the same event will have the same trigger (see **Trigger On** in **AkEvent**).

- **Show Attenuation Sphere:**
  Shows a sphere that defines the space where the sound played by an event can be heard.
  **For this to work you need to enable Max Attenuation in the SoundBank settings in your Wwise project (Project-Project Settings-Soundbanks->Max attenuation)**
  - **Dont_Show:**
    No attenuation sphere is shown.
  - **Current_Event_Only:**
    Shows the attenuation spheres for all the sounds that would be played after a call to **AkSoundEngine.PostEvent** while in the current mode.
    If in **Simple_Mode**, then only the attenuation sphere of the sound coming from the game object is shown.
    If in **Large_Mode**, then an attenuation sphere is shown
for each point.
If in *MultiPosition_Mode*, then an attenuation sphere is shown for every other *AkAmbient* in *MultiPosition_Mode* with the same event.

- *All_Events*:
  Shows the attenuation sphere of all *AkAmbient* instances in the scene.
Using Wwise with Unity Timeline

For Unity's Timeline feature, there are custom Wwise tracks for triggering Wwise events and setting Wwise RTPC values.

- Wwise Timeline Integration
**AkEnvironment (Reverb Zones)**

**Wwise Reverb Zone**  **Environment**  **Auxiliary Sends**  **Wwise**

AkEnvironmentColliderTo add an AkEnvironment to your scene:

- **Using the Wwise Picker.** This is the simplest way to add an AkEnvironment. AuxBusWwise PickerUnity ViewerInspector

  AkAmbientGame Object

- **"Add Component"** AkEnvironmentUnity Game ObjectInspector

- **Using scripts.** You can call

  AkSoundEngine.SetGameObjectAuxSendValues() at any time from a C# script.

2. This is useful if a game object is standing between two rooms or in a tunnel connecting two environments.

- To add an environment portal to your project, go to GameObject- >Wwise->Environment Portal in Unity’s menu bar.

AkGameObj

AkEnvironmentPortal objects will automatically detect AkEnvironment objects that overlap it. Inspector2

Wwise

Those 4 environments are selected as follows:

- The environments that are connected to a portal and that have the highest priority are selected until we reach 4 environments or until there are no more environments connected to a portal.

- If we still don’t have 4 selected environments, we select the environments that are not connected to a portal as follows:

  - Environments with the highest priority will be selected until we reach 4 environments (if the Default and Exclude Others flags
• **AkEnvironment** component:
  Only 4 environments can be active at the same time.
  ◦ **Priority:**
    Defines the priority of an environment.
    A smaller number has a higher priority.
    If a game object is inside more than 4 environments, only the 4 environments with the highest priority will be active (if the Default and Exclude Others flags are not set).
  ◦ **Default:**
    A default environment will be active only if it's the only environment containing your game object.
    If your game object is inside more than one default environment, then only the one with the highest priority will be active.
  ◦ **Exclude Others:**
    An environment with this flag can't be overlapped by other environments.
    If your game object is inside an environment with the Exclude Others flag, then all other environments will get discarded.
    If your game object is inside more than one environment with the Exclude Others flag, only the one with the highest priority will be active.
  ◦ **AuxBus Name:**
    Specifies the name of the current AuxBus. AuxBusAuxBus AuxBusAuxBus AuxBus Wwise PickerAuxBus

• **AkEnvironmentPortal** component:
  You can create an environment portal in Unity by going to GameObject->Wwise->Environment Portal.
  You can place an environment portal between two environments to combine their effects while your game object is inside the portal.

  The closer the game object is from an environment, the more that environment will contribute towards the final effect.
  ◦ **Environment #1:**
The portal will automatically detect all environments that intersect the portal.

- **Environment #2:**
  The portal will automatically detect all environments that intersect the portal.

- **Axis:**
  The axis is used to find the contribution of each environment. For example, if the z axis is chosen, then moving along the x axis won't have any effect on the contribution of each environment. 

  **Note that the axis is in object space. So, rotating the portal will also rotate the axis.**

- Integrating Environments and Game-defined Auxiliary Sends
- AK::SoundEngine::SetGameObjectAuxSendValues
Using C# code to control the sound engine

Most Wwise SDK functions are available in Unity through the `AkSoundEngine` class. C++

SDK API

GameObjectID GameObjectUnityAkGameObj

GameObject

`EventBankID`

Wwise APIIDWwiseC#

Wwise_IDS.h

Wwise_IDS.cs Assets > Wwise > Convert Wwise SoundBank

IDsPython

Sending MIDI to Wwise.

MIDI can be sent to Wwise by filling the `AkMIDIPost` members of `AkMIDIPostArray` class and calling any of the following methods:

- `AkMIDIPostArray.PostOnEvent()`
- `AkSoundEngine.PostMIDIOnEvent()`
- `AK.Wwise.Event.PostMIDI()`

The following is a basic script that sends MIDI messages to the sound engine:

```csharp
public class MyMIDIBehaviour : UnityEngine.MonoBehaviour
{

    public AK.Wwise.Event SynthEvent;

    private void Start()
    {
        AkMIDIPostArray MIDIPostArrayBuffer = new AkMIDIPostArray(6);
        AkMIDIPost midiEvent = new AkMIDIPost();
    }
}
Using the Audio Input Source Plug-in in Unity.

The audio input source plug-in can be used via C# scripting. See Audio Input Source Plug-in from the Wwise SDK documentation.

The following is a basic script that sends a test tone to the audio input
public class MyAudioInputBehaviour : UnityEngine.MonoBehaviour
{

    public AK.Wwise.Event AudioInputEvent;
    public uint SampleRate = 48000;
    public uint NumberofChannels = 1;
    public uint SampleIndex = 0;
    public uint Frequency = 880;
    private bool IsPlaying = true;

    bool AudioSamplesDelegate(uint playingID, uint channelIndex, float[] samples)
    {
        for (uint i = 0; i < samples.Length; ++i)
        {
            samples[i] = UnityEngine.Mathf.Sin(Frequency * 2 * UnityEngine.Mathf.PI * (i + SampleIndex) / SampleRate);
        }

        if (channelIndex == NumberofChannels - 1)
        {
            SampleIndex = (uint)(SampleIndex + samples.Length) % SampleRate;
        }

        // Return false to indicate that there is no more data to provide. This will also stop the associated event.
        return IsPlaying;
    }

    void AudioFormatDelegate(uint playingID, AkAudioFormat audioFormat)
    {
        audioFormat.channelConfig.uNumChannels = NumberofChannels;
        audioFormat.uSampleRate = SampleRate;
    }
}
private void Start()
{
    AkAudioInputManager.PostAudioInputEvent(AudioInputEvent, gameObject, AudioSamplesDelegate, AudioFormatDelegate);
}

// This method can be called by other scripts to stop the callback
public void StopSound()
{
    IsPlaying = false;
}

private void OnDestroy()
{
    AudioInputEvent.Stop(gameObject);
}

Apply Custom Positioning in Unity

By default, the AkGameObj component is attached to a specific Unity gameObject and uses its transform (with an optional offset) for full positioning. This is usually adequate for many games, such as first-person shooters. However, games with custom camera angles, such as many third-person games, may find it difficult to accommodate the two aspects of positioning (distance attenuation and spatialization) by simply attaching the audio listener to one game object, such as the main camera in Unity. Other games may want players to experience other custom positioning.

To this end, the AkGameObj component class provides overridable positioning to Unity users. Through the three virtual methods GetPosition(), GetForward(), and GetUpward(), users can derive a subclass from AkGameObj and use that subclass component to customize any number of Unity gameObjects' positioning.
Here is a simple example of how to use a custom component to override the default `AkAudioListener` behavior. With a third-person project integrated with Wwise, remove the existing `AkAudioListener` and its associated `AkGameObj`. Then attach the following script to the MainCamera object, attach `AkAudioListener`, and finally specify the target Unity `gameObject` (such as the player avatar) that the audio listener's position will follow. After this, the distance attenuation of all the emitters will rely on the selected target Unity `gameObject`'s position as the listener position (an on-screen distance listener), while the orientation of all the emitters is still based on the main camera orientation as the listener orientation (an off-screen orientation listener).

```csharp
#if !(UNITY_DASHBOARD_WIDGET || UNITY_WEBPLAYER || UNITY_WII || UNITY_WIIU || UNITY_NACL || UNITY_FLASH || UNITY_BLACKBERRY) // Disable under unsupported platforms.

// Copyright (c) 2017 Audiokinetic Inc. / All Rights Reserved

using UnityEngine;
using System;
using System.Collections.Generic;

[assembly: AddComponentMenu("Wwise/AkGameObj3rdPersonCam")]
[assembly: ExecuteInEditMode] // Necessary to maintain proper state of isStaticObject.

public class AkGameObj3rdPersonCam : AkGameObj
{
    public Transform target; // The position that this camera will be following. User can specify this to the player character's Unity gameObject in the Inspector.
}
// Sets the camera position to the player's position to handle distance attenuation.
public override Vector3 GetPosition ()
{
    return target.GetComponent<AkGameObj> ().GetPosition ();
}

#endif // #if !(UNITY_DASHBOARD_WIDGET || UNITY_WEBPLAYER || UNITY_WII || UNITY_WIIU || UNITY_NACL || UNITY_FLASH || UNITY_BLACKBERRY) // Disable under unsupported platforms.
Wwise Unity Integration » Wwise Unity Integration
SoundBank

SoundBank: Generate SoundBanks
Picker

WwiseUnity: SoundBanks generation successful
WwiseUnity: SoundBanks generation has warning(s)
WwiseUnity: SoundBanks generation error
Console

Note: Wwise Settings
Wwise Windows Installation Path
Mac
Wwise Application
Unity: Generate SoundBanks
Wwise Picker

Wwise PickerGame ObjectInspector:

- Event: AkAmbient
- SoundBank: AkBank
- Switch Value: AkSwitch
- State Value: AkState
- Aux Bus: AkEnvironment
Wwise Types

Wwise Types:

```csharp
public class WwiseTypesExample : UnityEngine.MonoBehaviour
{
    public AK.Wwise.Bank MyBank = null;
    public AK.Wwise.Event MyEvent = null;
    public AK.Wwise.RTPC MyRTPC = null;

    public void Awake()
    {
        MyBank.Load();
    }

    public void Start()
    {
        MyEvent.Post(gameObject);
    }

    private float CalculateMyValue()
    {
        return (float)System.Math.Sin(System.Math.PI * UnityEngine.Time.timeSinceLevelLoad);
    }

    public void Update()
    {
        MyRTPC.SetValue(gameObject, CalculateMyValue());
    }
}
```
public AK.Wwise.CallbackFlags MyCallbackFlags = null;

public void Start()
{
    MyEvent.Post(gameObject, MyCallbackFlags, EventCallback);
}

private void EventCallback(object cookie, AkCallbackType type, AkCallbackInfo info)
{
    if (type == AkCallbackType.AK_Marker)
    {
        var markerInfo = info as AkMarkerCallbackInfo;
        if (markerInfo != null)
        {
            // ...
        }
    }
}
**Adding New Triggers for Wwise Events**

In most of Wwise components for Unity, there is a "Trigger On" property from which you can select which Unity notification/event will trigger the Wwise component (Event, Switch, State, etc). Unity Integration

```
AkTriggerBase C#AkTriggerBase "Trigger On"
triggerDelegate(GameObject in_target) "target"
Collider WwiseColliderPost

GetComponent<YourTriggerClass>() .triggerDelegate(GameObject in_target)
```

**Note:** Currently the maximum number of derivative classes of **AkTriggerBase** is 32.

Here is an example, with a custom function:

```csharp
public class TriggerOnGunHit : AkTriggerBase
{
    void Hit()
    {
        if (triggerDelegate != null)
        {
            triggerDelegate(null);
        }
    }
}
```

From your game code, you could have this code:

```csharp
if (playerIsShot)
{
    GetComponent<TriggerOnGunHit>().Hit();
}
```
TriggerOnGunHitWwiseNote that in many simple situations, this is completely superfluous; you could also simply call the base Wwise SDK through `AkSoundEngine.PostEvent("GunHit", gameObject)` and let the sound designer handle the effect of this event in Wwise.
Wwise Timeline Integration

Timeline is Unity's cinematic editing tool. It can be used to create gameplay sequences, audio sequences, and other cinematic content.

For more information on Timeline refer to Timeline Docs.

For instructions on how to create a Timeline Asset and Timeline instance, refer to Timeline Usage.
**Wwise Timeline Tracks**

Wwise Unity Integration  
AkRTPC  
AkEvent 2  
AkRTPC  
TrackGame Parameter  
AkEvent  

Tracks and **AkEvent** tracks can be added to a Timeline in Unity using the "Add" dropdown button.

**Adding AK tracks to Timeline**

**Ak Track Object Bindings**

**AkEvent** tracks and **AkRTPC** tracks have object bindings which can be set to any Game Object. If this is left as None, the Event messages or RTPC messages in the track will be triggered on the Game Object that owns the Timeline (this is the Game Object that has a PlayableDirector component, which is used to trigger the Timeline). If the track binding is set to a different Game Object, the messages will be triggered on that Game Object.

**The AkRTPC Parameter Property**

**AkRTPC** tracks have one property in their property inspector: **Parameter**. This is the Wwise RTPC parameter that will be set from the various clips in the track.
RTPC Track Parameter Property

**Wwise Timeline Playable Clips**

The *AkEvent* track and the *AkRTPC* tracks each hold corresponding clips: the *AkEventPlayable* clip and the *AkRTPCPlayable* clip. These can be added to a track using the track's menu.

**Adding AK clips to AK tracks**

**Timeline Clip Properties**

All clips in Timeline have timing properties that are displayed in the top portion of the inspector. For more information on these properties, refer to Clip Properties.

**AkRTPCPlayable Clip Properties**

The following properties are accessible from the inspector of an *AkRTPCPlayable* clip:
AkRTPCPlayable Clip Properties

- **Set RTPC Globally**: If this is checked, the RTPC will be set globally for all objects.
- **Override Track Object**: If this is checked, the RTPC will be set for the selected Game Object in the RTPC Object property (only available when Set RTPC Globally is unchecked).
- **RTPC Object**: The Game Object for which the RTPC will be updated. If this is set to None, the RTPC will be set on the track object (or the Timeline's owner object if the track object is set to None). This is only available when Override Track Object is checked.
- **Animated Value: RTPC Value**: This is the value that will be animated by the RTPC track. In order to begin animating this value you need to arm the track for recording, set the playhead to somewhere within the RTPCPlayable clip, and then edit the value from the property inspector.
Animating RTPC Values from Timeline

Following these steps will insert a keyframe into the RTPC track, below the RTPC clip. You can then use this Unity curve editor to add more keyframes. The keyframes have a context menu that can be used for more precise values.
RTPC Keyframe Context Menu

Editing RTPC Keyframe Values

**AkEventPlayable Clip Properties**

The following properties are accessible from the inspector of an AkEventPlayable clip:

- **Override Track Object**: This can be checked in order to have the Event emitted from a specific Game Object, different to the track binding object. This will also override the Timeline owner object if the track binding is set to None (see Ak Track Object Bindings).
- **Emitter Object Ref**: The Game Object from which the Event should be emitted (only available when Override Track Object is checked).
- **Event**: The Wwise Event that will be triggered from this clip.
Currently scrubbing is only supported in Play mode in the Unity editor. This is due to an issue with Unity Playable callbacks in edit mode. Unity are aware of the issue: 953109. This means that "Play from anywhere" behavior is not supported in edit mode either. If you place the playhead somewhere between the beginning and end of a clip and play the timeline, while in edit mode, the event will not be triggered.
Wwise Project Setup

AkAudioEvent  Tracks  JSON
Project Settings  SoundBanks
Estimated duration  SoundBank Settings
JSON Metadata  Generate
Metadata Options
AkEvent
Wwise Unity Integration
Build your Unity Game for a Target Platform

Building your Unity game for a target platform is now as simple as building your application using Unity's standard pipeline or a build script. Pre- and post-build steps that generate, copy and delete SoundBanks can be enabled/disabled within the **Edit > Wwise Settings**...
Wwise plug-in deployment

The Unity build system automatically packages the Wwise plug-in with the game. **Profile** Wwise

**Plugins** Wwise **Release**

- **Note:** Debug is only used for debugging specific issues with the sound engine, which is usually done with the assistance of AudioKinetic support.

- Android
- Build for iOS
- Linux
- UnityWwise
StreamingAssets folder

Unity normally creates a `StreamingAssets` folder (manually create one if it does not exist in your Unity project) in the Unity project's `Assets` folder, as shown here: `<UNITY_PROJECT_ROOT>\Assets\StreamingAssets`. Unity

SoundBank  Audio\GeneratedSoundBanks  StreamingAssetsUnity
WwisePlatform  Manager
<UNITY_PROJECT_ROOT>\Assets\StreamingAssets\Audio\GeneratedSoundBank
<YourPlatform>

**Note:** If necessary, you can modify `AkInitializer::basePath` to change the default SoundBanks path.

Unity EditorSoundBankWwiseProject SettingsWwise GeneratedSoundBankWwise WindowsMac
Deploying SoundBanks in single-platform projects

If you are working on a single platform title, instead of physically moving SoundBank files into the StreamingAssets folder, the SoundBank output path in a Wwise project can be set to the desired folder under the StreamingAssets folder so that the SoundBanks can be generated directly into the desired location. Using this method, be sure to disable the pre-build steps related to generating and copying SoundBanks within the Wwise Settings window.
Deploying SoundBanks in multi-platform projects

For productions that ship on multiple platforms, the pre-build steps can generate the SoundBanks and copy them over to the StreamingAssets folder before the binary is built, and then delete the SoundBanks in a post-build step to ensure that only SoundBanks for the target platforms are deployed.

