FMOD DESIGNER 2010

USER MANUAL

http://www.fmod.org

Updated: Oct 10

Legal Notice

The information in this document is subject to change without notice and does not represent a commitment on the part of Firelight Technologies. This document is provided for informational purposes only and Firelight Technologies makes no warranties, either express or implied, in this document. Information in this document, including URL and other Internet Web site references, is subject to change without notice. The entire risk of the use or the results of the use of this document remains with the user. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Firelight Technologies.

TABLE OF CONTENTS

CHAPTER ONE: FMOD Designer 2010

1

Welcome to the FMOD Designer 2010 user manual. This manual contains instructions, reference material, performance optimizations and tips on getting the best results using the FMOD Designer 2010 software. In this chapter an introduction to FMOD Designer 2010 is given.

Key topics in this chapter:

- A description of FMOD Designer 2010
- Platform Support

What is FMOD Designer?

FMOD Designer 2010 is a tool for creating complex audio events for playback with the FMOD Ex Music and Sound Effects System. This application is designed to allow a sound designer to create immersive auditory environments without the need for programming.

Using the FMOD Designer 2010 tool, a sound designer can control both the sound and the behavior of sound within a game title. The sound designer can model real world sounds by creating events that contain layers, parameters, crossfading, DSP effects and randomized behavior.

The software will then produce an output file that allows a programmer to dynamically control events at runtime in order to realize a rich interactive soundscape.

Platform Support

The FMOD Designer 2010 software is currently available for the following platforms:

- Microsoft Windows series
- Macintosh OS X

FMOD Designer 2010 can create content for the following platforms:

- Microsoft Windows series 32-bit
- Microsoft Windows series 64-bit
- Linux
- Linux 64-bit
- Solaris
- Macintosh OS8 / 9 / X / X86
- Apple iPhone
- Sony PlayStation 2
- Sony PlayStation Portable
- Sony PlayStation 3
- Microsoft Xbox
- Microsoft Xbox 360
- Nintendo Gamecube
- Nintendo Wii

FMOD Designer 2010 supports all platform-specific audio compression methods, such as XADPCM for Microsoft Xbox, VAG for Sony PlayStation 2 and PlayStation Portable and XMA for Xbox 360. All hardware features are available to the sound designer where applicable.

CHAPTER TWO: Getting Started

2

In this chapter, the FMOD Designer 2010 user interface is explored. To help users navigate through all the FMOD Designer 2010 functionalities, a suggested workflow philosophy is explained and the basic operation of the software are described.

Key topics in this chapter.

- Exploring the basic operation of FMOD Designer 2010
- The functionality provided in the main views in FMOD Designer 2010

The Interface

Overview of Key Terms

There is a complete hierarchy of objects used within FMOD Designer 2010 interface. Below is a quick description of the key terms mentioned in this user manual.

Object	Description
Workspace	A workspace allows for multiple projects to be open simultaneously and to be linked within the workspace file.
Project	A file that contains all the FMOD Designer 2010 content In general a single project will contain all the material required for a single game project. Usually a single project file would contain all the audio elements for a single game title, it however possible to use multiple project files within a game project if desired.
Event Category	A virtual folder used to organize event data into generic groups such as music. Allows volume and pitch changes to be applied to all events within a category
Event Group	A virtual folder used to organize event data. Event groups and Event Categories exist simultaneously. Groups allow for easier workflow during production, while categories allow for group manipulation such as volume changes during runtime.
Simple Event	An event is the most used object in FMOD Designer 2010, as it specifies the actions taken when audio is requested. A Simple Event is a self contained object that directly references Sound Files and includes the properties of how those Sound Files will playback through FMOD
Multi-Track Event	An event is the most used object in FMOD Designer 2010, as it specifies the actions taken when audio is requested. A Multi-Track Event can include multiple layers enabling the mixing and simultaneous playback of many sounds. Multi-Track Events utilize Sound Defs instead of using sound files directly.
Event Template	Any Event can be turned into an Event Template. An Event Template allows for properties to be pre-set when the Event is created.
Event Layer	Provides an event with the ability to play multiple audio pieces simultaneously.
Event Layer Effect	A variety of DSP effects can be added to Event Layers. These include things like Volume, Pitch, Reverb and a variety of filters.
Effect Automation	Various Layer Effects can be controlled via a graphical line interface with nodes attached. The nodes can be positioned to set values at any point during

	playback
Sound Def Instance	A unique occurrence of a Sound Def. Within an event multiple versions of the same Sound Def can exist and can be assigned unique properties. These are referred to as Sound Def Instances.
Parameter	Provides functionality to adjust the characteristics of the audio playback in real-time.
Effect	Digital signal processing used to add impact to the audio. Includes low pass filtering, distortion, flanging, etc.
Sound Def	An object that contains audio files or other sound producing entities such as oscillators.
Sound Def Folder	A way of grouping Sound Defs together in the Sound Def Browser pane. Folders can be titled as required and used to sort Sound Defs.
Sound File (or wavetable)	An existing audio file.
Oscillator	A real-time generated audio signal.
Banks	Specifies the characteristics and content of the files exported to the game title.
Reverb Defs	Specifies the characteristics of the reverberation used in the project.

Note:

Key terms and objects used exclusively in the Music View are explained Chapter 9.

Working with the Interface

Before starting an FMOD Designer 2010 project, it is important to consider the most efficient way of working. In the next section, a brief overview of the FMOD Designer 2010 tool is given and a workflow philosophy is suggested.

Overview and Workflow Philosophy

Most new users simply want to add a set of audio files and start building things. However, this is not how the FMOD Designer 2010 workflow is structured. Instead FMOD Designer 2010's workflow encourages the sound designer to think about the sound events that appear in the game.

Each important step in FMOD Designer 2010's workflow is laid out using window views. As you can see in the figure below, the 'Events' View is first. Once an event is created, the sound designer can then attach sounds within the 'Event View.

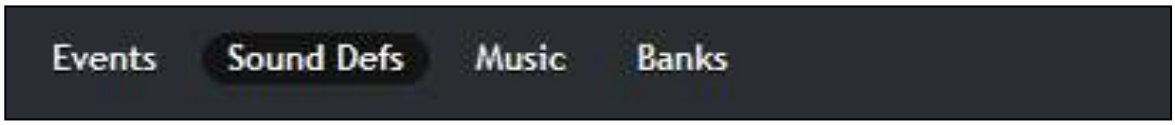


Figure 2-1: FMOD 2010 View Selection Buttons

The four main views in the FMOD Designer 2010 tool represent the:

- Event View
- Sound Defs View
- Banks View
- Music View

In the next section, the role of each view will be briefly described.

Learning the Interface

The Events View

Event Browser Pane

Before discussing the Events View, it is important to define what an event is. An event is a trigger that a Programmer will call from within their game code to play or update a sound. A typical event might be the sound for a gun, monster, explosion, car engine or even music…anything that makes a sound! Using FMOD Designer 2010, the sound designer not only attaches sounds to an event but may also create parameters that allow the sound to be manipulated.

An FMOD Designer 2010 event contains a combination of:

- Sound sources
- Layers,
- DSP Effects, and
- Parameters to manipulate the behavior of the sound

As stated in the first chapter, it is the role of the sound designer to create the events and define the behavior of the sound in the game.

The Programmer is responsible for starting and stopping events, as well as, updating event data from the game code. While the Programmer updates values at run-time, the audio content and behavior is defined within the sound designer's event.

Consider the following example of a first-person shooting game. Each time an in-game player fires a weapon, a sound event (or events) is triggered.

The sound designer is responsible for tasks such as:

- Creating a Gunfire event.
- Selecting audio samples.
- Setting the properties of an event, for example, one-shot or looping sounds
- Specifying an event to be played in a 2D or 3D space environment.
- Creating a parameter that modifies the audio based on a character's distance from the generation point.
- Mixing the relative volumes of the gunfire event with the other game events.

The Events View is responsible for creating and organizing events. Event definitions are created in the Event Browser Pane, under the Events View. This View can be seen in the figure on the next page. A Sound Designer can create both Simple Events and Multi-Track Events.

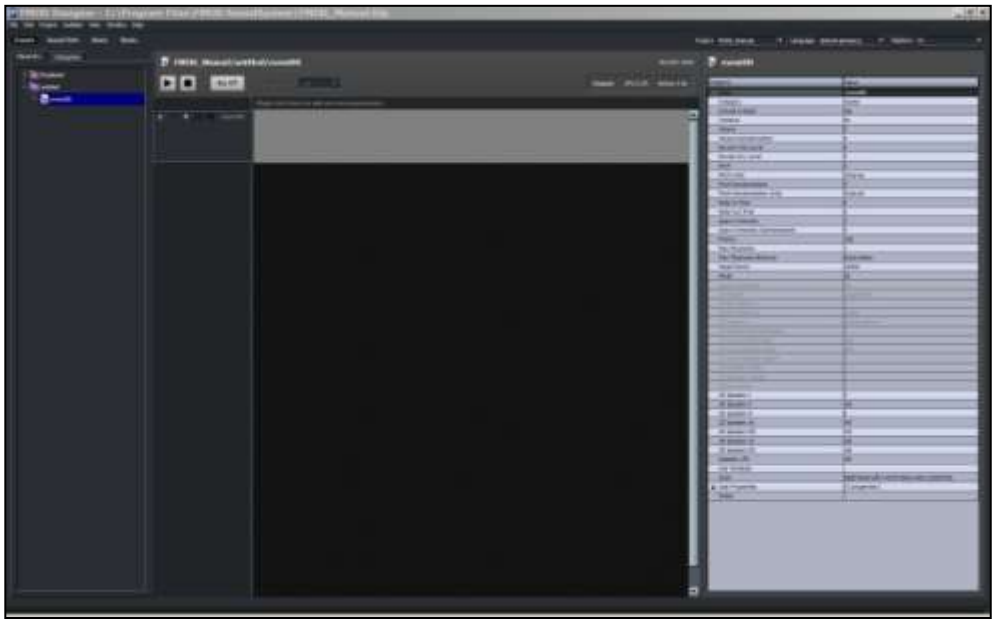


Figure 2-2: The Events View

The Event Browser Pane can be used to sort events into event groups. The event group can be seen as a nested group in the Event Groups tree. The event groups in the figure above demonstrates this. Event groups are essential for data organization and efficient memory usage. This concept is explained in further detail in Chapter Six, Optimizations.

The Event Category pane can be used to group events for mixing. An event category works somewhat like a sub-group on a mixing desk (controlling the volume of multiple channels with a single fader). An event category allows the sound designer to change the volume and pitch of all events within the category in one location. This is very useful when fine-tuning the relative mix of many events.

Simple Event Editor Pane

Using FMOD Designer 2010 a sound designer can create two types of events. Simple events are the most basic and usually resource efficient type of events. A Simple Event can be created by right-clicking in the Event Browser pane and selecting Add Simple Event… .

Once created the Event Editor area will display the Simple Event Editor Pane (this area will automatically display either the Simple Event Editor or the Multi-Track Event Editor dependant on what type of event has been selected). In this Pane the user can add sound files directly to the event and alter the various properties that effect how the sound will play.

Figure 2-3: The Simple Event Editor Pane

The Simple Event Editor uses essentially the same set of properties used to create Sound Defs. The interface uses graphical controls similar to most common DAW applications. A Simple Event can contain multiple sound files in the same way a Sound Def can; how the list is accessed, its looping behavior and build options can all be set within the Simple Event Editor Pane.

Multi-Track Event Editor Pane

Once a Multi-Track Event has been created in the Event Browser pane, it can be edited. It is the Multi-Track Event Editor pane that allows the sound designer to combine the various elements to create an event. This can be done by adding layers, adding Sound Defs to the layers and adding effects to the layers. Altering the events properties is done via the Event Properties pane

To edit a Multi-Track Event, simply click the event in the browser pane. FMOD Designer 2010 will display the event components in the Multi-Track Event Editor Pane and the event properties in the Event Properties Pane.

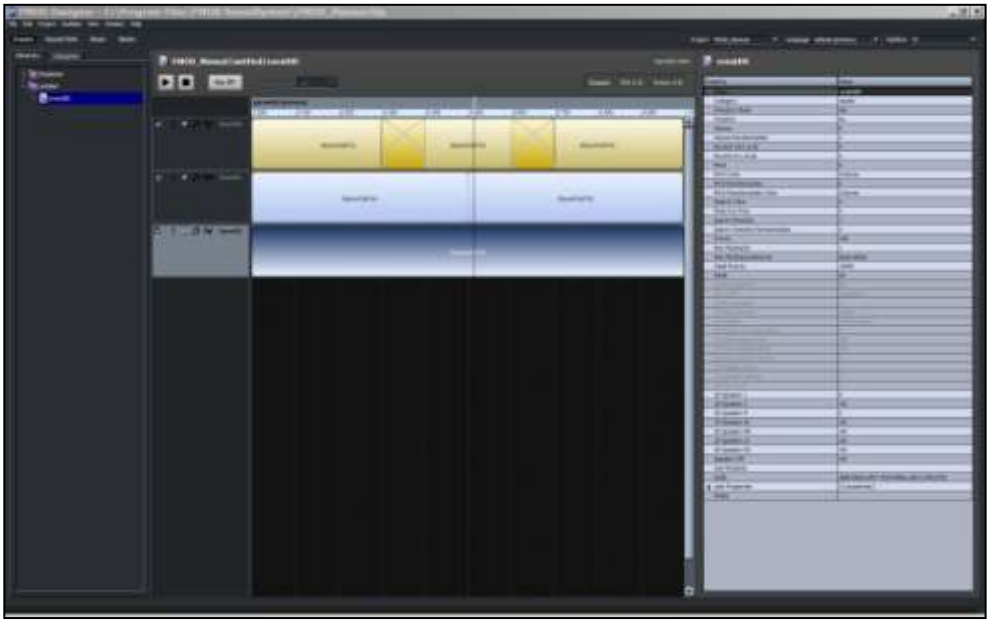


Figure 2-4: The Multi-Track Event Editor Pane

The major components of an event are designed within this pane. Using the Multi-Track Event Editor it is possible to create complex events, that can contain multiple sounds, multiple layers, cross fading, effects, sequencing, pitch bending, event parameter definition and more. It is possible to create an event with no parameters and only a single sound within the Multi-Track Event Editor, however it is far more efficient to use the Simple Event function for this process.

Some complex event examples have been provided with the FMOD Designer 2010 tool and the topic is discussed further in Chapter 5, Tutorial Projects.

The Multi-Track Event Editor pane can be seen in figure 2-5. This example has three layers with a different number of Sound Defs on each layer, as well as a single parameter bar.

Figure 2-5: The Multi-Track Event Parameter Bar

While the parameter bar looks very much like a 'timeline' in its default setting the bar has no relationship with time. Instead, the bar simply represents the current value of a parameter.

Parameters are typically used to represent variable data from the game engine such as engine RPM (Revolutions per minute) or the distance between the listener and the location of the sound emitter. Using parameters a sound designer can create dynamic and non-linear sequenced sound events from otherwise static samples.

Note:

Audio files can be dragged from an operating system window to FMOD Designer and dropped into the Event Browser pane to automatically create a simple one-shot event or dropped into the Sound Def Browser pane to automatically create Sound Defs. You can add any type of audio file to an FMOD Sound Def, even midi files!

Bulk Editor Pane

The Event Editor area can also display the Event Bulk Editor Pane. Whenever multiple Events are selected in the Event Browser Pane, FMOD 2010 automatically switches to the Event Bulk Editor Pane and displays controls that allow certain properties to be altered across the selected events simultaneously. To select multiple events Shift-Click or CTRL-Click on the desired events within the Event Browser Pane.

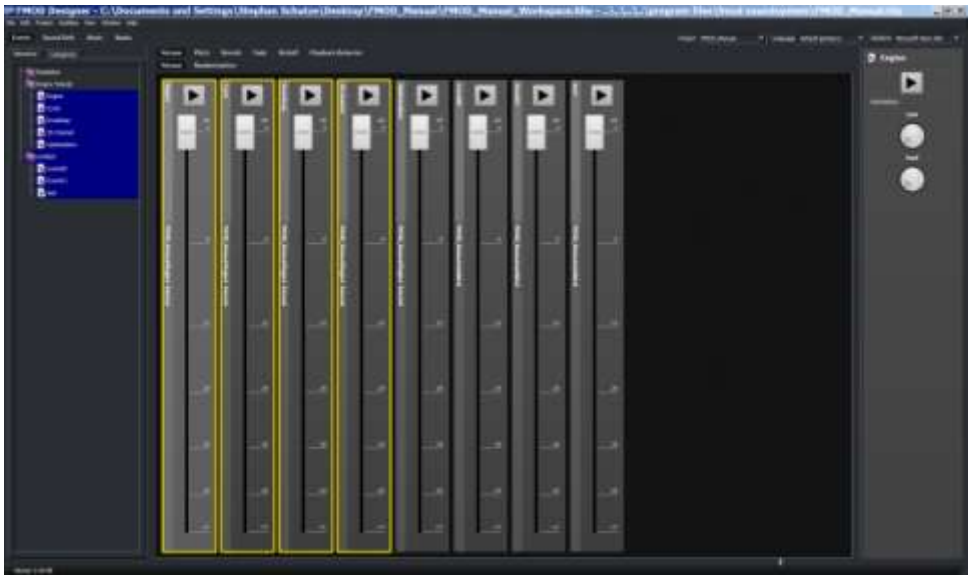


Figure 2-6: The Event Bulk Editor Pane

Within the Event Bulk Editor pane multiple events can be simultaneously selected. The selected events will be outlined in yellow and any property alterations will occur across the selection. The Properties Pane on the left hand side will display the Parameters the last event to be clicked on. The buttons above the Event Bulk Editor Pane allow the

user to switch between the various properties. These properties include. **Volume**, **Pitch**, **Reverb**, **Fade**, **Rolloff** and **Playback Behavior**.The orientation and types of controls within the Event Bulk Editor depends on which property is selected.

The Sound Defs View

Many users new to FMOD Designer 2010 mistakenly assume a Sound Def to be an audio file. The term 'Sound Def' can be used to describe a comprehensive set of sounds, as well as a single sound file.

A Sound Def shown in the event editor may contain multiple sound files (or even real-time synthesized audio) with behavior specified for randomization and attribute variation.

The Sound Def allows the sound designer to control a variety of playback characteristics, such as how and whether sounds are selected randomly from within the Def or randomization of pitch and volume. It may also control spawn behavior, which specifies the manner in which the sound continuously triggers.

The Sound Def View (as seen in the figure below) contains the pane that allows the sound designer to add audio to a Sound Def. As previously stated, Sound Defs can be a single audio file, multiple audio files or real-time generated oscillations (such as white noise, sine wave, square wave etc). These oscillators use no memory but take a small amount of CPU time to generate.

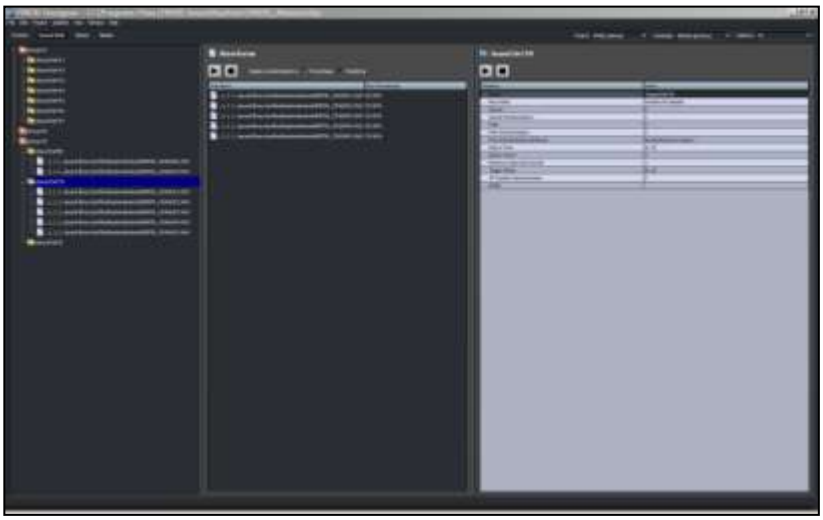


Figure 2-7: The Sound Defs View

Finally if you double click on an entry in the Sound Def Browser pane, it will automatically take you to the next view, which is the Banks View.

The Banks View

Before any content can be exported into a game, it must be organized into 'Banks'. The Banks View provides the functionality for packing Events and Sound Defs into banks. The banks are exported as '.fsb' files and are loaded at runtime by the FMOD EX sound engine.

When creating a Bank, FMOD Designer 2010's build process compresses the audio data using the desired format regardless of the source file type.

For example, when working on the PS2 platform, the source audio files used in the Sound Defs may use MPEG and PCM audio files, but the Banks that are exported to the game will only contain VAG files.

The Sound Files pane displays the audio data to be exported. When constructing banks, audio is generally grouped into logical groups, such as banks per game level or banks that are streamed (i.e. music or ambience). The organization of banks should optimize memory usage. There is no single solution to bank structure. It requires a memory loading strategy and coordination between the programmer and sound designer. This is discussed in Chapter 6: Optimizations.

As a general rule, when designing the sound for a game with ten levels, the sound designer should avoid putting the audio data for all ten levels into a single bank!

The Banks View is displayed in figure 2-8.

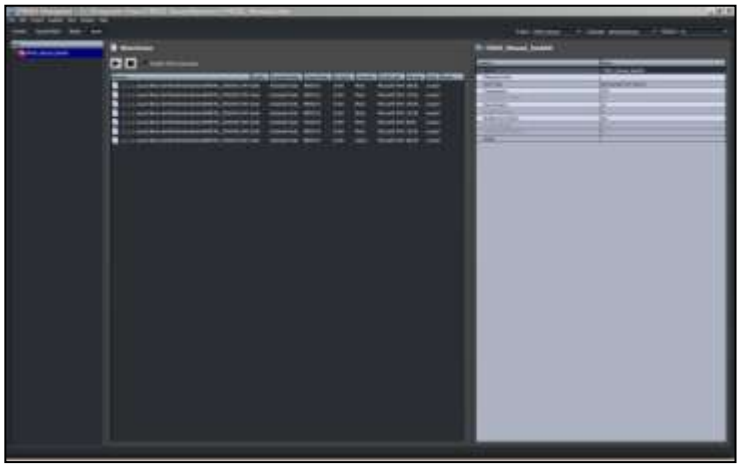


Figure 2-8: The Banks View

The Music View

The Music View within FMOD Designer 2010 allows the sound designer to create a game soundtrack that features cues, musical themes, transitions between music themes and flourishes.

All musical transitions and flourishes can be instructed to playback in a musically intelligent and beat-matched manner. Each theme and transition can be attached to cues and parameterized easily for the Programmer.

The Music View is displayed in the figure below.

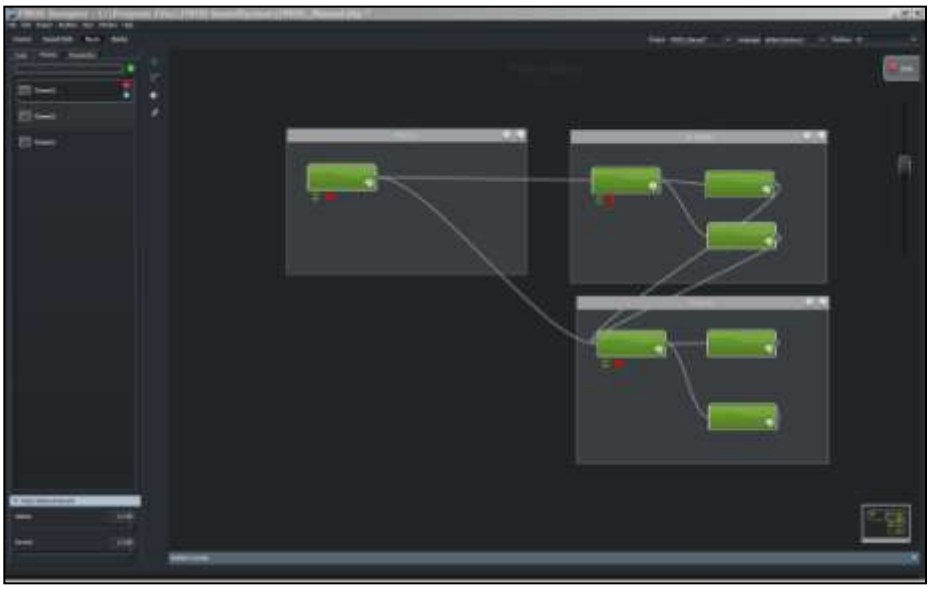


Figure 2-9: The Music View

CHAPTER THREE: Using FMOD Designer 2010

3

In this chapter, the instructions for all basic FMOD Designer 2010 operations will be discussed.

Key topics in this chapter include:

- Managing Projects
- Maintaining Events, Event Categories and Event Groups
- Adding parameters to an event
- Using Sound Defs
- Auditioning events
- Auditioning with FMOD Designer 2010
- Managing Banks and preparing for export
- Managing Workspaces

Projects

Creating a New Project

To create a new project, complete the following steps:

1. Select 'File' from the Menu Bar.
2. Select the menu item 'New Project. A pop-up window should appear.
3. Enter the name and a location of the FMOD Designer 2010 project (.fdp) file.
4. Press 'SAVE'. The project will now be saved to their respective files.

The created '.fdp' file will contain all the data to be exported to the game. The location of any audio files relating to this project will be relative to the location of this file unless the user specifies an absolute audio source directory.

Loading a Project

To load an existing project file:

1. Select 'File' from the Menu bar.
2. Select the menu option 'Open Project…'. A file selection pop-up window should appear.
3. Select the file to load and press 'Open'.
4. The file is now loaded.

Saving a Project

To save a new project:

1. Select 'File' from the Menu bar.
2. Select the menu option 'Save As'. A file selection pop-up window should appear.
3. Select the location and the filename and press 'Save'.
4. The file is now saved.

Choosing a Target Platform

Most sound designers will be creating content for multiple target platforms. FMOD Designer 2010 provides a method for switching easily between target platforms and also allows the sound designer to use different build settings for each platform type.

To change platforms, select the desired platform from the 'Platform Dropdown List'. This dropdown list can be found in the top right corner of the FMOD Designer 2010 tool interface at all times.

Currently, the following properties can be individually assigned for each platform:

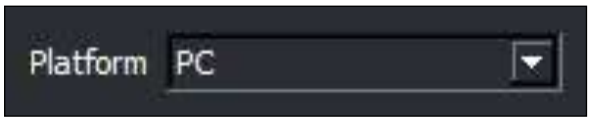


Figure 3-1: The Platform Dropdown List

- 'Build directory'. This property can be found in the Project Properties pop-up window in the Menu Bar under Window
- 'Compression'. This property can be found in each 'Banks' Parameters pane
- 'Enable/Disable Layer icon'. This icon is found in the Event View, in the Event Editor Pane on the Layer panel.
- 'Layer Effects'. The special effects played on layers can be enabled or disabled on a 'per-platform' basis using the 'Enabled effect' checkbox. DSP effects not supported by a platform are automatically disabled.

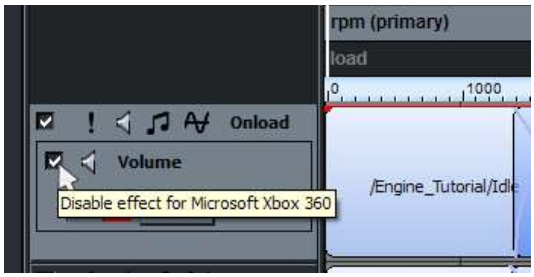


Figure 3-2: Enable/Disable Layer Effects Icon

- 'Bank type'. This property can be found in the Banks' View and allows the sound designer to specify an fsb's sample/streaming behavior on a per platform basis.
- 'Reverb Def properties'. These properties can be applied to all platforms, or on a per-platform basis. Reverb Def properties are located in the Reverb Defs pop-up window accessible though the Window menu in the Menu bar

Figure 3-3: Enable/Disable Layer Icon

Why Is My Platform NOT Available?

Not all platforms supported by FMOD Designer 2010 appear in the platform dropdown list by default (a list of FMOD supported platforms appears in Chapter One). There are a few reasons for this.

The default platform 'PC' is a generic label that represents a number of platforms including operating systems such as Windows, Macintosh and Linux.
Console platforms such as PS2, PSP, PS3, Xbox, Xbox 360, Gamecube etc, will only appear in the platform list box when the relevant SDK is included. For example, FMOD Designer 2010 will automatically detect the Microsoft XDK and make the Xbox and Xbox 360 available in the platform dropdown list.

FMOD Designer 2010 will also detect if the FMOD programmer API for a particular platform is installed. If a console's FMOD programmer API is installed after FMOD Designer 2010, the installer for the programmer API will automatically enable the platform to be listed in the FMOD Designer 2010's platform dropdown list.

Note:

Unfortunately FMOD Designer cannot list all platforms by default.
Some platforms require Non-Disclosure Agreements (NDA) for sensitive or protected codecs and are not to be used by the general public.

To obtain console specific platform support, contact support@fmod.org to verify you are registered as an official developer of the platform.
When verified, you will be given access to the relevant SDK.

Developing Content for Multiple Languages

Many game titles will contain audio for multiple languages to cater for different audiences. In this circumstance, while the audio content is different, the game triggers do not change.

FMOD Designer 2010 supports localization of content, providing the sound designer an easy method of constructing events using a primary language, then allowing the easy substitution of secondary languages when building the project's FSB files.

The language to audition or build with is selected using the 'Language' dropdown list.

Figure 3-4: The Language Drop down menu

More information, about multiple language support can be found in the sections 'Building a Project' and 'Localization'.

Setting Project Properties

Before commencing work on a project, the project properties should be set. While these properties can be set at any time, it is good practice to set an ‘Audio Source’ target directory and a target ‘build directory’ (or directories, if building for multiple platforms) when a project is first created.’

To set the project properties, complete the following steps:

1 Select a platform from the Platform list box.
2 Select Project Properties from Window under the Menu Bar.
3 The Project Properties pop-up window will display.
4 Select the audio source directory by clicking on the […] button in the audio source directory's value cell.
5 Select the build directory location by clicking on the […] button in the build directory's value cell.

Figure 3-5: The Project Properties Pop-Up Window

Protecting Project Data

FMOD provides optional cryptographic protection for the project's audio data. This encryption is useful in stopping end-users making digital copies of the audio data.

The data is protected once the 'Encryption Key' value in the project properties panel protects is set. The key is not stored within the data, it is provided by the programmer in the code side of the application (using an FMOD API function) so that it can be unlocked.

For further information on this subject refer to ‘the EventSystem::load function and FMOD_EVENT_LOADINFO struct’ in the FMOD API documentation

Removing Unused Sound Defs and Sound Files

Before a project is exported to a game title, all unused media and data that are not used should be removed. FMOD Designer 2010 provides an automated method to remove unused Sound Defs and audio media. To clean the project:

1. Select Project from the Menu Bar.
2. Select the menu option Remove Unused Sound Defs and Sound Files. A confirmation pop-up window should appear.
3. Select 'Yes'.
4. All unused Sound Defs and Sound Files will now be removed from the project.

Merging Project Data

The 'Merge' function allows the event data from one project to be copied into another project file. To merge '.fdp' project files:

1. Open the target project '.fdp' file by selecting Open Project...' from the File menu. An open file pop-up window should appear.
2. Select the file to open and press 'Open'.
3. To import event data from another project, select 'File' then 'Merge...' from the Menu bar. A file selection pop-up window should appear.
4. If the source audio cannot be found, FMOD Designer 2010 will prompt for the new location. Press 'OK' in the warning Menu pop-up window, then select the source audio file location using the file selection pop-up window.
5. The new data has been merged into the target project file.

Note:

If an existing project tries to import duplicate data from a mergin project file, the merging data will be ignored and the existing data will remain unchanged.

Building a Project

Once all the content has been created, the sound designer must process their data into a form that can be exported to the game title. The process is known as 'Building the Project'. The output files of the build process are placed in the directory specified in the project property 'Build Directory'. The build process must be repeated for each target platform.

The Programmer (and game title) requires the following files:

- The .FEV file. This file contains all the Defs and properties for the events, event groups, Sound Defs, etc. This file contains no audio data.
- All .FSB files. Each .FSB files represents one of the Banks defined in the project. The .FSB file is a binary file that contains the audio data in the format specified in the wave bank definition.
- The .TXT file [Optional]. This file is intended to be a project reference guide for the Programmer. It contains a list of all the major objects used in the project, along with index, range and important property values.
- The .H file [Optional]. This file is a source code header file that allows the Programmer to call the EventGroup::getEventByID() method, which uses a numbered index, rather than the EventGroup::getEvent() method.
- The .LST file [Optional] This file lists wave files used in the sound banks and is useful if you intend to rebuild the .FSB file using the fsbankex utility.

The building process will also create cache files, used to speed up any subsequent build of the project. The Programmer does not require these files.

Options for the Creator/Coder: Check boxes

Before building a project:

1 Complete all the desired events, Sound Defs and wave bank contents to be exported to the game title.

2 Select the language to build from the 'Language' dropdown list. Ensure all audio files for the language are in the correct location (as specified by the wave bank's 'Filename Prefix' attribute).
3 Select the platform to build from the 'Platform' dropdown list.
4 Click on the 'Events' View and ensure the 'Build Directory' in the 'Project Properties' panel is set to the desired location. Ensure all source audio files are in the Audio Source Directory specified in the Project Properties pop-up window

Take the following steps to build a project:

1 Open the Project menu and select the Build… menu item.
2 Select the target platform from the 'Platform' drop-down list.
3 If the Programmer requires the programmer report or header file, leave the checkboxes checked, otherwise deselect as necessary.
4 If you require a listing of the files within the sound bank (useful if you intend to re-build the .fsb file using the FMOD Soundbank generator utility) click the 'Build wave bank listings (.LST)' checkbox.
5 Select (or deselect) the Banks to be created using the checkboxes in the 'Select which Banks to build' panel.
6 Select whether changes made to templates should be automatically applied to Events before the build process by setting the ‘Apply Template Changes’ checkbox.
7 To hide compilation messages during the build process, press the 'Output' button.
8 Press the 'Build' button. The automatic portion of the build should take place.
9 The newly created files will now be in the directory specified in the project property 'Build Directory'.

Note:

When building projects using XMA, some rare combinations of wave files and quality settings can cause a burst of white noise in the audio of the resulting .FSB file.

This is a known issue with the XMA encoder (not FMOD). If this occurs, try changing the quality value slightly (for example, 100 to 99). This should fix the problem.

Advanced Building Using Build Commands

Pre and post build commands allow the user to execute shell scripts, programs, or batch file jobs before and after the build process has taken place.

Pre- and post-build commands in Designer 2010 can include macro references, which will be evaluated before executing the command. A macro reference looks like '$MacroName' or '$(MacroName)', where MacroName is a valid macro name.

Macro references are replaced with the value of the specified macro; for example, if the project is located in 'C:/media', a pre-build command of '$(ProjectDir)/test.bat' will become 'C:/media/test.bat' when the pre-build command is executed.

Valid macros are list in the table below.

Macro	Description
ProjectName	The name of the current project, as specified in the project properties.
ProjectDir	The directory where the current project is located.
AudioSourceDir	The current audio source directory, as specified in the project properties.
BuildDir	The current build directory, as specified in the project properties.
BuildFiles	The list of files created by the build process. Each item is contained between ' characters and the list is space delimited.
CurrentPlatform	This indicates the target platform to be built. Possible values include: • 'PC' for PC (includes Windows, Mac, Linux, Solaris) • 'XBOX' for Microsoft Xbox • 'XBOX360' for Microsoft Xbox 360 • 'GC' for Nintendo Gamecube • 'WII' for Nintendo Wii • 'PS2' for Sony PlayStation 2 • 'PSP' for Sony PSP • 'PS3' for Sony PlayStation 3
CurrentLanguage	The current language selected.

Closing a Project

To close the current project:

1. From the File menu select Close Project
2. The project is now closed.

Events

Once a project file has been created and the project properties set, then construction of the event content can begin. Typically a project will contain:

- Event Groups
- Events
- Layers
- One or more audio files
- Parameters

There is also a special type of event called a 'simple event'. FMOD Designer 2010 and the FMOD Event System have an optimized code and data path for 'simple events' that uses less memory and less CPU at runtime.

Event Groups

As mentioned in Chapter Two, events are the sounds that are triggered by the actions occurring within the game. A project can add as many events as required. To provide organization and efficiency when loading data into memory during game execution, event groups are required.

Each event group contains events or other event groups, making it possible to create a full hierarchy to organize your game audio. All events and event groups are visible in the Event Groups browser pane, as shown in the figure below.

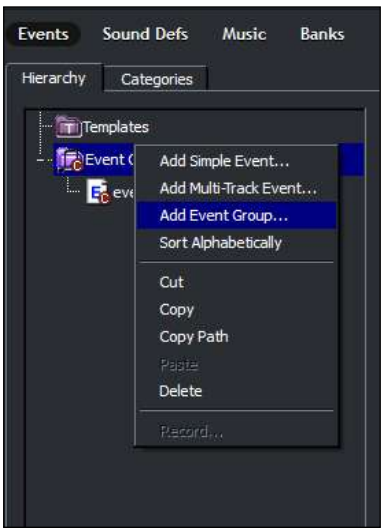


Figure 3-6: Creating an Event Group

Adding a New Event Group
To add a new event group:

1. Click on the Events button.
2. Right click the Groups pane. A Menu pop-up window should appear (as shown in the figure above).
3. Select the menu item 'Add event group…' A pop-up window will ask for a name to be entered.
4. Enter a name and press the [enter] key.

The new event group name should appear in the Groups pane and in the Properties panel.

Deleting an Existing Event Group
To delete an existing event group, complete the following steps:

1. Click on the Events button.
2. Right-click on the Event Group (in the Event Groups pane) to be deleted. A Menu pop-up window should appear.
3. Select the menu item 'Delete'. A pop-up window will ask for confirmation.
4. Click the 'YES' option.
5. The event group and its contents will be deleted. A pop-up window may ask whether unused objects should be deleted. See ‘Remove unused Sound Defs and Sound files’ for information about unused objects.

Setting an Event Group's Properties
To modify the properties of an event group, complete the following steps:

1. Click on the Events button.
2. Click on the desired event group in the Event Groups view.
3. When selected, the properties of the 'event group' will be displayed in Event Properties pane.
4. Enter the desired value into the 'Name', 'Notes' or 'User Property' value cell and press the [enter] key.

When changing the name property, the new event group name should appear in the Event Groups pane and the Event Properties pane.

Event Categories

Events can be organized into categories as well as groups. The Event Categories pane can be seen in the Event Browser pane (click on the 'Category' View).

Unlike event groups, an event category allows the sound designer to change the volume and pitch of all events within the category in one location. This is very useful when fine-tuning the relative mix of many events.

There are two default event categories that cannot be deleted - the master and music categories. The master category represents the top level of the hierarchy and contains all other categories.

The music category has a different, yet important purpose. Microsoft requires a title's soundtrack be replaceable with the end-user's own music files. The Xbox 360 allows end-users to import their own music using the Xbox 360 dashboard.

To comply with this requirement, all non-game related (soundtrack) music should be placed in the 'music' category. This allows the FMOD engine to automatically suspend the playback of audio in the music category and instead use the end-user supplied music tracks.

When the end-user disables their own music from the dashboard, FMOD will automatically resume the playback of the events in the music category.

All events are visible in the event category tree view, as shown in the figure on the next page.

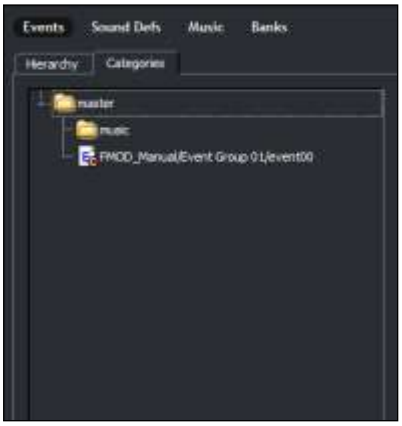


Figure 3-7: The Event Category View

Adding a New Event Category
To add a new event category:

1. Click on the Events button.
2. Click on the Categories tab
3. Right click the Categories pane. A Menu pop-up window should appear.
4. Select the menu item 'Add event category…' A pop-up window will ask for a name to be entered.
5. Enter a name and press the [enter] key.

The new event category name should appear in the Categories pane and the Event Properties pane.

Deleting an Existing Event Category
To delete an event category, complete the following steps:

1. Click on the Events button.
2. Click on the Categories tab
3. Right-click on the Event Category (in the Categories pane) to be deleted. A pop-up window should appear.
4. Select the menu item 'Delete'. A pop-up window will ask for confirmation.
5. Click the 'YES' option.
6. The Event category and any categories it contains will be deleted. Any events it contains will be placed in the Master Category

Setting an Event Category's Properties
To modify the properties of an event category, complete the following steps:

1. Click on the Events button.
2. Click on the categories tab.
3. Click on the desired event group in the Categories pane.
4. When selected, the properties of the 'Event category' will be displayed in the Properties pane.
5. Enter the desired value into the property value cell and press the [enter] key. All Event category properties are explained in Chapter Four, Reference Guide.

When changing the name property, the new event category name should appear in the hierarchy tree view and above the properties panel.

Maintaining Events

Starting a new project will create a default event group that contains a default event. This and any other event can be selected from the Hierarchy tree view.

Adding an Event to an Event Group
Events are added to an event category in the following manner:

1 Right-click the event group (in the Event Browser pane) that will contain the event. A Menu pop-up window should appear.
2 Select the menu item 'Add event…' A pop-up window should appear.
3 Enter the name of the event into the 'Name' Field.
4 Select the option 'Do not use a template'. (Use of templates is discussed later in this chapter).
5 Press the OK button.

The event should now appear in the Event Browser pane.

Deleting an Event from the Event Group
To delete an event, complete the following steps:

1 Click on the Events button.
2 Right-click on the Event (in the Event Browser pane) to be deleted. A Menu pop-up window should appear.
3 Select the menu item 'Delete'. A pop-up window will ask for confirmation.
4 Click the 'YES' option.
5 The event will be deleted. A pop-up window may ask whether unused objects should be deleted. See ‘remove unused Sound Defs and Sound files’ for more information about unused objects

Adding an Event to a Event Category
All events are placed in the master category by default. To place an event in an event category:

1 Click on the Events button.
2 Click on the 'Categories' hierarchy button.
3 Click (and hold) the event to move. Drag and drop the event into desired event category.
4 The event is now placed in the new category.

Setting an Event's Properties
To set the properties of an event:

1 To expand the Event Group, double-click the Event Group (from the Event Hierarchy Tree) in which the event is placed.
2 Select the Event.

3 The Event's properties will be displayed in the event / event group properties panel. This panel is shown in the figure on the next page.

4 To change the Event's properties, simply replace the value in the appropriate 'Value' cell and press the [enter] key. An explanation of each property is given in Chapter 4: Reference Guide.

Figure 3-8: The Event Properties Pane

Simple Events

Simple Events are the most basic sound producing object in FMOD 2010. A Simple Event can consist of nothing more than a single Sound File. The Event structure provides it with the necessary information to allow a coder to trigger the sound in-game and for FMOD to package up the sound file and its properties as part of the build process.

Simple Events have a set of properties that define how they play and in most ways function in the same manner as Sound Defs. It is possible to create a simple event that contains multiple sound files and define how FMOD 2010 will access those sound files.

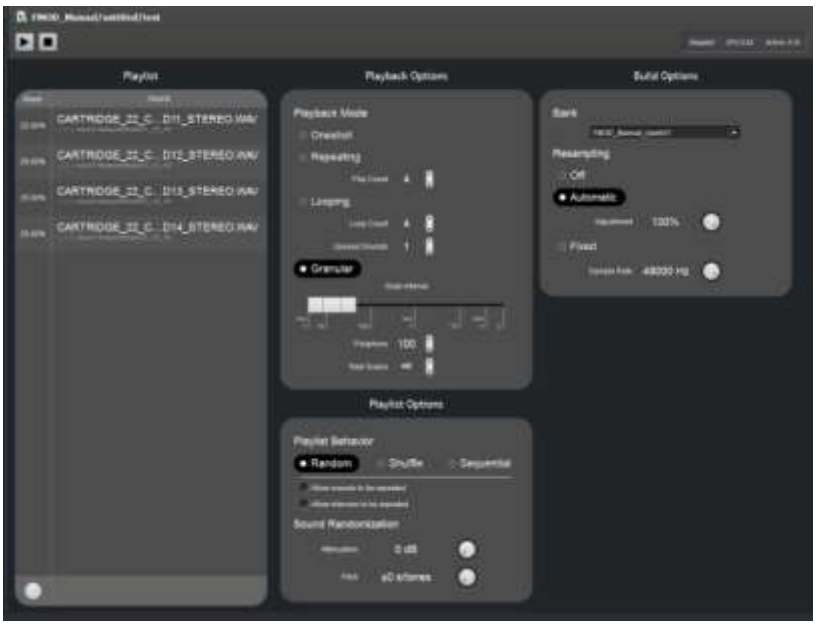


Figure 3-9: The Simple Event Editor Pane

Creating a Simple Event

A Simple Event can be added in the following way:

1 If not already selected, view the Event's View by clicking on the Events Button
2 Right-click in the Events Browser pane and select the menu option 'Add Simple Event…'.
3 A new Simple Event will be added and the Event Editor area will automatically display the Simple Event Editor Pane

The first step in working with a Simple Event is to add some sound files.

Adding Sound Files

Sound Files can be added in the following way:

1 If not already selected, select the Simple Event by clicking on the Event in the Events Browser Pane.
2 Within the Playlist box Right-Click and select the menu item ‘Add Sound…’
3 A pop-up browser window will allow you to navigate and select the desired sound files.
4 The selected Sound Files should now be displayed within the Playlist box as per Figure 3-9

Notes:

- It is also possible via the Playlist Box to add an instance of silence which can be a useful tool
- It is possible to add multiple Sound Files to a Simple Event

Deleting Sound Files

Sound Files can be deleted in the following way:

1 If not already selected, select the Simple Event by clicking on the Event in the Events Browser Pane.
2 Within the Playlist box select the Sound File you wish to delete.
3 Press the 'Delete' key on the keyboard
4 The selected Sound Files should now be deleted from the Playlist

By default when there is more than one sound in the Playlist FMOD 2010 will select a Sound File at random every time the event is triggered. This behavior can be altered via the Playlist options box. It is possible to set the playlist to play randomly, sequentially as displayed in the Playlist or to shuffle the order and then play sequentially.

Note:

The displayed list will not visually alter to match these selections, but playback behavior will change.

When random playback is selected the frequency of how often any one sound file is played is determined by the percentage values displayed on the right hand side of the Playlist.

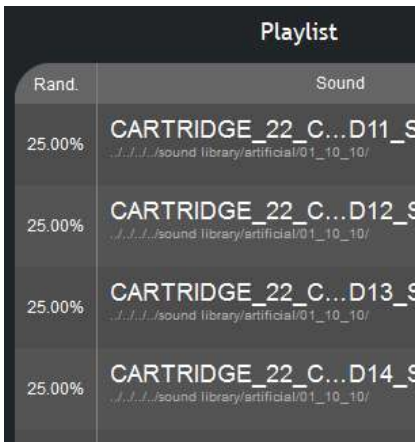


Figure 3-10: Playlist Randomization percentages

The weighting percentages can be altered in a couple of ways.

Altering Playback Percentages

Sound Files can have their randomization weighting percentage altered in the following way:

1 If not already selected, select the Simple Event by clicking on the Event in the Events Browser Pane.
2 Within the Playlist box Right-Click on the Sound File you wish to alter.
3 A pop-up menu should appear. Select ‘Set Percentage’
4 A second pop-up menu should appear
5 Input the desired percentage value
6 Hit ‘Enter’ on the keyboard or click on the close button on the pop-up menu

This will set the random weighting percentage value for the selected Sound File and adjust all the others so that all values combine to a total of 100%. The Right-Click menu on the Sound Files also allows the sound designer to lock the percentage values so they will not automatically adjust.

Locking Playback Percentages

Sound Files can have their randomization weighting percentage locked in the following way:

1 If not already selected, select the Simple Event by clicking on the Event in the Events Browser Pane.
2 Within the Playlist box Right-Click on the Sound File you wish to alter.
3 A pop-up menu should appear. Select ‘Lock Percentage’
4 The selected Sound File is now locked at the set percentage and a locked icon will indicate this status

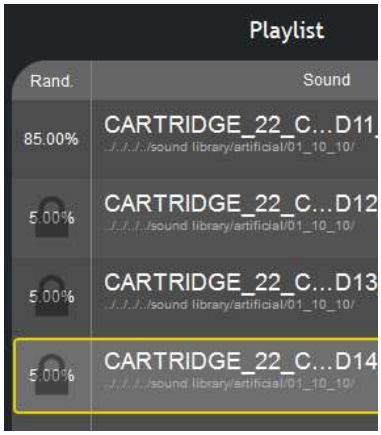


Figure 3-11: Playlist Randomization percentages

Note:

The percentages of all the Sound Files within a Playlist must always add up to 100%. In Figure 3-11 the bottom three Sound Files have been locked at 5% each. The remaining Sound File must be 85% as a result and cannot have its percentage altered unless another Sound File is unlocked first. Essential the automatic system requires at least two files to be unlocked so that alteration in one Sound File can be balanced out in the other to maintain the total of 100% overall.

The other method of altering randomization weighting percentages is to use the value adjustment dial at the bottom left of the Playlist. The dial will only function when one or more Sound Files are selected in the Playlist. As with the lock mechanic there always needs to be at least one Sound File not selected as adjustments of selected files need to be balanced out in the values of non selected files to maintain the 100% overall total.

Figure 3-12: Value Adjustment Dial

Altering Playback Percentages via Adjustment Dial

Sound Files can have their randomization weighting percentage altered in the following way:

1 If not already selected, select the Simple Event by clicking on the Event in the Events Browser Pane.
2 Within the Playlist Click on the Sound File you wish to alter.(Multiple selections can be made via Shift-Click or CTRL-Click)
3 Click and hold on the Value Adjustment Dial
4 Move the mouse in either a circular or up-down/left-right motion to turn the dial
5 The displayed values for the randomization weight percentages will alter accordingly
6 Percentages will remain at whatever value was displayed when dial movement is ceased.

Loop Options, Playlist Options and Build Options all contain parameters that allow a sound designer to define how the event will play and how it is formatted within the overall project. Refer to Chapter 4 for a complete list of Simple Event functions and description.

Multi-Track Events

What is a Multi-Track Event?

A Multi-Track Event is a combination of sounds layered together to create a more complex sound event. Multi-track Events utilize Sound Defs which in most ways function in the same way as Simple Event and have similar Properties. Sound Defs are placed in Layers within a Multi-Track Event.

What is a Layer?

An event is not limited to playing a single sound at a time. In fact, it is possible for an event to play many sounds, either simultaneously, using crossfades or in a sequenced fashion. Event Layers are part of Multi-Track Events.

This feat is made possible by splitting an event into multiple 'Layers'. Layers also allow a sound in an event to be modified (using effects) whist the playback of another sound in the same event is unaffected.

Sounds must be attached to an event layer. FMOD Designer 2010 will automatically display the Event layer information when a single event is selected in the Events Browser view

The figure on the next page shows a layer called 'layer00'. To the left of the layer name are five icons, being (from left to right):

- 'The disable icon'. When clicked, the current layer will be ignored – the layer will not contribute to the CPU load (unlike the mute icon), nor will the layer be included in the output file if the project is built. The disable icon status is linked to the 'Target Platform' selection, allowing sound designers to select which layers are included in each target platform.
- 'The solo icon'. When clicked the current layer will be heard in isolation, any other layer not in 'solo' mode will be inaudible and disabled from editing.
- 'The mute icon'. When clicked, the mute icon will silence all 'Sound Def Instances' on the layer. This is useful when auditioning events with multiple layers.
- 'The sound lock icon'. When clicked, the sound lock icon will disable any repositioning of the Sound Defs in the layer.
- 'The effect lock icon'. When clicked, the effect lock icon will disable any changes being made to the effects on the layer.

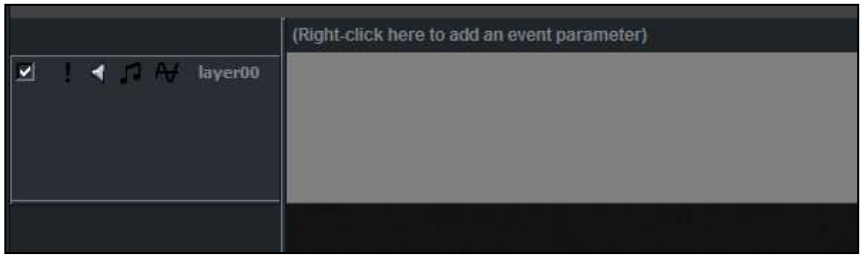


Figure 3-13: An Event Layer called 'layer00'

Note:

Muted layers are locked from modification.

Adding a Layer

A layer can be added in the following way:

1. If not already selected, view the Event's layers by clicking on the Event in the Event Groups in the Events Browser Pane.
2. Right-click on the layer panel and select the menu option 'Add layer'.
3. A new layer will be added.

Deleting a Layer

To delete an existing layer:

1. If not already selected, view the Event's layers by clicking on the Event in the Event Groups in the Events Browser Pane.
2. Right-click on the layer name and select the menu option 'Delete layer'. A pop-up window for confirmation should appear.
3. Press 'Yes'.
4. The layer should now be deleted. A dialog may ask whether unused objects should be deleted. See ‘Remove unused Sound Defs and Sound files’, for information about unused objects.

