

The Function of Sound in Games

This article is devoted to the various disciplines that encapsulate game audio. It's written merely as a formal description of each facet to enlighten the reader, and potentially assist those who are either considering working in the area of game audio or to help producers and designers work with their audio teams.

There are 4 main areas of sound, which I've subdivided into further disciplines (in brackets). I will talk about each area from a purely creative point of view, and try not to be too specific on the differences between consoles & PC.

- ◆ Music (linear, non-linear and [source music](#))
- ◆ Sound Design (spot FX, ambience, [Foley](#))
- ◆ Dialogue (in-game, non-linear)
- ◆ Implementation

Music

This is the most obvious one of all the disciplines. Unfortunately, it's also **completely** subjective. This isn't helped by the fact that by its very nature, music can be frustratingly hard to describe in any sort of detail. What suits a particular project is down to taste and (depending on your viewpoint) any previously set precedent. Whether you look to pop music, film scores, previous games, or simply a visual style to take your reference from is entirely up to the individual. Whatever the course of action chosen, 2 rules of thumb for music are that it should:

1. Enhance the mood and atmosphere
2. Help set the pace

Irrespective of the musical style chosen, the composer should be briefed in as much detail as possible with the use of reference material and avoiding the age old trap of having the composer write music that appeals to development tastes rather than what's best for the product. We've all done it!

LINEAR MUSIC

Exactly what it says on the tin! Games by their definition are interactive and are therefore Non-Linear. There are certain games which run 'on rails' in that they start at point A and after a fixed period of time arrive at point B, so are significantly more linear than a game where you start in a 3D world and pick which ever direction you want to go. It's actually a lot easier to write a piece of music to accompany a game that's 'on rails' because you know that it's fixed - essentially linear. You can time music exactly to what's happening on screen and the methodologies used are very close to the techniques employed by film composers.

Most games, however, aren't on rails. They give the player complete freedom within a predefined world to go where he or she pleases. So, how does linear music fit into this category? Games that

follow a strong narrative normally have cut scenes / FMV. These are points where you meet new characters / flesh out more of the story / come to major junctures in the game etc. They are generally linear and normally involve dialogue. Linear music works well here simply because the game itself becomes linear and therefore non-interactive.

Title sequence / credits are also normally linear, so the same rule applies. In essence, where the game becomes linear for whatever reason, it's easy to place music over this, and the music can rise and fall according to the visuals.

NON-LINEAR MUSIC

Where the waters become muddied is 'in-game'; where the game is clearly interactive. If music is to be pre-rendered in a studio as opposed to using the host machine to generate the music, it **has** to be linear. In the good old days, a composer may be asked to compose a piece of music to accompany a particular level / area / race track for example. The problem occurs when that level encompasses a whole wide range of tempo's and emotions - large exploratory slow sections in the same level as fast upbeat combat. How is it possible to write a piece of music that works on all of those levels? In the 'I level I piece' scenario it is simply impossible. Most composers would simply write a generic piece of music that fits the overall 'mood' of the level but still stands apart from the rest.

A more contemporary take would be to follow the film / TV methodology - to write music that is Event Driven (tm!) i.e. something that happens on screen triggers a piece of music to be played. Clever programming could even anticipate such a scenario - in much the same way that the audience knows that the protagonist in a film is about to enter a dangerous situation before the protagonist knows.

If the music is 'chip music' (created by the host machine) rather than 'digital audio' (pre-recorded music created in a studio) the composer is afforded far greater control over what kind of music plays in any scenario. This raises an important issue of quality with regards to chip music vs. digital audio. The advantage of more interactive control is traded off with restrictions regarding the music generating capabilities of the host machine. Studio music has no such restrictions, but is limited to what has already been created in the studio.

In a perfect world scenario, a composer would write cues to accompany every scenario within a game environment, but in reality this simply isn't possible - it may require 100's of hours of music. So, the next best thing is to be as creatively prolific as possible within the time and budgetary constraints... and then cheat for the rest! Either by repetition, writing a selection of generic cues to match given situations (e.g. action 1, 2 & 3), and use sound effects, ambience, dialogue or silence to complete the audio picture!

SOURCE MUSIC

This can be thought of as music used in the background – e.g. music coming from a jukebox, radio, lift etc. It may not be anything to do with the game whatsoever but fits nicely in the given environment. It can be very effective to fade between source music other types of game music.