Handling multiple custom platforms for a reference platform

If, in your Wwise project, you have defined multiple versions of the same reference platform, such as an iPad and an iPhone platform targeting iOS, some more scripting will be needed. UnityWwise
Exclude SoundBank metadata

Wwise generates SoundBank metadata files (such as TXTs and XMLs).
Wwise Unity Integration » Build your Unity Game for a Target Platform
Android

APKAndroidSoundBankLoadBank() API StreamingAssets
SoundBankUnityAPK AssetsLow-Level IOSoundBankAPK

Low-Level IOAndroidSoundBankSDCard
AkSoundEngine.AddBasePath(YourPath) APK
APKAPKDLC
Wwise Unity Integration » Build your Unity Game for a Target Platform
Build for iOS

UnityiOS  Build and Run Unity  Editor  UnityXcode

UNITY_PROJECT_ROOT/Assets/Plugins/iOS

UnityThumbXcodeThumb
Building for release

The Wwise Integration library, libAkSoundEngine.a, normally contains all debug symbols. (iOS) Player Settings

Settings  Stripping Level 12MB
**Linux**

Unity/Linux/Unity Editor Build/Build/RunUnity

Linux .x8632 bit.x86_64 64 bit

Wwise Unity Integration for Linux needs `libSDL2` installed on the machine to work. "DLLNotFoundException"

*libSDL2 Ubuntu 12.04:*

- `sudo apt-get install build-essential libasound2-dev`
- `wget http://www.libsdl.org/release/SDL2-2.0.3.tar.gz`
- `tar -zxvf SDL2-2.0.3.tar.gz`
- `cd SDL2-2.0.3`
- `./configure`
- `make`
- `sudo make install`
- `sudo ldconfig`

Requirements may vary depending on the machine's configuration. [libsdl.org](http://www.libsdl.org)
Wwise Unity Integration » Build your Unity Game for a Target Platform
UnityWwise

Wwise Unity Integration Wwise

C#

Note: SoundBank

SoundBank

GetPlatformName
GetPlatformName
1. Wwise Platform Manager: "iPod", "iPhone", "iPad"
   Setting Up Your Projects > Managing Platforms

2. Unity

```csharp
public partial class AkBasePathGetter
{
#if UNITY_IOS
    static partial void GetCustomPlatformName(ref string platformName)
    {
        switch(UnityEngine.iOS.Device.generation)
        {
            case UnityEngine.iOS.DeviceGeneration.iPodTouch1Gen:
            case UnityEngine.iOS.DeviceGeneration.iPodTouch2Gen:
            case UnityEngine.iOS.DeviceGeneration.iPodTouch3Gen:
            case UnityEngine.iOS.DeviceGeneration.iPodTouch4Gen:
            case UnityEngine.iOS.DeviceGeneration.iPodTouch5Gen:
            case UnityEngine.iOS.DeviceGeneration.iPodTouchUnknown:
                platformName = "iPod";
                break;

            case UnityEngine.iOS.DeviceGeneration.iPad1Gen:
            ```
n.iPad2Gen: case UnityEngine.iOS.DeviceGeneration.iPad2Gen:

n.iPad3Gen: case UnityEngine.iOS.DeviceGeneration.iPad3Gen:

n.iPadMini1Gen: case UnityEngine.iOS.DeviceGeneration.iPadMini1Gen:

n.iPad4Gen: case UnityEngine.iOS.DeviceGeneration.iPad4Gen:

n.iPadAir1: case UnityEngine.iOS.DeviceGeneration.iPadAir1:

n.iPadMini2Gen: case UnityEngine.iOS.DeviceGeneration.iPadMini2Gen:

n.iPadMini3Gen: case UnityEngine.iOS.DeviceGeneration.iPadMini3Gen:

n.iPadAir2: case UnityEngine.iOS.DeviceGeneration.iPadAir2:

n.iPadUnknown: platformName = "iPad";
break;

case UnityEngine.iOS.DeviceGeneration.iPhone:

case UnityEngine.iOS.DeviceGeneration.iPhone3G:

case UnityEngine.iOS.DeviceGeneration.iPhone3GS:

case UnityEngine.iOS.DeviceGeneration.iPhone4:

case UnityEngine.iOS.DeviceGeneration.iPhone4S:

case UnityEngine.iOS.DeviceGeneration.iPhone5:

case UnityEngine.iOS.DeviceGeneration.iPhone5C:

case UnityEngine.iOS.DeviceGeneration.iPhone5S:
n.iPhone6:  
  \n  case UnityEngine.iOS.DeviceGeneration.iPhone6:
  \n  n.iPhone6Plus:  
  \n  case UnityEngine.iOS.DeviceGeneration.iPhone6Plus:
  \n  n.iPhoneUnknown:  
  default:
    platformName = "iPhone";
    break;
  
} 

Note: platformName Unity

3. Either, (a) create a C# that extends AkBuildPreprocessor or (b) create a C# script that uses the functionality within AkBuildPreprocessor. The contents of the file could be as follows:

```csharp
public class WwiseIOSBuildPreprocessor : IPreprocessBuild, IPostprocessBuild
{
    public int callbackOrder { get { return 0; } }

    string iPodDestinationSoundBankFolder = string.Empty;
    string iPadDestinationSoundBankFolder = string.Empty;
    string iPhoneDestinationSoundBankFolder = string.Empty;

    public void OnPreprocessBuild(BuildTarget target, string path)
    {
        if (target == BuildTarget.iOS)
        {
            AkBuildPreprocessor.CopySoundbanks(true, "iPod", iPodDestinationSoundBankFolder);
```
AkBuildPreprocessor.CopySoundbanks(true, "iPad", iPadDestinationSoundBankFolder);
AkBuildPreprocessor.CopySoundbanks(true, "iPhone", iPhoneDestinationSoundBankFolder);
}
}

public void OnPostprocessBuild(BuildTarget target, string path)
{
    DeleteSoundbanks(iPodDestinationSoundBankFolder);
    DeleteSoundbanks(iPadDestinationSoundBankFolder);
    DeleteSoundbanks(iPhoneDestinationSoundBankFolder);
}

4. Or in Wwise, generate the SoundBanks for all three platforms "iPhone", "iPod", and "iPad", and copy the three resulting folders to UNITY_PROJECT_ROOT/Assets/StreamingAssets/Audio/GeneratedSoundBanks
5. UnityiOS
6. SoundBank
**Wwise**

DLCWwiseFile PackageAuthoring: Wwise > Help > Finishing Your Project > Managing File Packages > Downloadable Content OverviewBNKWEMDLC Wwise/UnityLow-Level IO

UnityAkInitializerBase Path
AkSoundEngine.LoadFilePackage() Wwise

Base PathiOSAndroid

AndroidiOS
AkSoundEngine.LoadFilePackage() AkSoundEngine.AddBasePath()

Operating System:

```csharp
#if UNITY_IPHONE
    string fileNameBase = Application.dataPath.Substring(0, Application.dataPath.LastIndexOf('/'));
    string fileName = fileNameBase.Substring(0, fileNameBase.LastIndexOf('/') ) + '/Documents/' + FILE_NAME;
#elif UNITY_ANDROID
    string fileName = Application.persistentDataPath + '/' + FILE_NAME; 
#else
    string fileName = Application.dataPath + '/' + FILE_NAME;
#endif

Androidsdcard:

- **OBB (Android)**
Build the Native Integration Plug-in from Source

The Wwise-Unity integration is a thin layer of C# calling the native Wwise API.
The procedure

1. Install the Wwise SDK for the platforms you wish to build using the Wwise Launcher.
2. WwiseUnityOSWindowsMacsourcezip
   LauncherUnityzipzip_Src.zip
3. WindowsMaczip
4. Integration
5. zip

6. 2: **Build using console utility**  **Integrated Development Environment (IDE)**

7. Deployment API
Notes

- During the build, you can monitor warnings, errors, or critical messages, if any, by reading the detailed log messages found in the Build Log.
## Integration:

<table>
<thead>
<tr>
<th>All platforms</th>
<th>Unity 5 PersonalPro</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Wwise SDK</td>
</tr>
<tr>
<td></td>
<td>SDK Wwise SDK</td>
</tr>
<tr>
<td></td>
<td>&gt; Platform Requirements</td>
</tr>
<tr>
<td></td>
<td>Python 2.7.x3.xPython</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Android</th>
<th>Cygwin (Windows)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Android SDK: 32bitAPI964bitAPI21</td>
</tr>
<tr>
<td></td>
<td>Android NDK r10e.</td>
</tr>
<tr>
<td></td>
<td>Apache Ant 1.8.4.</td>
</tr>
<tr>
<td></td>
<td>Wwise SDK</td>
</tr>
<tr>
<td></td>
<td>Unity</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CYGWIN_HOME CygwinWindows</td>
</tr>
<tr>
<td></td>
<td>ANDROID_HOME Android SDK</td>
</tr>
<tr>
<td></td>
<td>ANDROID_NDK_ROOT Android NDK</td>
</tr>
<tr>
<td></td>
<td>ANT_HOME Apache Ant</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>iOS</th>
<th>Xcode 7.2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>iOSWwise SDK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Linux</th>
<th>Build-essential (sudo apt-get install build-essential)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SDL2 (SDL2)</td>
</tr>
<tr>
<td></td>
<td>LinuxWwise SDK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mac</th>
<th>Xcode 7.2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Wwise SDK for Mac, and a WWISESDK environment variable pointing to it</td>
</tr>
<tr>
<td>Platform</td>
<td>Visual Studio Edition</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>Windows Store</td>
<td>Visual Studio 2015.</td>
</tr>
</tbody>
</table>
How to install the source package

Unzip the source package. Wwise SDK IntegrationWwiseUnityAssets IntegrationAssets
- **StreamingAssets**: SoundBank
- **Deploying SoundBanks in single-platform projects**
- **Wwise**:
  - **Deployment**: Integration
    - **API**: C++  C#  Wwise SDK
    - **Dependencies**: Unity
    - **Components**: Unity
    - **Plug-ins**: Unity
      - **Platform**
        - **Architecture**: Files shared by architectures of a multi-architecture platform.
        - **Debug**: Debug
          - **Wwise plug-in deployment** for detail
        - **Profile**: Profile
          - **Wwise plug-in deployment** for detail
        - **Release**: Release
          - **Wwise plug-in deployment** for detail
        - **DSP**: Wwise
  - **Documentaion**: Integration
  - **Editor**: WwiseUnityIntegrationEditor  WindowInspector
  - **Tools**:
- **Wwise:**
  - **AkSoundEngine**: IntegrationIDC
    - **Common**:
    - **Platform**: IDE
  - **Integration/Assets/Wwise/Deployment**: Integration
    - **API**: API
      - **Generated**: SWIGAPI
      - **Handwritten**: API
    - **Components**: Unity
    - **Plugins**: WwiseAkSoundEngine
      - **<platform>**:
        - **<architecture>**:
Build using console utility

You can build the Integration from the command line using the build script located at Wwise\AkSoundEngine\Common\BuildWwiseUnityIntegration.py. To see the usage and examples, assuming your current working directory is the parent folder of the script, type in a command console on your computer:

```
python BuildWwiseUnityIntegration.py -h
```
Integrated Development Environment (IDE)

Located in the Wwise Unity Integration source package under:

```
WwiseUnityIntegration_version_platform_Src.zip\Wwise\AkSoundEngine\YourPlatform
```

due to the solution (or Xcode project) allows you to build the Integration for the target platform in a supported IDE.

**Building for the Mac or iOS platform with Xcode**

When building the integration from the command line, the WWISESDK path is provided to Xcode automatically based on the one set in the environment variable $WWISESDK or the one provided to the build script using the -w option. Xcode IDEWWISESDKXXcodeMacIOS AkSoundEngine{platform} Build Settings User-Defined WWISESDK WWISESDK (ex: /Users/myUser/Wwise/SDK )

**Building for the Linux platform**

A premake script is included in the integration source package that generated the necessary makefiles. To build the Linux plug-in from source, simply enter the following commands into a terminal:

- cd <Integration source location>/AkSoundEngine/Common
- ./premake4 --akplatform=Linux gmake
- make -f AkSoundEngineLinux.make config=<config>

<config> debug32, profile32, release32, debug64, profile64, release64 <Integration source location>/Deployment/Plugins/Linux
Deploy the new lib

The resulting binaries will be found in Wwise\Deployment\Plugins\[Platform]. Assets\WwiseAssets
Under the hood

SWIGWwise SDKAPIUnity
WwiseUnitySWIGAPIWwise for Unity
SWIGC++

The build process performs the following task:

1. The Wwise SDK libraries are wrapped into a single library as the Unity plug-in, a dynamic library (.dll) on Windows, a loadable bundle (.bundle) on Mac OS X, a static library (.a) on iOS, or a shared library (.so) on Android. Deployment

: 

- Build Log
- C++Wwise Integration Package
Wwise Unity Integration » Build the Native Integration Plug-in from Source
**Build Log**

Warning Unity Editor Build Log Build Log Integration Build Log

UNITY_PROJECT_ROOT\Assets\Wwise\Logs\BuildWwiseUnityIntegration.log

<table>
<thead>
<tr>
<th>Time</th>
<th>Message type</th>
<th>Source Code</th>
<th>File name (Module name)</th>
<th>Line number</th>
<th>Message body</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013-09-26 09:29:56,490</td>
<td>INFO</td>
<td>BuildWwiseUnityIntegration.py (WindowsBuilder)</td>
<td>91</td>
<td>Building: Windows (Win32, Debug) ...</td>
<td></td>
</tr>
</tbody>
</table>

1Unity Integration
BuildUtil.CreateLogger()

WwiseUnityIntegration_version_platform_Src.zip\Wwise\AkSoundEngine\Common\AkSoundEngine\Common\BuildUtil.py

Python logging.handlers module

IDE Visual Studio

IDE
C++Wwise Integration Package

Warning:

- Unity
- Unity
- Wwise(Warning)

The upgrade is done in a 3-step process. Wwise Unity
UnityWwise

Wwise Launcher UnityWwise
Updating your Wwise project to the new version

1. Launch the Wwise version that matches the new Wwise Integration version you intend on using in Unity.
2. Wwise Project
3. Wwise 'Yes'
4. Wwise SoundBankUnityWwiseOnce completed, make sure to regenerate your Wwise SoundBanks to be ready to carry on with your work after updating the Unity project to the new Wwise version.
5. SoundBankUnityStreamingAssets
Updating your C++ code modifications

1. Install the updated Wwise SDK on your machine, for all your desired platforms.
2. Unity Unity Integration Source Code zip
3. Extract all the downloaded Unity Integration Source Code packages to the same directory
4. Merge the modifications you have made to the Unity Integration Source Code into the updated location
5. Compile the updated and merged integration source code
API

Wwise SDKIntegration:

- UnloadBank() 2
- iOSAPI
  AK::SoundEngine::iOS::ListenToAudioSessionInterruption()
- GetGameObjectFromPlayingID() Windows GameObject ID
- 32bit Windows 32bit 64bit
- PostEvent() GameObjectAPI GameObject null
- PostEvent()
- AK::Monitoring::SetLocalOutput()
  AkCallbackManager.SetMonitoringCallback()
- AK::SoundEngine::SetPosition()
  AkSoundEngine.SetObjectPosition()
- PostEvent() External Sources

- AK::SoundEngine::DynamicSequence API Unity API
  SWIG
  AkSoundEngine.Open() API
  AkSoundEngine.DynamicSequenceOpen()
- C++ Unity
  AkArray::operator[]
  AkPlaylistArray.ItemAtIndex(uint uiIndex) SWIG C++

- API:
  - iOSAPI
    - AkSpeakerVolumeMatrixCallbackInfo
    - AkSpeakerVolumeMatrixBusCallbackInfo
    - AkBusCallbackFunc
    - AK::SoundEngine::RegisterBusVolumeCallback
    - AK::SoundEngine::RegisterCodec
    - AK::SoundEngine::RegisterGlobalCallback
    - AK::SoundEngine::RegisterPlugin
    - AK::SoundEngine::Query::AkGameObjectsList
    - AK::SoundEngine::Query::GetActiveGameObjects()
    - AK::SoundEngine::Query::GameObjDst
    - AK::SoundEngine::Query::AkRadiusList
    - AK::SoundEngine::Query::GetMaxRadius(AkRadiusList&
io_RadiusList)

- Event
  - AK::SoundEngine::DynamicDialogue::ResolveDialogueEvent

---

Wwise Unity Integration Mon Jan 8 10:46:17 2018  doxygen 1.6.3
Android

- Android
- Background Mode (Android)
- OBB (Android)
- WwiseUnityDLC
iOS

- Build for iOS
- Audio Session Interruptions (iOS)
- WwiseUnityDLC
Linux

- Linux
Windows Store Apps

- Windows Store Apps Specific Information
Background Mode (Android)

AK::SoundEngine::Suspend
AkSoundEngine.WakeupFromSuspend() AkInitializer
OBB (Android)

Overview of the OBB and Wwise IO

AndroidUnity Player Settings > Publishing Settings > Split Application Binary, APK Expansion Files.obbZip
SoundBankStreamingAssetsAndroid SoundBank
SoundBankAPKOBBI/OOBBpersistent CPU
Loading banks in memory

AkMemBankLoader.cs

1. SoundBank.
2. InspectorSoundBank
   Bank Name
3. SoundBankInspector
   Is Localized Bank
   AkInitializer.cs
   SoundBank

2. AkMemBankLoader.LoadNonLocalizedBank()
   AkMemBankLoader.LoadLocalizedBank()
- SoundBankIOZip
- SoundBankIntegration
- streamingmanager_lowlevel
- 11SoundBankSoundBank
- SoundBankAPIAPIIntegration
Wwise Unity Integration »
**Audio Session Interruptions (iOS)**

The Unity Integration supports two audio session categories.

- **AkAudioSessionCategorySoloAmbient** *(Default)*
  SilentiPhoneRing/Silent

- **AkAudioSessionCategoryAmbient** **AmbientSound**

  BGM (BackGround Music)

  UI

  `AkCallbackManager.SetBGMCallback()`

SoloAmbientBGM AmbientBGM

**Note:** The background and foreground switching is handled internally on iOS, no need to call Suspend and WakeupFromSuspend manually on this platform.
Windows Store Apps Specific Information

- Universal Windows Platform Windows SoundBanks2SDK SoundBank
Licensing (Free & Commercial)

Wwise licenses are required for any games that are released publicly.