Modifying Layer Properties

A layer's properties can be modified in the following way:

1 If not already selected, view the Event's layers by clicking on the Event in the Event Groups in the Events Browser Pane.
2 Right-click on the layer name (the default layer is called 'layer00') and select the menu option 'Layer properties…'.
3 Change the 'Name', 'Control Parameter', and 'Priority' values to the desired value.
4 Press 'OK'.
5 If the layer's name is changed, the interface should now display the new name.

Note:

Only the parameter set as the 'control parameter' determines which Sound Def on the layer is audible.

Sound Defs and Sound Def Instances

Now that the event group, event and event layers have all been created, it is time to add some sounds! The Event Editor pane provides the functionality needed to add sounds to an event. A sound is represented as a box with rounded corners on the event layer this is called a 'Sound Def'.

The 'Sound Def' specifies a selection of audio files (and oscillators) and the general behavior of the audio to be used in the event. A 'Sound Def Instance' is an occurrence of the 'Sound Def'. There can be one or many such occurrences each one with slightly individualized properties. Both the properties of 'Sound Defs' and 'Sound Def Instances' are described in Chapter Four.

Once a Sound Def is added to an Event layer it effectively becomes a unique instance. In most cases users will probably not utilize multiple instances of the same Sound Def. For this reason it is easier to refer to the term Sound Def under normal work practices and Sound Def Instance when unique instancing is required.

Adding a Sound: Using a new sound file

In FMOD Designer 2010, audio files are placed within a layer as a 'Sound Def'. The simplest way to add a sound is:

1 If not already selected, view the Event's layers by clicking on the Event in the Event Groups in the Events Browser Pane.
2 Right-click on the grey area of the layer to which the sound is to be added. A menu should appear. This is demonstrated in the figure below.

Figure 3-14: Adding a new Sound Def

3 Select the menu option 'Add Sound'. A pop-up window should appear.
4 Click the 'New Sound File' button. A pop-up window to select files should appear.
5 [Optional] The added files can be auditioned by selecting the file and pressing the play button
6 Select an audio file (or files) to add to the Sound Def. Press 'OK' to add the files.
7 Press 'OK' on the pop-up window.
8 The Sound Def should appear as a box inside the layer.

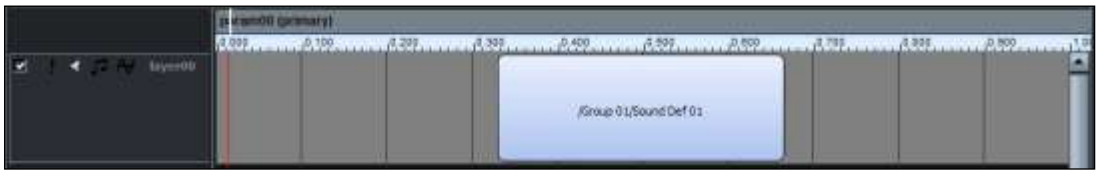


Figure 3-15: A Sound Def added to the layer

Note:

It is important to realize that a 'Sound Def' has been added to the event – not an audio file!

A Sound Def can contain multiple Sound Files and behaviours. If audio files are added directly to the layer, FMOD Designer will automatically create a new Sound Def with the audio file(s) in it. The complete Sound Def can be seen on the 'Sound Defs' View.

Adding a Sound: Using an oscillator

Not all sounds are generated using audio files. The FMOD EX engine supports a type of audio, known in FMOD Designer 2010 as an 'oscillator'. An Oscillator is a CPU generated signal such as, white noise, sine wave, etc. An Oscillator requires no memory space, but does use a small amount of CPU time.

To add an oscillator:

1 If not already selected, view the Event's layers by clicking on the Event in the Event Groups in the Events Browser Pane.
2 Right-click on the grey area of the layer to which the sound is to be added. A menu should appear. This is demonstrated in the figure below.

Figure 3-16: Adding a new oscillator Sound Def

3 Select the menu option 'Add Sound'. A pop-up window should appear.
4 Click the 'New oscillator' button. A pop-up window to configure the signal should appear.
5 Select the wave type and the frequency of the signal in Hertz. Each wave type is described in Chapter Four: Reference Guide.
6 Press 'OK' to add the oscillator.
7 [Optional] The added objects can be auditioned by selecting the object and pressing the play button.
8 Press 'OK' on the pop-up window.
9 An instance of the Sound Def should appear as a box inside the layer.

Adding an Existing Sound Def

Once a wavetable or Sound Def has been added to a project, it is possible to add an instance of the Sound Def to any event layer. This option is not available when using a new project, as the list of Sound Def will be empty!

To add a Sound Def instance to a layer, complete the following steps:

1. If not selected, view the Event to edit by clicking on the Event in the Event Groups view in the Events Browser pane.
2. Right-click on the grey area of the layer to which the sound is to be added. A menu should appear.
3. Select the menu option 'Add Sound'. A pop-up window should appear.
4. Select a sound from the list of available Sound Defs.
5. [Optional] The added files can be auditioned by selecting the file and pressing the play button.
6. Press 'OK' on the pop-up window.
7. An instance of the Sound Def will appear as a box inside the layer.

Modifying the Properties of an Instance

Once an instance of a Sound Def has been added to an event there are a number of properties that can be edited. It is important to note that these properties only refer to an instance of the Sound Def and not the Sound Def itself.

For example, a Sound Def may be added to a layer twice. One instance of the Sound Def may play the sound a single time (one-shot). The other instance may play the sound repeatedly (using a loop).

To modify the properties of a sound instance:

1. If not selected, view the Event to edit by clicking on the Event in the Event Groups view in the Events Browser pane.
2. Right-click on the Sound Def Instance to be modified. A Menu pop-up window will appear.
3. Select the menu option 'Sound instance properties…'. A pop-up window will appear.
4. Modify the properties' value to the desired settings. Each property in this pop-up window is discussed in detail in Chapter Four: Reference Guide.
5. [Optional] Press the play button in this pop-up window to audition the property changes made to the audio.

Resizing a Sound Def Instance

To resize the Sound Def instance:

1. Click the Event Editor button.
2. Move the mouse pointer to the left or right vertical edge of the instance box to be resized.
3. Click and hold to grab the edge of the Sound Def instance box.
4. Extend (or reduce) the size of the box with a left or right movement.

Figure 3-17: Resizing the Sound Def.

The Sound Def Instance box can also be expanded to fill all available adjacent space in a layer by holding [CTRL] and double-clicking the instance.

Repositioning a Sound Def Instance

To resize the Sound Def instance:

1. Click the Events button.
2. Press and hold the mouse button on the instance box to be repositioned.
3. Move the instance box to the desired location.
4. Release the mouse button.

Setting the Volume of an Instance

The volume of each sound instance object can be set independently. The volume is represented be a horizontal blue line within each sound instance. When this blue line is placed at the top of the instance no attenuation occurs. When placed at the bottom, the sound instance volume is fully attenuated (made silent). The blue line may be placed anywhere between these two extremes.

To set the volume of a sound instance, complete the following steps:

1. If no blue line is currently visible, click and hold the top border of the Sound Def box. Otherwise, click and hold the blue volume line.
2. While holding down the mouse button, drag the mouse to the desired position.
3. Release the mouse button.

Crossfading Between Instances

Crossfading is the practice of blending the transition between two sounds using volume sweeps. Creating a crossfade in FMOD Designer 2010 is incredibly simple, as the software automatically blends two Sound Def instances when they overlap on an event layer.

To create a crossfade:

1. Add a Sound Def instance to an event layer.
2. Position and size the instance on the layer.

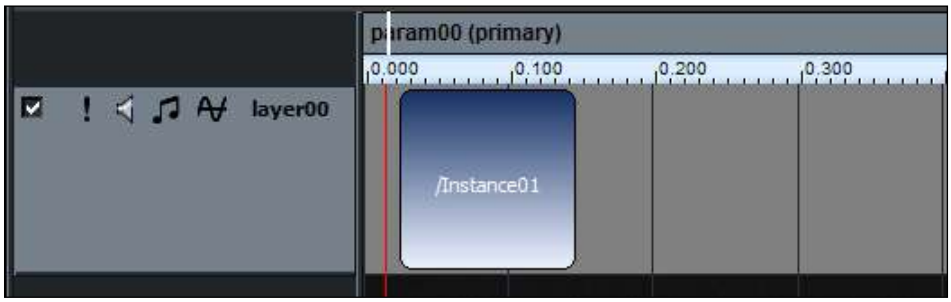


Figure 3-18: Adding the first Instance

3 Add another Sound Def instance to the same event layer.
4 Click (and hold) the second instance and overlap the first instance.
5 Once the instances are overlapped, two blue lines representing the volume change of each instance should appear. This is demonstrated in the figure below.

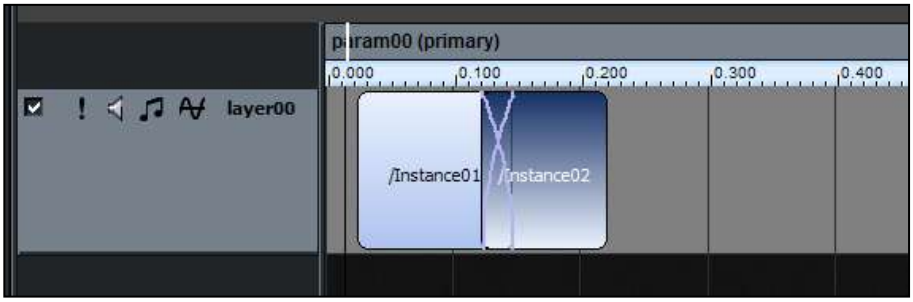


Figure 3-19: Creating crossfades

6 The type of crossfade can also be specified. The type of crossfade varies the relative rates of change in volume between the two instances. This is done by right-clicking the cross faded section. A pop-up window should appear.
7 Place the mouse over the menu option 'Crossfade type', a sub-menu should appear.
8 Select the desired crossfading style.
9 The blue lines should indicate the new crossfade style.

The sound of the crossfade is not just influenced by the crossfade type but also by the size of the overlap. The size of the overlap must be sized and positioned appropriately in relation to the parameter value. Parameters are explained later in the chapter.

Replacing the Sound Instance
To replace a Sound Def instance:

1 Click the Events Button.
2 Right-click on the Sound Def instance to change. A pop-up window should appear.
3 Select the menu option 'Replace Sound…'. A pop-up window should appear.

4 Select the new Sound Def and press the 'OK' button.
5 The Sound Def instance will now be modified.

Deleting an Instance
To delete a Sound Def instance:

1 Click the Events button.
2 Click on the Sound Def instance to delete.
3 Press the delete key, a confirmation menu should appear.
4 Select 'Yes'.
5 The Sound Def Instance will be deleted. A dialog may ask whether unused objects should be deleted. See ‘Remove unused Sound Defs and Sound files’, for information about unused objects.

Alternately, a Sound Def instance can be deleted by right clicking on the instance to delete and selecting 'Delete Sound' from the pop-up window.

Parameters

What is a Parameter?
An event parameter is a structure created by the sound designer to:

- To select which Sound Def instance(s) to play
- Manipulate an effect or sound property during run-time.

For example, an event simulating a car engine noise might use a parameter with the name 'rpm' with a range of 0 to 10,000. As the in-game engine revs harder, the 'rpm' parameter changes, signaling the event to use a different Sound Def to simulate the change in engine noise.

Parameters are optional. If an event does not need to be controlled by the game after it has started, parameters can be ignored. Most events in a game behave this way, especially simple one-shot sounds. Parameters must be manually added by the sound designer or included in an event template. Parameters are not included in an event by default.

Note:

Some older versions of FMOD Designer 2010 do add a default parameter called ‘param00’ to new events.

The effect of a parameter change on the audible sound is specified at design-time, providing the sound designer near complete control of the audio. The Programmer is only responsible for providing the parameter's value in real-time (during the game's execution).

Common examples of parameter use would be, controlling the volume of the sound, changing the pitch of the sound, and setting the 3D position of the sound.

In FMOD Designer 2010, a parameter is visible as a 'ruler' above the layers in the Event Editor view. The sound designer may specify the range and unit dividers displayed for each parameter. By default, the range is 0.0 to 1.0 but this can be changed to whatever the sound designer desires. This is shown in the figure below

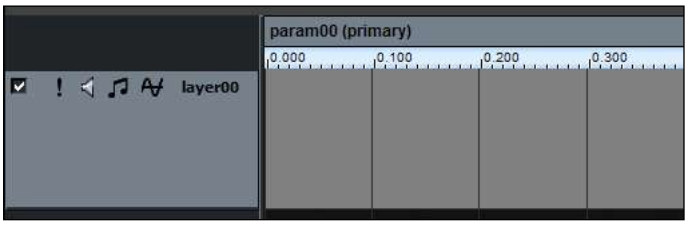


Figure 3-20: A Parameter called 'param00'

Most users new to FMOD Designer 2010 assume this ruler represents a 'timeline' to play the Sound Defs. While many program editors use this timeline philosophy, assuming FMOD Designer 2010 works this way is incorrect!

In its default form, the ruler represents a parameter and the cursor shows the current value of the parameter. The cursor will not move by itself, unless you specify a velocity in the parameter properties. The velocity property of a parameter is useful in creating time-based behaviors for sequenced events. The use of velocity-based parameters is demonstrated in Chapter Five.

An event can use up to 16 parameters (including the automatic parameters 3D distance, event angle and listener angle). Using three or more parameters can make it difficult to audition the event behavior in FMOD Designer 2010. The event would be better auditioned inside the game title.

Automatic Paramaters

Within FMOD there are three special parameters that are automatically updated by the FMOD engine. These parameters do not require programmer intervention to update their values. Instead the sound designer must set the event 'Mode' property to 3D and add one or more of the following parameters. The automatic parameters are:

- 3D Distance – Represents the distance between the listener and the event. This can be used to implement custom roll volume off and (or) 2D/3D pan morphing.
- 3D Event Angle – Represents the absolute angle between the event's direction and the location of the listener (ranging 0 to 180°). This can be used to implement custom cone roll off behavior.
- 3D Listener Angle – Represents the angle between the listener's direction and the location of the event (ranging -180° to 180°). This can be used to implement custom head related transfer functionality.

To add an automatic parameter, right-click the parameter panel and select the desired 3D parameter to add.

Adding a Parameter
To add a parameter:

1. If not already selected, view the Event in the Event View by clicking on the Event in the Event Groups in the Event Browser pane.
2. Right-click on the parameter bar (the panel above the 'ruler'). If there are currently no parameters, you can right-click on the message ‘Right-click here to add an event parameter’. A Menu pop-up window will appear.
3. Select the menu item 'Add parameter'.
4. Press 'OK'.
5. The parameter will be added, and assigned a generic name such as 'param00'. To rename this parameter, modify its 'name' property.

Modifying the Properties of a Parameter
To modify a parameter's properties:

1. If not already selected, view the Event in the Event View by clicking on the Event in the Event Groups in the Event Browser pane.
2. Right-click on the parameter bar (the panel above the ruler). A Menu pop-up window will appear.
3. Select the menu item 'Parameter properties…'. The parameter properties pop-up window will appear.
4. Change the value of the appropriate property. Each of these properties is described in detail in Chapter Four: Reference Guide.
5. Press 'OK'.
6. The properties have been changed,

Figure 3-21: The Parameter Properties Pop-up Window

Assigning a Parameter to an Event Layer

As stated earlier, the control parameter selects which sound instance to play. Each layer can have its own control parameter, meaning the playback of Sound Defs on separate layers can be controlled independently using multiple parameters. This feature of FMOD Designer 2010 is demonstrated in the engine example of Chapter 5, Tutorial Projects.

To select the control parameter for a layer:

1 Click the Events button.
2 Right-click the name panel of the layer to be changed. A menu should appear.
3 Select the menu option 'Layer properties…' A pop-up window should appear
4 Select the desired parameter from the 'Control parameter' drop-down menu
5 Press the ‘OK’ button
6 The selected parameter will now act as the 'control parameter' for this layer.

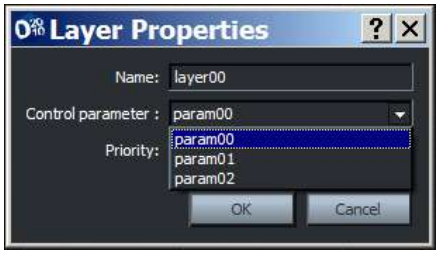


Figure 3-22: Layer Properties pop-up window

Effects

FMOD provides a number of free DSP (Digital Signal Processing) effect plug-ins. These plug-ins process the digital signal to change the characteristics of the audio in real-time. FMOD includes effects such as low pass filtering, distortion, flanger, etc.

Event layers can use one or more DSP effects the only limitation is the amount of CPU power available. Each effect placed on a layer may be individually enabled or disabled on a per platform basis. To do this, select the desired platform using the platform dropdown list and then set the 'Enabled effect' checkbox to the appropriate on or off value

A complete reference to the FMOD effects is given in Chapter Four, Reference Guide.

Adding an Effect

An effect can be added to layer by:

1 Right-click on the layer panel. A Menu pop-up window should appear. This is demonstrated in the figure below.

Figure 3-23: Adding an Effect

2 Select the menu option 'Add effect'. A pop-up window should appear.
3 Select the effect to add.
4 The effect panel should appear in the Layer panel and a parameter automation line is also added to the layer

Each visible effect parameter is designated a color. In the figure below, the volume effect parameter is set to -10.17 dB (the value of effect parameter is displayed as a horizontal red line)

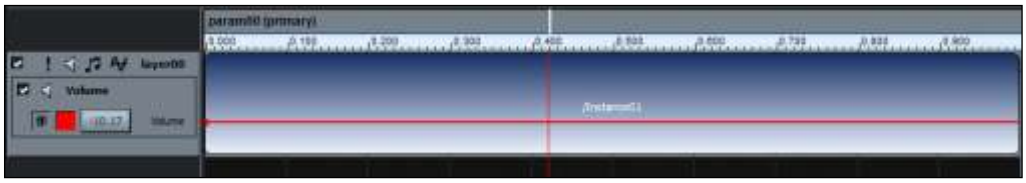


Figure 3-24: The Effect Parameter

Effects can also be copy, cut or pasted from one layer to another.

Take the following steps to copy (or cut):

1. Right-click on the effect panel to copy or cut. A Menu pop-up window should appear.
2. Select the menu option 'Copy effect' (or 'Cut effect').
3. Apply the effect by right-clicking the target layer panel. A Menu pop-up window should appear.
4. Select the menu option 'Paste effect'.
5. The pasted effect parameter should appear in the Layer panel and a parameter automation line is added.

Selecting the Effect Properties

Some effects such as 'FMOD Flange' have multiple properties that can be adjusted. FMOD Designer 2010 allows the sound designer to choose which effect properties are manipulated by the effect automation line. To select an effect property to set (or automate):

1. Right-click on the Layer Effect. A Menu pop-up window should appear.
2. Select the menu option 'Effect properties…'. A pop-up window should appear.
3. Select the effect parameters to be visible in the Layer pane.
4. Press 'OK'.
5. The effect parameters will now be visible within the layer in the Event Editor Pane.

Not all effects have multiple properties. A comprehensive guide to FMOD effects can be found in Chapter Four, Reference Guide.

Setting the Effect Property

In most circumstances, the parameters of an effect will be constant during the playback of an event. The static value of an effect property can be adjusted by:

1. Clicking on the parameter value box (to the left of the parameter name). A vertical slider should appear.
2. Set the vertical slider to the desired value.
3. To make the vertical slider disappear simply click on the layer space.

Creating an Effect Envelope

Not only can a parameter be used to select which sound instance in an event layer to play, parameters can also be assigned to automate various effects.

Many interesting sounds can be created by dynamically adjusting the values of these effect properties during run time. Using the event editor, values of effect properties can be specified as an envelope in relation to the event parameter value.

To modify an effect envelope:

1. Right-click on the line representing the effect property. The color of the line is denoted in the effect property list in the Layer Panel. A Menu pop-up window should appear.
2. Select the menu item 'Add point'
3. A point will appear on the effect envelope. Using the mouse, grab the point and move it to the desired position. Moving the point left or right changes the effect parameter's relationship with the event parameter, while moving the point up or down changes the parameter value.

The figure below demonstrates an effect envelope that produces a dynamic volume change. As the event parameter moves from 0 to 0.3, the volume effect property changes from approximately 50% to 100%.

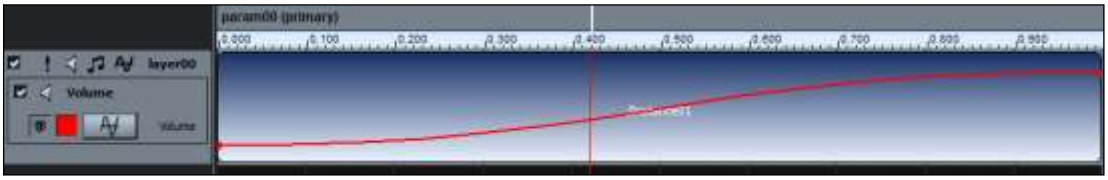


Figure 3-25: A Dynamic Effect Parameter

FMOD Designer 2010 supports four different transition styles between the points of the envelope. The available curve shapes are:

- Linear
- Logarithmic
- Flat Middle
- Flat Ended

To change the curve shape between points:

1 Right-click the point in the envelope effect in the Layer. A Menu pop-up window should appear.
2 Select the curve shape desired. For example, Curve Shape: Flat ended.
3 The transition between the points to the left and right of the curser on the envelope will change accordingly.

Removing an Effect Envelope Point
To remove a point on the effect envelope:

1 Right-click the envelope point to be removed. A Menu pop-up window should appear.
2 Select the menu item 'delete point'
3 The point will disappear and the envelope will be redrawn

Changing the Colour of an Envelope
To change the color of an effect envelope:

1 Click on the event to be changed in the Event Groups or Categories panes.
2 Right-click the effect panel of the envelope to be changed.(If the event has more than one property, right click on the one whose colour is to be changed.) A Menu pop-up window should appear.
3 Select the menu option 'Envelope color…'. A color palette pop-up window should appear.
4 Select the desired color from the color palette and press the 'OK' button.
5 The effect event should now display the new color.

Reordering the Signal Chain
To chain an effect's position in the signal chain:

1 Click on the event to be changed in the Event Groups or Categories panes.
2 Right-click the effect panel of the effect to be moved. A Menu pop-up window should appear.
3 Select the menu option 'Move effect up' or 'Move effect down'. The layer will be redrawn to display to new positioning. Repeat this step as required.

Note:

Reordering the signal chain requires the layer to include two or more effects.

Removing an Effect
To remove an effect from a layer:

1 Click on the event to be changed in the Event Groups or Categories panes.
2 Right-click on the effect to be removed. A Menu pop-up window should appear.
3 Select the menu option ' Delete effect'.
4 The effect will be removed.

Event Templates

To increase workflow and reduce repetitive actions, FMOD Designer 2010 provides 'Event Templates'. Event templates allow the sound designer to create their own preset events or starting points for events. For example, a template could be created as a starting point for gunshot events, car events, etc.

Each event created using the template will share characteristics such as event properties, layers and effects. The sound designer can even choose which event properties are part of the template!

Template files are specified per project, rather than as a global (workspace) setting. This allows the sound designer to use different sets of templates for different projects. sound designers can use the 'Merge Templates...' command, to copy templates from an existing template file into the current project's template file.

Note:

As default, new projects will not contain a template file – as this avoids blanking any existing template file on the initial save.

Whilst templates are specified on a 'per project' basis, the templates are not stored in the project file, but instead the project contains the filename of a 'templates' file. The location and filename of the template is specified in the project property 'template file'. The template location and filename can be set as a relative path to the project's location. (In fact, this is recommended, as then it's easy to move the project and templates around as part of a source tree).

Creating a Template
Before a template can be created, an event must be made. This event will serve as the starting point for the derived events.

A template is created using the following steps:

1 Click on the Events button.

2 Right-click on the event that will become the template to open the context sensitive menu.
3 Select the ‘Create Template From Event’ context menu item
4 The event will now be displayed within a template folder at the top of the 'Event Hierarchy Tree.

Figure 3-26: Selecting the Properties for the Template

Creating an Event with a Template

To create an event using a template:

1 Click on the Events button.
2 Right-click on the Event Group that will hold the event, from the Event Groups pane. A pop-up window should appear.
3 Select the option, ‘Add event…’ A pop-up window should appear.
4 Enter the name of the new event.
5 Select the ‘Use this template’ button

Figure 3-27: Applying a template to an event

6 Select the template to apply from the drop-down list
7 Press the 'OK' button.
8 The newly created event will now use the same properties, number of layers and effects as defined in the event template.

Modifying the Template
Any changes made to the event template can be applied to all the derived events automatically.

To apply changes to the derived events:

1 Make changes to the event template.
2 Click on the 'Edit' option in the Menu Bar. A pop-up window should appear.
2 From the Project menu select the Apply All Template Changes menu item
4 All changes to the event template have been applied to the derived events.

You can quickly modify which attributes are active within the template by:

1 Selecting the template event in the event hierarchy view.
2 Clicking the button within the 'value' field of the property. A checked checkbox means the attribute is active within the template, while an empty checked shows the attribute is not applied with the template. Properties that do not have a checkbox cannot be included within templates (i.e not supported).

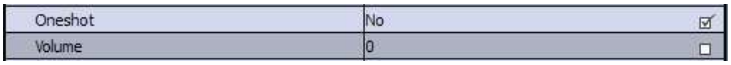


Figure 3-28: An inactive and active property within a template

Protecting Properties from a Template
All properties included in the template are applied to the derived event by default. However, the sound designer can choose to protect an event property from being overwritten by a template. To do this, the event must be using a template. The properties of the event that are provided by a template will be displayed in a blue row, with a lock icon in the value field.

Figure 1: The pitch randomization property is derived from a template

To stop an attribute of the template overwriting the event's property value, simply click the lock icon (of the property to protect). The lock icon should change to an open lock icon. This means the value is no longer derived from the template.

This technique provides a lot of flexibility and power in using templates. Many events may share a template, but each event may use a unique combination of properties.

Missing Templates

A template may attempt to derive its value from a template that is not contained within the project's template file. A red lock within the value field indicates property is a missing template. This situation can be resolved by:

- Changing template file
- Merging another template file into the current project
- Adding a new template event with the appropriate name
- Removing the template reference

Figure 3-30: Missing template

The Event Bulk Editor

The Event Bulk Editor allows for multiple Events to be selected together and for certain properties to be edited simultaneously across the selection. The Event Bulk Editor can also be used to audition multiple events simultaneously.

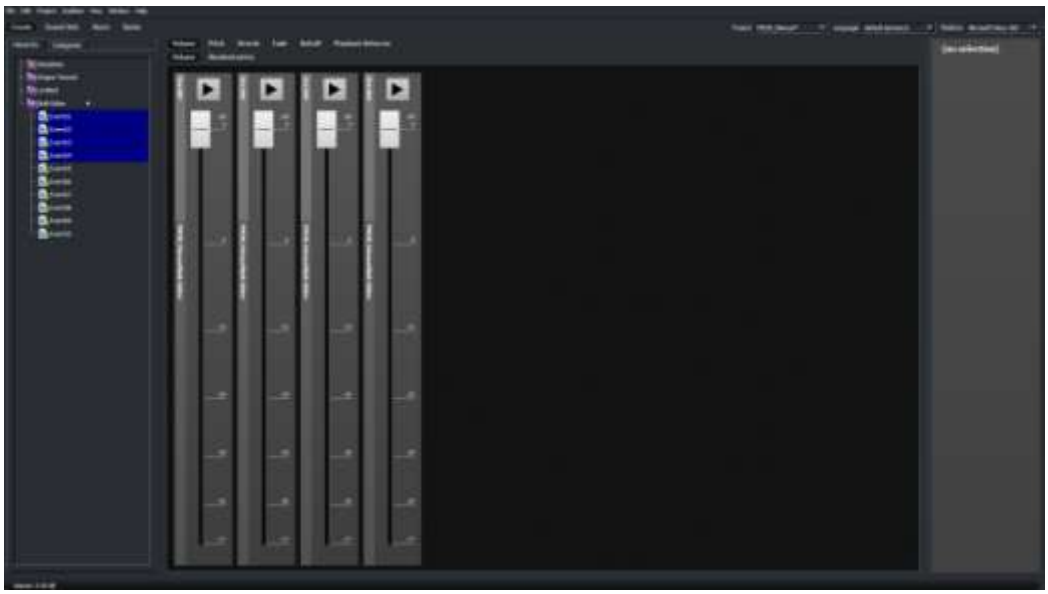


Figure 3-31: The Event Bulk Editor

Accessing the Event Bulk Editor

To display the Event Bulk Editor:

1. Click on the Events button.
2. In the Event Browser pane, SHIFT-Click or CTRL-Click on two or more Events.
3. The Event Bulk Editor will automatically display in the Event Editor pane area as in Figure 3-31.

The Event Bulk Editor has a range of Event properties that can be group edited. Buttons for each of the editable properties are positioned at the top of the Event Editor pane. Beneath the main property buttons are sub property buttons relevant to the currently selected main property. For Example in Figure 3-32 the main property displayed is Volume, so the sub properties of Volume and Volume Randomization are also displayed.

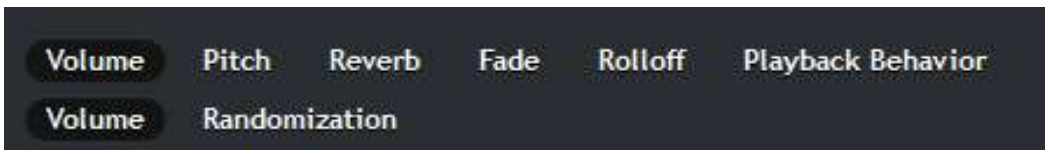


Figure 3-32: The Event Bulk Editor Property Buttons

Accessing the Event Bulk Editor Properties

To display the Event Bulk Editor:

1 (If not already in the Event Bulk Editor) Click on the Events button.
2 In the Event Browser pane, SHIFT-Click or CTRL-Click on two or more Events.
3 Click on the button that matches the desired Property to be edited.

Note:

The Orientation of the control sliders differs depending on which property is being edited. Volume, Pitch and Reverb use vertical controls, while Fade, Rolloff and Playback Behavior use horizontal controls

For a complete list of Bulk Editor functions refer to Chapter 4.

The Bulk Editor will display Events as controller sliders either vertically or horizontally dependant on the Property selected. The number of control sliders FMOD 2010 will display is dependent on screen resolution and window size. Should there be more Events selected than screen space available FMOD 2010 will automatically add a 'Birdseye' view of the relevant values to assist in editing.
The Birdseye view will be positioned horizontally along the bottom of the Bulk Editor for Properties that use vertical controllers and will display vertically down the right hand side of the Bulk Editor for Properties that use horizontal controllers.

Figure 3-33: The Event Bulk Editor Birdseye View

The Bulk Editor Birdseye view will automatically resize the Event Property values it displays dependant on how many total events are selected. For Example: Figure 3-33 contains 10 Event Property values which is only slightly more than can be displayed in the Event Bulk Editor, so the width of the Event Property values is only slightly narrower than the control sliders. If however there 30 Events selected then each Event Property Value would be very thin to allow for more to be displayed as in Figure 3-34.

Moving the Event Bulk Editor Birdseye View

To display the Event Bulk Editor:

1 (If not already in the Event Bulk Editor) Click on the Events button.

2 In the Event Browser pane, SHIFT-Click or CTRL-Click on multiple Events. (Enough to force the Birdseye View to display on your computer)

3 The Birdseye View includes both a Yellow Screen Display Area box as well as a standard navigation slider

4 Click and drag either the Yellow Screen Display Area box or the navigation slider in the direction of the Event Property values you wish to view.

5 The Bulk Editor will scroll both the Birdseye View and the Bulk Editor pane in the chosen direction to display the desired controls.

Figure 3-34: Event Bulk Editor Birdseye View Navigation

Once displayed within the Events Bulk Editor pane Event Property Controllers can be selected individually or group selected for bulk editing. The Event Properties pane will display information on the most recent Event Controller that was selected.

Figure 3-35: Event Bulk Editor Properties Pane

Figure 3-35 illustrates selecting multiple Event Controllers within the Event Bulk Editor. Selected Event Controllers are highlighted with a yellow box and their corresponding Event Property Values are also coloured yellow. Note that the very left hand Event Controller is also displayed in a lighter shade of gray. This is a separate highlight to indicate the most recent selection. It is this selected Event Control that has its Properties displayed in the Event Property pane on the far right hand side.

Group Editing Event Parameters

To display the Event Bulk Editor:

1 (If not already in the Event Bulk Editor) Click on the Events button.
2 In the Event Browser pane, SHIFT-Click or CTRL-Click on the Events you wish to edit.
3 In the Event Bulk Editor select the Property you wish to alter using the Property buttons
4 In the Event Bulk Editor SHIFT-Click or CTRL-Click on the Event Controllers of the Events you wish to edit.
5 Click and drag the control sliders to the desired values.

Note:

When multiple Event Controllers are selected dragging the control sliders will move all sliders relative to their starting positions. For Example: Selecting Event 01 which is set to 100% and Event 02 which is set to 50% and then dragging the value of Event 01 to the 50% position will result in a relative decrease in Event 02 leaving it at 0%.

Not all the controls in the Event Bulk Editor are slider controls. Playback Behavior and Curve Type under the Rolloff Property use selection buttons to set their values. These Properties can still be group selected and edited in the same manner as other properties they simply have a smaller range of values to display and so make use of a button mechanic.

Figure 3-36: Event Bulk Editor Playback Behavior

The Playback Behaviour function is slightly different in that it includes an extra information display. Different types of Playback behaviour are indicated by both the selected button turning black as well as the selected Events being assigned a colour value. The display strip at the bottom of the Event Bulk Editor indicates the overall ratio of behaviour types. In Figure 3-36 the top 4 Events have all been set to the ‘Steal Quietest’ Playback behaviour and assigned the green colour value. The ratio display strip is indicating that overall the number of Events with the same setting is a little more than 10% and that the Events assigned the red colour value make up the bulk of the Events currently in the Event Bulk Editor.

Basic Multi-Track Events

A 'Basic Multi-Track Event' is an event with special characteristics that allows the FMOD Event System to use an optimized code and data path that uses less memory and less CPU at runtime. The sound designer does not explicitly create a Basic Multi-Track Event. Instead, FMOD Designer detects a 'Basic Multi-Track Event' at build-time and builds it into the .FEV file differently than normal event.

FMOD Designer automatically treats an event as a 'Basic Multi-Track Event' if it meets the following criteria. A 'Basic Multi-Track Event' has:

- No parameters
- No effects
- No user properties
- Only one layer
- Only one sound (the sound definition may have multiple entries)
- No 'programmer' sounds

Note:

An event will be considered a 'Basic Multi-Track Event' only if **all** the above criteria are met.

Due to the optimized code and data path, Basic Multi-Track Events use significantly less resources than a standard event. By all means make 'feature' events as complicated as required, but Basic Multi-Track Events are highly encouraged where possible.

When FMOD Designer is deciding whether an event is basic or not, it takes into account the layer enable/disable checkbox. This means that it's possible to create an event with multiple layers but as long as only one of those layers is enabled, it will be considered a simple event.

The benefit of this is, it is possible to create a single 'Basic Multi-Track Event' with alternate content for different platforms within a single event. This technique is demonstrated in CHAPTER SEVEN: Advanced Techniques.

If multi platform usage is not required it is recommended users create Simple Events as these follow the same criteria as Basic Multi-Track Events and are more straightforward to work with.

Sound Defs

A Sound Def is more than just an audio file or collection of audio files and oscillators. Using the correct properties, Sound Defs can be used to add dynamic or random behavior to event playback. Using Sound Defs to create randomized behavior is demonstrated in later chapters.

Sound Defs can contain:

- Wavetables (audio files)
- Oscillators (generated signals)
- Programmer sounds
- Don't Play entries

The purpose of the first two categories has been explained in earlier sections. Programmer sounds and don't play entries must be explained further.

Programmer sounds allow the sound designer to create Sound Def entries that do not specify the source audio filename at design time. Instead the Programmer must supply the filename at runtime using a callback. A programmer sound entry is useful when using a large amount (10,000) of audio files and an algorithm to determine which audio file to use. Using a single event with a Sound Def containing a programmer sound entry is a very memory efficient method for handling large amounts of dialogue audio. This is discussed in greater detail later in the manual.

The 'Don't play' entry allows the sound designer to insert silent entries into a Sound Def. These entries are useful when combined with random entry selection. This allows the sound designer to add greater variation or randomness to dynamic events.

Creating Folders

A typical game title may use upwards of a hundred (some even use more than a thousand) Sound Defs. To keep this number of Sound Defs manageable, FMOD Designer 2010 provides 'Sound Def Folders'.

To create a 'Sound Def Folder':

1. Click the Sound Defs button.
2. Right-click on the Sound Def Browser pane. A Menu pop-up window should appear.
3. Select the option 'Add Sound Def folder…'. A pop-up window should appear.
4. Enter the desired name of the folder and press the 'OK' button.
5. The folder has been created.

Setting Sound Def Properties

In this section the steps required to set a Sound Def property are explained. All Sound Def properties are described in chapter 4, Reference Guide.

To set the properties of a 'Sound Def':

1 Click the 'Sound Defs' View.
2 Click on the Sound Def to be modified in the Sound Def hierarchy. The properties of the select Sound Def should appear in the upper properties panel.
3 Update the 'Value' cell of the property to be changed.
4 Press the [enter] key.
5 The Sound Def property will now display the new value.

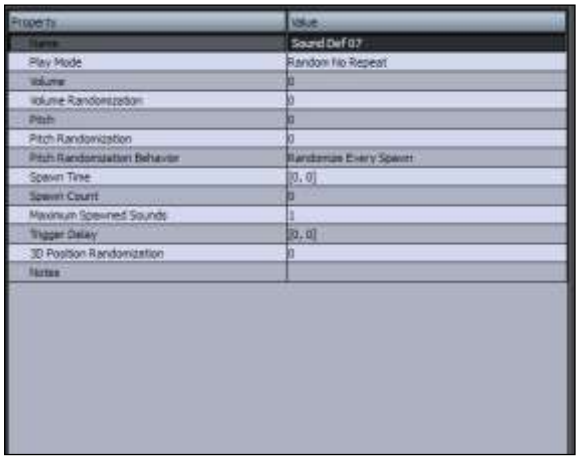


Figure 3-37: The Sound Def Properties Pane

Deleting Sound Defs and Folders

To delete a 'Sound Def' or 'Sound Def Folder':

1 Click the 'Sound Defs' View.
2 Right-click on the object in Sound Def hierarchy to be deleted. A Menu pop-up window should appear.
3 Select the option 'Delete'. A confirmation pop-up window should appear.
4 Press the 'Yes' button.
5 The Sound Def (or folder) has been deleted.

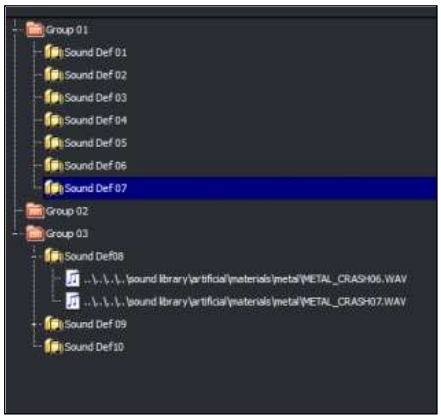


Figure 3-38: The Sound Def Browser pane

Banks

Banks represent the organization of the final package of audio data that is exported from FMOD Designer 2010 to the game title.

Adding a Bank

To add a bank:

1 Click the 'Banks' View.
2 Right-click on the 'Bank List' panel. A Menu pop-up window should appear.
3 Select the option 'Add bank'.
4 The new bank will appear in the ' Bank Browser' pane.

Figure 3-39: The Wave Bank Browser Pane

The Current Bank

Any audio files that are added to the project are automatically added to whichever bank is set as 'current'. A red music folder icon indicates the 'current' bank. All other banks have a blue music folder icon.

To set a bank as current:

1 Click the ' Banks' View.
2 Right-click on the bank to set as current, from the 'Bank Browser pane. A Menu pop-up window should appear.
3 Select the option 'Set as current bank'.
4 The bank icon should turn from blue to red, indicating it is now the current bank.

Setting the Bank Properties

A complete guide to the bank properties is given in Chapter Four, Reference Guide. To set the properties of a wave bank:

1 Click on the ' banks' View.
2 Click on the bank to edit, from the 'Bank Browser Pane'. The properties should appear in the ' Bank Properties' pane.

3 Click on the property to change, and set the 'Value' cell to the desired value. If the value property requires text input, press the [enter] key.
4 The new value should now be displayed.

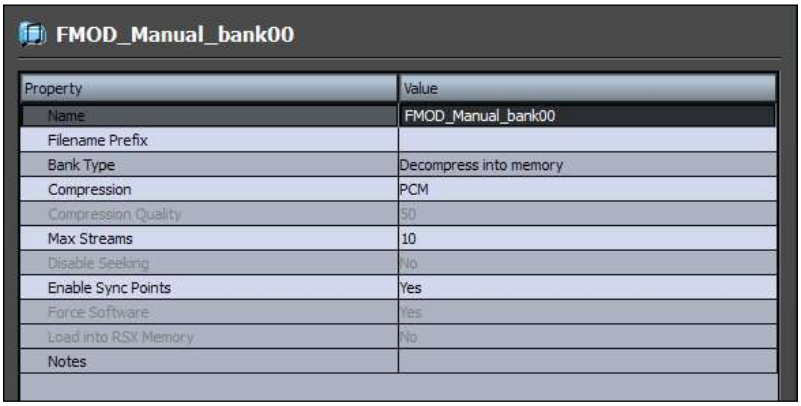


Figure 3-40: Bank Properties Pane

Increasing Workflow with Banks

FMOD Designer 2010 provides two simple functions to speed up the creation and set up of new banks.

The first function creates a simplified bank template. Once a bank is created, it can be used as a default setting for future banks. To do this:

1 Click on the ' Banks' View.
2 Create a bank and set its properties. These bank properties will be used as the new default settings.
3 Right-click on the bank to use as default, from the Bank Browser Pane. A pop-up window should appear.
4 Select the option 'Use properties as defaults for new banks'.
5 Any banks that are added from this point will be initialized with these properties.

The second function applies the properties of one bank to all the other banks.
To do this:

1 Click on the banks' View.
2 Right-click on the bank (with the desired properties), from the bank hierarchy tree panel. A pop-up window should appear.
3 Select the option 'Apply properties to all existing banks'. A confirmation pop-up window should appear.
4 Press the 'Yes' button.
5 All banks will now use these properties.

Moving Files Between Banks

All audio files can be moved from one bank to another. To move an audio file between banks:

1 Click on the Banks' button.
2 Click on the bank that contains the file to transfer, from the Bank Browser Pane
3 Select the file to transfer (or select multiple files using [SHIFT]+Click) and drag and drop the file (or files) into the desired bank.
4 The files have been transferred.

Deleting a Bank

Only empty banks can be deleted. To delete a non-empty bank, the files must be transferred into another bank.

To delete a bank:

1 Click the Banks button.
2 Right-click on the bank to be deleted from the Bank Browser pane. A Menu pop-up window should appear.
3 Select the option 'Delete Bank'.
4 The bank should disappear from the Bank Browser pane

Encoding Banks: Advanced Concepts

The following section provides technical details on specific encoding and decoding issues relating to bit rates, sample rates, compression quality, seamless looping and multi-channel encoding.

Bit Rates and Sample Rates for MPEG Data

The following table shows the available bit rates and sample rates available for MPEG data within FMOD Ex:

MPEG 1 Bit rates (kbps)	MPEG 1 Sample rates (kHz)	MPEG 2 Bit rates (kbps)	MPEG 2 Sample rates (kHz)
32	32	8	8
48	44.1	16	11.025
56	48	24	12
64		32	16
80		40	22.05
96		48	24
112		56	
128		64	
160		80	
192		96	
224		112	
256		128	
320		144	
384		160	

Should the user attempt to use a sample rate not listed, FMOD Designer 2010 will automatically resample the file (upwards) to the next valid sample rate. For example, a file with a sample rate of 15kHz will be resampled to 16hHz.

Quality and Bit Rate

What is the relationship between bit rate and the 'compression quality' property?

Within FMOD Designer 2010, the compression quality property is found in the Bank Property Pane. The relationship between bit rate and the compression quality property (when dealing with constant bit rate compression), is appropriately:

bit rate = quality * 3.2

Multi-channel MPEG Encoding

FMOD Designer 2010 is able to create MPEG files with up to 16 channels (eight stereo pairs). To do this, the build process:

- Encodes each stereo pair into fixed sized MPEG frames. The size of the frames is determined by the selected bit rate. The size of the frame must be a multiple of 16 bytes. To insure this a pad of 0 to 15 bytes is placed at the end of each frame.
- Interleaves a frame from each stereo pair into a multi-channel frame.

This process is illustrated in the figure below.

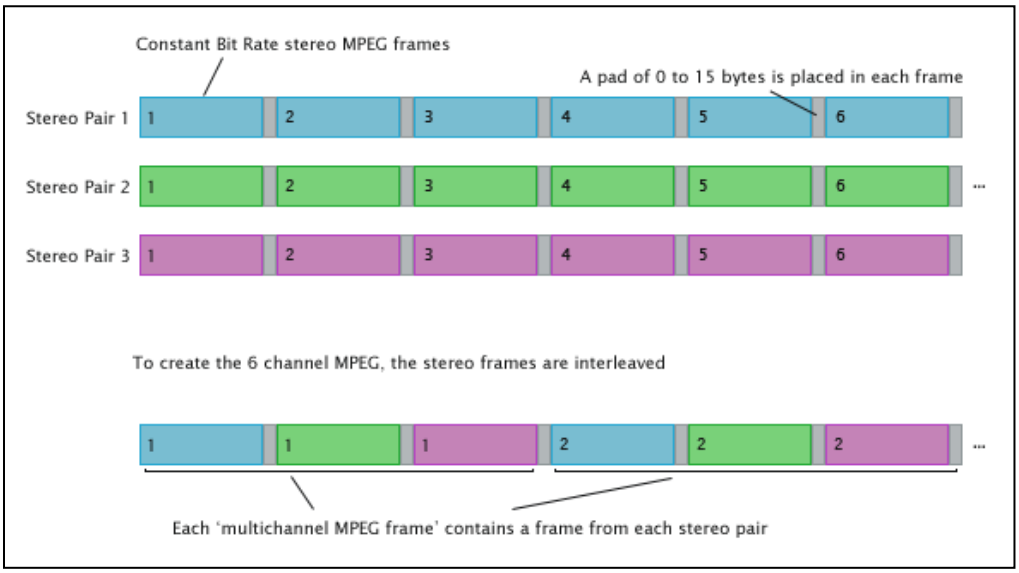


Figure 3-41: Encoding a multi-channel MPEG file

For example, let's consider a six-channel MPEG file using a constant bit rate of 128 kbps. The six channels are encoded into three stereo pairs. Each frame of stereo MPEG data is 432 bytes (including a 14 byte buffer). FMOD Designer 2010 then interleaves the stereo frames every 432 bytes into a multi-channel MPEG frame. The size of the multi-channel MPEG frame can be calculated as *frame size* × *Number of stereo pairs*. In this example, the multi-channel MPEG frame is 432 × 3, giving 864 bytes.

Encoding mp3 Files for Seamless Looping

Typically when an mp3 file is looped, an audible gap can be heard when playback loops back to the start. This gap is obvious when the loop requires a sample accurate stitching from the last sample to the first.

The gap occurs for a number of reasons, the two major factors being:

- MPEG 1 layer 3 encodes the audio data into frames of 1152 samples. If the audio data doesn't fill a frame (most importantly the last frame), the encoder will pad the frame with silent samples (some encoders will add an entire silent frame).
- The decoding of an mp3 frame is dependent on the previous frame. When a loop occurs, the decoder will require data from the last frame to smoothly loop back to the first frame.

Without special encoding, it is not possible for mp3 data to loop seamlessly – fortunately FMOD Designer 2010 does provide a method to do just that! The special encoder is automatically used if a Sound Def instance is set to loop and the wave bank compression property to 'MP3'.

Note:

If the Sound Def instance is set to 'one-shot' the standard mp3 encoding is used.

So what does FMOD Ex do to provide seamless loop of mp3 data?

Firstly, FMOD Designer 2010's encoder will resample and stretch the last frame to ensure that all 1152 samples of the frame are used. This will ensure the frame is not padded with silent samples.

When used on some sources, this process may cause a slightly audible pitch change artifact. If this is the case, sound designers are encouraged to repeat the audio within the file to increase the file size, so the time stretch distance becomes less significant. sound designers can also resize the length of their audio to a multiple of the frame size.

The table below lists the frame size for various formats.

Format	Frame size (samples)
MPEG 1	1152
MPEG 2 (2.5)	576
XMA	2048
VAG	28
GCADPCM	36

With the removal of any padding within the last frame, FMOD Designer 2010's encoder must then prime the first frame with data from the last frame. The last frame is then removed. This allows the FMOD Ex MPEG decoder to avoid issues of frame dependency between the first and last frame and provide a seamless loop.

In most situations the encoder and decoder will perform perfect looping of mp3 content. However some audible artifacts can be introduced, this is illustrated in the figure on the next page.

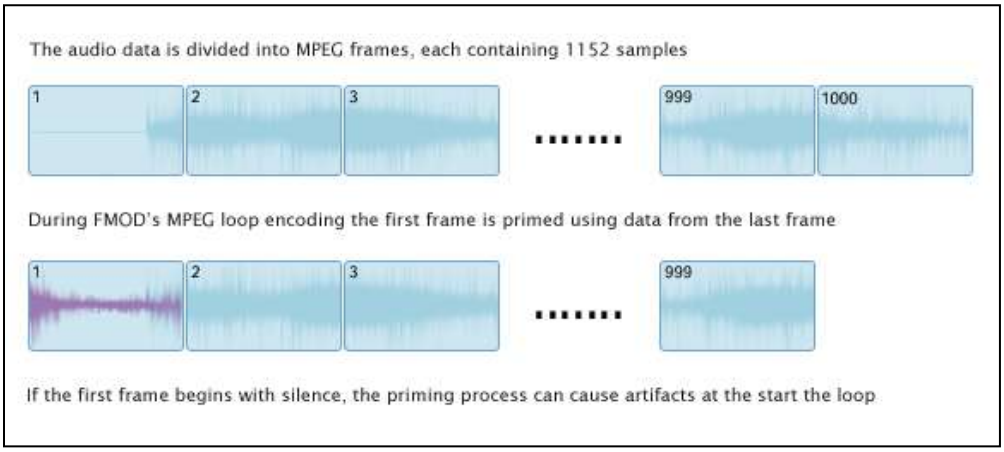


Figure 3-42: Encoding MPEG frames for seamless looping

When the first frame contains silence and the last frame contains an audible signal, the interpolation used in priming the first frame will result in an audible 'pop'. If sound designers require silence in the first frame of their loop, they should:

- Make sure the original PCM wave loops properly, or
- Pad the end of the file with a frame of silence

XMA Quality and Compression

As specified (in part) in the Xbox SDK documentation:

- The XMA encoder allows the sound Designer to specify a quality setting between 1 and 100, where:
  - 1 provides the highest compression level and the lowest quality, and
  - 100 provides the lowest compression level and the highest quality
- XMA's variable bit rate compression is content dependent, meaning compression ratios can vary greatly between pieces of content.
- This means the quality settings do not translate directly to specific compression ratios.
- The Xbox 360 Development Kit suggests a compression rate between 8:1 and 15:1 will provide adequate quality for most game audio assets.

Reverb Defs

The Reverb Defs pop-up window is under the Window menu in the Menu bar.
The 'Reverb Defs' pop-up window makes it possible to define the reverb used in the game title. This reverb should not be confused with the reverb DSP effect that is applied to individual event layers. The DSP effects are used to apply reverb to individual events (or event layers), whereas the reverb specified in this window are used by the programmer and may be applied to all events to simulate an environment.

For example, when the game character enters a cave, the 'scary cave' reverb would be applied to all sounds using the reverb defined in the 'Reverb Defs window.

The Reverb Defs window allows the sound designer to define the reverb and audition the reverb using the same sounds that will be used in the game title.

Adding an Empty Reverb Def
To add an empty Reverb Def:

1. From the Window menu select Reverb Defs
2. Right-click on the Reverb Def Browser pane. A pop-up window should appear.
3. Select the option 'Add an empty Sound Def…' A pop-up window should appear.
4. Enter the desired reverb Def name.
5. Press the 'OK' button.
6. The reverb Def will appear in the hierarchy.

Figure 3-43: The Reverb Def Browser Pane

Reverb Def Properties

Once a reverb Def has been created, its properties can be set. There are two ways to define a reverb's properties by,

Preset Reverb Properties
FMOD Designer 2010 includes 25 preset reverb settings (originally defined by Creative Labs). To use preset properties in a reverb Def:

1 From the Window menu select Reverb Defs
2 Select the Def to edit from the Reverb Def Browser pane. The properties of the Def should appear in the Reverb Def Properties pane.
3 Click on the property 'Reverb Preset'. A drop-down list arrow should appear.
4 Press the dropdown list arrow and select the desired preset. A confirmation pop-up window should appear.
5 Click the 'Yes' button.
6 The new properties should now appear in the Reverb Def Properties pane.

Figure 3-44: The Reverb Defs Properties Pane

User-Defined Reverb Properties

All reverb def properties may be customized. A comprehensive description for each property is given in Chapter Four, Reference Guide. To manually set the properties in a reverb Def:

1 From the Window menu, select reverb Defs.
2 Select the Def to edit from the Reverb Def Browser pane. The properties of the Def should appear in the Reverb Def Properties pane.
3 Click on the property to be changed. Enter the new value in the 'Value' cell and press the [enter] key. If this reverb Def is using preset settings, the preset property will automatically change to 'User'.
The new property value should now appear in the Reverb Def Properties pane
4 The Reverb Def property can be specified for all platforms, or on a per-platform basis. To set the property to 'all platforms', press the icon in the value column so it displays all solid squares -  . To set the property to be specific to the current target platform, press the icon in the value column so it displays a single solid square -  .

