Abstract

In recent years, a new breed of composer has surfaced in response to the growing demand for elaborate audio for video games and multimedia. The “game music composer” faces a slew of challenges unheard of in established venues such as film scoring. My goal is to illustrate the techniques used that have advanced the compositional “bar” for the game music composer, and to analyze several games that have been historically significant in the development of multimedia scoring as an emerging field. I intend to discuss both the current state of the field and what demands future gaming technology will place on the multimedia composer.

1 Introduction

The game music composer acts as both audio programmer and scoring musician. The scoring process involves both understanding the audio technology available in the game, and then composing within its limits. The composer cannot work in an isolated fashion. Part of the excitement in developing a game is that all of the people involved are dependent on each other’s work. Similar to how the visual artists must closely match their illustrations with the animation programmers, the music composer works with the developers who program in the audio cue points and who design the code for interactive sound. My objective in this paper is to explain how interactive sound works in video games and to show why it is an attractive challenge to the computer music composer.

2 Adaptive Music

The idea behind any successful soundtrack is to adapt the music to match events that occur on screen. Many adventure games do this simply by changing the music whenever a new scene is encountered, or in an action game, when a player completes a “level” and moves on to the next. When closely analyzed, however, this is only an attempt to apply familiar methods of composition to a new medium. The composer approaches the soundtrack from the perspective of composing complete, fixed-length pieces of music. Unfortunately, the game player will ignore even the most deftly composed tracks after a few listens. Truly adaptive music responds to specific game events with varying degrees of subtlety. The idea runs parallel to the non-linear medium of computer gaming. “An adaptive soundtrack that responds well to game events is one of the best ways to envelop the player in a game experience… Event-driven music isn’t composed for linear playback; instead, it is written in such a way as to allow a certain music sequence (ranging in size from one note to several minutes of music) to transition into one or more other sequences at any point in time.” (Bernstein, 1998)

A classic example of an event-driven music engine is found in the space combat game Wing Commander (Origin, 1990). The game puts the player in control of a combat pilot character on a space station. While it is a 3D-space combat game, an emphasis is placed on the development of the story-line, which can branch out in many directions depending on how well the character is fighting against the enemy aggression. The music in the game changes in accordance to the current game state during each mission. As the player flies through space on a patrol, for example, music segment A is heard. When enemy fighters appear on the radar screen, this music switches to an introductory engagement theme B within a few moments. Then the music begins to closely follow the combat sequence. As the player approaches the enemy fighters, the music builds intensity and cycles through segments C and C1. When an enemy fighter chases the player, the timpani come crashing down and the sound track takes on a gripping tone in segment D, which may be repeated a few times if the player remains in close combat. When the player finally destroys a fighter, which takes some work, a victory sequence E is played momentarily, and then the music changes back from “win” state E to “advantage” state C1. The combat theme B is re-introduced, but this time, variation B1 might play instead of segment B. While there were some distracting moments when the music sequences changed in the middle of a beat, disturbing the rhythm, for the most part the sound was seamless and contributed to an engaging experience for the game player. More recent titles have refined the technology and smoothed out the bumps, which is also due in part to much faster hardware.
The single most important thing a composer should do is work with the game developers from the beginning. "Integrating an interactive sound engine at the beginning of the development cycle can make things a lot easier. Booking time at regular intervals in the development cycle for effects and music integration allows the composer invaluable feedback as to how their material can be modified in order to better fit the game." (Hoffberg, 1997). There is a limit to how much the game programmers should be expected to do, however. I would draw the line at involving the programmers in composition; i.e. don’t ask them to actually change MIDI instruments on a track, for example. This should be encoded in the piece as part of the composition in sysex messages.

An event-driven music engine corresponds to the finite state machine that dictates the game logic. The structure of an event-driven music engine can be broken down into two distinct parts. The first part is the control logic, which is a series of commands or scripts that control the flow of the music depending on the game state. The second part consists of the segments of audio that are cued by the control logic. (Bernstein, 1997) These segments can be of any length and combine to form a backdrop to the game that changes dynamically depending on the progress of the player.

_Wing Commander’s_ soundtrack succeeded not because of the control logic behind the music – the logic was primitive, in fact -- but because the music itself was high quality and well matched to the on-screen action. Bernstein says it best: “The music must always come first. If it doesn’t gel as a musical composition, you’re better off writing a linear score.”

