# Sound xR Core for Unity
# Release Note / Installation Manual

## Introduction

- Sound xR® is a spatial audio solution for earphones/headphones that has been developed based on YAMAHA's ViReal™ total solution technology for immersive sound.
- Sound xR Core enables you to design an interactive and high-quality sound image localization and sound space in the application area of xReality, which is a generic term for VR/AR/MR/SR.
- Sound xR Core for Unity (hereinafter referred to as "this package") provides the Plug-in for developing Windows, macOS, Android, and iOS applications under the Unity environment, in the Unity Package format.

## Before using this package

- To use this package, you need to obtain a license from YAMAHA Corporation.
- For the terms of use of this package, refer to such as "Terms of Use" and "License Agreement" that you agree to when obtaining a license from YAMAHA Corporation.
- It is prohibited to use the contents of this package beyond the scope permitted by YAMAHA Corporation.

## Assumed environment

The usage environment assumed for this package is as follows:

- Platform
  - Windows10/11 (64bit)
  - macOS 11.6.2 or later (Intel/Apple silicon)
  - Android 10 or later (ARM64)
  - iOS 14.2 or later
- Unity
  - Unity 2020, 2021
- Headphone/earphone output
  (You cannot obtain the original sense of localization if you play back using speakers.)

## Functions provided

- Object-based spatial audio processing function
  - mono in / stereo out
  - Sampling frequency: only 48 kHz is supported
- High Order Ambisonics (HOA) playback function
  - Supports 1st to 5th order Ambisonics.
  - Supports B-Format (AmbiX) ambisonic format. File format: wav file
  - Sampling frequency: only 48 kHz is supported
- This package supports Assembly Definition.

## Setup procedure

1. If Unity is not installed, access the Unity official website and install it.
2. Create a new Unity project or open an existing project.
3. Appropriately set Platform in Build Settings and change the target Platform in Switch Platform.
4. From the UnityEditor menu, select 'Assets' > 'Import Package' > 'Custom Package.....'
5. When the Import Package... dialog box opens, select SoundxRCore.unitypackage.
6. When the Import Unity Package dialog box opens, check the contents to import and press the Import button.
7. When the import has completed, the SoundXR directory is added directly under Assets in the Project tree.
8. Read the license key provided by selecting 'Sound xR' > 'Activation' > 'Activate License Key' and complete the activation only once at the first startup. Setup is now complete.

## Configuration

```
SoundxRCore.unitypackage
    ├─ Assets
        ├─SoundXR
            ├─ Common            : Common use program
            ├─ Docs
            |   ├─ Readme_jp.pdf : This file
            ├─ Effect            : Effect processing program
            |   ├─ Spatializer   : Spatial audio processing
            ├─ Examples          : Samples scenes, Scripts
```

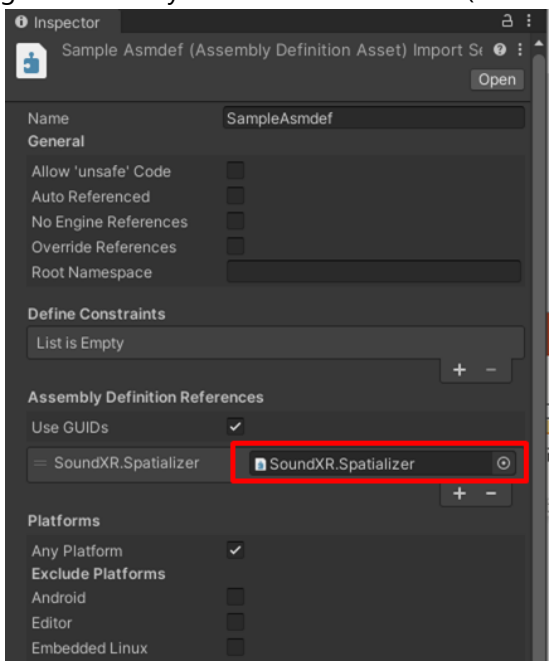
# How to use (convert audio source into spatialized audio)

Sound xR Core Unity Plug-in can convert audio source to be spatialized.