Sound Design

The role of the Sound Designer has only come to the fore in game audio over the past 4 or 5 years. Historically, the in-house composer or freelancer would take on all sound effects responsibilities, but with advances in technology and greater emphasis being placed on sound effects and their importance, Sound Design has evolved into a wholly separate discipline (much like it is in film).

A BIT OF HISTORY

Up until the early 90's, sound effects were **synthesised** using clever programming techniques and the sound capability of the host machine. With the proliferation of sampling and mass storage it became necessary to create sound effects from source material. The quickest method of doing this was by investing in a library of Sound Effects CD's.

As Sound Designer's became more experienced, it soon became obvious that to create unique sounding games, it would be necessary to create **unique** sound effects (not just pilfer sounds from pre-recorded CD libraries). This can be achieved in several ways:

- ◆ Going into the field and recording your own library of unique sounds
- ◆ Layering and mixing individual sounds together in multi-tracking software
- ◆ Processing the sounds in some way (e.g. radically changing the pitch to make it sound deeper) either in software or using [outboard gear](#)

It is still commonplace for more straightforward sounds to still be acquired direct from a CD, but only where it's relative importance is low.

SPOT FX

This category probably accounts for the majority of sound effects heard in-game. It incorporates all character sounds (vocal sounds, impact sounds etc.) and any other entity sounds (e.g. aeroplanes, bombs etc.) These sounds are almost always resident in RAM as they are synchronised with entities controlled by the game engine. Their behaviour is unpredictable and therefore can never be pre-rendered.

AMBIENCE

Ambience are the sounds present to help set the atmosphere, whether that be a cacophony of tropical birds to convey a jungle to the wail of car horns, police sirens and crowds to depict a busy city, it's function is the same.

Ambience can be treated in one of two ways. Ambient sound effects can be resident in RAM (as with Spot FX) or pre-rendered ambient streams can be read directly from CD / hard drive (as with music.) As with music, the usual advantages / disadvantages of this apply. In most instances it is best to use a combination of both. When a long pre-rendered sequence reaches the end and cycles back to the beginning the join can be masked with the use of a RAM resident sound.

FOLEY

Foley is an up and coming area of game audio. (See Glossary of Terms for description of FOLEY). A simple description of Foley is that it is the other sounds that are not classed as Spot FX (sound effects which are focussed on) or ambience. The rustle of a leather jacket as a bad guy walks past the player, the shell casings flying out of a hi-speed automatic gun – unless the camera chooses to focus upon them in some clichéd slow-mo sequence.

So why has it not been used in games up until now? Well for many reasons. The main reasons are limited sound fidelity and reproduction. The limited sound fidelity refers to the low sample rate and the need to squeeze in 100s of sound effects into a tiny amount of RAM. Not only did sound designers not have the space to put Foley in, a low-sample rate rustle of a leather jacket wouldn't sound anything like a leather jacket anyway! The reproduction refers to the speakers that most players had – game audio people had to cater for Mum's black and white muffled mono TV or cheap PC speakers, neither of which would be good enough for you to hear the Foley over the sound of everything else.

It's becoming popular to use Foley for cutscenes / FMV where the audio is pre-rendered. It imbues a greatly enhanced sense of realism to a scene or character and brings audio that much closer to film quality sound.

Dialogue

Dialogue has an obvious role in games, to bring life and character to what is essentially either a few hundred pixels or polygons. Designers wish players to be able to relate to a character on an emotional context, to *empathise* with his or her plight. They want you to care about what happens to the character, that drive to make you plough on, regardless of how tough the game is getting. Certain genres' of games rely upon a good story to draw you in, in much the same way that a film does, a strong performance by a good actor can help do that in ways that no other medium can.

Even Hollywood script writers are now being drafted in to help doctor game scripts to bring them up to scratch. Game dialogue is a LOT harder to write than most people think because the script has to cover all eventualities – what happens if the player chooses to go here instead of here, what happens if the main character doesn't meet her before she meets him, or neither! These different outcomes can balloon a game script into being 3 or 4 times larger than the average film so it's always worth getting the script working as early as possible.

In game it's become the norm to record 'place holder' dialogue. Rather than hiring expensive voice over actors from the word go, record members of the team with the best voices to record their own versions and see how they work in-game. It'll prevent costly mistakes later on.

Implementation

Implementation is an area much overlooked by developers, but can enhance the audio experience immeasurably if it is given the correct amount of time and expertise. This area has already been discussed in our [Develop article - The Importance of Implementation.doc](#).