

Native SoundManager

v. 0.4

1. Overview

SoundManager is a simple sound asset management class which allows sound assets to be easily replaced without recompilation. SoundManager is driven by a JSON file in which sounds are mapped as key-value pairs, where a value is the actual path to the wav file. For example:

```
"sv_touch_active" : "sv_touch_active.wav"
```

In code, we use the key to play the sound, which SoundManager then resolves to the actual asset. For example:

```
app->PlaySound( "sv_touch_active" );
```

The string `"sv_touch_active"` is first passed to SoundManager, which resolves it to an absolute path, as long as the key was found during initialization.

These two paths indicate whether the sound file is in the res/raw folder of VrLib (e.g., for sounds that may be played from any app, such as default sounds or Universal Menu sounds) or the assets folder of a specific app:

```
"res/raw/ sv_touch_active.wav"
```

or

```
"assets/ sv_touch_active.wav"
```

2. Implementation details

If SoundManager fails to resolve the passed-in string within the `App->PlaySound` function, the string is passed to `playSoundPoolSound` in the `VrActivity` class in Java. In `playSoundPoolSound`, we first try to play the passed-in sound from `res/raw`, and if that fails, from the current assets folder. If that also fails, we attempt to play it as an absolute path. The latter allows for sounds to be played from the phone's internal memory or SD card.

The JSON file loaded by SoundManager determines which assets are used with the following scheme:

1. Try to load `sounds_assets.json` in the *Oculus* folder on the sdcard:
`sdcard/Oculus/sounds_assets.json`
2. If we fail to find the above file, we then load the following two files in this order:
`res/raw/sounds_assets.json`
`assets/sounds_assets.json`

The loading of the `sounds_assets.json` in the first case allows for a definition file and sound assets to be placed on the SD card in the *Oculus* folder during sound development. The sounds may be placed into folders if desired, as long as the relative path is included in the definition.

For example, if we define the following in `sdcard/Oculus/sounds_assets.json`:

```
"sv_touch_active" : "SoundDev/my_new_sound.wav"
```

we would replace all instances of that sound being played with our new sound within the `SoundDev` folder.

The loading of the two asset definition files in the second step allows for overriding the `VrLib` sound definitions, including disabling sounds by redefining their asset as the empty string. For example:

```
"sv_touch_active" : ""
```

The above key-value pair, if defined in an app's `sounds_assets.json` (placed in its asset folder), will disable that sound completely, even though it is still played by `VrLib` code.

Last Update: Nov 10, 2014

OCULUS VR is a registered trademark of Oculus VR, LLC. (C) Oculus VR, LLC. All rights reserved. All other trademarks are the property of their respective owners.