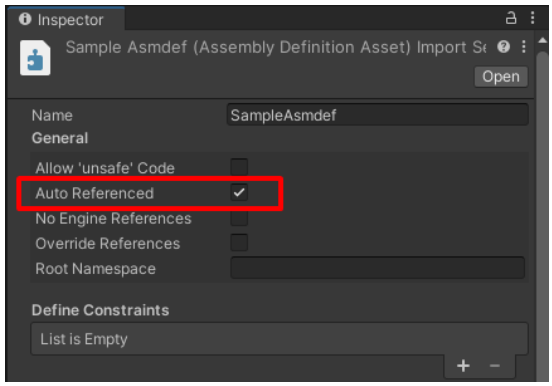
1. Activate Sound xR Plug-in.
    - 'Select Edit' > 'Project Settings' > 'Audio' > 'Spatializer Plugin' > 'Sound xR Core'.

2. Set up of Assembly Definition
    - Create an .asmdef file in the appropriate folder with 'Create' > 'Assembly Definition'.
    - Set 'SoundXR.Spatilizer' in the 'Assembly Definition References' (recommended) OR turn on the 'Auto Referenced' check box. (See below)

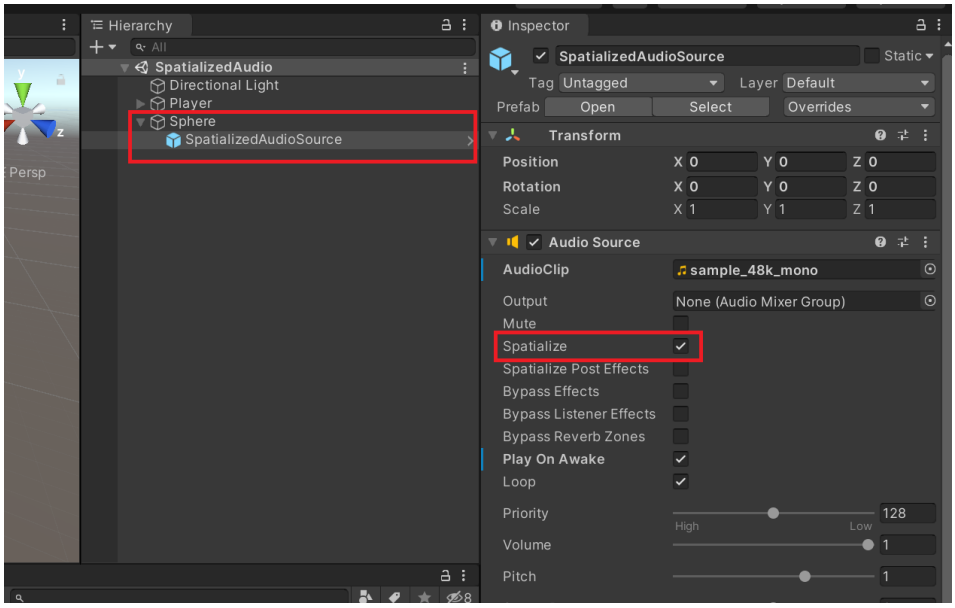
ex. setting of 'Assembly Definition References' (recommended)




ex. setting of 'Auto Referenced'



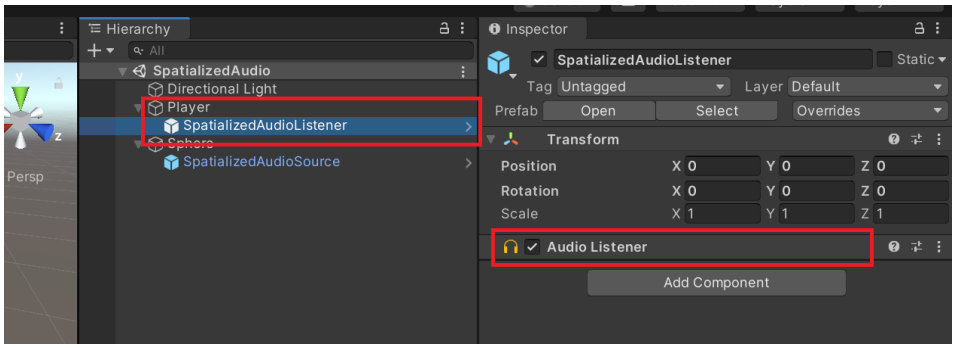

3. Select the Spatialize check box for each audio source (AudioSource) to be converted to spatial audio.




In this image, the audio source of the Sphere object is converted into spatialized audio.