Deleting a Reverb Def
To delete a Reverb Def:

1 From the Window menu select Reverb Defs.
2 Right-click the Def to delete from the 'Reverb Def List' panel. A Menu pop-up window should appear.
3 Select the menu option 'Delete'. A confirmation pop-up window should appear.
4 Click the 'Yes' button.
5 The reverb Def should now disappear from the 'Reverb Def List' panel.

Event Auditioning

Basic Auditioning
With a valid event, event layer and Sound Def in place, the resulting audio can be auditioned. The audio can be heard by pressing the play button.

Figure 3-45: The Play, Stop and Key Off Buttons

'I pressed the play button, but there was no sound!'

The most common reason for an event remaining silent is caused by the layer's control parameter value. If there is no Sound Def instance for a given parameter value, the event will be silent. This scenario can be seen in the figure below.

Figure 3-46: No Sound Def Selected To Play

The vertical line that flashes red is known as the 'parameter cursor'. In the figure above, the parameter cursor represents the current value for the primary parameter (param00). No sound is audible, as 'layer00' does not contain a Sound Def instance for this parameter value.

There are two ways to remedy this situation. The first option is to move the parameter cursor. To move the parameter cursor, use the mouse to grab the parameter cursor and move it over the Sound Def box as demonstrated in the figure on the next page.

The second option is to move or resize the Sound Def box. Resizing the Sound Def has been explained previously, in the section 'RESIZING A SOUND DEF INSTANCE'.

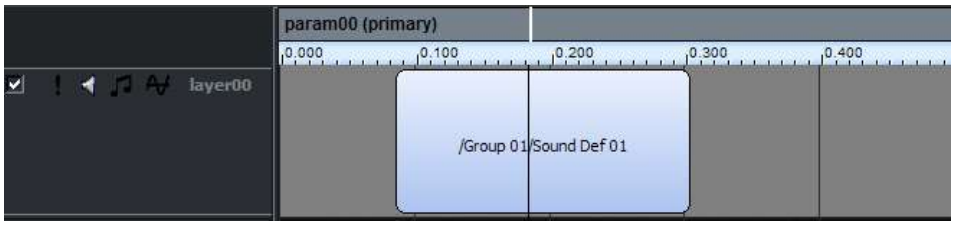


Figure3-47: Moving the Parameter Cursor

If the play mode is active (shown when the play button remains highlighted), a Sound Def instance will be heard as the parameter cursor is dragged above it.

If any of the effects that have automation are on layers controlled by the Event Parameter, the differences between sections of these automations will also be audible as the parameter cursor moves.

Using the Audition Tools

FMOD Designer 2010 provides a number of tools that can be used to simplify the auditioning process. These tools are:

- FMOD Engine Designer
- Audition 3D
- Surround Pan
- Cone Designer

It is important to understand these tools are not exported to or used by the game title, they are provided to help the sound designer in setting properties or auditioning events using a simple GUI.

Each tool is discussed in detail in Chapter Four, Reference Guide.

Using The Record Mode

FMOD Designer 2010 allows the sound designer to send the audio output to file (as a .wav file). This facility is useful for reducing CPU load, by rendering FMOD DSP effects to static files, instead of generating the effect in real-time. The resulting audio can then be imported back into FMOD Designer 2010 via a Sound Def.

Setting the Recording Directory

All audio files are written to a directory specified by the user. To specify the location of the audio files:

1 Open the audition menu and select Record Directory
2 Select the desired location (directory) for the audio files.
3 Press the 'OK' button. The target directory has now been set.

Starting and Stopping the Record Mode

To start the Recording Mode:

1 Click on the Events button
2 Open the Audition menu and select Record Output. The word Recording should appear in the top right hand corner of the Event Editor Pane.
3 Whilst in Recording Mode, each time an event is auditioned the audio output is written to a file. FMOD Designer 2010 will use the event's name as the filename, appended with an automatically incremented number (and the .wav extension.

Note:

Events or Sound Defs that use loops will not be rendered, as the Recording Mode is designed for one-shot sounds, and will ignore any looped audio.

To stop the Recording Mode:

1 Click on the Events button
2 Open the Audition menu and select Record Output. The word Recording should disappear from the top right hand corner of the Event Editor pane

Rendering Multiple Events to Audio Files

Multiple events can be rendered in a single batch operation. To render multiple events in the Recording Mode:

1 Click on the Events button
2 Open the Audition menu and select Record Output. The word Recording should appear in the top right hand corner of the Event Editor Pane.
3 Select the event(s) to render from the Event Groups pane. [CTRL] + Click or [SHIFT] + Click can be used to select multiple events.
4 Right-click the highlighted (selected) events. A Menu pop-up window should appear.
6 Select the menu option 'Record...'. A pop-up window should appear.
7. Enter the number of recording passes required for each event. The sound designer will generally choose to record multiple variations if the event contains randomized behavior,

such as randomized volume, pitch or Sound Def playback. Press the [enter] key. A progress pop-up window should appear.

8 The files will be written to the location as specified by the 'Recording Directory'.

Note:

Events or Sound Defs that use loops will not be rendered, as the Recording Mode is designed for one-shot sounds, and will ignore any looped audio.

Network Auditioning

When using FMOD Ex and FMOD Designer 2010, it is possible to audition properties setting from inside the game title whilst running on the target platform. This allows sound designers to hear their events, effects and Sound Defs on the target platform, therefore removing any guess work and time needed to fine-tune balances, effects, etc.

To establish a connection with a console (or FMOD AuditionClient), the programmer must first enable the audition network from the console or PC using the FMOD Ex engine. For more information about this, please refer to the FMOD Ex API Documentation.

Developers of PlayStation3 and Xbox 360 titles may also use the ready-made Audition Client application to establish a network connection between FMOD Designer 2010 and the target console.

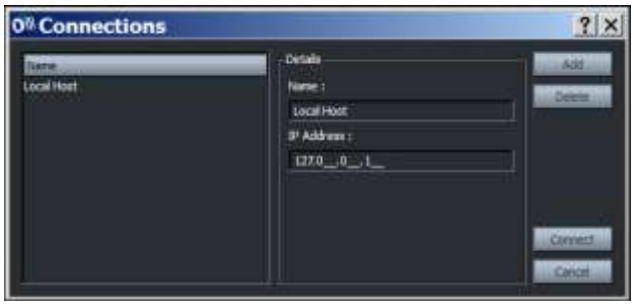


Figure 3-48: The Manage Connection pop-up window

Creating with a New Network Connection
To add a connection from the FMOD Designer 2010 to the target machine:

1. Open the Audition menu.
2. Select the option 'Manage Connections…'. A pop-up window should appear, as shown in the figure above.
3. Press the 'Add' button.
4. Enter a name for the connection in the 'Name' field.
5. Enter the IP address of the target machine in the 'IP Address' field.
6. Press the 'Connect' button. This will save the connection details.
7. FMOD Designer 2010 will now attempt to connect to the target machine. A pop-up window should appear to inform the user whether or not the connection is successful.
8. If successfully connected, the remote IP will appear in the FMOD Designer 2010's title bar.

Connecting with Existing Connection Details
To connect FMOD Designer 2010 using existing connection details:

1. Open the Audition menu
2. Select the option 'Manage Connections…'. A pop-up window should appear, as shown in figure 3-35.
3. Select the connection name from the 'Name' list.
4. Press the 'Connect' button.
5. FMOD Designer 2010 will now attempt to connect to the target machine. A pop-up window should appear to inform the user whether or not the connection is successful.
6. If successfully connected, the remote IP will appear in FMOD Designer 2010's title bar.

Auditioning via the Network

Once FMOD Designer 2010 is connected to the remote FMOD Ex Engine (i.e. the console running the game title) any changes to the project made via the FMOD Designer 2010 interface will be (almost) instantly reflected in the audio outputted from the remote machine.

Auditioning a large title with multiple projects can be simplified through the use of workspaces. Simply load all projects used in the title into the current workspace. Individual projects can then be fine-tuned by selecting the project using the 'Project' dropdown list.
The following properties may be modified and auditioned during run-time:

- Event category volume
- Event volume
- Event pitch

- Event priority
- Event 2D speaker levels
- Event envelopes
- Event 3D mindistance
- Event 3D maxdistance
- Event mode (2D or 3D)
- Event 3D rolloff (logarithmic, linear or custom)
- Event 3D position (world relative or head relative)
- Event cone settings
- Event doppler scale
- Event reverb dry level
- Event reverb wet level
- Event 3D speaker spread
- Event 3D pan level
- Sound volume (blue envelope)
- Reverb

Disconnecting from the Network
To disconnect FMOD Designer 2010 using existing connection details:

1 Open the Audition menu
2 Select the option 'Manage Connections…'. A pop-up window should appear, as shown in figure 3-35.
3 Press the 'Disconnect' button.
4 FMOD Designer 2010 will now disconnect from the target machine.

Deleting Connection Details
To delete connection details:

1 Open the Audition menu
2 Select the option 'Manage Connections…'. A pop-up window should appear.
3 Select the connection name from the 'Name' list.
4 Press the 'Delete' button. The connection details will now be deleted.

User Properties

User Defined Properties are a provided by FMOD Designer 2010 to allow sound designers to add their own properties to events and event groups.

Creating a User Defined Property
To create a user defined property:

1 Click the Event button.
2 Select the event (or event group) to add a property to.
3 Right-click on the 'User-property' cell from the properties panel. A Menu pop-up window should appear.
4 Select the option 'Add/Edit user properties'. A pop-up window should appear. This pop-up window is shown in the figure below.
5 Press the 'Add' button. An empty user property will be created.
6 Enter the 'name' and 'description', and then select the data 'type' from the User-property frame.
7 Press the 'OK' button. The property has been created.

Figure 3-49: The User Property pop-up Window

Editing a User Defined Property
To create a user defined property:

1 Click the Event button.
2 Select the event (or event group) with the user property to edit.
3 Right-click on the 'User-property' cell from the properties panel. A pop-up window should appear.
4 Select the option 'Add/Edit user properties'.
5 Select the user-property to be edited from the user property list.
6 Edit the 'name' and 'description', or 'type' from the User-property frame.
7 Press the 'OK' button. The property has been edited.

Deleting a User Defined Property

To create a user defined property:

1. Click the Event button.
2. Select the event (or event group) to delete a property from.
3. Right-click on the 'User-property' cell from the properties pane. A pop-up window should appear.
4. Select the option 'Add/Edit user properties'.
5. Select the user-property to be edited from the user property list.
6. Press the 'Delete' button. The user-defined property should be removed from the event or event group.

Localization

As more and more games are released worldwide, sound designers find themselves handling content for each language of the target markets.

‘How can a sound designer work with multiple languages?’

Previously this situation required the sound designer to duplicate their project work for each language. Such redundancy slows the sound designer's workflow and is not a productive use of their time.

Localization within FMOD Designer 2010 eliminates the need for duplicate event data. From within the application, the sound designer can construct events using a primary language, and then easily substitute secondary languages when auditioning events or building the project's FSB files.

This section details FMOD Designer 2010 localization support and provides instructions on its use.

Maintaining Languages

FMOD Designer 2010 implements localization control using filename prefixes to identify the language the file belongs to.

For example, if our primary language is English and the project uses the following filenames:

- hello.wav
- howistheweather.wav
- goodbye.wav

All subsequent languages will derive their filenames and location from these filenames. Continuing this example, the secondary language is German. In this case the German versions of the files will be identified with the prefix 'de_'. The following files must be provided:

- de_hello.wav
- de_howistheweather.wav
- de_goodbye.wav

Some users may choose to keep the different language files in different directories. This can be done using a prefix such as 'german/', which would require the files to be placed in a 'german'

directory. Both a directory and filename prefix can be used together in a prefix, such as 'german/de_'.

Projects must contain at least one language, (acting as a 'primary' language). When starting a new project, the primary language is set to 'default'. Whilst the primary language does not require a prefix, one may be used. If the primary language has a prefix, it is removed before the secondary language prefix is added, whilst leaving any path elements.

For example, if the primary prefix is ‘english’ and the current prefix is ‘german’, ‘samples/english/hello.wav’ will be converted to ‘samples/german/hello.wav’.

Note how that the prefix replacement is ‘anchored’ to the last path separator in the base filename - meaning ‘dialog/english/hello.wav’ matches whereas a reference to file ‘english/dialog/hello.wav’ does not.

Prefix replacement also works with both path and filename elements. For example if the primary prefix is ‘dialog/en_’ and current prefix is ‘german/dialog/’ the file reference ‘audio/dialog/en_hello.wav’ would be converted to ‘audio/german/dialog/hello.wav’.

Selecting a Language

The target language is selected using the 'Language' dropdown list. Simply click the down arrow, and select the language from the menu that appears.

Figure 3-50: The Language List Drop Down Menu

Changing the selected language causes the following:

- The prefix of the selected language is added to the start of all audio file names within 'Sound Defs', 'sound Instances' and Banks.
- Any auditioning will feature the audio from the selected language.
- When building the project, the Banks placed into the .FSB file will use the Sound Files of the selected language.

Adding a Language

To add a target language, complete the following steps:

1 Click the 'Language' drop down list. A menu should appear.

2 Select the menu item 'Edit languages…'.
3 Press the 'Add…' button. A pop-up window should appear.
4 Enter the new language name into the 'Name' text field.
5 Press 'OK'.

Editing a Language
To rename a target language, complete the following steps:

Click the 'Language' drop down list. A menu should appear.
2 Select the menu item 'Edit languages…'.
3 Press the 'Rename…' button. A pop-up window should appear.
4 Enter the new language name into the 'Name' text field.
5 Press 'OK'.

Removing a Language
Removing a language will also delete any wave bank language prefixes attached to the language.
To delete a target language, complete the following steps:

Click the 'Language' drop down list. A menu should appear.
2 Select the menu item 'Edit languages…'.
3 Press the 'Delete' button.
4 Press 'OK'.

Setting the Prefix
To set the filename prefix for a language, complete the following steps:

1 Select the language using the 'Language' drop down list.
2 Click the Bank View.
3 Click on the bank to localize.
4 Enter the desired prefix into the 'Filename prefix' cell and press the [enter] key.

The path and (or) filenames of the Sound Files inside the bank should change accordingly.

Prefixes are set on a per bank basis. This allows the sound designer maximum flexibility when associating banks with languages.

For example, imagine we are creating a sport title in both English and German. The audio, in part, includes a 'commentary' bank featuring the vocal talent and the 'on-court' environment sounds bank. While the 'commentary' bank would require both English and German versions of the audio files, the audio for the 'on-court' bank could be the same for both language versions.

In this situation where the audio is shared, the bank 'Filename prefix' attribute for both languages should be set blank. This will ensure both languages use the same files.

Missing Files at Build Time

When building projects, FMOD Designer 2010 will attempt to use the files specified by the 'Language' dropdown list. When building languages other than the primary, it is especially important that the files are present in the location shown in the Sound File panel under the Bank View.

As all non-primary filenames are derived from the primary language filename, missing files cannot be found manually using a 'Find file' dialog (which is typically what happens when a file cannot be found during a build). To fix this error, the sound designer must either move the files to the appropriate location (as per the prefix) or change the Bank prefix.

Workspaces

What is a Workspace?

A workspace allows sound designers to open multiple projects simultaneously within FMOD Designer 2010. A workspace can be thought of as a list of open project files. Sound designers may save and load these lists as needed. Workspace data is saved as a FMOD Designer 2010 workspace file, with the extension '.fdw'.

As previously stated, while multiple projects may be opened within a workspace, each project is saved to its own project file (.fdp) and can be used independently from a particular workspace.

When multiple projects are opened within FMOD Designer 2010, the sound designer may choose the current project using the 'Project' dropdown list, shown in the figure below.

Figure 3-51: Selecting the project 'FMOD_Manual' from the dropdown list

Opening a Workspace

To open a workspace:

1. Select 'File' from the Menu Bar. A Menu pop-up window should appear.
2. Choose the menu option 'Open Workspace…' A pop-up window should appear.
3. Select the workspace file (.fdw) to load.
4. Press 'OPEN'. The workspace and project data will now be loaded. Sound designers can check the message log (by pressing the [CTRL] key + L) to ensure all data was loaded successfully.
5. Projects opened within the workspace can now be selected using the 'Project' dropdown list.

Saving a Workspace

The list of open projects within a workspace can be saved to disk using the 'Save Workspace' or 'Save Workspace As…' menu items under File from the Menu Bar. Choosing to save a workspace will also cause all projects within the workspace to be saved.

To save a workspace to a new file:

1 Select 'File' from the Menu Bar. A Menu pop-up window should appear.
2 Choose the menu option 'Save Workspace As…'. A pop-up window should appear.
3 Select the filename of workspace file (.fdw) to save.
4 Press 'SAVE'. The workspace and all open projects will now be saved to their respective files.

Closing a Workspace

Closing a workspace causes all open projects to be closed and creates a new blank workspace. To close a workspace:

1 Select 'File' from the Menu Bar. A Menu pop-up window should appear.
2 Choose the menu option 'Close Workspace'.
3 If any project requires saving, a pop-up window will appear. Click the 'Yes' button to save and enter the filename of the workspace, if desired, otherwise press 'No'.
4 The current workspace is now closed and an empty workspace exists.

Note:

Even projects not associated with the current workspace will be closed clearing the Designer 2010 interface of all data.

CHAPTER FOUR: Reference Guide

4

This chapter provides a comprehensive reference guide to FMOD Designer 2010. Readers are encouraged to read this section before attempting to produce complex soundscapes. The purpose of this chapter is to provide a simple explanation for each property, option box, menu item or effect contained in FMOD Designer 2010.

Key topics in this chapter:

- Reference guide for all property pop-up windows
- Reference guide for FMOD effects

Properties Reference Guide

Events View Properties

This section provides a comprehensive reference guide to all property panes contained within the Events View.

Event Category Properties

Property	Description & Notes
Name	The name of this event category.
Volume	An overall volume trim (in decibels) that is applied to all events in the category.
Pitch	A relative pitch adjustment (in octaves) that is applied to all events in the category.
Max Playbacks	
Max Playbacks Behavior	
Notes	A text note attached to the event category.

Event Group Properties

Property	Description & Notes
Name	The name of this event group.
User properties	FMOD Designer 2010 allows sound designers to add their own properties to an event group.
Notes	A text notes attached to the Event Group.

Event Properties Pane

Property	Description & Notes
Name	The name of an event. Events should have a short simple name, as the programmer may use this name to retrieve the event while in the game.
Category	
Include in Build	
Oneshot	If set to 'Yes', this event will automatically stop if no Sound Def instances (in this event) are active. This property is useful when the programmer wants a callback or to release the event instance once the event is not playing.
Volume	The overall volume for the entire event. This property is generally not used, however, can be useful when adjusting the relative volume of events during the final stages of in-title mixing.
Volume Randomization	Sets the maximum volume deviation that may randomly be applied to the volume of the event.
Reverb Wet Level	dB level of the 'wet' (reverberated) signal.
Reverb Dry Level	dB level of the 'dry' (non-reverberated) signal.
Pitch	The overall pitch of the whole event. This property is generally not used, however can be useful when adjusting the relative pitch of

Property	Description & Notes
	events during the final stages of in-title mixing.
Pitch units	Sets the unit displayed when altering the pitch of the event. Units include octaves, semitones and tones. Setting this Property to semitones will snap the pitch to the nearest semitone or tone respectively.
Pitch Randomization	Sets the maximum pitch shift deviation that may randomly be applied to the pitch of the event.
Pitch Randomization Units	Setting this Property to semitones will snap the pitch to the nearest semitone or tone respectively.
Fade In Time	Specified in milliseconds, this property represents the time it takes for the event volume to gradually increase from silence to full volume.
Fade Out Time	Specified in milliseconds, this property represents the time it takes for the event volume to gradually decrease from full volume to silence.
Spawn Intensity	Applies a factor to all spawn times contained within the event. This property can be used as a global spawn frequency modifier for all Sound Defs contained within the event. **Note on Spawn Frequency:** Spawn frequency is an Event Layer Effect. Spawn Frequency scales the rate at which spawned sounds are triggered within the Event. This applies to Sound Defs in the Event which have non-zero Spawn Times.
Spawn Intensity Randomization	Adds a random behavior to the 'Spawn intensity' property.
Priority	The importance of this event in relation to all other events. This property is used to determine if this event has priority over other certain events, when there are many events during playback. This is one of a number of properties used by the FMOD engine to decide which audio is audible. For example a gunshot event of a player might have higher priority than that of a footstep event. A sound may be discarded when all hardware or software voices on a platform are being used and FMOD needs to choose between event sounds as to which is the most important. Event priority is determined **before** any audibility calculation (3D distance attenuation, volume envelopes, occlusion, etc) is accounted for. 0 = most important, 256 = least important. Default = 128.
Max playbacks	This important property determines how much memory FMOD should allocate for instances of this event, and how many times at once this event can possibly be heard. This is important, as specifying a number too large will result in the memory usage of FMOD being higher than it should be. If the event will only be heard once simultaneously, then this property should be set to 1. If the event is triggered multiple times and each occurrence needs to be heard, then the max playbacks value should be set to a number greater than 1. It is also possible to control the manner in which FMOD 'steals' playback from an event instance, using property, 'Max playbacks behavior'.
Max playbacks behavior	This property allows the sound designer to choose which method FMOD Ex uses when the number of event instances exceeds the 'max playbacks' value.

Property	Description & Notes
	• **Steal oldest:** The event that has been playing the longest will be replaced. • **Steal newest:** The most recent event to have been playing will be replaced. • **Steal quietest:** The event with the lowest volume (taking into consideration 3D distance attenuation and event volume) will be replaced. • **Just fail:** The new instance of the event will not steal any other instance and will just fail to be played, letting the other instances play as they were. • **Just fail if quietest:** If an event is calculated to be the quietest voice, it will not steal any other instance and will fail to be instantiated.
Mode	Selects whether the event will be managed in a two or three-dimensional space by the FMOD Engine. If '2D' is selected, the sound will be unaffected by any 3D calculations. If '3D' is selected, the event will be placed in 3D space, and positioned by the programmer at run time. 3D events are affected by distance attenuation, speaker position in regards to location relative to the listener and Doppler (based on how fast the sound is moving in relation to the listener). Certain event properties are only available to the 2D mode and others to the 3D mode. For example, the 3D rolloff factor cannot be used in a 2D event. 5.1 fixed speaker positions cannot be set for 3D events (as the 3D engine already does this automatically).
Ignore Geometry	
3D Rolloff [3D events only]	Sets the style in which the volume of an event will attenuate in the 3D world. There are three types of attenuation: Logarithmic, Linear and Custom. • **Logarithmic:** Selecting 'Logarithmic' rolloff causes the volume to halve as the distance doubles between the listening entity and the sound source. This is the most common method of sound attenuation as it models real world physics. In this case volume will only start to attenuate from '3D Min distance' (see definition below) onwards. Below the '3D Min distance' the volume will have no attenuation (it stays at full volume). At '3D Max distance' the volume will stop attenuating from whatever point it may be. In this mode the rolloff is constant and uses the rule specified above to simulate real world attenuation. Note that as FMOD assumes that game distance units are a certain length, using other values may result in obviously unrealistic behavior. • **Linear:** Selecting 'Linear' rolloff will cause the volume to linearly interpolate from full to silence between 'the distances set in the 3D Min distance' and '3D Max distance' properties. This mode gives the sound designer more control over the distance at which the sound will become silent to the listener but is not real world behavior. • **Custom:** Selecting 'Custom' rolloff allows the sound designer to define how the volume of a sound attenuates over a distance. This method uses the Events Layer pane and a volume or occlusion envelope. See Chapter 5, Tutorial Projects. '3D Min distance' and '3D Max distance' are ignored in this mode and are unavailable for editing. Using this mode, it is even possible to include low pass filtering to change the sound source as it gets further away from the listener.

Property	Description & Notes
3D Min distance [3D events only]	The distance from the sound source to the listener where the sound starts attenuating in volume. Before this point, the sound is played at full volume. This property can be used to model the volume characteristics of the source sound. For example, the sound of a bumblebee will not carry far. The sound level drops quickly. To simulate this use a small 3D min distance (i.e. 1cm). A jumbo jet however, is incredibly loud and can be heard at great distances. To model this, a large 3D Min distance such as 100 meters would be used. Choice of unit measurement is not important, it may be feet, yards or some other unit – just be consistent! **Note:** Consult with your programmer on which distance units are being used in the title.
3D Max distance [3D events only]	Note: The usage of this property is dependent on the '3D Rolloff' mode setting. In 'Logarithmic' rolloff mode, the amount of attenuation is calculated using a fixed rate. In this rolloff mode, the 3D Max distance property represents the distance where the sound will stop attenuating (i.e. At this point the sound will not get any quieter). In 'Linear' rolloff mode, this property represents the distance between the listener and sound source where the sound is fully attenuated (i.e. this is the distance where the sound becomes completely silent). In 'Custom' rolloff mode, this parameter has no effect. Choice of unit measurement is not important, it may be feet, yards or some other unit – just be consistent! **Note:** Consult with your programmer on which distance units are being used in the title.
3D Position [3D events only]	Specifies whether the event is a 'World relative event' or a 'Head relative event'. • **World Relative [default]:** The World relative mode allows an event to be placed in 3D space using an absolute world position. The position of this event does not change as location of the listener changes. For example, when using the World Relative mode, if an event is placed at 0,0,0 in world space, it will stay there. A majority of events will use this mode. • **Head Relative:** Head relative is a special mode that allows an event to be 'attached' to the listener. The sound appears to automatically follow the listener around the world. For example, using the Head Relative mode, an event placed at position 0,0,0 would sound like it is coming from inside the listener's head, regardless of where the listener moves. If the event is placed at 0,0,1 where z = forwards and the units are in meters, then it would appear that the sound was just in front of the listener at all time, no matter where the listener moves. This mode is most useful for things like cockpit sounds, player related sounds (i.e. gunfire for the main player in an FPS) and the like.
3D Position	This property specifies a range of positional deviation of a 3D

Property	Description & Notes
Randomization [3D events only]	event. It represents the maximum distance the location of the event can vary along each of the x,y and z axes. To audition this feature, the project must be built and the event triggered in game or using a tool such as FMOD Event Player. In some circumstances, where the listener's position is far from the event or the amount randomization is small, changes to the event's 3D position may not create a change in the listener's surround sound panning. The random position offset is calculated each time the event is triggered.
3D Cone inside angle [3D events only]	The inside angle of the cone, and the field of sound that will have full volume (no attenuation). A 3D Cone with an inside angle of 90° and an outside angle of 180° can be seen in the diagram above. The programmer will set the orientation of the event (and therefore direction of the sound) at runtime.
3D Cone outside angle [3D events only]	The outside angle of the 3D cone and the field of sound that will be attenuated. The volume of this space is set by 3D Cone outside volume property. The outside angle must be larger than the inside angle. The FMOD engine will linearly interpolate the volume from full to the volume specified from the inside angle to the outside angle. A 3D Cone with an inside angle of 90° and an outside angle of 180° can be seen in the diagram above. The programmer will set the orientation of the event (and therefore direction of the sound) at runtime.
3D Cone outside volume 3D Events only	This property specifies the amount of attenuation a sound should receive when the listener is positioned in the outside angle.
3D Doppler Factor	
3D Speaker Spread	
3D Pan Level	
2D Speakers *[2D events only]*	*A 2D event can be mixed and positioned in a 5.1 or 7.1 speaker field. These levels let the designer specify levels for each speaker.* *A more interactive method is available by going to the menu and selecting [Tools | Surround pan]. This tool enables the speaker placement to be made using a graphical interface.* *Note: These properties do not operate on 3D events, as 3D events are automatically panned in 5.1 (or 7.1) based on their 3D position.*
2D Speaker L	dB level in the 'Left' channel.
2D Speaker C	dB level in the 'Centre' channel.
2D Speaker R	db level in the 'Right' channel.
2D Speaker LS	dB level in the 'Left Side' channel.
2D Speaker RS	dB level in the 'Right Side' channel.
2D Speaker LR	dB level in the 'Left Rear' channel.
2D Speaker RR	dB level in the 'Right Rear' channel.
Speaker LFE	dB level in the 'Low Frequency Emitter' channel.
Use template	Specifies the template that this event is derived from.

Property	Description & Notes
User properties	FMOD Designer 2010 allows sound designers to add their own properties to the event. Adding these events allows the sound designer to add game specific data to the audio engine. For example, in a game featuring stealth, the sound designer could add an 'AI radius' property to an event. This property would represent the distance required between the sound source and AI character to remain undetected. The programmer would then use this property and the world position of the event and AI characters to test whether the AI characters (i.e. guards) have been alerted to the sound. Creating a user-defined property is discussed in detail in Chapter Three.
Notes	Any event related information can be stored here as a note. This data is not exported to the final game data file.

Layer Properties

Property	Description & Notes
Name	The name of the layer.
Control parameter	Specifies which parameter selects the active Sound Def in the layer.
Priority	Specifies the layer's priority, where -1 uses the event's priority by default, 0 = highest priority, 256 = low priority. When not set to -1 the priority will be determined by the Sound Defs used in the Layer. Events are not prioritized, the Sound Defs within an Event are prioritized independent of the Event.

Sound Instance Properties

Property	Description & Notes
Start position	The starting position of a Sound Def instance box in the event editor. While this property can be set using the mouse, manually entering this value provides greater accuracy. For example if the parameter has a range of 0.0 to 1.0, the sound instance can be manually placed at exactly 0.5.
Length	The length of the Sound Def instance (according to the parameter). While this property can be set using the mouse, manually entering this value provides greater accuracy. Assuming the starting position of the Sound Def instance is 0.5, the length could be manually specified as 0.25. This would mean the Sound Def instance would range from 0.5 to 0.75 exactly.
Start mode	Determines how this particular instance of a Sound Def will start playback. • **Immediate:** The instance will start playback immediately when activated by a parameter. • **Wait for previous:** The instance will start playback only when activated by a parameter and the previous instance in the layer has ended.
Loop mode	Determines how this particular instance of a Sound Def should play. • **Oneshot:** The instance will play once and stop. • **Loop and cutoff:** The instance will play repeatedly and will stop immediately if deactivated by a parameter. • **Loop and play to end:** The instance will play repeatedly and will stop at

Property	Description & Notes
	the end of the Sound File when deactivated by the parameter.
Loop Count	If the 'Loop mode' of the sound instance is not set to 'Oneshot', this property specifies the number of times playback is repeated. If this property is set to zero ('0'), the playback will loop indefinitely.
Autopitch enabled	When this property is set to 'Yes', the FMOD engine will automatically pitch bend the sound based on the position of the parameter cursor within the parameter range (shown on the ruler), Autopitch at min property and the Autopitch reference property. This technique is useful for automatically manipulating sounds, removing the need for customized pitch envelopes. This technique is used predominately in the car engine model example included with the FMOD Designer 2010 software.
Autopitch Parameter	
Autopitch reference	This property represents the parameter value that will act as the original pitch of the Sound Def. This is commonly referred to as the 'middle C' or center frequency. For example, assume that: The property 'Autopitch enabled' is set to 'Yes'. The Property 'Autopitch at min' is set to 0 The event parameter (ruler) range is set from 0.0 to 1.0. The property 'Autopitch reference' is set to 0.25. The Sound Def instance starts at 0.0 and stops at 1.0. If the parameter cursor is placed at: 0.25, the sound is heard at its original pitch. 0.125, the sound is heard at half the original pitch. 0.5, the sound is heard at double its original pitch. 0.75, the sound is heard at triple its original pitch. 1.0, the sound is heard at quadruple its original pitch.
Autopitch at min	This property determines the value of pitch when the parameter value is at its minimum. When 'Autopitch At Min' is set to 0, the pitch of the sound instance will be shifted to 0Hz, when the parameter is at its minimum value. This Property determines the pitch of the sound when the Autopitch parameter is at the start position of the Sound Def Instance. It is expressed as a fraction of the reference pitch
Fine tune	An offset value in hertz (hz) to the original pitch. This property may be set using the Fine Tuner pop-up window. The Fine Tuner pop-up windows allow the pitch of the Sound Def instance to be modified while the event is playing both the instance and other sounds in the background. This is useful when tuning the Sound Def instance against a reference sound. The Fine tune property can be controlled coarsely (-20khz to +20khz), or finely (-2kHz to +2kHz). Alternatively, a Hz offset can be manually set using the text entry box.

Parameter Properties

Property	Sub-Properties	Description & Notes
Name		The name of the parameter. A parameter should be named with a short simple name, as the programmer may use this name to retrieve the parameter while in the game.
Primary parameter		
Range		This group of properties affects the attributes of the event parameter. Changes to these properties will be reflected in the appearance of the ruler displayed above the event layers.
	Minimum Value	This property specifies the smallest value the parameter can represent.
	Maximum Value	This property specifies the largest value the parameter can represent.
	Units	This property is a text label used to describe the unit type represented by the parameter.
	Ruler Spacing	This property specifies the size of the divisions displayed in the ruler.
Velocity		This group of properties affects how the parameter value will change over time. These properties can be used to time sequence Sound Defs, effects and parameter properties.
	Velocity	This property specifies the speed at which the event parameter cursor automatically increments from its minimum value to its maximum value. This value is specified in units per second and can be negative.
	Loop Behavior	This property specifies how the parameter cursor behaves. In 'Oneshot' mode the parameter cursor will move through the time sequence once. In 'Oneshot and stop event' mode, the parameter cursor will move through its time sequence and then end the entire event. This will cause the playback of any other layers or Sound Defs to also end. In 'Loop' mode, the parameter cursor will continually move through the time sequence in a repeating fashion.
	Seek speed	Provides an alternate method for time-sequencing events. This parameter specifies (in units per second) how quickly the parameter cursor reacts to changes in the parameter value. Using the Seek speed makes it possible to automatically interpolate between the new parameter value and previous parameter value. Note: For this property to be effective, the velocity parameter must be set to 0.0.
Keyoff on silence	[none]	Provides an alternate behavior for sustain points. When used, sustain points are automatically released when the Sound Def is silent. When combined with a fast velocity parameter it is possible to create sequences of one-shot Sound Defs. **To use this feature, the event's 'Oneshot' property must be set to 'Yes'.**

Simple Event Editor Pane Properties

Property	Sub-Property	Description & Notes
Playlist		The Playlist is where Sound Files are added to the Simple Event and displayed.
	Rand	The percentage values displayed in this area relate to the percentage chance of each Sound File in the Playlist being triggered in Random play mode.
	Value Adjustment Dial	This control alters the percentage values of Rand for any selected Sound File. (Note: combined total value must always equal 100%)
	Sound	This is a list of all the Sound Files included in the Simple Event. Double-Clicking on any Sound File entry will audition the selected Sound File.
Loop Options		The Loop Options allow for a variety of playback behaviors to be set for the Simple Event
	Oneshot	When triggered the Simple Event will play a single sound from the playlist as defined by the Playlist Options
	Repeating Loop	This function creates a loop by constantly repeating the first entry in the Playlist. If there is more than one file in the Playlist FMOD 2010 will only select the first file to loop. **Play Count:** This value defines how many times the sound will loop before it stops. It can be set to loop infinitely or with values 1-999999999
	Successive Loop	This function creates a loop by playing sounds end to end continuously working through the Playlist. If there is only one Sound File in the Playlist this function will behave the same as Repeating Loop. **Loop Count:** This value defines how many times the sound will loop before it stops. It can be set to loop infinitely or with values 1-999999999 **Queued Sounds:** This value
	Granular	The granular function triggers entries from the Playlist dependant on the values it is given. It is possible to create infinitely varying loops using this function. It is also possible using the Polyphony value to have more than one sound playing simultaneously. **Grain Interval:** This slider requires two values that are determined by moving the ends of the slider independently. Moving the middle of the slider will alter its overall relative position on the timeline. Double-Clicking on either end of the slider opens a pop-up window that allows for numbers to be inputted for more accurate values. The two values represent the minimum and maximum amount of time in which a sound will be triggered from the Playlist. This effectively creates a window in time within which FMOD 2010 will randomly trigger a Playlist sound. FMOD 2010 will then count another value within that defined window of time and trigger the next sound up to the value of Total grains. If both values are the same then there is no random time value and the sounds will be triggered at exactly the value defined. **Polyphony:** By default this value is set to 1 as a result only one sound can play at a time. With this value FMOD 2010 must wait until a sound that is playing is finished before it can trigger the next sound. Increasing this value beyond 1 will allow that number of sounds to play simultaneously. In this way it is possible to create an overlap of sounds. **Note**: It is a combination of the Grain Interval value and the length of the Sound File being played that will determine if an overlap is even necessary. A grain interval of over 5 seconds for sounds that are only 2 seconds long will mean no overlap is ever needed. **Total Grains:** This function is the same as Play Count and Loop Count in that it works as a counter before the event stops triggering new sounds. It can be set to infinite or 1-999999999
Build Options		These properties set how FMOD 2010 will format the Sound Files within this Simple Event at build time.
	Bank	FMOD 2010 will automatically create a default Bank for any new project and place all the Sound Files within that

Property	Sub-Property	Description & Notes
		Bank. This drop down menu allows the sound designer to select from other Banks that they may have created for the project. The Sound Files for this Simple Event will be included In the selected Bank during the build process.
	Resampling	FMOD 2010 can automatically analyze and resample all Sound Files included in a project to minimize the disk and memory footprint. **Note:** Resampling does NOT alter the actual sound files on the computer. At build time FMOD formats all necessary files into the correct format for the selected platform and packages them appropriately. The Resampling function takes place at this time to create the most efficient package for the project. The actual Sound Files are not altered. Off: The Resampling is not active and all Sound Files will remain at the rate at which the actual sound file is set. **Automatic:** FMOD 2010 will resample all Sound Files based on their frequency components to maximize their efficiency. The Adjustment value dial will further alter the level of resampling down from the initial optimized level. **Fixed:** This allows the sound designer to define exactly the desired sample rate to which FMOD 2010 will resample the Sound Files. The adjustment Value Dial is used to alter the Sample rate to the desired value.
Playlist options		These Properties define how FMOD 2010 will access the Sound Files from the Playlist and if it will alter their volume and pitch
	Random	(Default) FMOD 2010 will randomly select one of the Sound Files from the Playlist each time it is triggered. This can be either when manually auditioned by the user or by a game event, or each time the event itself triggers a sound via the Loop Options functionality. **Allow sounds to be repeated:** By default FMOD 2010 will randomly select a Sound File from the Playlist but will not play that same sound again until all sounds have played at least once. This checkbox removes that guideline and allows for true random selection of Sound Files. **Allow silences to be repeated:** It is often desirable to include silences within a selection of Sound Files. This checkbox allows the repeat of Silence instances without having to alter the behavior for all Sound Files.
	Shuffle	The Shuffle function randomizes the order of the Sound File entries within the Playlist and then plays through that newly shuffled order in a sequential manner. **Share playlist globally:** This checkbox makes the playlist part of the overall project so the shuffle mechanic becomes a global reordering.
	Sequential	When played sequentially the playlist is triggered exactly as it is displayed. FMOD 2010 will work through the list from top to bottom and then return to the top of the list. **Remember playlist position:** When checked FMOD 2010 will remember progress through the list so it can continue its progress from the last point when next triggered. **Share playlist position globally**: This checkbox makes the playlist part of the overall project so the sequential mechanic becomes part of the global order.
	Sound Randomization	These values allow for the Sound Files volume and pitch to be randomized each time a sound is triggered. **Attenuation:** The values range from 0 to -60dB **Pitch:** This value creates a range over which the sounds pitch will be randomized. The values range between +-0 to +-48 semitones

Bulk Editor Pane Properties

Property	Sub-Property	Description & Notes

Property	Sub-Property	Description & Notes
Volume		
	Volume	The overall volume for the entire event. This property is generally not used, however, can be useful when adjusting the relative volume of events during the final stages of in-title mixing.
	Volume Randomization	Sets the maximum volume deviation that may randomly be applied to the volume of the event.
Pitch		
	Pitch	The overall pitch of the whole event. This property is generally not used, however can be useful when adjusting the relative pitch of events during the final stages of in-title mixing.
	Pitch Randomization	Sets the maximum pitch shift deviation that may randomly be applied to the pitch of the event.
Reverb		
	Reverb Wet Level	dB level of the 'wet' (reverberated) signal.
	Reverb Dry Level	dB level of the 'dry' (non-reverberated) signal.
Fade		
	Fade In	Specified in milliseconds, this property represents the time it takes for the event volume to gradually increase from silence to full volume.
	Fade Out	Specified in milliseconds, this property represents the time it takes for the event volume to gradually decrease from full volume to silence.
Rolloff		
	Min&Max Distance	Min Distance is the distance from the sound source to the listener where the sound starts attenuating in volume. Before this point, the sound is played at full volume. This property can be used to model the volume characteristics of the source sound. For example, the sound of a bumblebee will not carry far. The sound level drops quickly. To simulate this use a small 3D min distance (i.e. 1cm). A jumbo jet however, is incredibly loud and can be heard at great distances. To model this, a large 3D Min distance such as 100 meters would be used. Choice of unit measurement is not important, it may be feet, yards or some other unit – just be consistent! **Note:** Consult with your programmer on which distance units are being used in the title. **Max Distance** Note: The usage of this property is dependent on the '3D Rolloff' mode setting. In 'Logarithmic' rolloff mode, the amount of attenuation is calculated using a fixed rate. In this rolloff mode, the 3D Max distance property represents the distance where the sound will stop attenuating (i.e. At this point the sound will not get any quieter). In 'Linear' rolloff mode, this property represents the distance between the listener and sound source where the sound is fully attenuated (i.e. this is the distance

Property	Sub-Property	Description & Notes
		where the sound becomes completely silent). In 'Custom' rolloff mode, this parameter has no effect. Choice of unit measurement is not important, it may be feet, yards or some other unit – just be consistent! **Note:** Consult with your programmer on which distance units are being used in the title.
	Curve Type	Sets the style in which the volume of an event will attenuate in the 3D world. There are three types of attenuation: Logarithmic, Linear and Custom. • **Logarithmic:** Selecting 'Logarithmic' rolloff causes the volume to halve as the distance doubles between the listening entity and the sound source. This is the most common method of sound attenuation as it models real world physics. In this case volume will only start to attenuate from '3D Min distance' (see definition below) onwards. Below the '3D Min distance' the volume will have no attenuation (it stays at full volume). At '3D Max distance' the volume will stop attenuating from whatever point it may be. In this mode the rolloff is constant and uses the rule specified above to simulate real world attenuation. Note that as FMOD assumes that game distance units are a certain length, using other values may result in obviously unrealistic behavior. • **Linear:** Selecting 'Linear' rolloff will cause the volume to linearly interpolate from full to silence between 'the distances set in the 3D Min distance' and '3D Max distance' properties. This mode gives the sound designer more control over the distance at which the sound will become silent to the listener but is not real world behavior. • **Custom:** Selecting 'Custom' rolloff allows the sound designer to define how the volume of a sound attenuates over a distance. This method uses the Events Layer pane and a volume or occlusion envelope. See Chapter 5, Tutorial Projects'3D Min distance' and '3D Max distance' are ignored in this mode and are unavailable for editing. Using this mode, it is even possible to include low pass filtering to change the sound source as it gets further away from the listener.
Playback Behavior		
	Steal oldest	The event that has been playing the longest will be replaced.
	Steal newest	The most recent event to have been playing will be replaced.
	Steal quietest	The event with the lowest volume (taking into consideration 3D distance attenuation and event volume) will be replaced.
	Just fail	The new instance of the event will not steal any other

Property	Sub-Property	Description & Notes
		instance and will just fail to be played, letting the other instances play as they were.
	Just fail if quietest	If an event is calculated to be the quietest voice, it will not steal any other instance and will fail to be instantiated.
	Max Playbacks	This important property determines how much memory FMOD should allocate for instances of this event, and how many times at once this event can possibly be heard. This is important, as specifying a number too large will result in the memory usage of FMOD being higher than it should be. If the event will only be heard once simultaneously, then this property should be set to 1. If the event is triggered multiple times and each occurrence needs to be heard, then the max playbacks value should be set to a number greater than 1. It is also possible to control the manner in which FMOD 'steals' playback from an event instance, using property, 'Max playbacks behavior'.

Sound Defs View Properties

This section provides a comprehensive reference guide to all property panes contained within the Sound Def View.

Sound Def Properties

Property	Description & Notes
Name	The name of the Sound Def.
Playmode	Specifies which method will be used to select the Sound File for playback. • **Sequential:** Sound Files are played back in the order in which they are listed in the Sound files view. Further playbacks will result in the repetition of the final entry • **SequentialEventRestart:** Sound Files are played back in sequential order, however when the event is restarted, playback returns to the start of the queue. • **Random:** Sound Files are played back in random order. • **RandomNoRepeat:** Sound Files are played back in a quasi-random order, with an added restriction that no Sound File is played twice in a row. • **Shuffle:** Sound Files are played back in a quasi-random order, with an added restriction that no Sound File is repeated until all Sound Files have been played. • **ShuffleGlobal:** Sound Files are played back in a quasi-random order, with an added restriction that no Sound File is repeated **in any event instance** until all Sound Files have been played. • **ProgrammerSelected:** When the Sound Def is played (or respawned) a callback is created. The programmer then selects which Sound File to play by supplying an index – where index 0, is the first Sound File, 1 is the second, 2 is the third, etc.
Volume	The volume in which the Sound Def will be played. The volume is specified in decibels (dB).
Volume randomization	The maximum allowable negative gain deviation to be applied to the Sound Def. The volume randomization is specified in decibels.
Pitch	The relative pitch that the Sound Def will be played.
Pitch randomization	The maximum allowable pitch deviation to be applied to the Sound Def. The pitch randomization is specified in octaves.

Property	Description & Notes
Pitch randomization behaviour	Determines when the pitch of the event is recalculated. • **Randomize every spawn:** Every spawn (default behavior). Randomized once for each newly spawned Sound Def Instance. This is the default setting. • **Randomize when triggered by Parameter:** Randomized each time the Sound Def instance is triggered by the control parameter. • **Randomize When Event Starts:** Randomized only when a Sound Def is started, and not on subsequent spawns or parameter changes
Spawn Time	Creates a random spawn time (normally distributed) between the two specified values, measured in milliseconds. 'Spawning' means the Sound Def triggers repeated playback of itself. However, unlike looping, each repeat (or spawn) can overlap the previous sound and starts a new Sound Def. If both values are set to 0, spawning will be disabled **This spawn time range can be scaled using the 'Spawn Intensity' effect.** **Note on Spawn Frequency:** Spawn frequency is an Event Layer Effect. Spawn Frequency scales the rate at which spawned sounds are triggered within the Event. This applies to Sound Defs in the Event which have non-zero Spawn Times.
'Play Count	The specified value acts as a counter. The Sound Def will spawn until that value is reached and then stop
Max Polyphony	The number of sound instances within the Sound Def that can be played simultaneously.
Trigger Delay	Creates a window of time between the two specified values. If the two values are equal the Sound Def will be delayed the exact amount of time. If the two values are not equal they create a window of time in which the delay will randomly be determined.
3D Position randomization	This property specifies a range of positional deviation of a Sound Def. This variation is calculated each time the Sound Def is spawned. **This property is only functional when the event mode is set to 3D.**
Notes	A text note attached to the Sound Def.

Sound Def Sound File Properties

Property	Description & Notes
Sound File	States the name (and directory) of the entry within the Sound Def.
Weight	States the weighted chance of the Sound File being selected for playback. Altering Weight alters Play Percentage, and vice versa **Note:** Weight and Play Percentage are different terms for the same function; they are both included to provide choice to different users. Only one of the functions can be displayed at any one time and the check-box must selected to choose between them.
Play Percentage	States the percentile chance of the Sound File being selected for playback and whether playback percentage is locked. Altering Play Percentage alters Weight, and vice versa

Banks View Properties

This section provides a comprehensive reference guide to all property panes contained within the Banks View.

Bank Properties

Property	Description & Notes
Name	The name of the Bank.
Filename prefix	This attribute represents the prefix appended to the beginning of the filename to identify its language group. See the section 'Localization' for more information.
Bank type	Specifies if the data is loaded partially or completely into memory or streamed from disk. Each option varies in terms of memory and CPU usage. **Note:** Specified on a per-platform basis. • **Stream from disk:** Audio data is streamed from disk. This method reads the audio file from the disk and decompresses it in real time (if required). This method is best for large files such as background music. Using this method with small repetitive audio pieces causes a large processing and disk access overhead. • **Load into memory**: Audio data is loaded completely into memory and decompressed in real time each time the audio is played. This method can save memory usage but is more CPU intensive. • **Decompress into memory:** Audio data is first decompressed and then loaded into memory in raw PCM form. This method requires more CPU during loading and more memory, however, there is less delay for playback – making it ideal for small repetitive sounds that require a fast response time (such as a gunshot).
Compression	For more information on compression, see File Formats in Chapter Six. Some of the compression formats are only available for specific platforms. • **PCM:** The Bank is saved as PCM data. This format uses the most disk space but is the least CPU intensive. • **ADPCM:** The Bank is saved as PCM data. This format uses the most disk space but is the least CPU intensive. • **MP3:** The Bank is saved as MPEG 1 layer 3 data. This format uses the least space but is the most CPU intensive. • **MP2:** The Bank is saved as MPEG 1 layer 2 data. This format reduces the required space but is relatively CPU intensive. Unlike MP3, this format does not require a license. • **VAG(Sony Playstation2 and PSP only):** The Bank is saved using a variant ADPCM format. This format uses a simple compression method to reduce the required disk space and required CPU time. • **GCADPCM(GameCube, Wii only):** The Bank is saved using a variant ADPCM format. This format uses a simple compression method to reduce the required disk space and required CPU time. • **XMA(xbox 360 only):** The Bank is saved using a variable bit rate ' lossy' compression. This format requires little space and has hardware support to negate any load on the CPU. • **CELT:** The Bank is saved as CELT data. This format is similar to mp3 but requires less space, is more CPU intensive and does not require a license.
Compression quality	The amount of compression to be used. High compression results in smaller disk usage, but will reduce quality (creating audible distortion). Bit-rate approximately equals 3.2 x this value.
Max Streams	The maximum number of simultaneous disk streams that may be created when loading from this wave bank. This property is only used when the 'Bank type' property is set to 'Stream From Disk'. The default value of 10 is generally more than adequate, and will rarely need to be any higher. For more information, see Chapter 6.

Property	Description & Notes
Disable Seeking (Xbox 360 with XMA only)	This option will disable the inclusion of XMA seek tables – reducing the memory required by the wavetable. **Note:** Disabling seek tables may cause virtual voices to reposition themselves incorrectly and repositioning audio using a time offset property will no longer be possible. **Sound designers should consult with the programmer before setting this option as 'Yes'.**
Enable syncpoints	If syncpoint callbacks are not required, set this property to 'No' to reduce memory usage.
Force software	Specifies whether or not the wave bank is forced to play back through FMOD's software pipeline. This property should generally be set to 'Yes'. For more information, see Chapter 6.
Load into RSX memory (PS3 only)	Audio data from this bank is loaded into the PS3's RSX memory. The RSX memory is generally reserved for graphics, however FMOD can use it. This option is especially useful when the main memory is overburdened. **Sound designers should consult with the programmer before setting this option as 'Yes'.**
Notes	A field that allows the sound designer to append text notes to the wave bank.

Wave Bank Sound File Properties

Property	Description & Notes
Filename	States the source name (and directory) of the Sound Files within the Wave Bank.
Quality	Specifies the amount of compression applied to the individual Sound File within the FSB. The lower the quality value the higher the compression.
Resample rate	The sample rate of the sound file after FMOD optimization
Original rate	The original sample rate of the sound file external to FMOD
Bit depth	States the bit depth of the pre-optimized audio file.
Channels	States the number of channels in the pre-optimized audio file. This property also has the abilities to route multi-channel wave bank entries into preset speaker layouts. These routes are useful in applications such as interactive music, where the source audio may be located on many channels, but should be played to the front speakers. For more information, see the tutorial 'Fun with Multi-Channel Files.' • **n x Mono:** All channels are consider mono feeds and are placed in the centre speaker. • **n x Stereo:** Channels are routed in pairs to the front left and right speakers. • **Mono:** A mono channel Sound File. • **Stereo:** A stereo channel Sound File. • **5.1:** A six channel Sound File, routed to the speakers as: FL, FR, FC, LFE, RL, RR • **7.1:** An eight channel Sound File, routed to the speakers as: FL, FR, FC, LFE, RL, RR, SL, SR.
Sound type	States the file format of the pre-optimized audio file.
File size	States the file size of the pre-optimized audio file.
Used	States how audio will be used: SW = Software HW = Hardware 2D = 2D sound Engine 3D = 3D sound Engine

Property	Description & Notes
	Music=Software mode for music system
Notes	Text notes attached to the wave bank.
XMA Filtering (Xbox360 with XMA only)	Specifies whether XMA filtering will be applied to the Sound File before encoding. The XMA filter uses a low pass filter that allows the encoder to achieve a higher compression rate. sound designers may choose to disable XMA filtering when the Sound File contain requires high fidelity in the upper frequencies.

Reverb Pop-up Window Properties

This section provides a comprehensive reference guide to the property pane contained within the Reverb Defs pop-up window.

