Audio in Unity

An introduction to audio in Unity

Programming – Game Development Foundations

Last modified 03/02/2016 by Richard Taylor

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Contents

Audio in games

- Audio is an important aspect of many games
  - User feedback
  - Immersion
  - Mood setting

- Audio elements
  - Music
  - Sound effects

2D and 3D sound

- 2D sounds
  - Do not have position
  - Volume controlled directly
- 3D sounds
  - Requires that a “listener” is defined with a position and orientation
  - Sounds are “spatialised” by modifying their volume, pitch, and other properties based on their position and movement relative to the listener
  - Thankfully, Unity looks after most of this complexity for us!

Sound effects and music

- The engine or audio system usually doesn’t make a distinction between sound effects and music

- We often want to treat them differently in our games
  - Organisation
  - Optimisation

Sound effects

- Effects related to an event usually play once at the time and place of the event
  - Eg: A “hit” sound effect when the player successfully strikes an opponent
- Effects related to a place in a scene often loop. May be called “ambient sounds”.
  - Eg: water running into a drain
- Some effects don’t come from the scene, so are played in 2D
  - Eg: button clicks in a game’s menu
- Sound effects (“SFX”)
  - Usually short, so played directly from memory
  - Often related to a position in a scene and played in 3D
  - Typically mono

Music and sound effects

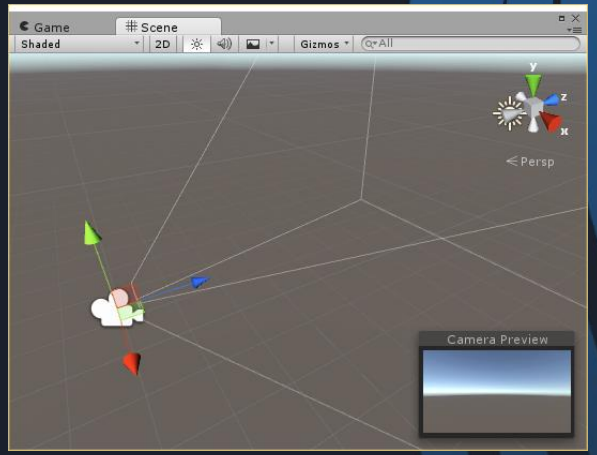
- Often plays in the background, repeating for the duration of a scene
- Does not move around the player
- Changing track or volume is useful for feedback or mood setting
  - Eg: Switching between an “exploration” to a “combat” track both helps to inform the player of the situation and increase or decrease tension.
- Music
  - Often long, so typically streamed from disk
  - Not a part of the scene, so usually played in 2D
  - Typically Stereo or multi-channel

Audio in Unity

- Unity provides functionality for both 2D and 3D audio

- Relevant components and resources include:
  - Audio Listener
  - Audio Source
  - Audio Clip
  - Audio Mixer

