

audio**k**inetic

# Wwise

ユーザーガイド

2017.2.4 編集



# Wwise

## ユーザーガイド

Wwise 2017.2.4 Revision 3176

製作著作 © 2018 Audiokinetic Inc.

This document (whether in written, graphic or video form) is supplied as a guide for the Wwise® product. This documentation is the property of Audiokinetic Inc. (“Audiokinetic”), and protected by Canadian copyright law and in other jurisdictions by virtue of international copyright treaties. It may be used by you in accordance with the following.

This documentation may be duplicated, reproduced, stored or transmitted, exclusively for your internal, non-commercial purposes, but you may not alter the content of any portion of the documentation. Any copy of the documentation shall retain all copyright and other proprietary notices contained therein.

The content of this documentation is furnished for information purposes only, and its content is subject to change without notice. Reasonable care has been taken in preparing the information contained in this document, however, we disclaim all representations, warranties and conditions, whether express, implied or arising out of usage of trade or course of dealing, concerning this documentation and assume no responsibility or liability for any losses or damages of any kind arising out of the use of this guide or of any error or inaccuracy it may contain, even if we have been advised of the possibility of such loss or damage.

Wwise®, Audiokinetic®, Actor-Mixer®, SoundFrame® and SoundSeed® are registered trademarks, and Master-Mixer™, SoundCaster™ and Randomizer™ are trademarks, of Audiokinetic. Other trademarks, trade names or company names referenced herein may be the property of their respective owners.

---

# 目次

Wwise 2017.2.4へようこそ .....	xi
I. スタート前の準備 .....	1
1. Wwiseのロードマップ .....	3
スタート前の準備 .....	4
Wwiseロードマップの紹介 .....	4
ドキュメントの表記方法 .....	4
Wwise Help .....	5
サポートへの連絡 .....	7
2. Wwiseを使い始める .....	8
Wwiseの起動と終了 .....	9
Wwiseのインターフェースについて .....	10
Wwiseのインターフェースの基礎 .....	12
ワークスペースのユーザー設定 .....	20
作業スピードを速める方法 .....	34
II. プロジェクトをセットアップする .....	46
3. Wwiseのプロジェクトについて .....	48
概要 .....	49
プロジェクトの管理 .....	51
Project Settingsの設定 .....	55
プロジェクトのデフォルトユーザー設定 .....	80
プロジェクトのトラブルシューティング .....	82
学習コーナー: オブストラクションとオクルージョンについて .....	88
プロジェクトマネジメントに関する、コツとベストプラクティス .....	90
4. プラットフォームの管理 .....	91
概要 .....	92
Platform Managerを使用する .....	92
5. ランゲージの管理 .....	96
概要 .....	97
Language Managerを使う .....	97
6. Workgroupの活用 .....	102
概要 .....	103
プロジェクトをWork Unitで分割 .....	106
プロジェクトファイルのステータス確認 .....	114
Wwiseとソースコントロールシステムを使う .....	116
プロジェクトの不整合の解消 .....	116
Workgroupプラグインを使ったプロジェクトファイルの管理 .....	118
Workgroupに関する、コツとベストプラクティス .....	131
7. メディアファイルの活用 .....	134
概要 .....	135
インポート処理 .....	135
メディアファイルのインポート .....	143
メディアファイルの置換 .....	160
ファイルインポート関連の問題への対処 .....	162
Originalsフォルダのメディアファイルの並べ替え .....	163
キャッシュクリア .....	164
外部エディタでオーディオファイルを編集 .....	166

---

プラグインを使ってソースを作成する .....	167
メディアファイル管理に関する、コツとベストプラクティス .....	168
8. Actor-Mixer Hierarchyの構築 .....	169
概要 .....	170
プロジェクト階層内のプロパティについて .....	174
Actor-Mixer Hierarchyの構築 .....	182
ランダム化したプロパティ値で、再生を強化 .....	187
Actor-Mixer Hierarchyに関する、コツとベストプラクティス .....	188
9. アウトプットバス構造の構築 .....	193
概要 .....	194
バスのプロパティ設定 .....	198
バス階層の構築例 .....	202
III. サウンドやモーションでゲームを強化する .....	205
10. オブジェクトの再生方法の設定 .....	207
概要 .....	208
オブジェクトの再生方法の設定 .....	208
Random Container、Sequence Containerの再生方法の設定 .....	213
Switch Containerの中身と、再生方法の設定 .....	225
Blend Containerの中身と、再生方法の設定 .....	230
オブジェクト再生に関する、コツとベストプラクティス .....	241
11. Positioningの定義 .....	243
Positioning定義の概要 .....	244
WwiseのPositioningを理解する .....	245
2Dオブジェクトの活用 .....	247
3Dオブジェクトの活用 .....	249
BusにPositioningを適用 .....	252
3Dポジショニングのイメージ図 .....	253
距離で変化する減衰設定 .....	265
アニメーションパスを利用した、空間ポジショニング .....	278
センタースピーカーの、オーディオ信号のルーティング .....	289
Positioningに関する、コツとベストプラクティス .....	289
チャンネルコンフィギュレーションを理解する .....	296
スピーカーとヘッドフォンのパンニングルール .....	298
Ambisonicsを使う .....	300
12. Priorityの管理 .....	306
概要 .....	307
オブジェクトのプライオリティを理解する .....	309
オブジェクトの再生インスタンスの制限 .....	312
再生Priorityの設定 .....	316
低ボリュームのオブジェクトの管理 .....	318
プライオリティに関する、コツとベストプラクティス .....	320
13. エフェクトの活用 .....	322
概要 .....	323
エフェクトの使用 .....	323
環境音響を実現する、エフェクトの使用 .....	331
エフェクトに関する、コツとベストプラクティス .....	335
14. モーションの活用 .....	336
概要 .....	337
ゲームのモーションを作成 .....	339

---

---

モーション用に、アウトプット構造の構築 .....	345
モーションに関する、コツとベストプラクティス .....	346
IV. ゲームとのインテラクション .....	348
15. イベントの管理 .....	350
概要 .....	351
イベントの作成 .....	355
イベントの使い方 .....	362
イベントに関する、コツとベストプラクティス .....	365
16. ダイナミックダイアログの管理 .....	367
概要 .....	368
ダイナミックダイアログシステムを理解する .....	368
ダイアログイベントの作成 .....	372
ダイアログイベントの活用 .....	382
ダイアログイベントに関する、コツとベストプラクティス .....	386
17. Stateの活用 .....	388
概要 .....	389
Stateの活用 .....	390
同じStateやState Groupの、State間のトランジションを設定する ...	393
オブジェクトやバスにStateをアサインする .....	394
Stateに関する、コツとベストプラクティス .....	399
18. Switchの使い方 .....	400
概要 .....	401
Switchの使い方 .....	402
Switchに、Game Parameter値をマッピングする .....	405
Switchに関する、コツとベストプラクティス .....	405
19. RTPCの使い方 .....	407
概要 .....	408
RTPCで使う、Game Parameterの管理 .....	409
Game Parameterを使って、プロパティ値をコントロールする .....	414
LFOの活用 .....	421
エンベロープの活用 .....	422
ゲームオブジェクトの観察 .....	424
RTPCに関する、コツとベストプラクティス .....	425
20. Triggerの使い方 .....	427
概要 .....	428
Triggerの使い方 .....	428
21. ダイナミックダイアログの、StateやState Groupの活用 .....	431
概要 .....	432
StateやState Groupの活用 .....	433
V. インタラクティブミュージックを作成する .....	436
22. インタラクティブミュージックを理解する .....	439
概要 .....	440
インタラクティブミュージックを理解する .....	440
インタラクティブミュージックに関する、コツとベストプラクティス .....	442
23. Interactive Music Hierarchyの構築 .....	445
概要 .....	446
Music Segmentとは .....	446
コンテナの種類 .....	448

---

---

Interactive Music Hierarchyの、プロパティについて .....	449
Interactive Music Hierarchyに、オブジェクトを追加する .....	450
親オブジェクトの追加 .....	452
Interactive Music Hierarchyの、ミュージックオブジェクトの管理 .....	453
Interactive Music Hierarchyの構築に関する、コツとベストプラクティス .....	454
24. ミュージックオブジェクトの再生方法の設定 .....	455
概要 .....	456
ミュージックオブジェクトの時間設定 .....	456
Music Playlist Containerの、再生方法の設定 .....	457
Music Switch Containerの中身と、再生方法の設定 .....	460
インタラクティブミュージックの再生に関する、コツとベストプラクティス .....	464
25. Music Track、Music Segmentの活用 .....	466
概要 .....	467
セグメントに、Music Trackを追加する .....	468
Music Trackの再生方法の設定 .....	469
トラックに、サブトラックを追加する .....	472
サブトラックをSwitchやStateに結びつける .....	472
トラックの中身を入れる .....	473
セグメントから、トラックやサブトラックを削除する .....	474
Music Editorの、目盛に合わせたスナップ機能 .....	474
クリップの使い方 .....	475
セグメントの試聴 .....	477
キーの使い方 .....	479
26. MIDIの活用 .....	483
MIDIコンテンツを作成する .....	484
MIDIファイルのインポート .....	484
MIDIコンテンツとMIDIターゲットを理解する .....	484
MIDIコンテンツとオーディオコンテンツをミックスする .....	486
MIDIテンポを理解する .....	486
MIDIの再生スピードを変更する .....	487
27. MIDIインストゥルメントの作成 .....	488
Synth Oneインストゥルメントをデザインする .....	489
シンプルなMIDIサンプルインストゥルメントをデザインする .....	489
MIDIノートトラッキングを理解する .....	490
MIDIフィルタを理解する .....	490
MIDIイベントを理解する .....	491
MIDIイベントにフェードイン、フェードアウトを追加する .....	492
MIDIデータを使ってオブジェクトのプロパティ値をコントロールする .....	492
MIDI Keymap Editorを使う .....	492
MIDIキーボードでインストゥルメントをテストする .....	493
MIDIをDAWからWwiseにルーティングする .....	494
28. トランジションの活用 .....	495
概要 .....	496
トランジションを理解する .....	497
トランジションの追加 .....	498

---

---

トランジションのコピー、ペースト .....	499
トランジションの削除 .....	500
ソースやデスティネーションのプロパティを設定する .....	500
トランジションセグメントの使用 .....	504
インターラクティブミュージックのトランジションに関する、コツと ベストプラクティス .....	505
29. スティンガーの活用 .....	507
概要 .....	508
スティンガー (Stinger) の追加 .....	509
スティンガー (Stinger) の再生方法の設定 .....	511
スティンガーの削除 .....	512
スティンガー (Stinger) の試聴 .....	512
VI. プロジェクトを仕上げる .....	515
30. アウトプットの管理 .....	518
概要 .....	519
オブジェクトのアウトプットルーティングの指定 .....	519
User-Defined Auxiliary Send (ユーザーが定義するAUXセンド) を使う .....	521
Game-Defined Auxiliary Sendを使う .....	522
ラウドネスノーマライゼーション、またはメイクアップゲインを使 い、ボリュームを調整する .....	523
ボイスパイプラインを理解する .....	524
組み込みオーディオデバイス .....	526
セカンダリアウトプットを理解する .....	528
HDRを理解する .....	533
HDRの使用 .....	537
HDRの追加情報 .....	546
ファイナルミックスの作成 .....	550
31. プラットフォームとランゲージのバージョン管理 .....	567
概要 .....	568
複数のプラットフォームに対応するオーサリング .....	568
プロジェクトのローカリゼーション作業 .....	589
バージョンに関する、コツとベストプラクティス .....	594
32. シミュレーションの作成 .....	602
概要 .....	603
シミュレーションを構築する .....	604
シミュレーション中の再生の管理 .....	607
ゲームシンクを使ったシミュレーション .....	610
シミュレーションを使った、プロパティの最終調整 .....	613
シミュレーション作成に関する、コツとベストプラクティス .....	616
33. Wwiseにおけるメモリ管理 .....	618
概要 .....	619
Memory Managerの各コンポーネントについて .....	619
各種メモリプールのサイズを設定する .....	620
メモリ関連のトラブルシューティング .....	622
メモリプールの最適化 .....	623
メモリ管理に関する、コツとベストプラクティス .....	628
34. Profiling .....	630
概要 .....	631

---

Wwiseにおける、プロファイリングの種類を理解する .....	632
ローカルまたはリモートで、ゲーム機に接続する .....	637
サウンドエンジンからくるデータをキャプチャーする .....	640
Performance Monitorを使った、モニタリングとトラブルシューティング .....	649
Game Object Explorerで、オブジェクトやリスナーをトラッキングする .....	652
Game Object 3D Viewerで、オブジェクトを観察する .....	656
Game Sync Monitorで、ゲームシンクを評価する .....	660
プロファイリングに関する、コツとベストプラクティス .....	661
35. SoundBankの管理 .....	664
概要 .....	665
ゲームにSoundBankをロードする方法を理解する .....	667
SoundBankの構築 .....	672
SoundBankの管理 .....	689
SoundBankのカスタム属性を設定する .....	692
プロジェクトの各種SoundBankの生成 .....	700
CopyStreamedFilesツールの利用 .....	704
SoundBank管理の戦略 .....	705
SoundBankに関する、コツとベストプラクティス .....	718
36. ファイルパッケージの管理 .....	722
概要 .....	723
File Packagerプロジェクトの使い方 .....	723
プロジェクト内のファイルパッケージの管理 .....	726
DLC（ダウンローダブルコンテンツ）の概要 .....	730
ファイルパッケージを生成する .....	731
コマンドラインで、File Packager引数を使用 .....	732
ファイルパッケージャに関する、コツとベストプラクティス .....	735
VII. Wwiseを使いこなす .....	737
37. Project Explorer画面について .....	740
概要 .....	741
Project Explorerの記号や色文字 .....	743
Project Explorerで作業する .....	744
38. Event Viewer画面について .....	747
概要 .....	748
Event Viewerで作業する .....	748
39. Property Editor画面を使う .....	753
概要 .....	754
Property Editorを使って作業する .....	770
40. Contents Editor画面について .....	774
概要 .....	775
Contents Editorで作業する .....	782
41. Transport Control画面について .....	791
概要 .....	792
再生方法の設定 .....	794
Transport Controlでオブジェクトをピン付けする .....	797
コンテンツの再生、一時停止、停止 .....	798
再生中にゲームシンクを使う .....	798
42. Schematic View画面について .....	803

---

---

概要 .....	804
Schematic Viewのカスタマイズ .....	804
Schematic Viewを活用する .....	806
43. グラフビューについて .....	809
概要 .....	810
グラフビュー表示設定を変更する .....	811
グラフビューのコントロールポイントの使い方 .....	816
グラフビューのカーブの使い方 .....	818
44. タイムラインについて .....	822
概要 .....	823
ポジショニングにおけるタイムラインの使い方 .....	826
Music Segment Editorのタイムラインを使う .....	827
45. 検索、クエリ、レファレンスの活用 .....	829
概要 .....	830
プロジェクトエレメントを検索する .....	830
あるオブジェクトをレファレンスする、プロジェクトエレメントを 検索する .....	832
クエリの使い方 .....	837
クエリに関する、コツとペストプラクティス .....	845
46. プリセットを使用する .....	846
概要 .....	847
プリセットを使用する .....	847
47. コントロールサーフェスの使用 .....	851
概要 .....	852
コントロールサーフェスデバイスをWwiseに接続する .....	852
Control Surface Sessionを作成する .....	853
コントロールサーフェスのバインディングを理解する .....	853
コントロールサーフェスのバインディングを作成する .....	854
コントロールサーフェスのView Groupsを理解する .....	857
Control Surface Sessionのコンフリクト対応 .....	858
コントロールサーフェスのツールバーの使用 .....	859
VIII. 付録 .....	861
A. ダウンミキシングの動作 .....	863
標準的なコンフィギュレーション .....	864
ダウンミックステーブル .....	865
B. 正規表現クリックアレンスガイド .....	869
C. ショートカット .....	871
ズームできるEditor画面 .....	872
Attenuation Editor .....	872
オーディオファイルの管理 .....	872
Contents Editor .....	873
Game Object 3D Viewer .....	873
Game Profiler .....	874
グローバル（全体） - 起動中のビューや選択中のオブジェクト .....	874
Music Segment Editor .....	874
Position Editor (User-defined)画面 .....	875
Project Explorer .....	875
RPC Graph View .....	875
Schematic View .....	875

---

Soundcaster .....	875
Transport Control .....	876
用語集 .....	877

---

# Wwise 2017.2.4へようこそ

Wwise® version 2017.2.4はaudiokinetic社が開発した、ビデオゲーム用の優れたオーディオを作成するための、強力なミドルウェアソリューションです。Wwiseでは、高度なオーサリングアプリケーションと、これに緊密に統合されたロバストなサウンドエンジン、さらに両者をコントロールする徹底したSDKを備えることで、オーディオデザイナーの生産性を向上させ、クリエイティブな能力を充分に発揮できる手段を提供します。

ゲーム開発パイプラインの具体的なニーズに応えるために構築されたWwiseは、デザイナーやコンポーザー、そしてプログラマーに、特殊なソリューションを提供するものです。ゲームのビジュアル面の制作と同時進行で、オーディオ、ミュージック、そしてモーションをWwise上で開発できるため、開発中のどの段階においても、洗練されたオーディオやモーションの設計とオーサリングを、柔軟に行えます。

## 本ガイドについて

このガイドはゲーム開発業界で活躍するデザイナー、インテグレータ、スクリプトライター、コンポーザー、そしてプログラマー向けに書かれています。コンピューター やオーディオの一般的な知識を有することを前提としています。一部の基本コンセプトは説明されていますが、Wwiseソフトウェアとの関連情報に限定されています。本ガイドは、Wwiseで頻繁に行う一般的なタスクの具体的な情報や手順を説明します。

情報の構成については、[「Wwise Help」](#) のページを参照してください。

## はじめる前に

Wwiseを使い始める前に、ドキュメント「Wwise Fundamentals (Wwiseの基本)」を読んで、Wwiseの主要コンセプトやワークフローを理解するとよいでしょう。「Wwise Fundamentals」に目を通してから、Helpの各章を読むことが推奨されます。Help各章で、Wwiseを使うまでの概念的な説明や、手順、事例、コツ、ベストプラクティスなどが紹介されています。

## その他のドキュメントやサポートの活用

Wwise Helpの他にも、ユーザー向けに数多くの資料やリソースが提供されています。他のガイドやリソースについては[「Wwise ロードマップの紹介」](#) を参照してください。

**audio**kinetic

# パート I. スタート前の準備



---

1. Wwiseのロードマップ .....	3
スタート前の準備 .....	4
Wwiseロードマップの紹介 .....	4
ドキュメントの表記方法 .....	4
Wwise Help .....	5
サポートへの連絡 .....	7
2. Wwiseを使い始める .....	8
Wwiseの起動と終了 .....	9
Wwiseのインターフェースについて .....	10
Wwiseのインターフェースの基礎 .....	12
ワークスペースのユーザー設定 .....	20
作業スピードを速める方法 .....	34

---

# 第1章 Wwiseのロードマップ

スタート前の準備 .....	4
Wwiseロードマップの紹介 .....	4
ドキュメントの表記方法 .....	4
Wwise Help .....	5
サポートへの連絡 .....	7

## スタート前の準備

Audiokinetic社は、Wwiseのユーザー向けに包括的な資料やリソースを提供しています。Wwiseを使い始める前にロードマップを見て、入手可能な情報をご確認ください。オーディオ制作パイプラインにWwiseを導入する際にこれらの資料やリソースをご活用ください。

## Wwiseロードマップの紹介

Wwiseに関する全ての資料やリソースの名称と内容を、下表に示します。

	Release Notes	Release Notes に新機能の情報、製品の制約やその対処方法などを掲載。
	Wwise Installation and Migration Guide	Wwiseや内部アプリケーションのインストールや再インストールの方法、別バージョンへのプロジェクトの移行方法など。
	Wwise Video Tutorials	Wwiseのコンセプトや具体的なタスクの実践方法を説明した、誰でも閲覧できるビデオチュートリアルをウェブサイトで提供。
	Wwise Online Help	Wwiseの各種オプションやビューに関する情報を画面上で確認するための、オンラインヘルプ。
	Wwise User Guide	タスクごとの基本情報や上級者向け情報。
	Wwise Game Simulator Help	様々なプラットフォームでゲームシナリオをテストするための、スクリプトの作成方法。
	Wwise SDK documents	サウンドエンジン、SoundFrame®、プラグインなどの実装情報。
	Wwise Sample Project	Wwiseで実際に作成したサウンドデザイン事例を細かく理解できるサンプルプロジェクトと、付随ドキュメント。
	Audiokinetic Q&A	Wwiseユーザーーやエキスパート達と質問や解凍のやり取りができる Audiokinetic Q&A。
	Wwise Support Center	オンラインサポートセンターへのアクセス。

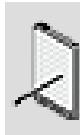
## ドキュメントの表記方法

ユーザー向けの全てのWwiseドキュメントにおいて、次の表記方法が使われています。

- ・「アイコン表示」
- ・「キーボードやマウスの用語」

### アイコン表示

次のアイコンは、情報の種類を表します。

**注記**

Note (注釈) は重要な関連情報を示します。

**Tip**

Tip (ヒント) は特定の状況で使える便利な情報、迂回策、ショートカットなどを示します。

**注意**

Caution (注意) はデータの削除や、簡単には元に戻せないアクションなど、情報の損失や破壊の恐れがあることを示します。このようなタスクが実行される前に必ず“Caution”が表示されます。

## キーボードやマウスの用語

Wwiseの操作にマウスの左右中央の3つのボタンを使います。特記のない限り左ボタンを使います。マウスやキーボードに関する用語の説明を下表に示します。

用語	マウスやキーボードの操作
クリック	マウスの左ボタンを押して素早く離す。特記のない限り、必ず左ボタンを使う。
右クリック	マウスの右ボタンを押して素早く離す。
ダブルクリック	マウスの左ボタンを素早く2回クリックする。
Shift+クリック、 Ctrl+クリック、 Alt+クリック	ShiftまたはCtrlまたはAltキーを押しながら、マウスのボタンをクリックする。
ドラッグ	マウスの左ボタンを押しながらマウスを移動する。ドラッグ&ドロップに相当する。
Alt+キー、 Ctrl+キー、 Shift+キー	最初のキーを押しながら、2つ目のキーを押す。例えば“Ctrl+Zを押す”は、Ctrlキーを押しながらZキーを押すことを意味する。