4. For each AudioSource, add SpatializedAudioSource component from Add Component by selecting Sound xR > Effect > Spatializer.

5. Set the listening position (AudioListener).

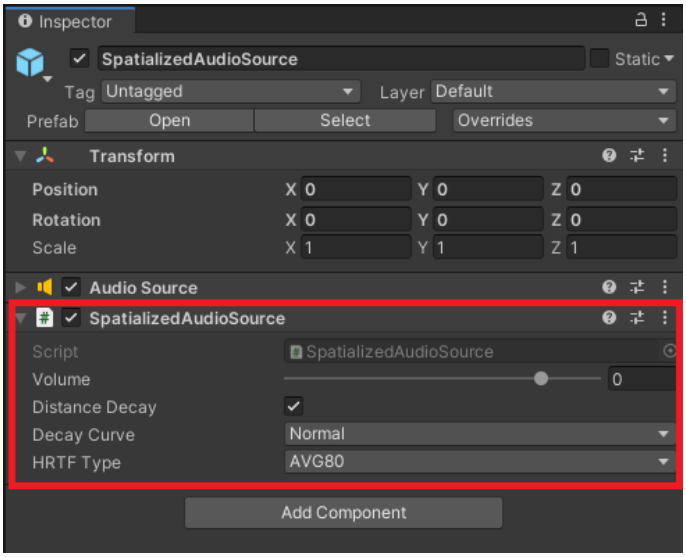



In this image, Audio Listener is set using Player as the listening position.
(The listener-side does not need to be specially set.)

## Parameter operation

You can operate the following parameters from the Inspector window.

| Name | Description | Value |
|---|---|---|
| Volume | Set the audio source output volume. The default is 0 dB. | -96 to 0 dB |
| Distance Decay | Set whether or not to decay the audio source volume by distance. This check box is not selected by default. If the Unity standard's distance decay function is also used, both distance decay functions are applied. | ON/OFF |
| Decay Curve | Set the degree of distance decay for the audio source. The default is Normal. At the same distance, the amount of decay decreases with Slow and increases with Fast. | Slow/Normal/Fast |
| HRTF Type | Select an HRTF characteristic set (described below). The default is AVG80. | AVG80/TC4 |



## About HRTF characteristic sets

You can select the HRTF (head-related transfer function) characteristic set used for spatial audio processing from the following two types.

| **HRTF Type** | **Feature** |
|---|---|
| AVG80 | A natural sense of localization is obtained in all directions. Widely suitable for AR/MR applications. |
| TC4 | When the audio object is in front, there is no spatial effect, and when the audio object is away from the front, the original sense of localization is applied.<br>This feature, which is used for a VR application including video, is effective when you want to suppress changes in the sound quality of the audio object located in front. |

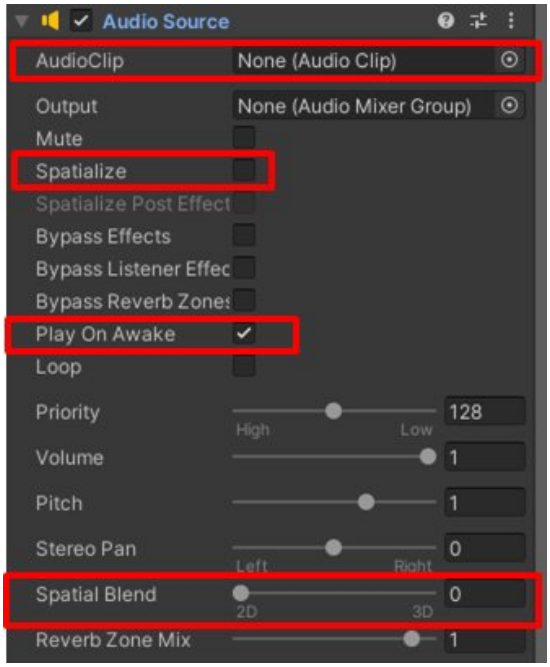
## Sample scene - Spatial audio function

- This package contains a spatial audio playback sample scene. Use it as a reference.
- From the project window, view Assets\SoundXR\Examples\SpatializedAudio\Scenes\SpatializedAudio .
- When you start the preview, the audio object will start moving and playing sample voice.

# How to use (HOA playback function)

Sound xR Core Unity Plug-in can play an HOA file.