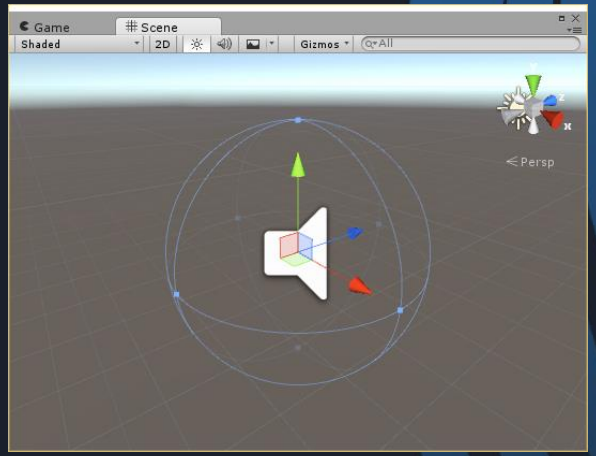
Audio Listener component

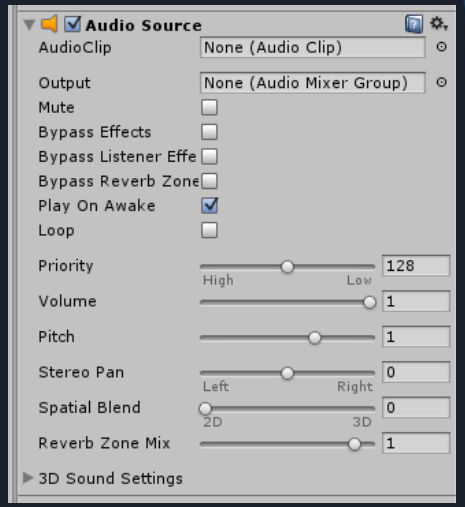


- Represents the object in the scene which can “hear” sounds
- Gets position and orientation from its Transform
- Usually attached to the main camera, so that audio matches rendered image
- You can only have one Audio Listener in the scene at once!
  - If you make a second camera you usually need to remove its Audio Listener.
- Visible in the Inspector, but has no properties to change

Audio Source component

- Represents the object in the scene which makes sound
  - Can make 2D or 3D sounds
- Gets position from its Transform if needed
- Often attached directly to the GameObject that makes the sound
- You can have many Audio Sources in a scene

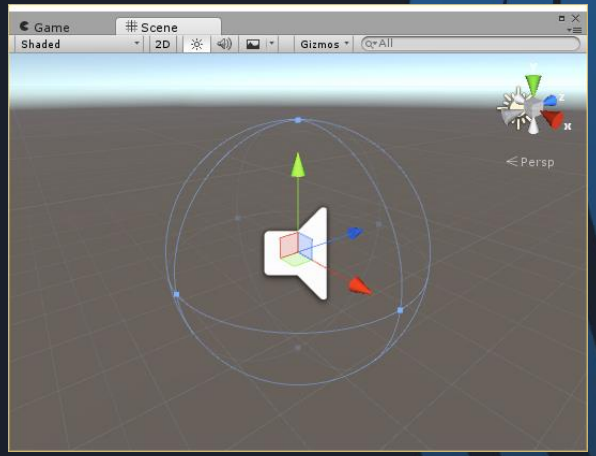


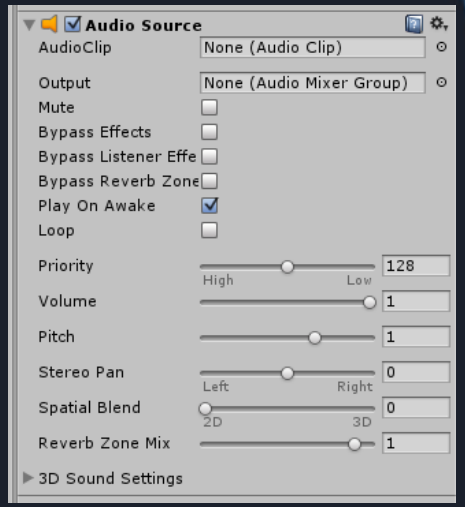


aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Audio Source component

- Many Inspector properties. Take note of:
  - Audio Clip sets the clip to play
  - Play on Awake sets whether the source plays automatically on creation
  - Loop determines if the clip is played once or repeats until told to stop
  - Spatial Blend determines whether the sound is played in 2D or 3D
  - 3D Sound Settings has Min and Max Distance settings (the Audio Listener can't hear a source from outside of its Max Distance)