### 3 Cueing

Many games have a soundtrack composed of fixed length pieces of music. These usually loop indefinitely, until the player turns the music off in utter disgust. Jonathan Hoffberg, a composer whose company credits include LucasArts, said “there is an often unchallenged assumption in video game production that there must be music everywhere, all the time… And so it goes that the best laid tracks are doomed to a silent death… You can get 10 times the mileage out of 3 minutes of music if you break it up into small chunks, play them back in a variety of different orders, and then fade them in and out.” _Tomb Raider_ (Eidos Interactive, 1996) is an Indiana Jones-style adventure game set in first-person perspective. The soundtrack in this game succeeds largely because it is sparsely scored. Approximately 90% of the time, the only sound is the footfalls of the lead character echoing in the vast caverns and the occasional sounds of combat. Whenever the player encounters a piece of a puzzle, or even simply a room with a distinct feature such as a pool of water or a deep chasm, a piece of music of approximately 30 seconds in length is cued and played. The music is consistent in style and varies greatly in dynamics to match the importance of the discovery, or the intensity of the challenge.

Some composers write one or two-measure segments of music. Then they group them so that the segments can be arranged and played in real-time. It may be best to compose a single large piece of music with all layers in place. If the piece sounds good this way, then usually some of the tracks can be removed safely while retaining the integrity of the music. (Griffin, 1998). Bernstein asserts that in an event-drive music engine, successful cues can be connected in a modular fashion. “Decouple segments horizontally and vertically. Compose your music so that different segments may be combined end to end (horizontally) as well as on top of each other (vertically).”

I should note that the technology used for the audio determines whether or not modular music is possible. Both MIDI and MOD files require little overhead while the game is running. It is a simple task to instantaneously switch from one cue to the next, or to decide in real time that the track should be played by a flute against the classical guitar in counterpoint rather than relegated to the string section and played as a solo. However, the current trend in many games is to record music directly to the CD-ROM as Red Book audio and play it back in real time. This insures that the soundtrack will be consistent across multiple platforms and will not depend on how good the MIDI instrument samples in the computer or console system is. Yet the access time of even today’s CD-ROM drives adds a significant delay when switching tracks, and makes it impossible to play several different tracks simultaneously. Mark Miller, the audio and video director for Crystal Dynamics, comments “Red Book audio sounds great, but our programmers know how to get the most out of the CD-ROM drive for game play. Therefore, it’s not always available for audio.”

One final point is to ensure that the game play is not imbalanced by improperly cueing music before its time, thereby giving the player advance notice that something is about to happen. Using a real-time strategy game as an example, suppose the player was about to be sabotaged by the computer-controlled enemy army. The computer has lined up all of the various infantry, tanks, and so forth behind trees and is about to launch a full-scale attack on the player’s complex. By all means, don’t start playing the “sneak-attack cue” before the player realizes what is going on! The composer must again work with the programmers to ensure that the timing of the cue builds tension as effectively as possible. (Bernstein, 1997)
4 Composition

In this section I will not attempt to recommend particular genres of music, as that should be matched to the game’s atmosphere, setting, and graphical style. Instead I would like to focus on the specifics of composition. The idea is to enforce a creative approach to scoring without calling undue attention to the music itself. Curtis Macdonald, a freelance media scorer, describes the job of the composer as “a designer of composition as well as a director of musical transparency.” He asserts that the music should be pure minimalist, and to “make sure that your composition’s harmonic progression or tonality is arranged with less than three elements (an element may be a melody line, a rhythm pattern, orchestration, etc.).” Donald Griffin also warns against writing strong melodies in a score. He argues that “[musical] textures are just as pleasing as, and less annoying than, a catchy phrase. This has the added benefit of causing your important transitional themes to stand out much more.”

The balance between textural, ambient music and strong transitional themes is effective in maintaining a consistent feel between the different sections of the game and ensuring that they all connect smoothly. Take, for example, the cut scenes in many adventure games, where the music is brought to the forefront while another part of the story is unveiled before control is returned to the player for the next part of the game. *Prince of Persia* (Interplay, 1989) is a 2D side-scrolling adventure game where each character is assigned a short motif. A variation on the motif is played whenever that character is introduced in a cut scene or during the actual game play. The game is largely silent otherwise. This is ideal because it served to emphasize the motifs used in the transitional sequences. This is an extreme example; in many cases, some textural or ambient music will serve more effectively than silence.