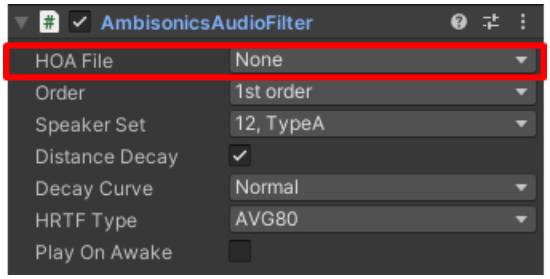
1. Activate Sound xR Plug-in.
    - Select Edit > Project Settings > Audio > Spatializer Plugin > Sound xR Core.
2. Set up of Assembly Definition
    - Create an .asmdef file in the appropriate folder with 'Create' > 'Assembly Definition'.
    - Set 'SoundXR.Spatilizer' in the 'Assembly Definition References' (recommended) OR turn on the 'Auto Referenced' check box.
3. Audio source-side settings (1)
    - Prepare an object to be used as the HOA audio source, assign AudioSource to it, and make the following settings.
        - Select the Play On Awake check box.
        - Set Spatial Blend of AudioSource to 2D.
        - You do not need to specify AudioClip of AudioSource and select the Spatialize check box.




4. Audio source-side settings (2)

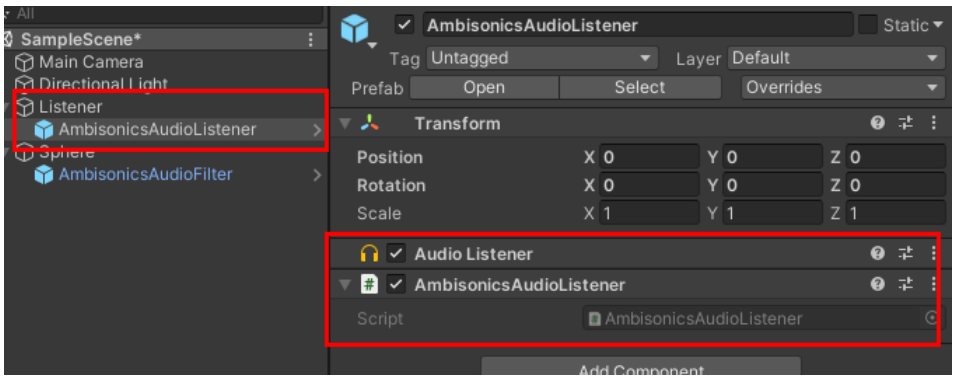
- From 'Add Component', add AmbisonicsAudioFilter component to the object to be used as the HOA audio source by selecting Sound xR > Effect > Spatializer.
- Place the HOA file in the Assets\StreamingAssets folder. If the StreamingAssets folder does not exist, please create it.
- From Inspector, specify the HOA file you want to play in 'HOA File' of AmbisonicsAudioFilter.
- For other settings, set the necessary items by referring to "Parameter operation" described below.

---




4. Listener-side settings

- Prepare an object to be used in the listener side. At this time, be sure to observe the following items.
  - Be sure to select different objects for the HOA audio source object and the listener-side object. If you use the same object, the audio source also rotates when the listening direction rotates, which results in the sound not changing even if the listening direction is changed.
  - The listener-side object's Position must be the same as the HOA audio source object's Position. If either one moves, be sure that both move together. Abnormal playback sound will be generated.
- Assign AudioListener to the listener-side object.
  - (No particular setting is required for AudioListener.)
- From Add Component, add AmbisonicsAudioListener component to the listener-side object by selecting Sound xR > Effect > Spatializer.
  - (No particular setting is required for AmbisonicsAudioListener.)