1Audiokinetic

- Free Limited Commercial License
- Educational & Non Commercial License
- Royalty-based License
- Commercial License
- Licensing FAQ
Wwise

Wwise Unity Integration
Wwise Launcher
Unity "Recent Unity Projects"

Unity Integration

3D""
Wwise Demo Scene Wwise Launcher
Wwise Launcher Unity Wwise

: 

- SoundBank
- Wwise Project Assets Demo <DEMO_SCENE_ROOT>/WwiseProject Wwise
Wwise Demo Scene

Wwise Unity Integration

Wwise Demo Scene:

1. Launcher
   - Unity
   - Wwise Demo Scene
   - Unity
   
   Modify Wwise in
   Project...
   - Launcher
2. Deployment Platform
   - Modify
3. Unity
4. SoundBank
5. Generated SoundBanks
   - StreamingAssets/Audio
6. Unity
Footsteps

Wwise Project RandomSwitch
Footstep

Footstep_material 4

Box Collider Footstep_MaterialSwitch Value Wwise Picker
Window Box Collider First Person Controller Collider Switch
"AkTriggerEnter" Ak Switch "Use Other Object"
Inspector Window

0.3 Footstep First Person Controller Wwise
Sound Engine
Subtitle

Unity

Delegate AkTriggerButtonPressDelegate
AkTriggerBase triggerDelegate
AkTriggerBaseWwise Component
Inspector "trigger" Ak Ambient

WAVAk Ambient Inspector "Use Callback"
GameObject (SubtitleDemo.cs) GameObject
Object "Callback Function ( Callback Flags "Marker"
MarkerCallback uIdentifier
SubtitleDemoCallback
unity_use_AkEvent_AkAmbient
Environment

StationEnvironmentZone21Box Collider AuxBusWwise Picker

Wwise2Auxiliary BusLittle Sequence"Use game-defined Auxiliary sends"

EnvironmentZoneWwiseAuxBus

Environment Portal23Environment PortalAuxiliary

Ak Environemnts2Box Collider1'z'RedBlue lAk Environment Portal2

EnvironmentEnvironment PortalAkEnvironmentAkEnvironmentPortal Inspector (Reverb Zones)
**Timeline Demo**

This station demonstrates the use of an AkEventTrack and an AkRTPCTrack within a Timeline in Unity. There are two cubes, which, when the button is pressed, are animated towards each other and stopped when they reach impact.

In order to inspect the Timeline, select Window->Timeline. Then in the Hierarchy, expand the TimelineDemo object, and then expand the Timeline Demo Button object. Then select the Button object. The Timeline editor will now show the Timeline that controls the animation and Wwise Events for the cubes. The Timeline is controlled by thePlayable Director component of the Button object.

The Timeline contains two animation tracks, one for each cube. These tracks are used to animate the z position of each cube over time. The Timeline also contains an AkEventTrack and an AkRTPCTrack, marked by a white and red tab, respectively. The AkEventTrack contains two AkEventPlayable clips: the first triggers the PlayCubeMovement Wwise Event, and the next triggers the PlayImpact Wwise Event. The names of these Wwise Events are displayed in the clips. You may need to increase the size of the Timeline editor view and zoom in in order for the name to be displayed correctly. The PlayCubeMovement Wwise Event plays a sine wave source, which has an RTPC affecting its pitch. The name of this RTPC is CubeAcceleration. The AkRTPCTrack increases this RTPC over time as the cubes move towards each other. This causes the pitch of the sine wave to increase, producing a simple sound effect to indicate the acceleration of each cube towards the other.

For more information on the Timeline integration, see **Wwise Timeline Integration**
Using UnityWwise Spatial Audio

This chapter gives an overview of how to use the Spatial Audio features of the Unity integration. It is broken down into:

- A preparation page you need to follow before going to the tutorials:
  - Preparation for the Spatial Audio Tutorials
    - 1. Create a Unity Project
    - 2. Wwise Project Preparation
    - 3. Unity Project Preparation

- Two independent tutorials:
  - Surface Reflectors Tutorial
  - Rooms and Portals Tutorial

- And a supplementary page addressing the use of obstruction and occlusion:
  - Obstruction and Occlusion

**Note:** Completion of sections using the Wwise Reflect plug-in require the appropriate license.
Preparation for the Spatial Audio Tutorials

The following steps must be followed beforehand to successfully complete the Surface Reflectors Tutorial and the Rooms and Portals Tutorial:

1. Create a Unity Project
2. Wwise Project Preparation
3. Unity Project Preparation
   - 3.A. Environment
   - 3.B. Character Listener
   - 3.C. Third-Person Emitters
   - 3.D. Scene
1. Create a Unity Project

WwiseUnity

1. Launch Unity and create a new project.
2. Unity
3. Start the Wwise Launcher and integrate Wwise into your Unity project.
2. *Wwise Project Preparation*

For the tutorials, you will need a Sound SFX, an Event to play it, and a SoundBank.

1. From the Launcher, open your project in Wwise.
2. Add a Sound SFX under the Actor Mixer Hierarchy's **Default Work Unit**. In the integration Wwise Project, we added a dialogue.
   1. Import a sound into it.
   2. In the Sound Property Editor's
      1. General Settings tab, enable **Use game-defined auxiliary sends**.

   ![Sound Property Editor General Settings tab](image)

2. Positioning tab, enable positioning and choose **3D**. Optionally, add an Attenuation.
3. Right-click on the Sound SFX within the Actor-Mixer Hierarchy, then select **New Event > Play**.

4. Go to the SoundBank layout (shortcut F7) and add a new SoundBank. Drag the Event into the new SoundBank. Here’s how the SoundBank Manager should look in the integration **Wwise Project**:
SoundBank Manager and Editor

5. Save your project.
3. Unity Project Preparation

In this section we will create our game area and the objects in it. We will need a listener, emitters, and two rooms. From the Launcher, open your project in Unity. Open the Wwise Picker (Windows > Wwise Picker), Refresh Project, and Generate SoundBanks.

![Wwise Picker](image)

3.A. Environment

1. Create a floor, walls, and ceilings using cubes: **GameObject > 3D Object > Cube.**

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Floor</td>
<td>(0, 0, 0)</td>
<td>(0, 0, 0)</td>
<td>(50, 0.5, 50)</td>
</tr>
<tr>
<td>Wall Front Left</td>
<td>(-4.5, 3, -10)</td>
<td>(0, 0, 0)</td>
<td>(7, 6, 0.5)</td>
</tr>
<tr>
<td>Wall Front Middle</td>
<td>(0, 5, -10)</td>
<td>(0, 0, 0)</td>
<td>(2, 2, 0.5)</td>
</tr>
<tr>
<td>Wall Front Right</td>
<td>(2.5, 3, -10)</td>
<td>(0, 0, 0)</td>
<td>(3, 6, 0.5)</td>
</tr>
</tbody>
</table>
2. Add spotlights inside rooms (GameObject > Light > Spotlight).

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spotlight Small Room</td>
<td>(-3, 5.75, -6)</td>
<td>(0, 0, 0)</td>
<td>(1, 1, 1)</td>
</tr>
<tr>
<td>Spotlight Large Room</td>
<td>(11.5, 5.75, 5)</td>
<td>(0, 0, 0)</td>
<td>(1, 1, 1)</td>
</tr>
</tbody>
</table>

1. In the Light component, change:
   1. Range to 30.
   2. Spot Angle to 179.

3.B. Character Listener

1. Create a character with the method of your choice.

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>MainCharacter</td>
<td>(0, 1.3, -20)</td>
<td>(0, 0, 0)</td>
<td>(1, 1, 1)</td>
</tr>
</tbody>
</table>

1. Remove any Unity Audio Source.
2. Add an Ak Audio Listener component to the camera.

3. Add an Ak Spatial Audio Listener component to the camera.
Ak Spatial Audio Listener component

3.C. Third-Person Emitters

1. Create buttons using cylinders (GameObject > 3D Object > Cylinder).

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Button Outside</td>
<td>(-3, 0.75, -15)</td>
<td>(0, 0, 0)</td>
<td>(0.15, 0.5, 0.15)</td>
</tr>
<tr>
<td>Button Small Room</td>
<td>(-3, 0.75, -5)</td>
<td>(0, 0, 0)</td>
<td>(0.15, 0.5, 0.15)</td>
</tr>
<tr>
<td>Button Large Room</td>
<td>(7.5, 0.75, 5)</td>
<td>(0, 0, 0)</td>
<td>(0.15, 0.5, 0.15)</td>
</tr>
</tbody>
</table>

2. Add an Ak Spatial Audio Emitter component.

3. In the Ak Game Obj component:
   1. Enable the Environment Aware option.
   2. Enable the Use Default Listener option.
4. Add an **Ak Ambient** component:
   1. Choose your preferred way to start the sound in **Trigger On**.
   2. Choose the **Play_sound** Event in **Event Name**.

5. Add an **Ak Bank** component:
   1. Add the SoundBank created in **2. Wwise Project Preparation** to **Bank Name**.
6. Start the game, connect it to Wwise, and open the Profiler layout (shortcut F6).
7. Play the sound from a button. You should hear the sound you imported in the Wwise project.
8. In the Voices Graph tab, you should see the following graph.

![Profiler Layout](image)

**Button Outside Voice Graph with no Effect**

3.D. Scene

1. Save your Scene to a convenient place.
2. The final look of the scene is shown in the following image. This scene is available with the Unity integration package. Colors were added to elements of the scene for convenience. A mobile first-person controller was also added.
Tutorial Unity Scene
Surface Reflectors Tutorial

In the following steps, this tutorial will show you how to use the new Reflect plug-in with Spatial Audio. You will need a Reflect plug-in license to make the project work.

- A. Wwise Project
- B. Spatial Audio Emitter
- C. Surface Reflector Component
  - C.1. Use Existing Meshes
  - C.2. Alternative: Create a New Volume

Note: This tutorial presumes that you have completed the Preparation for the Spatial Audio Tutorials. However, it does not consider if the Rooms and Portals Tutorial was done; both tutorials are independent.
A. Wwise Project

We need to update our wwise project with an Auxiliary Bus with the Reflect plug-in and Acoustic Textures.

1. Add an Auxiliary Bus under the Master Audio Bus.
   1. In the Auxiliary Bus Property Editor,
      1. Under the Effects tab, add the Wwise Reflect effect.

   ![Auxiliary Bus Property Editor Effects tab for Reflect](image)

   **Auxiliary Bus Property Editor Effects tab for Reflect**

   2. Under the Positioning tab, enable positioning and choose 2D.
2. Import Reflect Acoustic Textures from: Project > Import Factory Assets...
   1. They will be added to the Virtual Acoustics folder in the ShareSets tab of the Project Explorer view.

 Auxiliary Bus Property Editor Positioning tab for Reflect
Factory Reflect Acoustic Textures in the Project Explorer ShareSets tab

2. (optional) You can add custom Acoustic Textures in the Virtual Acoustics folder under Default Work Unit.

3. Save your project.
B. Spatial Audio Emitter

We need to modify the spatial audio emitter to use our newly created reflect aux bus.

1. Refresh your Wwise project in the Wwise Picker: Windows > Wwise Picker
   1. Generate SoundBanks

   ![Wwise Picker]

   **Wwise Picker**

2. Modify the **Ak Spatial Audio Emitter** component of each button.
   1. Pick the Reflect Auxiliary Bus for **Reflect Aux Bus**.
   2. Choose 2 for the **Reflections Order**.
   3. Choose 1 for the **Reflections Aux Bus Gain**.
   4. Choose 100000 for the **Reflection Max Path Length**.
   5. Check the **Draw First Order Reflections** and **Draw Second Order Reflections** under **Debug Draw**.
Ak Spatial Audio Emitter for Surface Reflector
C. Surface Reflector Component

You can simulate early reflections with the Ak Surface Reflector component. One way to have surface reflectors is to use existing meshes of the game. If the shape is too complex, you may want to add a new object as a surface reflector.

C.1. Use Existing Meshes

The advantages of using the existing meshes are that you don't need to create new objects and each mesh can have a different acoustic texture. The disadvantages are that if you have a complex figure, you may send too much triangles to the Spatial Audio API and this could slow down your game. Also, another disadvantage is that each mesh only gets assigned one texture. In the case of the this tutorial, this means that it is not possible to have a different acoustic texture on the outside than on the inside of the rooms. In the scene provided by the unity integration, this is the method used for surface reflectors.

1. Select all the walls, ceiling and floor from section 3.A. Environment and add an Ak Surface Reflector component.
   1. In the Acoustic Texture parameter, choose an acoustic texture or leave it at None to have an unfiltered reflection.

   ![Ak Surface Reflector component]

2. Since we have enabled the drawing of reflections under Debug Draw in section , you should see rays going from the emitter to the listener in the Scene window. (The ceiling was removed in the following image to see the rays clearly) The blue rays are first order reflections, i.e. they reflect from one surface before reaching the listener, and the yellow rays are second order reflections. The name of the triangles are displayed at the reflection point. A name is made up of the game object's name
and the triangle’s number. A cube has 12 triangles, two per face.

Top view of early reflection drawings for existing mesh surface reflectors

2. You can see the rays in the Game window by enabling Gizmos; as we can see in the next image. From this point of view, we can also see that the Debug Draw option draws the triangles of the surfaces that are being reflected on. Leaving the ceiling in this view, we can see that the sound will also reflect from the ceiling on triangles 7 and 6.
First person view of early reflection drawings for existing mesh surface reflectors

3. Connect the game in Wwise and go to the Profiler Layout. You should see a similar graph when Button Small Room is emitting.
Button Small Room Voice Graph with Reflect Effect

4. Double click on the Auxiliary Bus with the Reflect Effect in the graph. Navigate to the Effects tab and double click on the Reflect Effect. When playing the sound, you should see the current reflections in the graph and the list of the Reflect Effect Editor. In the following image, there is a Drywall acoustic texture for all the walls and the ceiling and a Tile acoustic texture for the floor. When there are two textures in the list, it means that the reflection is a second order one.

![Reflect Effect Editor View](image)

C.2. Alternative: Create a New Volume

Creating a new game object with no mesh renderer can simplify a complex shape. In this optional section, we will show you how to have a surface reflector volume for the interior of the small room. You can also mix solutions by having some meshes be Surface Reflectors and having an object with no mesh renderer as well. For example. Having each mesh be a surface reflector to have different texture per wall inside of the
house and adding and exterior volume to have a completely different texture on the outside of the house. It is also possible to use **Quads** instead of cubes to have a different texture wherever desired. But be careful with intersections when using quads; you don’t want your sound to get out of your shape.

1. Create a simpler unrendered mesh using a cube: GameObject > 3D Object > Cube

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Room Interior</td>
<td>(-2, 2.875, -6)</td>
<td>(0, 0, 0)</td>
<td>(11, 5.25, 7.5)</td>
</tr>
</tbody>
</table>

1. Remove the **Mesh Renderer** component
2. Check the **Is Trigger** in the **Box Collider** component
3. add an Ak **Surface Reflector**.

1. Since we simplified the shape of our room, the door is now part of the reflective surface. We don’t have any reflections from Button Small Room if we are standing outside. We even get a reflection from Button Outside on the door opening. You may have noticed the name of the reflection point is now referencing to Small Room Interior.
Top view of early reflection drawings for a new surface reflector volume when the emitter is outside

2. Entering the small room, Button Small Room's debug drawings appear and the ones for Button Outside disappear.
Top view of early reflection drawings for a new surface reflector volume when the emitter is inside
Rooms and Portals Tutorial

This tutorial will show you how to use Rooms and Portals with Spatial Audio. The following sections detail each step in the tutorial:

- A. Wwise Project
- B. Spatial Audio Emitter
- C. Rooms
- D. Portals
- E. Portals and Reverb
- F. Surface Reflectors and Reverb

Note: This tutorial presumes that you have completed the Preparation for the Spatial Audio Tutorials. However, it does not consider if the Surface Reflectors Tutorial was done; both tutorials are independent. If you did the Surface Reflectors Tutorial and wish to continue with this tutorial, you can do so without changing anything to your scene except for the optional step of C.2. Alternative: Create a New Volume, which should be reverted. If you wish to start anew, and you’ve done the Surface Reflectors Tutorial, you can still keep the same scene and just remove any Ak Surface Reflector components.
A. Wwise Project

For the rooms and portals tutorial, we will need to add Reverb Effects and modify Obstruction curves to hear diffraction through a portal when losing sight of an emitter.

1. Add two Auxiliary Busses under the **Master Audio Bus** called SmallRoom and LargeRoom. In the **Auxiliary Bus Property Editor**,
   1. Under the **Effects** tab, add the Wwise RoomVerb effect. I chose two different factory presets: Rooms/Room_Medium for SmallRoom and Cathedrals/Medium for LargeRoom.

![Auxiliary Bus Property Editor Effects tab for rooms](image)

2. Under the **Positioning** tab, enable positioning and choose 3D.
Auxiliary Bus Property Editor Positioning tab for rooms

2. Project > Project Settings > Obstruction/Occlusion
   1. Change the curves:

<table>
<thead>
<tr>
<th>Curve</th>
<th>Point 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Obstruction Volume</td>
<td>0</td>
</tr>
<tr>
<td>Obstruction LPF</td>
<td>0</td>
</tr>
</tbody>
</table>

2. Here's how the Obstruction Volume curve looks like:
Obstruction/Occlusion curves in the Wwise Project Settings

3. Save your project.
B. Spatial Audio Emitter

We need to modify the spatial audio emitter to use our newly created reflect aux bus.

1. Refresh your Wwise project in the Wwise Picker: Windows > Wwise Picker
   1. Generate SoundBanks

   ![Wwise Picker](image)

   **Wwise Picker**

2. Modify the **Ak Spatial Audio Emitter** component of each button.
   1. Apply a gain of 1 for the **Room Reverb Aux Bus Gain** parameter.
   2. Check **Draw Sound Propagation** under **Debug Draw** and uncheck all the others.
Ak Spatial Audio Emitter for Rooms and Portals

3. We will use rooms in the next section, so we need to add a Rigidbody component to our emitter; as explained in the warning message on the Ak Game Obj component.
   1. Click on Add RigidBody

Ak Game Obj component
C. Rooms

Each of the rooms will have their own room reverb. To do so, we will use the Ak Room component.