## Wwise Help

Wwise HelpはWwiseの全てのビュー、フィールド、オプションなどの概要を説明したレファレンストピックやタスク別の情報で構成された、コンテキストセンティブなヘルプシステムです。本ドキュメントはいくつかに分けられ、Wwiseに関する包括的で見つけやすい情報を提供します。

- ・ **パートI「スタート前の準備」** - Wwiseに関する各種ドキュメンテーションの概要、Wwiseの主要コンセプトの紹介、すぐにWwiseを使い始めるための初期設定の情報など。
- ・ **パートII「プロジェクトをセットアップする」** - プロジェクトを1人またはWorkgroup環境内で作業するための準備の仕方、プロジェクトのプラットフォームの設定や調整の仕方、プロジェクト内のアセットの扱い方、Wwiseのアセット階層を構築して活用する方法などのコンセプトと手順。
- ・ **パートIII「サウンドやモーションでゲームを強化する」** - プロパティや動作、エフェクト、モーション、ポジショニング設定、再生プライオリティなどの多様な

設定を使いこなして、リッチで没入間のあるゲーム環境に発展させるためのコンセプトと手順。

- **パートIV「ゲームとのインタラクション」** - ゲームのサウンドを動かすイベント、ダイナミックシークエンス、Wwiseゲームシンクの使い方のコンセプトや手順。
- **パートV「インタラクティブミュージックを作成する」** - Wwiseにおけるインタラクティブミュージックの概念や紹介と、ゲームのインタラクティブミュージックを作成するための細かい作業手順。
- **パートVI「プロジェクトを仕上げる」** - 最終ミックスの作成方法、プロジェクトの多様な面のトラブルシューティングやシミュレーションの方法、ゲームのSoundBankやファイルパッケージの生成方法などに関する情報。
- **パートVII「Wwiseを使いこなす」** - Wwiseをすぐに使いこなすために役立つ各ビューの機能の説明。
- **Wwise Reference (Wwiseユーザーガイドに含まれない)** - Wwiseの様々なオプション機能やビューを完全に網羅した情報を、現在の内容に合わせて表示するヘルプ（コンテキストヘルプ）で提供。
- **Wwise Source/Effect Plug-ins (Wwiseユーザーガイドに含まれない)** - Wwiseに同梱されるソースプラグインやエフェクトプラグインの各種オプションの説明。
- **Wwiseのツール (Wwiseユーザーガイドに含まれない)** - SoundFrameやMulti-Channel Creatorなど、Wwiseに同梱される様々なツールの情報。
- **パートVIII「付録」** - Wwiseのダウンミックス動作や、よく使われる表現、デフォルトのキーボードショートカットなどに関する参考情報を提供。
- **用語集** - Wwiseやオーディオ関連の用語をアルファベット順に説明した用語集。

## 関連トピック

- [「Documentation Preferencesの設定」](#)

## Wwiseのコンテキストセンシティブヘルプをアクセスする

Wwiseのオプションやビューについての情報を見たい時は、そのビューからHelpを開くだけで関連情報が表示されます。

## Wwiseのコンテキストセンシティブヘルプをアクセスするには:

1. 以下のいずれかを行う:

- ビューやダイアログボックス右上のHelpアイコンをクリックする。
- F1を押す。

Wwise Helpが開き、現在のビューやダイアログボックスについての情報が表示される。

## Wwise Helpをアクセスする

Helpシステム全体を見たい場合は、Wwise Helpのスタートページを開いて様々なトピックを閲覧することができます。

## Wwise Helpを開くには:

1. メニューバーで、Help > Wwise Helpをクリックする。

Wwise Helpが開き、スタートページが表示される。

## サポートへの連絡

Audiokinetic [Customer Portal](#)から、サポートチケットを申請できます。



### Tip

ウェブサイトの[Q&A](#)を使って、Wwiseユーザーコミュニティに質問することもできます。

---

## 第2章 Wwiseを使い始める

Wwiseの起動と終了 .....	9
Wwiseのインターフェースについて .....	10
Wwiseのインターフェースの基礎 .....	12
ワークスペースのユーザー設定 .....	20
作業スピードを速める方法 .....	34

## Wwiseの起動と終了

Wwiseは、Microsoft Windows® 7・8・10と、Mac® OS X®で稼働します。Wwiseへ簡単にアクセスできるように、Wwise LauncherアプリケーションでWwise関連の全てのインストールや移行を管理できます。まだWwiseをインストールしていない場合は、最初にaudiokineticの[ダウンロード](#)ページで[Download Wwise](#)をクリックして、Wwise Launcherをインストールします。あとは、希望するWwiseのバージョンの大きいボタンをクリックするだけで、Wwiseのインストールを開始できます。

### Wwiseを起動するには:

#### 1. 以下のいずれかを行う:

- デスクトップのWwise Launcherアイコンをダブルクリックする。
- Windows Startメニュー、またはMac Dockで、**Wwise Launcher**をクリックする。

Wwise Launcher が開く。

#### 2. **Log in**をクリックしてaudiokineticカスタマーポータルにログインすると、自分のユーザーアカウントに付随するあらゆる得点にアクセスできる。

#### 3. 既にWwiseを使ったことがある場合は、**Projects**タブに進む。

別 の方法として、またWwiseを使ったことがない場合には、**Wwise**タブに進む。

#### 4. 既存プロジェクトの場合は、**Open with Wwise ####.#.#.####**（希望するバージョンに該当するもの）をクリックする。Wwiseのこのバージョンで、プロジェクトが開く。

あるいは、**Launch Wwise**をクリックして希望するバージョンを開く。このバージョンのWwiseで、Project Launcherが開く。

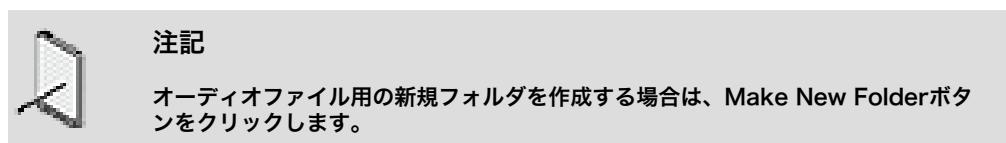
#### 5. プロジェクトを選択するか、**New...**を選択して新しいプロジェクトを作成する。

前者の場合は、Wwiseが選択したプロジェクトをロードするので、作業を開始できる。

後者の場合は、New Projectダイアログボックスが開くので、以下の情報を入力する。

a. **Name**フィールドに、プロジェクト名を入力する。

b. プロジェクトファイルの保存先を変えるには、以下を行う:



- Browseボタン(...)をクリックする。
- Browse For Folderダイアログボックスが開く。

- Wwiseのプロジェクトファイルの保存先フォルダを選択する。
  - OKをクリックする。
- c. オリジナルのオーディオファイルを保存するディレクトリを変えるには、以下を行う:
- Browseボタン(...)をクリックする。
  - Browse For Folderダイアログボックスが開く。
  - Wwiseプロジェクトにインポートするオーディオファイルの、オリジナルコピーの保存先フォルダを選択する。
  - OKをクリックする。
- d. OKをクリックする。

Wwiseの新規プロジェクトが開く。

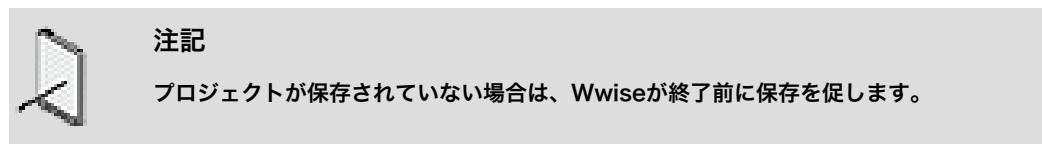
### Wwiseの終了

ワークセッションが完了したあとに、作業内容を保存してからアプリケーションを終了させます。

#### Wwiseを終了するには:

1. メニューバーで、Project > Exitをクリックする。

プロジェクトが閉じて、Wwiseが終了する。



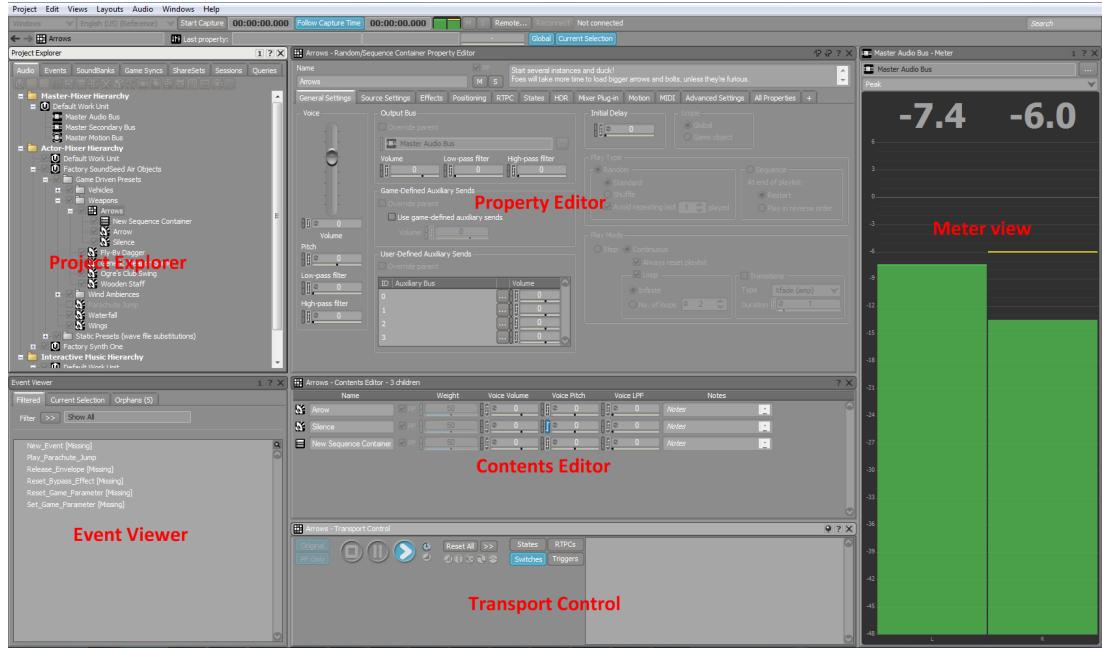
## Wwiseのインターフェースについて

Wwiseのインターフェースはいくつかのビューに分かれています。それぞれのビューの使用目的に合わせて、ゲームのコンテンツを管理したり定義したりするのに必要なツールやオプションがそろっています。特定のタスクや作業が実行しやすいようにビューをまとめてレイアウトが編成されています。Wwiseには様々なレイアウトがあります。詳細は「[レイアウトの活用](#)」を参照。

各レイアウトの上部にメニューバーとツールバーがあります。メニューバーを使って、プロジェクトの保存、レイアウト変更、ビューを開くなど、基本的なコマンドが全て行えます。ツールバーからは、Platform Selector、Language Selector、Capture、Remote Platform Connector、Searchなどの各種ツールに簡単にアクセスできます。



Wwiseを初めて起動すると、Designerレイアウトが表示されます。



Designerレイアウトは、以下のビューで構成されています：

- **Project Explorer** - Wwiseプロジェクトの様々なエレメントを管理し、整理する中心的なエリア。Project Explorerには、以下のタブがあります。
  - **Audio** - Windows ExplorerやMac Finderとよく似た階層ツリーで、プロジェクトの各種アセットを整理できるタブ。Audioタブの主要な階層は、マスター・ミキサー階層、アクターミキサー階層、インターラクティブ・ミュージック階層の3つ。
  - **Events** - プロジェクトにあるAction EventやDialogue Eventを表示。
  - **SoundBanks** - プロジェクトの全てのSoundBankを表示。
  - **Game Syncs** - プロジェクトの全てのSwitch、State、Game Parameter、Triggerを表示。
  - **ShareSets** - プロジェクトの全てのEffect ShareSetやAttenuation ShareSetを表示。
  - **Sessions** - プロジェクトの全てのSoundcasterセッションを表示。
  - **Queries** - プロジェクトの全てのQueriesを表示。
- **Event Viewer** - 現在のプロジェクトで作成した様々なEventを表示。Event Viewerには、Filtered、Current Selection、Orphansの3つのタブがあり、それぞれ異なるフィルタでEventが表示される。
- **Property Editor** - サウンド構造、ミュージック構造、またはバス構造の中にある特定のオブジェクトの特徴を全般的に定義するための、プロパティオプションや動作オプションをまとめて表示。
- **Contents Editor** - Property Editorにロードされた親オブジェクト内のオブジェクト（単数または複数）を表示。また、各オブジェクトに設定されるボリュームやピッチなど、最も一般的なプロパティをContents Editorで簡単にアクセスできる。

- **Transport Control** - 各種オブジェクトを再生できるビュー。Transport Controlには、再生、停止、一時停止など、従来からのオーディオ再生の機能がある。
- **Meter view** - チャンネル毎にPeak、True Peak、RMSの3種類のレベル値が、色分けして表示される。

## Wwiseのインターフェースの基礎

標準的なOSやWwise独自の機能について、以下のセクションで説明します。

- [「Wwiseにおけるネーミング規則」](#)
- [「テキストボックスの使用」](#)
- [「リストの使用」](#)
- [「スライダの使用」](#)
- [「表の使用」](#)
- [「Wwiseのアイコン」](#)
- [「Wwiseのアクションを元に戻す、やり直すには」](#)

### Wwiseにおけるネーミング規則

Wwiseのプロジェクト名やオブジェクト名に、全てのUnicode文字を使用できます。

ただし一部のゲームエンジンにおいて名前に関する制約があるため、Wwiseの以下のエレメント名には一定の制限があります:

- SoundBank
- Events
- Dialogue Events
- Effect ShareSets
- Switch Group
- Switches
- State Groups
- States
- RTPCGame Parameter
- Trigger
- Work Unit

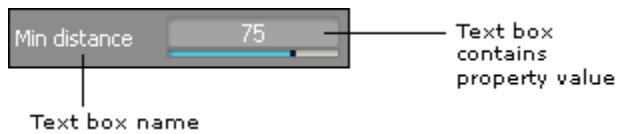
これらのエレメントの名前には英数字（アクセント記号なし）とアンダースコア以外は使えません。また、名前の最初の文字には数字を使えません。

#### 注記

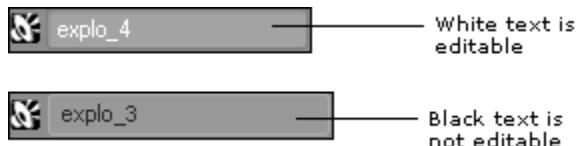
 Wwiseの大部分のオブジェクト名は、最大260文字に制限されます。なおSoundBank名のデフォルト制限数は252文字ですが、実際の文字制限の判断はSoundBank生成時にプロジェクト階層のフルファイルパス内でディスクに書き込む時に行われます。SoundBankのファイルパスがOSの制限を超えてしまった場合（例えばWindows 10では260文字、macOSでは255文字）は、生成が失敗してerror 14: Can't write bank fileのエラーが発生します。

## テキストボックスの使用

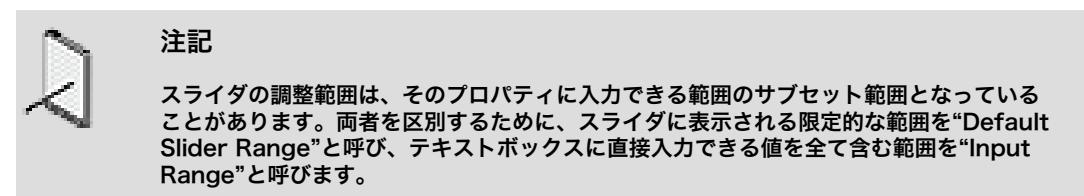
Wwiseのビューの多くに、オブジェクトに関するプロパティ値や特定情報を入力できるフィールドやテキストボックスがあります。テキストボックスの名前から、そのフィールドに表示される情報の種類が分かります。テキストボックスの名前の位置はビューによって異なり、フィールドの横や上下に表示されます。



インターフェースのエリアによって、テキストボックスに表示されるテキストを編集できないことがあります。Wwiseでは、テキスト編集の可否を文字の色で区別します。白字のテキストは編集可能です。黒字のテキストは編集できませんが、テキストを右クリックしてCopy Textを選択し、コピーすることは可能です。



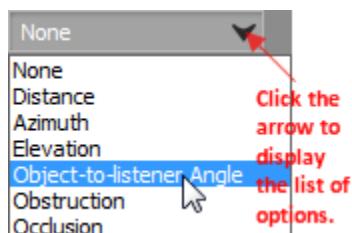
プロパティ値を表示する多くのテキストボックスに縦バーやボックス下のスライダがあり、ドラッグしてテキストボックスの値を変更できます。スライダの詳細については「[スライダの使用](#)」を参照してください。



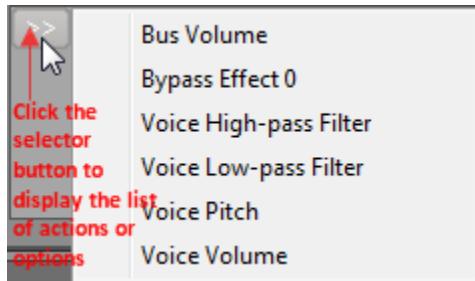
プロパティ値をデフォルト設定に戻すには、テキストボックス上でCtrl+クリックします。

## リストの使用

Wwiseでは、ドロップダウンリストとショートカットリスト（別名コンテキストリスト）の2種類のリストがあります。ドロップダウンリストは本書で単に「リスト」と表現され、事前に設定されたオプションのリストが表示されます。リストを表示させるには、フィールド右側にある矢印をクリックします。



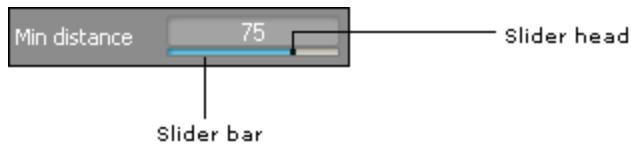
セレクタボタン ([>>]) でいくつかのオプションやアクションが表示されます。場合によっては、ショートカットメニューで選択したオプションを示すフィールドがあります。ボタンをクリックすると、オプションメニューが表示されます。



## スライダの使用

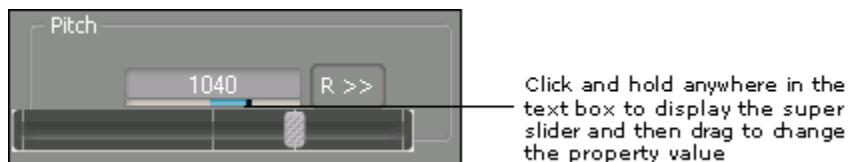
スライダは、選択可能な値の範囲を示すコントロールです。プロパティ値が表示されたフィールドやテキストボックスの多くは、真下に横型のスライダがあります。

横型スライダは、スライダヘッドとスライダバーで構成されます。スライダヘッドは小さい黒点で、現在のプロパティ値を表します。スライダバーは青いバーで、設定可能な範囲内において、現在の値が占める位置を表します。



スライダヘッドのスライダ上の位置は、そのプロパティの設定可能な範囲におけるデフォルト値の位置を示します。また、そのプロパティの設定可能な範囲におけるデフォルト値の位置によって、スライダバーの開始位置や移動方向が変わります。

スライダヘッドを左右にドラッグして、プロパティ値を増減させます。テキストボックス上の任意の位置でクリックしたままにすると、スーパースライダが表示され、プロパティ値の微調整が簡単にできます。マウスボタンを離すと大きいスライダが消えます。値をさらに正確に調整したい時は、スライダをShift+ドラッグして値をより細かく増減できます。

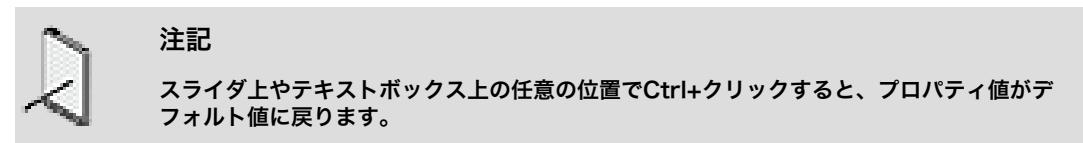
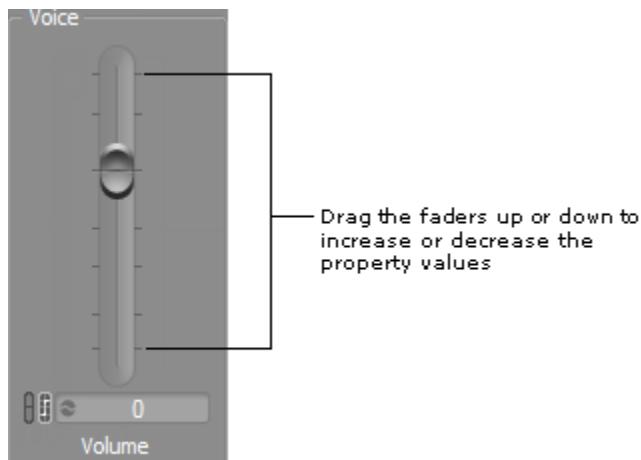


### 注記

多くのスライダの範囲は、テキストボックスで入力できる範囲の一部に限定したデフォルトレンジとなっています。あるプロパティの設定できる範囲を広げたい場合は、デフォルトスライダレンジ外の値を入力します。各プロパティのデフォルトスライダレンジや最大インプットレンジを確認するには、Helpを参照してください。

## フェーダの使用

Volumeなどの一部のプロパティは、横型スライダの代わりに縦型スライダまたはフェーダを使って値を変更します。Volumeコントロール用は縦型スライダを採用して、ハードやソフトのミキサーにあるフェーダを再現しています。フェーダを上下にドラッグして、プロパティ値を増減させます。値をさらに正確に調整したい時はフェーダの上または下をShift+クリックすることで、値をより細かく増減できます。



## 表の使用

表とは、データをRow（行）やColumn（列）に整理したものです。最初の列の情報は他の列の情報と直接関係します。Wwiseは表を使ってEventやPresetなどの情報を表示します。

No.	Actions	Objects	Scope	Notes
1	>> Play	rlaunch	Game object	
2	>> Play	Thrust	Game object	
3	>> Set State	PlayerInWater : No	Global	Rockets do not fire underwater.