In most games it is impossible to determine how long a player will remain in a given segment of the game. Consequently, avoid putting cadences anywhere in the music, e.g. V-I or IV-I cadences (Macdonald, 1998). This only makes the loop point that much more obvious, and can confuse the player as well, who may be mislead by the “end” of the piece and expect something new to happen. Often the composer is only allotted a short amount of time for each loop; in these cases it is important to compose in a minimalist style and refrain from putting in a memorable phrase or riff. Instrumentation and octave changes are a cheap way to add some variety to a MIDI score, since the sysex messages take up very little memory. However, be mindful of the relative velocity of different MIDI instruments and also be wary of how octave changes can affect your score. A major seventh can become a minor second, thereby introducing more dissonance into the piece than was expected. (Griffin, 1998).

Music during a particular section can be broken down into sub-segments, and the pieces interchanged throughout a scene. Griffin notes, “An interactive composition may have a beginning, a middle, and an end, or it may not. The composer may not know for sure if the exposition will happen before the conclusion. Each small piece of music must have a much stronger internal structure than it would if it were in a larger linear composition.” Compose toward a closed, modular design for all of the music. In other word, a piece of music should not depend on the one before it in order to make musical sense. Each segment of music should stand up on its own.

5 Sound effects

Ambient sound effects can add to the realism of a game by immersing the player in an environment. In an underground cavern, for example, the program can cycle through various sound effects such as water dripping into a pool, a torch burning, or the distant sound of rats echoing through the chamber. For an effective presentation, these sounds should be played at random intervals, sometimes one overlapping the other. Bernstein suggests, “try to create fine gradations of ambiences.” If the player moves from a small room to a large cavern, a sudden increase in reverb decay will not seem natural. By categorizing ambient sounds into groups such as “small room,” “hallway,” “side of cavern,” and “center of cavern,” we are better able to convey the transition of environments to the player. (Bernstein, 1997). The gradual change in environmental sounds will make the transition seem more realistic.

For game sound effects (e.g. laser cannons), Hoffberg notes that “the more unusual the effect, the fewer times it can be heard before it loses its dramatic punch.” Sounds that will be repeated indefinitely should be more subtle in nature. Incorporate some sonic variation as well. While one could randomly alternate from a set of laser cannon sounds, it would take up much less memory to use just one sound and introduce small variations in pitch or decay each time the sound is played.

6 Future technology

The state of music and sound in video games is closely tied with the technology available on each platform, be it a console system (such as the Sony Playstation) or a computer game (such as for a Mac or PC). While the
compositional requirements I outlined above are static, the technology used to implement adaptive music and event-driven engines varies from platform to platform. Various groups are working on new standards for interactive music. 3D sound, which gives the player spatial clues to the position of characters or objects in a 3D world, has been around for years on the PC platform. Due to the conflict in interest between the various companies behind the technology, however no major game has yet to make effective use of it. Composing for 3D sound currently has the added difficulty of first figuring out whether or not the implementation chosen will even work properly. The effectiveness of 3D sound is also determined on the speaker configuration; if the player is wearing headphones, the results will be vastly different than if the computer system is hooked into a Dolby Digital 5.1 surround setup. Provisions for selecting the speaker configuration must be coded as part of the game.

The Interactive Audio Special Interest Group (IA-SIG), a group of sound technology experts in the gaming industry, has been working on putting together an industry standard for 3D sound. They have also proposed a basic standard for downloadable samples called DLS-1. (Brandon, 1998). Yamaha will use this technology in their SWx00 series sound cards to enable developers to import their own instrument samples into each game. This would allow composers to use MIDI, which I find to be vastly preferable from the viewpoint of composing for algorithmic control logic, while still maintaining a high level of instrument quality. Current MIDI instrument banks are encoded in ROM in game consoles and current sound cards, which effectively limits the realism level of a MIDI-based soundtrack. They are also working on an Adaptive Audio Engine (AAE) in cohorts with Thomas Dolby, Mark Miller, and Alex Stahl of Pixar, which would introduce a standard event driven music engine.

7 Conclusion

Currently, the multimedia composer must first work with the developers and determine the technology used for the sound in a game. Once the tools and boundaries are agreed on, the composer can then begin work on the music and sound effects. The composer must work closely with the game programmers throughout the development cycle. With new developments in interactive audio just over the horizon, this requirement may be reduced in the future. Already we can see the huge advances that have been made on today’s computer and game console systems. The days of squeezing an entire soundtrack into 512 bytes of memory have long past. It is up to the composer to take advantage of the myriad available tools and ideas in order to write an effective score.

References


McCree, Nathan, and Iveson, Martin, Tomb Raider, Eidos Interactive, 1996.

Miller, Mark Steven, “Producing Interactive Audio: Thoughts, Tools, and Techniques,” Game Developer, 10/97.


Warwick, Rich, “Avoiding a DirectSound3D Disaster,” Game Developer, 1/98.