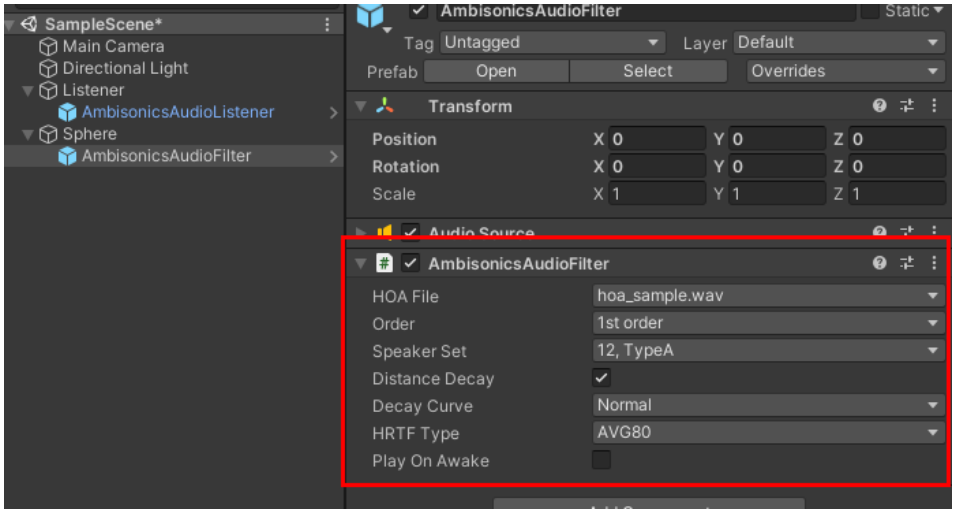

In this image, Audio Listener is set using Listener as the listening position.

## Parameter operation

You can operate the following parameters from the Inspector window.

| Name | Description | Value |
|---|---|---|
| HOA File | Specify the HOA file you want to play.<br>The files placed in Assets\StreamingAssets are displayed in a list format. (Create the StreamingAssets folder if it does not exist.) | |
| Order | Set the HOA order. The default is 1st order. The higher the order is, the more detailed sound image localization can be expressed, however, the computational load increases. If the order set here is different from the order of the HOA file actually loaded, the smaller order is applied. | 1st/2nd/3rd/4th/5th order |

| Name | Description | Value |
| --- | --- | --- |
| Speaker Set | Set the number of "virtual speakers" used for internal processing when playing back in the Ambisonics format. The default is 12ch: 12, TypeA (Only TypeA is currently supported as Type). The more virtual speakers there are, the more detailed the sound image localization can be expressed, however, the computational load increases. If the order of the HOA file to be played is low, even if there is a large number of virtual speakers, no effect is obtained. | 12/20/32/42, TypeA |
| Volume | Set the audio source output volume. The default is 0 dB. | -96 to 0 dB |
| Distance Decay | Set whether or not to decay the audio source by distance. This check box is selected by default. When this check box is selected, the volume is decayed depending on the Scale of the object to which the AmbisonicsAudioFilter.cs component is added. The higher the Scale is, the greater the volume decay becomes. | ON/OFF |
| Decay Curve | Set the degree of distance decay for the audio source. The default is Normal. At the same Scale, the amount of decay decreases with Slow and increases with Fast. | Slow/Normal/Fast |
| HRTF Type | Select an HRTF characteristic set. The default is AVG80. | AVG80/TC4 |
| Play On Awake | When this check box is selected, the HOA audio source starts playing at the same time the application starts. This check box is not selected by default. | ON/OFF |