表中を矢印キーを使って上下に移動できます。ほとんどのテーブルで、列のサイズ変更、列ヘッダ別の情報のソート、表中のプロパティ値の直接編集などができます。

### 表のColumn（列）サイズを変更するには:

1. 列の境界線にマウスポインタを合わせる。

ポインタが二重矢印に変わる。

2. 以下のいずれかを行う:

- ・列幅を広げるには、境界線を右にドラッグする。
- ・列幅を狭めるには、境界線を左にドラッグする。

### 表のColumn（列）を設定するには:

1. 列のヘッダ行にマウスを合わせる。
2. ヘッダ行を右クリックする。

ショートカットメニューでオプションが1つだけ表示される:Configure Columns...。

3. Configure Columns...を選択する。

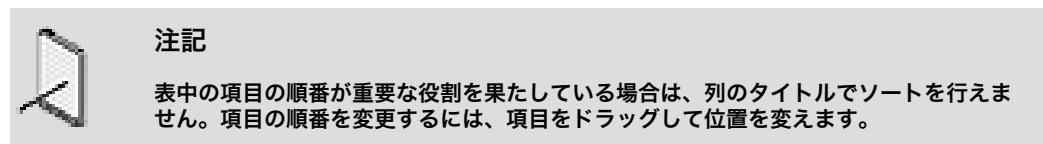
Configure Columnsダイアログボックス...が表示されるので、表示する列と順番を指定する。

### 表の情報をソートするには:

1. 列のタイトルをクリックする。

表の情報が、その列にある情報の昇順に並べ替えられる。

2. 情報を降順に並べ替えるには、列のタイトルをもう一度クリックする。



### 表の内容にフィルタをかけるには:

1. 右上にあるSearchアイコンをクリックする。デフォルトショートカットはCtrl +F3。

標準的な英数字検索フィールドが表示される。



2. フィールドに数字、文字、アンダースコア、スペースなどを自由に入力できる。

同じ文字パターンがどの列にも含まれない項目は、検索フィルタによって表示されない。リストに表示される項目数を減らすことで、求めているものが探しやすくなる。



3. 検索フィールドを閉じてフィルタをリセットするには、Searchアイコン左のCloseアイコンをクリックするか、Escを押す。



### テーブルテキストへのリンクを追加する

テーブルフィールドのテキストは、最初に適切なプロトコルハンドラを追加することによって、他の場所にリンクすることができます。ハンドラはフィールド内のすべてのテキストに適用されるため、入力は正確で完全でなければなりません。たとえば、`http://www.audiodkinetic.com`を Notes フィールドに追加すると、List Viewから見て、下線付きのクリック可能なテキストが表示されます。この場合、プロトコルハンドラとURLは、AudiodkineticのWebサイトにブラウザをはっきりと開きます。また、SoundFrame Test Applicationを使用してゲームオブジェクトを作成し、メールアプリケーションにリンクする`mailto : someemailaddress @ someemailserver`という名前を付けることもできます。ただし、`HKEY_CLASSES_ROOT\PROTOCOLS\Handler`に定義されているプロトコルハンドラを使用することもできます。

## Wwiseのアイコン

Wwiseのインターフェースでは、アイコン、つまり視覚表現を使って各種エレメントを表しています。Wwiseで使う様々なアイコンを、下表に示します。

アイコン	名前	内容
	Virtual Folder	Project Explorer全体でプロジェクト要素をグループ化するために追加できるフォルダです。
<b>Master Mixer アイコン</b>		
	Audio Bus	サウンドオブジェクトをまとめたグループのことで、ゲームの様々なサウンド構造を使った作業ができる。例えば、全ての環境音構造やミュージック構造を1つのAudio Busにまとめて、全てのプレイヤーサウンド構造を別のAudio Busにまとめることができます。
	Auxiliary Bus	プロジェクトのどこにでも設定できるサウンドオブジェクトのサブグループのこと、ボリューム、チャンネルコンフィギュレーション、ポジショニング、RPCなどを調整したり、Effect、State、ミキサー・プラグインなどを適用してから親のAudio Busに戻すことができる。AUXバスでは、ダッキング、ボイス、HDRミックスの調整はできない。
<b>Actor-Mixer アイコン</b>		
	Audio source	オーディオファイルとオブジェクトの間に独立した抽象化レイヤ。プロジェクトにインポートしたオーディオファイルにリンクしていく、アクティブなゲームプラットフォームのコンバージョン設定をここで決める。
	Source Plug-in	Wwise外にあるソース・プラグインによって作成されたオーディオソース。
	Sound SFX	サウンドエフェクトの入ったサウンドオブジェクト。
	Sound Voice	ボイスオーバーやキャラクターダイアログの入ったサウンドオブジェクト。
	Work Unit	プロジェクトを別々のセグメントに分ける個別のXMLファイルで、複数のチームメンバーが同時にプロジェクトの異なる部分の作業を進めることができる。このXMLファイルは、既存のソース・コントロール・システムで簡単に管理できる。
	Actor-Mixer	1つ以上のサウンド、コンテナ、Actor-Mixerなどで構成される階層構造。Actor-Mixerを使って、その下位にある全てのオブジェクトにプロパティを適用できる。
	Blend Container	同時に再生されるオブジェクトやコンテナ（単数または複数）のグループ。Blend Container内のサウンドやコンテナは、RPC使用によるGame Parameter値へのサウンド・プロパティのマッピングが実行されるブレンド・トラックにグループ分けが可能。Game

アイコン	名前	内容
		Parameter値に基づいて、ブレンドトラック内のオブジェクト間にクロスフェードを適用することも可能。
	Random Container	ランダムに再生されるオブジェクトやコンテナ（単数または複数）のグループ。
	Sequence Container	特定の順番またはプレイリストに従って再生されるオブジェクトやコンテナ（単数または複数）のグループ。
	Switch Container	一連のSwitchやStateに整理されたオブジェクト、またはコンテナのグループで、関連するSwitchやStateがゲームに呼ばると、それが再生される。
Interactive Musicアイコン		
	Music Track	個々のミュージッククリップのアレンジが入ったミュージックオブジェクトであり、Music Segment内で視覚的に整列できるようウェーブフォームで表示されている。
	# #Music Track - Random Step	# #親セグメントが再生される度に、サブトラックがランダムに再生されるMusic Track。
	# #Music Track - Sequence Step	# #親セグメントが再生される度に、サブトラックが順番に再生されるMusic Track。
	# #Music Track - Switch	# #Music Trackの1種で、指定されたSwitch GroupまたはState GroupのSwitchまたはStateに従い、結びつけられたサブトラックを再生する。
	Music Segment	複数のMusic Trackが入ったミュージックオブジェクトのこととで、インラクティブミュージックの音楽アレンジメントをつくるために、シンクポイントを使って、各Music Trackをそろえることができる。
	Music Playlist Container	セグメント（単数または複数）が入ったグループで、セグメントが一定の方法で整理され、ランダムまたは特定の順番に従って再生される。
	# Music Switch Container	一連のSwitchやStateに整理されたMusic Segmentやコンテナ（単数または複数）のグループで、関連するSwitchやStateがゲームに呼び出されると、それが再生される。
その他のプロジェクトエレメントのアイコン		
	Event	Wwiseオブジェクト（単数または複数）に適用された、再生、ミュート、一時停止などのアクションまたは一連のアクションを使用してゲーム内にオーディオやモーションをトリガーする方法。
	Dialogue Event 「ダイアログイベント」	ダイアログまたはダイアログ以外のオーディオやモーションをゲーム中にトリガーする方式で、State GroupやSwitch GroupとStateやSwitchを組み合わせてパスを編成して、パスにアサインされているWwise Actor-Mixer Hierarchyのオブジェクトを使う。
	SoundBank	ゲーム内の特定の時点にゲームのプラットフォームメモリにロードされるイベント、Wwiseオブジェクト、メディアなどのグループ。
	Switch Group	ゲーム内の特定エレメントのために存在する様々な選択肢を管理するために、グループにまとめられた関連性のあるSwitchの集まり。
	Switch	ゲーム中の特定エレメントのために存在する選択肢。
	State Group	ゲーム環境内で発生するグローバルな変更を管理するために、グループにまとめられた関連性のあるStateの集まり。
	State	ゲーム中の物理的および環境的状況変化を表現する、ゲームオーディオのプロパティへのグローバルなオフセットまたは調整。
	Game Parameter (RPC) 「ゲームパラメータ (RPC)」	例えば、カーレーシングゲームのスピードやRPMなど、RPCを使用してWwiseプロパティ値にマップできるゲームのパラメータ。

アイコン	名前	内容
	Trigger	短い音楽を再生させる、ゲーム中の特定のアクション。
	Effect ShareSet	ゲームのオーディオを強化するために使用できるオーディオエフェクト設定。設定はShareSetとして保存され、オブジェクト間で共有可能。
	Attenuation ShareSet	リスナーからの距離に基づく、サウンドのボリュームに関する減衰設定。設定はShareSetとして保存され、オブジェクト間で共有可能。
	Conversion Settings ShareSet	サンプルレート、オーディオフォーマット、チャンネル数など、オーディオアウトプットの全体的な品質を定義するコンバージョン設定。設定はShareSetとして保存され、オブジェクト間で適用と共有が可能。
	Soundcaster Session	シミュレーションを目的として、特定の順番に配列されたサウンド、ミュージック、モーション、イベントの各モジュールをグループ化したもので、ゲームシンク設定と共に保存される。
	Mixing Session	ゲームのオーディオミックスの微調整を目的として、一種のミキシングコンソール内に保存した複数のバスやオブジェクトとそれぞれのプロパティをまとめたグループ。
	Query	特定のオブジェクトやプロジェクトエレメントを探すために使用される検索条件のセット。
<b>プロパティ値関連</b>		
	Link	このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしている。
	Unlink	このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしていない。
	Partial Unlink	このプロパティ値は、現在のプラットフォームでは他のアクティブなプラットフォームにリンクしているが、他の1つ以上のプラットフォームにおいて、この設定値がリンクしていない。
	Link Mixed	選択されたオブジェクトによっては異なるリンクステータスがあります。リンクされていたり、リンクされていなかったり、一部だけリンクされています。
	RTCP Disabled RTCP無効	このプロパティ値は、ゲーム内のパラメータ値に結び付いていない。
	RTCP Enabled RTCP有効	このプロパティ値は、ゲーム内のパラメータ値に結び付いている。例えば、ゲーム内の車のスピードを、Wwiseにおけるピッチのプロパティに直接、結びつけることができる。ゲーム中に車のスピードが加速すると、Wwise側のピッチもリアルタイムで上昇する。
	RTCP Mixed	ゲーム内のパラメータ値は、Multi Editorにロードされたオブジェクトの一つ以上のすべてではないプロパティ値にを関連付けられています。Property Editor やContents Editorではこのインジケーターが見られません。
	Randomizer Enabled	Randomizerエフェクトが適用されたプロパティ値。
	Randomizer Disabled	Randomizerエフェクトが適用されていないプロパティ値。
	Randomizer Mixed	Randomizer エフェクトは Multi Editor にロードされた 1 つ以上のすべてではないプロパティ値に適用されます。Property Editor やContents Editorではこのインジケーターが見られません。
	State Disabled	このプロパティ値は、ステートに結びついていない。
	State Enabled	ステートグループがこのプロパティ値に結びついている。ボリュームなど、ここで対象となっているプロパティが、適用されるステートによって変化することを意味します。

アイコン	名前	内容
プロパティの灰色表示		
	Unsupported feature	プロパティが薄い青で表示されている場合は、現在のプラットフォームがこの機能に対応していないことを示す。
プリセット設定のアイコン		
	Save Preset	現在のビューにある全ての値を、現状のままプリセットとして保存するコマンド。
	Load Preset	既に保存されたプリセットをロードするコマンド。
ビュー設定のアイコン		
	Help	現在のビュー、ウィンドウ、またはダイアログボックスのオンラインヘルプを表示させるコマンド。
	View Settings	現在のビューを定義するための設定項目を表示するダイアログボックスを開くコマンド。
	Collapse View	フローティングビューを最小化するコマンド。
	Expand View	フローティングビューを最大化するコマンド。
	Close View	フローティングビューを閉じるコマンド。

### Wwiseのアクションを元に戻す、やり直すには

プロパティ値の変更、オブジェクトの移動、イベントの作成など、Wwiseで実行するアクションの多くを「元に戻す」ことが可能です。また、誤って元に戻したアクションを「やり直し」して、前の値やStateに戻すこともできます。

アクションを元に戻すには、Edit > Undo <アクション名>をクリックするか、Ctrl +Zを押します。元に戻すことができるアクション数は、直前から最大200個までです。

アクションをやり直すには、Edit > Redo <アクション名>をクリックするか、Ctrl +Yを押します。1回元に戻すと、1回までやり直せます。

## ワークスペースのユーザー設定

Wwiseの作業環境をカスタム設定する方法は、いくつかあります。可能なカスタム設定（カスタマイズ）の種類を、下表に示します。

カスタマイズの対象	内容
「ビューを活用する」	フラットビューやフローティングビューのサイズや位置の設定方法
「レイアウトの活用」	表示させるレイアウトの設定方法と、そのレイアウト内に表示させるビューの種類や位置の設定方法
「ユーザープリファレンスの設定」	ユーザー設定の設定方法

### ビューを活用する

Wwiseのインターフェースは複数のビューで構成されています。ビューとは、インターフェースに表示される個別のウィンドウのこと、あるタスクに関連した情報やコマンドがまとめられています。

ビューには、フローティング型とフラット型の2種類があります。フラットビューはレイアウト内に組み込まれたビューのことで、フローティングビューはウィンドウのようにレイアウトの上に重ねて表示されるビューのことです。両者とも、位置やサイズを調整できます。

### ビューを表示するには:

1. メニューバーで、Views > <ビュー名>をクリックする。

選択したビューがフローティングビューとして表示される。

### フローティングビューを切り替えるには:

1. メニューで、Windows > <ビュー名>をクリックする。

選択したビューが全てのフローティングビューの最前面に表示される。

### 関連トピック

- 「複数のビューの同期」
- 「ビューのタイトルバーのアイコンについて」
- 「レイアウトのフローティングビューの管理」
- 「レイアウトのビューのサイズ変更」
- 「レイアウトのビューの追加と削除」
- 「レイアウトの活用」

### ビューのタイトルバーのアイコンについて

ビューのタイトルバーの右側に、アイコンがいくつか表示されます。アイコンをクリックすると、そのビューに関連したコマンドが実行されます。表示されるアイコンの組み合わせは、ビューによって異なります。ビューのタイトルバーに表示されるアイコンのリストを、下表に示します。

アイコン	名前	内容
	Save Preset	現在のビューにある全ての値を、現状のままプリセットとして保存する。
	Load Preset	保存されたプリセットをロードする。
	Sync Group	特定グループ（1～4）にビューをシンクさせる。ビューをグループにシンクさせると、そのグループ内にある全ビューで同じタブや項目が表示される。例えば、3つの異なるProject Explorerビューを、それぞれ別のレイアウトに設定して、Group 1にシンクさせたとする。そのうち1つのProject ExplorerビューでタブをGame Syncsタブに切り替えると、同じグループの他のProject Explorerビューでも自動的にGame Syncsタブに切り替わる。 なお、この機能はProject ExplorerビューとEvent Viewerビューのみで利用できる。

アイコン	名前	内容
	Help	現在のビュー、ウィンドウ、またはダイアログボックスのオンラインヘルプを表示する。
	View Settings	現在のビューで表示させる項目を設定するための、ダイアログボックスを表示する。
	Close View	フローティングビューを閉じる。

### レイアウトのフローティングビューの管理

あるレイアウトで作業している時に、フローティングビューをいくつか同時に開いておくことができます。これらのフローティングビューはウィンドウマネージャで効率的に管理できます。ウィンドウマネージャを使って、簡単にフローティングビューを開き、ビューの最大・最小化や、ビューを元のサイズに戻したり、移動したり、閉じたりすることもできます。

#### レイアウトのフローティングビューを管理するには:

1. メニューで、Windows > Windows...をクリックする。

Windowsダイアログボックスが開く。

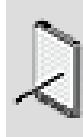
2. View Nameリストからビューを1つ以上選択して、次のいずれかをクリックする。
  - **Activate** - 選択したフローティングビューをアクティブにして、全ウィンドウの最前面に表示。
  - **Minimize** - 選択したフローティングビューの最小化。
  - **Maximize** - 選択したフローティングビューの最大化。
  - **Restore** - 選択したフローティングビューを最大・最小化した状態から、元のサイズと位置に戻す。
  - **Move To...** - 選択したフローティングビューをモニター画面（単数または複数）の新しい位置に移動。
  - **Close Window(s)** - 選択したフローティングビューを閉じる。
3. 終了後にCloseをクリックして、Windowsダイアログボックスを閉じる。

### 関連トピック

- 「[ビューを活用する](#)」
- 「[レイアウトのビューのサイズ変更](#)」
- 「[レイアウトのビューの追加と削除](#)」
- 「[ファクトリレイアウトのデフォルト設定を復元する](#)」
- 「[ビューのタイトルバーのアイコンについて](#)」

### 複数のビューの同期

Project ExplorerやEvent Viewerを異なるインスタンスとして複数のレイアウトで開く場合、これらのビューをシンクさせることもできます。シンクすると、ビューで実行した選択や移動が、他のビューにも反映されます。



### 注記

同一レイアウトの2つのビューを、同じシンクグループに登録することはできません。

## ビューのグループをシンクさせるには:

1. Project ExplorerまたはEvent Viewerのタイトルバーにある、Sync Groupアイコンをクリックする。
2. コンテキストメニューで、以下のオプションを1つ選択する。
  - No sync group - ビューをグループにシンクさせない。
  - Sync Group 1 - ビューをGroup 1にシンクさせる。
  - Sync Group 2 - ビューをGroup 2にシンクさせる。
  - Sync Group 3 - ビューをGroup 3にシンクさせる。
  - Sync Group 4 - ビューをGroup 4にシンクさせる。
3. 他のレイアウトのProject ExplorerやEvent Viewerもシンクさせるには、そのレイアウトに切り替えて上記1、2を繰り返す。

## 関連トピック

- 「[ビューを活用する](#)」
- 「[レイアウトのビューのサイズ変更](#)」
- 「[レイアウトのフローティングビューの管理](#)」
- 「[ビューのタイトルバーのアイコンについて](#)」
- 「[レイアウトのビューの追加と削除](#)」
- 「[レイアウトへのビューのドッキングとドッキング解除](#)」
- 「[ファクトリレイアウトのデフォルト設定を復元する](#)」

## レイアウトの活用

いくつかのビューを組み合わせて、新しいレイアウトを作ることもできます。Wwiseには8種類のデフォルトレイアウトがあり、それぞれ特定のタスクや作業が行いやすいように最適化されています。作業の内容に合わせて、レイアウトを変えて使います。

Wwiseには次のレイアウトがあります:

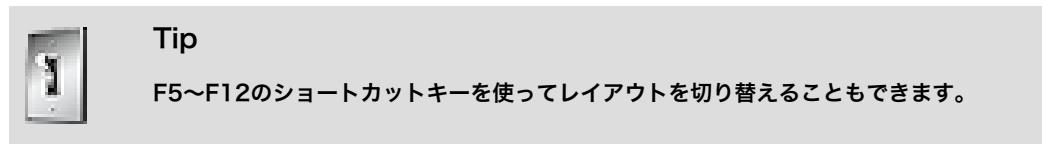
- **Designer** - ゲームのサウンドエフェクト、ダイアログ、モーションを作り上げるアセットを管理し、ビルドし、定義する時のレイアウト。
- **Profiler** - ゲームのオーディオエレメントやモーションエレメントの実行中のパフォーマンスをモニターし分析するためのレイアウト。
- **SoundBank** - プロジェクトのSoundBankを作成し、管理し、生成するためのレイアウト。
- **Mixer** - プロトタイプを作成し、Wwiseで様々なオブジェクトをミックスするためのレイアウト。

- **Schematic** - プロジェクト構造の全体像を図式化して確認するためのレイアウト。
- **Interactive Music** - ゲームのインタラクティブミュージック部分を作り上げるミュージックアセットを管理し、ビルドし、定義するためのレイアウト。
- **Dynamic Dialogue** - ゲームのダイナミックダイアログを動かすダイアログイベントを管理し、ビルドするためのレイアウト。
- **Game Object Profiler** - ゲームオブジェクトの観点からオーディオパフォーマンスをモニターし分析するためのレイアウト。

### レイアウトを切り替えるには:

1. メニューで、Layouts > レイアウト名をクリックする。

新しいレイアウトが表示される。



### 関連トピック

- 「ビューを活用する」
- 「レイアウトのビューのサイズ変更」
- 「レイアウトのビューの追加と削除」
- 「レイアウトへのビューのドッキングとドッキング解除」
- 「ファクトリレイアウトのデフォルト設定を復元する」

### レイアウトのビューのサイズ変更

レイアウト内のビューのサイズを変えて、デフォルトレイアウトを調整することができます。

### フラットビューのサイズを変更するには:

1. 2つ以上のビューの間に分割バーの上まで、マウスのポインタを移動する。

分割バーがハイライトされ、マウスのポインタが双方向矢印に変わる。

2. ハイライトされた分割バーをドラッグして、バーに接する全てのビューのサイズを変更する。

### 関連トピック

- 「レイアウトのビューの追加と削除」
- 「レイアウトへのビューのドッキングとドッキング解除」
- 「ファクトリレイアウトのデフォルト設定を復元する」

- ・「レイアウトの活用」
- ・「ビューを活用する」

### レイアウトのビューの追加と削除

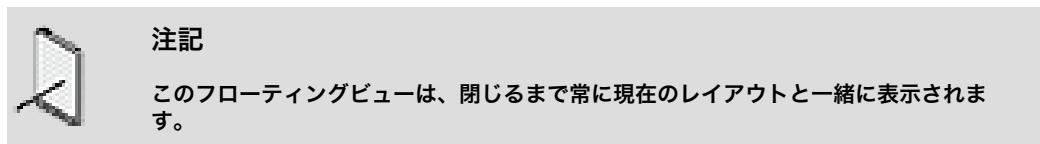
Wwiseのファクトリレイアウトはワークフローを最大限に効率化するために最適化されていますが、レイアウトにビューを追加したり削除したりすることもできます。フローティングビューもフラットビューも、削除されるまでレイアウトの一部として残ります。

#### レイアウトに新しいビューを追加するには:

1. メニューで、Views > <ビュー名>を選択する。

選択したビューがレイアウト上に、フローティングビューとして開く。

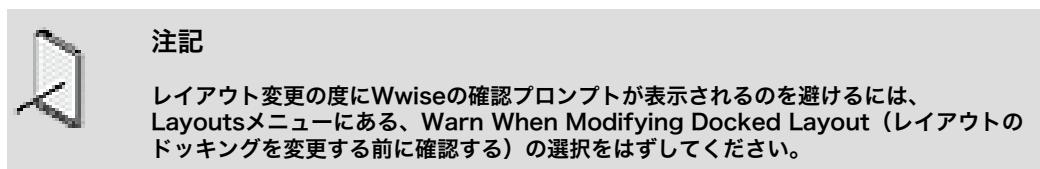
2. フローティングビューのサイズを変更して、モニター画面（単数または複数）の好きな位置に移動する。