Reverb Properties

Properties	Description & Notes
Name	The name of the Reverb Def.
Reverb Preset	Sets all Reverb Properties for the selected preset. 'User' is displayed when no preset is selected or when preset values have been modified by the user.
Master Level	The overall level for this reverb Def
Decay time	The length of time the room reverberation takes to fade to silence in milliseconds. Ranging from 0.1 to 20.0 seconds, this feature can be used to simulate the size of the acoustic environment.
HF Decay Ratio	The ratio of the high frequency and low frequency content in the reverberation decay.
Pre Delay	The delay time of the first reflection in milliseconds, ranging from 0.0 to 0.3 seconds.
Late Delay	The delay time of the late reflection (relative to the first reflection) in milliseconds, ranging from 0.0 to 0.1 seconds.
Early Reflections	The volume level of the early reflections relative to the volume of room reverberation and ranges from -100 to 10dB
Late Relfections	The volume level of the late reflections relative to the volume of room reverberation and ranges from -100 to 20dB
Diffusion	The rate at which the density of the reverb increases during the decay. This echo density ranges from 0.0 to 100.0 percent.
Density	Represents how tightly the reflections in the decay are packed. This modal density ranges from 0.0 to 100.0 percent.
HF Gain	
LF Gain	
HF Crossover	The high frequency value referenced by the other parameters. The HF reference range from 20Hz to 20kHz.
LF Crossover	
Notes	A text note attached to the Reverb Def.

Project Properties

This section provides a comprehensive reference guide to Project Properties the Reverb Defs pop-up window.

Property	Description & Notes
Name	The name of this project.
Encryption key	A cipher key that can be used to encrypt the exported data of this project. This property is platform-specific
Template file	Specifies the location of the template file. Can be absolute or relative to the project path.

Property	Description & Notes
Audio source directory	The location of all audio data shall be relative to this value. This can be absolute or relative to the project path. If left blank, the project file path is used as default. This property is platform-specific.
Build directory	The directory location of the output files. This property supports macros Note: Each target platform may have its own build directory location.
Pre build commands	A list of commands that FMOD Designer 2010 will execute **before** building a project. This property is platform-specific.
Post build commands	A list of commands that FMOD Designer 2010 will execute **after** building a project. This property is platform-specific.
Pre save commands	A list of commands that FMOD Designer 2010 will execute **before** saving. Useful for version control. Output is written to the Message Log pop-up window [ctrl-L].
Post save commands	A list of commands that FMOD Designer 2010 will execute **after** saving. Output is written to the Message Log pop-up window [ctrl-L].
Build interactive music	Specifies whether the interactive music data is included in the project build. This property is platform-specific.
Notes	A text note attached to the project.

Tools Reference Guide

This section provides a comprehensive reference guide to all the tools provided with FMOD Designer 2010. The FMOD tools are designed to simplify auditioning and setting properties within events, including multiple parameters and 3D, surround or directional sound effects.

FMOD Engine Designer

The FMOD Engine Designer tool is specially designed to simulate engine sounds using a combination of engine RPM (revolutions per minute) and engine loading. This tool automatically adjusts the engine load parameter proportionally to changes in the RPM parameter value thus making it easy to hear how the parameters affect each other and create the overall sound.

The FMOD Engine Designer tool is shown below.

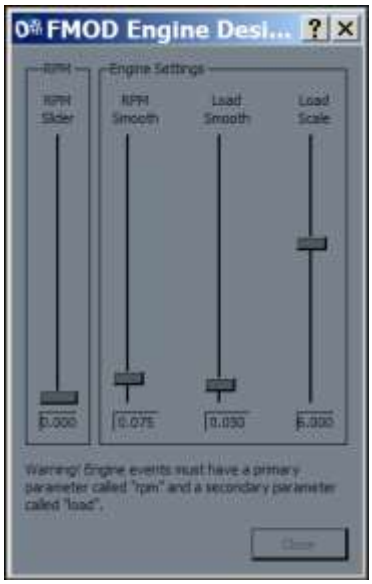


Figure 4-1: The FMOD Engine Designer pop-up window

This tool requires the event to have two parameters called 'rpm' (the primary parameter) and 'load'. Once created, the tool allows the sound designer to modify the relationship between two parameters using the vertical sliders, 'RPM Smooth', 'Load Smooth' and 'Load Scale'.

The FMOD Engine Designer has the following attributes.

Attribute	Sub-Properties	Description & Notes
RPM Slider	[none]	The RPM value of the engine.
RPM Smooth	[none]	The rate of change of the RPM parameter value.
Load Smooth	[none]	The rate of change of the Load parameter value.
Load Scale	[none]	The amount of change the RPM parameter invokes on the Load parameter value,

Audition 3D

The Audition 3D tool is specially designed to allow a sound designer to audition a 3D event in a 3D space and hear the audible impact of parameter changes in real-time.

The Audition 3D tool is shown below.

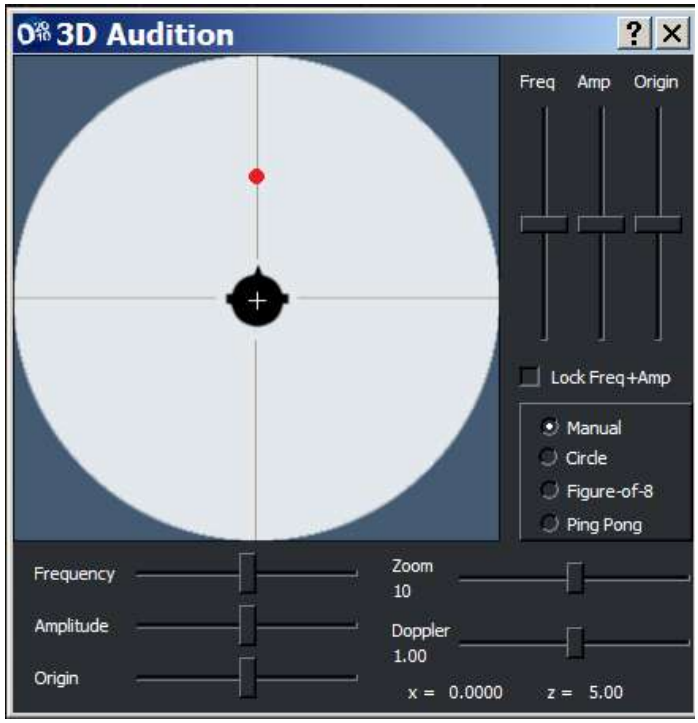


Figure 4-2: The 3D Audition pop-up window

The Audition 3D tool has the following attributes.

Attribute	Sub-Properties	Description & Notes
Front/Back Sliders		
	Freq	The rate of oscillation on the z-axis (front/back speakers).
	Amp	The amount of oscillation on the z-axis (front/back speakers).
	Origin	The starting position on the z-axis (front/back speakers).
Left/Right Sliders		
	Freq	The rate of oscillation on the x-axis (left/right speakers).
	Amp	The amount of oscillation on the x-axis (left/right speakers).
	Origin	The starting position on the x-axis (left/right speakers).
Lock Freq + Amp	[none]	Locks the front/back and left/right frequency and amplitude sliders together. Therefore a change in the front/back value is reflected in the left/right domain (and vice versa).
Movement Preset		
	Manual	Allows positioning of the event's location in the 3D sound field is manually positioned.
	Circle	The frequency, amplitude and origin are set to make the event's 3D position in the sound field move around the listener in a circular manner.
	Figure-of-eight	The frequency, amplitude and origin are set to make the event's 3D position in the sound field move in front of the listener in a figure of eight pattern.
	Ping Pong	The frequency, amplitude and origin are set to make the event's 3D position in the sound field move from left to right in front of the listener.
Doppler	[none]	A scaling factor that is applied to the Doppler effect.
Zoom	[none]	A scaling factor that is applied to the frequency and amplitude values.

Surround Pan

The Surround Pan tool is specially designed to allow a sound designer to select the position of an event in the surround sound field using a Graphical User Interface, rather than manually setting each channel volume independently.

The Surround Pan tool is shown below.

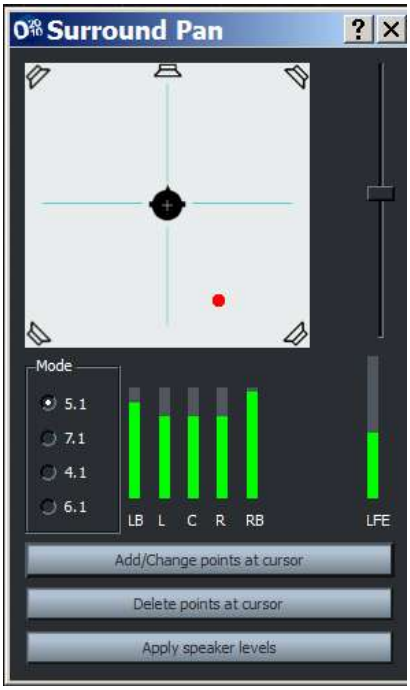


Figure 4-3: The Surround Pan pop-up window

The Surround Pan tool has the following attributes.

Attribute	Description & Notes
Position Cursor	A small red circle that indicates the position the event will appear in the surround field. The value of each channel will change as the cursor is moved.
Mode	Selects the number of channels used. • **4.1:** Selects 4 channels (plus LFE). • **5.1:** Selects 5 channels (plus LFE). • **6.1:** Selects 5 channels (plus LFE). • **7.1:** Selects 7 channels (plus LFE).
Levels	• ***LB:*** *Displays the signal volume in the left back channel.* • ***LS (7.1 and 6.1Mode Only):*** *Displays the signal volume in the left side channel.* • ***L****: Displays the signal volume in the left front channel* • ***C:*** *Displays the signal volume in the centre front channel.* • ***R:*** *Displays the signal volume in the right front channel.* • ***RS (7.1 and 6.1Mode Only):*** *Displays the signal volume in the right side channel.* • ***RB:*** *Displays the signal volume in the right back channel.* • ***LFE:*** *Displays the signal volume in the low frequency emitter channel.*

Attribute	Description & Notes
Add/Change points at cursor (button)	When pressed, the position in the surround field indicated by the position cursor will be applied to the selected event layer at the parameter cursor. Note: Before pressing this button, ensure that a 2D event, layer and parameter cursor is selected in the Event Editor View.
Delete points at cursor (button)	When pressed, the position in the surround field indicated by the parameter cursor (on the Event Editor View), will be applied to the selected event layer at the parameter cursor.
Apply Speaker Levels	

Cone Designer

The Cone Designer tool is specially designed to allow a sound designer to select the inside and outside angle of a 3D cone using a graphical representation, rather than manually setting each angle and outside level. For more information about the 3D Cone, see the 3D Cone entry under the section Event Properties in this chapter.

The Cone Designer tool is shown below.

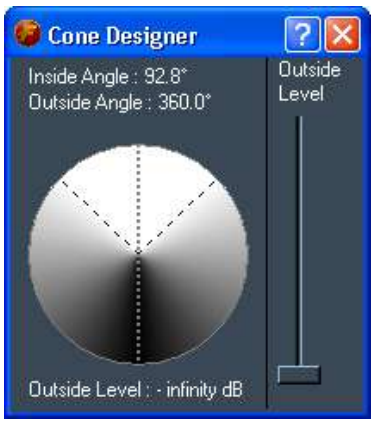


Figure 4-4: The Cone Designer Tool

The Surround Pan tool has the following attributes.

Attribute	Sub-Properties	Description & Notes
Inside Angle	[none]	The inside angle of the cone and the field of sound that will have full volume (no attenuation).
Outside Angle	[none]	The outside angle of the 3D cone and the field of sound that will be attenuated.
Outside Level	[none]	This property specifies the amount of attenuation a sound should receive when the listener is positioned in the outside angle.

Effects Reference Guide

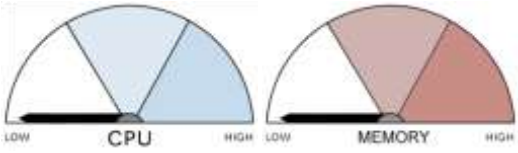
This section describes all the properties of effects used in FMOD Designer 2010. There are two types of effects:

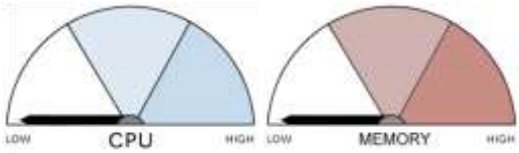
- Property Automation Effects. This type of effect provides the sound designer the ability to manipulate or sequence properties such as volume or pitch. These effects use no additional CPU.

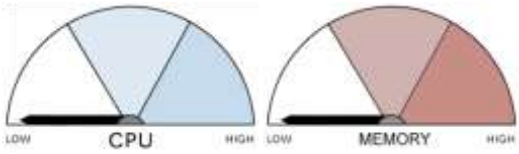
- FMOD DSP Effect. This type of effect uses DSP (digital signal processing) plug-ins to manipulate the audio stream in real-time, allowing the sound designer to radically alter a sound's timbre by adding distortion, filtering and echo. The properties of these effects can be set or sequenced using effect envelopes. In general, DSP plug-in effects require more CPU. Note: Versions of Designer 1.07.09 and above, support multiple pitch effects on each layer.

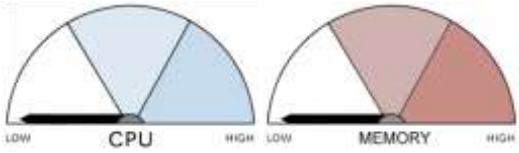
For each effect, a relative CPU and memory usage rating is given. The CPU and memory cost of each effect is categorized into four levels, those being none, low, medium and high.

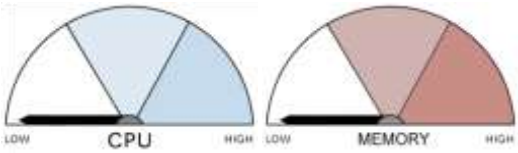
Property Automation Effects

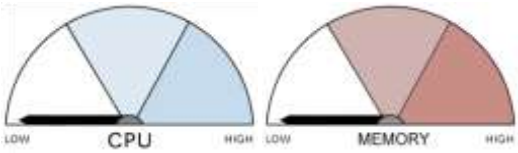
Effect & Properties	Description & Notes	Resource Usage
Volume	This effect can be used to control the volume of an event.	
Volume	This property (specified in dB) represents the overall volume of the event layer.	

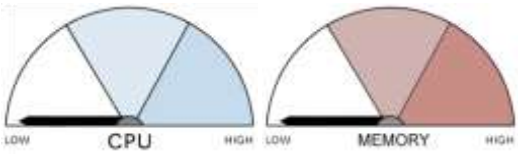
Effect & Properties	Description & Notes	Resource Usage
Pitch	This effect can be used to control the pitch of an event. The pitch is manipulated by varying the sampling rate of the playback. Unlike the 'Pitch shifter' DSP effect, this effect makes no attempt to retain the time duration of the sound.	
Pitch	This property specifies the desired change in pitch (measured in semi-tones).	

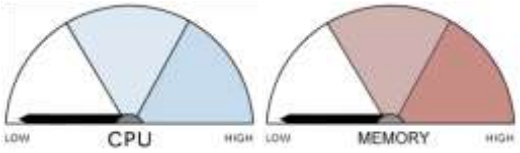
Effect & Properties	Description & Notes	Resource Usage
Spawn Intensity	This effect can be used to scale the spawn time range of an event.	
Intensity	Ranging from 0 to100, this value scales the minimum and maximum spawn times. The scaling factor is the average spawn time (max – min /2) divided by the Intensity Value (**Note:** the sound will not re-spawn when this value is 0).	

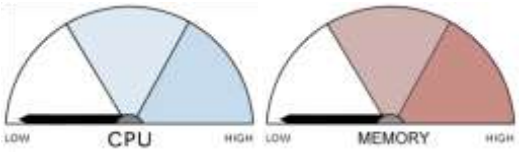
Effect & Properties	Description & Notes	Resource Usage
Pan	This effect can be used to dynamically control the panning of a Sound Def in the stereo field.	
Pan	This property specifies the desired position in the stereo field and ranges from –1.0 (left channel only) to 1.0 (right channel only).	

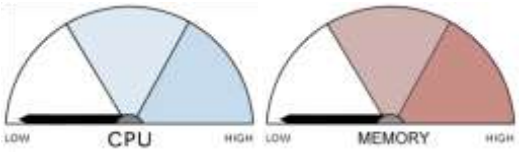
Effect & Properties	Description & Notes	Resource Usage
Surround Pan	This effect can be used to dynamically control the panning of a Sound Def in a surround sound field.	
Left / right Pan	This property specifies the desired stereo position, ranging from –1.0 (left channels only) to 1.0 (right channels only).	
Front / back pan	This property specifies the desired depth position, ranging from –1.0 (rear channels only) to 1.0 (front channels only).	
LFE send	This property specifies the amount of signal sent to the Low Frequency Emitter, ranging from 0.0 (no signal) to 1.0 (full volume).	

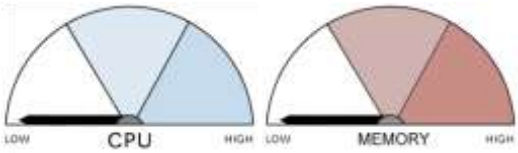
Effect & Properties	Description & Notes	Resource Usage
Occlusion	This effect controls the direct and reverb levels for the layer. **Note:** If the programmer sets the flag FMOD_INIT_SOFTWARE_OCCLUSION to true, the occlusion will be done using a low pass filter, otherwise it is rendered using volume attenuation.	
Direct	This property represents the attenuation of the direct signal.	
Reverb	This property represents the attenuation of the reverberated signal.	

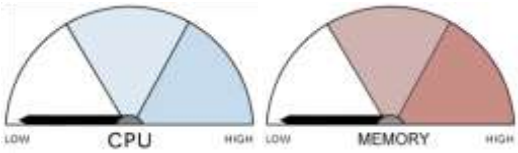
Effect & Properties	Description & Notes	Resource Usage
3D Speaker spread	This effect can be used to dynamically control the spread of sound in a surround sound field. The effect can be used to make the sound appear directional (coming from a particular speaker driver) or immersive (coming from all speaker drivers) in nature.	
Angle	This property specifies how directional the sound is, ranging from 0 degrees (highly directional) to 360 degrees (omni-directional).	

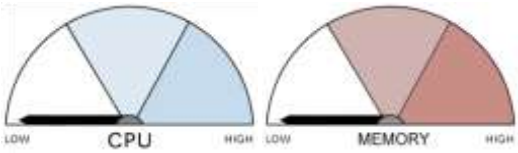
Effect & Properties	Description & Notes	Resource Usage
Reverb Level	This effect can be used to dynamically control the amount of reverb used by the event layer. This effect utilizes the reverb defined under the 'Reverb' View.	 Note: Whilst the automation of the reverb level requires minimal use of the CPU, the reverb effect itself requires significant CPU time.
Reverb Level	This property specifies the master volume of the reverb effect, ranging from 0db (full signal) to -60 db (approximately no signal).	

Effect & Properties	Description & Notes	Resource Usage
Reverb Balance	This effect can be used to adjust the reverberated signal of the event layer, proportionally to the unaffected signal. This effect utilizes the reverb defined under the 'Reverb' View.	 Note: Whilst the automation of the reverb level requires minimal use of the CPU, the reverb effect itself requires significant CPU time.
Dry Level	This property specifies the volume of the direct signal, ranging from 0db (full signal) to -60 db (approximately no signal).	
Wet Level	This property specifies the volume of the signal passed to the reverb effect, ranging from 0db (full signal) to -60 db (approximately no signal).	

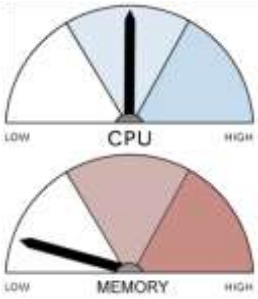
Effect & Properties	Description & Notes	Resource Usage
3D Pan Level	This effect can be used to dynamically morph between a 2D and 3D panning control.	
3D Pan Level	This property ranges from 0.00 (which causes the sound to pan according to 2D channel levels) to 1.00 (which causes the sound to pan according to its 3D position).	

Effect & Properties	Description & Notes	Resource Usage
Time Offset	This effect can be used to select an offset time starting point in the Sound Def. This effect allows the starting content in the audio file to be skipped. Note: When using one-shot Sound Defs, if the time offset value is greater than the length of the audio, no sound will be heard.	
Time Offset	This property specifies the time-offset, ranging from 0 to 60 seconds.	

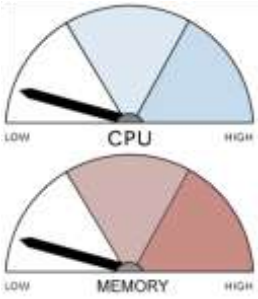
Effect & Properties	Description & Notes	Resource Usage
Channel Mix	This effect can be used to independently set the volume of channels within an audio file. Channels may also be grouped.	
Envelope	This parameter controls the volume of a channel (or group of channels) within a Sound File. This effect can be used in conjunction with Sound File 'Channel' property, found in the Wave bank View.	

Effect & Properties	Description & Notes	Resource Usage
Speaker Level	This effect can be used to set the volume of individual speakers. Speakers can also be combined into groups.	
Envelope	This parameter controls the volume of a speaker (or group of speakers). It is important to note that this effect gives direct access to the speaker level, and is **not** a panning effect. This effect is dependent on other mixing effects such as 3D calculations, panning, occlusion, etc. Therefore, if there is no signal in the speaker – any changes made via this effect cannot be perceived.	

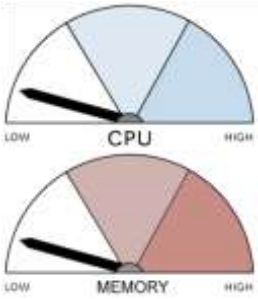
FMOD DSP Effects

Effect	Description & Notes	Resource Usage
FMOD Lowpass	This effect is a second-order (resonance) filter that leaves low frequencies unaffected but attenuates (or reduces) frequencies above a cutoff frequency point. This effect is used to reduce the amount of treble in a signal. Whilst the FMOD Lowpass effect is more computationally expensive than the other lowpass filters provided by FMOD, this effect produces a high-resolution emulation of an analog resonance filter.	

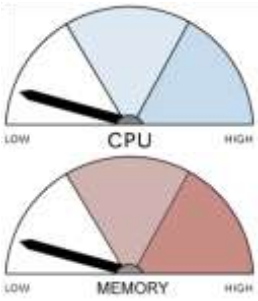
Property	Description
Cutoff freq	This property represents the frequency at which the filter will start attenuating. The cutoff frequency range is 10 to 22000Hz.
Resonance	This property alters the frequency response of the filter, by boosting the frequencies just below the cutoff frequency. The resonance property ranges from 1 to 10.

Effect	Description & Notes	Resource Usage
FMOD IT Lowpass	This effect is a second-order (resonance) filter that leaves low frequencies unaffected but attenuates (or reduces) frequencies above a cutoff frequency point. This IT effect was specifically designed to: 1. Support Impulse Tracker files. 2. Provide a resonant low pass filter at the cheapest CPU costs. This filter is limited to an effective cutoff of 10000Hz.	

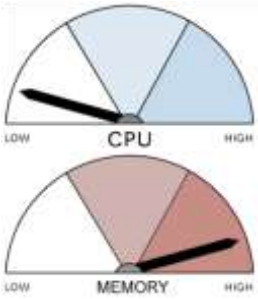
Property	Description
Cutoff freq	This property represents the frequency at which the filter will start attenuating. The cutoff frequency range is 1 to 22000Hz.
Resonance	This property alters the frequency response of the filter, by boosting the frequencies just below the cutoff frequency. The resonance property ranges from 1 to 10.

Effect	Description & Notes	CPU Usage
FMOD Lowpass Simple	This effect is a first-order filter that leaves low frequencies unaffected but attenuates (or reduces) frequencies above a cutoff frequency point. This effect is used to reduce the amount of treble in a signal. FMOD Lowpass Simple effect was specifically designed to provide full bandwidth response (up to 22kHz) at the cheapest CPU costs. It does not provide any resonance control.	

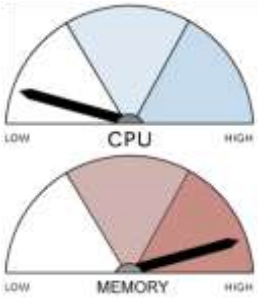
Property	Description
Cutoff freq	This property represents the frequency at which the filter will start attenuating. The cutoff frequency range is 10 to 22000Hz.

Effect	Description & Notes	Resource Usage
FMOD Highpass	This effect is a second-order filter that leaves high frequencies unaffected but attenuates (or reduces) frequencies below a cutoff frequency point. This effect is used to reduce the amount of bass frequency in a signal.	

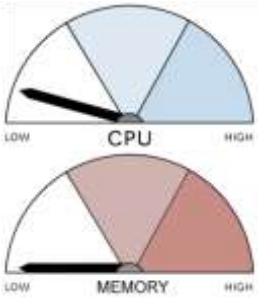
Property	Description
Cutoff freq	This property represents the frequency at which the filter will stop attenuating. The cutoff frequency range is 10 to 22000Hz.
Resonance	This property alters the frequency response of the filter, by boosting the frequencies just the above the cutoff frequency. The resonance property ranges from 1 to 10.

Effect	Description & Notes	Resource Usage
FMOD Echo	This effect creates a reverb like sound by replaying the Sound Def with additional time delays. This effect is less CPU intensive than reverb and useful for simulating environments such as large rooms, halls, caves, car parks, etc.	

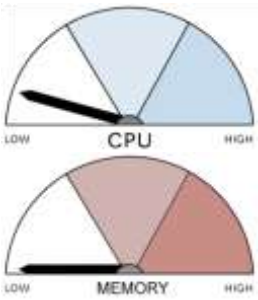
Property	Description
Delay	The property represents the time between the original sound and its echoed reflection. The delay can be set from 1ms to 5000ms.
Decay	The property represents the amount of echoed signal that is fed back into the echo plug-in. The decay can be set from 0.000 to 1,000 Note: Using 1,000 decay will cause an infinite feedback loop, meaning the echo will never stop.
Max Channels	This property sets the number of channels (and memory allocation) to be used from 0 (default) to 16. The value 0 represents FMOD's default polyphony, 1 = mono, 2 = stereo, etc.
Dry Mix	This property represents the volume of the original sound, ranging from 0.000 (fully attenuated) to 1,000 (no attenuation).
Wet Mix	This property represents the volume of the echo, ranging from 0.000 (fully attenuated) to 1,000 (no attenuation).

Effect	Description & Notes	CPU Usage
FMOD Flange	This effect is created with a signal that is combined with a slightly delayed copy of itself. Using a continuously variable time delay (of up to ± 10ms) causes the comb-filtering effect to move up and down the frequency spectrum, giving the filter its famous 'jet'-like sound.	

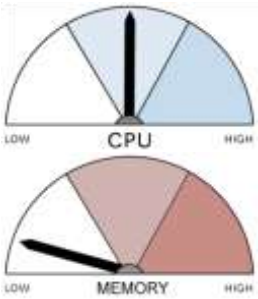
Property	Description
Dry Mix	This property represents the volume of the original sound, ranging from 0.000 (fully attenuated) to 1,000 (no attenuation).
Wet Mix	This property represents the volume of the delayed copy, ranging from 0.000 (fully attenuated) to 1,000 (no attenuation).
Depth	The depth property represents the maximum swing in delay time, ranging from 0.00 (± 0 ms) to 1.00 (± 10 ms).
Rate	The rate property represents the speed at which the comb filtering modulates from its lowest to highest depth. The rate can be set from 0Hz to 20Hz.

Effect	Description & Notes	Resource Usage
FMOD Distortion	Distortion adds harmonics and texture to an existing signal by boosting the signal's Sound File until clipping occurs. Whilst distortion reduces the fidelity of a sound, at times the increased harmonics and reduced dynamic range can add an appealing 'crunching' quality to the sound.	

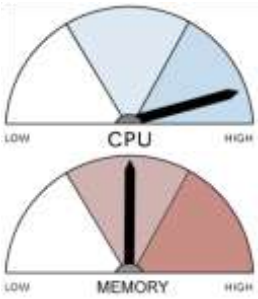
Property	Description
Level	This property specifies the gain adjustment to the signal ranging from –60dB (inaudible distortion) to 0 dB (extreme clipping).

Effect	Description & Notes	Resource Usage
FMOD Normalize	This effect amplifies the signal based on the maximum peaks within the signal.	

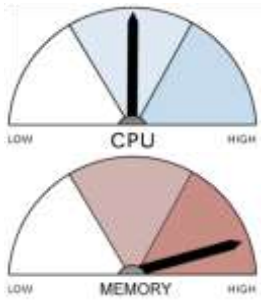
Property	Description
Fade in time	This property represents the time to attenuate the signal to silence, ranging from 0ms to 20000ms.
Lowest volume	This property represents the minimum threshold of the signal before normalization occurs, ranging from -60dB to 0db. When set correctly, this parameter is useful in stopping the amplification of any hiss, hum or noise in quiet sections.
Maximum amp	This property represents the additional gain boost to the signal, it accepts values as low as 0.000 and as high as 100000.000

Effect	Description & Notes	Resource Usage
FMOD ParamEq	This effect is a parametric equalizer that attenuates or amplifies a selected frequency (and its neighboring frequencies to a lesser degree).	

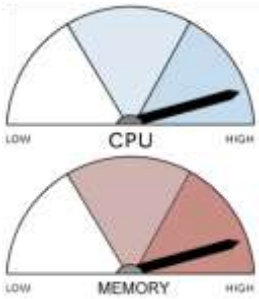
Property	Description
Center freq	This property represents the frequency to boost or cut, ranging from 20 Hz to 22 kHz.
Octave range	This property specifies how many neighboring frequencies will be affected. This is commonly known as the Q value. The octave parameter ranges from 0.2 octaves to 5 octaves (large overlap).
Frequency gain.	This property represents the boost or cut to the frequency. The values 0.05 to 3 represent a cut to the frequency, whilst 1.00 to 5.0 boosts the frequency.

Effect	Description & Notes	Resource Usage
FMOD Pitch Shifter	This effect allows a sound to be raised or lowered in pitch without changing its time duration. This plug-in is very CPU intensive. However, using mono files, selecting a smaller FFT size and reducing the sampling rate of the source file can reduce the CPU load (at the expense of sound quality).	

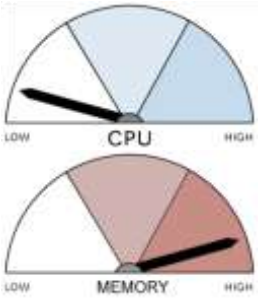
Property	Description
Pitch	This property represents the change to the original pitch ranging from 0.5 (one octave down) to 2.0 (one octave up). A value of 1.0 represents the original pitch.
FFT size	The property specifies the spectral resolution used in the FFT process. A higher number reduces the number of unwanted artifacts in the sound (but requires more CPU time). The FFT size ranges from 256 to 4096.
Max Channels	This property makes it is possible to set the number of channels (and memory allocation) to be used by the Pitch Shifter plug-in. This property can be set from 0 (default) to 16. The value 0 represents FMOD's default polyphony, 1 = mono, 2 = stereo, etc.

Effect	Description & Notes	Resource Usage
FMOD Chorus	This effect is used to simulate a sound coming from multiple sources. This chorus effect uses 3 chorus taps (copies of the original signal). Each tap is 90° out of phase with the previous tap.	

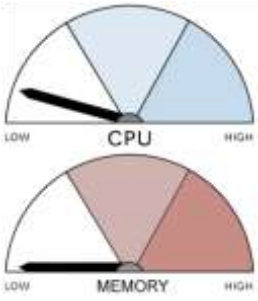
Property	Description
Dry Mix	This property represents the volume of the original sound, ranging from 0.00 (fully attenuated) to 1.00 (no attenuation).
Wet mix tap 1, 2 & 3	This property represents the volume of chorus taps, ranging from 0.00 (fully attenuated) to 1.00 (no attenuation).
Delay	The delay property specifies the total buffer size used by the effect. The greater this value, the more phase modulation occurs. The delay ranges from 0ms to 100ms).
Rate	This property represents the speed of the chorus modulation. The rate can be set from 0Hz to 20Hz.
Depth	The depth property represents the modulation of the time delay, ranging from 0.00 to 1.00.
Feedback	The feedback property controls how much of the wet (chorus) signal is placed back into the chorus buffer. It can range from 0.000 to 1,000

Effect	Description & Notes	Resource Usage
FMOD Reverb **(Deprecated)**	This effect is a reverb that can be used independently from the system reverb defined under the 'reverb' View. This DSP effect is stereo only; meaning that only the first two channels of the 5.1 outputs will be affected. Whilst the quality of this reverb is inferior to the SFX reverb, it is less demanding in terms of CPU time. **Note: Whilst supported for older titles, this effect is no longer available for new projects. Firelight Technologies strongly encourages it users to now use SFX Reverb.**	

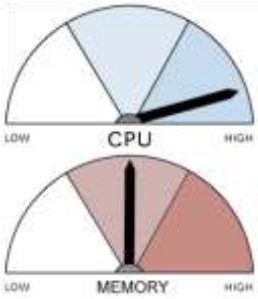
Property	Description
Roomsize	The property 'Roomsize' represents the physical size of the acoustic space to be simulated. The value ranges from 0.0 (a small space) to 1.0 (a large space).
Damp	The property 'Damp' specifies the amount of high frequency dampening applied to the reverberated signal.
Wet	This property represents the volume of the reverberated sound, ranging from 0% (fully attenuated) to 100% (no attenuation).
Dry	This property represents the volume of the original sound, ranging from 0% (fully attenuated) to 100% (no attenuation).
Width	Ranging from 0.0 (mono) to 1.0 (full stereo separation) the 'Width' parameter represents the amount of stereo separation in the reverberated signal.
Mode	The 'Mode' parameter switches the reverb's decay to infinite. Value 0.0 represents off, while 1.0 represents on.

Effect	Description & Notes	Resource Usage
FMOD IT Echo	This effect creates a reverb-like sound by replaying the Sound Def with additional stereo time delays. This IT effect is specifically designed to emulate DirectX effects and support Impulse Tracker files.	

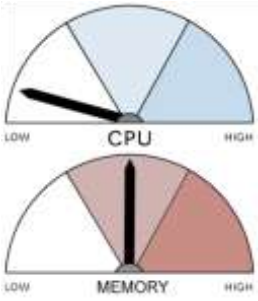
Property	Description
WetDryMix	This property represents the proportion of the original signal and echoes in the final output.
Feedback	The property represents the amount of echoed signal that is fed back into the echo plug-in. The decay can be set from 0% to 100%. Note: Using 100% decay will cause an infinite feedback loop, meaning the echo will never stop.
LeftDelay	The property represents the time between the original sound and its echoed reflection in the left channel. The delay can be set from 0ms to 2000ms.
RightDelay	The property represents the time between the original sound and its echoed reflection in the right channel. The delay can be set from 1ms to 2000ms.
PanDelay	This property sets the delay to switch stereo sides with each repetition. This behavior is commonly referred to as a''ping-pong delay'. The value 0.0 represents off, while 1.0 represents on. This feature is currently **not** supported by the FMOD Ex engine.

Effect	Description & Notes	Resource Usage
FMOD Compressor	A compressor is a gain reduction device used to reduce the dynamic range of a signal. The main role of this effect is to stop the audio from clipping the FMOD software mixer. A compressor may also be used to increase the perceptual volume of an audio signal.	

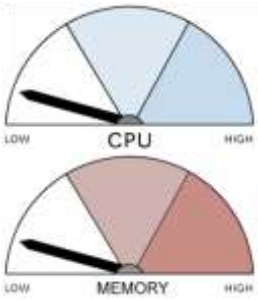
Property	Description
Threshold	This property represents the volume at which the compression will begin. This parameter ranges from 0dB (no compression) to –60dB (maximum compression).
Attack	This property specifies the speed in which the compressor will react to a signal above the set threshold level. The value ranges from 10 to 200 ms. Note: While a small attack time will allow the compressor to catch all transient peaks, it may cause some audible distortion artifacts.
Release	This property specifies the speed in which the compressor will release the signal back to its original gain level. The value ranges from 20 to 1000 ms.
Make up gain	This property represents the desired gain boost to the entire signal after peak limiting has been applied. It can range from 0.00 to 30.00.

Effect	Description & Notes	Resource Usage
SFX Reverb	This effect is a reverb that can be used independently from the system reverb accessed through the Reverb Def pop-up window. The quality of this reverb is superior to the FMOD reverb but uses more CPU time. This effect models both the early reflections and longer reverb swell (called 'room') present in real-world reverberation.	

Property	Description
Dry Level	This represents the level of dry signal in the output signal. The dry level ranges from –10000.0 to 0.0 mB.
Room	This property represents the volume level of the ambient reverberation. The room level ranges from –10000.0 to 0.0 mB.
Room HF	This property represents the overall high frequency attenuation of the room reverberation. The attenuation ranges from –10000.0 to 0.0 mB (no attenuation).
Room LF	
Room Reference	
Room rolloff	This property specifies the room rolloff factor and ranges from 0.000 (default value) to 10,000
Decay time	This property specifies the length of time the room reverberation takes to fade to silence. Ranging from 0.1 to 20.0 seconds, this feature can be used to simulate the size of the acoustic environment.
Decay HF ratio	This property represents the ratio of the high frequency and low frequency content in the reverberation decay. It can range from 0.100 to 2.00
Reflections	The reflection property represents the volume level of the early reflections relative to the volume of room reverberation and ranges from –10000.0 to 1000.0 mB.
Reflect delay	This property specifies the delay time of the first reflection, ranging from 0.0 to 0.3 seconds.
Reverb	The reflection property represents the volume level of the late reflections relative to the volume of room reverberation and ranges from –10000.0 to 2000.0 mB.
Reverb delay	This property specifies the delay time of the late reflection (relative to the first reflection), ranging from 0.000 to 0.100 seconds.
Diffusion	The rate at which the density of the reverb increases during the decay. This echo density ranges from 0.0 to 100.0 percent.
Density	Represents how tightly the reflections in the decay are packed. This modal density ranges from 0.0 to 100.0 percent.
HF reference	This property specifies the high frequency value referenced by the other properties. The HF reference ranges from 20Hz to 20kHz.

Effect	Description & Notes	Resource Usage
FMOD Delay	This effect places a sample accurate delay on each channel of the audio file. This effect is useful for manipulating the phase between channels of an audio sample.	

Property	Description
Delay ch0..ch15	This property represents the delay length (in milliseconds) placed on the individual channel. **Note:** The output is 100% wet, and not a mixture of the original and delayed signal. This effect supports files up to 16 channels.
Max delay	Specifies (in milliseconds) the upper buffer size for all the delays used. For example, if maxdelay=1000 ms for a stereo channel, the Delay effect will set up two buffers (one for each channel) that are 1000 ms long.

Effect	Description & Notes	Resource Usage
FMOD Tremolo	This effect produces a periodic variation in the amplitude of the signal.	

Property	Description
Frequency	Specifies the speed at which the variation in amplitude occurs. Ranges from 0.01 to 20Hz. The default is 5Hz.
Depth	Specifies the amount of variation in amplitude. Ranges from 0 (no change) to 1 (full signal). The default is 1.
Shape	Specifies the LFO shape, morphing between triangle and sine. Ranges from 0 (triangle) to 1 (sine). The default is 0.
Skew	Specifies time-skewing of LFO cycle. Ranges from -1 to 1. The default is 0.
Duty	Specifies the LFO 'on-time', ranging from 0 (always off) to 1 (always on).. The default is 0.5.
Square	Specifies the flatness of the LFO shape, ranging from 0 to 1. The default is 0.
Phase	Specifies an offset to the phase of the LFO. Ranges from 0 (LFO starts at its low point) to 1. Setting the 'Phase' to 0.5 means the LFO at its high point, allowing the start of the signal to be heard. The default is 0.
Spread	Creates rotation / auto-pan effects in multichannel audio files by spreading the LFO phrase evenly between channels (**Note:** In 5.1 the LFE channel is synced to the center channel). Ranges from -1 to 1, with the default being 0.
	Notes: Settings for create basic LFO shapes: **Triangle**: shape = 0, skew = 0, square = 0 **Sine:** shape = 1, skew = 0, square = 0 **Rising saw**: shape = 0, skew = 1, square = 0 **Falling saw**: shape = 0, skew = -1, square = 0 **Square**: skew = 0, square = 1 When using a pure square or saw tooth saw LFO shape, the presence of clicks should be expected as a natural byproduct of instantaneous volume change. Clicks can be avoided by setting 'skew' and 'square' values within their extremes (not ±1). This provides some time to ramp the volume, and avoid clicks.

CHAPTER FIVE: Tutorial Projects

5

FMOD Designer 2010 is capable of producing highly realistic and dynamic aural environments. In this chapter, a series of examples will be created in a step-by-step fashion. These examples will demonstrate advanced techniques, such as multiple parameters, multiple layers, randomization effects.

Key topics in this chapter:

- Advanced FMOD Designer 2010 techniques
- Multiple Layers and Multiple Parameters
- Using Sustain Points
- Creating Complex Sound Defs

Creating a Realistic Engine

Creating a great engine sound is a goal for many a sound designer. The success of today's favorite car titles is in part determined by how well the individual details of each car are modeled.

In this tutorial, the basic method of producing an engine event is covered. The sound of the engine is based on two parameters, the first represents the RPM of the engine and the second represents the engine load.

This following tutorial demonstrates:

- Using multiple parameters
- Crossfading between Sound Def instances
- Using the auto-pitch feature
- Auditioning events using tools

Creating the Event Structures

1. Create new event. Do this by right clicking on an event group, and selecting 'Add Multi-Track Event' from the sub-menu that appears.
2. Set the following event properties:
   - Name: Engine
   - Mode: 3d
3. Click the ‘Engine’ event in the Events Browser pane. The event will now be displayed in the Event Editor pane.
4. Add a parameter by right-clicking the message ‘Right-click here to add an event parameter’. A Menu pop-up window should appear.
5. Select the option 'Add parameter'.
6. Right-click the new parameter. A Menu pop-up window should appear.
7. Select the option 'Parameter properties...'. A pop-up window should appear.
8. Set the parameter range from 0 to 8800. This range represents the RPM range of the engine being modeled. Other engines may use a different range. Set the following parameter properties for 'param00':
   - Name: rpm
   - Minimum Value: 0.000
   - Maximum Value: 8800.000
   - Ruler spacing: 1000.000

   Press the 'OK' button.
9. Add another parameter.

10 Apply the following properties to the new parameter:
- Name: load
- Minimum Value: 0.000
- Maximum Value: 1.000
- Ruler spacing: 0.100

Press the 'OK' button.

11 Rename the exist layer ('layer00') to 'onload'. This is done by right-clicking the Layer name and selecting the option 'Layer properties...' from the Menu pop-up window. This layer will contain the sounds of the engine under load.
12 Set the 'name' property to 'onload',
13 Press the 'OK' button.
14 Create a new layer. This is done by right-clicking on the layer panel and selecting the option 'Add layer' from the Menu pop-up window.
15 Rename the new layer ('layer01') to 'offload'. This is done by right-clicking the Layer ('layer01') name and selecting the option 'Layer properties...' from the Menu pop-up window.
16 Set the 'name' property to 'offload'. This layer will contain the sounds of the engine when not under load.
17 Press the 'OK' button

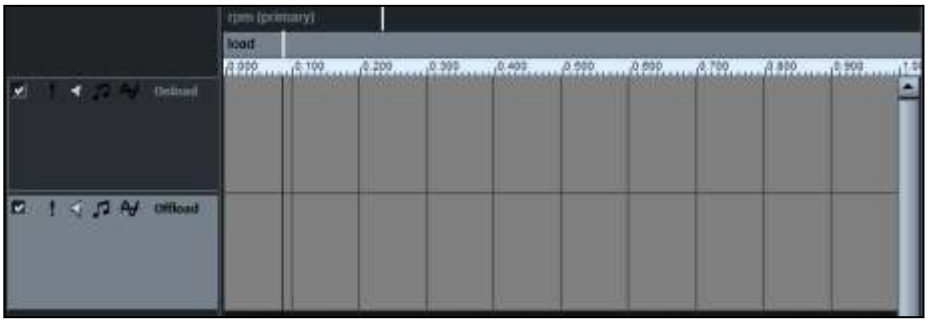


Figure 5-1: The Engine Event - Layers and Parameters

Organizing the Sound Defs

The tutorial continues with construction of the Sound Defs and the Sound Def instances.

18 Press [F4] to switch to the 'Sound Def' View.
19 Open an explorer window (using the operating system) and open the directory holding the car engine samples, by default examples\media\car in the directory in which FMOD Designer 2010 has been installed. This tutorial is using the files 'idle.ogg', 'onlow.ogg', 'onmid.ogg', 'onhigh.ogg', 'offlow.ogg', 'offmid.ogg' and finally 'offhigh.ogg'. Select all the supplied '.ogg' files and 'drop and drag' them into the Sound Def Browser pane. Seven new Sound Defs will be added.

20 Starting with the layer 'onload', add the following sound instances (in the following order):
- idle
- onlow
- onmid
- onhigh

Add an instance of the Sound Def by right-clicking the layer space. A pop-up window should appear. Select the option 'Add Sound...'. A pop-up window should appear. Select the Sound Def to add from the listbox and then press the 'OK' button.

21 In this step, the sounds are placed in the layer, roughly matching the RPM recorded in the audio file with the appropriate parameter value. Reposition each instance according to the following values:
- idle: Start 0 Length 1800
- onlow: Start 1400 Length 3500
- onmid: Start 3900 Length 3600
- onhigh: Start 6300 Length 2500

Note how FMOD Designer 2010 automatically creates crossfades between the overlapped Sound Def instances.
The desired layout is demonstrated in the following graphic.

Figure 5-2: The Engine Event - Sound Instance Layout

Using the Auto-Pitch Feature

The tutorial continues using the auto-pitch feature to create a smooth transition from 0 to 8800 RPM.

22 If you audition the event at this point (press [SPACEBAR] and drag the parameter cursor using the mouse), you will notice the engine does not change smoothly, instead jumps between the different RPM sounds. To fix this, the auto-pitch feature must be used. The auto-pitch function automatically increases the pitch of a sound as the parameter is increased. In this case the auto-pitch reference is the RPM value that the engine was recorded at. To apply the auto-pitch feature, right-click on the idle Sound Def instance. A Menu pop-up window should appear.

23 Select the option 'Sound instance properties...'. A pop-up window should appear.

24 Set the following properties for the 'idle' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 1135 (remember to press [enter])

This and the following auto-pitch references match the RPM of recorded engine.

25 Set the following properties for the 'onlow' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 3000

26 Set the following properties for the 'onmid' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 4000

27 Set the following properties for the 'onhigh' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 6000

28 If the crossfades are set correctly, the pitch of the engine should now move smoothly through the RPM range.

Creating the Offload Layer

The tutorial continues with the construction of the 'offload' layer. Much of this section repeats techniques discussed previously.

30 Starting with the layer 'offload', add the following Sound Def instances (in the following order):
- idle
- offlow
- offmid
- offhigh

31 Reposition each instance according to the following values:
- idle: Start 0 Length 1800
- offlow: Start 1400 Length 3500
- offmid: Start 3900 Length 3600
- offhigh: Start 6300 Length 2500

32 Set all crossfade types in the event to 'Bezier' and set the volume as demonstrated in the figure below.

33 Set the following properties for the 'idle' Sound Def instance (on the 'offload' layer):

- Autopitch enabled: Yes
- Autopitch reference: 1135

34 Set the following properties for the 'offlow' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 3000

35 Set the following properties for the 'offmid' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 4000

36 Set the following properties for the 'offhigh' Sound Def instance:
- Autopitch enabled: Yes
- Autopitch reference: 6000

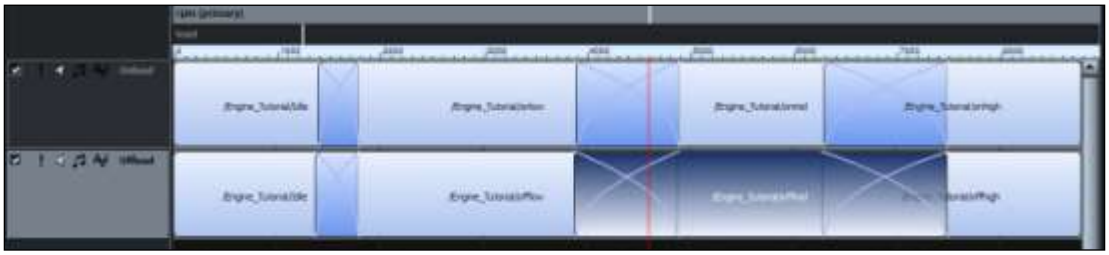


Figure 5-3: The Engine Event - Laying Out The Second Layer

Applying Volume Effects

In this section, volume effect envelopes on the load parameter are specified. These volume envelopes allow the FMOD engine to mute either the 'offload' or 'onload' engine sounds based on the load parameter. This helps to create a more realistic engine sound.

37 Making sure the 'load' parameter is selected by clicking on it. Add a 'Volume' effect to each layer. A layer effect is added by right-clicking on the layer name and selecting 'Add Effect...' from the menu option. A pop-up window should appear.

38 Select the effect 'Volume' from the list and press the 'OK' button. (Remember to repeat for the second layer)

39 Add two points to the Volume effect line (a point can be created using [SHIFT] + click on the line)

40 The volumes curve shape should be set 'Flat Ended', to do this right-click the point and select the curve shape.

41 Position the points on Layer 'onload' so that:
- point 1 at 0 load is set to -inf dB
- point 2 at 0.477273 load is set to -inf dB

- point 3 at 0.534091 load is set to 0 dB

42 Position the points on Layer 'offload' so that:

- point 1 at 0 load is set to -2.94 dB
- point 2 at 0.477273 load is set to -2.94 dB
- point 3 at 0.534091 load is set to -inf dB

This arrangement is demonstrated in the figure below.

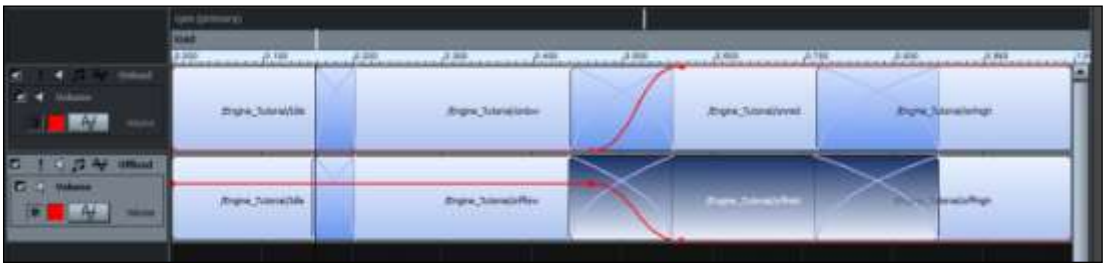


Figure 5-4: The Engine Event - Complete Layout

Auditioning the Engine

The event is now complete. The tutorial continues with the auditioning of the engine.

43 The engine event is now complete. Press the [SPACEBAR] key to start the auditioning process.

44 To audition the engine, open the Window menu and select the menu item FMOD Engine Designer. The FMOD Engine Designer pop-up window should open.

45 Using the RPM slider, you can hear the engine increasing and decreasing in RPM and load.

The advantage of the FMOD Engine Designer is, a single fader can control two parameters and any changes to the parameters are smoothed, making the audition more like testing the engine event with the data from the game engine.

Remember the FMOD engine designer is only used for auditioning and is not included with the game title – instead, the game engine should supply the required data.

Creating a Cycle

Many sounds require a sequence that begins with a starting sound, then looped in a sustained sound and are concluded with a stopping sound. If the length of the time of sustained sound is static, a single sample event can be used, but what happens if the sustained loop is dynamic in length?

In such a case, a more complex event is needed, one that uses a sustain point. This tutorial demonstrates how sustain points can be used to create a 'cycle' in FMOD Designer.

This method could be applied to objects such as a gatling gun, a jet engine, or anything else that requires a sequence of different sound samples.

The following tutorial demonstrates:

- Using the sustain points
- Using a velocity based parameter
- Auditioning events

Creating the Event Structure

The tutorial will begin with the creation on the event and layers.

1. Create a new event by right-clicking the event group and selecting 'Add Multi-Track event'
2. Set the following properties of the new event.
   - Name: Cycle
3. Click the 'Cycle' event in the Event Browser pane to view it in the Event Editor pane.
4. Three sounds need to be added to the layer. This tutorial is using 'cycle-start.ogg', 'cycle-sustain.ogg' and 'cycle-stop.ogg'
5. To add 'cycle-start.ogg' to the layer, right-click the layer space, a Menu pop-up window should appear.
6. Select the option 'Add sound...'. A pop-up window should appear.
7. Press the 'New Sound File' button and a select the file 'cycle-start.ogg' from the file dialog.
8. Set the loop mode to 'oneshot'.
9. Select the Sound Def 'cycle-start.ogg', from the listbox and press 'OK'. The Sound Def instance will be added to the layer.

10 To add 'cycle-sustain.ogg' to the layer, right-click the layer space, a Menu pop-up window should appear.
11 Select the option 'Add Sound...'. A pop-up window should appear.
12 Press the 'New Sound File button and a select the file 'cycle-sustain.ogg' from the file dialog.
13 Set the loop mode to 'looping'.
14 Select the Sound Def 'cycle-sustain.ogg', from the listbox and press 'OK'. The Sound Def instance will be added to the layer.
15-To add 'cycle-stop.ogg' to the layer, right-click the layer space, a Menu pop-up window should appear.
16 Select the option 'Add Sound...'. A pop-up window should appear.
17 Press the 'New Sound File' button and a select the file 'cycle-stop.ogg' from the file dialog.
18 Set the loop mode to 'oneshot'.
19 Select the Sound Def 'cycle-stop.ogg', from the listbox and press 'OK'. The Sound Def instance will be added to the layer.

