**OGRE3D – Son**

<http://ogre3d.fr/poster-0-120-919-bibliotheque-sonore.html>

**OpenAL -> OgreAL**

<http://loulou.developpez.com/tutoriels/openal/premiers-pas/>

<http://fr.wikipedia.org/wiki/OpenAL>

## Principe[[modifier](http://fr.wikipedia.org/w/index.php?title=OpenAL&action=edit&section=3)]

La bibliothèque permet de modéliser un ensemble de sources sonores se déplaçant dans un espace en 3D, et un auditeur dans ce même espace. Des modifications sur les sources peuvent être ajoutées comme de l'[écho](http://fr.wikipedia.org/wiki/%C3%89cho_(acoustique)) ou d'autres [effets sonores](http://fr.wikipedia.org/wiki/Bruitage). La bibliothèque peut alors calculer le rendu final, en tirant parti des accélérations des cartes audio récentes.

Les spécifications sont séparées en deux, d'une part le noyau de l'interface (core API) constitué des fonctions accessibles de OpenAL, d'autre part le ALC API (pour *Audio Library contexts*) qui fournit des fonctions bas niveau. L'ALC permet ainsi de séparer les fonctions de l'API du dispositif qui réalise le rendu final, que ce soit une carte électronique, un [démon](http://fr.wikipedia.org/wiki/Daemon_(informatique)), un [driver](http://fr.wikipedia.org/wiki/Pilote_informatique), ou même un[service](http://fr.wikipedia.org/wiki/Daemon_(informatique)). Il facilite l'utilisation de différents dispositifs coexistant sur un même système, en les énumérant et en se les attribuant, permet de connaître les extensions présentes sur le système, ainsi qu'un moyen d'accéder à l'entrée microphone si elle existe.

Une partie des sources est maintenue par le constructeur [Creative Labs](http://fr.wikipedia.org/wiki/Creative_Labs), mais la bibliothèque est prévue pour accueillir les spécificités propres de chacun des autres constructeurs et de leurs différentes [cartes audio](http://fr.wikipedia.org/wiki/Carte_audio), par le biais d'extensions à l'API. Après un certain temps, ces fonctions peuvent même devenir des fonctions standard de l'API. Entre les deux les extensions peuvent être promues ARB (Architecture Review Board) qui permet de savoir lesquelles seront maintenues dans les versions futures de la bibliothèque.

**DirectSound**

Fait partie de DirectX

DirectSound is a software component of the Microsoft DirectX library for the Windows operating system.

**EAX**

<http://fr.wikipedia.org/wiki/Environmental_audio_extensions>

EAX a supplanté l'alternative [A3D](http://fr.wikipedia.org/w/index.php?title=A3D&action=edit&redlink=1) (Aureal 3-Dimensional) en 2001.

L'objectif d'EAX est d'ajouter de l'ambiance dans les jeux vidéo en simulant l'environnement audio du monde réel.

L'[API](http://fr.wikipedia.org/wiki/Interface_de_programmation) EAX est une bibliothèque étendant l'API [DirectSound3D](http://fr.wikipedia.org/wiki/DirectX) de Microsoft, elle ajoute au système de positionnement audio, les effets d'environnement. Les développeurs utilisant EAX choisissent un environnement (salle de bain, théâtre ...) et la carte son utilisera le filtre correspondant.

EAX ne s'occupe pas du tout de la position d'une source dans l'environnement. Le positionnement est réalisé par une autre bibliothèque tel DirectSound3D ou Open Audio Library.

http://www.hardware.fr/articles/147-1/guide-son-3d-partie-1-eax.html

Ainsi, dans un jeu, c’est DS3D qui s’occupe du positionnement du son à proprement parler et de la distinction des sources alors que l’EAX enrichit la scène en créant un environnement sonore virtuel grâce à divers effets utilisant la réverbération et la réflexion.

**I3DL2**

http://www.iasig.org/pubs/3dl2v1a.pdf

« The level 2.0 enhancements include :

* Environment reverberation model (conveying a sense of the space where the listener is located)
* Enhanced distance model, taking advantage of the reverberation cues in the current environment
* Occlusion and obstruction models for rendering the muffling effects of obstacles inside environments or partitions between environments
* A DirectSound3D property set which provides support for level 2.0 functionality in hardware on PCs with Microsoft’s Windows

**A3D**

<http://en.wikipedia.org/wiki/A3D>

A3D (Aureal 3-Dimensional) was a technology developed by [Aureal Semiconductor](http://en.wikipedia.org/wiki/Aureal_Semiconductor) for use in their [Vortex](http://en.wikipedia.org/wiki/Aureal_Vortex) line of PC sound chips to deliver three-dimensional sound through headphones, two or even four speakers. The technology used [head-related transfer functions](http://en.wikipedia.org/wiki/Head-related_transfer_function) (HRTF), which the human ear interprets as spatial cues indicating the location of a particular sound source.

**FMOD**

<http://fr.wikipedia.org/wiki/FMOD>

FMOD est une [bibliothèque](http://fr.wikipedia.org/wiki/Biblioth%C3%A8que_logicielle) multiplateforme de gestion du son, pouvant être utilisée au travers de nombreux [langages de programmation](http://fr.wikipedia.org/wiki/Langage_de_programmation). Avec le lancement de la version 4.03 la bibliothèque a été renommée FMOD Ex.  
FMOD est gratuite pour une utilisation personnelle et non commerciale.

**Audiere**

<http://en.wikipedia.org/wiki/Audiere>

Audiere does not support three dimensional positional audio

**Irrklang**

<http://khayyam.developpez.com/articles/cpp/irrklang/>

IrrKlang est une bibliothèque C++ de gestion audio, elle permet de jouer des sons aux formats mp3, ogg et wav dans un espace 3D. IrrKlang est multi-plateforme, elle utilise DirectSound sous Windows et Alsa sous Linux.

IrrKlang est extrêmement simple à utiliser, une poignée de fonctions permettent d'accéder à toutes les fonctionnalités. Les fonctions utiles sont les suivantes :

* créer un moteur de sons
* jouer un son en 2D
* jouer un son en 3D
* sélectionner l'emplacement de l'utilisateur dans l'univers 3D
* éventuellement spécifier quelques options pour les sons