1. Create unrendered meshes using cubes: GameObject > 3D Object > Cube

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Room Interior</td>
<td>(-2, 2.875, -6)</td>
<td>(0, 0, 0)</td>
</tr>
<tr>
<td>Large Room Interior</td>
<td>(11.5, 2.875, 3.25)</td>
<td>(0, 0, 0)</td>
</tr>
</tbody>
</table>

1. Remove the **Mesh Renderer** component.
2. Check the **Is Trigger** in the **Box Collider** component.
3. Add an **Ak Room** component to the Small and Large Room Interior Volumes.
   1. Pick the corresponding SmallRoom or LargeRoom Auxiliary Busses for **Reverb Aux Bus**.
   2. If you added a **Rigidbody** to your emitters, you don't need one here. If not, click on Add Rigidbody.

![Ak Room component](image)

**Ak Room component**

2. Start the game, connect to Wwise and go to the Game Object Profiler layout
   1. Watch all the emitters and the listener.
   2. Open a Game Object 3D Viewer view. You should see each emitter in their respective rooms. The next image is a top view of the scene. The listener and Button Outside are not in a room, so they are put by default in the "Outdoors" room. If everything is in the Outdoors room instead of their respective rooms, it means spatial audio is disabled. You must have forgotten to add
an Ak Spatial Audio Listener component to the listener. Follow step 3.B.

Emitters in their respective rooms and the Listener Outdoors

3. Move the listener to a room and see the room name change under your listener game object in the Game Object 3d Viewer. In the following image, the listener is in the large room.
Emitters in their respective rooms and the Listener in the Large Room Interior

3. Press the button in one of the rooms to hear the reverb.
   1. In the small room, the voice graph should look like this:
2. When a listener is in a different room than the emitter, the sound is completely occluded. You will need to add portals to hear the sound coming from rooms through desired openings.

**Note:** To add a reverb outside, add a Room around the entire area. Make sure that the Room has a lower priority than the Small Room Interior and Large Room Interior.
D. Portals

Each Room needs a Portal to be connected to another area. We need two Portals: one on the door between the outside area and the small Room and one between both Rooms.

1. Create two Ak Room Portal components: GameObject > Wwise > Room Portal

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Position</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside Portal</td>
<td>(0, 2, -10)</td>
<td>(0, 0, 0)</td>
</tr>
<tr>
<td>Inside Portal</td>
<td>(3.75, 2, -4.5)</td>
<td>(0, 90, 0)</td>
</tr>
</tbody>
</table>

2. In the Ak Room Portal component:
   1. **Open On** Start.
   2. **Close On** Nothing.
   3. Make sure that the Back Room and Front Room display the right Ak Room components. If not, refresh the inspector by deselecting and selecting the Portal again.

3. This is how the Inside Portal Ak Room Portal component will look like:

   ![Ak Room Portal component](image)

4. In the **Scene** window, a Portal is represented with a yellow ribbon around it indicating its size and the orientation in which it should be placed. The red line shows the separation between front and back areas of the Portal. The front is in the same direction as the local z axis.
5. You should already see the Draw Sound Propagation drawings indicating where the sound will be coming from through the portal. The sphere on the left comes from the small room emitter and the one on the right from the large room emitter. You can see the latter is actually coming from the Portal connecting the small and large Rooms. The sound propagation is represented by red and blue spheres that change in size depending on the wet and dry diffraction angles respectively.
Game window at Start

1. Wwise
2. Play the button outside and go inside the small room.
   1. You should see the same sound propagation drawings coming from the outside emitter through the portal connecting with the outside.

   Sound Propagation drawings from Button Outside when the listener is inside Small Room Interior

2. Move in and out of sight to hear the diffraction applied on the sound. You will hear more or less diffraction depending
on the angle of the listener vs. the portal. In the next image, we are watching the listener and Button Outside. You can see a top view of the scene in the Game Object 3D Viewer. In dark red, we can see the portals. The one in the middle is the Outside Portal, the area on top of the Portal is the Small Room Interior and the area on the bottom is the outside. The Listener is in the Small Room Interior and the playing emitter is Button Outside. On the image, you can see the sound propagation in green between the listener and the emitter. The Diffraction angle is 103.3. Also, you can see the virtual position of the Button Outside emitter on the right. That's where the sound you hear will go through the portal.
E. Portals and Reverb

Sound emitted through Portals can also use the reverb from the current Room the listener is in.

1. Open the SmallRoom Auxiliary Bus Property Editor and check **Use game-defined auxiliary sends** in the General Settings tab. This will send the small Room reverb to other reverbs in the scene.

2. When playing the small room emitter, and going into the large room, you will see the small room's reverb feeding into the large room's reverb.
Button Small Room reverb sending to Large Room Interior room's reverb on the Voices Graph
F. Surface Reflectors and Reverb

It is also possible to feed early reflections into the reverb of a Room. Let's add surface reflectors in our scene.

1. Following section **Surface Reflectors Tutorial**,  
   1. Add an Auxiliary Bus with the Reflect Effect.  
   2. Link each **Ak Spatial Audio Emitter** to the new Reflect Auxiliary Bus.  
   3. Add **Ak Surface Reflector** components on all the walls and the floor.

2. In the Wwise project, open the Auxiliary Bus with the Reflect Effect applied to it.  
   1. In the General Settings tab, check **Use game-defined auxiliary sends**.

3. Play the scene and press the button in the small Room. With the
player in the small Room, here is the Voices Graph we see:

![Image of Voices Graph]

**Early reflections in the small room are sent to the small room's reverb in the Voices Graph**

Wwise Unity IntegrationMon Jan 8 10:46:17 2018  doxygen 1.6.3
Obstruction and Occlusion

It is possible to obstruct or occlude sounds in Unity. See the following sections for details:

- **A. Emitter Obstruction/Occlusion**
- **B. Portal Obstruction**
A. Emitter Obstruction/Occlusion

An emitter can be obstructed or occluded in Unity when the **Ak Emitter Obstruction Occlusion** component is added to its GameObject. When your map doesn't use spatial audio, that is it doesn't have any **Ak Room** components, the **Ak Emitter Obstruction Occlusion** component will only use occlusion. In a spatial audio scene, this same component will use obstruction with the spatial audio listener only; the GameObject with the **Ak Spatial Audio Listener** component. Emitter obstruction is applied to emitters that are in the same room as the spatial audio listener.

1. Add an **Ak Emitter Obstruction Occlusion** component to Button Outside.
   1. In **Layer Mask**, select layers that would block audio. In the SpatialAudioTutorial scene provided with the integration, we have put the first person character on a user created layer called "Player". This layer is not selected in the **Layer Mask** option.
Layer Mask options for the Ak Emitter Obstruction Occlusion component

2. Play the scene.
   1. Play Button Outside.
   2. Go behind the Barrier GameObject to hear it occluded or obstructed depending on the presence of any GameObjects with the Ak Room component in your scene.
B. Portal Obstruction

Sound emitted through portals can also be obstructed. Use the **Ak Room Portal Obstruction** component on a game object with an **Ak Room Portal** component to do so. See section **Rooms and Portals Tutorial** to add rooms and portals to your scene, if not done already.

1. **Add an Ak Room Portal Obstruction** component to Outside Portal.
   1. In **Layer Mask**, select layers that would block audio. In the SpatialAudioTutorial scene provided with the integration, we have put the rooms on a user created layer called "Audio Rooms". This layer is not selected in the **Layer Mask** option.

   ![Ak Room Portal Obstruction component](image1)

   **Ak Room Portal Obstruction component**

   ![Layer Mask options for the Ak Room Portal Obstruction component](image2)

   **Layer Mask options for the Ak Room Portal Obstruction component**

2. Play the scene.
   1. Play Button Small Room.
   2. Go behind the Barrier GameObject to obstruct the portal.
UnityDllNotFoundException

Wwise Profiler
Error during installation - Unity
Wwise for MacSoundBank - Wwise_IDs.h
Error: Plugin
MacUnity Console"Multiple plugins with the same name"
Xbox One"Failed to initialize the sound engine"-in
Wwise 2013.2.8
UNITY_PROJECT_ROOT\Assets\Wwise\Deployment\Examples Components Examples

UnityPackage orms.
UnityDllNotFoundException console

WindowsAkSoundEngineDllNotFoundExceptionUnity2

- DirectX
- Wwise-Unity IntegrationDebug Microsoft Visual Studio 2010 Debug Redistributables
- WindowsMacEditorEditor
- SoundBankUnityEditorWwiseGenerated
- SoundbanksSoundBank1StreamingAssetsWwise
- SoundBank
  Target Platform
- Sound EngineScript Execution Order (menu Edit > Project Settings > Script Execution Order) AkInitializer
  AkTerminator
- Unity...
- Wwise ProfilerWwiseF7Capture Log
- Wwise ProfilerSoundBankProfiling Settings (Alt-G)
  SoundBankAkBank

- Wwise ProfilerCapture LogEventSoundBank
  AkEvent, AkAmbient
Wwise Profiler

Windows

- background Run Unity (menu File > Build Settings > Player Settings)
- port 24024
- Wwise-Unity Plug-in
- Task Manager "adb.exe" Android TCP Android Google

- WwiseRemote Connection
- port 24024
Error during installation - Unity

Wwise: Error during installation: Access to the path is denied

- Plug-in (Debug, Profile, Release) Assets > Wwise > Install Plugins
  - Unity Integration
  - Unity EditorScene
- UnityPlug-in (Debug, Profile, Release)
Wwise for MacSoundBank - Wwise_IDs.h

Wwise for MacSoundBank Wwise_IDs.h:

- Wwise Project
- Project Settings SoundBanks
- Header file path SoundBanks
Error: Plugin

WwiseAkSoundEngine
MacUnity Console"Multiple plugins with the same name"

Mac Unity IntegrationUnity Console

Multiple plug-ins with the same name 'aksoundengine' (found at 'Assets/Wwise/Deployment/Plugins/Mac/Release/AkSoundEngine.bundle' and 'Assets/Wwise/Deployment/Plugins/Mac/Debug/AkSoundEngine.bundle').
Xbox One "Failed to initialize the sound engine" - in

Xbox OneUnityWwiseAssets > Project Settings > Audio "Disable audio"

Unity 5.0.2 "Disable audio"
AkInitializer.cs:

```csharp
AkPlatformInitSettings platformSettings = new AkPlatformInitSettings();
    AkSoundEngine.GetDefaultPlatformInitSettings(platformSettings);
    platformSettings.uLEngineDefaultPoolSize = (uint)lowerPoolSize * 1024;
    platformSettings.fLEngineDefaultPoolRatioThreshold = memoryCutoffThreshold;

AkPlatformInitSettings platformSettings = new AkPlatformInitSettings();
    AkSoundEngine.GetDefaultPlatformInitSettings(platformSettings);
    platformSettings.uLEngineDefaultPoolSize = (uint)lowerPoolSize * 1024;
    platformSettings.fLEngineDefaultPoolRatioThreshold = memoryCutoffThreshold;
#if UNITY_XBOXONE && !UNITY_EDITOR
    platformSettings.uMaxXMAVoices = 0;
#endif
```

XMA
UnityIntegration

Unity

**Scripting Define Symbols:**

<table>
<thead>
<tr>
<th>Platform, Architecture</th>
<th>Custom Preprocessor</th>
<th>Default Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vita, SW</td>
<td>AK_ARCH_VITA_SW</td>
<td>Default</td>
</tr>
<tr>
<td>Vita, HW</td>
<td>AK_ARCH_VITA_HW</td>
<td>NO</td>
</tr>
</tbody>
</table>

**Build Settings > Player Settings... > Other Settings > Scripting Define Symbols**

**Note:**

Wwise Unity Integration Mon Jan 8 10:46:17 2018
<table>
<thead>
<tr>
<th>AK::Wwise::Trigger</th>
</tr>
</thead>
<tbody>
<tr>
<td>AkAmbient</td>
</tr>
<tr>
<td>AkAudioListener</td>
</tr>
</tbody>
</table>
AkBank

AkCallbackManager

AkEmitterObstructionOcclusion

AkEnvironment

AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm
<table>
<thead>
<tr>
<th>AkEnvironmentPortal</th>
</tr>
</thead>
<tbody>
<tr>
<td>AkEvent</td>
</tr>
<tr>
<td>AkEventCallbackMsg</td>
</tr>
<tr>
<td>AkGameObj</td>
</tr>
<tr>
<td>AkInitializer</td>
</tr>
<tr>
<td>--------------</td>
</tr>
<tr>
<td>AkMemBankLoader</td>
</tr>
<tr>
<td>AkRoom</td>
</tr>
<tr>
<td>AkRoomPortal</td>
</tr>
<tr>
<td>AkRoomPortalObstruction</td>
</tr>
<tr>
<td>AkSpatialAudioEmitter</td>
</tr>
<tr>
<td>Class</td>
</tr>
<tr>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>AkSpatialAudioListener</td>
</tr>
<tr>
<td>AkState</td>
</tr>
<tr>
<td>AkSurfaceReflector</td>
</tr>
<tr>
<td>AkSwitch</td>
</tr>
<tr>
<td>Ak Terminator</td>
</tr>
<tr>
<td>AkTriggerBase</td>
</tr>
</tbody>
</table>
AK::Wwise::AcousticTexture
AK::Wwise::AcousticTexture

This type represents an Acoustic Texture.
This type represents an Acoustic Texture.
AK::Wwise::AuxBus
AK::Wwise::AuxBus

This type represents an auxiliary send in the Master-Mixer Hierarchy.

AK::Wwise::AuxBus

AK::Wwise::BaseType

AK::Wwise::AuxBus
This type represents an auxiliary send in the Master-Mixer Hierarchy.
AK::Wwise::Bank
AK::Wwise::Bank

This type can be used to load/unload SoundBanks.
This type can be used to load/unload SoundBanks.
AK::Wwise::BaseGroupType
AK::Wwise::BaseGroupType

This type represents the base for all Wwise Types that also require a group GUID, such as **State** and **Switch**.
This type represents the base for all Wwise Types that also require a group GUID, such as State and Switch.
AK::Wwise::BaseType

This type represents the base for all Wwise Types that require a GUID.

AK::Wwise::BaseType
This type represents the base for all Wwise Types that require a GUID.
AK::Wwise::CallbackFlags
AK::Wwise::CallbackFlags

This type represents the values of the flags used when posting an Event with a callback.
This type represents the values of the flags used when posting an Event with a callback.
AK::Wwise::Event
AK::Wwise::Event

This type can be used to post Events to the sound engine.
## Public

<table>
<thead>
<tr>
<th>uint</th>
<th><strong>Post</strong> (GameObject gameObject)</th>
<th>Posts this <strong>Event</strong> on a GameObject.</th>
</tr>
</thead>
<tbody>
<tr>
<td>uint</td>
<td><strong>Post</strong> (GameObject gameObject, CallbackFlags flags, AkCallbackManager.EventCallback callback, object cookie=null)</td>
<td>Posts this <strong>Event</strong> on a GameObject.</td>
</tr>
<tr>
<td>uint</td>
<td><strong>Post</strong> (GameObject gameObject, uint flags, AkCallbackManager.EventCallback callback, object cookie=null)</td>
<td>Posts this <strong>Event</strong> on a GameObject.</td>
</tr>
<tr>
<td>void</td>
<td><strong>ExecuteAction</strong> (GameObject gameObject, AkActionOnEventType actionOnEventType, int transitionDuration, AkCurveInterpolation curveInterpolation)</td>
<td>Executes various actions on this event associated with a GameObject.</td>
</tr>
<tr>
<td>void</td>
<td><strong>PostMIDI</strong> (GameObject gameObject, AkMIDIPostArray array)</td>
<td>Posts MIDI Events on this <strong>Event</strong> associated with a GameObject.</td>
</tr>
<tr>
<td>void</td>
<td><strong>PostMIDI</strong> (GameObject gameObject, AkMIDIPostArray array, int count)</td>
<td>Posts MIDI Events on this <strong>Event</strong> associated with a GameObject.</td>
</tr>
<tr>
<td>void</td>
<td><strong>StopMIDI</strong> (GameObject gameObject)</td>
<td>Stops MIDI Events on this <strong>Event</strong> associated with a GameObject.</td>
</tr>
<tr>
<td>void</td>
<td><strong>StopMIDI</strong> ()</td>
<td>Stops all MIDI Events on this <strong>Event</strong>.</td>
</tr>
</tbody>
</table>
This type can be used to post Events to the sound engine.
AK::Wwise::RTPC
AK::Wwise::RTPC

This type can be used to set game parameter values to the sound engine.
This type can be used to set game parameter values to the sound engine.
AK::Wwise::State
AK::Wwise::State

This type can be used to set Wwise States.

AK::Wwise::State

AK::Wwise::BaseType

AK::Wwise::BaseGroupType

AK::Wwise::State
This type can be used to set Wwise States.
AK::Wwise::Switch
AK::Wwise::Switch

This type can be used to set **Switch** values on gameobjects.
This type can be used to set **Switch** values on gameobjects.
AK::Wwise::Trigger
AK::Wwise::Trigger

This type can be used to post triggers to the sound engine.
This type can be used to post triggers to the sound engine.
AkAmbient

Use this component to attach a Wwise Event to any object in a scene. The sound can be started at various moments, dependent on the selected Unity trigger. This component is more useful for ambient sounds (sounds related to scene-bound objects) but could also be used for other purposes. Since AkAmbient has AkEvent as its base class, it features the play/stop, play multiple, stop multiple and stop all buttons for previewing the associated Wwise event.