Figure 5-5: Cycle Event - Adding The Sound Defs

Note:

At this point the order of the instances will be backwards, don't panic they are re-organized later in the tutorial.

Adding a Velocity-based Parameter

To create a full cycle, the parameter must move from a starting sound, to a sustained sound where it loops until the 'key off' signal is triggered and then moving to an ending sound.

The tutorial will now continue with the creation of a parameter that uses a velocity to move over time.

20 Add a parameter (if one does not already exist). To do this, right-click the message '(Right-click here to add an event parameter)', and the Menu pop-up window should appear.
21 Select the option 'Add parameter'.

22 Edit the parameter properties. To do this, right-click on the 'param00' parameter and the Menu pop-up window should appear.
23 Select the option 'Parameter properties…'.
24 Set the following properties for the parameter:
- Name: Cycle
- Minimum Value: 0.000
- Maximum Value: 5.000
- Units: Seconds
- Ruler spacing: 1.000
- Velocity: 1.000

Press 'OK' to set the parameter properties.

Figure 5-6: Cycle Event: Adding a Parameter

Adding a Sustain Point

The tutorial continues with the addition of a sustain point. The sustain point is required to make the parameter remain on a particular looped sound instance.

25 Add a sustain point, by first right-clicking on the ruler that appears just below the parameter 'cycle'. A Menu pop-up window should appear.
26 Select the option 'Add Sustain Point'. A yellow sustain point (with vertical line) should appear.
27 (Click and hold, then) Drag the Sustain Point to the position '2.5' on the ruler. This should be in the middle of the 'cycle-sustain' instance. The sustain point can also be set numerically by double-clicking the sustain point icon, entering the value 2.5 into the value field and pressing [OK].
28 Drag and reorder the instances as 'cycle-start', 'cycle-sustain', and 'cycle-stop'. To position the Sound Defs evenly across the layer, right–click on the layer space, a select the option 'Lay out sounds evenly' from the Menu pop-up window. A confirmation message should appear.
29. Press the 'YES' button. The events should be of equal size across the layer.

Figure 5-7: Cycle Event: Complete Layout

Note:

To make a smoother transition of sounds, the borders of the 'cycle-sustain' instance can be extended to create crossfades between the other instances.

Auditioning the Event

The event is now complete and is ready to be auditioned. The tutorial continues.

30 Press the [SPACEBAR] key. The parameter should move automatically from left to right until it hits the sustain point. At this point the 'cycle-start' instance has played, and playback has transitioned into the looping 'cycle-sustain' instance.
31 To transition from the ' cycle-sustain' instance to the ' cycle-stop' instance press the button 'Key off' which is located above the parameter.

This tutorial demonstrated how to produce an event cycle which moves automatically from a starting sound to a sustained sound, then finally to a stopping sound.

Simulating Distance

As sound travels through the air, frequency energy dissipates at different rates as the general volume reduces. These 'rolloff' characteristics are highly complex, and are influence by atmospheric conditions and surrounding materials. In this example, a technique used to simulate the perceived affects of distance to a listener is presented. The technique uses a combination of volume attenuation, low pass filtering (from the 'Occlusion') effect and 3D to 2D pan morphing.

The following tutorial demonstrates:

- Using a distance parameter
- DSP effects
- Morphing between 3D and 2D sounds
- Auditioning events

Creating the Event Structure

The tutorial will begin with the creation on the event and layers.

Create a new event by right-clicking the event group and selecting 'Add Multi-Track Event'

2 Set the following properties of the new event.
   - Name: SimDistance
   - Mode: 3d
   - 3D Rolloff: Custom
   - 2D Speaker L: -3.01
   - 2D Speaker C: -60
   - 2D Speaker R: -3.01
   - 2D Speaker LS: -60
   - 2D Speaker RS: -60
   - 2D Speaker LR: -60
   - 2D Speaker RR: -60

3 Double-click the 'SimDistance' event in the hierarchy tree view to open it in the 'Event Editor' View.

4 To add a sound to the layer, right-click the layer space. A Menu pop-up window should appear.

5 Select the option 'Add Sound...'. A pop-up window should appear.

6 Press the 'New wavetable' button and a sound to add. In this case the file 'drum-loop.ogg' is being used.

7 Select 'drum-loop' from the listbox and press 'OK'. The Sound Def instance will be added to the layer.
8 Press [CTRL] and double-click the Sound Def instance to fill all the available area. If the Sound Def does not fill the entire layer, hold down [ctrl] and double-click it.

Figure 5-8: Simulating Distance - Sound Def

9 If a default parameter doesn't exist (i.e. 'param00'), right-click the message '(Right-click here to add an event parameter)'. A Menu pop-up window should appear.
10 Select the option 'Add 3D distance parameter'.
11 We will specify the how the 3D rolloff behaves over 100 yards. To do this, the parameter range will need to be set appropriately. Right-click the parameter. A Menu pop-up window should appear.
12 Select the option 'Parameter properties'.
13 Set the following properties for the (distance) parameter:
- Minimum Value: 0.000
- Maximum Value: 100.000
- Units: Yards
- Ruler spacing: 10.000

Press 'OK'.

Figure 5-9: Simulating Distance - Distance Parameter

Adding the Rolloff Effects

In this section of the tutorial, the 'Occlusion' effect is added to create the custom rolloff. The 'Occlusion' effect is a great choice for producing both volume and high frequency attenuation. This effect automates the channel's volume and low pass filter in FMOD Ex's DSP Mixer. As these processes already exist (for standard playback), the 'Occlusion' effect is computationally inexpensive, as no additional DSP processing is required.

14 The first characteristic of the rolloff to model is the volume decay, using the 'Occlusion' effect. To do this, right-click the layer panel, a Menu pop-up window should appear.
15 Select the option 'Add effect'. A pop-up window should appear.

16 Select 'Occlusion' from the effect listbox and press 'OK'. Effect envelopes should appear over the layer.

17 Two envelopes should appear on the layer, one of which should be overlapping the other at the bottom of the layer. Click and drag the envelopes so that they are separated.

18 By consulting the Effect Color Key on the Layer panel, determine which of the two envelopes represents the ‘Direct’ effect property. Right-click on this envelope, then select ‘Add point’ from the menu that appears.

19 Double-click on the newly created point. A property box labeled ‘Occlusion handle’ should appear.

20 In this box, set the ‘(distance) (Yards)’ property to 50, set the ‘Direct’ property to 1, then click the ‘OK’ button. The envelope will change, indicating complete attenuation of the dry signal at 50 yards.

21 At the far left end of the envelope another point should already exist. Double-click on it.

22 Another ‘Occlusion Handle’ property box should appear, this time with only a single ‘Direct’ property. Set this to 0, then click the ‘OK’ button. The envelope should change again, so that there will be no direct occlusion at 0 yards distance.

The envelope of the direct signal is shown in the figure below.

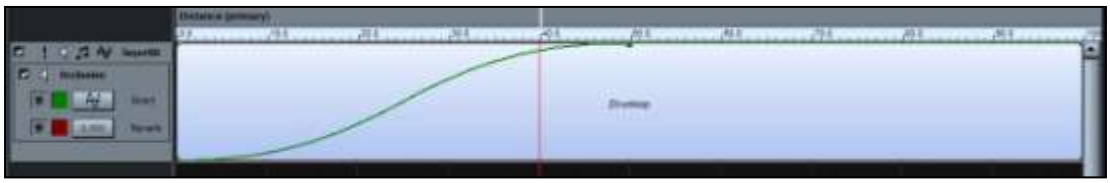


Figure 5-10: Simulating Distance – Direct

18 The next step is to add an envelope to control the reverberated signal. The reverb envelope created will become more perceptible as the listener moves away from the sound source.

Note:

To hear reverb while auditioning in Designer, you must activate it under the 'Reverb' View. For this exercise, a large easy to hear reverb such as the 'Cathedral Hall' preset is recommended. Adding reverbs is explained on page 80 of the manual.

Utilizing the same methods used to create and assign values to points o the Direct curve create the following points on the Reverb automation

- Point 1 – Distance: 0, Reverb: 1
- Point 2 – Distance: 20, Reverb: 0.78
- Point 3 – Distance: 100, Reverb: 1
- Set both segments to ‘Flat Ended’ by Right-Clicking on the curve segements

Both envelopes are shown in the figure below:

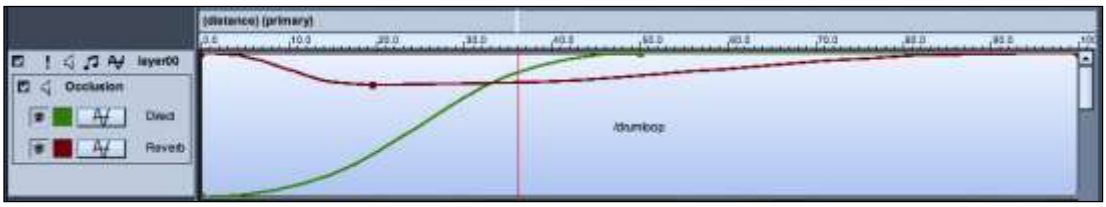


Figure 5-11: Simulating Distance – Reverb Envelope Added

19 The final method of simulating the distance uses a powerful feature of FMOD Ex - morphing 3D panned sound into 2D (and back again). When used, the drums will appear to immerse the listener at a very close distance, but will appear be very directional when the listener is a distance away. To do this, right-click the layer panel, a Menu pop-up window should appear.

20 Select the option 'Add effect'. A pop-up window should appear.

21 Select '3D Pan Level' from the effect listbox and press 'OK'. A new effect envelope should appear over the layer.

22 To create a sense of immersion when close to the sound source, the 3D pan level must be 0 for low (distance) parameter values. The pan level should then be set to 1 for higher parameter values. To do this press [SHIFT] and click on the 3D Pan Level effect envelope. This will add a point to the line.

23 Click and drag the left most point on the Level automation line to the bottom-left corner of the layer. Then click and drag the other point on the automation line to the top of the layer at the ten yard mark.

24 Change the envelope style of the 3D pan level by right-clicking the first point (of the Level effect envelope). A Menu pop-up window should appear.

25 Select the option 'Linear'. The envelope will redraw itself.

26 Move the parameter cursor to 0. The complete custom rolloff is shown in the figure below.

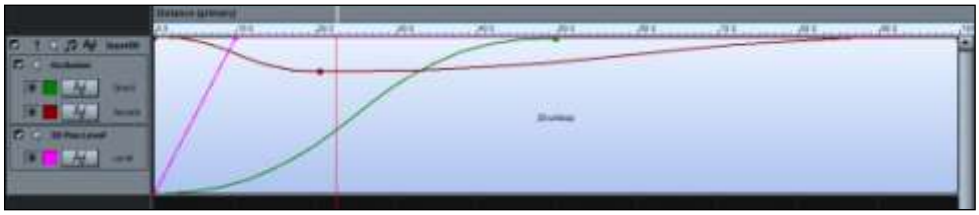


Figure 5-12: Simulating Distance: Complete Layout

Auditioning the Event

Now the event is complete, the simulated distance can be auditioned. To experience the full impact of the distance parameter a 5.1 speaker system is required.

27 Press the [SPACEBAR]. The audition should start.

28 Using the mouse to move the (distance) parameter to the right, note how the high frequencies and volume decrease as the value of the (distance) parameter is increased.

29 To hear the effect, open the Window menu and select Audition 3D. The Audition 3D pop-up window should appear.

30 Set the sliders as shown in the figure on the next page, making sure the zoom slide is set to 100.

Figure 5-13: The 3D Audition Tool

33 The event should now be automatically moving in a virtual 3D space. The drums should appear to be getting closer, immersing the listener (while in 2D panning) and then moving away. As the drums are in the distance, the high frequencies are filtered, and the reverb should be more obvious.

This tutorial demonstrated how to simulate distance using FMOD Designer 2010 and FMOD Ex.

Important Programmer Note:

The 'Occlusion' effect will only use a low pass filter when the flag FMOD_INIT_SOFTWARE_OCCLUSION is set to **TRUE** during the initialization of the FMOD system or event object. If this flag is not set the signal will only be rendered using volume attenuation.

Fun with Multi-Channel Files

This tutorial demonstrates how dynamically mixing the channels of a multi-channel audio file can be used to create sample accurate blends between audio content.

This tutorial uses the following audio file, 'MultiChannel.wav'

This tutorial demonstrates:

- Using a parameter
- Using an audio file with multiple channels
- Using the 'Channel Mix' effect
- Setting the 'Channel Setup' attribute of a wave bank
- Auditioning events

Multi-Channel Audio File

The tutorial requires use of a .wav file with four channels. This file will be created from two independent stereo audio files. This technique works very well with music. For example, the first stereo pair contains drums, the second pair contains more instrumentation. This scenario allows dynamic mixing of the instruments and drums while retaining accurate sample synchronization!

To create a multi-channel audio file you can use a number of tools including software such as 'wavewizard', 'Sony SoundForge 10' and FMOD's own FSBankEx, which is freely available from the FMOD Ex API Download.

Creating the Event Structure

The tutorial will demonstrate how to dynamically switch between a pair of stereo channels with sample accuracy.

1. Create a new Multi-Track Event by right-clicking the event group and selecting 'Add event'.
2. Set the following properties of the new event.
   - Name: FunWithMultiChannel
3. Click the ' FunWithMultiChannel ' event in the Event Browser pane to view it in the Event Editor pane.
4. Right click the empty layer. A context menu pop-up window should appear.
5. Select the option 'Add Sound…'. A window should appear.
6. Press the button 'New Sound File…'. A file open dialog should appear.
7. Select the file 'multi-channel.wav' and click the 'Open' button.
8. The filename should now be highlighted in the list of Sound Defs. Click 'OK'. A sound instance should appear in the layer.

Apply the Effect

At this point the basic elements of the event have been set. If the event is auditioned at this point, all channels of the file will be heard in its default channel routing – which can sound awful if the content of each stereo pair is not related! This section will describe how to use the effect 'Channel Mix' and envelope curves to hear one pair at a time.

9 Create a parameter by right clicking on the message 'Right-click here to add an event parameter', then select ‘Add parameter’ from the context-sensitive menu.
10 Right click on the layer panel. A Menu pop-up window should appear.
11 Select the option 'Add effect…'. A window should appear.
12 Select the effect 'Channel Mix' and click the 'OK' button. The effect should be added to the layer.
13 Right click on the effect panel. A menu should appear.
14 Select the option 'Effect properties…'. A window should appear.

Each 'channel' item in the list represents a volume envelope for a channel (or group of channels) in the audio file. In the case of this tutorial, the first envelope will control channel 0 and 1 (the first two channels in the audio file). The second envelope will control channels 2 and 3 (or the second pair of channels) within the audio file.

Figure 5-14: Fun With Multi-Channel Files – Channel Envelopes

15 Double click the 'Channel 0' list item. A list of available channels to control should appear.
16 Click the checkboxes for the channels 0 and 1 and click the 'OK' button. The first envelope will now control these channels.

17 Double click the 'Channel 1' list item. A list of available channels to control should appear.
18 Uncheck channel 1. Check the channels 2 and 3 and click the 'OK' button. The second envelope will now control these channels.

19 Now the two channel control envelopes have been configured. Click the 'OK' button. The envelope names should appear in the effects panel.
20 To gain easier access to the envelopes, change the base values of the effects by clicking on one of the numbered buttons in the Effect Panel, then adjusting the slider that appears.
21 Add two points to the first envelope by holding down Shift and clicking on the curve', then edit each point by double clicking it. Set the following:
   - Point 1 (dB)=0
   - Point 2 parameter=0.47 and (dB)=0
   - Point 3 parameter=0.53 and (dB)=-60
22 Add two points to the second envelope. Set the following:
   - Point 1 (dB)=-60
   - Point 2 parameter=0.47 and (dB)=-60
   - Point 3 parameter=0.53 and (dB)=0

Figure 5-15: Fun With Multi-Channel Files - Envelopes

Setting the Routing

FMOD Ex has some default routing that must be changed for this event to playback as intended. For example, a six channel file is generally routed in 5.1, with the first channel being panned to front left, the second panned to the front right, etc. In the case of music, it would be better to override the default behavior by having the file played as a collection of stereo pairs each routed to the front left and right channel (the channels could also be routed as mono channels to the center speaker).

23 Click the Banks View (or press the [F5] key)
24 Click the wave bank containing the multi-channel audio file. The files contained in the wave bank should appear in the Sound Files panel.
25 Right click the Sound File 'multi-channel.wav' (from the Sound File panel). A Menu pop-up window should appear.
26 Follow the option 'Channel Setup' and select the routing '2xstereo'. The routing uses the format 'n x stereo', dividing the file into pairs, routing the first channel to front left, the second channel to front right.

Auditioning the Event

The event is now complete and is ready to be auditioned.

27 Click on the ‘Events’ button to return to the Events view.
28 Press the [SPACEBAR] key.
29 To transition between the audible channels of the audio file, simply drag the parameter cursor over the instance.

This tutorial demonstrated how to dynamically mix the channels of an audio file and route the audio the front left and right speakers.

CHAPTER SIX: Optimization

6

For many sound designers, producing a high quality aural environment is not the only development requirement. Projects for game titles generally have many other requirements, such as processing and storage limits - sound designers spend much of their time fine-tuning the balance between quality and resource optimization.

In this chapter, a number of optimization techniques will be discussed. These techniques will help maximize CPU, memory and disk space usage.

Key topics in this chapter.

- Compression and File Formats
- Optimizing CPU and Memory Usage
- Disk Streaming
- Hardware Vs Software
- Maximum Playbacks
- Organizing Banks and Event Groups

File Formats

What is Compression?

PCM (pulse code modulation) is a standard method for representing an analogue audio signal in the digital domain. PCM data is a simple format, where the analogue audio signal is measured at regular intervals (known as the sample rate) and the measurement quantitized to a digital value. The sample rate and the amount of data used to store the quantitized measurement determine the quality of the PCM representation.

If PCM works, why are other formats needed? The biggest problem with the PCM format is the storage size. For example, a 3-minute song recorded in stereo at CD quality requires 254,016,000 bits (or approximately 30 MB)

To reduce this data bandwidth burden and reduce the required storage space, compression algorithms were created. There are 2 distinct types of compression algorithms, 'lossless' and 'lossy'.

Lossless compression algorithms allow the original data to be perfectly reconstructed from its compressed form. This means the size of the audio file can typically be reduced 30% (and up to 50% for particular files) without reducing the quality of the audio.

Lossy compression algorithms reduce the quality of the audio signal by attempting to remove data that is perceptually unimportant. This allows the signal to be reduced up to 80% of its original size with little distortion.

Choosing a File Format

It is important to select the best file format for each audio piece used in the project. FMOD Designer 2010 may export data to game titles using the following formats:

- PCM (PC, Microsoft Xbox 360, Nintendo Gamecube,
- Sony PlayStation 2,
- Sony PSP,
- Sony PlayStation 3,
- Nintendo Wii)
- ADPCM (PC, Microsoft Xbox 360, Sony PlayStation 3)
- XMA (Microsoft Xbox 360)
- GCADPCM (Nintendo Gamecube, Nintendo Wii)
- MPEG Layer 3 (mp3) (PC, Microsoft Xbox 360, Sony PlayStation 3, Nintendo Wii)
- MPEG Layer 2 (mp2) (PC, Microsoft Xbox 360, Nintendo Wii)
- VAG (Sony Playstation 2, Sony PSP)
- CELT (PC)

To best optimize audio Banks, the sound designer needs to balance CPU load, memory usage, disk space and disk access. There are two attributes of a wave bank that make a significant impact on performance those attributes being 'Compression type' and 'Wave bank type'.

Choosing the compression type is not as simple as choosing the file format with the smallest file size. File formats using variable bit-rates are capable of remarkable compression ratios (with little distortion), however require relatively high CPU load in comparison to PCM or ADPCM to decompress the audio before playback.

Note:

Some platforms use hardware decoders to negate this issue.

The PCM format uses no compression, therefore uses the most disk space but creates no additional CPU load for playback.

ADPCM (Adaptive Differential Pulse Code Modulation) and its variants use a simple compression method to reduce the required disk space. While ADPCM compression is classified as a 'lossy' compression, in some situations it is lossless. Typical compression ratios range from 3.5:1 to 4:1. ADPCM encoding provides a good compromise between file size, sound quality and the CPU load required for decompression.

The MPEG 1 Layer 3 data (MP3) and XMA formats are capable of compression ratios of about 10:1 with little effect on sound quality, but are the most CPU intensive when decompressing.

MPEG 1 Layer 2 data (MP2) is capable of compression ratios of around 6:1, but is requires little CPU load in comparison to MP3. MP2 also has the advantage over MP3 of not requiring an additional license for use.'

The wave bank attribute 'bank type' plays a significant role in the CPU loading and the amount of memory used. Choosing the best wave bank type requires the sound designer to consider both the nature of the audio piece and the behavior of the event that triggers the audio. There are three bank types available, those being:

- Stream from disk. This option causes the data to be read from disk and decompressed in real time for immediate playback. Using 'Stream from disk' is appropriate with large files such as background music. Using this method with small repetitive audio pieces causes a large processing and disk access overhead.

- Load into memory. This option causes the data to be loaded completely into memory in its current format. If required the audio is decompressed in real time each time the audio is played. The main benefit of this method is its reduced memory usage - the disadvantage being it comes at the cost of an additional CPU load each time the audio is played.

- Decompress into memory. This option causes the audio data to be decompressed, then loaded into memory in raw PCM form. This method requires CPU time during loading and will use more memory space than the other options. However there is no additional CPU load required for playback.

Recommended Formats

For some platforms, FMOD Ex has the ability to play compressed files without the need to decompress the data to PCM first. For example the Xbox 360 is able to playback XMA data without (any significant) CPU usage. This platform possesses audio hardware to carry out the XMA decompression, instead of using the CPU.

Below is a table listing the recommended file formats for each platform supported by FMOD. These recommendations are based on which format provides the most CPU efficiency for the platform (in most cases this is done by selecting the native format for the playback hardware).

Platform	Recommended Formats
PC	PCM (and MP2 for large files)
Xbox360	XMA
PSP	VAG
PS2	VAG
PS3	ADPCM (and MP3 for large files)
GameCube	GCADPCM
Wii	GCADPCM

This table is only a guideline, not a fixed rule. In many cases, memory and disk storage limitations may be imposed by the Programmer's specification - therefore the format choice will require extensive use of MP3, MP2 or XMA compression.

Efficient CPU Usage

As stated earlier, optimization is the process of minimizing the processing resources required to create the soundscape. The effect of file formats and wave bank 'bank types' on CPU, memory and disk space have been discussed in the sections above. In this section a number of CPU optimization techniques are considered.

DSP Effects

DSP effects are capable of adding an extra dimension to otherwise static sounds. Every DSP effect causes an additional load to the CPU - therefore the sound designer should choose DSP usage wisely. Where possible, the sound designer should look for opportunities to render an effect to file rather than use a real-time DSP effect.

Efficient Memory and Disk Usage

In this section, a number of memory optimization techniques are considered.

DSP Effects

Generally speaking, the memory footprint required for DSP Effects is small. DSP effects that are more memory intensive include, FMOD Echo, FMOD Chorus or FMOD Flanger effect. Other types of DSP effects generally do not allocate any memory.

Limiting Polyphony

It is possible to minimize the amount of memory required to play an event using the 'max playbacks' attribute (listed in the event properties, under the events View). FMOD will only allocate memory for the number of instances specified.

- If the 'Max playbacks' is set to 1, FMOD Ex declares enough memory to play one instance of the event.
- If the 'Max playbacks' is set to 3, FMOD Ex declares enough memory to play three instances of the event simultaneously.

Setting the 'Max playbacks' correctly requires the balancing of the 'artist vision' (or creating a realistic and immersive soundscape) and the memory resources required to play the all occurrences of an event.

For example, imagine a first person shooting game that contains 32 players and the event 'Footsteps' that contains the sound of running feet. Commonsense may suggest the 'Max playbacks' property of the 'Footsteps' event be set to 32, this way all players are represented.

Before doing so, the sound designer should consider:

- Does the player need to hear 32 pairs of feet?
- How many pairs of feet can the player distinguish (when other events are playing)?
- How much memory am I willing use for this sound? Is it really that important?

Perhaps when using anything more than three simultaneous 'Footsteps' events, it becomes hard to hear how many players are around anyway! Setting the 'max playbacks' to 3 means FMOD Ex will allocate much less memory than is required for 32 instances of 'foot steps'.

If the number of playbacks exceeds the number set in the 'Max playbacks' property, the FMOD Ex engine will 'steal' the audibility of one event occurrence and give audibility to another occurrence. This steal behavior is controlled using the 'Max playbacks behavior' property. The FMOD Ex engine can be set to:

- Steal oldest: the playback of the oldest event is stolen.
- Steal newest: the playback of the newest event is stolen.
- Steal quietest: the playback of the quietest event is stolen.
- Just fail: the playback of the current trigger event is ignored.
- Just fail if quietest: the event is not instantiated if it is calculated as being the quietest voice.

In the scenario described above, by selecting the 'Max playbacks behavior' property 'Steal quietest', it may be possible to keep the 'Max playbacks' at 3, thereby reducing the memory requirement while producing an artistically pleasing result.

Disk Streaming

As described in the section, 'Choosing a File Format', disk reading is used to read an audio file from disk for immediate decompression and real-time playback.

Choosing an appropriate 'max streams' value is very important. The 'max streams' value sets to the number of concurrent streams within a wave bank that can be played. Where possible, this value should be lower than the default '32' to minimize the amount of memory and disk access overhead.

As previously stated, streaming is best applied to the playback of large files such as music tracks, dialog or extended environmental audio tracks.

Compression Levels

When the 'compression quality' property of a wave bank is set as a 'variable bit rate' (such as MP2, MP3 or XMA), the 'compression quality' property may be set. This property allows the size of every file in the wave bank to be reduced using a single property.

As the 'compression quality' property is reduced (from 100 to 1), more 'lossy' compression is applied, resulting in a smaller file size with less quality.

If the amount of compression causes a small number of files within the waveband to exhibit unacceptable compression artifacts (distortion), the compression level of individual files can also be set.

To set the compression level of an individual file:

1 Click on the Banks View.
2 Click on the wave bank that contains the audio file to be adjusted. The wave bank's 'compression' property must be either MP2, MP3 or XMA (Xbox 360 only).
3 Right-Click on the desired filename in the 'Sound Files' panel (shown in figure 6-1). A Menu pop-up window should appear.
4 Select 'Adjust Sound File quality'
5 To increase the sound quality (and the file size) adjust the slider towards 100.
To reduce the file size (and the sound quality) adjust the slider towards 1.
6 Press the [▸] button (play) to preview the change.
7 If the sound quality is still unacceptable, repeat steps 3 through 6 until the desired result is achieved.

Manual Sample Rate Optimization

As explained at the beginning of this chapter, the sample rate represents the number of measurements taken from the signal each second. The largest frequency that can be represented by PCM data is equal to half the sample rate. Therefore the sample rate has a strong impact on the perceived sound quality.

For example, if an audio file only has frequency content below 4000Hz, then a sample rate of 44100 (which supports frequency content up to 22000Hz) is an inefficient use of file size (and memory). It would be better to reduce the sample rate towards 8000Hz

FMOD provides an automatic method for sample rate optimization, which can be activated for a file in the Waveforms list in the Banks view by right-clicking on it, selecting the ‘Adjust sample rate’ context menu item, then by clicking the ‘Automatic Rate’ radio button'. This method analyses the audio frequency content and determines the optimal sample rate. When the project is built, the audio file is converted to the new (optimized) sample rate.

Sometimes the automated optimizer is quite conservative in its sample rate reduction, and the sound designer may decide a further sample rate reduction would still have acceptable quality. To manually reduce the sample rate:

1 Click on the Banks View.

2 Click on the bank that contains the audio file to adjust.
3 Right-Click on the desired filename in the Sound Files pane (shown in the figure on the next page) A Menu pop-up window should appear.
4 Select 'Adjust sample rate'.
5 Ensure that either the ‘Automatic Rate’ or ‘Fixed Rate’ radio button is selected, then drag the slider downward to reduce the sample rate, or upward to increase it.
6 Press the [▸] button (play) to preview the change.
7 If the sound quality is still unacceptable, repeat steps 5 and 6 until the desired result is achieved.

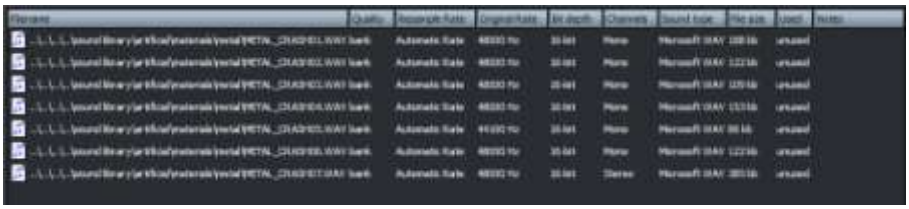


Figure 6-1: The Sound Files Pane

Sample rate optimization may also be disabled per Sound File. To disable optimization on a Sound File:

1 Click on the Banks View.
2 Click on the Bank that contains the audio file to adjust.
3 Right-Click on the desired filename in the 'Sound Files' panel (shown in Figure 6-1) A Menu pop-up window should appear.
4 Select 'Adjust sample rate'
5 Click on the ‘No resampling’ button.
6 Click outside the pop-up tool to close the optimizer pop-up window.

CHAPTER SEVEN: Advanced Techniques

7

FMOD Designer 2010 can be used in many creative ways. In this section, a number of tips and techniques that can be applied to typical in-game scenarios are discussed. In some situations, multiple options are considered, with discussion of the advantages and disadvantages for each solution.

Key topics in this chapter.

- Defining sequences with 'Key off on silence'
- Creating random loops using spawning
- Selecting Sound Files to play at runtime

Introduction

This section of the user manual discusses some FMOD Designer 2010 tips and techniques that can be applied to typical in-game scenarios. For each technique, a scenario is described and possible solutions are considered. **Remember** that these techniques can be **combined** to produce new possibilities and even more creative audio content!

Note:

Take a look at the FMOD Wiki: (http://www.fmod.org/wiki) for upcoming sample project files, techniques, features and more!

Sequences with 'Keyoff on Silence'

Scenario

Have you ever wondered how to create an event that contains a sequence of Sound Defs? Sure it is possible to create a parameter, set up a velocity and then manually size the sound instances appropriately – but there is an easier way, which removes the need for time keeping!

Solution

This solution will take advantage of the 'Keyoff on silence' behavior and is perfect for playing a sequence of oneshot sounds. Try the following:

- Set the event 'Oneshot' property to 'Yes'.
- Add a Parameter to the Event (if one doesn't exist).
- Add a number of sound instances to the layer. If there are gaps between the Sound Def instances these will play as periods of silence.
- For each sound instance, set the 'Loop mode' property to 'Oneshot'.
- Set a sustain point over each sound instance.
- Check the parameter property 'Keyoff on silence'.
- Set the parameter property 'Velocity' to a large value (such as 10000).

When auditioned, the event parameter should jump to the first sustain point and play the Sound Def. When the sound instance finishes, the parameter will jump to the next sustain point and play that sound instance, and so on until the max parameter value is hit, ending the event.

Going Further

Sequences combining both oneshot and looping Sound Defs can also be created. Just note that the sustain points over the looping Sound Defs must use a manual keyoff to continue with the sequence. Setting the sound instance properties, 'Start mode' to 'Wait for previous' and 'Loop mode' to 'Loop and play to end' can help maintain the timing of a sequence.

Random Loops using Spawning

Scenario

Have you ever needed a Sound Def that uses a different Sound File each time it loops? When using a sound instance set to loop, the standard behavior is to repeatedly play the current entry – the sound instance must be triggered again to select a new entry to play.

An alternative looping method is needed – hence this technique of creating random loops using spawning.

Spawning means the Sound Def triggers repeated playback of itself. Unlike looping, each repeat (or spawn) can overlap the previous sound and select a new Sound Def entry.

In FMOD Designer 2010, the frequency of spawning is specified as a range of time, using the Sound Def properties 'Minimum spawn time' and 'Maximum spawn time'. If these minimum and maximum values are different, then a random time between them is used. The number of simultaneous spawns is controlled by the property 'Max Polyphony'.

Solution

It is possible to create a loop that uses a new Sound File with each cycle, by taking advantage of the Sound Def's spawn behavior. Try the following:

- Add a sound instance to the layer. The Sound Def should contain multiple Sound Files or oscillations.
- Set the 'Loop mode' of each sound instance to 'Oneshot'.
- Double-click the sound instance to editor the Sound Def properties.
- Set the Sound Def property 'Spawn time' maximum value to 1 ms.
- Set the Sound Def property 'Max Polyphony' to 1ms.
- Set the Sound Def property 'Play mode' to 'Random' (this isn't too important – pick the one most appropriate to your situation).

The technique works by attempting to respawn the Sound Def each millisecond, however the 'Max Polyphony' property only allows a single instance of the Sound Def to play. When the sound instance completes playback, a new instances is spawned (in 1ms) and a new Sound File from the Sound Def is selected.

This technique is well suited to creating dynamic ambient environments – but could be used with larger musical elements.

Granular Synthesis

Scenario

There are many occasions where a looping sample is required - events such as an automatic machine gun, a car engine idling or background ambience. Unfortunately, if the loop is short in length the resulting sound may become repetitive or artificial, as the audience can hear the point where the sample 'loops'.

In this technique section, we will look at how static loops can be replaced by repeating small dynamic sequences – using granular synthesis. Granular synthesis is a method of generating sound using a collection of very small sound samples, commonly referred to as a 'grains'.

Solution

Dividing the source audio into grains

The first step requires the sound designer to cut the loop (or samples) into small 'grains' using their audio editor of choice. For example, if the audio loop is a machine firing 10 rounds, the file could be divided, with each round becoming a grain file. Within FMOD Designer 2010, grain sample can be as small as 35 ms in length.

The sound designer must be careful to make sure each grain starts and stops on a zero crossing. If any grain starts or stops on a value other than zero, an audible click may be introduced.

Setting up the Sound Def

Within FMOD Designer 2010, all the 'grain' samples are placed in a Sound Def. The following Sound Def properties must be set:

- Play mode: set to 'Random', 'RandomNoRepeat' or 'Shuffle'
- Spawn time: 1 (minimum) and 1 (maximum)

These settings tell the FMOD engine to attempt to select and play a random grain each millisecond.

Setting up the sound instance

Add the Sound Def to an event layer and set the following sound instance properties:

- 'Start mode' to 'Wait for previous'
- 'Loop mode' to 'Oneshot'.

These settings tell the FMOD engine join the current sample and the next sample together (in a seamless manner), and to not loop playback of the current sample.

Issues when stitching small samples

There is one last property that must be set - the Sound Def's 'Max Polyphony' property. The best value of this property is determined by how often the programmer calls '*eventsystem::update'* within the game title.

To provide granular synthesis without clicking artifacts, FMOD Designer 2010 needs to queue up (or sequence) enough grains (samples) to play between update calls – because if it runs out of audio data, a click may be heard.

The sound designer can choose the number of samples to queue by setting the 'Max Polyphony' property. For example, if the samples are 6 ms each, and the update is called every 15 ms, the 'Max Polyphony' property should be set to 3.

Note:
Additional memory is used for each additional grain that is queued via the 'Maximum spawned Sound' property. It is worth discussing the trade off between small granularity and memory usage with your programmer.

Selecting Sound Files at Runtime

Scenario

Choosing an audio file to play at runtime is a typical function required for dynamic dialogue (such as game commentary).

Solution

Choosing the audio Sound File to play at runtime in FMOD Ex can be done in two ways:

- Including a Programmer sound entry in a Sound Def or,
- Setting the Sound Def property 'Play mode' to 'ProgrammerSelected'

Each method has it's own advantages and disadvantages – and in some cases both methods could be used in same event (but more on that later).

Programmer Sounds

As described earlier in the user manual a Programmer sound entry can be added to a Sound Def by right-clicking the Sound Def (under the 'Sound Def' View) and selecting 'Add ‘Programmer’ sound' from the menu that appears.

When the sound instance (and programmer sound) is triggered, a callback is created and must by handled by the Programmer.

The sound designer does not have control over which audio is played. Instead, the Programmer supplies a 'sound' object within the source code. This technique is very memory efficient when dealing with thousands of files containing dialog audio. The sound designer can create a single event to handle every line of dialog!

This technique does break the data-driven model somewhat as the sound designer loses control, however the memory savings can be significant enough to warrant this choice.

Most sound designers find the biggest limitation of this technique is the inability to easily audition the event (dialog audio in this case) from within FMOD Designer 2010.

Notes for the programmer:

For information about the 'Programmer sound' callback, consult the FMOD Ex API Event documentation. The FMOD Ex Programmers API also includes a C source code example of this technique.

Relevant sections include:

- FMOD_EVENT_CALLBACKTYPE_SOUNDDEF_CREATE
- FMOD_EVENT_CALLBACKTYPE_SOUNDDEF_RELEASE
- Event::SetCallback

The 'ProgrammerSelected' Play Mode

FMOD Designer 2010 offers a second method for choosing a Sound File to play at runtime. This option is selected by first navigating to the appropriate Sound Def (under the 'Sound Def' View). Once the Sound Def is selected, the 'Play mode' property is set to 'ProgrammerSelected'.

During runtime whenever this Sound Def is triggered (or respawned) a callback is created and must be handled by the Programmer.

The Programmer supplies the index of the Sound File in the Sound Def to play. For example, an index value of 0 plays the first Sound File, 1 plays the second Sound File, etc.

Only Sound Files within the Sound Def can be played. The benefits of this technique are:

- It is much easier to audition the audio within FMOD Designer 2010.
- All resource management is handled by the event system.
- The sound designer retains more control over which audio can be played

Notes for the programmer:

For information about the 'ProgrammerSelected' mode and related callback, consult the FMOD Ex API Event documentation.

Relevant sections include:

- FMOD_EVENT_CALLBACKTYPE_SOUNDDEF_SELECTINDEX
- Event::SetCallback

Combining the Techniques

This technique can be combined with Programmer sound entries, resulting in two callbacks. The first callback would require the index for the Programmer sound entry within the Sound Def. The second callback would return the actual sound to play!

A Basic Multi-track Event for Multiple Platforms

Scenario

While working on games titles, sound designers are frequently required to develop content for multiple platforms – and in many cases, these platforms have significant differences in CPU power and memory capacity.

Consider the following scenario:

- An event is required for multiple platforms, in this case PC and Wii.
- Dynamic variation during playback is required, so we will specify that our event will use a random selection from a number of samples within the Sound Def.
- Each platform has different memory requirements. The PC will have 7 samples within the Sound Def, while the Wii will only have 3 samples.
- The event only requires basic playback, therefore we want to take advantage of 'simple event' status.

On first look, a sound designer might assume that two different events are needed. In the next section, a technique to create platform-specific basic multi-track events using a single event Def is discussed.

Solution

Before we jump straight into the solution, with need to understand what a ' basic multi-track event ' is. To recap (in part), an event is considered to be a ' basic multi-track event' if all the following criteria are met.

A basic multi-track event has:

- No parameters
- No effects
- No user properties
- Only one layer
- Only one sound (the Sound Def may have multiple Sound File entries)
- An event can have multiple layers defined, but still be classified as a ' basic multi-track event ' (by the build process) if only one layer is enabled for the platform.

As stated earlier in the manual, the benefit being that the FMOD Event System has an optimized code and data path for ' basic multi-track event ' that uses less memory and less CPU at runtime.

Now let's consider a solution. If a basic multi-track event can only have a single layer and one sound, how can we design a single event with platform specific sounds?

The key to this technique is the use of the 'disable layer' checkbox within FMOD Designer 2010. This checkbox is specified on a per-platform basis. The 'disable layer' checkbox is shown the figure below.

Figure 7-1: The 'disable layer' checkbox shown at the far left on the layer

An event can have multiple layers defined, but still be classified as a ' basic multi-track event ' (by the build process) if only one layer is enabled for the platform.

To construct the event for the scenario describe earlier:

1 Create a Sound Def (in this case called 'samples-pc'). This Sound Def will contain all seven samples for the PC platform.
2 Create another Sound Def (in this case called 'samples-wii'). This second Sound Def shall contain only three samples, and will be used for the Wii version of the event.
3 Create an event, and open it in the event editor View.
4 Add a second layer to the event.
5 Double click the 'layer00' layer label, a pop-up window should appear. Rename the layer 'PC'. This step isn't required, but will provide a quick visual indicator that this layer is for the PC platform only.
6 Double click the 'layer01' layer label, a pop-up window should appear. Rename the layer 'Wii'. Again, this step isn't required, but will provide a quick visual indicator that this layer is for the Wii platform.
7 Add an instance of the 'samples-pc' Sound Def to the layer labeled 'PC'.
8 Add an instance of the 'samples-wii' Sound Def to the layer labeled 'Wii'. The event should look like the figure below.

Figure 7-2: Setting up the event

At this point, we are ready to use the 'disable layer' checkbox to specify which layer is active for the appropriate platform.

9 Ensure that 'PC' is selected in the 'Platform' dropdown list.

Figure 7-3: 'PC' set as the target platform

10 With the PC platform selected, disable the Wii layer by clicking on the 'disable layer' checkbox (on the layer labeled Wii). The Wii layer will become grayed, to indicate it's disabled.

Figure 7-4: Only the PC layer is enabled

11 Now select 'Nintendo Wii' from the 'Platform' dropdown list (you'll notice that both layers are active).

Figure 7-5: Nintendo Wii selected as the target platform.

12. With the Wii platform selected, disable the PC layer by clicking on the 'disable layer' checkbox (on the layer labeled 'PC'). The PC layer will become grayed, to indicate it's disabled.

Figure 7-6: Only the Wii layer is enabled

Jumping between the PC and Wii platforms, you'll now notice that only one layer is ever active. Therefore at build time, the event will qualify as a ' basic multi-track event ' because it meets the criteria of 'one sound' and 'one layer'.

This technique improves workflow efficiency, allowing a sound designer to create platform-specific basic multi-track event from a single event definition.

CHAPTER EIGHT: The Profiler

8

The 'Profiler' within FMOD Designer 2010 allows a user to connect to an FMOD Ex application and monitor its resource usage.

Key topics in this chapter.

- Usage of the Profiler
- Connecting to an Application
- Interpreting data from the Profiler

Introduction to the Profiler

The FMOD Profiler allows a user to connect to an FMOD Ex application and monitor its resource usage. The profiler connects to the application in the same way as the network audition functionality.

In use, the Profiler can also be used to identify which DSP effects are active in the FMOD DSP network during the application's execution. It is possible to identify the complete signal path of a Sound File through to the soundcard. The profiler may also be used to monitor the overall CPU usage as well as the CPU usage of each DSP node. Finally the profiler can also monitor the codec pool and channel usage.

Note:

The Profiler tool will only identify processing in the software domain. Any processing through hardware channels, or OpenAL will not be shown.

Getting Started

Initialization of FMOD Ex (Programmer)

Before the Profiler can view the DSP network or monitor resource usage, the FMOD Ex engine (running as part of the target application) must be correctly initialized using the flag:

```
FMOD_INIT_ENABLE_PROFILE
```

For example, initializing FMOD Ex (using the low level API) to enable profiling, would look something like:

```
result = pSystem->init(32, FMOD_INIT_NORMAL | FMOD_INIT_ENABLE_PROFILE, 0);
```

Profiling Applications

Before using the profiler, please ensure:

- FMOD Ex (System or EventSystem) is initialized using the flag FMOD_INIT_ENABLE_PROFILE
- The target game (or application) is running, and it's IP address is noted

Starting the Profiler

To run the profiler, complete the following steps:

1. Click on the Profiler View inside FMOD Designer 2010 application.
2. In the 'IP text box, enter the IP address of the target game (or application). For example, if the game is running on the same machine as the Profiler tool, you would enter 127.0.0.1.
3. If the default port number is not being used, enter the port number into the 'Port' text box.
4. Press the 'Connect' button. The tool will attempt to connect to the target game.
5. When connected the DSP network of the target game will be displayed and the CPU usage of the DSP network will start monitoring.

Note:

The connection status message (found in the top left hand corner of the profiler View) may contain the warning 'Old DSPNet version detected'. This message is displayed when the profiler is using 'compatibility' mode – usually when the profiler version is connected to an earlier version of the FMOD Ex API.

When this message is displayed, some functionality of the profiler may not be available, as the client is no sending the required data

Understanding the DSP Network

Once the tool has connected to the FMOD Ex DSP network, it draws each node and the connections between them. On the next page is an image of a simple DSP network.

The output from the soundcard can be traced back to the input by reading from left to right. On the far left is the ‘soundcard’ unit. The soundcard then pulls data from the ‘FMOD ChannelGroup Target Unit’, which pulls data from a master group, reverb and channels, etc.

Active DSP network nodes are identified with a light-blue background. Inactive DSP network nodes, such as paused sounds or virtual voices are identified with a gray background. Nodes that are bypassed are displayed with a light gray (or silver) background.

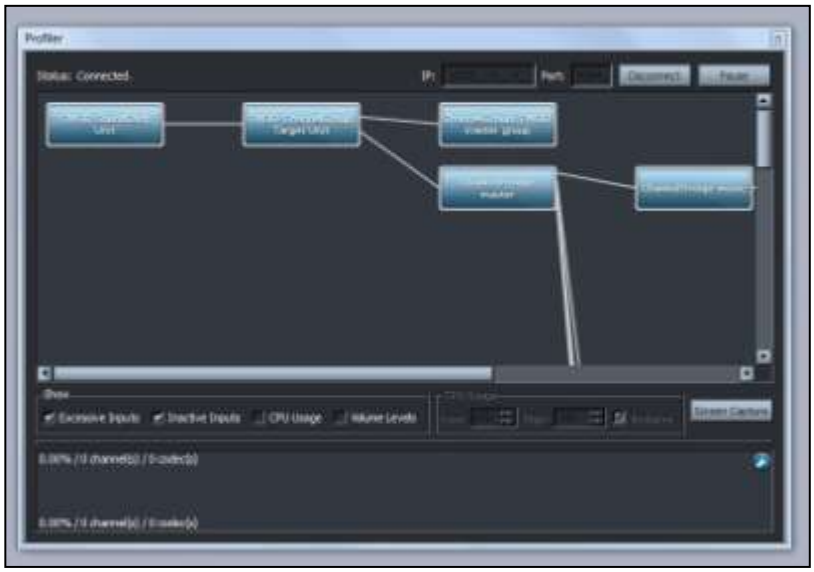


Figure 8-1: A DSP network

Monitoring CPU Usage per Node

By clicking the 'CPU Usage' checkbox in the 'Show' panel, the profiler will display the CPU usage of each node on the DSP network. Clicking the 'CPU Usage' checkbox will also enable the 'CPU Usage' panel for use.

When active, each node will fluctuate between the following colors:

- **Green** – This means the node is using a low amount of CPU time.
- **Yellow** – This means the node is using a moderate amount of CPU time.
- **Red** – This means the node is using a large amount of CPU time.

Rather than assign fixed CPU usage values as low, moderate and high, the profiler tool allows the sound designer to specify the color behavior using the 'CPU Usage' panel.

- The 'Low' field value represents the CPU usage at which the node remains as green.
- The 'High' field value represents the CPU usage time at which the node flagged as red.

Any CPU usage between the 'Low' and 'High' values will produce a gradient value between green and red (generally a tone of yellow). This feature effectively allows the sound designer to identify a node using a large amount of CPU within the DSP network.

Figure 8-2: Monitoring CPU usage per node

Inclusive Mode

The 'Inclusive' mode makes any CPU intensive nodes more visible to the sound designer, by propagating a visually effect to nodes further along the DSP network. This allows the sound designer to trace a path from right to left to find the offending node.

Within the 'CPU Usage' panel there is a checkbox labelled 'Inclusive'. When the 'Inclusive' mode is enabled, CPU usage values for each node are summed with the node (or nodes) connected to the left.

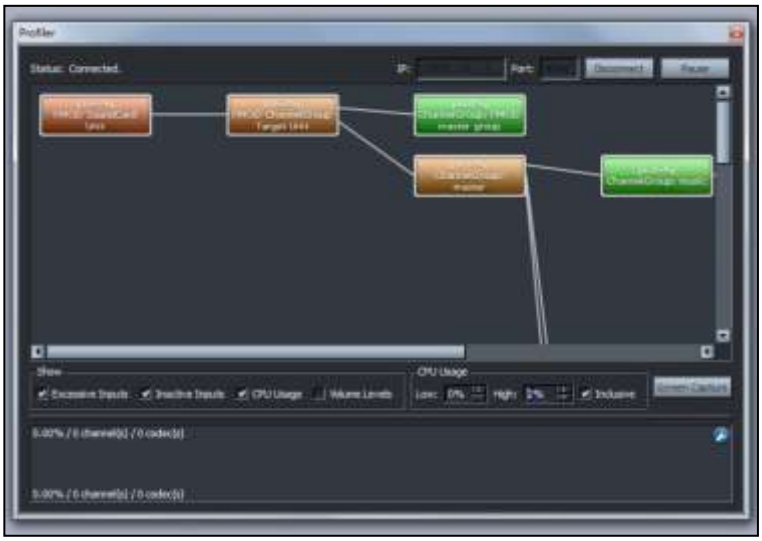


Figure 8-3: Effect of the Inclusive mode

Other DSP Network Features

Saving

To save the current state of the DSP network, the tool can create a screenshot. The 'Save' button, produces a .PNG image of the complete DSP network (and is not affected by the current zoom level).

To use this feature:

1. Press the 'Screen Capture' button. A pop-up window should appear.
2. Enter the desired filename and location of the image file.
3. Press 'OK'. The image file will be created.

Zooming

The Profiler allows the sound designer to inspect nodes in closer detail or view all nodes at once, using the zooming function. The feature requires a mouse with a scroll wheel facility.

To zoom in (for close viewing) roll the scroll wheel up. The zoom out (for distant viewing) roll the scroll wheel down.

Hiding Inputs

The checkbox labeled 'Excessive Inputs' is useful when dealing with complex DSP networks which contain many nodes. When deselected, the number of child nodes displayed under any node will be limited.

The checkbox labeled 'Hide Inactive Inputs' allows any inputs not is use to be ignored, and not displayed in the DSP network tree.

Monitoring Volume per Node

The profiler allows users to monitor the signal volume at each node. The volume monitor is intended as a diagnostic tool to find issues within the DSP network and indicate clipping where possible.

To enable this feature, click the 'Volume Levels' checkbox. Enabling volume monitoring causes a peak meter for each channel in the signal path (above the node) to be displayed. The Profiler queries all nodes each FMOD update cycle and determines the signal's peak value for each channel. Using this data, the volume meter displays the following colors:

- **Green** – This means the sample value is under maximum. The height of the green bar indicates the amount of headroom remaining.
- **Red** – This means the signal has clipped maximum output.

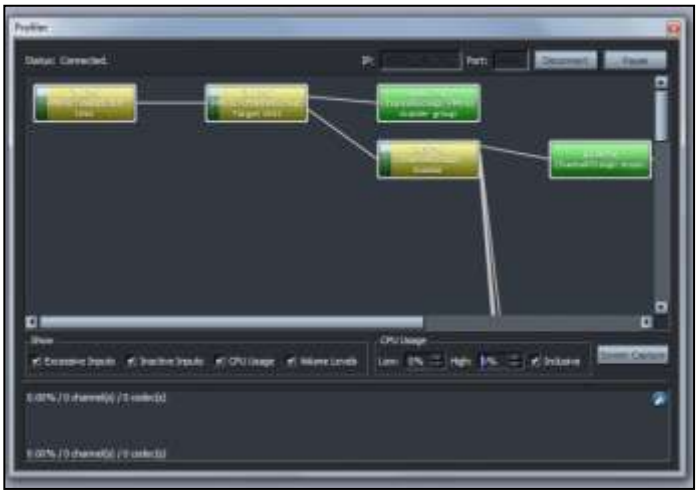


Figure 8-4: Enabling volume level monitoring

Usage Monitoring

The profiler tool allows users to monitor CPU, mixer channel and codec pool usage. This display is located below the DSP network panel.

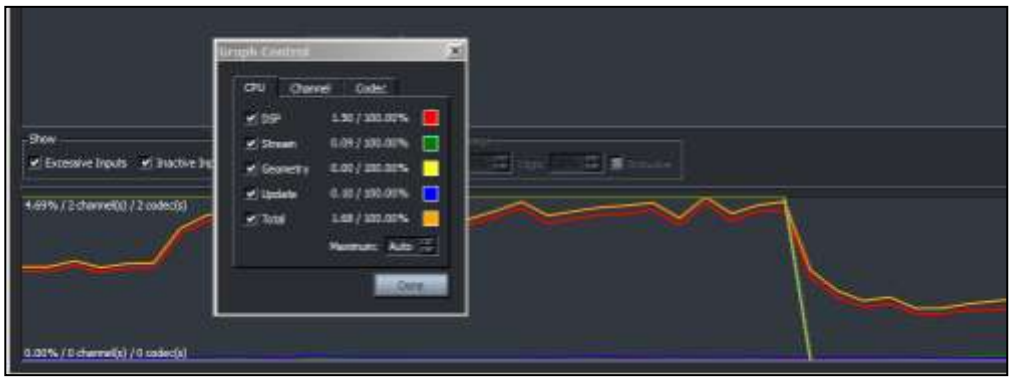


Figure 8-5: Resource monitoring

Selecting an Item to Monitor

To select the resource to monitor:

1. Press the button on the graph panel. This will cause the 'graph control' dialog to appear.
2. Navigate through the three Views and click on the checkboxes to be displayed in the graph. Each item is explained in the section below.
3. To stop the monitoring of an item, simply click the item checkbox again.