#### レイアウトからビューを削除するには:

1. フローティングビューまたはフラットビューのタイトルバーにある閉じるアイコン (X) をクリックして、フローティングビューを閉じる。

ビューがレイアウト内に組み込まれている（ドッキングされている）場合は、レイアウト変更の前にWwiseの確認プロンプトが表示される。



2. Yesをクリックする。

組み込まれていたビューがレイアウトから削除され、空いたスペースを埋めるために、残ったビューのサイズが変わる。

### 関連トピック

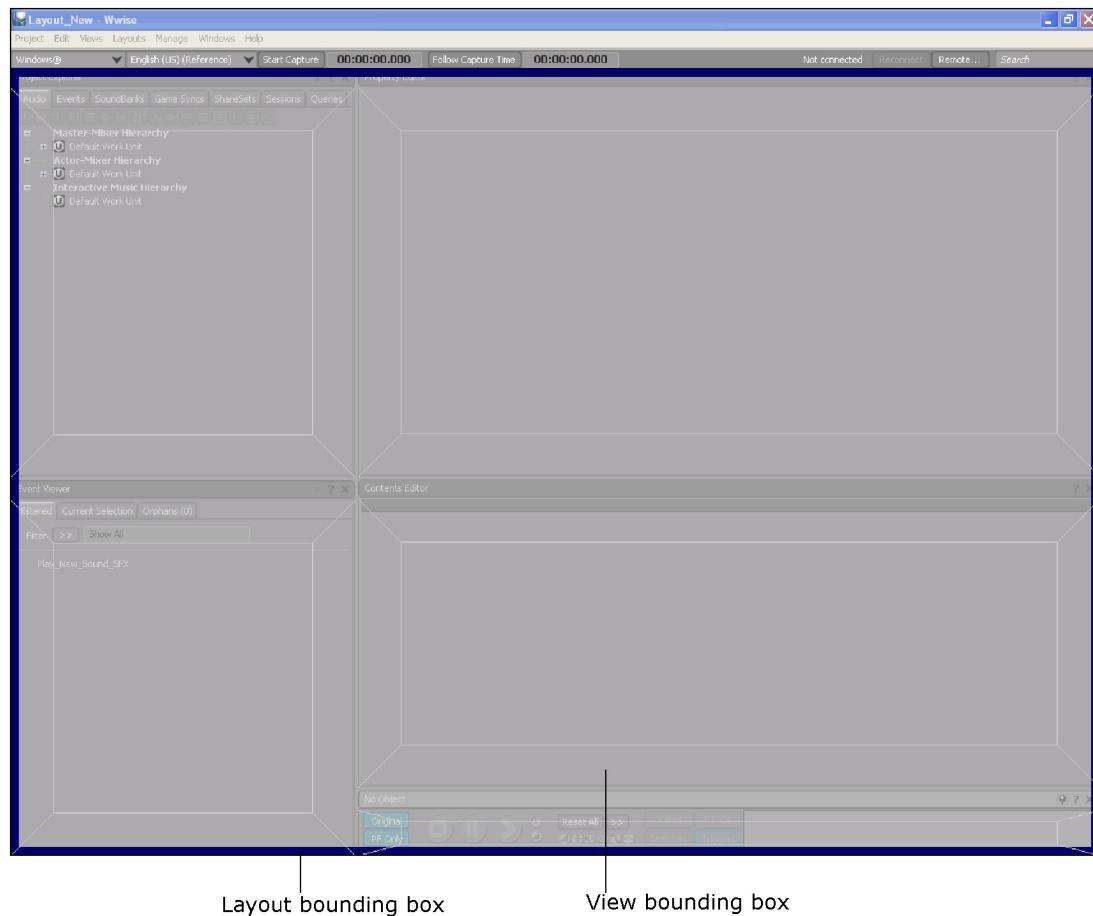
- ・「レイアウトへのビューのドッキングとドッキング解除」
- ・「レイアウトのビューのサイズ変更」
- ・「ファクトリレイアウトのデフォルト設定を復元する」
- ・「レイアウトの活用」
- ・「ビューを活用する」

### レイアウトへのビューのドッキングとドッキング解除

デフォルトレイアウトはワークフローを最大限に効率化するために設計されていますが、状況に応じてレイアウト内のビューの内容や配置を変更することもできます。

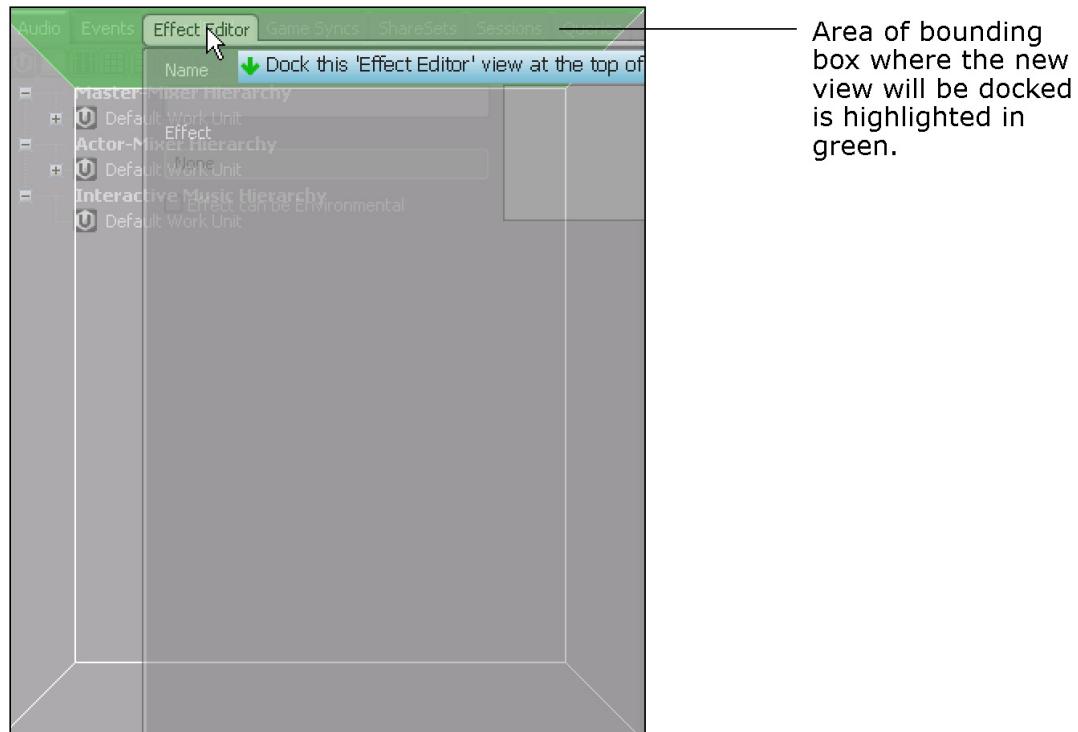
レイアウトにビューをドッキングする時、新しいビューを入れる位置をいくつかの選択肢から選べます。WwiseのEdit layoutモードでは、レイアウト内の各ビューを分ける分割バーに従って、レイアウトが複数のエリアに分かれます。また次の各エリアの周りにバウンド（囲い）ボックスも表示されます。

- ・ レイアウト全体の周り（青の太線）
- ・ レイアウト内の各ビューの周り

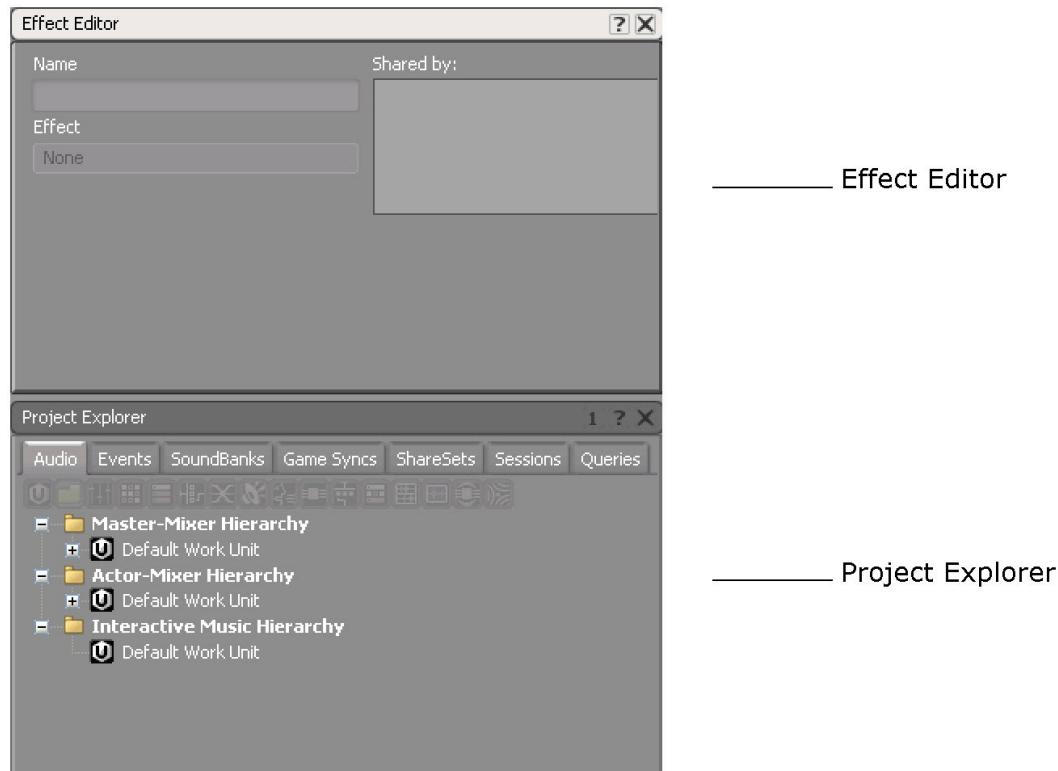


レイアウトにおける各バウンドボックスの上下左右の4線が、新ビューをドッキングできる位置です。例えば、DesignerレイアウトのProject Explorerの真上に、Effect Editorを組み込みたいとします。この場合、Effect Editorを開きProject Explorerのバウンドボックスの上部までドラッグします。

## Wwiseを使い始める



Project Explorerが占めていたエリアが2つに分割され、Effect Editorが上半分、Project Explorerが下半分に配置されます。



Effect Editorをレイアウトの幅全体に渡って配置させるには、ビューのバウンドボックスではなく、レイアウト全体のバウンドボックスに対してドッキングさせます。

### レイアウト内にフローティングビューをドッキングするには:

1. フローティングウィンドウ内のビューのタイトルバーをクリックして、ビューをドラッグし始める。

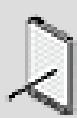
インターフェースが、edit layoutモードになる。

2. レイアウト内でビューをドッキングさせたい場所のバウンドボックスの上で、フローティングビューをドラッグする。

当該バウンドボックスのエリアが緑にハイライトされる。

3. マウスボタンを離す。

レイアウト変更の確認プロンプトが表示される。



#### 注記

レイアウト変更の度にWwiseの確認プロンプトが表示されるのを避けるには、Layoutsメニューにある、Warn When Modifying Docked Layout (レイアウトのドッキングを変更する前に確認する) の選択をはずしてください。

4. Yesをクリックする。

フローティングビューが、レイアウト内の指定した場所に組み込まれ、フラットビューに変わる。

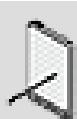
5. 必要に応じて各フラットビューのサイズを調整して、好みのレイアウトに変える。

### レイアウトからビューを外すには:

1. レイアウトから外すビューのタイトルバーをクリックして、ビューをドラッグし始める。

Wwiseが、edit layoutモードになる。

2. バウンドボックスが緑にハイライトされないことを確認しながら、レイアウト内でビューをドラッグする。



#### 注記

バウンドボックスのエリアが緑にハイライトされてしまうと、現在のビューがレイアウト内の他の位置に移動することになります。

3. マウスボタンを離す。

レイアウト変更の確認プロンプトが表示される。



### 注記

レイアウト変更の度にWwiseの確認プロンプトが表示されるのを避けるには、Layoutsメニューにある、Warn When Modifying Docked Layout (レイアウトのドッキングを変更する前に確認する) の選択をはずしてください。

4. Yesをクリックする。

フラットビューがフローティングビューに変わる。

## 関連トピック

- ・「レイアウトのビューのサイズ変更」
- ・「レイアウトのビューの追加と削除」
- ・「ファクトリレイアウトのデフォルト設定を復元する」
- ・「レイアウトの活用」
- ・「ビューを活用する」

### ファクトリレイアウトのデフォルト設定を復元する

編集したレイアウトを、デフォルトのレイアウト設定に戻すことができます。いつでもWwiseのデフォルトレイアウトを復元することができます。ただし、現状では編集したレイアウトの設定を保存することができません。ファクトリレイアウトに戻した場合は、自分で設定したレイアウト変更が失われます。

### ファクトリレイアウトのデフォルト設定を復元するには:

1. メニューで、Layouts > Reset Factory Layoutsをクリックする。

Reset Factory Layoutsダイアログボックスが開く。

2. デフォルト設定に戻したいレイアウトを選択する。
3. OKをクリックする。

選択したファクトリレイアウトが、デフォルト設定に戻される。

## 関連トピック

- ・「レイアウトのビューのサイズ変更」
- ・「レイアウトのビューの追加と削除」
- ・「レイアウトの活用」
- ・「ビューを活用する」

### ユーザープリファレンスの設定

ユーザープリファレンスの設定で、Wwiseにおける以下の点をカスタム設定できます。

- ・「確認メッセージの有効化」
- ・「外部オーディオエディタの選択」

- ・「アウトプットバッファのレイテンシ設定」
- ・「オーディオデバイスの優先設定」
- ・「オーディオチャンネルコンフィギュレーションの設定」
- ・「Music Trackのルックアヘッドタイムの設定」
- ・「Documentation Preferencesの設定」

#### 確認メッセージの有効化

確認メッセージで、Wwiseで特定のタスクを実行している時に便利な情報を提供します。確認メッセージによっては、表示された時にDon't ask againオプションをクリックしてそのメッセージを無効にし、今後は表示しないように設定できます。全てのメッセージを再び表示させるには、ユーザープリファレンスで再度、有効にします。

#### 確認メッセージを有効にするには:

1. メニューで、Project > User Preferencesをクリックする。

User Preferencesダイアログボックスが開く。

2. グループボックスConfirmation Messagesで、Resetをクリックする。

確認のダイアログボックスが表示される。

3. OKをクリックする。

これで、Wwiseで特定のアクションを実行する前の確認メッセージが表示される。

#### 外部オーディオエディタの選択

Wwiseでは、好きなオーディオファイルエディタを使ってオーディオファイルを直接編集できます。ただしその前に、使用するプログラムを利用可能なエディタのリストに追加する必要があります。

#### 外部オーディオエディタソフトを選択するには:

1. メニューで、Project > User Preferencesをクリックする。

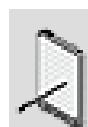
User Preferencesダイアログボックスが開く。

2. グループボックスExternal Editorsで、Add...をクリックする。

Open Fileダイアログボックスが開く。

3. 実行可能なオーディオエディタ(exe)を選択して、Openをクリックする。

利用可能なオーディオエディタのリストに、エディタが追加される。



#### 注記

リストにあるエディタを削除するには、そのエディタを選択してDeleteをクリックします。

#### 4. OKをクリックして、設定を保存する。

これで、追加したエディタをWwiseから直接アクセスできる。

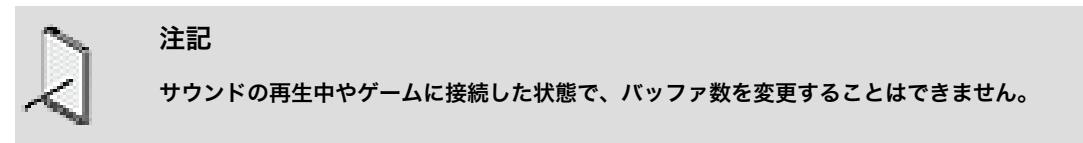
#### 外部オーディオエディタをデフォルトに選択するには:

1. エディタのリストから、デフォルトとするエディタを選択する。
2. Set As Defaultをクリックする。（リストに2つ以上のエディタがなければ、このボタンは有効にならない。）

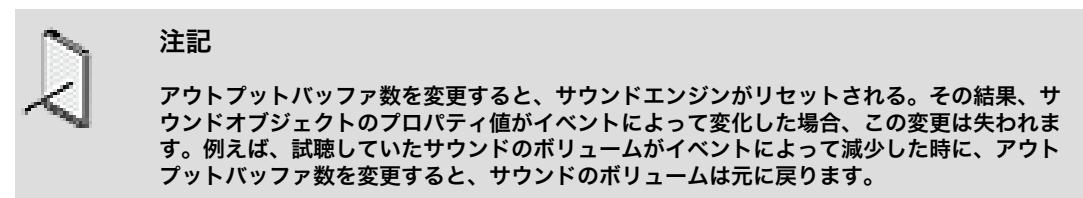


#### アウトプットバッファのレイテンシ設定

Wwiseでサウンドを再生すると、再生レイテンシを削減するためにWwiseのサウンドエンジンはプリフィルされたバッファを使用します。デフォルト設定で、サウンドエンジンはこのバッファを4つ使用します。ただし、Wwiseの使用中にパフォーマンス上の問題が発生する場合の対策として、サウンドエンジンが使うバッファ数を調整することもできます。



Wwiseでサウンドオブジェクトを再生する時にボイススターベーションが発生しているようであれば、アウトプットバッファ数を増やしてみてください。逆に、再生レイテンシを減らしたい時は、バッファ数を減らしてみてください。



#### アウトプットバッファのレイテンシを設定するには:

1. メニューで、Audio > Audio Preferencesをクリックする。

Audio Preferencesダイアログボックスが開く。

2. グループボックスSound Engineで、アウトプットバッファ数をリストから選択する。かつての数字は、該当するレイテンシを示す。
3. OKをクリックして、設定を保存する。

これで、Wwiseでサウンドを再生する時にサウンドエンジンは指定されたアウトプットバッファ数を使う。

## オーディオデバイスの優先設定

デフォルトで、Wwiseで再生する音は全て、Windowsのデフォルトプレイバックデバイスでアウトプットされます。Wwiseで直接アウトプットデバイスを変更できるほか、オーサリングアプリケーション内で試聴する時に使うAudio Deviceシェアセットを変更することができます。

### アウトプットデバイスを変更するには:

- メニューで、**Audio > Audio Preferences**をクリックする。

Audio Preferencesダイアログボックスが開く。

- Hardware Devices**グループボックスで、使用するデバイスを選択する。
- OKをクリックして、設定を保存する。

これからはサウンドエンジンが、必ずバスに設定されたデバイスへ、アウトプットします。

デバイス一覧に、現在アクティブなオーディオデバイスプラグインが対応する、全てのデバイスが含まれます。複数のプラグインからアクセスできるハードウェアは、この一覧に何回も出てくる可能性があります。Defaultのままの場合は、そのバスが使うオーディオデバイスプラグインが、デバイスを選択します。Windowsが対応していないオーディオデバイスプラグインの場合は、Windowsのデフォルトオーディオデバイスを使います。この選択は、現在のユーザー用の設定で、ゲームでは使われません。

## オーディオチャンネルコンフィギュレーションの設定

### システムのデフォルトチャンネルコンフィギュレーション

- メニューで、**Audio > System Default Channel Configuration** をクリックする。
- Wwiseはデフォルトで、Windowsのコントロールパネルのスピーカー設定を使用する。Windows のコントロールパネルで選択した値を選ぶには、本オプションを選択する。

### ステレオチャンネルコンフィギュレーション（スピーカー）

- メニューバーで、**Audio > Stereo Channel Configuration (Speakers)** をクリックする。
- パンのルール（スピーカーやヘッドフォン）については「[スピーカーとヘッドフォンのパンニングルール](#)」を参照。

### ステレオチャンネルコンフィギュレーション（ヘッドフォン）

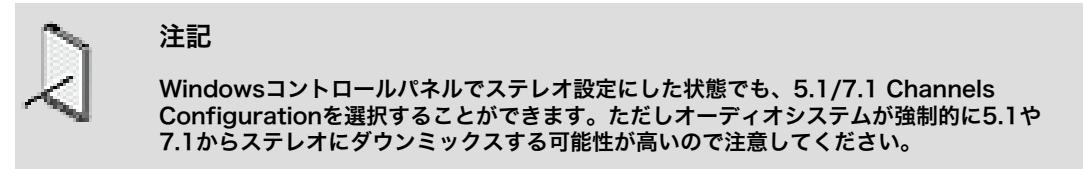
- メニューバーで、**Audio > Stereo Channel Configuration (Headphones)** をクリックする。
- パンのルール（スピーカーやヘッドフォン）については「[スピーカーとヘッドフォンのパンニングルール](#)」を参照。

## 5.1チャンネルコンフィギュレーション

- メニューで、Audio> 5.1 Channel Configuration をクリックする。

## 7.1チャンネルコンフィギュレーション

- メニューで、Audio> 7.1 Channel Configuration をクリックする。



## Music Trackのルックアヘッドタイムの設定

Wwiseでオーサリング中のオーディオ再生は、常にストリーミングです。既にストリーミングを指定してMusic Trackのルックアヘッド（先読み）タイムをMusic Track Property Editorで設定してあれば、ストリーミング中にその値が使われます。ストリーミングオプションが指定されていないと、Music Trackのデフォルトのルックアヘッドタイムである200ms（ミリ秒）が適用されます。柔軟性を高めて再生中のシンクやボイススターべーションの問題を避けるために、Music Trackのルックアヘッドタイムをユーザープリファレンスで設定できます。

### Music Trackのルックアヘッドタイムを設定するには:

- メニューで、Audio > Audio Preferencesをクリックする。

User Preferencesダイアログボックスが開く。

- グループボックスSound EngineにあるMusic Track look-ahead time (ms)の入力欄に、ルックアヘッドタイムをミリ秒単位で入力する。
- OKをクリックして、設定を保存する。

これで、ストリーミングの詳細設定をしていないMusic TrackをWwiseで再生する時は、サウンドエンジンがこのルックアヘッドタイムを使いストリーミングデータをシークして音楽を再生する。