AkAmbient

```
AkEvent
  ↑
AkAmbient
```
**Public**

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>int</code></td>
<td><code>eventID = 0</code></td>
<td>ID of the Event as found in the WwiseID.cs file.</td>
</tr>
<tr>
<td><code>GameObject</code></td>
<td><code>soundEmitterObject = null</code></td>
<td>Game object onto which the Event will be posted. By default, when empty, it is posted on the same object on which the component was added.</td>
</tr>
<tr>
<td><code>bool</code></td>
<td><code>enableActionOnEvent = false</code></td>
<td>Enables additional options to reuse existing events. Use it to transform a Play event into a Stop event without having to define one in the Wwise Project.</td>
</tr>
<tr>
<td><code>AkActionOnEventType</code></td>
<td><code>actionOnEventType</code> = AkActionOnEventType.AkActionOnEventType_Stop`</td>
<td>Replacement action. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td><code>AkCurveInterpolation</code></td>
<td><code>curveInterpolation</code> = AkCurveInterpolation.AkCurveInterpolation_Linear</td>
<td>Fade curve to use with the new Action. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td><code>float</code></td>
<td><code>transitionDuration = 0.0f</code></td>
<td>Duration of the fade. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td><code>const int</code></td>
<td><code>MAX_NB_TRIGGERS = 32</code></td>
<td>Since our mask is a 32 bits integer, we can't have more than 32 triggers.</td>
</tr>
<tr>
<td><code>List&lt;int&gt;</code></td>
<td><code>triggerList = new List&lt;int&gt;() { START_TRIGGER_ID }</code></td>
<td>List containing the enabled triggers.</td>
</tr>
<tr>
<td><code>bool</code></td>
<td><code>useOtherObject = false</code></td>
<td>This property is usefull only when used with colliders. When enabled, the target of the action</td>
</tr>
</tbody>
</table>
will be the other colliding object. When disabled, will be the current object.
<table>
<thead>
<tr>
<th>static Dictionary&lt; uint, string &gt;</th>
<th>triggerTypes = AkTriggerBase.GetAllDerivedTypes ()</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Will contain the types of all the triggers</td>
</tr>
<tr>
<td></td>
<td>derived from <strong>AkTriggerBase</strong> at runtime.</td>
</tr>
</tbody>
</table>
Use this component to attach a Wwise Event to any object in a scene. The sound can be started at various moments, dependent on the selected Unity trigger. This component is more useful for ambient sounds (sounds related to scene-bound objects) but could also be used for other purposes. Since AkAmbient has AkEvent as its base class, it features the play/stop, play multiple, stop multiple and stop all buttons for previewing the associated Wwise event.

: 

- AkAmbientInspector
- AkGameObj
- AkEvent
- Integration Details - Events (Note: This is described in the Wwise SDK documentation.)
AkAudioListener

Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. isDefaultListener determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their AkGameObjListenerList's.
Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. isDefaultListener determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their AkGameObjListenerList's.

- Integrating Listeners (Note: This is described in the Wwise SDK documentation.)
AkBank

Loads and unloads a SoundBank at a specified moment. Vorbis sounds can be decompressed at a specified moment using the decode compressed data option. In that case, the SoundBank will be prepared.

AkUnityEventHandler
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>override void HandleEvent (GameObject in_gameObject)</td>
<td>Loads the SoundBank.</td>
</tr>
<tr>
<td>void UnloadBank (GameObject in_gameObject)</td>
<td>Unloads a SoundBank.</td>
</tr>
</tbody>
</table>
**Public**

<table>
<thead>
<tr>
<th>Type</th>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>bankName = &quot;&quot;</td>
<td>Name of the SoundBank, as specified in the Wwise project.</td>
</tr>
<tr>
<td>bool</td>
<td>loadAsynchronous = false</td>
<td>Check this to load the SoundBank in the background. Be careful, if Events are triggered and the SoundBank hasn't finished loading, you'll have &quot;Event not found&quot; errors.</td>
</tr>
<tr>
<td>bool</td>
<td>decodeBank = false</td>
<td>Decode this SoundBank upon load.</td>
</tr>
<tr>
<td>bool</td>
<td>saveDecodedBank = false</td>
<td>Save the decoded SoundBank to disk for faster loads in the future.</td>
</tr>
<tr>
<td>List&lt;int&gt;</td>
<td>unloadTriggerList = new List&lt;int&gt;() { AkUnityEventHandler.DESTROY_TRIGGER_ID }</td>
<td>Reserved.</td>
</tr>
<tr>
<td>const int</td>
<td>MAX_NB_TRIGGERS = 32</td>
<td>Since our mask is a 32 bits integer, we can't have more than 32 triggers.</td>
</tr>
<tr>
<td>List&lt;int&gt;</td>
<td>triggerList = new List&lt;int&gt;() { START_TRIGGER_ID }</td>
<td>List containing the enabled triggers.</td>
</tr>
<tr>
<td>bool</td>
<td>useOtherObject = false</td>
<td>This property is usefull only when used with colliders. When enabled, the target of the action will be the other colliding object. When disabled, it will be the current object.</td>
</tr>
</tbody>
</table>
## Static Public

<table>
<thead>
<tr>
<th>static Dictionary&lt; uint, string &gt;</th>
<th>triggerTypes = AkTriggerBase.GetAllDerivedTypes ()</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Will contain the types of all the triggers derived from <strong>AkTriggerBase</strong> at runtime.</td>
</tr>
</tbody>
</table>
Loads and unloads a SoundBank at a specified moment. Vorbis sounds can be decompressed at a specified moment using the decode compressed data option. In that case, the SoundBank will be prepared.
**AkCallbackManager**

This class manages the callback queue. All callbacks from the native Wwise SDK go through this queue. The queue needs to be driven by regular calls to `PostCallbacks()`. This is currently done in `AkInitializer.cs`, in `LateUpdate()`. []
Static Public

<table>
<thead>
<tr>
<th>static void</th>
<th>SetMonitoringCallback (AK.Monitor.ErrorLevel in_Level, MonitoringCallback in_CB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Call this to set a function to call whenever Wwise prints a message (warnings or errors).</td>
</tr>
<tr>
<td>static void</td>
<td>SetBGMCallback (BGMCallback in_CB, object in_cookie)</td>
</tr>
<tr>
<td>static int</td>
<td>PostCallbacks ()</td>
</tr>
</tbody>
</table>
This class manages the callback queue. All callbacks from the native Wwise SDK go through this queue. The queue needs to be driven by regular calls to `PostCallbacks()`. This is currently done in `AkInitializer.cs`, in `LateUpdate()`.
AkEmitterObstructionOcclusion

Obstructs/Occludes the emitter of the current game object from its listeners if at least one object is between them.

AkObstructionOcclusion
### Public

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LayerMask</strong></td>
<td>Indicates which layers act as obstructers/occluders.</td>
</tr>
<tr>
<td><strong>refreshInterval</strong></td>
<td>The number of seconds between obstruction/occlusion checks.</td>
</tr>
<tr>
<td><strong>fadeOut</strong></td>
<td>The number of seconds for fade ins and fade outs.</td>
</tr>
<tr>
<td><strong>maxDistance</strong></td>
<td>The maximum distance at which to perform obstruction/occlusion. A negative value will be interpreted as infinite distance.</td>
</tr>
</tbody>
</table>
Obstructs/Occludes the emitter of the current game object from its listeners if at least one object is between them.

The current implementation does not support occlusion.
AkEnvironment

Use this component to define a reverb zone. This needs to be added to a collider object to work properly. AkEnvironment

Inspector (Reverb Zones)
<table>
<thead>
<tr>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>AkEnvironment_CompareBySelectionAlgorithm</td>
</tr>
</tbody>
</table>
Use this component to define a reverb zone. This needs to be added to a collider object to work properly. **AkEnvironmentPortal Inspector (Reverb Zones)**.

This component can be attached to any collider. You can specify a roll-off to fade-in/out of the reverb. The reverb parameters will be defined in the Wwise project, by the sound designer. All **AkGameObj** that are "environment"-aware will receive a send value when entering the attached collider.

:  
- **AkEnvironmentPortal Inspector (Reverb Zones)**
- Integrating Environments and Game-defined Auxiliary Sends (Note: This is described in the Wwise SDK documentation.)
- **AK::SoundEngine::SetGameObjectAuxSendValues** (Note: This is described in the Wwise SDK documentation.)
AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm
AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm

AkEnvironment::AkEnvironment_CompareByPriority
The selection algorithm is as follow:

1. Environments have priorities.
2. Environments have a "Default" flag. This flag effectively says that this environment will be bumped out if any other is present.
3. Environments have an "Exclude Other" flag. This flag will tell that this env is not overlappable with others. So, only one (the highest priority) should be selected.
AkEnvironmentPortal

Use this component to define an area that straddles two different AkEnvironments zones and allow mixing between both zones. 

AkEnvironmentPortalInspector (Reverb Zones)
```plaintext
| float | GetAuxSendValueForPosition(Vector3 in_position, int index) 
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The axis used to find the contribution of each environment.</td>
</tr>
</tbody>
</table>
```
Public

Vector3 axis = new Vector3(1,0,0)
Use this component to define an area that straddles two different AkEnvironments zones and allow mixing between both zones. AkEnvironmentPortalInspector (Reverb Zones) .
AkEvent

Helper class that knows a Wwise Event and when to trigger it in Unity. As of 2017.2.0, the AkEvent inspector has buttons for play/stop, play multiple, stop multiple, and stop all. Play/Stop will play or stop the event such that it can be previewed both in edit mode and play mode. When multiple objects are selected, Play Multiple and Stop Multiple will play or stop the associated AkEvent for each object.
### Public

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td><code>eventID = 0</code></td>
<td>ID of the Event as found in the WwiseID.cs file.</td>
</tr>
<tr>
<td>GameObject</td>
<td><code>soundEmitterObject = null</code></td>
<td>Game object onto which the Event will be posted. By default, when empty, it is posted on the same object on which the component was added.</td>
</tr>
<tr>
<td>bool</td>
<td><code>enableActionOnEvent = false</code></td>
<td>Enables additional options to reuse existing events. Use it to transform a Play event into a Stop event without having to define one in the Wwise Project.</td>
</tr>
<tr>
<td>AkActionOnEventType</td>
<td><code>actionOnEventType = AkActionOnEventType.AkActionOnEventType_Stop</code></td>
<td>Replacement action. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td>AkCurveInterpolation</td>
<td><code>curveInterpolation = AkCurveInterpolation.AkCurveInterpolation_Linear</code></td>
<td>Fade curve to use with the new Action. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td>float</td>
<td><code>transitionDuration = 0.0f</code></td>
<td>Duration of the fade. See AK::SoundEngine::ExecuteEventOnAction().</td>
</tr>
<tr>
<td>const int</td>
<td><code>MAX_NB_TRIGGERS = 32</code></td>
<td>Since our mask is a 32 bits integer, we can’t have more than 32 triggers.</td>
</tr>
<tr>
<td>List&lt;int&gt;</td>
<td><code>triggerList = new List&lt;int&gt; { START_TRIGGER_ID }</code></td>
<td>List containing the enabled triggers.</td>
</tr>
<tr>
<td>bool</td>
<td><code>useOtherObject = false</code></td>
<td>This property is useful only when used with colliders. When enabled, the target of the action</td>
</tr>
</tbody>
</table>
will be the other colliding object. When disabled, will be the current object.
**Static Public**

<table>
<thead>
<tr>
<th>static Dictionary&lt; uint, string &gt;</th>
<th><code>triggerTypes = AkTriggerBase.GetAllDerivedTypes ()</code></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Will contain the types of all the triggers derived from <strong>AkTriggerBase</strong> at runtime.</td>
</tr>
</tbody>
</table>
Helper class that knows a Wwise Event and when to trigger it in Unity. As of 2017.2.0, the AkEvent inspector has buttons for play/stop, play multiple, stop multiple, and stop all. Play/Stop will play or stop the event such that it can be previewed both in edit mode and play mode. When multiple objects are selected, Play Multiple and Stop Multiple will play or stop the associated AkEvent for each object.

- **Edit Mode**
- **AkAmbientInspector**
- **Integration Details - Events** (Note: This is described in the Wwise SDK documentation.)
AkEventCallbackMsg

Event callback information. Event callback functions can receive this structure as a parameter. []
Public

<table>
<thead>
<tr>
<th>GameObject</th>
<th>sender</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AkSoundEngine.PostEvent callback flags. See the AkCallbackType enumeration for a list of all callbacks.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AkCallbackInfo</th>
<th>info</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GameObject from whom the callback function was called.</td>
</tr>
</tbody>
</table>
Event callback information. Event callback functions can receive this structure as a parameter.
AkGameObj

This component represents a sound object in your scene tracking its position and other game syncs such as Switches, RTPC and environment values. You can add this to any object that will emit sound, and it will be added to any object that an AkAudioListener is attached to. Note that if it is not present, Wwise will add it automatically, with the default values, to any Unity Game Object that is passed to Wwise.
## Public

<table>
<thead>
<tr>
<th>Type</th>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bool</td>
<td><strong>AddListener</strong> <em>(AkAudioListener listener)</em></td>
<td>Adds an <strong>AkAudioListener</strong> to the container of listeners listening to this gameobject.</td>
</tr>
<tr>
<td>bool</td>
<td><strong>RemoveListener</strong> <em>(AkAudioListener listener)</em></td>
<td>Removes an <strong>AkAudioListener</strong> from the container of listeners listening to this gameobject.</td>
</tr>
<tr>
<td>virtual Vector3</td>
<td><strong>GetPosition</strong> ()</td>
<td></td>
</tr>
<tr>
<td>virtual Vector3</td>
<td><strong>GetForward</strong> ()</td>
<td></td>
</tr>
<tr>
<td>virtual Vector3</td>
<td><strong>GetUpward</strong> ()</td>
<td></td>
</tr>
<tr>
<td>AkGameObjPositionOffsetData</td>
<td>m_positionOffsetData = null</td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td>----------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>When not set to null, the position will be offset relative to the Game Object position by the Position Offset.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>bool</th>
<th>isEnvironmentAware = true</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Is this object affected by Environment changes? Set to false if not affected in order to save some useless calls. Default is true.</td>
</tr>
</tbody>
</table>
This component represents a sound object in your scene tracking its position and other game syncs such as Switches, RTPC and environment values. You can add this to any object that will emit sound, and it will be added to any object that an AkAudioListener is attached to. Note that if it is not present, Wwise will add it automatically, with the default values, to any Unity Game Object that is passed to Wwise.

- **Integration Details - Game Objects** (Note: This is described in the Wwise SDK documentation.)
- **Integration Details - Events** (Note: This is described in the Wwise SDK documentation.)
- **Integrating Listeners** (Note: This is described in the Wwise SDK documentation.)
- **Integration Details - Switches** (Note: This is described in the Wwise SDK documentation.)
- **Integration Details - States** (Note: This is described in the Wwise SDK documentation.)
- **Integration Details - Environments and Game-defined Auxiliary Sends** (Note: This is described in the Wwise SDK documentation.)
AkInitializer
### Public

<table>
<thead>
<tr>
<th>Type</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td><code>basePath</code> = AkSoundEngineController.s_DefaultBasePath</td>
<td>Path for the soundbanks. This must contain one sub-folder per platform, with the same as in the Wwise project.</td>
</tr>
<tr>
<td>string</td>
<td><code>language</code> = AkSoundEngineController.s_Language</td>
<td>Language sub-folder.</td>
</tr>
<tr>
<td>int</td>
<td><code>defaultPoolSize</code> = AkSoundEngineController.s_DefaultPoolSize</td>
<td>Default Pool size. This contains the meta data for your project. Default size is 4 MB, but you should adjust for your needs.</td>
</tr>
<tr>
<td>int</td>
<td><code>lowerPoolSize</code> = AkSoundEngineController.s_LowerPoolSize</td>
<td>Lower Pool size. This contains the audio processing buffers and DSP data. Default size is 2 MB, but you should adjust for your needs.</td>
</tr>
<tr>
<td>int</td>
<td><code>streamingPoolSize</code> = AkSoundEngineController.s_StreamingPoolSize</td>
<td>Streaming Pool size. This contains the streaming buffers. Default size is 1 MB, but you should adjust for your needs.</td>
</tr>
<tr>
<td>int</td>
<td><code>preparePoolSize</code> = AkSoundEngineController.s_PreparePoolSize</td>
<td>Prepare Pool size. This contains the banks loaded using PrepareBank (Banks decoded on load use this). Default size is 0 MB, but you should adjust for your needs.</td>
</tr>
<tr>
<td>float</td>
<td><code>memoryCutoffThreshold</code> = AkSoundEngineController.s_MemoryCutoffThreshold</td>
<td>This setting will trigger the killing of sounds when the memory is reaching 95% of capacity. Lowest priority sounds are killed.</td>
</tr>
<tr>
<td>int</td>
<td><code>monitorPoolSize</code> = AkSoundEngineController.s_MonitorPoolSize</td>
<td>Monitor Pool size. Size of the monitoring pool, in bytes. This parameter is not used in Release build.</td>
</tr>
<tr>
<td>Type</td>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>int</td>
<td>monitorQueuePoolSize = AkSoundEngineController.s_MonitorQueuePoolSize</td>
<td>Monitor Queue Pool size. Size of the monitoring queue pool, in bytes. This parameter is not used in Release build.</td>
</tr>
<tr>
<td>int</td>
<td>callbackManagerBufferSize = AkSoundEngineController.s_CallbackManagerBuffersize</td>
<td>CallbackManager buffer size. The size of the buffer per-frame to transfer callback data. Default size is 4 KB, you should increase this, if required.</td>
</tr>
<tr>
<td>int</td>
<td>spatialAudioPoolSize = AkSoundEngineController.s_SpatialAudioPoolSize</td>
<td>Spatial Audio Lower Pool size. Default size is 4 MB, should adjust for your needs.</td>
</tr>
<tr>
<td>uint</td>
<td>maxSoundPropagationDepth = AkSoundEngine.AK_MAX_SOUND_PROPAGATION_DEPTH</td>
<td>Spatial Audio Max Sound Propagation Depth. Maximum number of rooms that sound can propagate through; must be less than or equal to AK_MAX_SOUND_PROPAGATION_DEPTH.</td>
</tr>
<tr>
<td>AkDiffractionFlags</td>
<td>diffractionFlags = AkDiffractionFlags.DefaultDiffractionFlags</td>
<td>Enable or disable specific diffusion features. See AkDiffractionFlags.</td>
</tr>
<tr>
<td>bool</td>
<td>engineLogging = AkSoundEngineController.s_EngineLogging</td>
<td>Enable Wwise engine logging. Option to turn on/off the logging of the Wwise engine.</td>
</tr>
</tbody>
</table>
This script deals with initialization, and frame updates of the Wwise audio engine. It is marked as `DontDestroyOnLoad` so it stays active for the life of the game, not only one scene. You can, and probably should, modify this script to change the initialization parameters for the sound engine. A few are already exposed in the property inspector. It must be present on one GameObject at the beginning of the game to initialize the audio properly. It must be executed BEFORE any other MonoBehaviors that use `AkSoundEngine`.