CPU Monitoring

Within this panel, the user can monitor the CPU usage in real time. The following data can be displayed:

- DSP
  - Identifies the CPU usage on the DSP network. This includes mixing and DSP effects.
- Stream
  - Identifies the CPU usage of the streaming engine. This CPU overhead is incurred whenever sounds are streamed from disk.
- Geometry
  - Identifies the CPU usage of the Geometry API thread.
- Update
  - Identifies the CPU usage of the processing undertaken in the **System::update(..)** or **EventSystem::update(..)** method of the source code. This includes calculations for 3D behavior, occlusion and envelopes.
- Total

- The total amount of CPU usage.

Note:

The total percentage may exceed 100% in some circumstances, such as systems that feature multiple CPUs and when the mixer thread does not receive adequate resources.

Mixer Channel Monitoring

Within this panel, the user can monitor the mixer channel usage in real time. The following data can be displayed:

- Hardware
  - Displays the number of hardware channels being mixed by the soundcard hardware.
- Software
  - Displays the number of software channels being mixed by the software mixer.
- Real
  - Identifies the number of channels in playback (hardware + software channels).
- Virtual
  - Identifies the number of channels that are active, but currently inaudible (virtual).
- Playing
  - Displays the total number of channels being mixed. This includes all active sounds that have a real channel, and those which are virtual.

Codec Pool Usage

Within this panel, the user can monitor the codec pool usage in real time. The following codecs can be displayed:

- MPEG
  - Displays the number of MPEG codecs currently being played.
- ADPCM
  - Displays the number of ADPCM codecs currently being played
- XMA
  - Displays the number of XMA codecs (Xbox 360 only) currently being played.
- PCM
  - Displays the number of PCM codecs (PS3 only) currently being played.

Choosing the Range to Display

By default, the profiler will maximize the display scale by dynamically adjusting the ‘Maximum’ value to highest value measured by the profiler.

The display scale can be fixed to a selected value by:

1 Pressing the button on the graph panel. This will cause the 'graph control' dialog to appear.
2 Navigating to the appropriate View (CPU, Channel or Codec).
3 Entering the desired range maximum into the 'Maximum: ' textbox field

The display range will be indicated by the values in the bottom left and top left corners of the graph panel.

CHAPTER NINE: Interactive Music

9

The 'Interactive Music' View within FMOD Designer 2010 allows the sound designer to create a game soundtrack that features cues, musical themes, transitions between music themes and flourishes.

All musical transitions and flourishes can be instructed to playback in a musically intelligent and beat-matched manner. Each theme can be attached to cues and transitions between themes are easily parameterized for the Programmer.

Key topics in this chapter.

- Introducing key concepts
- Explaining how the music system works
- Usage of the Interactive Music View

Introduction

The Interactive Music View allows a sound designer to construct musical movements and link these movements with seamless transitions. Transitions between themes are not the only feature of the Interactive Music system. The sound designer can also embellish these movements with additional concurrent flourishes. To preserve musicality, these flourishes can be synchronized to any beat of the existing movement, as selected by the sound designer.

The combination of seamless transitioning and flourishes provides useful musical variation, and allows for non-linear sequencing and composition that is easily connected to game state.

The Interactive Music View is one component of a complete music system, the other component being the FMOD Event System API. Within this chapter, the Interactive Music View refers to the tool used by the sound designer. The Interactive Music System (or music system) refers to the complete system (tool and engine combined).

This chapter has the following structure:

- Key terms and objects. A brief description and explanation of the common components of the music system
- Learning the Interface. A look at all the main dialog and property panes used when creating a score within the Interactive Music View.
- System Implementation Concepts. This section discusses the inner workings of the music system.
- Creating an Interactive Music Score. This section provides the practical 'how-to' information for creating a adaptive music piece within the Interactive Music View.
- Auditioning. This section discusses methods for previewing a work in progress or finished adaptive music piece.
- Property Reference Guide. This section details all properties and dialog options within the Interactive Music View.

Key Terms and Objects

In this section, the key terms and objects used in the music system are explained.

Cue

From the sound designer's perspective, a cue is a potential musical trigger. The sound designer can associate a cue with a particular piece of music. In the user interface (scene editor) a cue is represented with the icon below:

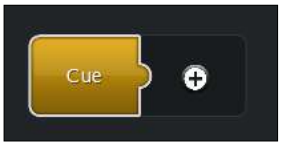


Figure 9-1: The 'Cue' icon

Cues are also displayed in a cue list. The figure below demonstrates a cue icon in the cue list.

Figure 9-2: A cue icon called 'retro-rock_arena' in the cue list

Parameter

Parameters are values passed from the game engine to the music system. Using these values allows the sound designer to specify how the music piece responds to changes in game state. For example, the value of a particular parameter may dictate the choice of one musical phrase over another.

Within the Interactive Music View, the following icon represents a 'Parameter' in the parameter list:

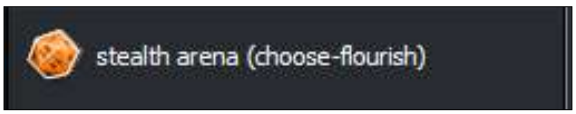


Figure 9-3: The 'Parameter' icon

Scene

For the sound designer, a scene is simply a cue sheet. It allows the sound designer to attach a cue to a theme. There is only one scene per project.

Theme

A theme is comprised of a number of musical segments and the links between them. Themes are associated with cues - themes specify the music played when the associated cue is triggered.

A theme is represented by a number of icons throughout the Interactive Music View. Within the theme editor, the theme icon is shown in the figure below.

Figure 9-4: The 'Theme' icon from the Theme Editor pane

When attaching a theme to a cue object, the theme icon shown in the figure below may be used.

Figure 9-5: A theme icon in the theme list

The figure below shows a theme attached to a cue within the Scene Editor pane.

Figure 9-6: The 'Theme' attached to a 'Cue' in the Scene Editor pane

Segment

The individual musical elements used to form sequences are called segments. Generally, each segment will contain one audio file of one or more bars of music. A segment may be also used to hold a number of samples, known as a 'playlist'. A sample from this playlist is chosen for playback in a sequenced or random fashion. This is useful for adding variation to a segment without the need for additional linking logic. A segment is shown in the figure below.

Figure 9-7: The 'Segment' icon

For each segment, the sound designer specifies a BPM (Beats Per Minute) and Time signature.

With the BPM and time signature set, a powerful feature of the music system is enabled. Within each segment, a step sequencer style widget is provided. Using this step sequencer, the sound designer can specify which 'beats' within the bar act as a synchronization point for any concurrent theme playback. The step sequencer is shown on page 203.

Links

The sequencing of segments is achieved using 'links'. A simple linear sequence of segments is achieved by creating one link between each segment. Non-linear sequences are achieved by created by linking one segment to multiple segments, or linking from multiple segments to one segment.

By default, links are followed once the audio file within the segment has finished playing. Links can also be evaluated at the start on each bar, or on selected beats. This behavior can be assigned at the segment level, or to the individual links within a segment.

The following diagram shows two segments and a link between them:

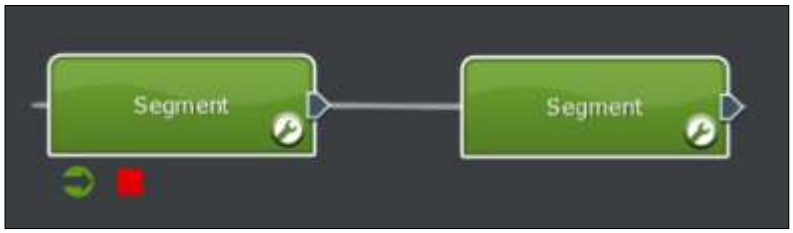


Figure 9-8: Linking segments

The sound designer can control the movement from segment to segment by attaching conditions and prioritizing links. For example, the music system may choose between two musical segments based on the 'health' of the in-game character. The sound designer could add intensity and darken the mood of the score when the in-game character is in mortal danger. The sound designer must prioritize links to specify the desired behavior when more than one valid path exists between segments.

Specialized links defining transitions between themes can also be specified. These links (shown in blue) contain a special condition that occurs when the music system attempts to transition from one theme to another.

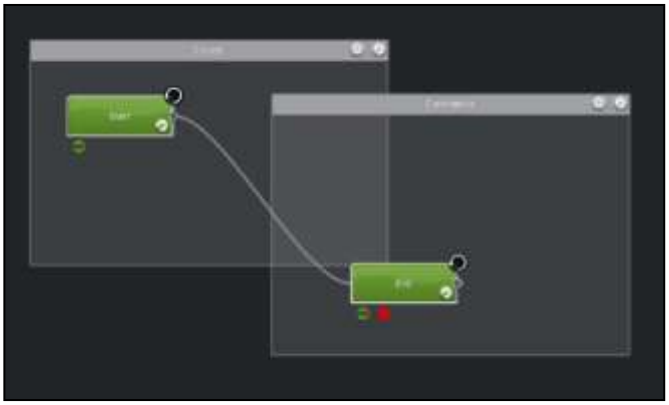


Figure 9-9: Creating a path between themes

Conditions

Conditions provide the mechanism to connect the behavior of the music score with game state data. Conditions can be used when evaluating:

- The link is followed.
- The 'start segment' to start a theme with.

Conditions are generally used with parameters. A condition tests the parameter data using logic operations such as 'greater than', 'less than', 'between', etc to provide conditional behavior. For example, the figure below shows a conditional link that is only followed when the parameter 'retro-rock-intensity' is between '3.30' and '6.60'.

Figure 9-10: Add conditions to a link

Conditions can also use 'cues' to provide logic switches. In this case, the link (or starting segment) will only be available when the conditional cue is active.

Shared Timeline

A 'shared timeline' is an object that allows the music system to distribute synchronization information between segments. Using shared timelines the sound designer can synchronize playback of multiple segments along a single timeline.

In use, a shared timeline makes the playback of segments behave as though the segments were started at the same time – with the music system intelligently muting, unmuting or cross fading volumes of the segments where appropriate.

This technique is useful when overlaying a flourish over a chord progression (within the audio of the segment already playing). In this situation, a shared timeline can be used to offset the starting point of the flourish audio. Rather than hear the flourish from the beginning of the audio file, the flourish will commence playback from the relative position of the chord progression – thus making easier for the sound designer to maintain harmonic structure.

For more information, see the Shared Timelines section on page 225.

Learning the Interface

In this section, the various screens and pop-up windows used in the Interactive Music View are listed and their role briefly explained.

Music Volume and Reverb

The 'Music Volume & Reverb' panel contains volume controls for the direct and reverb signals. This panel is located on the left of the Designer pane under the 'Cues', 'Themes' and 'Parameters' Views. The panel can be collapsed or expanded by clicking on the ► (or ▼) icon and is shown (in its expanded form) in the figure below.

Figure 9-11: Controls for direct and reverb volumes

The 'Volume' slider allows the composer to set the overall volume of the content within the music View.

The 'Reverb' slider allows the composer to specify how much signal is sent to the reverb effect. At 0dB, the available direct signal is also sent to the reverb effect. At -60dB, direct signal that is to be sent to the reverb effect is completely attenuated.

Note:

The reverb send is 'post fader' meaning changes to the direct signal will affect the volume of the reverb. This means that the reverb signal will always be proportional to the direct signal.

Scene Editor

The Scene Editor pane (shown in the figure below) allows the sound designer to associate cues to themes.

Figure 9-12: The Scene Editor

Theme Editor

The Theme Editor pane (shown in the figure below) provides the major design functionality need to create an interactive music piece. The Theme Editor contains palette tools, zoom control and audition console.

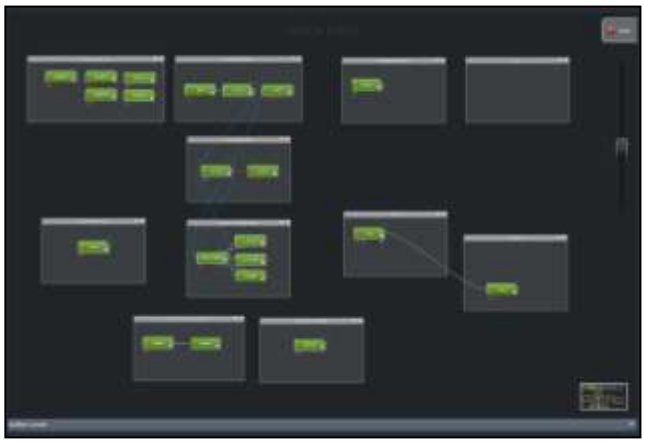


Figure 9-13: The Theme Editor

This Theme Editor pane allows the sound designer to:

- Create and link segments
- Set properties of segments
- Use a palette of tools for segment creation and static auditioning
- Access the Audition Console for dynamic auditioning

The Function Palette

Whilst the view is switched to the Theme Editor pane, a palette of tools becomes visible. The buttons available in the palette are:

The **SELECT** tool

The **MULTI-SELECT** tool

The **AUDITION** tool

The **SEGEMENT STAMP** tool

The Select Tool

This tool is used in the Theme Editor to move segments and themes around the editor space. This tool is selected as default when the Theme Editor is opened for the first time.

The Multi-Selection Tool

This tool is used in the Theme Editor to select a sequence of segments. Once the sound designer has created their desired sequence, the audition tool can be used to hear the selected segments played in sequence.

The Audition Tool

This tool can be used to audition an individual segment or audition a group of segments (sequenced using the Multi-Selection tool). See page 229 for more information.

The Segment Stamp Tool

This tool can be used to stamp new segments onto a theme.

Segment Editor

The Segment Editor pane allows the sound designer to set the properties of a segment. This window provides access to dialogs such the Beat-Sync Step Sequencer, Link Editor, Playlist Editor and Condition Editor. Using these dialogs, the sound designer may:

- Set basic properties of a segment such as name.
- Maintain a playlist (if more than one audio file is included in the segment).
- Specify which beats within the segment are synchronization points.
- Attached a shared timeline object.
- Order the priority of links leaving the segment.
- Apply conditions to links leaving the segment.

The Playlist Editor

A segment may contain zero or many audio files. In the case where a segment has more than one audio file, the playlist editor is used to maintain the list of audio files included in the segment. The playlist editor is also used to specify how a playlist item is selected at runtime. The playmode options include, 'Sequential', 'Random', 'Random no repeat' and 'Shuffle'. Each option is described in the properties section, found on page 236.

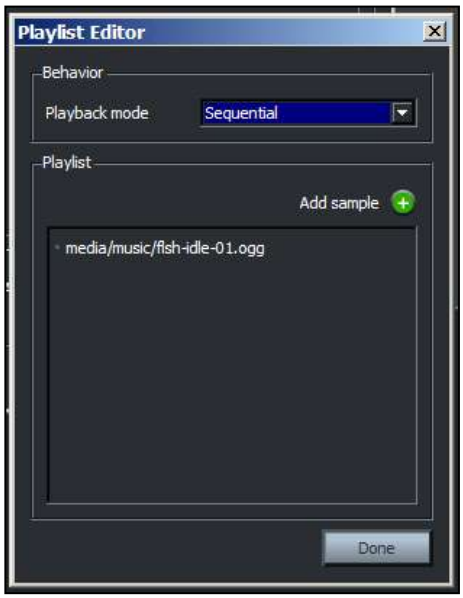


Figure 9-14: The playlist editor

Beat-Sync Step Sequencer

For each segment, the time signature and tempo (BPM) must be specified. Once entered, the Beat-Sync Step Sequencer can be used to specify which beats within each bar of the segment are synchronization points. Each button in the step sequencer represents a beat (of a bar) and when activated, specifies that concurrent themes may synchronize to that beat.

This control is shown in the figure below.

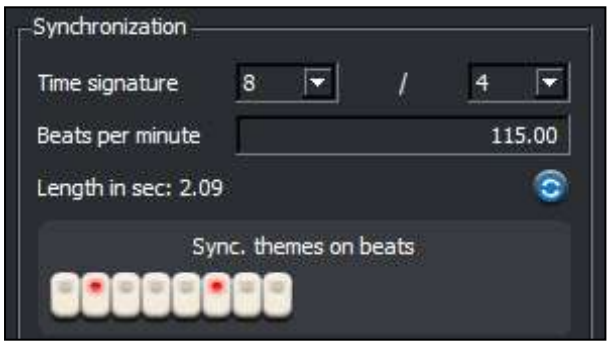


Figure 9-15: The step sequencer-style widget

Shared Timeline Interfaces

A shared timeline object can be attached to any segment by dragging an item from the 'timelines' list into the area shown in the figure below:

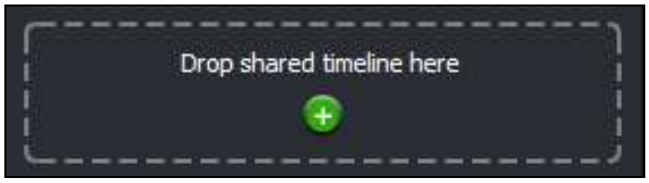


Figure 9-16: Attaching a shared timeline

The shared timeline pop-up window is shown in the figure below:

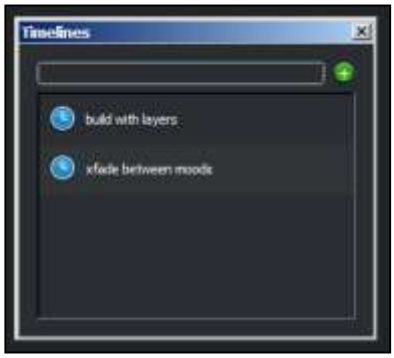


Figure 9-17: The Timelines Pop-up window

Link Editor

The Link Editor dialog allows the sound designer to specify the priority of links leaving the segment. The Link Editor also displays any condition currently attached to the link. The Link Editor may also be used to add a link back to the current segment (loop), as well as delete any link within the list.

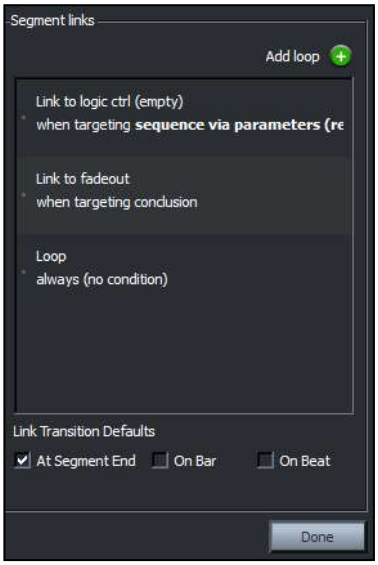


Figure 9-18: The Link Editor

Condition Editor

The Condition Editor dialog becomes available when editing a segment or theme's properties by double clicking on an item in the link list or start segment list. When editing a segment's properties, the Condition Editor allows the sound designer to apply cue or parameter conditions to links leaving the segment. Within the context of editing a theme's properties, the Condition Editor can be used to apply conditions to the theme's starting segments.

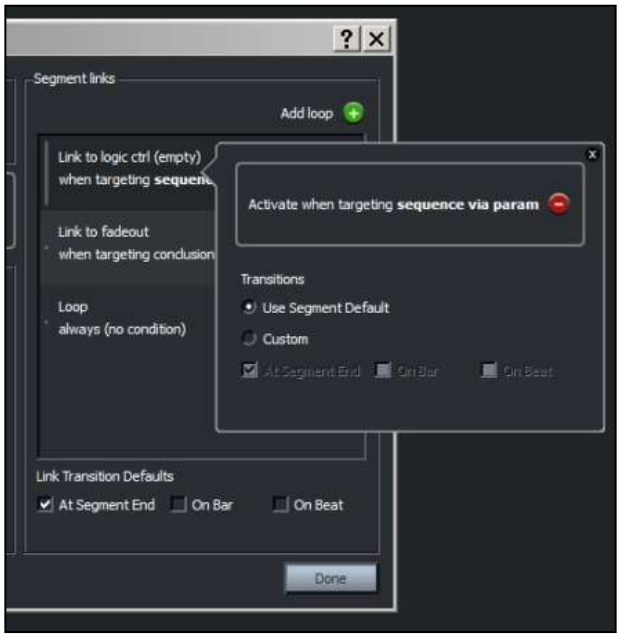


Figure 9-19: The Condition Editor (via Segment Links)

The condition editor also allows the sound designer to select when the transition to links are available. Selecting the option, 'Use Segment Default' means the link will defer to the option selected by the segment properties. Selecting the 'Custom' option allows an individual transition behavior to be specified for the link. The sound designer can specify that a link is evaluated once playback of the segment is complete, at the start of each bar, and/or on beats selected by the step sequencer.

Note:

The transition properties are only available in the Condition Editor when editing Segment Links, and not Theme Properties.

Audition Console

The Audition Console is initially located within the Theme Editor, but may be released from its dock to its own window. The Audition Console allows sound designers to dynamically audition the current interactive music score that is being created. The Audition Console provides much more

sophisticated auditioning than the static auditioning enabled by the function palette tools. The Audition Console allows non-linear sequences to be evaluated and previewed in the same way the game engine will interact with the Interactive Music System.

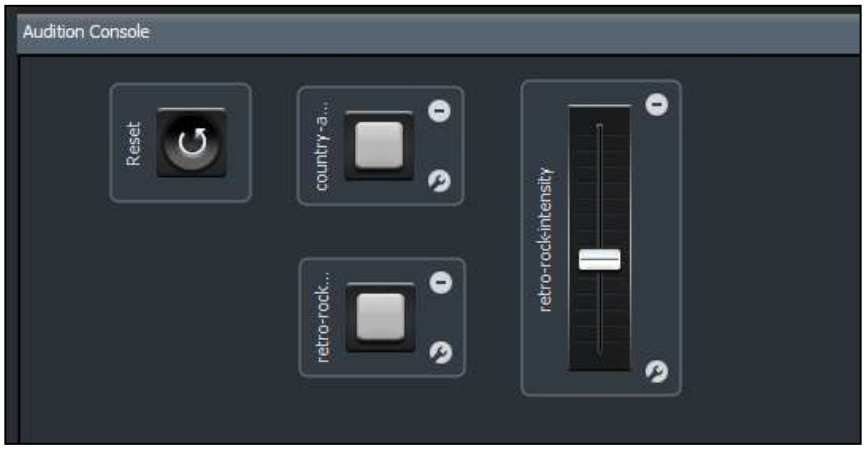


Figure 9-20: The Audition Console with controls

Icons at a Glance

In the table, the basic functions of the icons used in the Interactive Music View are described.

Icon	Name	Action
⊕	Add	Add the item.
⊖	Delete	Delete the item.
⊗	Close	Close the item.
	Edit	Edit this item.
	Include	Include the item.
	Exclude	Exclude the item.
	Centre	Opens the theme editor pane.
	Start Icon	Set the segment as a valid 'START' destination.
■	Stop Icon	Set the segment as a valid 'STOP' destination.
	Target	Target the current theme or segment.
	Recalculate	Recalculate the length of the audio file.
▶	Play	Start the audition playback.
⏸	Pause	Pause the audition playback.
	Release / Eject	Releases the dialog object from its dock.

System Implementation Concepts

In order to better understand how the Interactive Music System chooses the sequence of segments that are ultimately played, the inner workings of the system will be described. The following sections are not intended to be in-depth or highly technical - instead, it is hoped a sound designer can use this section as a reference to predict the run-time behavior of the music system.

Lifespan of Cues

Each cue has a lifetime. Once triggered, a cue is active until it is ended. This means that multiple cues may be active at any given time. The sound designer has freedom to choose how the system will respond to the multiple cues.

The sound designer might choose to create a transition segment between themes to provide allow a transition (for a sequence of themes). Or perhaps for more abstract compositions, it is quite possible to play themes concurrently - one musical piece on top of another.

Theme Stack and System States

Each time a cue (that is associated with a theme) is received from the game engine, the music system must react to the request whilst managing any previous request. The mechanism used to do this is called the 'theme stack'.

The music system uses this theme stack to manage transitions between themes using the 'Sequenced' playback method. There is one global theme stack for all sequenced themes in the music system.

The theme stack changes whenever a cue (associated with a sequenced theme) begins or ends:

- When a cue-theme association begins, a reference to it is added to the top of the stack
- When a cue-theme association ends, the appropriate reference is removed from the stack (regardless of its position in the stack)

The behavior of the music system is in part, dependent on the state of the theme stack. At any time, the music system is:

- Silent
  In this state the Interactive Music System is waiting for a new theme to be cued (the theme stack is empty).

- Determining which segment of a theme will start playback
  To enter this state, a cue has activated and the associated theme is placed on top of the

previously empty theme stack. To start playback of the theme, the music system must determine which 'start segment' is valid.

- In active playback
  Once the start segment has been determined, the music system will remain in this state whilst the cue remains active and its associated theme remains atop the theme stack.

- Transitioning from one theme to another theme
  This state occurs in two ways. The first occurs when a new cue is activated and a new theme is placed on top of the theme stack. The second scenario occurs when the cue associated with the theme currently on top of the theme stack is ended. In both cases, the music system will attempt to follow links 'targeting' the theme that is now on top of the theme stack.

- Attempting to silence playback
  When all cues are inactive, the theme stack will be emptied. In this scenario, the music system attempts to find a stop segment by following links that are targeting a conclusion.

Start Segments

Start segments are only evaluated when the theme stack is empty (no cues are active) and a cue with an associated theme is triggered. The music system then attempts to shift from its silent state to active playback of a theme,

To enter its active playback state, the music system looks at the theme's start segment list in order of priority to determine where to start. Playback is started at the first start segment in the list that satisfies its attached condition (if any).

If a theme has no active start segments when it is triggered, **nothing will play**.

A green start icon within the segment tray indicates that a segment has been nominated as a 'start segment'. A theme can have multiple start segments, each controlled by a condition. The theme properties dialog can be used to edit start segment conditions and priority (indicated by the order of start segments in the list).

Figure 9-21: A start segment is indicated by the green icon below the segment

Possible uses for multiple start segments include:

- Adding all flourishes to a generic concurrent (flourish) theme. Then selecting the appropriate flourish using a parameter and condition.
- Selecting different entry sequences into a theme based on a parameter representing 'intensity', 'mission outcome' or another mood characteristic.

To implement a Start Segment, see page 217 of the user manual.

Stop Segments

A red stop icon below the segment indicates a 'stop segment'. A theme can have multiple stop segments, all of which are always active. A stop segment indicates (to the music system) that it is musically acceptable for playback to be stopped once the stop segment has finished.

Stop segments only come into effect when the theme stack is in the conclusion state (no cues are active). In this case, if the current segment is a stop segment, no links are followed when the segment ends, causing playback to end.

Another feature of stop segments is that target conditions can be created. These specialized links (shown in red) are followed (to a stop segment) whilst the music system is in the conclusion state.

Figure 9-22: A Stop Segment

To implement a Stop Segment, see page 218 of the user manual.

Target Conditions

Targeting a Theme

The 'target theme' is the theme currently at the top of the theme stack. Once a new theme is placed on top of the theme stack, the music system will attempt to transition from the current segment to the new 'target' theme.

The sound designer guides this transition between themes using specialized links that include a 'Target theme' condition or by specifying Default Transition behavior. Links with the target condition are identifiable as blue in color.

For instructions on Targeting a Theme, see page 222 of the user manual.

Targeting a Conclusion

The conclusion state occurs when there is no target theme, i.e. when there are no active cues and the theme stack is empty. When the conclusion state is active, the music system will attempt to stop playback in a musically acceptable way. The sound designer guides the conclusion logic by creating a stop segment, targeting the stop segment and creating links with 'Targeting Conclusion' conditions. These links (shown in red) specify a pathway to the stop segments.

For instructions on Targeting Conclusion, see page 218 of the user manual.

Flourishes (Concurrent Themes)

Not all themes are designed for sequenced playback. Concurrent themes are intended to overlay an existing musical piece. The playback of a concurrent theme can be beat matched to a segment or started immediately (free from synchronization).

While originally intended to provide short musical 'stabs' to add dynamic flourishes to music pieces, there is no reason why concurrent themes cannot be used for more abstract music pieces. Concurrent themes also use start segments, stop segments and links between segments - just like a sequenced theme.

Creating an Interactive Music Score

Creating and Maintaining Cues

The 'Cue List' View allows the sound designer to create a cue and place it in the scene. To add a cue, simply type in the cue name into text field and press the ⊕ button. The new cue will appear in the Cues list, as shown by the 'Open' cue in the figure below.

The text field above the cue list also operates as a search filter. Typing in the text field will limit the cues that appear in the cue list. This is useful for finding cues in projects with a lot of data by removing the need to scroll through a large list.

Figure 9-23: Adding a cue

To place a cue into the scene, press the ⊘ button. This will cause a new cue icon to appear in the scene editor pane. This is shown in figure below.

Figure 9-24: Cues in the Scene editor

To rename a cue, double-click item the cue item in the cue list. A pop-up window will appear. Simply enter the new name and press the ⊗ (or press the [return] key). The new item name will be displayed.

A cue can be removed from the scene by selecting the appropriate cue from the list, and pressing the ⦸ button. The cue can be deleted entirely by selecting the appropriate cue from the list and pressing the ⊖ button.

Creating and Maintaining Themes

Themes can be created in two ways:

Method One: Pressing the ⊕ button on the empty half of the cue icon. Adding a theme in this way will:

1. Create a blank theme.
2. Add the new theme to the Theme list.
3. Open the Theme list View and finally.
4. Attach the new theme to the selected cue.

Method Two: The Theme list View panel allows the sound designer to create a theme and place it on the scene. To add a theme, type in the theme name into text field and press the ⊕ button. The new theme will appear in the Theme list.

The Theme list provides a number of functions. With the Scene editor open, themes can be attached to cues by dragging the theme icon (the grey item in the Theme List) and dropping it onto the cue icon. The cue icon will display a thick border when the theme can be dropped onto the cue.

To rename a theme, double-click item the theme item in the theme list. A pop-up window will appear. Simply enter the new name and press the ⊗ (or press the [return] key). The new item name will be displayed.

Switching to the Theme Editor Pane

The Theme Editor pane allows the sound designer to add segments and create links. To enter the Theme Editor either a) pressing the ⊙ button for a theme selected from the Theme list or b) Press the ⊙ located on the cue-theme association icon, as shown in the figure below.

Figure 9-25: The 'New Theme' edit icon

Creating Flourishes (Concurrent Themes)

As stated previously in the user manual, flourishes are sounds or musical phrases that are used to accent or provide variation to an existing theme that is playing. The audio within the flourish is played concurrently with the existing theme and can be played immediately or synchronized to the start of any bar or beat within the existing theme.

A flourish is just a theme with some specific properties. To create a flourish:

1. Create a new theme in the Theme list View.
2. Press the ⊙ button to open the theme in the Theme Editor pane
3. Press the theme's 🔧 button. This button can be found in the top right hand corner of the theme. A new property window should appear.
4. Set the theme to play concurrently with an existing theme by setting the **Playback Method** property to **Concurrent**.
5. Set the Quantization property as desired. [See the section Theme Properties on page 235 for more information]
6. Attach the concurrent theme to a cue.
7. Make sure any segments inside this theme are set as starting segments (see page 217). If using multiple segments within the theme, conditions and priority must be set. This is required for the music system to know which segment within the concurrent theme to play when the flourish is triggered.

When the cue is triggered in game, the segment within this theme will play concurrently with the existing theme.

Creating and Maintaining Parameters

The Parameter list View panel allows the sound designer to create a parameter and place it on the scene. To add a parameter, type in the theme name into text field and press the ⊕ button. The new parameter will appear in the Parameter list. This is shown in the figure below:

Figure 9-26: Adding a parameter

A parameter can be deleted from the list by selecting the appropriate entry and pressing the ⊖ button.

The parameter icon (shown above as an orange widget) can be dragged and dropped in the 'Condition editor'. This is method is described in greater detail, in the 'Creating and maintaining links' section on page 220.

Creating and Maintaining Segments

There are two methods of adding a segment to a theme:

- Using the 'Segment Stamp' tool. When selected from the function palette, this tool allows the sound designer to add new segments by clicking on the theme.
- Drag and drop. Dragging an audio file from the operating system window and dropping it on a theme (in the Theme Editor) automatically creates a segments.

To modify the properties of a segment, click the 🔧 icon of the segment to change. A segment properties pane should appear. This dialogue pane is shown in the figure below:

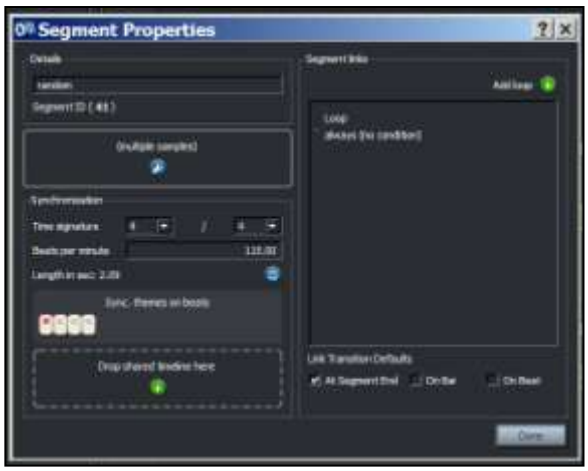


Figure 9-27: The segment property dialogue

The text name or descriptor of the segment can also be added into the text field under the 'Details' label in the segment property pane. This name field is provided for readability purposes, sound designers need not worry about using unique names - FMOD Designer 2010 uses an internal identifier for each segment.

Specifying the Audio in a Segment

As stated previously in the manual, each segment may contain zero, one or many audio files. A segment properties dialog provides access an additional 'playlist' editor when the segment contains multiple audio files.

1 To add (or replace) the audio file in a segment, click the icon of the segment to change. A segment properties pane should appear.
2 An audio file (or multiple files) can be added to the segment by clicking the ⊕ button, and selecting the audio file(s) from the file selection dialogue. The audio file(s) can also be dragged from an operating system window and dropped into the panel 'Drop a track here'.

Sorting the Segment Playlist
The audio files within a segment can be ordered by:

1 Click the icon of the desired segment. A segment properties pane should appear.
2 Click the icon on the audio track panel. The playlist editor should appear.
3 Drag the filename of the audio to move to its desired position in the list.
4 Press the 'Done' button.

Deleting the Audio from a Segment
To delete the audio in segment containing one audio file:

1 Click the icon of the desired segment. A segment properties pane should appear.
2 Press the ⊖ button from the audio track panel (shown in the figure below). The segment no longer contains the audio file.

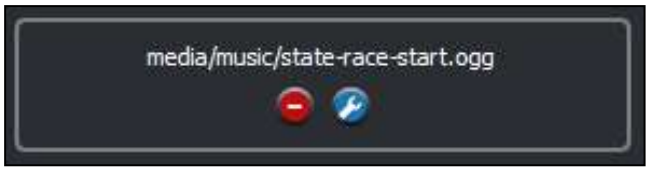


Figure 9-28: The audio track panel

To delete the audio in segment containing multiple audio files:

1 Click the icon of the desired segment. A segment properties pane should appear.
2 Click the icon on the audio track panel. The playlist editor should appear.
3 Select the audio file to remove from the playlist. A ⊗ button should appear.
4 Press the ⊗ button to remove the audio item from the playlist (and segment).

Repeat steps 3 and 4 until all the undesired audio is removed.
5 Press the 'Done' button.

Creating Start Segments

When a cue is activated to start a theme (from silence), the sound designer must specify which segment to start on. This is the role of a 'start segment'. A theme can have multiple start segments. The music engine will determine which segment to start on using priorities and/or conditions.

As previously stated, the starting segment is only evaluated when a cue is triggered and no other cues are currently active.

To specify a segment as a potential start point:

1. Move the mouse pointer over the segment. A tray of icons should appear beneath the segment.
2. Click the ➲ icon. The icon should now appear green and remain in view once the segment tray closes.

To remove the start segment status, simply re-click the ➲ icon.

Start Priorities and Conditions

To specify the priority of 'Start' segments:

1. Open the theme's property pane by clicking the ✇ button located on the theme (whilst in the Theme Editor pane). A dialogue should appear.
2. Order the items in the 'Start Segments' list, where the first position in the list represents the highest priority. To do this, simply click and drag the list item to the desired position within the list (a line of dots will indicate the new position).

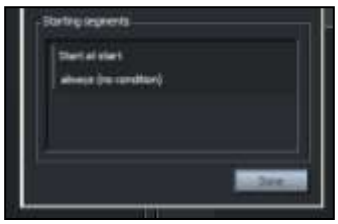


Figure 9-29: The 'Start Segments' list

3. Additional cue, parameter or theme conditions can be applied to the start segment by pressing the ✇ button located on the list items. A 'Condition Editor' window dialogue should appear.
4. Drag a cue or parameter icon from its respective list and drop it on the empty Condition Editor. If using a parameter, additional condition parameters will become available.

During run-time, the engine will evaluate the segments in the list from top to bottom until it finds the first item that satisfies all its conditions. This segment will be chosen as the starting segment.

Creating Stop Segments

When all cues are closed, the music engine will attempt to reach a stop (silent) state in a graceful and musical manner.

To specify a segment as a valid stop point:

1. Move the mouse pointer over the segment. The segment tray should appear.
2. Click the ■ icon. The icon should now appear red, and remain in view once the segment tray closes.

To remove the 'Stop' status, simply re-click the ■ icon.

In it's simplest form, a stop segment can be a single segment that when complete, allows the system to reach silence. However, the stop segment can be included as the final destination of a sequence of linked segments. This path is automatically followed when no themes are playing and the music system is attempting to reach silence.

To create a stopping sequence:

1 Move the mouse pointer over the segment. The segment tray should appear.

2 Make sure the segment is set as a stopping point. Click the ■ icon. The icon should now appear red, and remain in view once the segment tray closes.

3 Target the stop segment by pressing the ◎ icon. The message confirming the target status should appear. Any link made while targeting the stop segment will be identified as a red link. This path to the stop segment will be followed when all cues are inactive.

Intelligent Beat Matching

Beat matching is not a 'fix all' for providing a musically cohesive flourish. For example, the time signature 6/8 has six beats per bar - unfortunately synchronizing to any of these six beats may not provide listener with the sense of rhythmic unity. This is because humans tend to hear the beats one and four (in 6/8) as the strong beat to tap along to!

The Interactive Music View allows the sound designer to specify which beats within audio file can be accented with a concurrent theme (or flourish). To do this:

1 Open the segment's property window, by clicking the ✎ icon of the segment to change. A window dialogue pane should appear.

2 Enter the tempo of the segment's audio file using BPM (Beats Per Minute).

3 Enter the time signature of the segment's audio file. This specifies how many beats are in each bar (the audio file may contain one or more bars).

4 Press the refresh button, to calculate the length of the file.

5 A step sequencer should appear below the time signature. Each button (or step) represents a beat in the bar. For example if the time signature is 4/4, four buttons representing each 1/4 note (beat) will be visible. This is shown in the figure below.

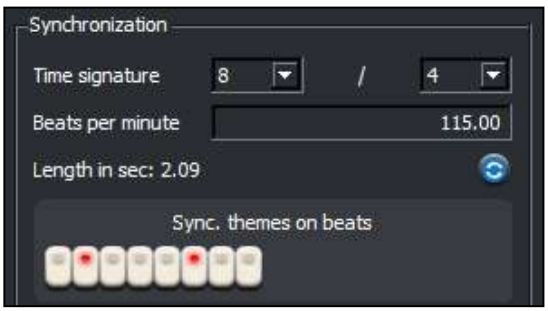


Figure 9-30: Selecting the beat

6 Select the beat(s) to declare as a potential synchronization point, by clicking the appropriate step. In the example above, beats one and three in each bar are specified as beats that can be synchronized with.

Deleting a Segment
To delete a segment:

1. Place the mouse pointer above the segment to delete. A segment tray will appear beneath the segment with a number of icons.
2. Click the ⊖ icon. A confirmation message should appear.
3. Choose 'Yes'. The segment will be deleted.

Creating and Maintaining Links

Creating Links
To create a link from one segment to another:

1. Using the mouse, click and hold the plug of the segment to leave. The plug insertion point on all valid destination segments will become visible. The segment plug shown in the figure below.

Figure 9-31: The Segment plug

2. Drag the plug to the destination segment. The destination segment's border should appear to glow border. Release the mouse button.
3. A link between the segments will become visible. If the source and destination are on the same segment, a loop icon will become visible.

Link Priorities and Conditions
To specify the priority of a segments link:

1. Open the segment's property pane by clicking the 🔧 button located on the segment inside the Theme Editor pane. A dialogue should appear.
2. Order the items in the 'Segment links' list. To do this, simply click and drag the list item to the desired position within the list (a line of dots will indicate the new position). The first position in the list represents the highest priority.

Figure 9-32: Setting the priority of links

3 With the links established, the sound designer must specify when transitions can occur. By default, transitions will occur when the audio sample has finished playback. Transitions may also be evaluated at the start of each bar, or on the beats specified in the step sequencer by clicking on the appropriate 'Link Transition Defaults' check boxes.

4 Additional cue, parameter or transition conditions can be applied to the link by double-clicking the list item. A 'Condition Editor' window dialogue should appear.

5 Drag a cue or parameter icon from its respective list and drop it on the empty Condition Editor. If using a parameter icon, you'll need to add the appropriate logic conditions such as 'less than', 'greater than', etc, as well as a numerical value.

The link may also specify its own transition behavior by selecting the 'Custom' option and clicking the appropriate checkboxes ('At segment end', 'On Bar' and/or 'On Beat') to specify when a transition can occur.

During run-time, the engine will evaluate the segment links in the list, from first to last, until it finds the first item that satisfies all its conditions. This segment link will be chosen as the path to follow.

Removing Links

To delete a segment link:

1 Open the segment's property pane by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.

2 Select the link to delete from the 'Segment Link' list.

3 Press the button located on the in list item to delete the link.

Transitioning Between Themes Using Targeting

Targeting a theme is one method for designing transitions between themes. In brief, when a cue is triggered, the associated theme is considered the 'target theme'. To make sure the music system responds correctly and transitions to the target theme, the sound designer can specify special paths. [For more information about targeting and the theme stack see page 208]

The 'target' function can be activated to clicking the button on the destination theme. When activated a message is displayed on the Theme Editor pane. This is shown in the figure below.

Figure 9-33: Targeting a theme

Once the destination theme is targeted, any links that are created between segments have a special condition included with the link. A link targeting a theme is identified by a blue colored link. This condition is satisfied when a cue associated with the target theme is triggered.

It is important to note, even when the target theme condition becomes active, the link is still prioritized within the segment link list.

The benefit of the target function is multiple links to the targeted theme can be created quickly and easily - without manually adding the theme to each link's condition editor.

Default Transitions

Links are not the only way for the music system to transition from one segment to another. 'Default transitions' can be used to define how to transition between themes – without the need for links.

Figure 9-34: Specifying Default Transition behaviour.

To edit the Default transition properties:

1. Open the theme's property pane by clicking the button located on the theme container inside the Theme Editor pane. A dialogue should appear.
2. Edit the properties inside the 'Synchronization' frame.
3. Press the 'Done' button.

Default transitions can be specified to:

- Wait a specified time to apply the default transition after being cued. This is useful for providing a fall back if the standard link transitions are taking too long or a link does not exist.

  To do this set the 'Transition timeout' property.

- Never apply a default transition.

  To do this, set the 'Default transition' property to 'Never'.

- Begin playback of the starting segment of the theme being cued, after the current segment completes playback.

  To do this, set the 'Default transition' property to 'Queue'.

- Transition immediately from the segment currently playing to the starting segment of the theme being cued.

  To do this, set the 'Default transition' property to 'Crossfade', the 'Crossfade duration'

property to 0ms and the 'Quantization' property to 'Free'.

- Crossfade between the segment currently playing and the starting segment of the theme being cued.

  To do this, set the 'Default transition' property to 'Crossfade', the 'Crossfade duration' property to 1000ms (for a one second crossfade) and the 'Quantization' property to 'Free'.

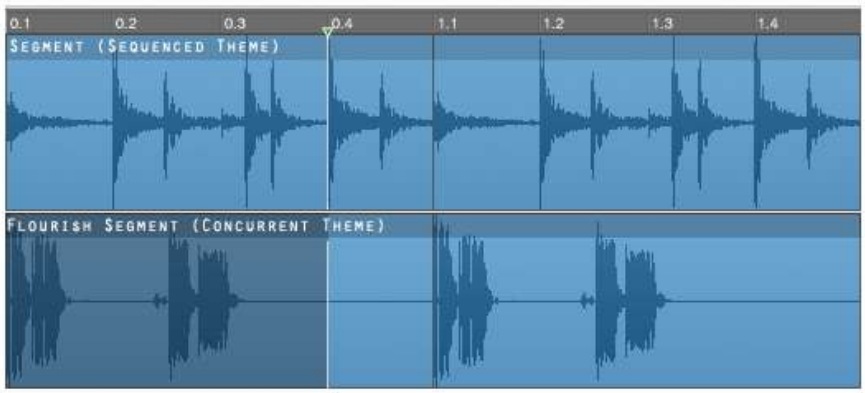
- Transition on a specified beat from segment currently playing to the starting segment of the theme being cued.

  To do this, set the 'Default transition' property to 'Crossfade', the 'Crossfade duration' property to 0ms (for a one second crossfade) and the 'Quantization' property to 'On beat'.

Shared Timelines

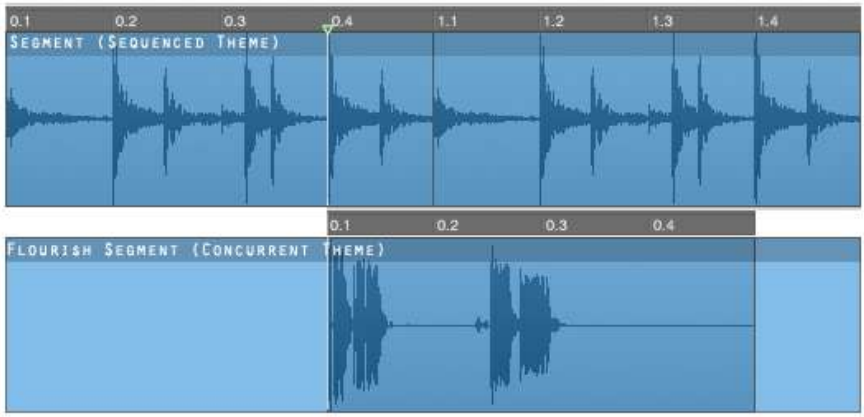
As stated earlier in the user manual, a 'shared timeline' is an object that allows the music system to distribute synchronization information between segments. A common timeline must be attached to all segments to be synchronized. Multiple timelines can be created within a project – however a segment can only be attached to one timeline.

In use, a shared timeline makes the playback of segments behave as though the segments are started at the same time – with the music system intelligently muting, unmuting or cross fading volumes of the segments where appropriate. This concept is illustrated in the figures below (assuming the segments are set to the same BPM and time signature):



Using **shared timelines** is much like aligning tracks within a DAW. This example demonstrates how the starting point of the flourish is offset in relation to the playback position of the segment currently playing.

Figure 9-35: Demonstrating shared timelines



Without **shared timelines** the flourish will play its entire audio file, at the position it was triggered.

Figure 9-36: Demonstrating concurrent playback without shared timelines

Shared timelines are best used between segments of different themes to provide more control between default transitions and flourish behaviour. For example, it is possible to create a sequenced theme featuring percussion, and cue additional harmonic content using concurrent

themes for bass, guitar, etc. Using a shared timeline between the segments will ensure the loops remain aligned, rather than just triggered in a beat-matched manner.

Creating a Timeline

To create a timeline object:

1. Open any segment's property pane by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.
2. Press the ⊕ button in the panel labelled 'Drop shared timeline here'. A pop-up window should appear. If the segment already has a timeline attached, it will need to be removed first.
3. Enter the name of the timeline in the text box of the 'Timelines' pop-up window and press the ⊕ button. The new timeline should appear in the list below – meaning it can now be added to any segment.
4. Close the 'Timelines' pop-up window.
5. Close the 'Segment Properties' pop-up window.

Renaming a Timeline

To rename a timeline object:

1. Open any segment's property pane by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.
2. Press the ⊕ button in the panel labelled 'Drop shared timeline here'. A pop-up window should appear. If the segment already has a timeline attached, it will need to be removed first.
3. Double-click the timeline to rename from the timeline list box. A pop-up window should appear.
4. Enter the new name into the text box prompt and press the ⊗ button to close the rename dialog. The timeline is now renamed.
5. Close the 'timelines' dialog and 'Segment Properties' pop-up windows.

Removing a Timeline

To rename a timeline object:

1. Open any segment's property pane by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.
2. Press the ⊕ button in the panel labelled 'Drop shared timeline here'. A pop-up window should appear. If the segment already has a timeline attached, it will need to be removed first.
3. Click the timeline to delete from the timeline list box. A ⊗ button should appear.
4. Press the ⊗ button to delete the timelines. The timeline is now removed from all segments using the timeline.

5 Close the 'timelines' dialog and 'Segment Properties' pop-up windows.

Adding a Shared Timeline to a Segment

A common timeline must be attached to all segments to be synchronized together. To add a timeline to a segment:

1 Open the property pane of segment to synchronize to the timeline by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.
2 Press the ⊕ button in the panel labelled 'Drop shared timeline here'. A pop-up window should appear.
3 Drag the timeline to attach from the timeline list box and drop it in the panel labelled 'Drop shared timeline here'. The timeline name should appear within the properties panel.
4 Press the button located on the next segment to be synchronized and repeat steps 2, 3 and 4 until the timeline is attached to all segments required.
5 Press 'Done' to close 'Segment Properties' pop-up window.

Removing a Shared Timeline from a Segment

To remove a timeline from a segment:

1 Open the property pane of segment to synchronize to the timeline by clicking the button located on the segment inside the Theme Editor pane. A dialogue should appear.
2 Press the ⊖ button in the panel labelled 'Sharing timeline <name>'.
3 The panel will change, and show now display 'Drop shared timeline here'.
4 Press 'Done' to close 'Segment Properties' pop-up window.

Music Auditioning

The Interactive Music Systems provides a number of tools that can be used by a sound designer to preview or audition their work.

- Within FMOD Designer 2010, basic auditioning of single segments or static sequences can be previewed using the function palette tools within the Theme Editor.
- More advanced pre-build auditioning, including concurrent themes and conditional sequences can be previewed using the Audition Console.
- For post-build auditioning, the standalone application FMOD Music Player is available for Windows and Mac platforms. This application is included with the FMOD Designer 2010 download.

Basic Auditioning

To audition an individual segment:

1 Select the 'Audition Tool' and click on the segment to audition. The segment will be automatically displayed in the audition transport bar.

The figure below shows a segment within the transport bar.

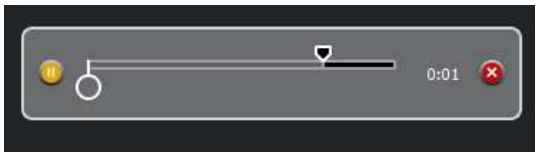


Figure 9-37: The transport bar

The position cursor of the transport bar can be moved manually to the desired point of the segment's audio file.

Auditioning Static Sequences

A sound designer can create a static linear sequence of segments to audition. This includes segments from different themes. To audition a sequence:

1 Using the 'Multi-Selection' tool, select the segments to audition in the desired sequence order. An example is shown in the figure below.

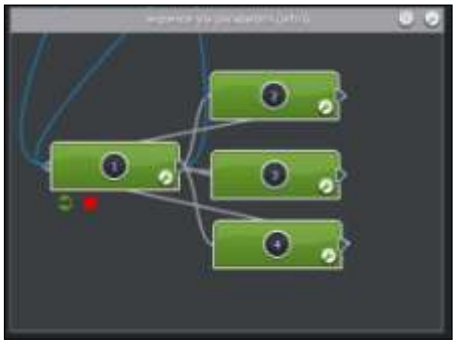


Figure 9-38: An example of multi-selection

2 Select the 'Audition Tool' and click on any of the selected segments. The selected segments will automatically displayed in the audition transport bar. The figure below shows three segments within the transport bar.

The sound designer can click any numbers in the transport bar to jump straight to the desired segment. The position cursor in the transport bar can also be moved manually to the desired point of the sequence.

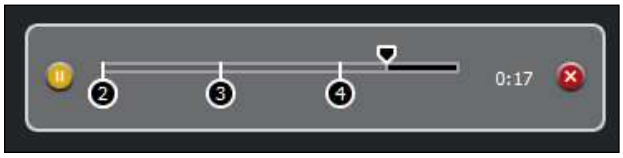


Figure 9-39: The transport bar

Dynamic Auditioning

The Audition Console allows sound designers (or Composers) to dynamically audition the current interactive music score that is being created. The Audition Console provides much more sophisticated auditioning than the static auditioning enabled by the function palette tools. The Audition Console allows non-linear sequences to be created in the same way the game engine will interact with the Interactive Music System.

Using the Audition Console, sound designers are able to audition:

- The activation and deactivation of cues
- Transition changes based on cues
- Default transitioning behavior
- Triggering of concurrent themes
- Triggering of bar and beat-matched themes
- Changes to parameter values

It should be noted, that the Audition Console operates on pre-build data, which means the sound designer hears the source audio files - not the audio file format specified in the properties of the wave bank. If the wave bank specifies high amounts of compression, the source audio files may not be an accurate indicator of the final sound.

Should a sound designer wish to audition the interactive music score post-build, the application 'FMOD Music Player' (which is included in the FMOD Designer 2010 download) may be used.

Opening and Resizing the Audition Console

The Audition Console can be found under the Theme Editor pane. In its default position the Audition Console is minimized in size and must be opened for first time use.

Figure 9-40: The Audition Console minimised

To open (or resize) the Audition Console:

1. Using the mouse, click (and hold) on the title bar of the Audition Console. The mouse pointer icon will change, indicating the bar can be moved up (or down).
2. Drag the title bar to the desired height. The Audition Console may only contain a reset button if no other controls have been added for the current project.