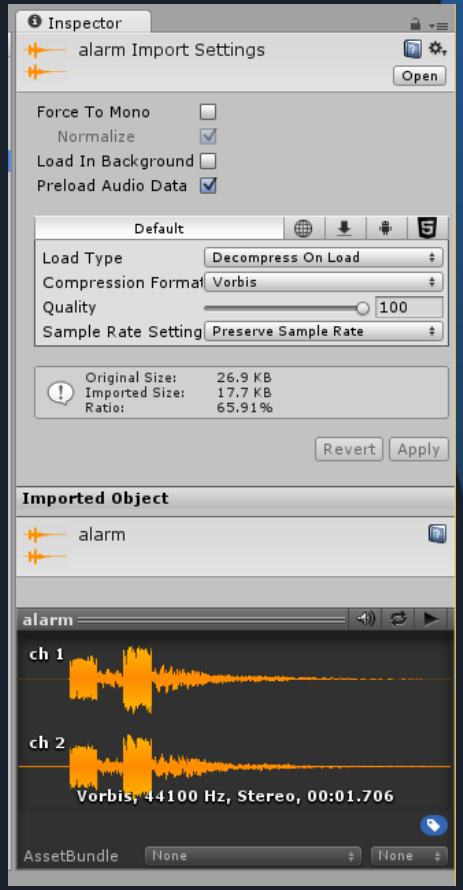




aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Audio Clip resource

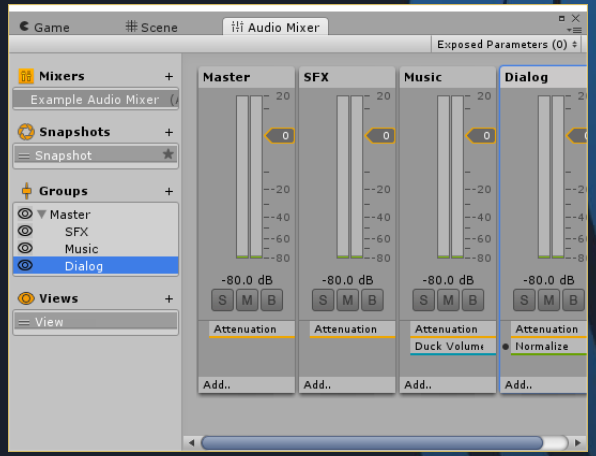
- Represents a sound which can be played by an Audio Source
  - One Audio Clip can be used by many Audio Sources
- Import Settings for each platform
  - Load type: streamed, compressed, decompressed
  - Audio compression and quality
- Preview window shows visual waveform and allows playback
  - Useful for testing looping

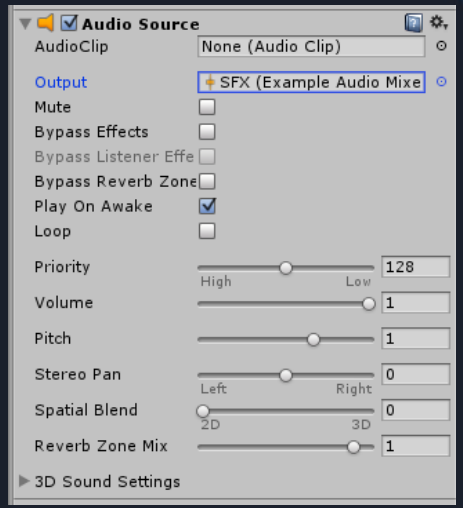


aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Audio Mixer resource

- Represents a configuration used for playing audio
- Audio Sources can be assigned to “Groups”
  - See the “Output” property in the Audio Source’s Inspector
- Volume and effects can then be applied to Groups instead of individual Audio Sources
- Audio Mixer configurations can be saved to “Snapshots”, which can then be switched between at runtime
  - Eg: SFX and Music volume might be reduced during conversation scenes





aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

Playing via script

```
public class PlayAudio : MonoBehaviour {

    public AudioSource m_source;

    void Update () {
        if (Input.GetKeyDown(KeyCode.Space)) {
            // Play the AudioSource with its current settings
            m_source.Play();
        }
    }

}
```

- Note that if you are playing audio via script you probably want to untick “Play on Awake” in the Audio Source

Playing via script

```
public class PlayAudio : MonoBehaviour {

    public AudioSource m_source;
    public AudioClip m_clip;
    public float m_volume = 1.0f;

    void Update () {
        if (Input.GetKeyDown(KeyCode.Space)) {
            // Set a clip
            m_source.clip = m_clip;

            // Change the volume
            m_source.volume = m_volume;

            // Play the audio source
            m_source.Play();
        }
    }

}
```

Summary

- Audio is often important in games
- Audio may be played directly as “2D” sounds, or spatialised within a scene as “3D” sounds
- It is often useful to treat music and sound effects differently, for organisation and optimisation
- Unity provides Audio Listener, Audio Source, Audio Clip and Audio Mixer components and resources for implementing sound in our games
- Audio Sources can be controlled via script

References

- *Unity Manual*, Audio chapter, Unity Technologies, accessed 03/02/2016
  - http://docs.unity3d.com/Manual/Audio.html

- *Unity Scripting API*, AudioSource, Unity Technologies, accessed 03/02/2016
  - http://docs.unity3d.com/ScriptReference/AudioSource.html