:  
  - **Initialize the Different Modules of the Sound Engine** (Note: This is described in the Wwise SDK documentation.)  
  - `AK::SoundEngine::Init()` (Note: This is described in the Wwise SDK documentation.)  
  - `AK::SoundEngine::Term()` (Note: This is described in the Wwise SDK documentation.)  
  - `AkCallbackManager`
AkMemBankLoader
### Public

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>void <strong>LoadNonLocalizedBank</strong> (string in_bankFilename)</td>
<td>Load a sound bank from WWW object.</td>
</tr>
<tr>
<td>void <strong>LoadLocalizedBank</strong> (string in_bankFilename)</td>
<td>Load a language-specific bank from WWW object.</td>
</tr>
<tr>
<td>Public</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td>string</td>
<td><strong>bankName</strong> = &quot;&quot;</td>
</tr>
<tr>
<td></td>
<td>Name of the bank to load.</td>
</tr>
<tr>
<td>bool</td>
<td><strong>isLocalizedBank</strong> = false</td>
</tr>
<tr>
<td></td>
<td>Is the bank localized (situated in the language specific folders).</td>
</tr>
</tbody>
</table>
This class is an example of how to load banks in Wwise, if the bank data was preloaded in memory. This would be useful for situations where you use the WWW class
AkRoom

An AkRoom is an enclosed environment that can only communicate to the outside/other rooms with AkRoomPortals.
| ulong   | **GetID** () | Access the room's ID. |
Public

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>float</td>
<td>reverbLevel = 1</td>
<td>The reverb control value for the send to the reverb aux bus.</td>
</tr>
<tr>
<td>float</td>
<td>wallOcclusion = 1</td>
<td>Occlusion level modeling transmission through walls.</td>
</tr>
<tr>
<td>int</td>
<td>priority = 0</td>
<td></td>
</tr>
</tbody>
</table>
An AkRoom is an enclosed environment that can only communicate to the outside/other rooms with AkRoomPortals.
AkRoomPortal

An AkRoomPortal can connect two AkRoom components together. [ ]

AkUnityEventHandler
## Public

<table>
<thead>
<tr>
<th>Type</th>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ulong</td>
<td><strong>GetID ()</strong></td>
<td>Access the portal's ID.</td>
</tr>
<tr>
<td>override void</td>
<td><strong>HandleEvent</strong> (GameObject in_gameObject)</td>
<td>Opens the portal on trigger event.</td>
</tr>
<tr>
<td>void</td>
<td><strong>ClosePortal</strong> (GameObject in_gameObject)</td>
<td>Closes the portal on trigger event.</td>
</tr>
</tbody>
</table>
AkRoomPortals can only connect a maximum of 2 rooms. Since our mask is a 32 bits integer, we can't have more than 32 triggers. This property is useful only when used with colliders. When enabled, the target of the action will be the other colliding object. When disabled, it will be the current object.
### Static Public

<table>
<thead>
<tr>
<th>static Dictionary&lt; uint, string &gt;</th>
<th><code>triggerTypes = AkTriggerBase.GetAllDerivedTypes ()</code></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Will contain the types of all the triggers derived from <em>AkTriggerBase</em> at runtime.</td>
</tr>
</tbody>
</table>
An AkRoomPortal can connect two AkRoom components together.
AkRoomPortalObstruction

Obstructs/Occludes the spatial audio portal of the current game object from the spatial audio listener if at least one object is between them.

AkObstructionOcclusion
## Public

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LayerMask</td>
<td><strong>LayerMask = -1</strong>&lt;br&gt;Indicates which layers act as obstructers/occluders.</td>
</tr>
<tr>
<td>float</td>
<td><strong>refreshInterval</strong> = 1&lt;br&gt;The number of seconds between obstruction/occlusion checks.</td>
</tr>
<tr>
<td>float</td>
<td><strong>fadeTime</strong> = 0.5f&lt;br&gt;The number of seconds for fade ins and fade outs.</td>
</tr>
<tr>
<td>float</td>
<td><strong>maxDistance</strong> = -1.0f&lt;br&gt;The maximum distance at which to perform obstruction/occlusion. A negative value will be interpreted as infinite distance.</td>
</tr>
</tbody>
</table>
Obstructs/Occludes the spatial audio portal of the current game object from the spatial audio listener if at least one object is between them.

If no spatial audio listener has been registered, there will be no obstruction.
AkSpatialAudioEmitter

Add this script on the GameObject which represents an emitter that uses the Spatial Audio API.

AkSpatialAudioBase
### Public

<table>
<thead>
<tr>
<th>AK.Wwise.AuxBus</th>
<th>reflectAuxBus</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Auxiliary Bus with a Reflect plug-in Effect applied.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>ReflectionOrder = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>uint</td>
<td>reflectionsOrder</td>
</tr>
<tr>
<td>float</td>
<td>reflectionsAuxBusGain = 1</td>
</tr>
<tr>
<td>The gain [0, 1] applied to the reflect auxiliary bus.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>reflectionMaxPathLength = 1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>float</td>
<td>reflectionMaxPathLength</td>
</tr>
<tr>
<td>The maximum path length a sound path can have from the emitter to the listener after reflecting on surfaces.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>roomReverbAuxBusGain = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>float</td>
<td>roomReverbAuxBusGain</td>
</tr>
<tr>
<td>Send gain (0.f-1.f) that is applied when sending to the aux bus associated with the room that the emitter is in.</td>
<td></td>
</tr>
</tbody>
</table>
Add this script on the GameObject which represents an emitter that uses the Spatial Audio API.
AkSpatialAudioListener

Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. `isDefaultListener` determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their `AkGameObjListenerList`'s.

AkSpatialAudioBase
Add this script on the game object that represent a listener. This is normally added to the Camera object or the Player object, but can be added to any game object when implementing 3D busses. 

isDefaultListener determines whether the game object will be considered a default listener - a listener that automatically listens to all game objects that do not have listeners attached to their AkGameObjListenerList's.

: 

- Integrating Listeners (Note: This is described in the Wwise SDK documentation.)
AkState

This will call AkSoundEngine.SetState() whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it. []

AkUnityEventHandler
<table>
<thead>
<tr>
<th><strong>Public</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>int</strong></td>
</tr>
<tr>
<td><strong>int</strong></td>
</tr>
<tr>
<td><strong>const int</strong></td>
</tr>
<tr>
<td><strong>List&lt;int&gt;</strong></td>
</tr>
<tr>
<td><strong>bool</strong></td>
</tr>
</tbody>
</table>
**Static Public**

| static Dictionary< uint, string > | **triggerTypes** = AkTriggerBase.GetAllDerivedTypes () | Will contain the types of all the triggers derived from **AkTriggerBase** at runtime. |
This will call AkSoundEngine.SetState() whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it.

- **Integration Details - States** (Note: This is described in the Wwise SDK documentation.)
AkSurfaceReflector

This component will convert the triangles of the GameObject's geometry into sound reflective surfaces.
## Static Public

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddGeometrySet</strong></td>
<td>Sends the mesh filter's triangles and their acoustic texture to Spatial Audio.</td>
</tr>
<tr>
<td><strong>RemoveGeometrySet</strong></td>
<td>Remove the corresponding mesh filter's geometry from Spatial Audio.</td>
</tr>
<tr>
<td><strong>AK.Wwise.AcousticTexture</strong></td>
<td><strong>AcousticTexture</strong></td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>All triangles of the component's mesh will be applied with this texture. The texture will change the filter parameters of the sound reflected from this component.</td>
<td></td>
</tr>
</tbody>
</table>
This component will convert the triangles of the GameObject's geometry into sound reflective surfaces.

This component requires a Mesh Filter component. The triangles of the mesh will be sent to the Spatial Audio wrapper by calling SpatialAudio::AddGeometrySet(). The triangles will reflect the sound emitted from AkSpatialAudioEmitter components.
AkSwitch

This will call AkSoundEngine.SetSwitch() whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it. 

AkUnityEventHandler
## Public

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td><code>groupId</code></td>
<td>Switch Group ID, as defined in WwiseID.cs.</td>
</tr>
<tr>
<td>int</td>
<td><code>valueID</code></td>
<td>Switch Value ID, as defined in WwiseID.cs.</td>
</tr>
<tr>
<td>const int</td>
<td><code>MAX_NB_TRIGGERS</code> = 32</td>
<td>Since our mask is a 32 bits integer, we can't have more than 32 triggers.</td>
</tr>
<tr>
<td>List&lt;int&gt;</td>
<td><code>triggerList</code> = new List&lt;int&gt;() { START_TRIGGER_ID }</td>
<td>List containing the enabled triggers.</td>
</tr>
<tr>
<td>bool</td>
<td><code>useOtherObject</code> = false</td>
<td>This property is useful only when used with colliders. When enabled, the target of the action will be the other colliding object. When disabled, it will be the current object.</td>
</tr>
</tbody>
</table>
### Static Public

<table>
<thead>
<tr>
<th>static Dictionary&lt; uint, string &gt;</th>
<th><code>triggerTypes = AkTriggerBase.GetAllDerivedTypes ()</code></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Will contain the types of all the triggers derived from <code>AkTriggerBase</code> at runtime.</td>
</tr>
</tbody>
</table>
This will call `AkSoundEngine.SetSwitch()` whenever the selected Unity event is triggered. For example this component could be set on a Unity collider to trigger when an object enters it.

- **Integration Details - Switches** (Note: This is described in the Wwise SDK documentation.)
This script deals with termination of the Wwise audio engine. It must be present on one Game Object that gets destroyed last in the game. It must be executed AFTER any other monoBehaviors that use AkSoundEngine.

- **Terminate the Different Modules of the Sound Engine** (Note: This is described in the Wwise SDK documentation.)
- **AK::SoundEngine::Term()** (Note: This is described in the Wwise SDK documentation.)
AkTriggerBase

AkTriggerCollisionEnter, AkTriggerCollisionExit, AkTriggerDisable, AkTriggerEnable, AkTriggerEnter, AkTriggerExit, AkTriggerMouseDown, AkTriggerMouseEnter, AkTriggerMouseExit, AkTriggerMouseUp
Public

<table>
<thead>
<tr>
<th>delegate void</th>
<th>Trigger (GameObject in_gameObject)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delegate declaration for all Wwise Triggers.</td>
</tr>
</tbody>
</table>
## Public

<table>
<thead>
<tr>
<th>Trigger</th>
<th><code>triggerDelegate = null</code></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All components reacting to the trigger will be registered in this delegate.</td>
</tr>
</tbody>
</table>
Base class for the generic triggering mechanism for Wwise Integration. All Wwise components will use this mechanism to drive their behavior. Derive from this class to add your own triggering condition, as described in Adding New Triggers for Wwise Events
• AK::Wwise::BaseType
  ▪ AK::Wwise::AcousticTexture
  ▪ AK::Wwise::AuxBus
  ▪ AK::Wwise::Bank
  ▪ AK::Wwise::BaseGroupType
    ▪ AK::Wwise::State
    ▪ AK::Wwise::Switch
  ▪ AK::Wwise::Event
  ▪ AK::Wwise::RTPC
  ▪ AK::Wwise::Trigger
• AK::Wwise::CallbackFlags
• AkAudioListener
• AkBank
• AkCallbackManager
• AkEmitterObstructionOcclusion
• AkEnvironment
• AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm
• AkEnvironmentPortal
• AkEvent
  ▪ AkAmbient
• AkEventCallbackMsg
• AkGameObj
• AkInitializer
• AkMemBankLoader
• AkRoom
• AkRoomPortal
• AkRoomPortalObstruction
• AkSpatialAudioEmitter
• AkSpatialAudioListener
• AkState
• AkSurfaceReflector
• AkSwitch
• AkTerminator
• AkTriggerBase
- a -
  - AcousticTexture : AkSurfaceReflector
  - actionOnEventType : AkEvent
  - AddGeometrySet() : AkSurfaceReflector
  - AddListener() : AkGameObj
  - axis : AkEnvironmentPortal

- b -
  - bankName : AkBank, AkMemBankLoader
  - basePath : AkInitializer

- c -
  - callbackManagerBufferSize : AkInitializer
  - ClosePortal() : AkRoomPortal
  - curveInterpolation : AkEvent

- d -
  - decodeBank : AkBank
  - defaultPoolSize : AkInitializer
  - diffractionFlags : AkInitializer

- e -
  - enableActionOnEvent : AkEvent
  - engineLogging : AkInitializer
  - eventID : AkEvent
- g -
  - ExecuteAction() : AK::Wwise::Event
  
  - GetAuxSendValueForPosition() : AkEnvironmentPortal
  - GetForward() : AkGameObj
  - GetID() : AkRoomPortal, AkRoom
  - GetPosition() : AkGameObj
  - GetUpward() : AkGameObj
  - groupId : AkSwitch, AkState

- h -
  - HandleEvent() : AkBank, AkRoomPortal

- i -
  - info : AkEventCallbackMsg
  - isEnvironmentAware : AkGameObj
  - isLocalizedBank : AkMemBankLoader

- l -
  - language : AkInitializer
  - loadAsynchronous : AkBank
  - LoadLocalizedBank() : AkMemBankLoader
  - LoadNonLocalizedBank() : AkMemBankLoader
  - lowerPoolSize : AkInitializer

- m -
  - m_positionOffsetData : AkGameObj
  - MAX_ROOMS_PER_PORTAL : AkRoomPortal
  - maxSoundPropagationDepth : AkInitializer
  - memoryCutoffThreshold : AkInitializer
  - monitorPoolSize : AkInitializer
  - monitorQueuePoolSize : AkInitializer
- p -

- Post() : \texttt{AK::Wwise::Event}
- PostCallbacks() : \texttt{AkCallbackManager}
- PostMIDI() : \texttt{AK::Wwise::Event}
- preparePoolSize : \texttt{AkInitializer}
- priority : \texttt{AkRoom}

- r -

- reflectAuxBus : \texttt{AkSpatialAudioEmitter}
- reflectionMaxPathLength : \texttt{AkSpatialAudioEmitter}
- reflectionsAuxBusGain : \texttt{AkSpatialAudioEmitter}
- reflectionsOrder : \texttt{AkSpatialAudioEmitter}
- RemoveGeometrySet() : \texttt{AkSurfaceReflector}
- RemoveListener() : \texttt{AkGameObj}
- reverbAuxBus : \texttt{AkRoom}
- reverbLevel : \texttt{AkRoom}
- roomReverbAuxBusGain : \texttt{AkSpatialAudioEmitter}
- rooms : \texttt{AkRoomPortal}

- s -

- saveDecodedBank : \texttt{AkBank}
- sender : \texttt{AkEventCallbackMsg}
- SetBGMCallback() : \texttt{AkCallbackManager}
- SetMonitoringCallback() : \texttt{AkCallbackManager}
- soundEmitterObject : \texttt{AkEvent}
- spatialAudioPoolSize : \texttt{AkInitializer}
- StopMIDI() : \texttt{AK::Wwise::Event}
- streamingPoolSize : \texttt{AkInitializer}

- t -

- transitionDuration : \texttt{AkEvent}
- Trigger() : \texttt{AkTriggerBase}
- triggerDelegate : \texttt{AkTriggerBase}
- u -

- UnloadBank() : AkBank
- unloadTriggerList : AkBank

- v -

- valueID : AkState, AkSwitch

- w -

- wallOcclusion : AkRoom
AK::Wwise::Event

uint AK::Wwise::Event::Post (GameObject \texttt{gameObject})

Posts this \texttt{Event} on a GameObject.

\hspace{1em} \texttt{gameObject}  The GameObject

\hspace{1em} Returns the playing ID.
void AK::Wwise::Event::PostMIDI ( GameObject gameObject, AkMIDIPostArray array )

Posts MIDI Events on this Event associated with a GameObject:

gameObject    The GameObject
array          The array of AkMIDIPost that are posted.
AkInitializer

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>basePath</strong></td>
<td>Path for the soundbanks. This must contain one subfolder per platform, with the same as in the Wwise project.</td>
</tr>
<tr>
<td><strong>callbackManagerBufferSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>defaultPoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>diffractionFlags</strong></td>
<td></td>
</tr>
<tr>
<td><strong>engineLogging</strong></td>
<td></td>
</tr>
<tr>
<td><strong>language</strong></td>
<td></td>
</tr>
<tr>
<td><strong>lowerPoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>maxSoundPropagationDepth</strong></td>
<td></td>
</tr>
<tr>
<td><strong>memoryCutoffThreshold</strong></td>
<td></td>
</tr>
<tr>
<td><strong>monitorPoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>monitorQueuePoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>preparePoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>spatialAudioPoolSize</strong></td>
<td></td>
</tr>
<tr>
<td><strong>streamingPoolSize</strong></td>
<td></td>
</tr>
</tbody>
</table>

```
string AkInitializer::basePath = AkSoundEngineController.s_DefaultBasePath
```

Wwise Unity Integration Mon Jan 8 10:46:18 2018 doxygen 1.6.3
### AkCallbackManager

#### PostCallbacks
- `SetBGMCallback`
- `SetMonitoringCallback`

#### static void AkCallbackManager::SetMonitoringCallback

Call this to set a function to call whenever Wwise prints a message (warnings or errors).