## Documentation Preferencesの設定

設定によって、Wwise HelpをローカルのCHMファイルから開くことも、audiokinetic.comにあるヘルプドキュメントのオンライン版から開くこともできます。どちらのソースタイプも、英語版と日本語版があります。

### Wwise Documentationプリファレンスを設定するには:

- メニューで、Project > User Preferencesをクリックする。

User Preferencesダイアログボックスが開く。

- グループボックスDocumentationのSourceオプションで、CHMを選択すると、ローカル保存してあるWwiseコンパイルのヘルプファイルを使ったコンテ

キストヘルプが開き、audiokinetic.comを選択すると、Audiokineticのオンラインウェブドキュメントを使ったコンテキストヘルプが開く。

3. English、Simplified ChineseまたはJapaneseを、Languageオプションで選択する。

ここで選択したランゲージで、Wwise Help（コンテキストヘルプを含む）と Wwise User's Guideヘルプドキュメントが表示される。

4. OKをクリックして、設定を保存する。

## 作業スピードを速める方法

Wwiseの基本操作に慣れてきたら、より高度な機能も使い始めてください。特定の操作やコマンドを簡単にアクセスするための以下のツールを活用すれば、生産性も向上します。

- 「ショートカットキーを使う」
- 「ショートカットメニューを使う」
- 「Batch Rename機能を使う」

### ショートカットキーを使う

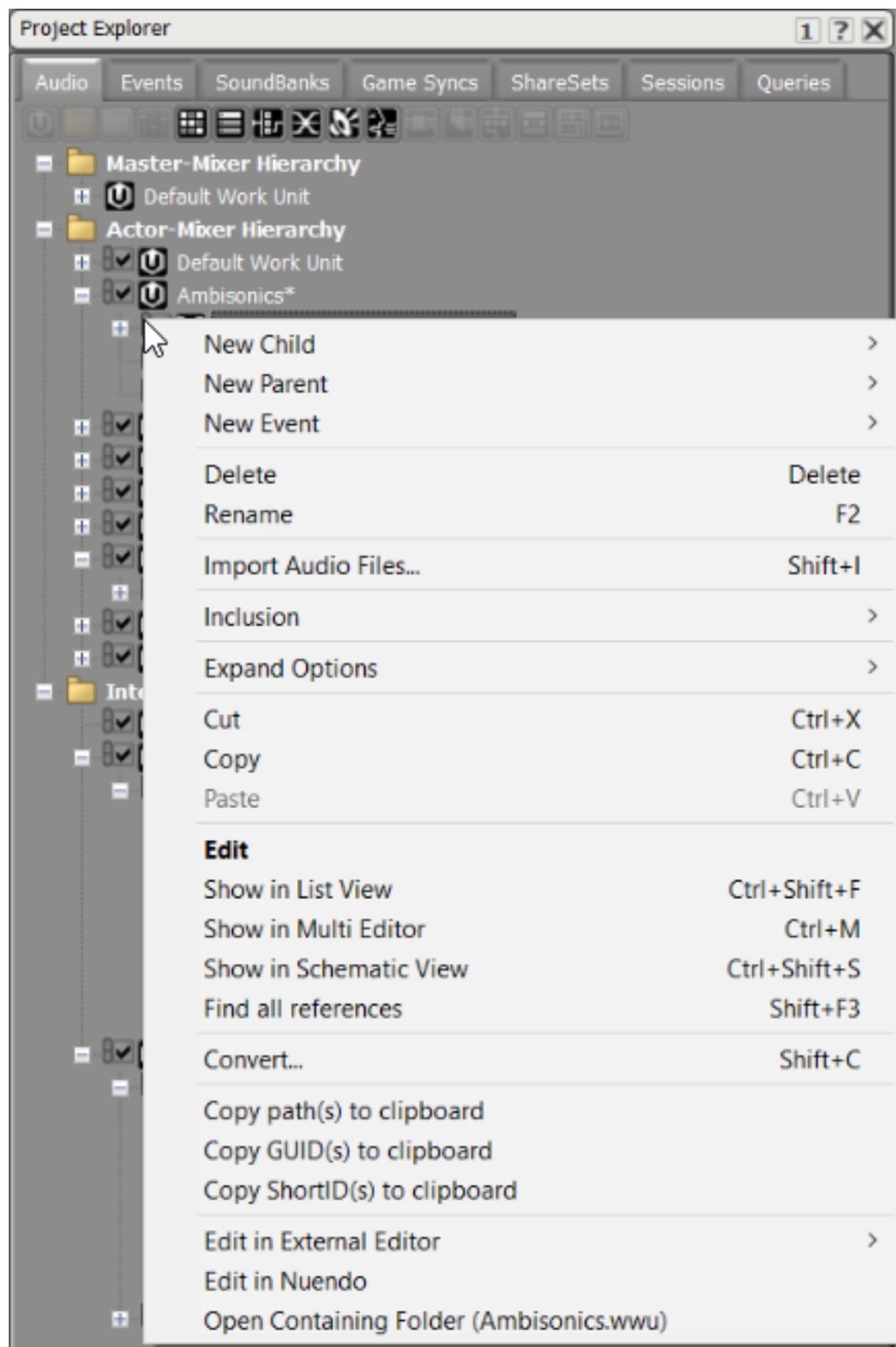
Wwiseで使う多くのコマンドや操作に、1つのキー、または複数のキーの組み合わせがショートカットキーとして設定されています。これらのアクションやコマンドは、いずれもマウス操作以外にもキーボードショートカットを使って実行できます。例えば、プロジェクトを保存する時にFile > Saveをクリックする代わりに、Ctrl+Sを押します。

全てのキーボードショートカットを見るには[付録C ショートカット](#)を参照してください。

### ショートカットメニューを使う

ショートカットメニューとはオブジェクト、プロジェクトのエレメント、または Wwiseのインターフェースの特定部分に関連するコマンドのリストです。ショートカットメニューは、オブジェクトやインターフェースの特定部分を右クリックすると表示されます。

例えば、Project Explorerにあるオブジェクトを右クリックすると、ショートカットメニューが開き、複数のコマンドが表示されます。このコマンドを使って、オブジェクトのカット、コピー、ペーストなどや、親オブジェクトや子オブジェクトの作成、特定オブジェクトに関連するイベントの作成、サウンドオブジェクトのコンバージョンなどを実行できます。



メニューの内容は状況に応じて変化しますが、どのメニューにも表示されるコマンドがいくつかあります。カット、コピー、ペースト、消去、名前変更など、Windowsの標準的なコマンドのほか、次のようなWwise独自のコマンドの多くが、オブジェクトやインターフェースの上で右クリックすると表示されます。

- **Edit** : 選択したオブジェクトやプロジェクト要素を対象エディタにロードする。
- **Find in Project Explorer** : 選択したオブジェクトやプロジェクト要素を、Project Explorerでハイライト表示する。このコマンドを使う時、Project Explorerが属するシンクグループがあれば、それを指定する必要がある。
- **Show in List View** : 選択したオブジェクトをListビューで表示する。
- **Show in Multi Editor** : 選択したオブジェクトをMulti Editorで表示する。
- **Show in Schematic View** : 選択したオブジェクトをSchematicビューで表示する。
- **Find All References** : 現在のオブジェクトへの直接レファレンスがある、全てのプロジェクトエレメントのリストを表示する。直接レファレンスのリストはReferenceビューに表示される。
- 以下のオプションは、Shiftを押しながらショートカットメニューを開いた場合のみ表示される。
  - **Copy path(s) to clipboard**: 選択したオブジェクト（単数または複数）のフルパスをクリップボードにコピーする。
  - **Copy GUID(s) to clipboard**: 選択したオブジェクト（単数または複数）の固有GUIDをクリップボードにコピーする。
  - **Copy ShortID(s) to clipboard**: 選択したオブジェクト（単数または複数）のShortIDをクリップボードにコピーする。
- **Edit in external editor** : User Preferencesで外部エディタを設定した場合、その外部エディタのリストを開く。
- **Edit in Nuendo**: そのエレメントを、Nuendoで開く。このオプションを選択できるのは、Nuendo Game Audio Connectプラグインがインストールしてあり、Nuendo Connection Settings...を正しく設定してあり（Projectメニューで開く）、Nuendoが実行中である場合に限る。
- **Create in Nuendo** (Music Segmentオブジェクトだけに提供される) : Steinberg Hubが起動して、Nuendoで新しいプロジェクトを作成できます。Nuendoにプロンプトが表示され**Do you want to import the transferred Segment into this Project** (転送されたSegmentをこのプロジェクトにインポートしますか)と聞かれます。あなたのMusic SegmentをProject Zoneに追加するには、Yesをクリックします。
- **Open containing folder** : Windows Explorer（またはMac Finder）でオブジェクトの親エレメントの場所を表示する。

### Batch Rename機能を使う

Batch Renameビューの高度な同時置換、削除、挿入機能を使って、編集可能なオブジェクトの名前や関連する備考欄（Notes）の内容を一括変更できます。これによりオブジェクト名の個別変換にかかる時間と手間が省け、場合によってはエラーの原因も排除できます。

Batch Renameビューの説明と事例を、以下のページに示します。

- ・「Batch Renameビューを開く」
- ・「Batch Rename Settingsを指定する」
- ・「Batch Rename変更を適用する」

### Batch Renameビューを開く

名前を変更するオブジェクトがビューにロードされているかどうかにかかわらず、様々な場所からBatch Renameビューを開けます。

オブジェクトを表示せずにBatch Renameビューを開くには:

1. Views > Batch Renameを選択するか、ショートカットを使う。デフォルトショートカットは、Ctrl+F2。

Batch Renameビューが開き、何もないPreviewパネルが表示される。

選択したオブジェクトを表示させたBatch Renameを開くには:

1. Project Explorer（またはList Viewなどその他のオブジェクトビュー）で、名前変更を行いたい編集可能なオブジェクトを選択する。
2. ショートカット（右クリック）メニューでBatch Rename...を選択するか、キーボードショートカットを使う。

Batch Renameビューが開き、Previewパネルに選択したオブジェクトが表示される。

Batch Renameビューにオブジェクトを追加するには:

1. Batch Renameを開いた状態で、Project Explorer（またはその他のオブジェクトビュー）で名前変更を行いたい編集可能なオブジェクトを選択する。
2. ショートカットメニューでBatch Rename...を選択するか、キーボードショートカットを使うか、選択したオブジェクトをPreviewパネルにドラッグする。

Previewパネルに表示されていた前のオブジェクトが、選択したオブジェクトに置き換えられる。



### 関連トピック

- ・「Batch Rename Settingsを指定する」
- ・「Batch Rename変更を適用する」

### Batch Rename Settingsを指定する

「Batch Renameビューを開く」の後は、Batch RenameビューのSettingsパネルで行うアクションによって、Previewパネルに表示されたオブジェクトに適用する変更が定義されます。

Settingsパネルは、以下の3エリアに分かれています。

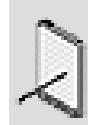
- Replace # 詳しくは「[テキストを置換する \(Replace\)](#)」を参照。
- Remove # 詳しくは「[テキストを削除する \(Remove\)](#)」を参照。
- Insert # 詳しくは「[テキストを挿入する \(Insert\)](#)」を参照。

### テキストを置換する (Replace)

Batch Rename Settingsパネルの3セクションのうち、最初のセクションがReplaceです。このセクションで、非常に単純な置換パターンから非常に複雑な置換パターンまでを使ってオブジェクト名やNotesのテキストを置換できます。

#### テキストを置換するには:

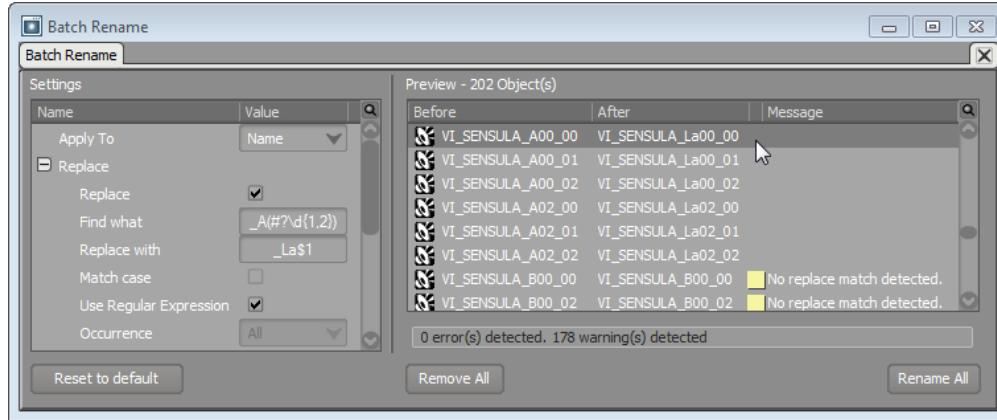
1. 名前変更を適用する対象がリストにあるオブジェクトの名前かコメントかを、Apply Toリストで指定する。
2. Replaceを選択する。
3. 検索するテキストを、場合によって正規表現 (regular expression) パターンの形式でFind whatフィールドに入力する。
4. 置換テキストを、場合によって正規表現の後方参照 (regular expression back reference) の形式でReplace withフィールドに入力する。
5. Find whatに入力したテキストと同じ大文字・小文字を使ったテキストのみを検索するには、Match caseを選択する。



#### 注記

Match caseは正規表現パターンに適用されず、必要に応じて大文字・小文字を識別するために設定します。

6. Use Regular Expressionを選択すると、WwiseがFind whatやReplace withの入力内容を正規表現シンタックスと解釈する。
7. Occurrenceリストで、指定パターンにマッチする件数が複数ある場合の対応を指定する:
  - First - 最初にマッチした項目だけを含める。
  - Last - 最後にマッチした項目だけを含める。
  - All - オブジェクト名やコメントのパターンがマッチするものを全て含める。
8. 変更内容を確認してから、Rename Allをクリックして内容を置換する。



### Example: 正規表現に置き換える

Wwise Sample Projectを例に、使い方を説明します。サンプルプロジェクトのActor-Mixer Hierarchyにある**MIDI Work Unit**には、何十個ものMIDIオブジェクトがあり、それぞれの名前の一桁として絶対ノート、つまりC00、D00、E00、F00、G00、A01、B01、C02などが使われています。さて、プロジェクトがフランスの自社スタジオに委託されることになり、そこでは固定ドのソルフェージュの「ドレミ」を使っているとします。全てのオブジェクト名を変えてあげた方が、フランスのオーディオデザイナー達は喜ぶでしょう。次のように変更します：

1. Ctrlを押してMIDI Work Unitをクリックする。

MIDI Work Unit階層が開き、中に多数のオブジェクトが表示される。

2. Project ExplorerでKalimba、Sansula、Voice、Wood Sansulaのオブジェクトを選択する。次にショートカットメニューを開いてBatch Rename...を選択する。

Batch Renameビューが開き、選択したオブジェクトがPreviewパネルに表示される。

3. SettingsパネルのApply ToがNameに設定されていない場合は、設定する。

PreviewパネルのBefore列に現在のオブジェクト名がリスト表示される。

4. Replaceを有効にして、その数列下にあるUse Regular Expressionを有効にする。

Find whatとReplace withフィールドが有効になり、正規表現の入力や解釈が可能になる。

5. Find whatフィールドに`_C(\#\d{1,2})`を入力する。次にPreviewパネルをクリックすると予測結果を確認できる。

名前に`_C`が使われ、#の有無にかかわらず次に数字1桁または2桁が続くオブジェクトは全て、マッチした文字が削除された状態でAfter列にリスト表示されるので、いくつかのオブジェクトにはエラーメッセージResulting name is already used by a sibling object or is otherwise reservedが表示される。他のオブジェクトは全てNo replace match detectedのメッセージが表示される。

6. Replace with フィールドに \_Do\$1 を入力する。次に Preview パネルをクリックすると予測結果を確認できる。これで、マッチする C ノートが削除されるだけでなく、After 列では C の代わりに Do が表示される。

7. Rename All をクリックして変更内容を適用する。

Preview パネルに、マッチした全オブジェクト、それぞれの更新された名前、そして Successfully renamed メッセージが表示される。

続いて残り 6 つのノートでも、同じ操作を繰り返す。

正規表現である \_C(#?\d{1,2}) とそれを置き換える \_Do\$1 がどのように組み合わされるかを、下表に詳しく説明します。

Before	After	Explanation
VI_VOICE_33_C#6	VI_VOICE_33_Do#6	<ul style="list-style-type: none"><li>_C は文字通りオブジェクト名の _C とマッチするので、正規表現でない置換と同じ置換である。</li><li>#?、つまり # が 0 個または 1 個あるかのマッチは、# が 1 個あるのでマッチする。</li><li>\d{1,2}、つまり数字 1 桁または 2 桁は、6 とマッチする。</li><li>#?\d{1,2} が () で囲まれているので、マッチする # と 6 は同一グループにまとめられる。</li></ul> <p>置換後は:</p> <ul style="list-style-type: none"><li>マッチした部分は _Do に置換され</li><li>\$1、つまり最初のグループの値は #6。</li></ul>
VI_WOOD_SANSULA_C00_HI	VI_WOOD_SANSULA_Do00_HI	<ul style="list-style-type: none"><li>_C は文字通りオブジェクト名の _C とマッチするので、正規表現でない置換と同じ置換である。</li><li>#?、つまり # が 0 個または 1 個あるかのマッチは、# が 0 個あるのでマッチする。</li><li>\d{1,2}、つまり数字 1 桁または 2 桁は、00 とマッチする。</li><li>#?\d{1,2} が () で囲まれているので、マッチする 00 は同一グループにまとめられる。</li></ul> <p>置換後は:</p> <ul style="list-style-type: none"><li>マッチした部分は _Do に置換され</li><li>\$1、つまり最初のグループの値は 00。</li></ul>

## 関連トピック

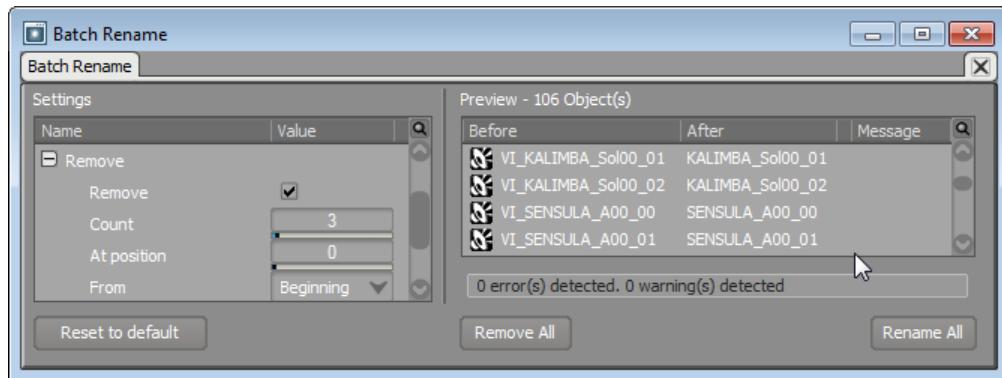
- 「Batch Rename ビューを開く」
- 「Batch Rename Settings を指定する」
- 「Batch Rename 変更を適用する」

### テキストを削除する (Remove)

Batch Rename Settingsパネルの3セクションのうち、2つ目のセクションが**Remove**です。このセクションで、簡単な文字ポジションシステムを使ってオブジェクト名やNotesのテキストを削除できます。

#### テキストを削除するには:

1. 名前変更を適用する対象がリストにあるオブジェクトの名前かコメントかを、**Apply To**リストで指定する。  
Preview パネルのBefore列とAfter列に、選択したオブジェクトのテキストが反映される。
  2. **Remove**を有効にする。
- Count、At position、Fromが有効になる。
3. 削除する文字数を指定するためにCountフィールドに整数を入力するか、そのスライダを希望する数値まで移動する。
  4. 削除の処理を行う文字ポジションを指定する。指定するにはAt positionフィールドに整数を入力するか、そのスライダを希望する数値まで移動する。
  5. At positionをカウントする向きを選択する:
    - Beginning (名前またはコメントの始まりから)
    - End (名前またはコメントの終わりから)
  6. 変更内容を確認してから、Rename Allをクリックして内容を削除する。



#### Example: マッチ結果から削除する

例えばWwise Sample Projectの中で、プロジェクトのオブジェクト名に含まれる余分な情報を落としてオブジェクト名を短くするようにプロジェクトディレクタに頼まれたとします。このプロジェクトでは、Kalimba、Sansula、Voice、Wood SansulaのBlend Containerに入っているSFXの名前は全てVirtual Instrumentsを示す"VI\_"で始まっています。これはオブジェクトを定義する上で必要不可欠な情報ではないので、削除します。

1. Batch Renameビューを開き、これらのBlend ContainerのSFXを全てProject ExplorerからPreviewパネルにドラッグする。

"VI\_"で始まるオブジェクト106個がリスト表示される。（違う場合は、**Apply To**リストのバッチ変更の対象が、オブジェクトのNameでありNotesでないことを確認する。）

2. **Settings**の第2階層に移動して、**Remove**を有効にする。

3. **Count**に3を入力する。

4. **At position**を0に指定する。

5. **From**リストで**Beginning**を選択する。

これでオブジェクト名の最初の3文字、つまり"VI\_"が対象となる。（もし「VI」とインストゥルメント名の間にあるアンダースコアだけを削除したければ、**Count**を1に、**At position**を2に指定する。）

6. **Preview**パネルをクリックする。

**After**列が更新されて全てのオブジェクト名から"VI\_"が消え、名前がSENSUSLA\_A00\_00などとなる。

7. **Rename All**をクリックして変更内容を適用する。

**Preview**パネルにBatch Renameの全オブジェクトの更新された名前と**Successfully renamed**メッセージが表示される。

## 関連トピック

- ・「Batch Renameビューを開く」
- ・「Batch Rename Settingsを指定する」
- ・「Batch Rename変更を適用する」

### テキストを挿入する (Insert)

Batch Rename Settingsパネルの3セクションのうち、3つ目のセクションが**Insert**です。このセクションで、簡単な置換や文字ポジションを使ってオブジェクト名やNotesにテキストやprintf format numberパターンを挿入できます。

### テキストなどを挿入するには:

1. 名前変更を適用する対象がリストにあるオブジェクトの名前かコメントかを、**Apply To**リストで指定する。

2. **Insert**を有効にする。

3. 挿入タイプを**Insert what**で指定する:

- ・ **Text: To insert**に入力したテキストをそのまま挿入する。

# start atが無効になる。

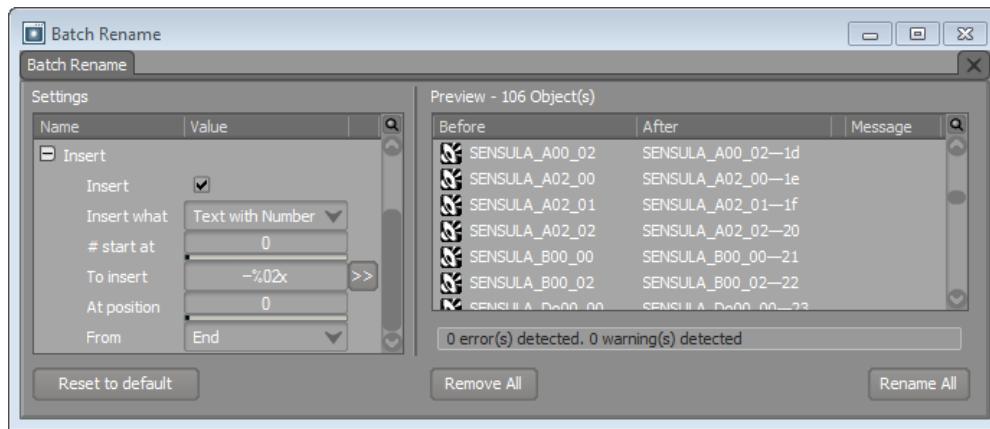
- ・ **Text with Number: To insert**に入力したテキストが挿入され、C++ printf format numberパターンが認識される。

# start at が有効になる。

4. **Text with Number**を選択した場合は、入力したC++ printf format numberパターンを開始させる位置を指定するために# start atフィールドに整数を入力するか、そのスライダを希望する数値まで移動する。

Textを選択した場合は、このフィールドは無効になる。

5. 挿入するテキストをTo insertフィールドで指定する。なおText with Numberを指定した場合は、ここに適切なC++ printf format numberパターンが入力されるものとWwiseが認識する。
  - また、セレクタを使って、一般的に使われるC++ printf format numberパターンの既定テキストを入力することもできる。
6. 挿入の処理を行う文字ポジションを指定する。指定するにはAt positionテキストボックスに整数を入力するか、そのスライダを希望する数値まで移動する。
7. At positionをカウントする向きを選択する:
  - Beginning (名前またはコメントの始まりから)
  - End (名前またはコメントの終わりから)
8. 変更内容を確認してから、Rename Allをクリックして内容を挿入する。



### Example: 数字を挿入する

もう一度、Wwise Sample Projectを例に説明します。例えば、サンプルプロジェクトのActor-Mixer HierarchyにあるMIDI Work UnitのKalimba、Sansula、Voice、Wood Sansulaのブレンドコンテナに入っている全てのバーチャルインストゥルメントSFXをトラッキングしたいとします。それぞれのバーチャルインストゥルメントを、専用のケースにしまってあげると良いでしょう。既に「[Example: マッチ結果から削除する](#)」で、グループ全体を識別するための冒頭の"VI\_"が個々の認識に重要でなかったため、削除しました。そこでQAチームが、個別に番号を振った方がオブジェクト名をトラッキングしやすいと判断したので、これに従いSFXオブジェクトの冒頭に個別番号を挿入することにします。

1. Project ExplorerでこれらのコンテナのSFXオブジェクトを全て選択して、Batch RenameショートカットのCtrl+F2 (デフォルト) を押す。

Batch Renameビューが開き、選択したSFXオブジェクトがPreviewパネルに表示される。

2. NameがApply toオプションとして選択されていることを確認して、InsertをSettingsパネルで有効にする。

挿入フィールド（Insert、Insert what、To insert、At position、From）が有効になる。

3. Text with Numberを選択する。

# start atフィールドが有効になる。

4. #start atは0のままにする。

5. To Insertフィールドにエムダッシュ(em-dash)、次に小文字16進数のprintf文字を2桁以上になるようにゼロパディングしたものを入力する: #%-02x。

6. At positionを0に指定する。

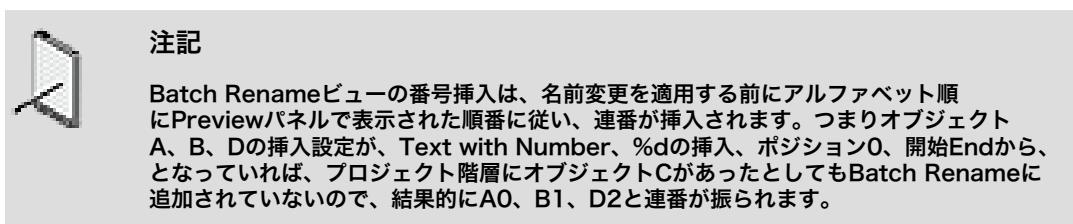
7. からのリストでを選びを終了

8. Previewパネルをクリックする。

9. After列が更新され、例えばSENSULA\_A02\_01#1fのように全てのオブジェクト名の末尾にエムダッシュ(元のオブジェクト名と適切な間隔を確保する役割)と、2桁の16進数が追加されて表示される。

10 Rename Allをクリックして変更内容を適用する。

PreviewパネルにBatch Renameの全オブジェクトの更新された名前とSuccessfully renamedメッセージが表示される。



## 関連トピック

- 「Batch Renameビューを開く」
- 「Batch Rename変更を適用する」

## Batch Rename変更を適用する

「Batch Rename Settingsを指定する」の後に、指定した名前変更の設定を適用する前に、その影響をPreviewパネルで確認するようにします。

### プレビューを確認するには:

- パネル直下のフィールドに表示されるエラーや警告の検知数を確認する。
- メッセージタイプの列でオブジェクトをソートする（Afterの右のタイトルなしの列ヘッダをクリックする）。

リストのオブジェクトが、エラーメッセージのあるもの、警告メッセージのあるもの、メッセージなしのものの順に表示される。

- Before列とAfter列を精査してメッセージがあるオブジェクトに目を通す。

After列に、Settingsパネルの入力内容の結果が表示される。

調整が必要なものがいればそれを隔離する（警告は必ずしも変更の必要性を意味しない）。

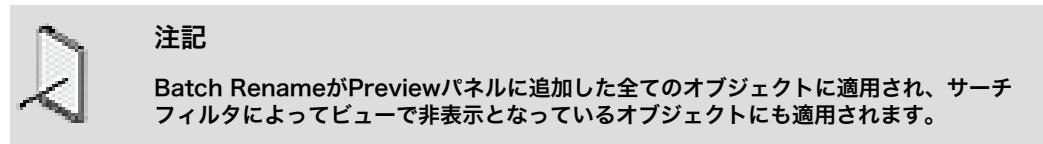
4. 必要に応じてSettingsパネルの入力を変更して、Previewパネルをクリックする。

パネルが更新されて変更内容が反映される。

5. エラーメッセージがなくなり予測結果（After列の表示）に満足できるまで、設定を調整する。

### 名前変更の設定をリストの全オブジェクトに適用するには:

1. Rename Allをクリックして、PreviewパネルのBefore列に表示されたオブジェクト名やコメントを変更する。



リストのオブジェクト名やコメントが全て、After列の値に変更される。

メッセージタイプ表示が緑の四角マークに更新され、Message列にSuccessfully renamedメッセージが表示される。

#### 関連トピック

- [「Batch Renameビューを開く」](#)
- [「Batch Rename Settingsを指定する」](#)

**audio**kinetic

## パート II. プロジェクトを セットアップする



---

3. Wwiseのプロジェクトについて .....	48
概要 .....	49
プロジェクトの管理 .....	51
Project Settingsの設定 .....	55
プロジェクトのデフォルトユーザー設定 .....	80
プロジェクトのトラブルシューティング .....	82
学習コーナー: オブストラクションとオクルージョンについて .....	88
プロジェクトマネジメントに関する、コツとベストプラクティス .....	90
4. プラットフォームの管理 .....	91
概要 .....	92
Platform Managerを使用する .....	92
5. ランゲージの管理 .....	96
概要 .....	97
Language Managerを使う .....	97
6. Workgroupの活用 .....	102
概要 .....	103
プロジェクトをWork Unitで分割 .....	106
プロジェクトファイルのステータス確認 .....	114
Wwiseとソースコントロールシステムを使う .....	116
プロジェクトの不整合の解消 .....	116
Workgroupプラグインを使ったプロジェクトファイルの管理 .....	118
Workgroupに関する、コツとベストプラクティス .....	131
7. メディアファイルの活用 .....	134
概要 .....	135
インポート処理 .....	135
メディアファイルのインポート .....	143
メディアファイルの置換 .....	160
ファイルインポート関連の問題への対処 .....	162
Originalsフォルダのメディアファイルの並べ替え .....	163
キャッシュクリア .....	164
外部エディタでオーディオファイルを編集 .....	166
プラグインを使ってソースを作成する .....	167
メディアファイル管理に関する、コツとベストプラクティス .....	168
8. Actor-Mixer Hierarchyの構築 .....	169
概要 .....	170
プロジェクト階層内のプロパティについて .....	174
Actor-Mixer Hierarchyの構築 .....	182
ランダム化したプロパティ値で、再生を強化 .....	187
Actor-Mixer Hierarchyに関する、コツとベストプラクティス .....	188
9. アウトプットバス構造の構築 .....	193
概要 .....	194
バスのプロパティ設定 .....	198
バス階層の構築例 .....	202

---

## 第3章 Wwiseのプロジェクトについて

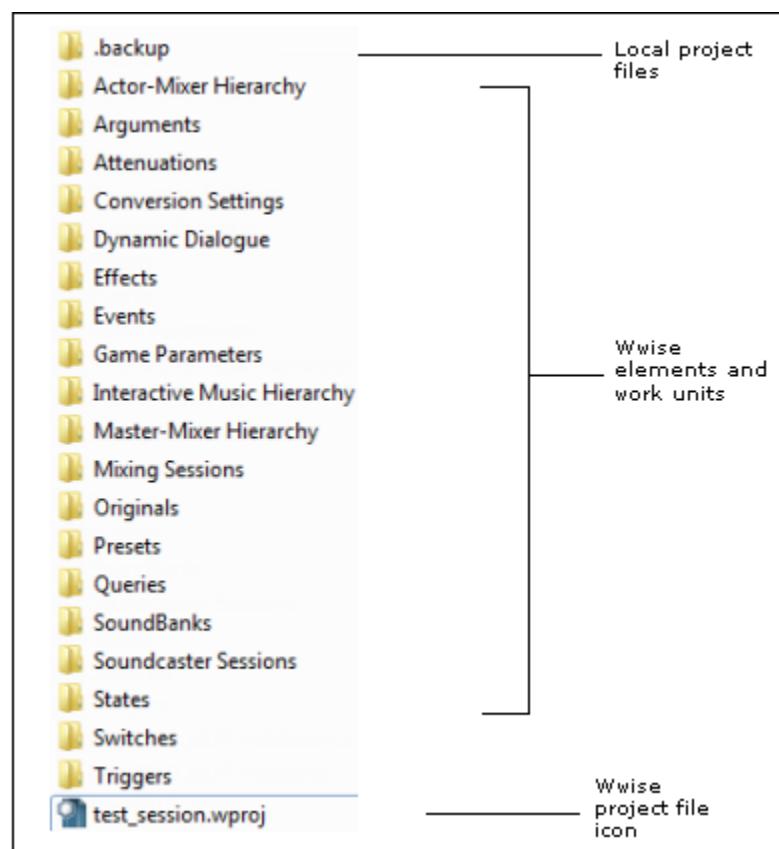
概要 .....	49
プロジェクトの管理 .....	51
Project Settingsの設定 .....	55
プロジェクトのデフォルトユーザー設定 .....	80
プロジェクトのトラブルシューティング .....	82
学習コーナー: オブストラクションとオクルージョンについて .....	88
プロジェクトマネジメントに関する、コツとベストプラクティス .....	90

## 概要

Wwiseでは、全てのアセットと、対応予定のプラットフォームやランゲージのプロパティや動作設定が、1つのプロジェクトに入ります。このプロジェクトにWwiseのエレメントであるイベント、プリセット、ログ、シミュレーションなどが入り、生成されたSoundBankも含まれます。つまり、全ての作業成果がプロジェクトに入るので、チームで作業している場合は、メンバー全員の作業成果が含まれることになります。

1つのゲームに対して、Wwiseで使うプロジェクトは1件だけです。ただし、1つのプロジェクトを複数のWork Unitに分割することができます。Work Unitについては「[プロジェクトをWork Unitで分割](#)」を参照してください。

プロジェクト開発のスタート地点となるのが、プロジェクトを作成して開くProject Launcher画面です。プロジェクトを作成すると、ワークステーションやネットワーク上の任意の場所に、複数のフォルダが作成されます。



Wwiseプロジェクトは、プロジェクトの複数のバージョンやユーザーを効率的に管理するために、各種プロジェクトエレメントのXMLファイルが入った複数のフォルダによって構成されています。典型的なプロジェクトフォルダに含まれるフォルダを、以下に示します。

- **.cache** - プロジェクトにインポートしたSFXアセット、ボイスアセット、プラグインアセットなどの、コンバージョン後のバージョン。ソースコントロールシステムにこのフォルダを含めてはいけない。

- **Actor-Mixer Hierarchy**— プロジェクトのサウンド構造やモーション構造の、デフォルトワークユニットやユーザーが作成したワークユニット。
- **Attenuations** - プロジェクトの減衰用のDefault Work Unitやユーザーが作成したWork Unit。
- **Conversion Settings** - プロジェクトのコンバージョン設定ShareSetの、Default Work Unitやユーザーが作成したWork Unit。
- **Dynamic Dialogue**— プロジェクトのダイアログイベントの、デフォルトワークユニットやユーザーが作成したワークユニット。
- **Effects** - プロジェクトのエフェクトShareSetの、Default Work Unitやユーザーが作成したWork Unit。
- **Events** - プロジェクトのイベントの、Default Work Unitやユーザーが作成したWork Unit。
- **Game Parameters** - Game Parameterの、Default Work Unitやユーザーが作成したWork Unit。
- **Interactive Music Hierarchy**— プロジェクトのミュージック構造の、デフォルトワークユニットやユーザーが作成したワークユニット。
- <e0>Master-Mixer Hierarchy</e0>— プロジェクトのアウトプットルーティングの、デフォルトワークユニットやユーザーが作成したワークユニット。
- **Mixing Sessions** - ミキシングセッションの、Default Work Unitやユーザーが作成したWork Unit。
- **Originals** - プロジェクトにインポートしたアセットなどの、オリジナルの正確なコピー。
- **Presets** - プロジェクトのプリセットの、Default Work Unitやユーザーが作成したWork Unit。
- **Queries** - クエリの、Default Work Unitやユーザーが作成したWork Unit
- **SoundBanks** - SoundBankの、Default Work Unitやユーザーが作成したWork Unit。プロジェクトのSoundBankを生成すると、新規Generated SoundBanks フォルダが表示される。
- **Soundcaster Sessions** - サウンドキャスターセッションの、Default Work Unitやユーザーが作成したWork Unit。
- **States** - Stateの、Default Work Unitやユーザーが作成したWork Unit。
- **Switches** - Switchの、Default Work Unitやユーザーが作成したWork Unit。
- **Triggers** - Triggerの、Default Work Unitやユーザーが作成したWork Unit。
- **IncrementalSoundBankData.xml** - プロジェクトにインポートしたSFXアセット、ボイスアセット、プラグインアセットなどの、コンバージョン後のバージョン。ソースコントロールシステムにこのフォルダを含めてはいけない。
- **.validationcache** - 現在のXMLスキーマバージョンに対して検証済みの、プロジェクトの全ファイルのリスト。検証済みのファイルを把握しておくことで、Wwiseでプロジェクトをロードする時に、そのファイルを毎回検証する必要がない。これにより、Wwiseのプロジェクトロード時間が大幅に削減できる。ソースコントロールシステムにこのフォルダを管理してはいけない。
- **.wsettings** - 現在のプロジェクト用のデフォルトコンバージョン設定や、他のオブジェクト設定。これらの設定はユーザー別に保存される。ソースコントロールシステムにこのフォルダを管理してはいけない。

- **.wproj** - Wwiseのプロジェクトファイル。プロジェクトアイコンをダブルクリックすると、プロジェクトが開く。

プロジェクトを作成した後に、全体を複数のWork Unitに分けて、ゲームのデザインに従いアセット構造を構築し始めます。同時に、Master-Mixer Hierarchyでプロジェクトのルーティング構造を構築し、プロジェクトのゲームシンクも作成します。

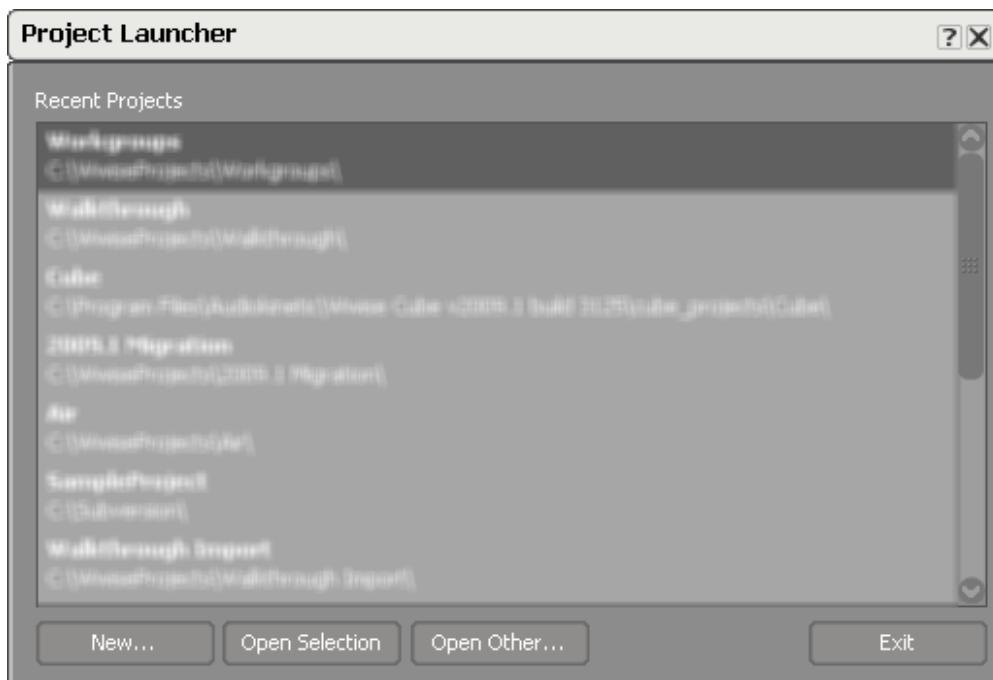
これらのフォルダ以外にも、プロジェクトのインテグレティレポートのファイルや、SoundBankのプロジェクトヘッダファイルなど、他のファイルが存在することもあります。

Wwiseのプロジェクトや各種フォルダは、外部のソースコントロール管理ツールと簡単にインテグレートできるように設計されています。複数のプロジェクトユーザーがいる場合の管理方法については [6章Workgroupの活用](#)を参照してください。

## プロジェクトの管理

ゲームのオーディオやモーションに関連する全てのサウンドアセット、ミュージックアセット、モーションアセット、プロパティ、SoundBankなどが、1つのWwiseプロジェクトに入っているので、プロジェクトの管理には充分に注意する必要があります。他のWindowsプログラムと同様に、Windows Explorerを使ってプロジェクトフォルダを管理できるほか、プロジェクトのコピー、移動、削除などの標準的なタスクも実行できます。ただし、プロジェクトの作成と保存はWwiseから行う必要があります。

WwiseをWwise Launcherから直接開くとProject Launcherが表示されるので、新規プロジェクトを作成するか、既存プロジェクトを開きます。



Wwiseでプロジェクトを開くと、プロジェクトの開閉、作成、保存などのコマンドをProjectメニューでアクセスできます。また、コマンドの多くはキーボードショートカットからも実行できます。全てのショートカットを見るには[付録C ショートカット](#)を参照してください。

## 新規プロジェクトの作成

Project Launcherを使ってNew Projectを開いてから、初めてのWwiseプロジェクトを作成します。Wwiseプロジェクトをロードした後は、New ProjectをメニューのProject > New...からアクセスできます。

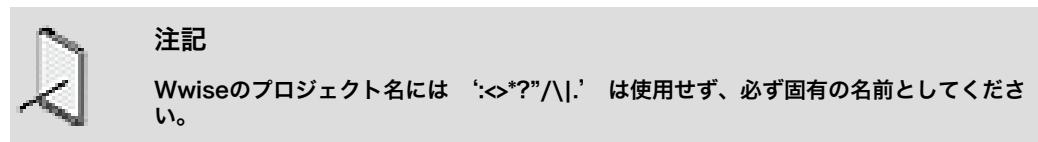
### 新規プロジェクトを作成するには:

1. 以下のいずれかを行う:

- Project Launcherで、Newをクリックする。
- Wwiseメニューバーで、Project > Newをクリックする。
- [Ctrl] + [N]を押す。

New Projectダイアログボックスが開く。

2. 新規プロジェクトの名前を、Nameに入力する。

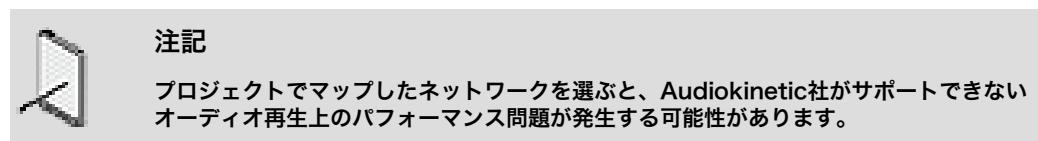


有効なプロジェクト名を入力していくにつれ、Project folderのプロジェクトパスが更新され、OKボタンが有効になる。

3. Locationで、以下のいずれかを行う。

- パスリストから、プロジェクトで以前使ったパスを選択する。
- Browseボタン(...)をクリックして、新規プロジェクトを作成する場所まで移動する。

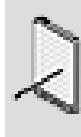
プロジェクトフォルダのために選択したパスが、Project folderに表示される。



4. Original filesにて、以下のいずれかを行う。

- リストから、Use Default Originals Directoryを選択する。このフォルダは、Wwiseプロジェクトフォルダにある。
- Browseボタン(...)をクリックして、Originalsフォルダを作成する場所まで移動する。

5. プラットフォームのリストを確認する。Add...をクリックすると「[プラットフォームを追加するには](#)」、Removeをクリックすると「[プラットフォームを削除するには](#)」、Renameをクリックするとプラットフォームリストで選択した名前の変更ができる。