Docking / Releasing the Audition Console

To release the Audition Console from its Theme Editor dock:

1 Press the icon. The Audition Console will appear in its own window.
2 Move the Audition Console window to the desired position.

To return the Audition Console to its Theme Editor dock:

1 Close the window containing the Audition Console.
2 The Audition Console will now be located under the 'Theme Editor'.

Controls

By default only a reset button is included in the Audition Console. This reset control cannot be added or removed. Currently two different types of control can be added to the Audition Console. They are:

- A trigger pad for starting and stopping cues
- A fader for setting the value of a parameter

Adding a Cue Trigger Pad

A trigger pad is used to provide real-time prompts (which may invoke cues) to the music system. To add a trigger pad to the Audition Console:

1 Click the View 'Cues' to display all the cues available in the project. Remember the text field above the list operates as a search filter. To see all the available cues, this text field must be empty.
2 Drag the yellow icon of the cue to be auditioned from the cue list into the Audition Console. On releasing the mouse button, a trigger pad should appear.
3 The trigger pad can now be turned on and off to simulate changes in game state – the music system will react accordingly.

Figure 9-41: An active cue trigger pad

Adding a Parameter Fader

A parameter fader is used to provide real-time parameter value changes to the music system. To add a parameter to the Audition Console:

1 Click the View 'Parameters' to display all the parameters available in the project. Remember the text field above the list operates as a search filter. To see all the available parameters, this text field must be empty.

2 Drag the orange icon of the parameter to be auditioned from the parameter list into the Audition Console. On releasing the mouse button, a fader should appear.
3 The fader can now be moved up and down to simulate changes in game state – the music system will react accordingly.

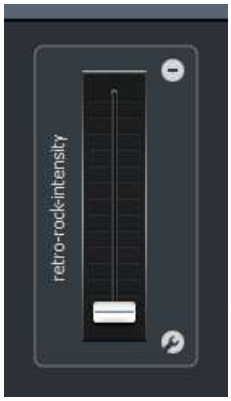


Figure 9-42: A fader controlling a parameter value

Positioning a Control

To move a control in the Audition Console:

1 Click (and hold) the outer frame of the control to be moved.
2 Drag the control to the desired position.
3 Release the mouse button.

Editing a Control's Properties

To edit the properties of a control:

1 Press the icon contained within the outer frame of the control to be modified. A pop-up window should appear.
2 Set the desired properties. A complete description of the properties is available on page 236 (for cue trigger properties) and page 237 (for fader properties).
3 Press the 'OK' button to confirm the changes.

Removing a Control

To edit the properties of a control:

1 Press the icon contained within the outer frame of the control to be deleted. A confirmation message should appear.
2 Press the 'OK' button to confirm the deletion.

Using the Audition Console

In this section, the mechanism to start and reset the Audition console is explained.

Starting the Music System

Before the Audition Console can start playback, the following must be true:

- At least one cue trigger must be added to the Audition Console.

- The cue controlled by this trigger must be associated with a theme.
- At least one segment within the associated theme must be specified as a 'Start Segment'.

With these objects in place, the Audition Console is able to begin audition. To start playback:

1 Press the cue trigger (which is associated with a theme). The cue trigger should change appearance, indicating the cue prompt is active. Any audible segment (in this case the Start Segment) in the Theme Editor will appear to glow, this indicates playback. The audio contained in the segment should now be audible.
2 Pressing another cue trigger that is associated with a 'sequenced' theme will cause the Audition Console to follow blue links to the new theme. The appropriate links and priorities must be set.
3 Pressing a cue trigger that is associated with a 'concurrent' theme will cause the Audition Console to overlay playback of the concurrent theme over the existing theme. The starting point of this 'flourish' may be immediate or at the next beat or bar – depending on the settings of quantization and sync point properties.
4 The sound designer can test conditions applied to links between segments by changing the value a parameter fader.
5 When no trigger pads are active, the Audition Console will attempt to find a Stop Segment by following red links.

Resetting the Music System

The reset button allows the sound designer to quickly terminate auditioning of the score. To reset auditioning, simply press the reset button.

Handover to the Programmer

During the build process, all the interactive music meta-data is placed into the .FEV (just like any other the event data). If a Programmer's Report is requested during the build, all cue and parameter identifications are included in the document.

The audio assets are placed in the .FSB file(s) as specified by the sound designer in Wave Bank View.

Property Reference Guide

Theme Properties

Property	Option	Description
Details		Provides a text identifier for the theme
Playback method		Specifies the behavior when multiple cues are active
	Sequenced	Playback of the new theme waiting for a transition from the existing theme
	Concurrent	The new theme will play concurrently with the existing theme. The playback starting point of the new theme is determined by the Quantization property.
Transition timeout (ms)		Specifies the maximum length of time a transition from one theme to another. If this time is exceeded, the Default transition type is invoked.
Default transition		Specifies a generic transition behavior between themes, when no path exists. This transition behavior is also invoked when the Transition timeout value is exceeded.
	Never	No transition to the new theme is made. **Note:** The cue will still exist.
	Queued	Playback of the new theme will begin when the current segment playback is complete.
	Crossfade	The new theme transitions from the old theme using a crossfade.
Crossfade duration (ms)		Specifies the length of time a crossfaded transition takes.
Quantization		Specifies when the theme will commence playback. **Note:** This property is only active when the Playback Mode is set to Concurrent.
	Free	The new concurrent theme is started as soon as the associated cue is activated.
	On Bar	The new concurrent theme is started when the existing segment starts a new bar.
	On Beat	The new concurrent theme is started on the next synchronization point of the segment currently playing.
Start Segment		A list of all the segments that are flagged as a 'Start' segment. The segment's order in this list defines its priority or weighting. Each 'Start' segment can have a cue or parameter condition attached to it.

Segment Properties

Property	Option	Description
Details		Provides a text identifier for the segment.
File		Specifies the audio file to use in the segment.
Time Signature		Specifies the time signature of the segment. Supports 1/1 to 16/8.
Beats per minute		Specifies the tempo of the segment.
Length in sec. **Non-editable**		Displays the length of the audio file in seconds.
Sync theme on beats		A step sequencer style widget that allows the sound designer to specify sync points for the segment. The sync point positions are based on the beats defined by the time signature and BPM property values.
Segment Links		A list of the links from this segment to other segments. The link's order in this list defines its priority. Each link can have a cue or parameter condition attached to it.

Playlist Properties

Property	Option	Description
Behavior		Specify the method used to select an audio item within the playlist. An item is selected each time the segment is played (or looped).
	Sequential	Selection of the item is determined by its position in the playlist. Item 1 is selected the first time the segment is played, item 2 is selected the next time the segment is played, and so on.
	Random	Selection of the item is determined randomly.
	Random without repeat	Selection of the item is determined randomly, with the additional restriction that the same item is never played twice in succession.
	Shuffle	Selection of the item is determined randomly, with the additional restriction that an item is never repeated until all items have been played.

Cue Trigger Properties

Property	Option	Description
Details		Provides a text identifier for the control.
Enable sticky mode		When this property set is to 'on' the button must be pressed once to activate, and again to deactivate. Turning this properties off is only recommended when 'prompting; concurrent themes.

Fader Properties

Property	Option	Description
Details		Provides a text identifier for the control.
Range		Specifies the range of values that are provided from the fader control to the parameter.
	Min	The minimum value avaliable on the fader.
	Max	The maximum value available on the fader.
Value always on		When this property is set to 'on' the current value of the fader is always visible. When this property is set to 'off' the current value of the fader is only visible when the mouse pointer is over the control.

Interactive Music Tutorials

Starting, Linking and Stopping

In this tutorial we shall build a basic music score. The score will contain a cue, a theme and three musical segments to demonstrate the use of:

- Starting segments
- Stopping segments
- Segment playlists
- Loops

This tutorial will also use the audition console to test the linking logic we create. You will need six audio files:

- state-race-start.ogg
- state-race-finish.ogg
- state-race-01a.ogg
- state-race-01b.ogg
- state-race-02a.ogg
- state-race-02b.ogg

These files are included with Designer download.

Getting started

1 The first thing we need to do is create a cue. A cue is the handle the programmer will use they want playback of a particular piece of music. To create a cue, type the name 'country-arena' into the text field on the 'Cue View' and press the + button.

Figure 9-43: Adding a cue

2 The cue should appear in the cue list. Click the checkbox 'Include in scene'. The cue will appear within the scene editor pane.

3 We have just set up the programmer handle. Now we need to associate the music to be played when the handle 'country-arena' is called. Locate the cue icon on the scene editor – you'll notice a + symbol where something needs to be added to the icon, so press the + button to attach a theme! A 'New Theme' will be created. You'll also notice that the GUI has switched to the theme View.

4 A theme is the container for our music. Inside the theme, you'll place segments. You can think of the theme as the 'song' and the segments as our verses, choruses, etc. We will start editing our theme by changing the name of 'New Theme'. Double-click the 'New Theme' icon in the theme list and enter 'country-music' into the text field.

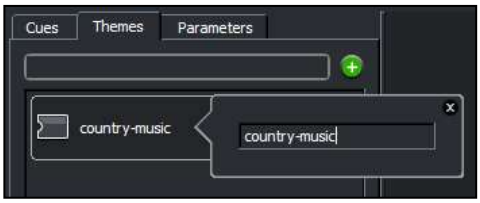


Figure 9-44: Renaming the theme

5 Press the x button to close the window.

6 Press the theme icon's edit button . This will take us to the theme editor pane.

Adding segments

7 We now need to add some audio to our theme (finally!). There are a few ways to do this. You can drag files directly into the theme from the O.S pop-up windows or you can use the 'Segment stamp' tool. We will use the latter. To select the segment stamp tool you can press 4 on your keyboard or click on the music note icon.

Figure 9-45: The segment stamp tool

8 Move the mouse pointer over the theme container. You'll notice the mouse pointer icon has changed. We need three segments – one for an intro or starting sample, one for our looping samples and one for a fadeout conclusion. Click three times on the theme container (in different locations) – this should create three segments.

Figure 9-46: Adding three segments

9 Press 1 on your keyboard to return to the selection tool.

10 Press the edit button on the first segment, the segment properties dialog should appear.

- In the text field at top, enter the name 'start'.

- Set beats per minute to '115'. (Setting this parameter is important as using the wrong BPM can create gaps between segments!)

11 In the panel 'Drop a sample here', press the + button and select the file 'state-race-start.ogg' then press the 'Done' button. If you would like to hear the segment you can use the 'audition' tool by pressing the number 3 and clicking on the segment. Warning: Banjos! Now Press 1 on your keyboard to return to the selection tool.

12 The next segment will contain main loop of our music score. To add some dynamic behavior to our music score will use a playlist of samples. Press the edit button of the second segment and again set the properties:

- In the text field at top, enter the name 'loop'.
- Set beats per minute to '115'. (Setting this parameter is important as using the wrong BPM can create gaps between segments!)

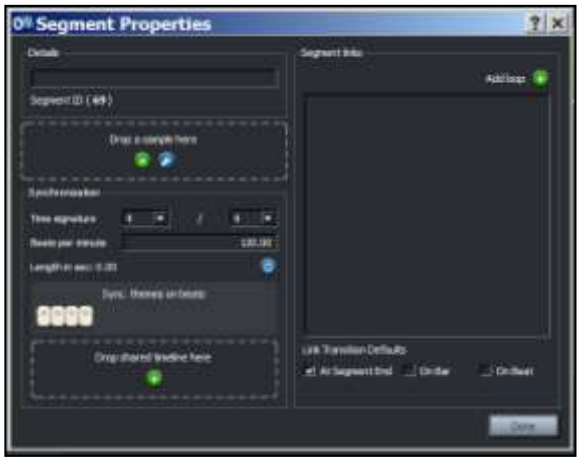


Figure 9-47: Setting up the first segment

13 In the panel 'Drop a sample here', press the edit button to display the 'Playlist Editor'.

14 Select the Playback mode 'Sequential' and press the 'Add sample' + button and add the files:

- state-race-01a.ogg
- state-race-01b.ogg
- state-race-02a.ogg
- state-race-02b.ogg

Within these files are 2 variations of the loop. You'll notice we have split the bar of music into multiple files. This is done to keep the music system responsive. By default, the music system only evaluates the next link once the current segment has finished playback – therefore the longer the sample the bigger the latency in responding to cue changes. This limitation can also be overcome using default transitions or the 'early transition' feature, and is demonstrated in later tutorials.

Each sample is only stored once in the build fsb, so you are free to add samples to the playlist multiple times, without impacting the memory usage. This allows us to use the playlist to handle some basic sequencing within the segment. Here the state-race-01 files multiple times, so the variation (state-race-02) is only heard every third loop.

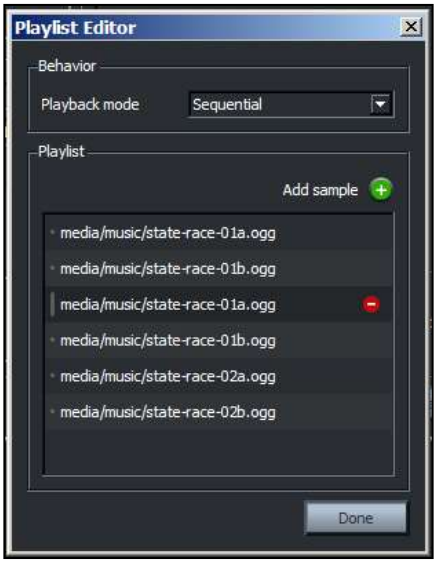


Figure 9-48: Adding the samples

Press the 'Done' button to return to segment properties dialog.

15 The final segment will contain the conclusion of our music score. The fadeout effect is baked into the sample. Press the edit button of the third segment and again set the properties:
   - In the text field at top, enter the name 'ending'.
   - Set beats per minute to '115'. (Setting this parameter is important as using the wrong BPM can create gaps between segments!)

16 In the panel 'Drop a sample here', press the + button and the file 'state-race-finish.ogg'.

17 Press the 'Done' button to return to theme editor pane.

Creating the links

18 Now we have placed the different sections with the theme, but we still need to give the music system more information about how to sequence them.

For each theme, we need to tell the music system which segments of the theme are potential starting points. To nominate a theme as a start segment, place the mouse pointer over the segment (a tray shall appear). Click the leftmost icon - a green arrow will now be visible beneath the segment. This indicates that theme can start playback on this segment.

Note:

A theme can have many 'start segments'. At run-time the music system will chose a segment by checking the priority list and conditional logic. This will be covered in another tutorial.

19 Now we need to create a link from our start segment to our main loop. To do this, grab the segment plug and drag it to the second segment. The border of the second segment will change color when a valid connection is possible. Release the mouse button to complete the connection.

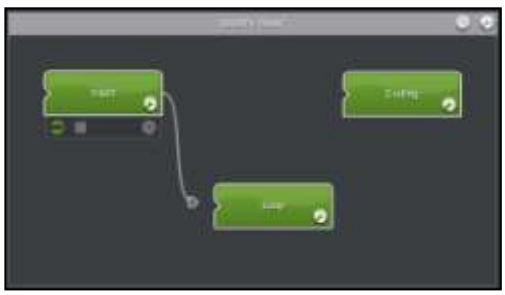


Figure 9-49: Creating a link

20 We want the second segment to loop while the cue is active. To create a loop, simply drag the segment plug of the second segment and connect it to the front of the segment. A loop icon will appear.

Making a graceful exit

21 The final link we want to make is for the conclusion. This link should only be followed when the 'country-arena' cue is ended. We need to specify a stop segment, then create a special link from the 'loop' segment to the 'ending' segment. To nominate a theme as a

stop segment, place the mouse pointer over the third segment (a tray shall appear). Click the icon, second from the left - a red square will now be visible beneath the segment. This indicates that theme can stop playback after this segment.

Note:

Stop segments (and paths to them) are only activated when no cues are active.

22 To create a 'stop path', place the mouse pointer over the third segment (a tray shall appear). Click the 'bulls eye' icon. The system is now targeting the final segment. When 'Targeting' mode is enabled, Designer 2010 will create the necessary conditions of the subsequent links automatically.

23. Create a link from the 'loop' segment to the 'ending' segment. You'll notice that this link is red.

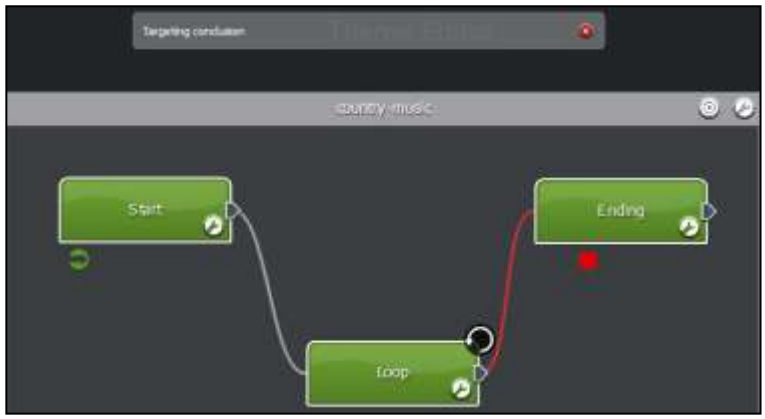


Figure 9-50: Targeting conclusion

24 Turn off targeting by pressing the x button from the 'Targeting conclusion' window.

25 The links are now complete. If you look in the second segment's properties, you notice there are two links. It is important to know this list in ranked by priority (the higher on the list, the higher the priority). The 'Link to ending' has the condition 'when targeting conclusion'. This means it will only be followed when no cues are active – even though it is on top of the link list.

Figure 9-51: Prioritzing links

As the 'Loop' item has no addition conditions, if it was on top of the list, the system would always follow it.

26 To test our linking logic, randomized playlist and start and stop segments, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.

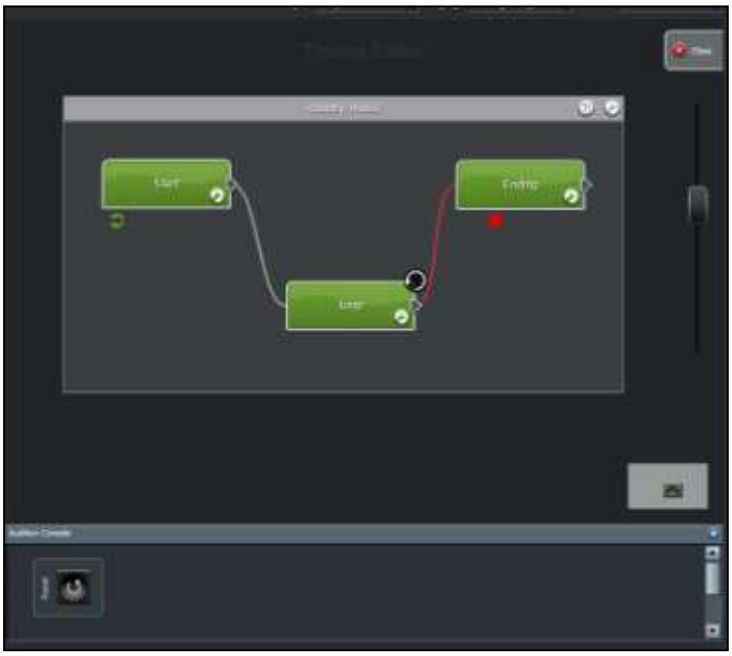


Figure 9-52: Exposing the Audition Console

27 Click the cue View.

28 Drag the 'country-arena' icon into the audition console. A button should appear.

29 Pressing the cue button will cause the music system to evaluate the start segments (in our case it only has one choice!). The start segment will be heard. When the segment is complete, the link is followed to the 'loop segment'. Playback will continue automatically to the 'loop' segment.

30 End the cue, by pressing the cue button again. As no cues are active, the music system will attempt to find a 'stop segment'. The system will then follow the link from the 'loop' segment to the 'ending' segment.

Congratulations, you now have a basic interactive music system that responds to start and stop commands.

Targeted Transitions

In the first tutorial we looked at the basic structure required to start and stop music – it was very limited, just one mood. Ideally we want our music score to move between different moods as required by the game's narrative. In this tutorial we will look at one method that allows the music system to transition between different themes.

A transition is requested when a cue is activated. Within the Interactive music system there are two distinct ways of transitioning between themes:

- Default transitions
- Targeted links

A 'Default transition' (if enabled) will allow the music system to directly transition from current segment to the starting segment of the new theme.

A 'Targeted' link is a special condition that allows the music system to follow a series of segments to the new theme. Targeted links are not limited to start points as a destination.

It is actually possible to use a combination of transition types – but that is best left for another tutorials. In this tutorial we shall build a basic music score. The score will contain three cues, three themes with a segment each to demonstrate the use of:

- targeted links

This tutorial will also use the audition console to test the linking logic we create. You will need three audio files:

- state-race-start.ogg
- state-idle-01.ogg
- interactive_music_3.ogg

These files are included with the Designer 2010 download.

Creating Cues and Themes

1 The first thing we need to do is create three cues – one for each scene in our music score. Create the first cue by typing the name 'sneaking' into the text field on the 'Cue View' and press the + button.

Figure 9-53: Adding a cue

2 Click the checkbox 'Include in scene' –and press the x button with the text field to clear the search filter.

3 Add two more cues called 'chase scene' and 'victory' and include them in the scene. The programmer would prompt these cues at the appropriate times during run time – as required by the game narrative.

Figure 9-54: Three cues in the scene editor

4 Press the + button on the 'sneaking' cue. A new theme will be created.

5 Rename the 'New Theme' by double clicking on the first 'New Theme' icon. Enter the new name 'sneaking' in the pop-up window provided.

6 Repeat steps 4 and 5 for the remaining cues using the 'chase scene' and 'victory' name.

Figure 9-55: Three cues with attached themes

7 Click the edit button of the 'sneaking' theme to enter the 'theme editor' pane. You should see three empty theme containers.

Adding Segments

8 Press 4 on the keyboard to select the 'segment stamp' tool. Click once in each theme container to create a segment for each theme. To return to the selection 'tool' press 1 on the keyboard.

9 Each theme in the music score must have a start and stop segment. For each segment, place the mouse pointer above, until the tray appears. Click on the two leftmost icons. A green arrow and red square should appear indicating that the segment will behave as a start and stop segment.

10 Press the edit button of the segment in the 'sneaking' theme. While in the property dialog:
- Set the name to 'sneak'
- Set the beats per minute to 115
- Add the file 'state-idle-01.ogg'
- Press the 'Add loop' + button
- Press the 'Done' button.

11 Press the edit button of the segment in the 'chase scene' theme. While in the property dialog:
- Set the name to 'chase'
- Set the beats per minute to 115
- Add the file 'state-race-start.ogg'
- Press the 'Add loop' + button
- Press the 'Done' button.

12 Press the edit button of the segment in the 'chase scene' theme. While in the property dialog:
- Set the name to 'win'
- Set the beats per minute to 135
- Add the file interactive_music_3.ogg'
- Press the 'Add loop' + button
- Press the 'Done' button.

Targeted Link

13 The contents of our music score are now in place – except for the links. The first link we will create will allow the system to move from the 'sneak' segment in the 'sneaking' theme to the 'chase' segment in the 'chase scene' theme.

Figure 9-56: Targeting a theme

Press the target icon of the 'chase scene' theme container. A dialog stating the 'chase scene' is targeted will appear. When 'Targeting' mode is enabled, Designer 2010 will place the necessary conditions on the subsequent links automatically.

14 Grab the segment plug of the 'sneak' segment and connect it to the 'chase' segment. A blue link will be created. This link will only be enabled when the 'chase scene' cue is triggered and the 'sneak' segment is playing.

15 Grab the segment plug of the 'win' segment and connect it to the 'chase' segment. A blue link will be created. This link will only be enabled when the 'chase scene' cue is triggered and the 'win' segment is playing.

16 Next we will specify how the music score can transition to the 'sneaking' theme. Press the target icon of the 'sneaking' theme container. A dialog stating the 'sneaking' is targeted will appear.

17 Grab the segment plug of the 'chase' segment and connect it to the 'sneak' segment. A blue link will be created. This link will only be enabled when the 'sneaking' cue is triggered and the 'chase' segment is playing.

18 Grab the segment plug of the 'win' segment and connect it to the 'sneak' segment. A blue link will be created. This link will only be enabled when the 'sneaking' cue is triggered and the 'win' segment is playing.

19 Finally we will specify how the music score can transition to the 'win' theme. To demonstrate the control that targeted links provide (over default transitions), we will specify that music system cannot transition directly from the 'sneaking' theme to the 'victory' theme – instead it must go via the 'chase' segment.

Press the target icon of the 'victory' theme container. A dialog stating the 'victory' is targeted will appear.

20 Grab the segment plug of the 'sneak' segment and connect it to the 'chase' segment. A thick grey link will be created (the thick link indicates multiple links exist). This link will only be enabled when the 'victory' cue is triggered and the 'sneak' segment is playing.

21 Grab the segment plug of the 'chase' segment and connect it to the 'win' segment.

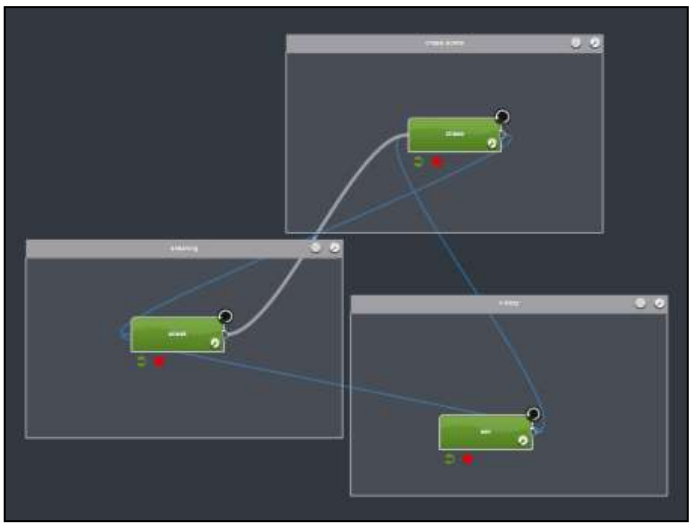


Figure 9-57: Targeted links between themes

22. Press the edit button of the 'sneaking' theme container. Set the default transition property to 'Never'.
23. Press the edit button of the 'chase scene' theme container. Set the default transition property to 'Never'.
24. Press the edit button of the 'victory' theme container. Set the default transition property to 'Never' and press the 'Done' button.

Auditioning

25. To test our linking logic, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
26. Click the cue View.
27. Drag each cue icon into the audition console. A button for each should appear.

Figure 9-58: Cues in the audition console

28 Trigger the 'sneaking' cue. The sneak segment should play.
29 Trigger the 'chase scene' cue. When the current segment finishes, the 'chase' segment should start playing.
30 End the 'chase scene' cue. Once the 'chase' segment is finished the music system should return to the 'sneaking' theme.
31 Now trigger the 'victory' theme. You notice the playback sequence follows the transition links from 'sneaky' to 'chase' to 'win'.

The targeted link method provides a lot of flexibility – it allows the composer to create elaborate and unique transitions between any segments until the desired theme is reached. While this method provides great flexibility and structure, it can suffer from a lack of responsiveness if the authored content is long in length. In the next tutorial we will look at how 'Default Transitions' can provide faster transitions.

Default Transitions

In the last tutorial we defined links between segments to sequence transitions between themes when a cue was triggered. This targeted link method provided great flexibility and structure, however if the audio files are long in length, the music system can seem unresponsive.

In this tutorial we will look at how 'Default Transitions' can provide a fast transition between the two 'moods' of our music score. The score will contain two cues, two themes and two musical segments to demonstrate the use of:

- Default transitions

You will need two audio files:

1. excited.ogg
2. relaxed.ogg

These files are included with Designer 2010 download.

Creating Cues and Themes

1. The first thing we need to do is create two cues – one for each mood in our music score. Create the first cue by typing the name 'relaxed' into the text field on the 'Cue View' and press the + button.
2. Click the checkbox 'Include in scene' –and press the x button within the text field to clear the search filter.
3. Add another cues called 'excited' and include it in the scene.
4. Press the + button on the 'relaxed' cue. A new theme will be created.
5. Rename the 'New Theme' by double clicking on the first 'New Theme' icon. Enter the new name 'relaxed' in the pop-up window provided.
6. Repeat steps 4 and 5 for the 'excited' cue using the 'excited'.

Figure 9-59: Cues with attached themes

7. Click the edit button of the attached 'relaxed' theme to enter the 'theme editor' pane. You should see two empty theme containers.

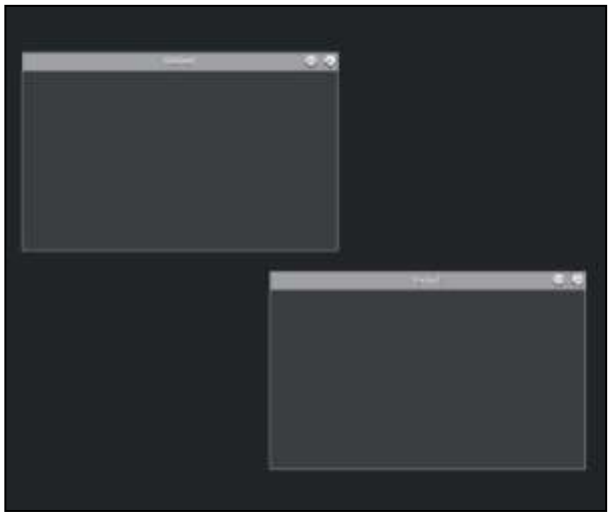


Figure 9-60: Two empty themes

Setting Up the Default Transitions

8 Click the edit button on the 'relaxed' theme container. The theme's properties should become visible.

9 By default, the music system sets new themes to use a 'queued' default transition. This allows the music system transition to the required theme, without a link, when the current segment is complete. This is a safe fall back and helps avoid the music system sitting in silence – unable to transition.

However, we want the music system to actively transition, not wait for the current segment to finish. To do this, set the 'Default transition' to 'Crossfade' and set the 'Crossfade duration (ms)' property to '2500'. Setting the default transition to crossfade tells the music system, that when the cue that is attached to this theme is trigger, the music system should immediately crossfade from the currently playing segment to the starting segment of the new theme over two and a half seconds.

Figure 9-61: Set the theme properties

If you want the music system to wait before crossfading you can use the Transition timeout (ms) property. This is handy when you want to give targeted links a chance to work, and then use the default transition as an override.

10 Click the edit button on the 'excited' theme container. The theme's properties should become visible.

11 Set the 'Default transition' to 'Crossfade' and set the 'Crossfade duration (ms)' property to '10'. This will make the transition almost immediate.

12 To add some 'musicality' to our crossfade, we will set the transition to occur on a selected beat. Set the 'Quantization' to 'On beat'.

13 Press the 'Done' button.

Setting Up the Segments

14 Press 4 on the keyboard to select the 'segment stamp' tool. Click once in each theme container to create a segment for each theme. To return to the selection 'tool' press 1 on the keyboard.

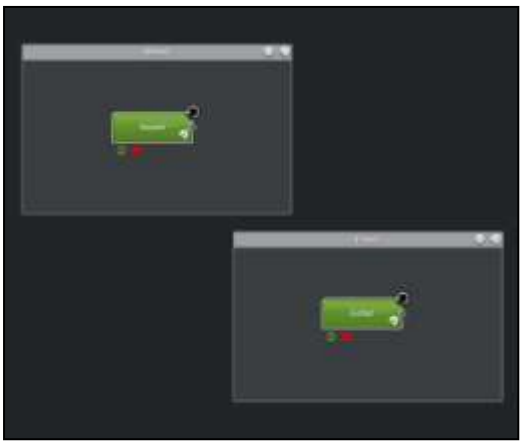


Figure 9-62: Two segments in two themes

15 Each theme in the music score must have a start and stop segment. For each segment, place the mouse pointer above, until the tray appears. Click on the two icons of the left. A green arrow and red square should appear indicating that the segment will behave as a start and stop segment.

Figure 9-63: Set the start and stop status

16 For the on-beat crossfade to work properly we are going to need to set up segment within the 'relaxed theme' appropriately. Press the edit button on the segment in the 'relaxed' theme and set the following properties:
   - Details (Name): 'relaxed'
   - Set the audio sample to: 'relaxed.ogg'
   - Time signature: '4 /4'
   - Beats per minute: '116'
17 Press the first and third buttons on the 'step sequence'. This means that the first and third beats of every bar are valid points to transition from.
18 Press the 'Add loop' + button.
19 Press the edit button on the segment in the 'excited' theme and set the following properties:
   - Details (Name): 'excited'
   - Set the audio sample to: 'excited.ogg'
   - Time signature: '4 /4'
   - Beats per minute: '116'
20 Press the 'Add loop' + button, then the 'Done' button.

Auditioning

21 To test our default transitions, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
22 Click the cue View.
23 Drag the 'relaxed' cue icon into the audition console. A button should appear.
24 Drag the 'excited' cue icon into the audition console. A button should appear.

Figure 9-64: Two cues in the audition console

25 Start the 'excited' theme by pressing the 'excited' cue. Playback should start immediately.
26 When ready, press the 'relaxed' cue. The music system should immediately begin to fade to the 'relaxed' theme. The crossfade will take two and a half seconds, as defined in the theme properties.
27 Transition back to the 'excited' theme by ending the 'relaxed' cue. Listen carefully to the transition. It might be worthwhile repeating steps 25 and 26 a number of times. Note how the transition from 'relaxed' to 'excited' is quick and falls musically on the beat.

Conclusion

This tutorial has demonstrated the use of default transitions. The only limitation with default transitions is that you can only transition to a starting segment within the new theme. Default transitions can be used to create highly responsive and musical transitions between themes or along side targeted links as a safety fall back.

Creating a Random Flourish

This tutorial demonstrates how a cue can be used to start playback of a random flourish. Within the FMOD lexicon, a sound played in a concurrent theme is called a 'flourish'.

One of the nice features of the music system is that the playback of the flourish can be synchronized to a beat of the underlying segment. In fact, the composer can choose which beats are valid sync points.

Using the technique described in this tutorial, you could tie a game action like picking up a coin, to trigger a musical phrase that plays over the current music score. Even better, should the character run through a series of coins, each pickup could be synchronized to the underlying beat to make a nice harmonic phrase!

This score will contain two cues, two themes and two musical segments to demonstrate the use of:

- concurrent themes
- beat matched flourishes
- segment playlist randomization

This tutorial will also use the audition console to test the cues we create. You will need eight audio files:

- flsh-idle-01.ogg
- flsh-idle-02.ogg
- flsh-idle-03.ogg
- flsh-idle-04.ogg
- flsh-idle-05.ogg
- state-idle-01.ogg
- state-idle-02.ogg
- state-idle-03.ogg

Setting up cues

1. The first thing we need to do is create a cue for the underlying music base. Create the cue by typing the name 'base layer' into the text field on the 'Cue View' and press the + button.
2. Click the checkbox 'Include in scene' – and press the x button with the text field to clear the search filter.
3. Create another cue called 'flourish' and add it to the scene.

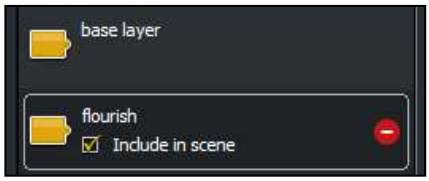


Figure 9-65: Two cues

Setting Up the Foundation

4 Attach a theme to the 'base layer' cue by pressing the + button on the 'base layer' cue icon (on the scene editor). A new theme will be created.

5 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'base theme' in the pop-up window provided.

6 Enter the 'theme editor' pane by pressing either the theme edit button on the 'base theme'.

7 Press 4 on the keyboard (or select the 'segment stamp' tool) – you'll notice the mouse pointer changes.

8 Add a segment to the empty 'base theme' container. This segment will provide the foundation of our music score.

9 Press 1 on the keyboard (or select the 'select' tool) – you'll notice the mouse pointer changes back to normal.

10 Specify the empty segment as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.

11 Press the edit button of the new segment and set the following properties:

- Name: Main loop
- Beats per minute: 115
- 12 Open an O.S window and find the files:
- state-idle-01.ogg
- state-idle-02.ogg
- state-idle-03.ogg

then drag and drop them into the panel labeled 'Drop a sample here'. The panel should now read '(multiple samples)'.

13 Press the edit button on the '(multiple samples)' panel to enter the playlist editor.

14 Set the 'Playmode mode' property to ‘Random without' repeat.

15 Press the 'Done' button of the playlist editor.

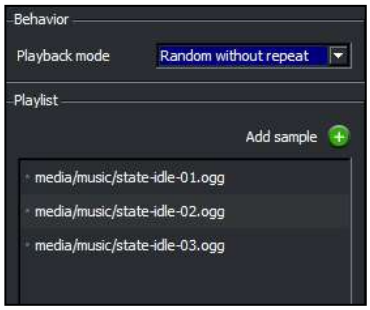


Figure 9-66: The main loop playlist

16 Press the first and third buttons of the panel labeled 'Sync. Themes on beats'. This specifies that the playback of any concurrent themes will occur on the first and third beats of this segment.

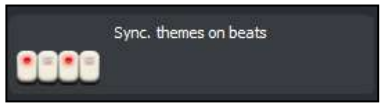


Figure 9-67: The step sequencer

17 Press the 'Add loop' + button.

Figure 9-68: The complete main loop segment

Adding the Flourishes

18 The foundation of our music score is now in place. We know need to create the theme that contains our random flourish. Press the close button on the 'theme editor' to return to the 'scene editor'.

Figure 9-69: Close the theme editor

19 Attach a theme to the 'flourish' cue by pressing the + button on the 'flourish' cue icon (on the scene editor). A new theme will be created.

20 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'flourish theme' in the pop-up window provided.
21 Enter the 'theme editor' pane by pressing either the theme edit button on the 'flourish theme'.
22 Press 4 on the keyboard (or select the 'segment stamp' tool) – you'll notice the mouse pointer changes.
23 Add a segment to the empty 'flourish theme' container. This segment will provide the foundation of our music score.
24 Press 1 on the keyboard (or select the 'select' tool) – you'll notice the mouse pointer changes back to normal.
25 Specify the empty segment as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.
26 Press the edit button of the new segment and set the following properties:
- Name: flourish
- Beats per minute: 115
- 27 Open an O.S window and find the files:
- flsh-idle-01.ogg
- flsh-idle-02.ogg
- flsh-idle-03.ogg
- flsh-idle-04.ogg
- flsh-idle-05.ogg

28 Select the files in the O.S window and drag and drop them in the panel labelled 'Drop a sample here'. The panel should now read '(multiple samples)'.
29 Press the edit button on the '(multiple samples)' panel to enter the playlist editor.
30 Set the 'Playmode mode' property to ‘Random without' repeat.

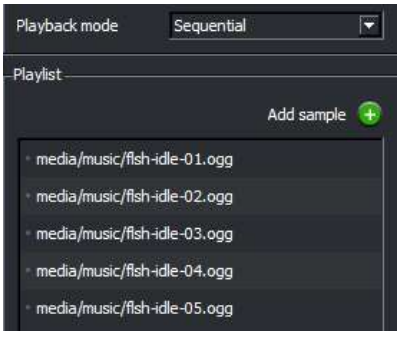


Figure 9-70: The flourish segment playlist

31 Press the 'Done' button of the playlist editor.
32 Press the 'Done' button of the segment properties pane

Figure 9-71: The complete flourish segment

33 The 'flourish theme' must now be set to play concurrently theme other themes. To do this, press the edit button on the 'flourish theme' container.

Figure 9-72: Edit the flourish theme

34 Set the 'Playback method' property to 'Concurrent'.
35 Set the 'Quantization' property to 'On beat'. This tells the interactive music system to start playback on the beats specified in the segments of the base theme.
36 Press the 'Done' button.

Auditioning

37 To test our random flourish, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
38 Click the cue View.
39 Drag the 'base layer' cue icon into the audition console. A button should appear (make sure the search filter is cleared so you can see all the cues).
40 Drag the 'flourish' cue icon into the audition console. Another button should appear.

Figure 9-73: Two cues to audition

Within the music system API there are two ways for a programmer to request music.

- 'PrepareCue' (where the cue must be started and explicitly ended).
- 'PromptCue' (which starts a cue then automatically ends it)

Prompts were generally designed to operate with 'one-shot' style flourishes. The programmer can 'prompt' a flourish and forget about the cue, as it will be stopped automatically. Of course this all occurs at the programmer level, and requires the programmer to make the appropriate prepare or prompt call.

The sound designer can test the affects of a prompt or prepare call using the audition console. By default, the button will behave as a 'prepare' call (the cue must be explicitly started AND stopped).

In this tutorial, the flourish will operate as a 'PromptCue'.

41 Press the edit button on the flourish cue button in the audition console. Using the properties pane that appears, you can turn off 'sticky mode'. The button will now behave like a prompt (the button starts the cue and immediately switches of).

   The 'sticky' mode button is for auditioning only! It does not change anything in the .fev or .fsb...it just lets you hear what happens if the programmer uses a 'prepare' or 'prompt' call.

Figure 9-74: Turn off sticky mode

42 Start auditioning by pressing the 'base layer' cue. Playback of the foundation of our music score will start.
43 Press the 'flourish' cue. You'll notice the cue button de-activates automatically. The playback may not be immediate, as the playback is synchronized to the first or third beat of each bar.
44 Keep pressing the 'flourish' cue and you will hear the random variations. Pressing the cue quickly you notice that you can stack multiple instances on a single beat.

Conclusion

In this tutorial we looked at just one on the benefits of a concurrent themes. In other tutorials we will investigate how to divide a musical arrangement (instruments) over multiple layers and how we can use a parameter to select which flourish is played.

Creating a Parameter-based Flourish

This tutorial demonstrates how the music system can choose from multiple start segments using a parameter. In this example, the parameter will determine which sound file is used as a 'flourish'. As noted in an earlier tutorial, a sound played in a concurrent theme is called a 'flourish'.

In a previous tutorial, we looked at triggering a random flourish – now we will now look at selecting a particular flourish sound by taking advantage of a theme's ability to specify multiple potential starting segments. This score will contain two cues, one parameter, two themes and six musical segments to demonstrate the use of:

- concurrent themes
- beat matched flourishes
- conditional start segment

This tutorial will also use the audition console to test the transition logic we create. You will need eight audio files:

- flsh-idle-01.ogg
- flsh-idle-02.ogg
- flsh-idle-03.ogg
- flsh-idle-04.ogg
- flsh-idle-05.ogg
- state-idle-01.ogg
- state-idle-02.ogg
- state-idle-03.ogg

Setting Up Cues and the Parameter

1. The first thing we need to do is create a cue for the underlying music base. Create the cue by typing the name 'base layer' into the text field on the 'Cue View' and press the + button.
2. Click the checkbox 'Include in scene' – and press the x button with the text field to clear the search filter.
3. Create another cue called 'flourish' and add it to the scene.
4. Click the 'Parameter' View.
5. Create the parameter by typing the name 'flourish' into the text field on the 'Parameter View' and press the + button.

Setting up the Foundation

6 Attach a theme to the 'base layer' cue by pressing the + button on the 'base layer' cue icon (on the scene editor). A new theme will be created.

7 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'base theme' in the pop-up window provided.

8 Enter the 'theme editor' pane by pressing either the theme edit button on the 'base theme'.

9 Press 4 on the keyboard (or select the 'segment stamp' tool) – you'll notice the mouse pointer changes.

10 Add a segment to the empty 'base theme' container. This segment will provide the foundation of our music score.

11 Press 1 on the keyboard (or select the 'select' tool) – you'll notice the mouse pointer changes back to normal.

12 Specify the empty segment as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.

13 Press the edit button of the new segment and set the following properties:

- Name: Main loop
- Beats per minute: 115
- 14 Open an O.S window and find the files:
- state-idle-01.ogg
- state-idle-02.ogg
- state-idle-03.ogg

15 Select the files in the O.S window and drag and drop them in the panel labeled 'Drop a sample here'. The panel should now read '(multiple samples)'.

16 Press the edit button on the '(multiple samples)' panel to enter the playlist editor.

17 Set the 'Playmode mode' property to ‘Random without' repeat.

18 Press the 'Done' button of the playlist editor.

19 Press the first and third buttons of the panel labelled 'Sync. Themes on beats'. This specifies that the playback of any concurrent themes will occur on the first and third beats of this segment.

20 Press the 'Add loop' + button.

Adding the Flourishes

21 The foundation of our music score is now in place. We know need to create the theme that contains our random flourish. Press the close button on the 'theme editor' to return to the 'scene editor'.
22 Attach a theme to the 'flourish' cue by pressing the + button on the 'flourish' cue icon (on the scene editor). A new theme will be created.
23 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'flourish theme' in the pop-up window provided.
24 Enter the 'theme editor' pane by pressing either the theme edit button on the 'flourish theme'.
25 Open an O.S window and find the files:
   - flsh-idle-01.ogg
   - flsh-idle-02.ogg
   - flsh-idle-03.ogg
   - flsh-idle-04.ogg
   - flsh-idle-05.ogg
26 Drag and drop the files into the empty 'flourish theme' container. Five new segments should be created.
27 For each of the five 'flourish theme' segments, press the edit button and set the property 'Beats per minute' to 115.
28 Specify all the 'flourish theme' segments as a 'start' and 'stop' segment by placing the mouse over each segment and then clicking the green arrow and red square icons from the tray that appears. This means they are potential starting points. Later in the project we will apply conditions to select each segment via a parameter value.
29 Press the 'Done' button of the playlist editor.
30 Press the 'Done' button of the segment properties pane.
31 The 'flourish theme' must now be set to play concurrently theme other themes. To do this, press the edit button on the 'flourish theme' container.
32 Set the 'Playback method' property to 'Concurrent'.
33 Set the 'Quantization' property to 'On beat'. This tells the interactive music system to start playback on the beats specified in the segments of the base theme.
34 Click on the 'Parameter View'.
35 In the 'Starting segments' list find the item 'flsh-idle-01' and double click it. A pop-up window should appear.
36 Drag the 'flourish' parameter on to the pop-up window labeled, 'Drag on cue, theme or parameter'. The condition editor will become visible.
37 Set the condition to 'Activate when Flourish is equal to 1'.
38 Press the close button.
39 Repeat steps 35 to 38, setting:

- flsh-idle-02 to activate when the parameter is equal to 2
- flsh-idle-03 to activate when the parameter is equal to 3
- flsh-idle-04 to activate when the parameter is equal to 4
- flsh-idle-05 to activate when the parameter is equal to 5

40. Press the 'Done' button.

Auditioning

41. To test our selected flourishes, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
42. Click the cue View.
43. Drag the 'base layer' cue icon into the audition console. A button should appear (make sure the search filter is cleared so you can see all the cues).
44. Drag the 'flourish' cue icon into the audition console. Another button should appear.
45. Drag in 'flourish' parameter to the audition console..
46. Press the edit button on the flourish cue button in the audition console. Using the properties pane that appears, you can turn off 'sticky mode'. The button will now behave like a prompt (the button starts the cue and immediately switches of).
47. Press the edit button on the 'flourish' fader.
48. Set the range from 1.0 to 5.0. This change only affects the control of the fader object while auditioning. It does not limit or validate the range of the parameter when executed in the game.
49. Press 'Done'.
50. Start auditioning by pressing the 'base layer' cue. Playback of the foundation of our music score will start.
51. Press the 'flourish' cue. You'll notice the cue button de-activates automatically. The playback may not be immediate, as the playback is synchronized to the first or third beat of each bar.
52. Move the parameter to (exactly) 2, 3, 4 or 5 and again press the 'flourish' cue.
53. Repeat steps 51 and 52. You'll notice you can now control which flourish is played.

Conclusion

In this tutorial we created a music score that let us pick which flourish is played, using a combination of starting segments, conditions and a parameter.

Choosing a Link via a Parameter

In this tutorial we will look at how a parameter can be used to determine the sequence of links that are followed. This technique uses a number of segments and applies conditions to the links between them.

This score will contain one cue, one parameter, one theme and four segments (including one empty segment to control linking logic) to demonstrate the use of:

- segment linking
- parameters
- linking

Parameters are the method that numerical data from the game engine can be used in the music system. In this tutorial we will create a parameter that represents the 'intensity' state of the game we are scoring. How this intensity value is created in the game, doesn't concern us – we are just going to assume it is a value between 0 and 10. This tutorial also demonstrates how an empty segment can be used creatively to control link logic.

This tutorial will use the audition console to test the transition logic we create.

You will need three audio files:

- interactive_music_1.ogg
- interactive_music_2.ogg
- interactive_music_3.ogg

These files are included with Designer 2010 download.

Creating a Cue, Parameter and Themes

1. The first thing we need to do is create a cue. Create the cue by typing the name 'music' into the text field on the 'Cue View' and press the + button.
2. Click the checkbox 'Include in scene' –and press the x button with the text field to clear the search filter.

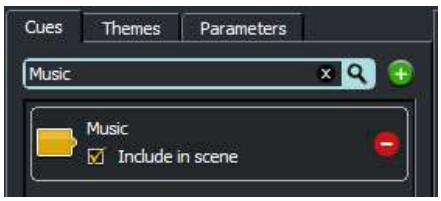


Figure 9-75: Adding cue

3 Click the parameter View.

4 Create a parameter by typing the name 'intensity' into the text field on the 'Parameter View' and press the + button.

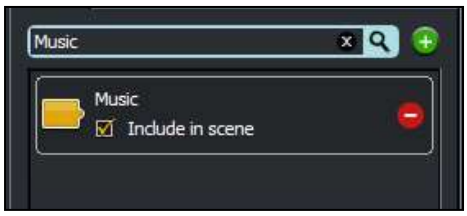


Figure 9-76: Adding parameter

5 Press the + button on the 'music' cue icon (on the scene editor). A new theme will be created.

6 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'intensity theme' in the pop-up window provided.

Figure 9-77: Renaming the theme

7 Enter the 'theme editor' pane by pressing either the theme edit button on the 'music-intensity theme' icon or using the center (looks like a crosshair) button on the theme icon within theme list.

Creating the Segments

8 Open the folder that contains the audio files, using your operating system.

9 Select the three audio files (interactive_music_1.ogg, interactive_music_2.ogg and interactive_music_3.ogg) and drag them into the empty theme container in the Designer 2010 application. This should create three new segments, each containing a file.

Figure 9-78: An empty theme container.

You should see the following:

Figure 9-79: Three segments

10 For each segment, press the edit button and set the 'Beats Per Minute' property to 135.

11 Press 4 on the keyboard (or select the 'segment stamp' tool) – you'll notice the mouse pointer changes.

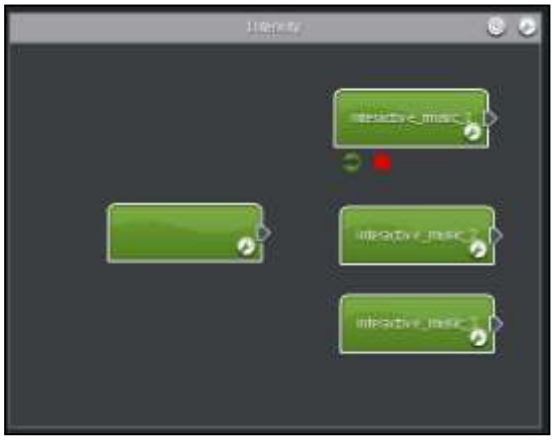


Figure 9-80: Four segments ready for linking

12 Add another segment to the theme. This segment will remain empty! We will use this segment purely to control the linking.