---

Wwise Unity Integration Mon Jan 8 10:46:17 2018  doxygen 1.6.3
AkMemBankLoader

<table>
<thead>
<tr>
<th>bankName</th>
<th>void AkMemBankLoader::LoadNonLocalizedBank</th>
</tr>
</thead>
<tbody>
<tr>
<td>isLocalizedBank</td>
<td></td>
</tr>
<tr>
<td>LoadLocalizedBank</td>
<td></td>
</tr>
<tr>
<td>LoadNonLocalizedBank</td>
<td>Load a sound bank from WWW object.</td>
</tr>
</tbody>
</table>

Wwise Unity Integration Mon Jan 8 10:46:18 2018 doxygen 1.6.3
### AkCallbackManager

<table>
<thead>
<tr>
<th>PostCallbacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>SetBGMCallback</td>
</tr>
<tr>
<td>SetMonitoringCallback</td>
</tr>
</tbody>
</table>

**static void AkCallbackManager::SetBGMCallback**

Call this to set a iOS callback interruption function.
AcousticTexture (AK::Wwise)

AkAmbient

AkAudioListener

AkBank

AkCallbackManager

AkEmitterObstructionOcclusion

AkEnvironment

AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm
static int AkCallbackManager::PostCallbacks

This function dispatches all the accumulated callbacks from the sound engine. It must be called regularly. By default this is called in AkInitializer.cs.
AK::Wwise::BaseType

AK::Wwise::BaseType
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ExecuteAction</strong></td>
<td>(GameObject gameObject, AkActionOnEventType actionOnEventType, int transitionDuration, AkCurveInterpolation curveInterpolation)</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>(GameObject gameObject)</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>(GameObject gameObject, CallbackFlags flags, AkCallbackManager.EventCallback callback, object cookie=null)</td>
</tr>
<tr>
<td><strong>Post</strong></td>
<td>(GameObject gameObject, uint flags, AkCallbackManager.EventCallback callback, object cookie=null)</td>
</tr>
<tr>
<td><strong>PostMIDI</strong></td>
<td>(GameObject gameObject, AkMIDIPostArray array)</td>
</tr>
<tr>
<td><strong>PostMIDI</strong></td>
<td>(GameObject gameObject, AkMIDIPostArray array, int count)</td>
</tr>
<tr>
<td><strong>StopMIDI</strong></td>
<td>(GameObject gameObject)</td>
</tr>
<tr>
<td><strong>StopMIDI</strong></td>
<td>()</td>
</tr>
</tbody>
</table>
# AK::Wwise::Event

```cpp
uint AK::Wwise::Event::Post ( GameObject CallbackFlags AkCallbackManager object )
```

Posts this **Event** on a GameObject.

```cpp
gameObject  // The GameObject
flags
callback
cookie  // Optional cookie received by the callback
```

Returns the playing ID.
```c++
uint AK::Wwise::Event::Post ( GameObject object, AkCallbackManager::EventCallback callback, uint flags, uint cookie )
```

Posts this **Event** on a GameObject.

- `GameObject` The GameObject
- `flags` Optional flags
- `callback` Callback function
- `cookie` Optional cookie received by the callback

Returns the playing ID.
## AK::Wwise::Event

### ExecuteAction

```cpp
void AK::Wwise::Event::ExecuteAction ( GameObject actionOnEventType, int transitionDuration, AkCurveInterpolation curveInterpolation )
```

Executes various actions on this event associated with:

- `gameObject`
- `actionOnEventType`
- `transitionDuration`
- `curveInterpolation`

Wwise Unity Integration Mon Jan 8 10:46:18 2018 doxygen 1.6.3
**AK::Wwise::Event**

```cpp
void AK::Wwise::Event::PostMIDI ( GameObject AkMIDIPostArray int count )
```

Posts MIDI Events on this `Event` associated with a GameObject:

- `gameObject` The GameObject
- `array` The array of `AkMIDIPost` that are posted
- `count` The number of elements from the array that are posted.
**void AK::Wwise::Event::StopMIDI ( GameObject & gameObject)**

Stops MIDI Events on this **Event** associated with a GameObject.

```cpp
gameObject  // The GameObject
```
void AK::Wwise::Event::StopMIDI() [inline]

Stops all MIDI Events on this Event.
AK::Wwise::Trigger

AK::Wwise::Trigger

Wwise Unity Integration Mon Jan 8 10:46:18 2018  doxygen 1.6.3
<table>
<thead>
<tr>
<th>AkAmbient</th>
<th>AkEvent</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionOnEventType</td>
<td>AkEvent</td>
</tr>
<tr>
<td>curveInterpolation</td>
<td>AkEvent</td>
</tr>
<tr>
<td>enableActionOnEvent</td>
<td>AkEvent</td>
</tr>
<tr>
<td>eventID</td>
<td>AkEvent</td>
</tr>
<tr>
<td>soundEmitterObject</td>
<td>AkEvent</td>
</tr>
<tr>
<td>transitionDuration</td>
<td>AkEvent</td>
</tr>
</tbody>
</table>
int AkEvent::eventId = 0

ID of the Event as found in the WwiseID.cs file.
### AkEvent

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>actionOnEventType</code></td>
<td></td>
</tr>
<tr>
<td><code>curveInterpolation</code></td>
<td></td>
</tr>
<tr>
<td><code>enableActionOnEvent</code></td>
<td></td>
</tr>
<tr>
<td><code>eventID</code></td>
<td></td>
</tr>
<tr>
<td><code>soundEmitterObject</code></td>
<td>Game object onto which the Event will be posted. By default, when empty, it is posted on the same object on which the component was added.</td>
</tr>
<tr>
<td><code>transitionDuration</code></td>
<td></td>
</tr>
</tbody>
</table>
AkEvent

### enableActionOnEvent

```cpp
bool AkEvent::enableActionOnEvent = false
```

Enables additional options to reuse existing events. Use it to transform a Play event into a Stop event without having to define one in the Wwise Project.
AkEvent

- `actionOnEventType`:
  Replacement action. See AK::SoundEngine::ExecuteEventOnAction().

- `curveInterpolation`:
- `enableActionOnEvent`:
- `eventID`:
- `soundEmitterObject`:
- `transitionDuration`:
AkEvent

- actionOnEventType
- curveInterpolation
- enableActionOnEvent
- eventID
- soundEmitterObject
- transitionDuration

**AkCurveInterpolation** `AkEvent::curveInterpolation`

Fade curve to use with the new Action. See `AK::SoundEngine::ExecuteEventOnAction()`.
<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionOnEventType</td>
<td></td>
</tr>
<tr>
<td>curveInterpolation</td>
<td></td>
</tr>
<tr>
<td>enableActionOnEvent</td>
<td></td>
</tr>
<tr>
<td>eventID</td>
<td></td>
</tr>
<tr>
<td>soundEmitterObject</td>
<td></td>
</tr>
<tr>
<td>transitionDuration</td>
<td></td>
</tr>
</tbody>
</table>

```
float AkEvent::transitionDuration = 0.0f
```

Duration of the fade. See `AK::SoundEngine::ExecuteEventOnAction()`.
Since our mask is a 32 bits integer, we can't have more than...
List<int> AkUnityEventHandler::triggerList = new List<int>() {
START_TRIGGER_ID
};

List containing the enabled triggers.
AkSwitch

This property is useful only when used with colliders. When enabled, the target of the action will be the other colliding object. When disabled, it will be the current object.
**AkSwitch**

**Dictionary<uint, string>** AkUnityEventHandler::triggerTypes = AkTriggerBase.GetAllDerivedTypes()

Will contain the types of all the triggers derived from `AkTriggerBase`.
<table>
<thead>
<tr>
<th>Method</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>bankName</td>
<td>AkBank</td>
</tr>
<tr>
<td>decodeBank</td>
<td>AkBank</td>
</tr>
<tr>
<td>HandleEvent(GameObject in_gameObject)</td>
<td>AkBank [inline]</td>
</tr>
<tr>
<td>loadAsynchronous</td>
<td>AkBank</td>
</tr>
<tr>
<td>saveDecodedBank</td>
<td>AkBank</td>
</tr>
<tr>
<td>UnloadBank(GameObject in_gameObject)</td>
<td>AkBank [inline]</td>
</tr>
<tr>
<td>unloadTriggerList</td>
<td>AkBank</td>
</tr>
</tbody>
</table>
override void AkBank::HandleEvent (GameObject)

Loads the SoundBank.
void AkBank::UnloadBank (GameObject in_)

Unloads a SoundBank.
AkBank

```cpp
string AkBank::bankName = ""
```

Name of the SoundBank, as specified in the Wwise project.
AkBank

`bool AkBank::loadAsynchronous = false`

Check this to load the SoundBank in the background. Be careful, if Events are triggered and the SoundBank hasn't finished loading, you'll have "Event not found" errors.
bool AkBank::decodeBank = false

Decode this SoundBank upon load.
AkBank

```cpp
bool AkBank::saveDecodedBank = false
```

Save the decoded SoundBank to disk for faster loads in the future.
AkBank

bankName
decodeBank
HandleEvent
loadAsynchronous
saveDecodedBank
UnloadBank
unloadTriggerList

List<int> AkBank::unloadTriggerList = new List<int>()

Reserved.
### AkCallbackManager

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PostCallbacks()</td>
<td></td>
</tr>
<tr>
<td>SetBGMCallback(BGMCallback in_CB, object in_cookie)</td>
<td></td>
</tr>
<tr>
<td>SetMonitoringCallback(AK.Monitor.ErrorLevel in_Level, MonitoringCallback in_CB)</td>
<td></td>
</tr>
</tbody>
</table>

Wwise Unity Integration Mon Jan 8 10:46:17 2018  doxygen 1.6.3
**LayerMask** AkObstructionOcclusion::LayerMask = -1 [inherited]

Indicates which layers act as obstructers/occluders.
float AkObstructionOcclusion::refreshInterval = 1 [inherited]

The number of seconds between obstruction/occlusion checks.
float AkObstructionOcclusion::fadeOutTime = 0.5f [inherited]

The number of seconds for fade ins and fade outs.
AkRoomPortalObstruction

float AkObstructionOcclusion::maxDistance = -1.0f [inherited]

- The maximum distance at which to perform obstruction/occlusion. A negative value will be interpreted as infinite distance.
AkEnvironment::AkEnvironment_CompareBySelectionAlgorithm

Wwise Unity Integration Mon Jan 8 10:46:17 2018  doxygen 1.6.3
### AkEnvironmentPortal

<table>
<thead>
<tr>
<th>axis</th>
<th>AkEnvironmentPortal</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetAuxSendValueForPosition(Vector3 in_position, int index)</td>
<td>AkEnvironmentPortal [inline]</td>
</tr>
</tbody>
</table>
Ax

GetAuxSendValueForPosition

float AkEnvironmentPortal::GetAuxSendValueForPosition

The axis used to find the contribution of each environment.
The array is already sorted by position. The first environment is on the negative side of the portal (opposite to the direction of the chosen axis). The second environment is on the positive side of the portal.
### AkEvent

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionOnEventType</td>
<td>AkEvent</td>
</tr>
<tr>
<td>curveInterpolation</td>
<td>AkEvent</td>
</tr>
<tr>
<td>enableActionOnEvent</td>
<td>AkEvent</td>
</tr>
<tr>
<td>eventId</td>
<td>AkEvent</td>
</tr>
<tr>
<td>soundEmitterObject</td>
<td>AkEvent</td>
</tr>
<tr>
<td>transitionDuration</td>
<td>AkEvent</td>
</tr>
</tbody>
</table>
AkEventCallbackMsg

info AkEventCallbackMsg
sender AkEventCallbackMsg
AkEventCallbackMsg

GameObject **AkEventCallbackMsg::sender**

AkSoundEngine.PostEvent callback flags. See the AkCallbackType enumeration for a list of all callbacks.
AkCallbackInfo AkEventCallbackMsg::info

GameObject from whom the callback function was called.
## AkGameObj

<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddListener</strong> (AkAudioListener listener)</td>
<td>AkGameObj [inline]</td>
</tr>
<tr>
<td><strong>GetForward</strong> ()</td>
<td>AkGameObj [inline, virtual]</td>
</tr>
<tr>
<td><strong>GetPosition</strong> ()</td>
<td>AkGameObj [inline, virtual]</td>
</tr>
<tr>
<td><strong>GetUpward</strong> ()</td>
<td>AkGameObj [inline, virtual]</td>
</tr>
<tr>
<td><strong>isEnvironmentAware</strong></td>
<td>AkGameObj</td>
</tr>
<tr>
<td><strong>m_positionOffsetData</strong></td>
<td>AkGameObj</td>
</tr>
<tr>
<td><strong>RemoveListener</strong> (AkAudioListener listener)</td>
<td>AkGameObj [inline]</td>
</tr>
</tbody>
</table>
bool AkGameObj::AddListener ( AkAudioListener listener )

Adds an \textbf{AkAudioListener} to the container of gameobject.

: 

\texttt{listener}

: 

Returns true if the listener was not previously, otherwise.
bool AkGameObj::RemoveListener ( AkAudioListener & listener )

Removes an AkAudioListener from the container of listeners listening to this gameobject.

: 

listener

: 

Returns true if the listener was previously in the list; false otherwise.
### AkGameObj

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener</td>
<td></td>
</tr>
<tr>
<td>GetForward</td>
<td></td>
</tr>
<tr>
<td><strong>GetPosition</strong></td>
<td>Gets the position including the position offset, if applyPositionOffset is enabled. User can also override this method to calculate an arbitrary position.</td>
</tr>
<tr>
<td>GetUpward</td>
<td></td>
</tr>
<tr>
<td>isEnvironmentAware</td>
<td></td>
</tr>
<tr>
<td>m_positionOffsetData</td>
<td></td>
</tr>
<tr>
<td>RemoveListener</td>
<td></td>
</tr>
</tbody>
</table>

```cpp
virtual Vector3 AkGameObj::GetPosition ()
```

The position.
<table>
<thead>
<tr>
<th>AkGameObj</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener</td>
</tr>
<tr>
<td>GetForward</td>
</tr>
<tr>
<td>GetPosition</td>
</tr>
<tr>
<td>GetUpward</td>
</tr>
<tr>
<td>isEnvironmentAware</td>
</tr>
<tr>
<td>m_positionOffsetData</td>
</tr>
<tr>
<td>RemoveListener</td>
</tr>
</tbody>
</table>

**virtual Vector3 AkGameObj::GetForward (**

Gets the orientation forward vector. User can method to calculate an arbitrary vector.

```
: The forward vector of orientation.
```
virtual Vector3 AkGameObj::GetUpward ()

Gets the orientation upward vector. User can method to calculate an arbitrary vector.

: The upward vector of orientation.
### AkGameObj

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener</td>
</tr>
<tr>
<td>GetForward</td>
</tr>
<tr>
<td>GetPosition</td>
</tr>
<tr>
<td>GetUpward</td>
</tr>
<tr>
<td>isEnvironmentAware</td>
</tr>
<tr>
<td>m_positionOffsetData</td>
</tr>
<tr>
<td>RemoveListener</td>
</tr>
</tbody>
</table>

#### AkGameObjPositionOffsetData AkGameObj

When not set to null, the position will be offset position by the Position Offset.
AkGameObj

- AddListener
- GetForward
- GetPosition
- GetUpward
- isEnvironmentAware
- m_positionOffsetData
- RemoveListener

**bool AkGameObj::isEnvironmentAware** =

Is this object affected by Environment changes? Set to false if not affected in order to save some useless calls. Default is true.
## AkInitializer

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>basePath</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>callbackManagerBufferSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>defaultPoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>diffractionFlags</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>engineLogging</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>language</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>lowerPoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>maxSoundPropagationDepth</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>memoryCutoffThreshold</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>monitorPoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>monitorQueuePoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>preparePoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>spatialAudioPoolSize</td>
<td>AkInitializer</td>
</tr>
<tr>
<td>streamingPoolSize</td>
<td>AkInitializer</td>
</tr>
</tbody>
</table>
AkInitializer

- basePath
- callbackManagerBufferSize
- defaultPoolSize
- diffractionFlags
- engineLogging
- **language**
- lowerPoolSize
- maxSoundPropagationDepth
- memoryCutoffThreshold
- monitorPoolSize
- monitorQueuePoolSize
- preparePoolSize
- spatialAudioPoolSize
- streamingPoolSize

```cpp
string AkInitializer::language = AkSoundEngineController.s_Language
```

Language sub-folder.
int AkInitializer::defaultPoolSize = ... 