### 注記

プロジェクトのプラットフォームを後から変えるには、Platform Manager をProjects メニューから開きます。

6. プロジェクトに含めたいアセットグループを、全て選択する。

7. OKをクリックする。

Wwiseプロジェクトフォルダの指定した場所に、新規プロジェクトが作成される。

## 関連トピック

- ・「[プロジェクトを開く、閉じる](#)」
- ・「[プロジェクトの保存](#)」
- ・「[Platform Managerを使用する](#)」

## プロジェクトを開く、閉じる

Wwiseを起動すると、最近開いたプロジェクトが、Recent Projectsリストに表示されます。プロジェクトを開くには、ここで表示されたプロジェクトを選ぶか、改めて他のプロジェクトまで移動して開きます。なお、一度開けるプロジェクトは1つだけです。既にプロジェクトを開いている時に別のプロジェクトを開くには、最初のプロジェクトを閉じる必要があります。

プロジェクトは、開かれる前にWwiseによって検証されます。もしXMLシナリクスエラーがあれば、プロジェクトはロードされません。また、プロジェクトに不整合があれば、Projectダイアログボックス Load Logが開き、不整合点のリストとフィックス案が提示されます。Wwiseの提案するフィックスを受け入れると、フィックスが実行されますが、実際に保存されるのはプロジェクトを保存する時です。このようなプロジェクト関連問題への対処方法については「[プロジェクトの不整合の解消](#)」を参照してください。

非常に大きなプロジェクトで、ロード時間がかかり過ぎるようであれば、いくつかのWork Unitをアンロードして、作業中のWork Unitだけを残すことができます。一旦プロジェクトからアンロードされたWork Unitは、プロジェクトを開く時にロード対象から外されます。これで、プロジェクトを開く時間が大幅に短縮されます。ワークユニットは、いつでも簡単にプロジェクトに再ロードできます。プロジェクトへのWork Unitのロードとアンロードについては「[プロジェクトのWork Unitの、ロードとアンロード](#)」を参照してください。

## Project Launcherからプロジェクトを開くには:

1. 以下のいずれかを行う:

- ・Project Launcherの、Recent Projectsリストから、プロジェクトを選び、Open Selectionをクリックする。
- ・Open Otherをクリックして、開きたいプロジェクトを選択する。

Loading Projectダイアログボックスが、プログレスバーを表示して、選択したプロジェクトがDesigner Layoutで開く。



### Tip

Recent Projectsリストからプロジェクトを選び、ダブルクリックして開くこともできます。

## Wwise内で、プロジェクトを開くには:

- 以下のいずれかを行う:
  - メニューで、Project > Openをクリックする。
  - [Ctrl] + [O]を押す。

Open folderが開き、開きたいプロジェクトまで移動する。

- Wwiseプロジェクトファイルを含むプロジェクトフォルダを選ぶ。
- WPROJファイルを選び、Openをクリックする。

選択したプロジェクトが開く。



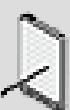
### 注記

開きたいプロジェクトフォルダまで移動してWwiseプロジェクトアイコンをダブルクリックして、プロジェクトを開くこともできます。なお、システム上にインストールされているWwiseのバージョンが2つ以上ある場合は、常にWwiseの最新バージョンが、プロジェクトアイコンで起動します。

## プロジェクトを閉じるには:

- 以下のいずれかを行う:
  - メニューで、Project > Closeをクリックする。
  - [Ctrl] + [F4]を押す。

プロジェクトが閉じ、Project Launcherに戻る。



### 注記

プロジェクト内のオブジェクトを削除した場合、プロジェクトを閉じた後も関連するオーディオファイルは削除されません。これらのファイルを削除するには、キャッシュをクリアする必要があります。詳しくは「[キャッシュクリア](#)」を参照してください。

## 関連トピック

- 「新規プロジェクトの作成」
- 「プロジェクトの保存」
- 「プロジェクトの不整合の解消」
- 「プロジェクトのWork Unitの、ロードとアンロード」

## プロジェクトの保存

現在のプロジェクトを変更すると、タイトルバーのプロジェクト名の横にアスタリスク記号が付き、Project Explorerでも、変更されたWork Unitの横にアスタリスク記号が付きます。定期的にプロジェクトを保存することを推奨します。

1つ以上の読み取り専用Work Unitを変更した場合、プロジェクトを部分的に保存できないことがあります。Work Unitで作業を開始する前に、変更を保存できるか確認して下さい。プロジェクトファイルの管理にソースコントロールシステムを導入している場合は、プロジェクトファイルをチェックアウトしなければ、保存できません。ソースコントロールシステムで管理しているプロジェクトファイルの保存については「[Perforce使用中に、プロジェクトを保存](#)」を参照してください。



### プロジェクトを保存するには:

#### 1. 以下のいずれかを行う:

- Wwiseメニューで、Project > Saveをクリックする。
- [Ctrl] + [S]を押す。

プロジェクトに加えた変更が保存され、プロジェクトに戻る。

### 関連トピック

- 「[新規プロジェクトの作成](#)」
- 「[プロジェクトを開く、閉じる](#)」

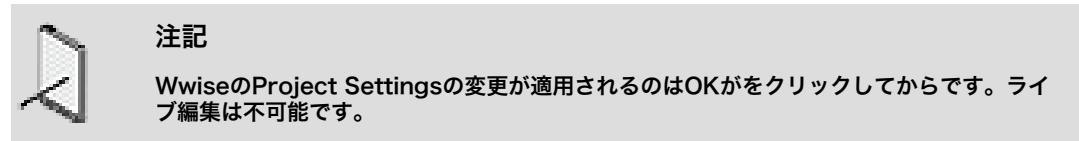
## Project Settingsの設定

Project Settingsダイアログボックスから、一部のプロジェクト設定を行えます。

Project Settingsダイアログボックスには、以下のタブが表示されます:

- [General](#)タブで、ソースコントロールプラグイン、各プラットフォームのボリューム閾値、プロジェクトの各種アセットのOriginalsフォルダの場所、サンプルレート自動検知の設定、イベント名作成の設定などを行う。
- [Source Settings](#) タブで、プロジェクトのデフォルトコンバージョン設定のShareSetや、サンプルレート自動検知設定などを行う。
- [SoundBanks](#) タブで、コンテンツファイル生成の有無、ヘッダファイル生成の有無、SoundBankのイベントの最大減衰の情報、SoundBank名の使用の可否、SoundBankの保存場所など、プロジェクトのSoundBankの詳細を設定するほか、SoundBank生成の前後に実行するカスタム手順も必要に応じて指定する。
- [Logs](#) タブで、コンバージョンやSoundBankのログに表示される警告、エラー、メッセージなどを管理する。

- [Obstruction/Occlusion](#)タブで、プロジェクトのオブストラクションやオクルージョンのボリュームカーブ、LPFカーブ、HPFカーブを設定する。
- [External Sources](#)タブで、外部ソースプラグインで使う外部オーディオソースの入力パスとアウトプットを指定する。
- [Network](#)タブで、ゲームがオーサリングアプリケーションと通信する際に使用する、ポート番号を指定する。
- Custom Propertiesタブで、Actor-Mixer Hierarchyのサウンドオブジェクトやオーディオソースオブジェクトのプロパティを設定する。



## プロジェクトのGeneral Settingsの設定

Project Settingsダイアログボックスの、Generalタブにおいて、以下のタスクを実行できます。

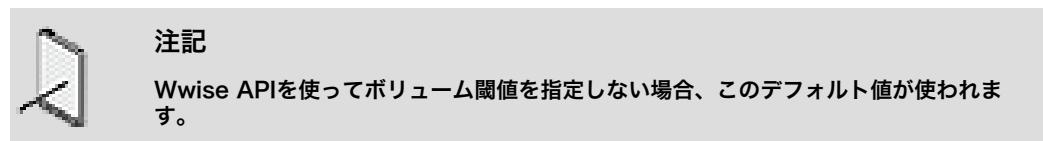
- 「[プロジェクトのボリューム閾値の設定](#)」
- 「[ソースコントロールプラグインの、コンフィギュレーション](#)」
- 「[Originalsフォルダの設定](#)」
- 「[Cacheフォルダの設定](#)」
- 「[Event Creation Settingsの設定](#)」

### プロジェクトのボリューム閾値の設定

ボリューム閾値より小さいボイスは、Property EditorのAdvanced Settingsで設定した動作をとります。Project Settingsダイアログボックスで、プロジェクトのデフォルトボリューム閾値を設定します。Wwise APIを使ったボリューム閾値の指定がない場合は、このデフォルト値が使われます。低レベルのサウンドやモーションオブジェクトの管理方法や、ボリューム閾値に達した時のオブジェクトの動作設定については「[低ボリュームのオブジェクトの管理](#)」を参照してください。

#### プロジェクトのボリューム閾値を設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. グループボックスPlatformsで、サウンドがどのポイントを超えた時にProperty EditorのAdvanced Settingsで設定した動作に管理されるかを指定する。



3. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

## ボイスインスタンスの最大数を指定する

プロジェクトのボリューム閾値の設定と同様に、同時に再生できるボイスインスタンス数も制限されます。プロジェクトで再生するボイスインスタンス数がこの制限を超えると、Property Editorで設定した再生中のオブジェクトのAdvanced Settingsの設定に従います。

オブジェクトのプライオリティをWwiseが確認して、どのボイスを保持すべきかを判断します。プライオリティが最も低いボイスが、バーチャルボイス数を超過した分だけ、バーチャル化されるかキルされ（消され）ます

## 関連トピック

- 「ソースコントロールプラグインの、コンフィギュレーション」
- 「Originalsフォルダの設定」
- 「Cacheフォルダの設定」
- 「サンプルレートの自動検知設定」
- 「プロジェクトのSoundBankの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオプストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」
- 「プロジェクトのトラブルシューティング」
- 「Event Creation Settingsの設定」
- 「あるプラットフォームの設定を、他のプラットフォームにコピーする」

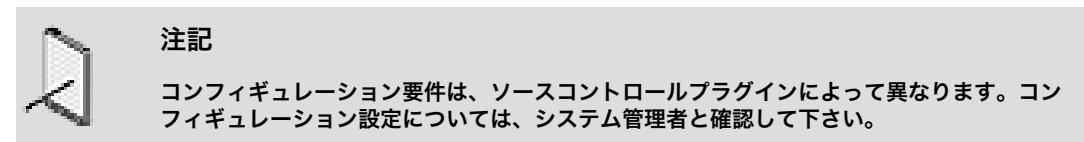
## ソースコントロールプラグインの、コンフィギュレーション

プロジェクトのアセットやファイルをソースコントロールシステムで管理する場合、Workgroup用のプラグインを選べます。使用中のWwiseのバージョンがそのプラグインに対応していれば、ワークスペース用にプラグインコンフィギュレーションを設定することも可能です。

以下の2つのソースコントロールプラグインが、Wwiseに同梱されています。

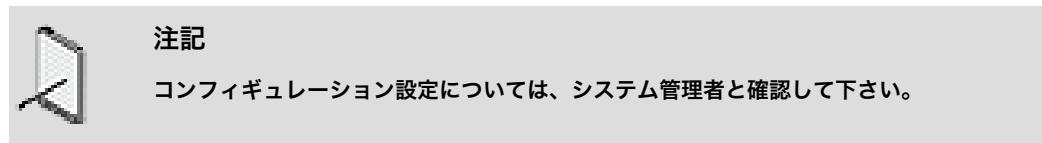
- **Perforce®** - バージョンの互換性については「[Perforce](#)」を参照してください。
- **Subversion** - バージョンの互換性については「[Subversion](#)」を参照してください。

Wwiseが対応しているPerforceやSubversionのバージョン情報について  
は「[Perforce/Subversionの対応バージョン](#)」または[Wwise SDKドキュメンテーション](#)を参照してください。



## プロジェクトのソースコントロールプラグインを設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. グループボックスWorkgroup の、Plug-inリストから、プロジェクトで使うソースコントロールプラグインを選択する。
3. プラグインのコンフィギュレーションを設定するには、Configをクリックする。  
Source Control Plug-in Configurationダイアログボックスが開く。
4. Source Control Plug-in Configurationダイアログボックスの各フィールドに、必要な情報を入力する。



5. OKをクリックして、コンフィギュレーション設定を保存し、Plug-in Configurationダイアログボックスを閉じる。
6. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

## 関連トピック

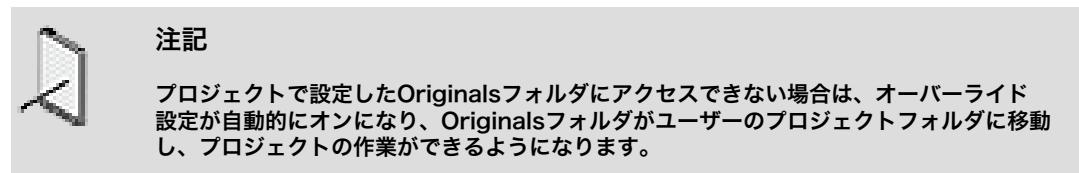
- 「プロジェクトのボリューム閾値の設定」
- 「Originalsフォルダの設定」
- 「Cacheフォルダの設定」
- 「サンプルレートの自動検知設定」
- 「プロジェクトのSoundBankの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」
- 「プロジェクトのトラブルシューティング」
- 「Event Creation Settingsの設定」

## Originalsフォルダの設定

プロジェクトにインポートしたオリジナルのオーディオファイルのコピーは、プロジェクトのOriginalsフォルダに保存されます。フォルダの場所は、プロジェクトを作成する時に決めます。Generalタブで、オリジナルファイルの保存場所を指定できます。プロジェクト全体の保存場所を選択することも、自分専用の別の保存場所を選択することもできます。なお後者は、以下のような場合に非常に便利です。

- 遠隔で作業をしているため、Originalsフォルダにアクセスできない。

- Originalsフォルダの中身を編集する権限がない。
- プロジェクトのOriginalsフォルダの場所を変えずに、Originalsフォルダの一時的な場所を作成する必要がある。



Originalsフォルダについては「[メディアファイルの構造](#)」を参照してください。

### プロジェクトのOriginalsフォルダの場所を設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. プロジェクトのOriginalsフォルダの場所を設定するには、以下のいずれかを行う：

グループボックス**Original Audio Files**に、Originalsフォルダのパスを入力またはペーストする。パスは、絶対パスでもプロジェクトフォルダに対する相対パスでもよい。デフォルトパスは、“Originals”である。

**Browse**ボタン(...)をクリックして、オーディオファイルのOriginalsフォルダを保存する場所まで移動する。

3. 自分専用のOriginalsフォルダの場所を設定するには、以下のいずれかを行う：
  - グループボックス**Original Audio Files**で、Override location for current userオプションを選択する。
  - **Browse**ボタン(...)をクリックして、オーディオファイルのOriginalsフォルダを保存する場所まで移動する。
4. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

### 関連トピック

- 「[プロジェクトのボリューム閾値の設定](#)」
- 「[ソースコントロールプラグインの、コンフィギュレーション](#)」
- 「[Cacheフォルダの設定](#)」
- 「[サンプルレートの自動検知設定](#)」
- 「[プロジェクトのSoundBankの設定](#)」
- 「[ログに表示されるメッセージの管理](#)」
- 「[プロジェクトのオブストラクションやオクルージョンの、カーブ設定](#)」
- 「[外部ソースのインプット・アウトプット場所を指定](#)」
- 「[プロジェクトのデフォルトユーザー設定](#)」

- ・「プロジェクトのトラブルシューティング」
- ・「Event Creation Settingsの設定」

### Cacheフォルダの設定

プロジェクトのCacheフォルダに、オーディオファイルのコンバージョンの過程や、やSoundBank生成の過程で、Wwiseが生成した中間データが保存されます。新規プロジェクトを作成すると、最初はプロジェクトのディレクトリ内の、.cache/に場所が設定されます。Project Settingsダイアログボックスの、Generalタブで、キャッシュファイルの場所を変更できます。プロジェクト全体の保存場所を選択することも、自分専用の別の保存場所を選択することもできます。後者は、以下のような場合に便利です。

- ・遠隔で作業をしているため、Cacheフォルダにアクセスできない。
- ・Cacheフォルダの中身を編集する権限がない。
- ・プロジェクトのCacheフォルダの場所を変えずに、Cacheフォルダの一時的な場所を作成する必要がある。



#### 注記

プロジェクトで設定したCacheフォルダにアクセスできない場合はオーバーライド設定が自動的にオンになり、Cacheフォルダがユーザーのプロジェクトフォルダに移動し、プロジェクトの作業ができるようになります。

Cacheフォルダの場所をオーバーライドした場合、Wwiseを次に起動する時に、新しい場所となっています。しかし、他のユーザーは引き続き、元の場所にあるフォルダにアクセスします。プロジェクトで設定した場所のCacheフォルダに再度アクセスするには、このオプションをオフにして下さい。Cacheフォルダについては「[メディアファイルの構造](#)」を参照してください。



#### 注意

同じCacheフォルダを、複数のユーザーが同時にアクセスすることは、絶対に避けてください。

### プロジェクトのCacheフォルダの場所を設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - ・Projectメニューで、Project Settingsを選択する。
  - ・[Shift]+[K]を押す。
2. プロジェクトのCacheファイルの、フォルダの場所を設定するには、以下のいずれかを行う：
  - ・グループボックス**Cached Audio Files**に、Cacheフォルダのパスを入力またはペーストする。パスは、絶対パスでもプロジェクトフォルダに対する相対パスでもよい。デフォルトパスは、“.cache”である。
  - ・Browseボタン(...)をクリックして、Cacheファイルのフォルダを保存する場所まで移動する。
3. 自分専用のCacheファイルのフォルダの場所を設定するには、以下のいずれかを行う：

- グループボックスCached Audio Filesで、Override location for current userオプションを選択する。
  - Browseボタン(...)をクリックして、Cacheファイルのフォルダを保存する場所まで移動する。
4. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

## 関連トピック

- 「プロジェクトのボリューム閾値の設定」
- 「ソースコントロールプラグインの、コンフィギュレーション」
- 「Originalsフォルダの設定」
- 「サンプルレートの自動検知設定」
- 「プロジェクトのSoundBankの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」
- 「プロジェクトのトラブルシューティング」
- 「Event Creation Settingsの設定」

### Event Creation Settingsの設定

Wwiseのイベントを作成するには、いくつかの方法があります。対象オブジェクトのないイベントをWwiseで作成すると、一般的なイベント名が新しいイベントに付けられます。一方、特定オブジェクト用のイベントを作成した時の新しいイベント名をWwiseでどのようにつくり出すかは、事前に指定できます。

### プロジェクトのEvent Creation Settingsを設定するには:

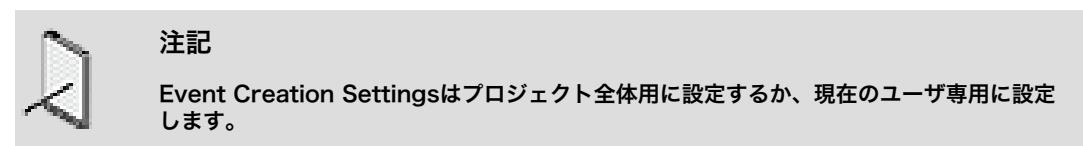
1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. 以下の通り、Event Creation Settingsをプロジェクト用に設定するのか、現在のユーザ専用に設定するのかを選択する:
  - プロジェクト用にEvent Creation Settingsを設定するには、Define settings for projectを選択する。
  - 現在のユーザ専用にEvent Creation Settingsを設定するには、Override settings for current userを選択する。
3. 以下の通り、Event Creation Settingsを選択する:
  - イベント名にアクション名を入れるにはAdd action nameを有効にする。
  - イベント名の大文字・小文字を区別するにはModify caseを有効にする。

詳細は以下を参照。

4. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

デフォルトでオブジェクト名を新しいイベント名として使う。新しいイベント名を以下のオプションで変更できる:

- **Add action name:** 有効にすると、イベントのアクション名がイベント名に含まれる。
  1. set as prefix アクション名がオブジェクト名の前に入る:  
`<action_name>_<object_name>`
  2. set as suffix アクション名がオブジェクト名の後に入る:  
`<object_name>_<action_name>`
- **Modify case:** 有効にすると、イベント名の大文字・小文字が区別される。
  1. all lowercase: 全て小文字のイベント名となる。
  2. all uppercase: 全て大文字のイベント名となる。



## 関連トピック

- 「プロジェクトのボリューム閾値の設定」
- 「ソースコントロールプラグインの、コンフィギュレーション」
- 「Originalsフォルダの設定」
- 「サンプルレートの自動検知設定」
- 「プロジェクトのSoundBankの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」
- 「プロジェクトのトラブルシューティング」
- 「Cacheフォルダの設定」

## プロジェクトのコンバージョン設定

Project Settingsダイアログボックスの、Source Settingsタブで、以下のタスクを実行できます。

- 「コンバージョン設定のデフォルト指定」
- 「サンプルレートの自動検知設定」

## コンバージョン設定のデフォルト指定

プロジェクトのコンバージョン設定の複数のShareSetを作成した後、デフォルトとして使うShareSetを指定します。

デフォルトのコンバージョン設定ShareSetは、以下の状況で使用します。

- 新規オブジェクトを作成した時 - デフォルトShareSetは、新規オブジェクトが最上位の親オブジェクトである場合に限り、適用されます。新規オブジェクトが他のオブジェクトの子であれば、親に設定されたコンバージョン設定を継承します。
- SoundBankを生成する時 - オブジェクトにコンバージョン設定ShareSetがアサインされていない場合、SoundBank生成前のオブジェクトのコンバージョン処理に、デフォルトShareSetが適用されます。

### デフォルトのコンバージョン設定ShareSetを指定するには:

- 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
    - Projectメニューで、Project Settingsを選択する。
    - [Shift]+[K]を押す。
  - Source Settingsタブに切り替える。
  - グループボックスDefault Conversion Settingsの、Browseボタン(...)をクリックする。
- Project Explorer - Browserが開く。
- プロジェクトのデフォルトコンバージョン設定に使う、ShareSetを選択する。
  - OKをクリックする。

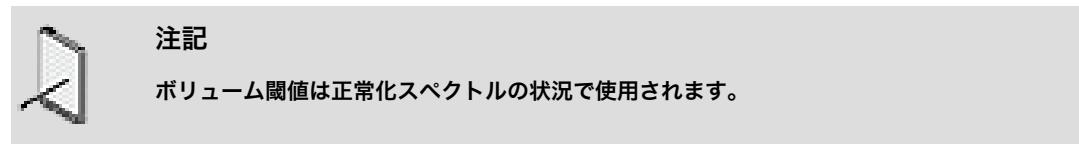
### 関連トピック

- 「プロジェクトのボリューム閾値の設定」
- 「ソースコントロールプラグインの、コンフィギュレーション」
- 「サンプルレートの自動検知設定」
- 「プロジェクトのSoundBankの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」
- 「プロジェクトのトラブルシューティング」