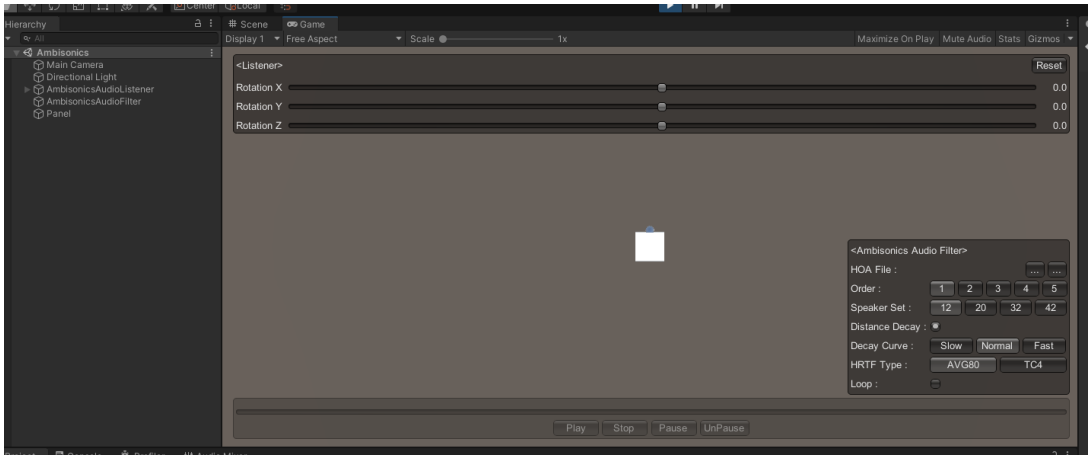


### Tips

- If the Play On Awake check box of AmbisonicsAudioListener.cs is not selected, Play() of AmbisonicsAudioListener.cs is called to start a playback.
- The Volume/Loop settings of the AudioSource with the AmbisonicsAudioFileter.cs component added are applied to each channel of the virtual speakers.
- Distance Decay, Decay Curve, and HRTF Type are the same as of the SpatializedAudioSource.cs component. These settings are applied to the audio output from each channel of the virtual speaker.
- The HOA file is expanded to the virtual speakers focusing around the position of the object to which AudioSource is assigned, and played. Therefore, place AudioListener and AudioSource at the same Position.
    - It is recommended that you fix the Position and change the AudioListener direction by Rotation for trial listening.

## Sample scene - HOA playback

- This package contains a HOA playback control sample scene. Use it as a reference when implementing HOA file playback control.
- From the Project window, view Assets\SoundXR\Effect\Spatializer\Examples\Ambisonics\Scenes\Ambisonics.
- This package does not contain a HOA file.




This is the screen that previews the sample scene.

# File import function

- This package contains a sample of a general-purpose file import function (plug-in). This function is provided as a sample; therefore, its operation is not guaranteed.
- It supports the function to read files under Assets\StreamingAssets and the function to read files outside the application.
    - It is also used in the sample scene Ambisonics described above, so use it as a reference.
- It is necessary to either add 'SoundXR.Common' in the 'Assembly Definition References' (recommended) OR turn on the 'Auto Referenced' check box.
- About a use on Android

    - You need to move AndroidManifest.xml and AndroidManifest.xml.meta included in Assets\SoundXR\Common\FileSystem\Plugins\Android of unitypackage of the imported Sound xR to Assets\Plugins\Android (create this folder if it does not exist).

    - If you already have AndroidManifest.xml, add the following content into the <application> tag.

      ```
      <activity android:name="com.yamaha.soundxr.ExActivity"
                  android:theme="@style/UnityThemeSelector">
          <intent-filter>
              <action android:name="android.intent.action.MAIN" />
              <category android:name="android.intent.category.LAUNCHER" />
          </intent-filter>
          <meta-data android:name="unityplayer.UnityActivity"
      android:value="true" />
      </activity>
      ```

    - You need to select Project Settings > Player > Other Setting and set the Write Permission setting to "External(SDCard)" from the Unity Editor settings.

        - When the app starts, a dialog box is displayed requesting the authority to access files in the device.
        - This authority is required to display the file selection screen and read the selected file.

## Notes

- **[Important] Do not disable the AudioSource which is attached a SpatializedAudioSource.**
  - If you have to disable the AudioSource, please disable it for each GameObject.
  - If you have to stop the sound, please use Mute(), Stop() or Pause() function.
- **[Important] It has been reported that 'File import function' does not work correctly when the apk built by Unity2021 is installed on Android12.**
  - Therefore, when trying out the file import function in Unity2021, avoid trial use of an Android 12 terminal.

## Customer support

If you have any problems or questions, send an e-mail to the following address.

| Department name | e-mail |
|---|---|
| YAMAHA Corporation<br>Cloud Business Department<br>Sound xR Support Section | soundxr-support-ML@music.yamaha.com |

## Release history

| Version | Release date | Modification |
|---|---|---|
| 1.3.4 | May 1st, 2023 | ・Changed the internal processing (No function changed) |
| 1.3.3 | Mar. 14th, 2023 | ・Fixed a bug in the HOA playback function |
| 1.3.2 | Jan. 26th, 2023 | ・Added Assembly Definition File |
| 1.3.1 | Dec. 5th, 2022 | ・Moved the sample scene to the 'SoundXR\Example' folder<br>・Fixed minor bugs |
| 1.3.0 | Jun. 16th, 2022 | ・Added HOA playback function |
| 1.2.1 | | ・Fixed bug in 'SpatializedAudioSource.cs' |
| 1.1.0 | | ・Made the HRTF dataset swichable (AVG80 / TC4) |

---



Manual Version: 2023.04.24