Default Pool size. This contains the metadata for your audio project. Default size is 4 MB, but you should adjust for your needs.
### AkInitializer

- `basePath`
- `callbackManagerBufferSize`
- `defaultPoolSize`
- `diffractionFlags`
- `engineLogging`
- `language`
- `lowerPoolSize`
- `maxSoundPropagationDepth`
- `memoryCutoffThreshold`
- `monitorPoolSize`
- `monitorQueuePoolSize`
- `preparePoolSize`
- `spatialAudioPoolSize`
- `streamingPoolSize`

```cpp
int AkInitializer::lowerPoolSize = AkSoundEngineController.s_LowerPoolSize;
```

Lower Pool size. This contains the audio processing buffers and DSP data. Default size is 2 MB, but you should adjust for your needs.
AkInitializer

basePath
callbackManagerBufferSize
defaultPoolSize
diffractionFlags
defaultPoolSize
engineLogging
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

int AkInitializer::streamingPoolSize

Streaming Pool size. This contains the streaming buffers. Default size is 1 MB, but you should adjust for your needs.
AkInitializer

- basePath
- callbackManagerBufferSize
- defaultPoolSize
- diffractionFlags
- engineLogging
- language
- lowerPoolSize
- maxSoundPropagationDepth
- memoryCutoffThreshold
- monitorPoolSize
- monitorQueuePoolSize
- preparePoolSize
- spatialAudioPoolSize
- streamingPoolSize

```cpp
int AkInitializer::preparePoolSize = AkSoundEngineController::s_PreparePoolSize;
```

Prepare Pool size. This contains the banks loaded using PrepareBank (Banks decoded on load use this). Default size is 0 MB.
float AkInitializer::memoryCutoffThreshold

This setting will trigger the killing of sounds when the memory is reaching 95% of capacity. Lowest priority sounds are killed.
AkInitializer

basePath
callbackManagerBufferSize
defaultPoolSize
diffractionFlags
genLogEngine
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

int AkInitializer::monitorPoolSize =

AkInitializer

basePath
callbackManagerBufferSize
defaultPoolSize
diffractionFlags
gineLogging
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

int AkInitializer::monitorQueuePoolSize

Monitor Queue Pool size. Size of the monitor pool.
Size of the monitoring queue pool, in bytes. This parameter is not used in Release build.
AkInitializer

basePath
/callbackManagerBufferSize
defaultPoolSize
diffractionFlags
engineLogging
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

int AkInitializer::callbackManagerBufferSize

CallbackManager buffer size. The size is 4 KB, but you should increase this, if required.
### AkInitializer

- `basePath`
- `callbackManagerBufferSize`
- `defaultPoolSize`
- `diffractionFlags`
- `engineLogging`
- `language`
- `lowerPoolSize`
- `maxSoundPropagationDepth`
- `memoryCutoffThreshold`
- `monitorPoolSize`
- `monitorQueuePoolSize`
- `preparePoolSize`
- `spatialAudioPoolSize`
- `streamingPoolSize`

**int AkInitializer::spatialAudioPoolSize**

Spatial Audio Lower Pool size. Default size is 4 MB, but you should adjust for your needs.
AkInitializer

basePath
callbackManagerBufferSize
defaultPoolSize
diffractionFlags
ingenLogging
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

uint AkInitializer::maxSoundPropagationDepth

Spatial Audio Max Sound Propagation Depth. Maximum number of rooms that sound can propagate through; must be less than or equal to AK_MAX_SOUND_PROPAGATION_DEPTH.
## AkInitializer

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>basePath</td>
<td></td>
</tr>
<tr>
<td>callbackManagerBufferSize</td>
<td></td>
</tr>
<tr>
<td>defaultPoolSize</td>
<td></td>
</tr>
<tr>
<td>diffractionFlags</td>
<td>Enable or disable specific diffraction features. See AkDiffractionFlags.</td>
</tr>
<tr>
<td>engineLogging</td>
<td></td>
</tr>
<tr>
<td>language</td>
<td></td>
</tr>
<tr>
<td>lowerPoolSize</td>
<td></td>
</tr>
<tr>
<td>maxSoundPropagationDepth</td>
<td></td>
</tr>
<tr>
<td>memoryCutoffThreshold</td>
<td></td>
</tr>
<tr>
<td>monitorPoolSize</td>
<td></td>
</tr>
<tr>
<td>monitorQueuePoolSize</td>
<td></td>
</tr>
<tr>
<td>preparePoolSize</td>
<td></td>
</tr>
<tr>
<td>spatialAudioPoolSize</td>
<td></td>
</tr>
<tr>
<td>streamingPoolSize</td>
<td></td>
</tr>
</tbody>
</table>

**AkDiffractionFlags** AkInitializer::diffractionFlags

Enable or disable specific diffraction features.
AkInitializer

basePath
callbackManagerBufferSize
defaultPoolSize
diffractionFlags
engineLogging
language
lowerPoolSize
maxSoundPropagationDepth
memoryCutoffThreshold
monitorPoolSize
monitorQueuePoolSize
preparePoolSize
spatialAudioPoolSize
streamingPoolSize

bool AkInitializer::engineLogging =

Enable Wwise engine logging. Option: 

Wwise Unity IntegrationMon Jan 8 10:46:18 2018  doxygen 1.6.3
<table>
<thead>
<tr>
<th>bankName</th>
<th>AkMemBankLoader</th>
</tr>
</thead>
<tbody>
<tr>
<td>isLocalizedBank</td>
<td>AkMemBankLoader</td>
</tr>
<tr>
<td><strong>LoadLocalizedBank</strong>(string in_bankFilename)</td>
<td>AkMemBankLoader [inline]</td>
</tr>
<tr>
<td><strong>LoadNonLocalizedBank</strong>(string in_bankFilename)</td>
<td>AkMemBankLoader [inline]</td>
</tr>
</tbody>
</table>
### AkMemBankLoader

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bankName</td>
<td></td>
</tr>
<tr>
<td>isLocalizedBank</td>
<td></td>
</tr>
<tr>
<td>LoadLocalizedBank</td>
<td>Load a language-specific bank from WWW object.</td>
</tr>
<tr>
<td>LoadNonLocalizedBank</td>
<td></td>
</tr>
</tbody>
</table>

```cpp
void AkMemBankLoader::LoadLocalizedBank
```
<table>
<thead>
<tr>
<th>AkMemBankLoader</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bankName</strong></td>
</tr>
<tr>
<td><strong>isLocalizedBank</strong></td>
</tr>
<tr>
<td><strong>LoadLocalizedBank</strong></td>
</tr>
<tr>
<td><strong>LoadNonLocalizedBank</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>string AkMemBankLoader::bankName =</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of the bank to load.</td>
</tr>
</tbody>
</table>
AkMemBankLoader

<table>
<thead>
<tr>
<th>bankName</th>
<th>bool AkMemBankLoader::isLocalizedBank</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Is the bank localized (situated in the language specific folders).</td>
</tr>
</tbody>
</table>

Wwise Unity IntegrationMon Jan 8 10:46:18 2018
### AkRoom

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GetID()</code></td>
<td></td>
</tr>
<tr>
<td><code>priority</code></td>
<td></td>
</tr>
<tr>
<td><code>reverbAuxBus</code></td>
<td></td>
</tr>
<tr>
<td><code>reverbLevel</code></td>
<td></td>
</tr>
<tr>
<td><code>wallOcclusion</code></td>
<td></td>
</tr>
</tbody>
</table>
AkRoom

ulong AkRoom::GetID ( ) [inline]

Access the room's ID.
<table>
<thead>
<tr>
<th>AkRoom</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GetID</strong></td>
<td></td>
</tr>
<tr>
<td><strong>priority</strong></td>
<td></td>
</tr>
<tr>
<td><strong>reverbAuxBus</strong></td>
<td></td>
</tr>
<tr>
<td><strong>reverbLevel</strong></td>
<td></td>
</tr>
<tr>
<td><strong>wallOcclusion</strong></td>
<td></td>
</tr>
</tbody>
</table>

AK.Wwise.AuxBus AkRoom::reverbAuxBus

The reverb auxiliary bus.
float AkRoom::reverbLevel = 1

The reverb control value for the send to the reverb aux bus.
**float AkRoom::wallOcclusion = 1**

Occlusion level modeling transmission through walls.
int AkRoom::priority = 0

In cases where a game object is in an area with two rooms, the higher priority room will be chosen for AK::SpatialAudio::SetGameObjectInRoom(). The higher the priority number, the higher the priority of a room.
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AkRoomPortal</td>
<td>ClosePortal(Scalar in_gameObject)</td>
</tr>
<tr>
<td>GetID()</td>
<td>AkRoomPortal</td>
</tr>
<tr>
<td>HandleEvent(Scalar in_gameObject)</td>
<td>AkRoomPortal [inline]</td>
</tr>
<tr>
<td>MAX_ROOMS_PER_PORTAL</td>
<td>AkRoomPortal</td>
</tr>
<tr>
<td>rooms</td>
<td>AkRoomPortal</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>ClosePortal</td>
<td></td>
</tr>
<tr>
<td>GetID</td>
<td>Access the portal's ID.</td>
</tr>
<tr>
<td>HandleEvent</td>
<td></td>
</tr>
<tr>
<td>MAX_ROOMS_PER_PORTAL</td>
<td></td>
</tr>
<tr>
<td>rooms</td>
<td></td>
</tr>
</tbody>
</table>
AkRoomPortal

ClosePortal
GetID
**HandleEvent**
MAX_ROOMS_PER_PORTAL
rooms

override void AkRoomPortal::HandleEvent

Opens the portal on trigger event.
AkRoomPortal

- **ClosePortal**
- **GetID**
- **HandleEvent**
- **MAX_ROOMS_PER_PORTAL**
- **rooms**

```cpp
void AkRoomPortal::ClosePortal (GameObject) {
    Closes the portal on trigger event.
}
```
<table>
<thead>
<tr>
<th>AkRoomPortal</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClosePortal</td>
</tr>
<tr>
<td>GetID</td>
</tr>
<tr>
<td>HandleEvent</td>
</tr>
<tr>
<td>MAX_ROOMS_PER_PORTAL</td>
</tr>
<tr>
<td>rooms</td>
</tr>
</tbody>
</table>

```cpp
const int AkRoomPortal::MAX_ROOMS_PER_PORTAL

AkRoomPortals can only connect a maximum of 2 rooms.
```
### AkRoomPortal

<table>
<thead>
<tr>
<th>ClosePortal</th>
<th>GetID</th>
<th>HandleEvent</th>
<th>MAX_ROOMS_PER_PORTAL</th>
<th><code>AkRoom [] AkRoomPortal::rooms</code></th>
</tr>
</thead>
</table>

The front and back rooms connected by the portal. The first room is on the negative side of the portal (opposite to the direction of the local Z axis). The second room is on the positive side of the portal.
AkRoomPortalObstruction

AkRoomPortalObstruction

Wwise Unity Integration Mon Jan 8 10:46:18 2018  doxygen 1.6.3
# AkSpatialAudioEmitter

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>reflectAuxBus</td>
<td>AkSpatialAudioEmitter</td>
</tr>
<tr>
<td>reflectionMaxPathLength</td>
<td>AkSpatialAudioEmitter</td>
</tr>
<tr>
<td>reflectionsAuxBusGain</td>
<td>AkSpatialAudioEmitter</td>
</tr>
<tr>
<td>reflectionsOrder</td>
<td>AkSpatialAudioEmitter</td>
</tr>
<tr>
<td>roomReverbAuxBusGain</td>
<td>AkSpatialAudioEmitter</td>
</tr>
</tbody>
</table>
AkSpatialAudioEmitter

<table>
<thead>
<tr>
<th>reflectAuxBus</th>
<th>reflectionMaxPathLength</th>
<th>reflectionsAuxBusGain</th>
<th>reflectionsOrder</th>
<th>roomReverbAuxBusGain</th>
</tr>
</thead>
</table>

AK.Wwise.AuxBus AkSpatialAudioEmitter

The Auxiliary Bus with a Reflect plug-in Effect.
AkSpatialAudioEmitter

**uint AkSpatialAudioEmitter::reflectionsOrder**

The maximum number of reflections that will be processed for a sound path before it reaches the listener. Reflection processing grows exponentially with the order of reflections, so this number should be kept low. Valid range: 1-4.
AkSpatialAudioEmitter

<table>
<thead>
<tr>
<th>reflectAuxBus</th>
<th>reflectionMaxPathLength</th>
</tr>
</thead>
<tbody>
<tr>
<td>reflectionsAuxBusGain</td>
<td>reflectionsOrder</td>
</tr>
<tr>
<td>roomReverbAuxBusGain</td>
<td></td>
</tr>
</tbody>
</table>

float AkSpatialAudioEmitter::reflectionsAuxBusGain

The gain [0, 1] applied to the reflect auxiliary bus.
float AkSpatialAudioEmitter::reflectionMaxPathLength

The maximum path length a sound path can have from the emitter to the listener after reflecting on surfaces.
### AkSpatialAudioEmitter

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>reflectAuxBus</td>
<td></td>
</tr>
<tr>
<td>reflectionMaxPathLength</td>
<td></td>
</tr>
<tr>
<td>reflectionsAuxBusGain</td>
<td></td>
</tr>
<tr>
<td>reflectionsOrder</td>
<td></td>
</tr>
<tr>
<td>roomReverbAuxBusGain</td>
<td></td>
</tr>
</tbody>
</table>

**float AkSpatialAudioEmitter::roomReverbAuxBusGain**

Send gain (0.f-1.f) that is applied when sending to the aux bus associated with the room that the emitter is in.
AkState

AkState
groupID AkState
valueID  AkState

Wwise Unity Integration Mon Jan 8 10:46:18 2018  doxygen 1.6.3
int AkState::groupID

State Group ID, as defined in WwiseID.cs.
<table>
<thead>
<tr>
<th><strong>groupID</strong></th>
<th><strong>valueID</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>int <strong>AkState::valueID</strong></td>
</tr>
</tbody>
</table>

State Value ID, as defined in **WwiseID.cs**.
## AkSurfaceReflector

<table>
<thead>
<tr>
<th>Method</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AddGeometrySet</code></td>
<td>(AK.Wwise.AcousticTexture acousticTexture, MeshFilter meshFilter)</td>
<td>AkSurfaceReflector</td>
</tr>
<tr>
<td><code>RemoveGeometrySet</code></td>
<td>(MeshFilter meshFilter)</td>
<td>AkSurfaceReflector</td>
</tr>
</tbody>
</table>

Wwise Unity IntegrationMon Jan 8 10:46:18 2018  

[doxygen] 1.6.3
**AkSurfaceReflector**

```cpp
static void AkSurfaceReflector::AddGeometrySet(AcousticTexture acousticTexture, const acr::MeshFilter* meshFilter);
```

Sends the mesh filter's triangles and their acoustic texture to Spatial Audio.

- `acousticTexture`
- `meshFilter`
<table>
<thead>
<tr>
<th>AcousticTexture</th>
<th>AddGeometrySet</th>
<th>RemoveGeometrySet</th>
</tr>
</thead>
</table>

**static void AkSurfaceReflector::RemoveGeometrySet**

Remove the corresponding mesh filter's geometry:

```
meshFilter
```
AkSurfaceReflector

AK.Wwise.AcousticTexture AkSurfaceReflector::AcousticTexture

All triangles of the component's mesh will be applied with this texture. The texture will change the filter parameters of the sound reflected from this component.
AkSwitch

AkSwitch

groupId AkSwitch

valueID AkSwitch
## AkSwitch

<table>
<thead>
<tr>
<th><strong>groupId</strong></th>
<th><strong>valueID</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>int AkSwitch::groupId</strong></td>
<td>Switch Group ID, as defined in WwiseID.cs.</td>
</tr>
<tr>
<td><strong>groupBy</strong></td>
<td><strong>valueId</strong></td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

**Switch Value ID**, as defined in WwiseID.cs.
AkTriggerBase

`Trigger(GameObject in_gameObject)`

triggerDelegate

AkTriggerBase
delegate void AkTriggerBase::Trigger ( GameObject in_gameObject )

Delegate declaration for all Wwise Triggers.

in_gameObject is used to pass objects when Colliders are used. Some components have the option "Use other object", this is
AkTriggerBase

**Trigger** `AkTriggerBase::triggerDelegate = null`

All components reacting to the trigger will be registered in this delegate.
- AddGeometrySet() : AkSurfaceReflector
- AddListener() : AkGameObj
- ClosePortal() : AkRoomPortal
- ExecuteAction() : AK::Wwise::Event
- GetAuxSendValueForPosition() : AkEnvironmentPortal
- GetForward() : AkGameObj
- GetID() : AkRoomPortal, AkRoom
- GetPosition() : AkGameObj
- GetUpward() : AkGameObj
- HandleEvent() : AkBank, AkRoomPortal
- LoadLocalizedBank() : AkMemBankLoader
- LoadNonLocalizedBank() : AkMemBankLoader
- Post() : AK::Wwise::Event
- PostCallbacks() : AkCallbackManager
- PostMIDI() : AK::Wwise::Event
- RemoveGeometrySet() : AkSurfaceReflector
- RemoveListener() : AkGameObj
- SetBGMCallback() : AkCallbackManager
- SetMonitoringCallback() : AkCallbackManager
- StopMIDI() : AK::Wwise::Event
- Trigger() : AkTriggerBase
- UnloadBank() : AkBank
- a -
  - AcousticTexture: AkSurfaceReflector
  - actionOnEventType: AkEvent
  - axis: AkEnvironmentPortal

- b -
  - bankName: AkBank, AkMemBankLoader
  - basePath: AkInitializer

- c -
  - callbackManagerBufferSize: AkInitializer
  - curveInterpolation: AkEvent

- d -
  - decodeBank: AkBank
  - defaultPoolSize: AkInitializer
  - diffractionFlags: AkInitializer

- e -
  - enableActionOnEvent: AkEvent
  - engineLogging: AkInitializer
  - eventID: AkEvent

- g -
- **i** -
  - info : AkEventCallbackMsg
  - isEnvironmentAware : AkGameObj
  - isLocalizedBank : AkMemBankLoader

- **l** -
  - language : AkInitializer
  - loadAsynchronous : AkBank
  - lowerPoolSize : AkInitializer

- **m** -
  - m_positionOffsetData : AkGameObj
  - MAX_ROOMS_PER_PORTAL : AkRoomPortal
  - maxSoundPropagationDepth : AkInitializer
  - memoryCutoffThreshold : AkInitializer
  - monitorPoolSize : AkInitializer
  - monitorQueuePoolSize : AkInitializer

- **p** -
  - preparePoolSize : AkInitializer
  - priority : AkRoom

- **r** -
  - reflectAuxBus : AkSpatialAudioEmitter
  - reflectionMaxPathLength : AkSpatialAudioEmitter
  - reflectionsAuxBusGain : AkSpatialAudioEmitter
  - reflectionsOrder : AkSpatialAudioEmitter
  - reverbAuxBus : AkRoom
  - reverbLevel : AkRoom
  - roomReverbAuxBusGain : AkSpatialAudioEmitter
  - rooms : AkRoomPortal
- s -
  - saveDecodedBank : AkBank
  - sender : AkEventCallbackMsg
  - soundEmitterObject : AkEvent
  - spatialAudioPoolSize : AkInitializer
  - streamingPoolSize : AkInitializer

- t -
  - transitionDuration : AkEvent
  - triggerDelegate : AkTriggerBase

- u -
  - unloadTriggerList : AkBank

- v -
  - valueID : AkState, AkSwitch

- w -
  - wallOcclusion : AkRoom