13 Press 1 on the keyboard (or select the 'select' tool) – you'll notice the mouse pointer changes back to normal.

14 Grab the 'segment plug' of the new (empty) segment and connect it to 'interactive_music_01'.

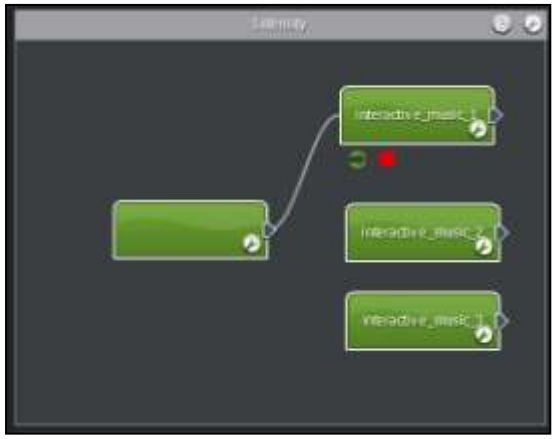


Figure 9-81: Linking to music segment one

15 Grab the 'segment plug' of 'interactive_music_1' and connect it back to the empty segment.

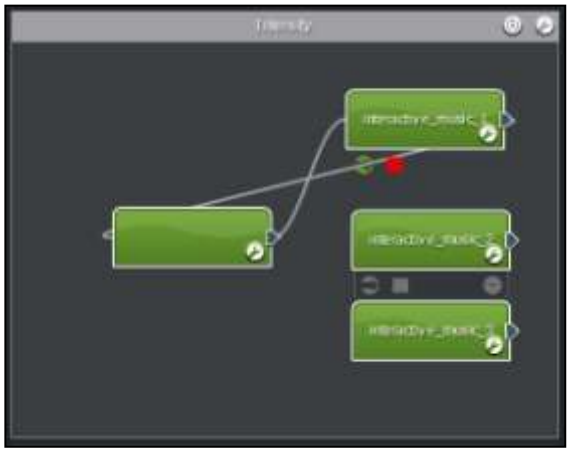


Figure 9-82: Linking back from music segment one

16 Grab the 'segment plug' of the new (empty) segment and connect it to 'interactive_music_02'.

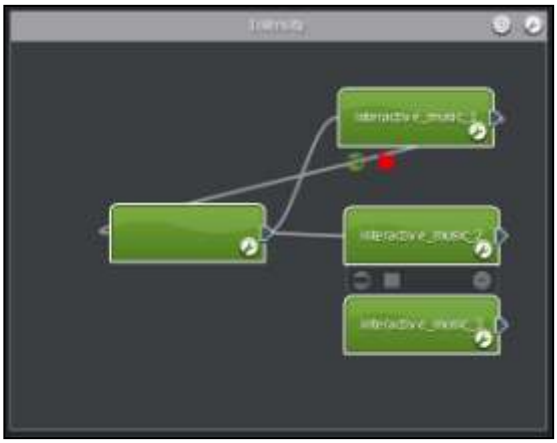


Figure 9-83: Linking to music segment two

17 Grab the 'segment plug' of 'interactive_music_2' and connect it back to the empty segment.

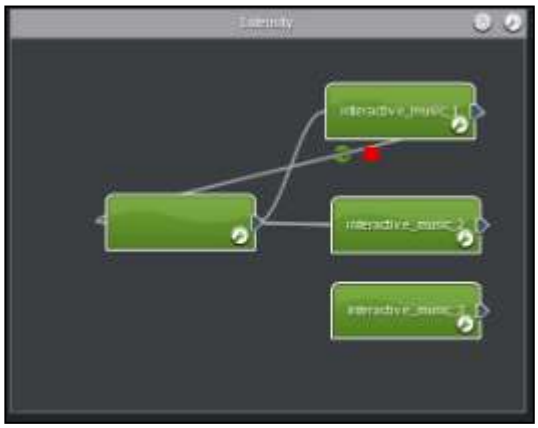


Figure 9-84: Linking back from music segment two

18 Grab the 'segment plug' of the new (empty) segment and connect it to 'interactive_music_03'.

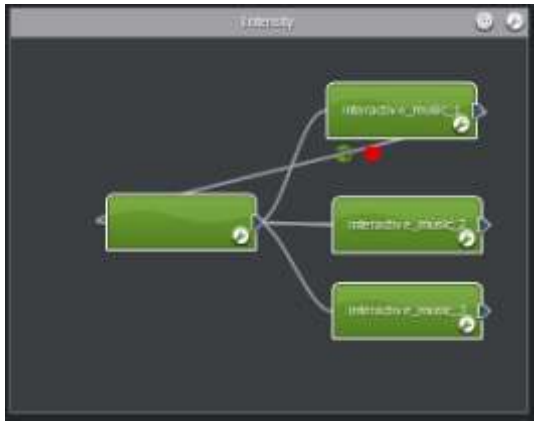


Figure 9-85: Linking to music segment three

19 Grab the 'segment plug' of 'interactive_music_3' and connect it back to the empty segment.

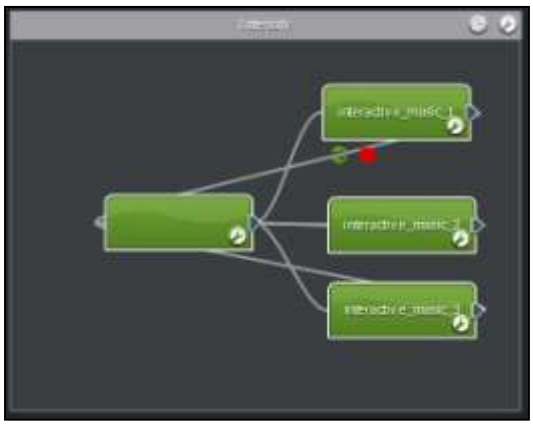


Figure 9-86: Linking back from music segment three

20 Specify the empty segment as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.

21 At this point you might also want to press the edit button of the empty segment and set the following properties.

- Name: link logic

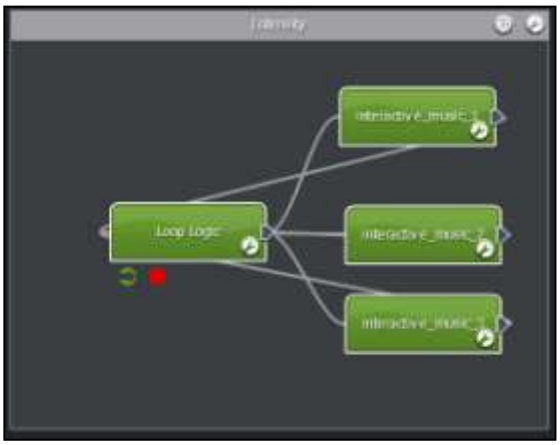


Figure 9-87: Segments and link now require conditions.

Using a Parameter

22 At this point our music score has the cue, all the music and links – however we need to tie the links to the 'intensity' parameter. In this example, will divide our parameter space of 0 to 10 as:

- From 0 to 4 will represent low intensity and play 'interactive_music_1'.
- Between 4 and 8 will represent regular intensity and play 'interactive_music_2'

- From 8 to 10 will represent high intensity and play 'interactive_music_3'

Figure 9-88: Press the edit button on the 'loop logic' segment

23 Double click the link item 'Link to interactive_music_3'. A dialog will appear.
24 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
25 Set 'Activate when 'intensity' is' greater than including 8.
26 Close the pop-up window pressing the x button.
27 Double click the link item 'Link to interactive_music_2'. A dialog will appear.
28 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
29 Set 'Activate when 'intensity' is' between 4 and 8.

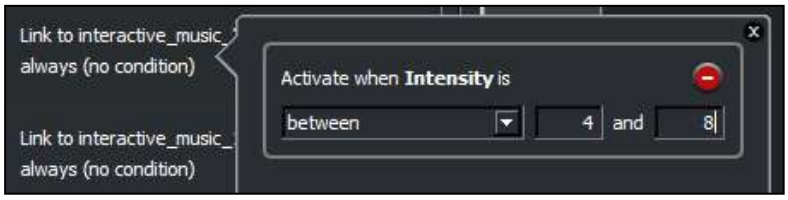


Figure 9-89: Setting conditions

30 Close the pop-up window pressing the x button.
31 Double click the link item 'Link to interactive_music_1'. A dialog will appear.
32 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
33 Set 'Activate when 'intensity' is' less than including 4.

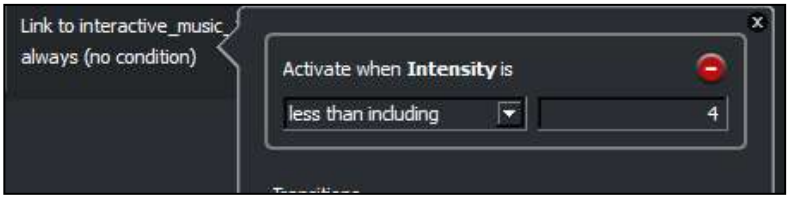


Figure 9-90: Setting conditions

34 Close the pop-up window pressing the x button.

35 Press the 'Done' button.

Auditioning

36 To test our default transitions, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
37 Click the cue View.
38 Drag the 'music' cue icon into the audition console. A button should appear.
39 Click the parameter View.
40 Drag the 'intensity' parameter icon into the audition console. A fader should appear.

Figure 9-91: The audition console

41 Press the 'music' cue button in the audition console. The score should start immediately. You'll notice that the segment 'interactive_music_1' is playing.

This is because the 'intensity' audition fader is still on 0.

42 Move the fader up until you reach around 5. When the current segment ends, you'll notice that 'interactive_music_2' is played.
43 Move the fader up to 10. When the current segment ends, you'll notice that 'interactive_music_3' is played.

Conclusion

So how does it work?

The cue is triggered making the music system play the 'music intensity' theme. The music system goes to the 'start' segment, which as we know is empty. This causes the music system to immediately evaluate which link to follow – it goes through our list of links, examining the value of the parameter and conditions. When it finds the valid link, the score transitions to that segment and continues playback.

Once the segment has finished, it returns back to our empty segment where the process is started again.

The great thing about this empty segment is it reduces the number of links we need to maintain. Without it, we would need to create a link for every possible transition between the three music segments…boring!

Who would have thought an empty segment could be so useful?

Build Music with Layers

In this tutorial we will look at how themes can be stacked to provide dynamic music scores. In this scenario, each theme is analogous to a 'track' in a DAW, which we can mute or unmute using cues. The guitar track in this example is connected to a parameter, to provide further dynamic behavior.

This score will contain three cues, one parameter, three themes and five segments (including one empty segment to control linking logic) to demonstrate the use of:

- current themes
- shared timelines
- parameters

This tutorial will also use the audition console to test the score we create.
You will need thirteen audio files:

- layer-bass-a.ogg
- layer-drums-a.ogg
- layer-drums-b.ogg
- layer-drums-c.ogg
- layer-drums-d.ogg
- layer-guitar-a.ogg
- layer-guitar-b.ogg
- layer-guitar-c.ogg
- layer-guitar-d.ogg
- layer-guitar-alt-a.ogg
- layer-guitar-alt-b.ogg
- layer-guitar-alt-c.ogg
- layer-guitar-alt-d.ogg

These files are included with Designer 2010 download.

Creating the Cues, Themes and Parameter

1. Click on the 'Cues' View.
2. Add a cue called 'drums', click the 'Add to scene' checkbox.
3. Add a cue called 'bass', click the 'Add to scene' checkbox.
4. Add a cue called 'guitar', click the 'Add to scene' checkbox.

Figure 9-92: Three cues

5 Click the 'Parameters' View.

6 Enter 'variation' in the text field, and press the + button. A parameter called variation should be created.

7 Click on the 'Themes' View.

8 Add a theme called 'drums'.

9 Add a theme called 'bass'.

10 Add a theme called 'guitar'.

11 Ensure the theme search filter is cleared, so all themes are visible in the theme list.

12 Drag and drop the 'drums' theme icon over the 'drums' cue in the Scene Editor.

Figure 9-93: Attaching a theme to the drums cue

13 Drag and drop the 'bass' theme icon over the 'bass' cue in the Scene Editor.

14 Drag and drop the 'guitar' theme icon over the 'guitar' cue in the Scene Editor.

15 Click the edit button on the 'drums' theme to enter the 'Theme Editor' pane.

At this point you might need to organize the theme containers so they don't overlap each other.

Figure 9-94: Editing a theme

Making the Drum Track

16 Press 4 (or select the segment stamp tool) and add an empty segment in the drums theme.
17 Press 1 to return to the selection tool.
18 Press the edit button on the empty segment, this should display the segment properties.
19 Open the folder that contains the audio files, using your operating system.
20 Select the four audio files, layer-drums-a.ogg, layer-drums-b.ogg, layer-drums-c.ogg, layer-drums-d.ogg and drag them into the'Drop samples here' panel with the segment properties pane.
21 Set the following segment properties:
   - 'Details' text field to 'drums'
   - 'Beats Per Minute' property to 91
22 Select all the beats in the step sequencer.

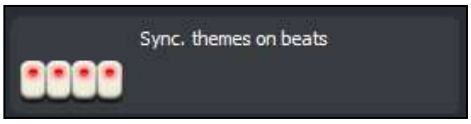


Figure 9-95: Setting the beats to sync to

23 Press the + button in the shared timeline panel.
24 Enter the name 'sync' into the text field, and press the + button. This will create a new 'shared timeline' called sync.
25 Drag the new 'sync' item back on the 'Drop shared timeline' panel of the system properties.
26 Set the drum segment to loop by pressing the 'Add loop' + button.

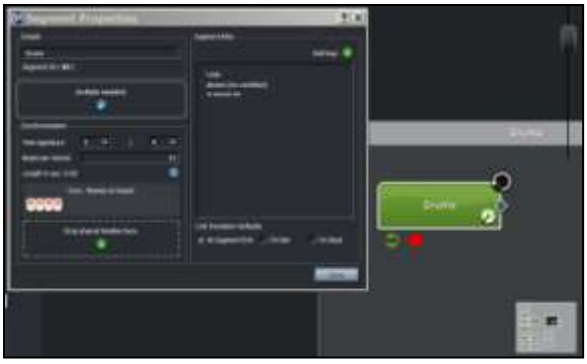


Figure 9-96: The settings for the drums segment

27 To complete the drum segment, specify it as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.

Making the Bass Track

28 Click the edit button of the 'bass' theme.
29 Set the following properties:
- Playback method: Concurrent
- Quantization: On beat

30 Press done.
31 Open the folder that contains the audio files, using your operating system.
32 Drag the sample layer-bass-a.ogg from the operating system window and drop it into the 'bass' theme container. This will create a segment.
33 Click the edit button if the new segment.
34 Set the following segment properties:
- 'Details' text field to 'bass' (from layer-bass-a)
- 'Beats Per Minute' property to 91

35 Press the + button in the shared timeline panel.
36 Drag the item 'sync' back on the 'Drop shared timeline' panel of the system properties.
37 Set the bass segment to loop by pressing the 'Add loop' + button.

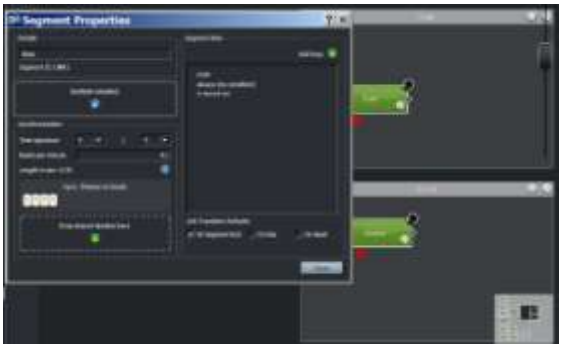


Figure 9-97: The settings for the bass segment

38 To complete the bass segment, specify it as a 'start' and 'stop' segment by placing the mouse over the segment and then clicking the green arrow and red square icons from the tray that appears.

Making the Guitar Track

In this section, we shall construct a dynamic guitar track that is able to switch between two different guitar pieces. To do this we will use an empty segment connected to a parameter that will pick the appropriate guitar piece to play at run time.

39 Click the edit button of the 'guitar' theme.
40 Set the following properties:
- Playback method: Concurrent
- Quantization: On beat

41 Press done.

42 Press 4 (or select the segment stamp tool) and add three empty segments in the guitar theme.

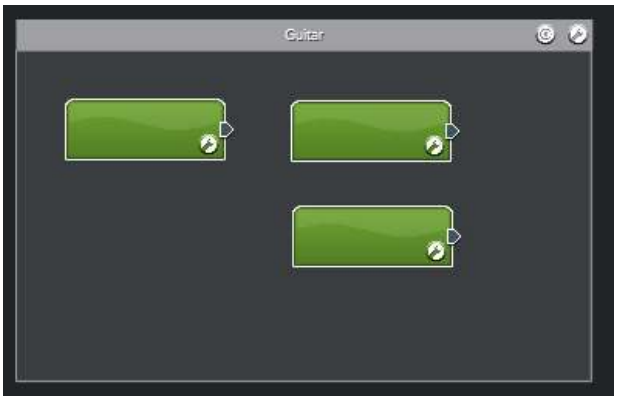


Figure 9-98: Layout of guitar segments

43 Press 1 to return to the selection tool.

44 Press the edit button on the first empty segment, this should display the segment properties.

45 Set the following segment properties:

- 'Details' text field to 'switch (empty)'
- 'Beats Per Minute' property to 91

46 Specify it as a 'start' and 'stop' segment by placing the mouse over the 'switch (empty)' segment and then clicking the green arrow and red square icons from the tray that appears.

Figure 9-99: Setting the guitar start and stop segment

47 Press the edit button on the second segment.

48 Open the folder that contains the audio files, using your operating system.

49 Select the four audio files, layer-guitar-a.ogg, layer-guitar-b.ogg, layer-guitar-c.ogg, layer-guitar-d.ogg and drag them into the'Drop samples here' panel with the segment properties pane.

50 Set the following segment properties:

- 'Details' text field to 'guitar alt'
- 'Beats Per Minute' property to 91

51 Press the + button in the shared timeline panel.

52 Drag the item 'sync' back on the 'Drop shared timeline' panel of the system properties.

53 Press the edit button on the third, and final guitar segment.

54 Open the folder that contains the audio files, using your operating system.

55 Select the four audio files, layer-guitar-alt-a.ogg, layer-guitar-alt-b.ogg, layer-guitar-alt-c.ogg, layer-guitar-alt-d.ogg and drag them into the'Drop samples here' panel with the segment properties pane.

56 Set the following segment properties:

- 'Details' text field to 'guitar'
- 'Beats Per Minute' property to 91

57 Press the + button in the shared timeline panel.

58 Drag the item 'sync' back on the 'Drop shared timeline' panel of the system properties.

At this point all the audio content has been added to the guitar track. Now all the linking logic must be added.

59 Link the 'switch (empty)' segment to the 'guitar' segment and then the 'guitar alt' segment.

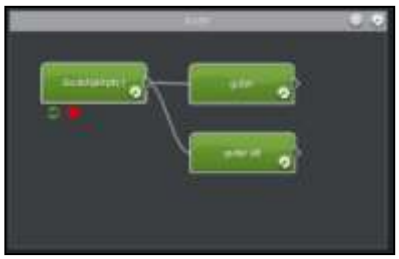


Figure 9-100: Linking segments

60 Link from the 'guitar alt' segment back to the 'switch (empty)'.

61 Link from the 'guitar' segment back to the 'switch (empty)'.

62 Click on the 'Parameters' View.

63 Press the edit button on the 'switch (empty)' segment.

64 Double click on the 'Link to guitar'.

Figure 9-101: Editing link properties

65 Drag and drop the 'variation' parameter icon onto the panel labeled 'Drop cue, theme or parameter'.
66 Set the link 'Activate when variation is' less than including 5, and hit the [return] key.
67 Double click on the 'Link to guitar alt'.
68 Drag and drop the 'variation' parameter icon onto the panel labeled 'Drop cue, theme or parameter'.
69 Set the link 'Activate when variation is' greater 5, and hit the [return] key.

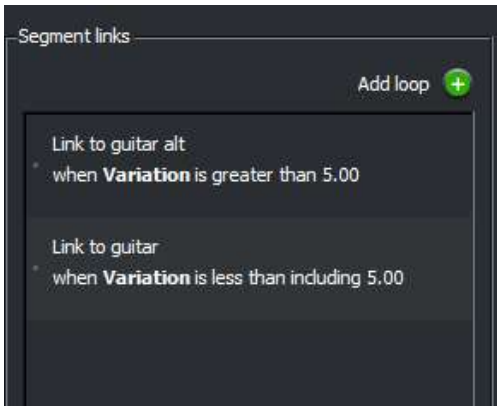


Figure 9-102: Setting conditions on a link

70 Press the 'Done' button.

Auditioning

71 To test our layered music, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.
72 Click the cue View.
73 Drag the 'drums', 'bass' and 'guitar' cue icons into the audition console. A button for each should appear.
74 Click the parameter View.
75 Drag the 'variation' parameter icon into the audition console. A fader should appear.

Figure 9-103: The audition console

76 Press the 'drums' cue button in the audition console. The drums should start immediately. You'll notice that the segment 'drums' is playing.
77 Now add the bass instrument by pressing the 'bass' cue and then the guitar by press the 'guitar' cue.
78 When ready to change the guitar score, move the fader to 10. The guitar will now sound different. You can return the guitar to its original piece by moving the fader back to 0 (or anywhere less than or equal to 5).

Conclusion

This tutorial shows how the combination of shared timelines and concurrent layers can provide a multi-track style behavior where the music is built using individual tracks. This tutorial also shows that these tracks needn't be static. Each theme (or track) can feature parameterized content and linking behavior such as 'stop paths'.

Food for thought: This tutorial only included a single tempo and time signature for all content. There is no reason why you can't explore polyrhythm with various segments (or 'tracks') containing different time signatures and tempo!

Crossfading Between Tracks with Shared Timelines

In this tutorial we will look at how 'shared timelines' can be used to maintain synchronization, as the music system switches between different themes. If you have already attempted the tutorial 'Default transition', you notice that when transitioning from one music piece to the other, the new segment always starts from the beginning of the audio file - but this can be avoided!

In this tutorial we look at how shared timelines can be used to crossfade between themes that appear to be running concurrently. This technique is really useful in maintaining a cohesive structure in the music score, while still allowing the mood to change dynamically. This technique can also be used to mimic the event system's ability to crossfade between tracks within a multi-channel audio file.

It is important to note that the audio files used in this tutorial share the same tempo and chord progression – they simply use different arrangement and rhythm to elicit different moods.

Note:

This tutorial extends the basic concepts of the 'Default transitions' tutorial. If you have already completed the 'Default transitions' tutorial, you can probably skip straight to the 'Setting up the shared timeline' section.

This score will contain two cues, two themes and two musical segments to demonstrate the use of:

- default transitions
- shared timelines.

This tutorial will also use the audition console to test the transition logic we create. You will need two audio files:

- excited.ogg
- relaxed.ogg

These files are included with Designer 2010 download.

Creating Cues and Themes

1 The first thing we need to do is create two cues – one for each mood in our music score. Create the first cue by typing the name 'relaxed' into the text field on the 'Cue View' and press the + button.
2 Click the checkbox 'Include in scene' –and press the x button with the text field to clear the search filter.
3 Add another cues called 'excited' and include it in the scene.
4 Press the + button on the 'relaxed' cue. A new theme will be created.

Figure 9-104: Adding cues

5 Rename the 'New Theme' by double clicking on the first 'New Theme' icon. Enter the new name 'relaxed' in the pop-up window provided.
6 Repeat steps 4 and 5 for the 'excited' cue using the 'excited'.
7 Click the edit button of the attached 'relaxed' theme to enter the 'theme editor' pane. You should see two empty theme containers.

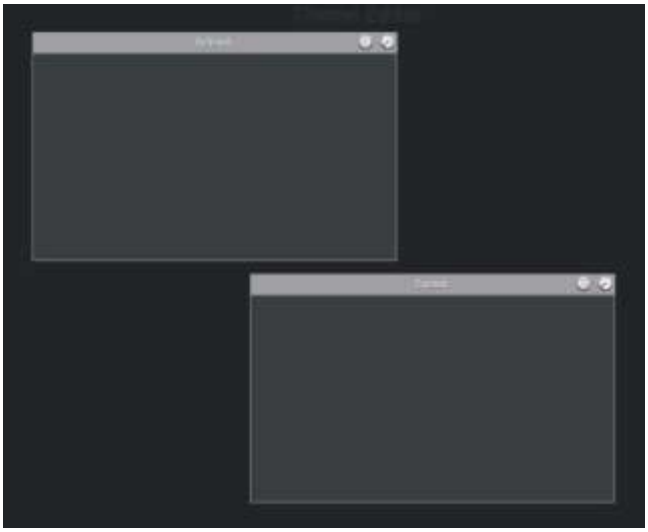


Figure 9-105: The empty themes

Setting up the Default Transitions

8 Click the edit button on the 'relaxed' theme container. The theme's properties should become visible.

Figure 9-106: Editing the theme properties

9 Set the following properties:
- Default transition: 'Crossfade'
- Crossfade duration (ms): 250
- Quantization: 'On beat'

10 Click the edit button on the 'excited' theme container. The theme's properties should become visible.

11 Set the following properties:
- Default transition: 'Crossfade'
- Crossfade duration (ms): 250
- Quantization: 'On beat'

12 Press the 'Done' button.

Setting up the Segments

13 Press 4 on the keyboard to select the 'segment stamp' tool. Click once in each theme container to create a segment for each theme. To return to the selection 'tool' press 1 on the keyboard.

Figure 9-107: Two segments in two themes

14 Each theme in the music score must have a start and stop segment. For each segment, place the mouse pointer above, until the tray appears. Click on the two icons of the left. A green arrow and red square should appear indicating that the segment will behave as a start and stop segment.

15 Press the edit button on the segment in the 'relaxed' theme and set the following properties:

- Details (Name): 'relaxed'
- Set the audio sample to: 'relaxed.ogg'
- Time signature: '4 /4'
- Beats per minute: '116'

16 Press the first and third buttons on the 'step sequence'. This means that the first and third beats of every bar are valid points to transition from.

17 Press the 'Add loop' + button.

18 Press the edit button on the segment in the 'excited' theme and set the following properties:

- Details (Name): 'excited'
- Set the audio sample to: 'excited.ogg'
- Time signature: '4 /4'
- Beats per minute: '116'

19 Press the first and third buttons on the 'step sequence'. This means that the first and third beats of every bar are valid points to transition from.

20 Press the 'Add loop' + button.

Setting up the Shared Timeline

21 When two or more segments share a timeline object, the music system will adjusted its playback to make it appear as though the files within the segment were started at the same time. This allows the composer to create a music score with layers, or in the case of this tutorial, to maintain a chord progression (harmonic structure) as we transition between different arrangements.

22 Press the edit button on the segment in the 'relaxed' theme.

23 Press the + button on the panel reading 'Drop shared timeline here'. A pop-up window should appear.

24 Type the name 'chord sync' into the text field and press the + button. A new 'shared timeline' object should be created.

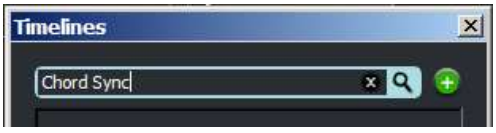


Figure 9-108: Entering the timeline name

25 Drag the 'chord sync' icon into the panel reading 'Drop shared timeline here'.

Figure 9-109: Dragging the timeline onto the segment properties

Figure 9-110: Added the timeline

26 Without closing any of the current pop-up windows press the edit button on the 'excited' segment (you might need to move the pop-up windows to see the segment on the theme editor).

27 Drag the 'chord sync' icon into the excited segment's 'Drop shared timeline here' panel.

28 Close the shared timeline dialog and press the 'Done' button.

Auditioning

29 To test our segments with shared timelines, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.

30 Click the cue View.

31 Drag the 'relaxed' cue icon into the audition console. A button should appear.

32 Repeat step 23 for the 'excited' cue.

Figure 9-111: Two cues in the console

33 Start the 'excited' theme by pressing the 'excited' cue. Playback should start immediately.

34 When ready, press the 'relaxed' cue. The music system should immediately begin to fade to the 'relaxed' theme. The crossfade will occur on the first or third beat of the bar.

35 Quickly transition back to the 'excited' theme by ending the 'relaxed' cue. Listen carefully to the transition. It might be worthwhile repeating steps 34 and 35 a number of times.

Note how the music system does not return to the starting position of the audio files, instead maintains its current position in the 'shared timeline'.

This tutorial has demonstrated one use of 'shared timelines'. We created responsive, beat matched crossfade between two pieces of music, whilst maintaining the overall harmonic structure (in this case a simple I-V-IV chord progression).

Creating Early Transition Points

In this tutorial we will look at how the early transition feature can make an interactive music score more responsive to changes in game state. This tutorial improves upon a previous tutorial, namely 'Choosing a link via a parameter', which uses a parameter to determine the sequence of links that are followed. The 'Choosing a link via a parameter' tutorial in its original form requires that the audio samples be played in their entirety before transitioning to the next. If the audio samples are long, this technique can cause latency when responding to game state. Sure, you can cut the segment audio into smaller bits and re-assemble them using a segment playlist, however this isn't a great solution because:

1. It isn't a great use of time cutting up the file into smaller files.
2. You lose the ability to use the playlist creatively for randomization

This score will contain a shared timeline, one cue, one parameter, one theme and three segments to demonstrate the use of:

- early transitions
- shared timelines

It is recommended that you attempt the tutorial 'Choosing a link via a paramete'r to understand the following:

- segment linking
- parameters
- linking

You will need three audio files:

- interactive_music_1.ogg
- interactive_music_2.ogg
- interactive_music_3.ogg

These files are included with Designer 2010 download.

Creating a Cue, Parameter and Themes

1 The first thing we need to do is create a cue. Create the cue by typing the name 'music' into the text field on the 'Cue View' and press the + button.

2 Click the checkbox 'Include in scene' – and press the x button with the text field to clear the search filter.

Figure 9-112: Adding a cue

3 Click the parameter View.

4 Create a parameter by typing the name 'intensity' into the text field on the 'Parameter View' and press the + button.

Figure 9-113: Setting up the parameter

5 Press the + button on the 'music' cue icon (on the scene editor). A new theme will be created.

6 Rename the 'New Theme' by double clicking on the 'New Theme' icon. Enter the new name 'intensity theme' in the pop-up window provided.

7 Enter the 'theme editor' pane by pressing either the theme edit button on the 'intensity theme' icon or using the center (looks like a crosshair) button on the theme icon within theme list.

Creating the Segments

8 Open the folder that contains the audio files, using your operating system.

9 Select the three audio files (interactive_music_1.ogg, interactive_music_2.ogg and interactive_music_3.ogg) and drag them into the empty theme container in the Designer 2010 application. This should create three new segments, each containing a file.

You should see:

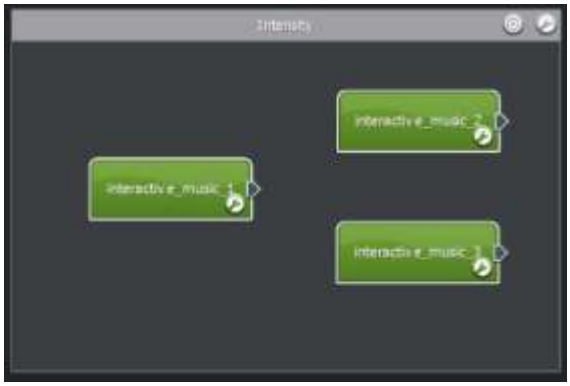


Figure 9-114: Creating and positioning the segments

10 For each segment, press the edit button and set the 'Beats Per Minute' property to 135.

11 Grab the segment plug of 'interactive_music_01' and connect it to 'interactive_music_2'.
12 Grab the segment plug of 'interactive_music_01' and connect it to 'interactive_music_3'.

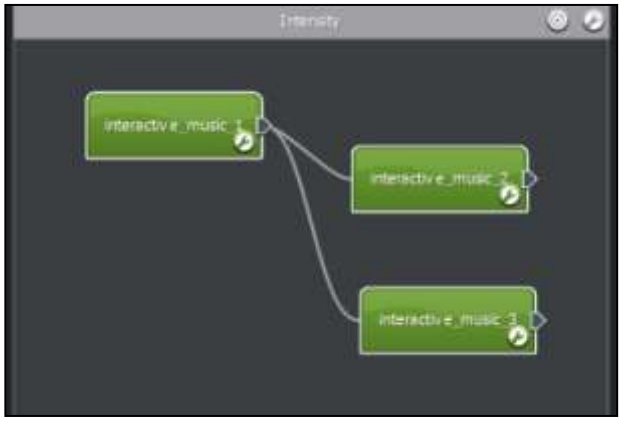


Figure 9-115: Links from 'interactive_music_01'

13 Grab the segment plug of 'interactive_music_02' and connect it to 'interactive_music_1'.
14 Grab the segment plug of 'interactive_music_02' and connect it to 'interactive_music_3'.

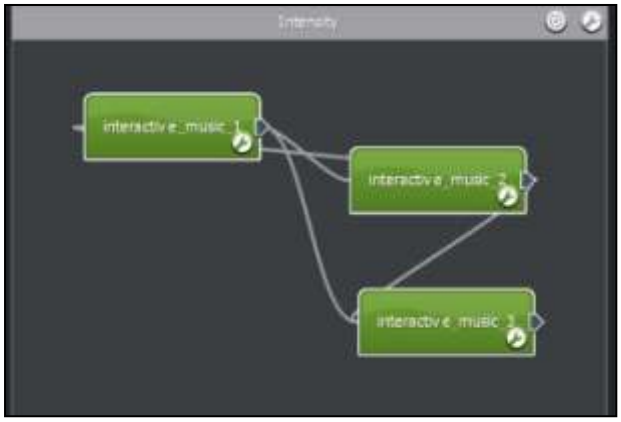


Figure 9-116: Links from 'interactive_music_02'

15 Grab the segment plug of 'interactive_music_03' and connect it to 'interactive_music_1'.
16 Grab the segment plug of 'interactive_music_03' and connect it to 'interactive_music_2'.
17 We will specify that the music score always starts with the lowest level of intensity (interactive_music_01). Specify the segment 'interactive_music_01' as a starting point by placing the mouse over the segment and then clicking the green arrow icon from the tray that appears.
18 Make sure all segments are designated as a stop segment by placing the mouse over each segment and then clicking the red square icon from the tray that appears.

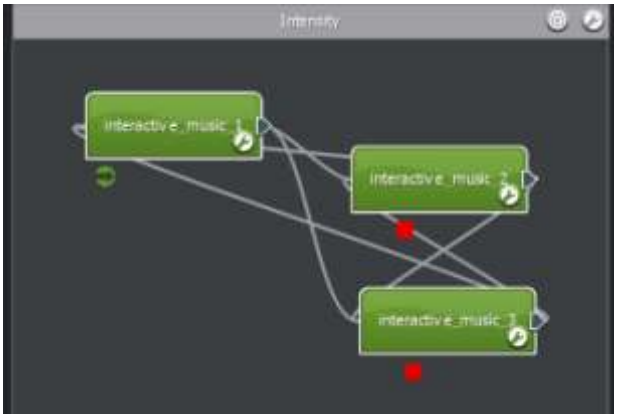


Figure 9-117: Theme construction continues

Using the Parameter

19 At this point our music score has the cue, all the music and links – however we need to attach conditions to the links using the 'intensity' parameter. In this example, will divide our parameter space of 0 to 10 as:
- From 0 to 4 will represent low intensity and play 'interactive_music_1'.
- Between 4 and 8 will represent regular intensity and play 'interactive_music_2'
- From 8 to 10 will represent high intensity and play 'interactive_music_3'
- Press the edit button on the 'interactive_music_1' segment.

20 Double click the link item 'Link to interactive_music_2'. A dialog will appear.

21 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.

22 Set 'Activate when intensity is' between 4 and 8.

Figure9-118: Setting up the link conditions

23 Close the pop-up window by pressing the x button.

24 Double click the link item 'Link to interactive_music_3'. A dialog will appear.

25 Drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.

26 Set 'Activate when intensity is' greater than including 8.

27 Close the pop-up window by pressing the x button.

28 We now need to tell the music system when the links can be followed. By default, the music system waits until a sample has finished playback but this can lead to latency and a lack of responsiveness (as the system waits for the segment to finish). Instead we will specify that links be evaluated on each beat selected in the segment's step sequencer. As the links are evaluated more often, the music system will appear more responsive.

Continue by selecting all beats in the step sequencer.

29 Press the 'Add loop' + button and make sure the loop link is at the bottom of the Segment links list.
30 Double click the link item 'Loop'. A dialog will appear. Click the 'Custom' radio button and leave the 'At Segment End' checkbox as checked. This means a loop will only occur once the sample has completed playback (and not on every beat!).
31 In the 'Link Transition Defaults' section, turn of 'At Segment End' and select 'On Beat'. All links within this segment are now evaluated on every beat, and not at the end of the sample playback. You can customize this behavior further, by changing which beats are active in the step sequencer.

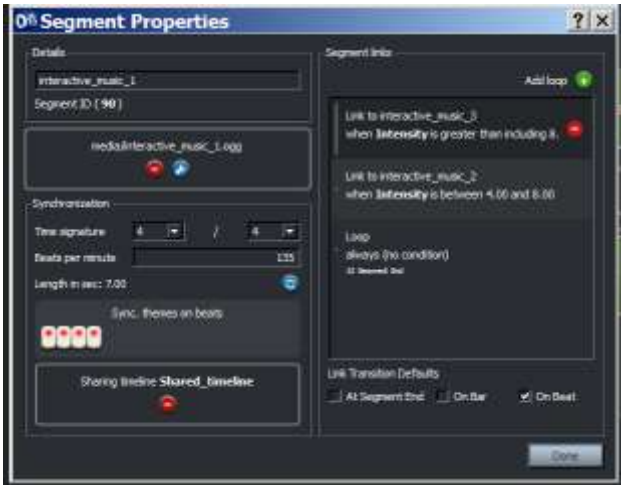


Figure 9-119: Setting up the first segment

32 Press the edit button on the 'interactive_music_2' segment.
33 Double click the link item 'Link to interactive_music_1'. A dialog will appear.
34 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
35 Set 'Activate when intensity is' less than including 4.
36 Close the pop-up window pressing the x button.
37 Now double click the link item 'Link to interactive_music_3'. A dialog will appear.

38 Drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
39 Set 'Activate when intensity is' greater than including 8.
40 Close the pop-up window pressing the x button.
41 Select all beats in the step sequencer.
42 Press the 'Add loop' + button and make sure the loop link is at the bottom of the Segment links list.
43 Double click the link item 'Loop'. A dialog will appear. Click the 'Custom' radio button and leave the 'At Segment End' checkbox as checked.
44 In the 'Link Transition Defaults' section, turn of 'At Segment End' and select 'On Beat'.

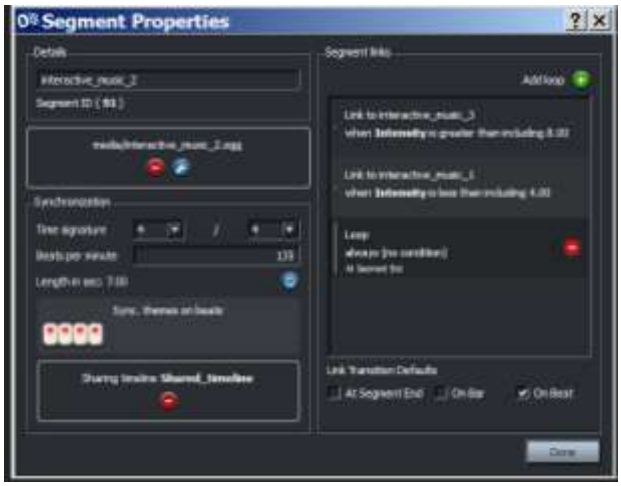


Figure 9-120: Setting up the second segment

45 Press the edit button on the 'interactive_music_3' segment.
46 Now double click the link item 'Link to interactive_music_1'. A dialog will appear.
47 Click the 'parameters View' and drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
48 Set 'Activate when intensity is' less than including 4.
49 Close the pop-up window pressing the x button.
50 Now double click the link item 'Link to interactive_music_2'. A dialog will appear.
51 Drag the 'intensity' parameter into the panel labeled 'Drag on cue, theme or parameter'. Some condition operators should appear.
52 Set 'Activate when 'intensity' is' between 4 and 8.
53 Close the pop-up window by pressing the x button.
54 Select all beats in the step sequencer.
55 Press the 'Add loop' + button and make sure the loop link is at the bottom of the Segment links list.

56 Double click the link item 'Loop'. A dialog will appear. Click the 'Custom' radio button and leave the 'At Segment End' checkbox as checked.

57 In the 'Link Transition Defaults' section, turn of 'At Segment End' and select 'On Beat'.

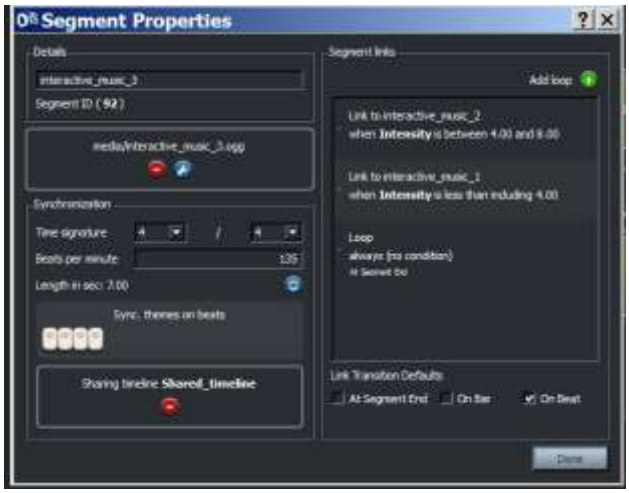


Figure 9-121: Setting up the third segment

58 Press the 'Done' button.

Auditioning

At this point we shall start auditioning by testing our link logic.

59 To test our link transitions, we need to use the audition console. Drag the audition console bar upwards, so it becomes visible.

60 Click the cue View.

61 Drag the 'music' cue icon into the audition console. A button should appear.

62 Click the parameter View.

63 Drag the 'intensity' parameter icon into the audition console. A fader control should appear.

Figure 9-122: Setting up the audition console

64 Press the 'music' cue button in the audition console. The score should start immediately. You'll notice that the segment 'interactive_music_1' is playing.

This is because the 'intensity' audition fader is still on 0.

65 Move the fader up until you reach around 5. You'll notice that the music system transitions to 'interactive_music_2' almost immediately, on beat.
66 Move the fader up to 10. Once again you'll notice that the music system transitions 'interactive_music_3' almost immediately, on beat.

When finished auditioning, end the active cue. You might have noticed, something not quite musical is happening! Each time the music system transitions from beat 2,3 or 4 the segment starts the audio sample from the beginning, which sounds a little odd (as the bar length changes unexpectedly). To remedy this we shall add a 'shared timeline' to each segment, this way the playback position (within the bar) will remain consistent as we switch between segments, giving us a more musical transition.

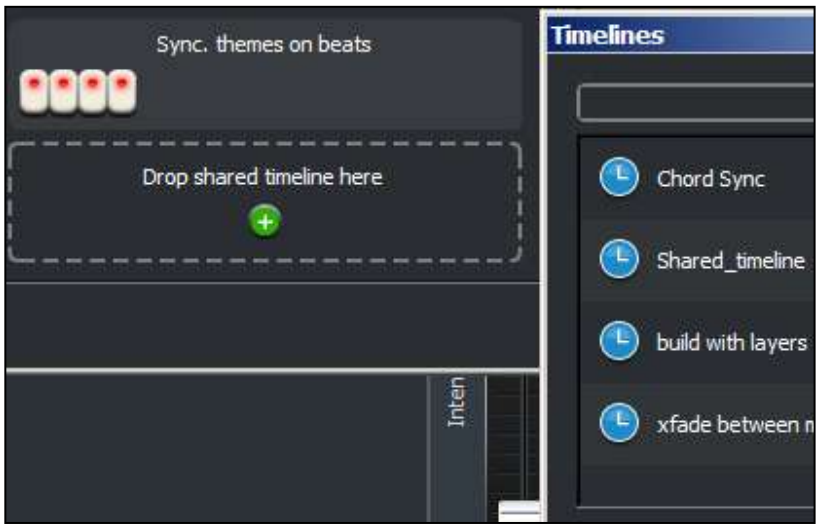


Figure 9-123: Setting up the shared timeline

67 Click the edit button on the segment 'interactive_music_1'.
68 Press the + button on the 'Drop shared timeline here' panel. A pop-up window should appear.
69 Enter the name 'sync' into the text field and press +. A timeline object called 'sync' will be created.
70 Drag and drop the 'sync' object onto the 'Drop shared timeline here' panel. The panel should now display 'Sharing timeline sync'. Do not close the Timelines window.
71 Click the edit button on the segment 'interactive_music_2'.
72 Drag and drop the 'sync' object onto the 'Drop shared timeline here' panel. The panel should now display 'Sharing timeline sync'. Do not close the Timelines window.
73 Click the edit button on the segment 'interactive_music_3'.

74 Drag and drop the 'sync' object onto the 'Drop shared timeline here' panel. The panel should now display 'Sharing timeline sync'.
75 Close the Timelines window.
76 Press the Done button.
77 Repeat the auditioning steps 64 to 66. You'll now notice that the position in the bar is maintained through the transition.

Conclusion

The variation between the segments is a change in arrangement (rendered into each sample). Using the early transition points the music system can respond quickly (each beat) to changes in the parameter. The shared timeline keeps the chord progression and rhythm consistent through the transitions. The end result is more responsive, musical and you don't need to spend time chopping up samples!

Early transitions are a really useful feature, allowing you to work with big samples while retaining responsiveness. This feature can also be used to end current themes early too - but that is another tutorial!

Note:

In this tutorial we built the score with multiple links from each segment. This can cause extra work to maintain the links should you want to change the ranges within the parameter space. This problem can be avoided by using an empty segment that operates as a logic switch, as shown in the 'Choosing a link via a parameter'.

Appendix

Frequently Asked Questions

This section is intended to answer the questions that are frequently asked on the FMOD Designer 2010 support forum, and provide references to the appropriate sections in this user manual.

Q. I have a question that is not answered in the documentation. What should I do?

A. Try the FMOD Discussion Forums (http://www.fmod.org/forum).

Q. My project (.fdp) won't load any more. How can I fix it?

A. Before FMOD Designer 2010 saves a '.fdp' file, it first creates a backup of the original with a '.bak' extension. Try renaming this '.bak' file to '.fdp' and you may be able to load the project again. Some changes may have been lost.

Q. Why doesn't the timeline work in the Event Editor?

A. The simplest answer is ‘It's NOT a timeline, it is a parameter bar!’

While the parameter bar looks very much like a 'timeline'. In its default setting the bar has no relationship to the time domain. Instead, the bar simply represents the current value of a parameter. Parameters are typically used to represent variables inside the games engine such as engine RPM or the distance between the listener and the location of the sound emitter.

The parameter can be used as a pseudo-timeline using the parameter's velocity property. This technique is demonstrated in Chapter Five, Tutorial Projects.

Q. I'm experiencing clicking during playback. How can I stop it?

A. There are a number of reasons that the audio contains clicking artifacts, the most common reason is a buffer size that is too small for the computer to keep up with the real-time audio processing.

To increase the buffer size:

1. Select 'Edit' from the Menu Bar. A sub-menu should appear.
2. Select the option 'Preferences…'. A dialog should appear.
3. Select the category 'sound' from the category list. The sound preferences should appear in the right panel.
4. Increase the buffer size 10 ms by moving the 'Buffer size' slider to the right.
5. Test the audio playback. If the clicking is still audible, repeat step 4 until clicking disappears.

If the clicking persists, try the FMOD Designer 2010 forum (http://www.fmod.org/forum).

Q. How do I create a custom distance parameter?

A. See the tutorial 'Simulating Distance' in Chapter 5, Advanced Projects.

Q. Do I really need to use the music category?

A. Developers of titles for the Microsoft Xbox 360, have a TRC requirement that states titles must allow the non-game specific audio (i.e. soundtrack) be replaced by the music specified in the end user's dash board.

The easiest way to comply with this requirement is to add all soundtrack events to the 'Music' category (shown under the Category tree on the Event View).

Q. How do I create separate builds for separate platforms?

A. A different build directory can be set for each target platform. The target platform is selected using the 'Platform' dropdown list. For information, see

- Chapter 3, section 'Selecting a target platform', page 26.
- Chapter 3, section 'Building a project', page 31.

Q. What should I set the 'Max playbacks' property to?

A. Each event's 'Max playbacks' property should be set as low as possible, as it directly influences the amount of memory used by the FMOD Ex engine.

If the 'Max playbacks' is set to 1, FMOD Ex declares enough memory to play one instance of the event.

If the 'Max playbacks' is set to 3, FMOD Ex declares enough memory to play three instances of the event simultaneously.

Setting the 'Max playbacks' correctly requires the balancing of the 'artist vision' (or creating a realistic and immersive soundscape) and the memory resources required to play the all events.

See page 163 for more information.

Q. How do the event templates work?

A. Templates are discussed in Chapter 3, in the section 'Event Templates' on page 61.

Q. Why does the event template keep overwriting my event properties?

A. If an event property belonging to the template is modified, this modification will be carried through to all events that use the template when the 'Apply template changes' function is executed

When an event template is created the sound designer selects the properties to be included in the template. It is important to note these properties will be INDENTICAL in ALL events that use the template. If an event property needs to be different to that of the template:
The property should not be included in the template, or the event should not use a template.

Q. How do I delete unused media from the wave bank?

A. Before building a project it is good practice to remove all unused media. FMOD Designer 2010 has a function to do this automatically, called 'Clean project'. To use this simply:

Select 'Edit' from the Menu Bar. A sub-menu should appear.
Select the option 'Clean project'. A confirmation menu should appear.
Press the 'Yes' button.
All used Sound Defs and Banks have been deleted.

Cleaning a project is discussed in Chapter Three; in the section 'Cleaning the project' on page **Error! Bookmark not defined.**.

Q. Why does .mp2 and .mp3 compression produce the same file size? I thought .mp3 compressed files more.

A. Both .mp2 and .mp3 are variable bit rate compression formats. Setting the data rate actually specifies how much data is used for each time unit of audio. Therefore, if the same date rate is used for each format, both will produce the same file size.

   The advantage of .mp3 however, is the quality of the audio. The quality of an .mp3 file will be greater than that of an .mp2 file using the same date rate. This means the sound designers can use greater compression levels when using .mp3 and still achieve the same quality of larger .mp2 files.

Short-Cut Guide

The Event View

Each of the following shortcuts can be applied to the event, event group in the event hierarchy tree or event categories.

Target	Keyboard / Mouse	Description
Event	[ALT] + Left-Click	Previews the event.
Event, event group or event category	[DEL]	Deletes the event, event group or event category.
Event, event group or event category	[CTRL] + c	Copies the event, event group or event category
Event, event group or event category	[CTRL] + x	Cuts the event, event group or event category
Event, event group or event category	[CTRL] + v	Pastes the event, event group or event category
Event, event group or event category	Click…[SHIFT] + Click	Multiple event selection
Event, event group or event category	[CTRL] + Click	Toggle event selection
Event Hierarchy Tree	'Drag and drop' audio files from an explorer directory into the Event Hierarchy Tree.	Automatically creates an event and Sound Def for each audio file.
Event	Double-Click	Opens the event in the Event Editor View

The Event Editor View

Each of the following shortcuts relate to the Event Editor View.

Target	Keyboard / Mouse	Description
[none]	[SPACE]	Starts / Stops the playback of the event
Sound Def Instance	[DEL]	Deletes the Sound Def instances.
Sound Def Instance	[CTRL] + Double-Click	Toggles the size of the Sound Def instance between all and half the available space.
Effect property line	[SHIFT] + Left-Click	Adds a point
Effect property point	[SHIFT] + Double-Click	Deletes the selected point
Effect property point	Hold down Z while moving mouse	Slows down the cursor speed when moving a point. Useful for fine-tuning.
Effect property point	Hold down [CTRL] while moving mouse	Locks the x-axis allowing the point position to move vertically.
	[←]	Moves the parameter cursor left
	[→]	Moves the parameter cursor right
	[CTRL] + [←]	Moves the parameter cursor left at a greater speed
	[CTRL] + [→]	Moves the parameter cursor right at a greater speed

The Sound Def View

Each of the following shortcuts can be applied to the Sound Def and, groups in the Sound Def list.

Target	Keyboard / Mouse	Description
Sound Def or group	[DEL]	Deletes the Sound Def, or group
Sound Def or group	[CTRL] + c	Copies the Sound Def or group
Sound Def, or group	[CTRL] + x	Cuts the Sound Def, or group
Sound Def or group	[CTRL] + v	Pastes the Sound Def, or group
Sound Def or group	Click…[SHIFT] + Click	Multiple Sound Def selection
Sound Def or group	[CTRL] + Click	Toggle Sound Def selection
Sound Def List	'Drag and drop' audio files from an explorer directory into the Sound Def List	Automatically creates a Sound Def for each audio file.
Sound Def	Double-Click	Shows the location of the Sound Def in the Banks View

The Banks View

Each of the following shortcuts can be used in the Sound File pane.

Target	Keyboard / Mouse	Description
Bank	[DEL]	Deletes the bank (if empty)
Bank	Click…[SHIFT] + Click	Multiple bank selection
Bank	[CTRL] + Click	Toggle bank selection
Sound File	Double-Click	Shows the Sound Def where the Sound File is located.

Command Line Usage

The command line version of FMOD Designer 2010 (fmod_designercl.exe) can be used to build projects or extract information from project using the command line (Window32). This facilitates the use of FMOD Designer 2010 in offline build processes and scripts etc.

Usage

fmod_designer <-pc|xbox|xbox360|gc|ps2|psp|ps3|wii> [-a|b|h|m|n|p|s|c] <project.fdp>

Target platform

The following switches can be used to select the target platforms:

-pc	PC
-xbox	Microsoft Xbox
-xbox360	Microsoft Xbox 360
-gc	Nintendo Gamecube
-ps2	Sony PlayStation 2
-psp	Sony PSP
-ps3	Sony PlayStation 3
-wii	Nintendo Wii

Switches

The following option switches can be used:

-a <pathname>	Alternate audio source directory
-b <pathname>	Alternate build directory
-h	Generate header file
-m	Display input/output files only (doesn't build .FEV or .FSB)
-n	Don't build .FEV or .FSB files
-p	Generate programmer report
-l	Generate a listing of files within each FSB
-s 0	Only build streaming FSB wavebanks, not static sample.
-s 1	Only build static sample FSB wavebanks, not streaming.
-s 2	Don't build any FSB wavebanks.
-t <pathname>	Specify the template file to use.
-T	Apply templates to events before building.

-c <format>	Override the compression format used for all wavebanks
-L	Selects which language to build.
-C	Clean the build (removes unused audio data from wavebanks).
-r	Rebuild (this option deletes all cache files before building).

Compression Format

The following switches can be used to select the output compression format:

-pcm
-vag
-gcadpcm
-xadpcm
-source
-adpcm
-xma
-mp3
-mp2

Examples

1. Build foobar.fdp for PC:

```
fmod_designercl -pc foobar.fdp
```

2. Build a programmer report only:

```
fmod_designercl -n -p foobar.fdp
```

3. Display dependencies only:

```
fmod_designercl -m foobar.fdp
```

4. Only build streaming Banks for foobar.fdp, and use alternate wave source directory c:\temp:

```
fmod_designercl -a C:\temp -s 0 foobar.fdp
```

Contact Details

The FMOD team can be contacted via the following channels:

- FMOD Website: http://www.fmod.org/
- FMOD Discussion Forums: http://www.fmod.org/forum
- FMOD Wiki: http://www.fmod.org/wiki
- Email support: support@fmod.org
- Sales inquiries: sales@fmod.org
- Education/teaching enquiries: education@fmod.org