### サンプルレートの自動検知設定

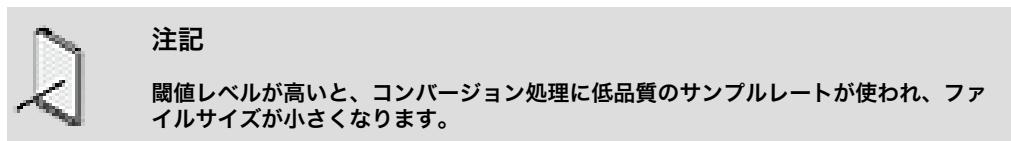
メディアファイルの最適なサンプルレートを決める作業は、手間がかかり必ずしも簡単ではありません。この処理を速くするために、Wwise上で各ファイルを高速フーリエ変換(FFT)アルゴリズムで分析することができます。Wwiseは非常に基本的なレベルで、FFTを使いハニング窓で音波を一部ずつ分析し、メディアファイルのスペクトル分析を生成します。ファイルのコンバージョンに最も適したサンプルレートを判断するための周波を、カットオフボリュームまたは閾値で明らかにします。プロジェクト設定の1つとして、FFTアルゴリズムで使用するハニング窓のサイズや、3つの品質基準(High、Medium、Low)のそれぞれの閾値を設定しま

す。この閾値は、サンプルレートのコンバージョン方式として、Auto High、Auto Medium、Auto Lowを選択した時に適用されます。



### プロジェクトのサンプルレート自動検知設定を設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. グループボックス**Sample Rate Automatic Detection**の、FFT窓サイズリストから、音波を分析するためにFFTアルゴリズムで使用するハニング窓のサイズを選択する。
3. グループボックス**Volume Thresholds**で、以下の品質オプションごとに、FFTアルゴリズムが採用するカットオフボリュームレベルを指定する。
  - **Low quality** - ファイルのコンバージョンに最適なサンプルレートを決めるために使う周波数を解明する、遮断ボリュームレベル。閾値Low quality（低品質）は、コンバージョンのサンプルレートとしてAuto Lowオプションを選択した時に適用される。
  - **Medium quality** - ファイルのコンバージョンに最適なサンプルレートを決めるために使う周波数を解明する、遮断ボリュームレベル。閾値Medium quality（中品質）は、コンバージョンのサンプルレートとしてAuto Mediumオプションを選択した時に適用される。
  - **High quality** - ファイルのコンバージョンに最適なサンプルレートを決めるために使う周波数を解明する、遮断ボリュームレベル。閾値High quality（高品質）は、コンバージョンのサンプルレートとしてAuto Highオプションを選択した時に適用される。



4. OKをクリックして、設定を保存し、Project Settingsダイアログボックスを閉じる。

### 関連トピック

- 「[プロジェクトのボリューム閾値の設定](#)」
- 「[ソースコントロールプラグインの、コンフィギュレーション](#)」
- 「[Originalsフォルダの設定](#)」
- 「[Cacheフォルダの設定](#)」
- 「[プロジェクトのSoundBankの設定](#)」
- 「[ログに表示されるメッセージの管理](#)」

- ・「プロジェクトのオプストラクションやオクルージョンの、カーブ設定」
- ・「外部ソースのインプット・アウトプット場所を指定」
- ・「プロジェクトのデフォルトユーザー設定」
- ・「プロジェクトのトラブルシューティング」

## プロジェクトのSoundBankの設定

それぞれのプラットフォームやランゲージ用に様々なSoundBankを生成する前に、SoundBankの詳細設定を決める必要があります。以下のSoundBank設定は、プロジェクトレベルで定義できます。

- ・「[SoundBankのプロジェクト設定](#)」 - 生成されたSoundBankに、どの情報を含めるかを設定。
- ・「[SoundBankの保存場所の指定](#)」 - ハードドライブまたはネットワークの、どの場所にSoundBankを保存するかを設定。
- ・「[SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定](#)」 - SoundBank生成の直前に実施するタスクの設定。
- ・「[SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定](#)」 - SoundBank生成の直後に実施するタスクの設定。

これらの設定をプロジェクトレベルで決めますが、このプロジェクト設定をオーバーライドしてカスタマイズしたユーザー設定を作成することもできます。SoundBankのプロジェクト設定をオーバーライドする詳細については「[SoundBankのカスタム属性を設定する](#)」を参照してください。

## SoundBankのプロジェクト設定

SoundBankを生成する前に、生成プロセスにどの情報を含めるのか、どのように含めるのか、またどのようなフォーマットで生成するのかなどを、決める必要があります。設定内容は、SoundBank内のデータやメディアを、ゲームがどのようにアクセスするのかによって、決まります。

## SoundBankのプロジェクト設定

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. SoundBanksタブに切り替える。
3. グループボックス**SoundBank Settings**で、以下のオプションを必要に応じて選択し、SoundBankのカスタム設定を決める。
  - **Allow SoundBanks to exceed maximum size:** SoundBankのサイズが、指定した最大サイズを超えて、生成する場合。
  - **Generate SoundBank content files:** 各SoundBankの、コンテンツリストのファイルを作成する場合。コンテンツファイルには、イベント、バス、State、Switchの情報や、ストリーム用やメモリ用の全オーディオファイルのリストが含まれる。

- **Generate header file:** イベント名、State名、Switch名、Game Parameter名を、それぞれのIDにマッピングするヘッダファイルを作成する場合。
  - **Generate max attenuation info for events (イベントの最大減衰に関する情報を生成)** : SoundBanksInfo.xmlファイルに、各イベントの最大減衰に関する情報を含める場合。
  - **Estimated duration** 各イベントの SoundBanksInfo.xml ファイルに、サウンドを永久にループさせる、もしくは1回のみのサウンドかを設定するだけでなく、各イベントの推定最長ならびに最短持続時間を含める場合。
  - **Use SoundBank Names :** 生成する「.bnk SoundBank」ファイルのファイル名や、バンクの中で他のSoundBankを参照する時に、SoundBank名を使うのか（チェック入り）、またはIDを使うのか（チェックなし）の設定。
4. ヘッダファイルを生成する場合は、その保存先を決める必要がある。そのためには以下のいずれかを行う：
- テキストボックスをクリックしてパスを直接入力する。
  - テキストボックスをダブルクリックするかBrowseボタン [...] をクリックしてブラウザをプロンプトして、保存場所まで移動する。



#### 注記

ヘッダファイルの保存場所は、フルパスまたは相対パスを使って指定します。相対パスを使う場合、パスの起点としてプロジェクトフォルダを使います。

5. SoundBankのコンテンツファイルを生成する場合は、**SoundBank content file format**オプションで、生成するテキストファイルのフォーマットを選択する。



#### Tip

ファイルパス、オブジェクト名、オブジェクトノーツに非ANSI文字が含まれる場合は、Unicodeフォーマットを使用するべきです。

6. OKをクリックして、設定を適用する。

## 関連トピック

- 「SoundBankの保存場所の指定」
- 「SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定」
- 「プロジェクトのGeneral Settingsの設定」
- 「ログに表示されるメッセージの管理」
- 「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- 「外部ソースのインプット・アウトプット場所を指定」
- 「プロジェクトのデフォルトユーザー設定」

### SoundBankの保存場所の指定

生成されたプロジェクトのSoundBankは、デフォルトで以下のフォルダに保存されます：

ProjectName\GeneratedSoundBanks\Platform\

この場所が適切でない場合は、ワークステーションやネットワーク上の任意のディレクトリに変更できます。

SoundBankの保存場所は、フルパスまたは相対パスを使って指定します。相対パスを使う場合、パスの起点としてプロジェクトフォルダを使います。例えば、以下のフルパスと相対パスは、同じ場所を指定しています。

- C:\Wwise Projects\My Project\GeneratedSoundBanks\Windows
- GeneratedSoundBanks\Windows\

## 保存するSoundBankの、新しい保存場所を指定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. SoundBanksタブに切り替える。
3. グループボックス**SoundBank Paths**で、以下のいずれかを行い、パスを指定する。
  - テキストボックスに、パスを直接入力する。
  - Browse (...) をクリックして、ブラウザで保存場所まで移動する。
4. OKをクリックして、変更内容を適用する。

## 関連トピック

- 「[SoundBankのプロジェクト設定](#)」
- 「[SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定](#)」
- 「[プロジェクトのGeneral Settingsの設定](#)」
- 「[ログに表示されるメッセージの管理](#)」
- 「[プロジェクトのオブストラクションやオクルージョンの、カーブ設定](#)」
- 「[外部ソースのインプット・アウトプット場所を指定](#)」
- 「[プロジェクトのデフォルトユーザー設定](#)」

## SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定

ワークフローによっては、SoundBank生成の直前、または直後に実行すべき処理やタスクがあるかもしれません。例えば、特定のSoundBank生成の前に、そのSoundBankファイルをソースコントロールシステムからチェックアウトする処理や、SoundBank生成直後に、ストリーム用ファイルをSoundBankのディレクトリにコピーする処理などです。

Wwiseでこのような種類のタスクを、SoundBankのプリジェネレーション (Pre-generation / 生成前) タスク、ポストジェネレーション (Post-generation / 生成

後) タスクと呼び、コマンドラインを作成して定義します。Wwiseには、必要なコマンドラインを必要な数だけ簡単にビルトするための、専用のコマンドラインエディタがあります。処理をさらにシンプルにするために、コマンドラインで使える全てのWwise固有の変数やWindows環境変数のリストが、エディタに入っています。

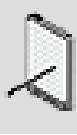
カスタマイズしたコマンドラインを書くために使えるWwise変数を、下表に具体的に示します。

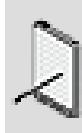
Command Line Variable	内容
<code>\$(AllowExceedMaximum)</code>	SoundBankが、設定された最大サイズを超過した場合、生成するか否かを示す。 <b>Allow SoundBanks to exceed maximum size</b> オプションを選択した場合、この変数は「true」に設定される。
<code>\$(ContentFileFormat)</code>	生成されるSoundBankのコンテンツファイルの、ファイルタイプを示す。以下が、可能な値。 <ul style="list-style-type: none"><li>• ANSI</li><li>• Unicode</li></ul>
<code>\$(GenerateContentFile)</code>	各SoundBankのコンテンツリストのファイルを作成するか否かを示す。コンテンツファイルには、イベント、バス、State、Switchの情報や、ストリーム用やメモリ用の全オーディオファイルのリストが含まれる。 <b>Generate SoundBank content files</b> オプションを選択した場合、この変数は「true」に設定される。
<code>\$(GenerateHeaderFile)</code>	イベント、State、Switch、Game Parameterの名前をIDにマッピングしたヘッダファイルを、生成するか否かを示す。 <b>Generate header file</b> オプションを選択した場合、この変数は「true」に設定される。
<code>\$(GenerateMaxAttenuationInfo)</code>	イベントの最大減衰の情報を生成するか否かを示す。 <b>Metadata Options: Max attenuation</b> オプションが選択されている時、この変数は「True」に設定される。
<code>\$(GenerateEstimatedDuration)</code>	イベントに推定最長ならびに最短持続時間、および持続時間タイプ情報が生成するか否かを示す。 <b>Metadata Options: Estimated Duration</b> オプションが選択されている時、この変数は「True」に設定される。
<code>\$(HeaderFileFullPath)</code>	ヘッダファイルのフルパスで、具体的には: <code>\$(HeaderFilePath)\Wwise_IDs.h</code>
<code>\$(HeaderFilePath)</code>	ヘッダファイルを保存する、パスまたは場所。 このパスは、テキストボックスHeader file pathから取得される。
<code>\$(InfoFilePath)</code>	現在のプラットフォームの、Infoファイルの完全なファイル名。
<code>\$(IsRunningFromCmdLine)</code>	Wwiseが、-generatesoundbanksフラグ付きのコマンドラインから起動されたか否かを示す。
<code>\$(LanguageList)</code>	コマンドラインにパスされた言語のリスト、またはSoundBank Managerで選択された言語のリスト。
<code>\$(Platform)</code>	現在のプラットフォーム名。



#### 注記

このリストは、半角スペース区切りのリストです。

Command Line Variable	内容
<code>\$(SoundBankList)</code>	コマンドラインにパスされたSoundBankのリスト、またはSoundBank Managerで選択されたSoundBankのリスト。
	 <b>注記</b> このリストは、半角スペース区切りのリストです。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。
<code>\$(SoundBankPath)</code>	現在のプラットフォームのSoundBankが保存されている、パスまたは場所。
<code>\$(UseSoundBankNames)</code>	生成されたSoundBankファイル (.bnk) の名前として、SoundBank名 (true) とID (false) のどちらを使用するかを指定し、これは SoundBank内から他のSoundBankのメディアをレファレンスする時にも、適用される。  Use SoundBank namesオプションを選択した場合、この変数は「true」に設定される。
<code>\$(WwiseExeDriveLetter)</code>	Wwise実行ファイル (Wwise.exe) がある、ワークステーションのドライブレター。
<code>\$(WwiseExePath)</code>	Wwise実行ファイル (Wwise.exe) のパスまたは場所。
<code>\$(WwiseExeProcessID)</code>	Wwise実行ファイル (Wwise.exe) の処理ID (Process ID) の数値。
<code>\$(WwiseProjectDriveLetter)</code>	Wwiseプロジェクトがある、ワークステーションのドライブレター。
<code>\$(WwiseProjectName)</code>	現在のプロジェクト名。
<code>\$(WwiseProjectPath)</code>	Wwiseプロジェクトのパスまたは場所。
<code>\$(WaapiWampPort)</code>	WA APIがWAMPプロトコル用に使うポート。
<code>\$(WaapiHttpPort)</code>	WA APIがHTTPプロトコル用に使うポート。

**注記**

さらに`$(WWISESDK)`などの環境変数を全て利用できます。

柔軟性を高めるために、Wwiseでは以下の種類のステップに対して、異なるコマンドラインを設定できます。

- **Global opening step** - 全プラットフォームにグローバル（全体的）に適用し、他のどのステップよりも前に実行するコマンドライン。
- **Platform-specific pre-generation step** - 特定プラットフォームに適用し、SoundBank生成前に実行するコマンドライン。
- **Platform-specific post-generation step** - 特定プラットフォームに適用し、SoundBank生成後に実行するコマンドライン。
- **Global closing step** - 全プラットフォームにグローバル（全体的）に適用し、全てのステップの後に実行するコマンドライン。

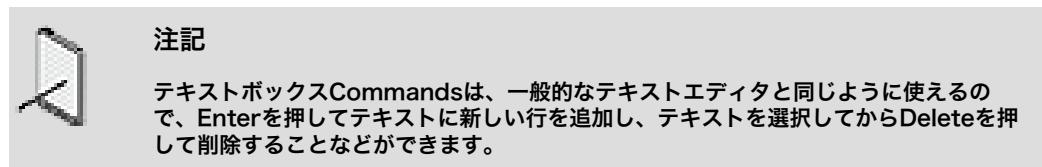
ポストジェネレーション（生成後）のデフォルトのステップとして、プラットフォームごとに、ストリーミングファイルをSoundBankのディレクトリにコピーするステップが、プロジェクトに必ず入っています。なお、他にもコマンドラインを設定して、どのような種類のタスクでも自動化できます。またWwiseには、SoundBankの中の全てのデータやメディアが入ったパッケージをFile Packager（ファイルパッケージャ）機能で生成する、別のファクトリコマンドラ

インも同梱されています。File Packagerについては[36章ファイルパッケージの管理](#)を参照してください。 ファクトリコマンドラインのロードについては「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」を参照してください。

作成したコマンドラインをファイル (.wcmdline) に保存して、後で同じプロジェクトや別のプロジェクトで使用したり、他のユーザーとシェアしたりすることもできます。コマンド保存については「[カスタムコマンドラインを、ファイルに保存](#)」を参照してください。

## SoundBank生成前に行うタスクを設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. SoundBanksタブに切り替える。
3. グローバル（プロジェクト共通）Pre-Generation Stepを作成するにはBrowseボタン [...]をクリックしてGlobal Opening Step Editorを開く。  
Pre-Generation Step Editorが開く。
4. テキストボックスDescriptionに、実行するステップやタスクの内容がよく分かる名前を入力する。
5. テキストボックスCommandsをクリックして、コマンドラインの作成を開始する。



6. コマンドに、組み込みマクロや環境変数をインサートしたい場合は、以下を行う:  
グループボックスMacrosで、以下のオプションを1つ選択する。
  - Built-in Macros - 組み込みマクロのことであり、Wwiseコマンドライン内で使える、Wwise固有の変数リストを表示。
  - Environment Variables - 環境変数のことであり、Wwiseコマンドライン内で使える、Windows固有の環境変数リストを表示。  
コマンドラインに変数を追加するには、以下のいずれかを行う:
  - リストの変数をダブルクリックする。
  - リストの変数を選択して、Insertをクリックする。

必要に応じて、コマンドラインに変数を追加する。

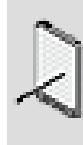
7. プリジェネレーションのグローバルステップを追加する場合は、最初の行の終わりで、Enterを押し、新しいコマンドラインを入力する。
8. OKをクリックして、コマンドラインを保存して、Pre-Generation Step Editorを閉じる。



#### 注記

コマンドラインをファイルに保存する場合は、Editorの、Save As ボタンをクリックします。カスタムコマンドラインの保存については「[カスタムコマンドラインを、ファイルに保存](#)」を参照してください。

9. プリジェネレーションのステップを、プラットフォームごとに作成するには、プラットフォームごとに上記3~8を繰り返す。

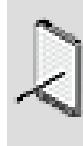


#### 注記

ファクトリコマンドや保存したカスタムコマンドラインを、Editorにロードするには、Loadボタンをクリックします。ファクトリコマンドやカスタムコマンドのロードについては「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」を参照してください。

### SoundBank生成後に行うタスクを設定するには:

1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - Projectメニューで、Project Settingsを選択する。
  - [Shift]+[K]を押す。
2. SoundBanksタブに切り替える。
3. グループボックスPost-Generation Stepにおいて、デフォルトでコマンドラインCopy Streamed Filesが表示される。このコマンドラインを変更したり追加コマンドを追加するには、Browseボタン [...] のいずれかをクリックする。  
Post-Generation Step Editorが開く。
4. テキストボックスDescriptionに、実行するステップやタスクの内容がよく分かる名前を入力する。
5. テキストボックスCommandsの、現在のコマンドライン末尾をクリックして、Enterを押す。新規コマンドラインの作成を開始する。



#### 注記

テキストボックスCommandsは、一般的なテキストエディタと同じように使えるので、Enterを押してテキストに新しい行を追加し、テキストを選択してからDeleteを押して削除することができます。

6. コマンドに、組み込みマクロや環境変数をインサートしたい場合は、以下を行う:

グループボックスMacrosで、以下のオプションを1つ選択する。

- Built-in Macros - 組み込みマクロのことであり、Wwiseコマンドライン内で使える、Wwise固有の変数リストを表示。
- Environment Variables - 環境変数のことであり、Wwiseコマンドライン内で使える、Windows固有の環境変数リストを表示。

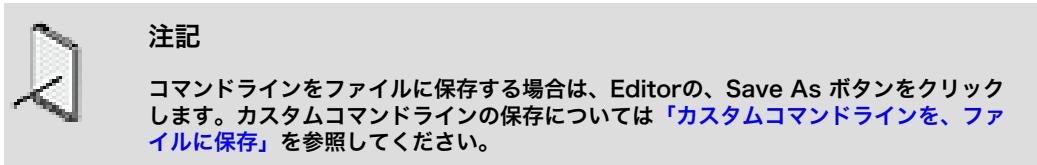
コマンドラインに変数を追加するには、以下のいずれかを行う:

- リストの変数をダブルクリックする。

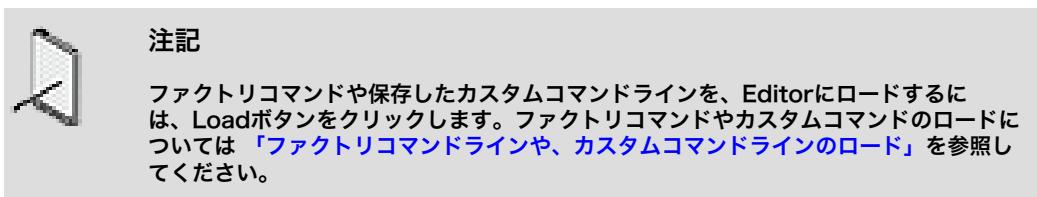
- リストの変数を選択して、Insertをクリックする。

必要に応じて、コマンドラインに変数を追加する。

- プリジェネレーションのグローバルステップを追加する場合は、最初の行の終わりで、<e0>Enter</e0>を押し、新しいコマンドラインを入力する。
- OKをクリックして、コマンドラインを保存して、Post-Generation Step Editorを閉じる。



- グローバルなクロージングステップや、プラットフォームごとのステップを追加するには、上記3~8を繰り返す。



## 関連トピック

- 「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」
- 「[カスタムコマンドラインを、ファイルに保存](#)」
- 「[SoundBankのプロジェクト設定](#)」
- 「[SoundBankの保存場所の指定](#)」
- 「[プロジェクトのGeneral Settingsの設定](#)」
- 「[ログに表示されるメッセージの管理](#)」
- 「[プロジェクトのオブストラクションやオクルージョンの、カーブ設定](#)」
- 「[外部ソースのインプット・アウトプット場所を指定](#)」
- 「[プロジェクトのデフォルトユーザー設定](#)」

### ファクトリコマンドラインや、カスタムコマンドラインのロード

Wwiseには数個の既成コマンドラインが同梱され、ストリーミングしたファイルを SoundBankのディレクトリにコピーするものや、複数のストリーミングファイルやSoundBankを、1つのパッケージにまとめるものなどがあります。これらをファクトリコマンドラインと言います。ファクトリコマンドラインや、ファイルに保存したカスタムコマンドラインを、ロードできます。

### ファクトリコマンドラインや、カスタムコマンドラインをロードするには:

- Pre/Post Generation Step Editorで、Load...をクリックする。

2. コンテキストメニューで、以下のいずれかをクリックする。
  - **From Factory Folder** - Explorer/Finderが開き、Wwiseファクトリコマンドラインのあるディレクトリが表示される。
  - **From Last Location** - Explorer/Finderが開き、コマンドラインを最後にロードしたディレクトリが表示される。

Openダイアログボックスが開く。

3. ロードするコマンドラインを選択して、Openをクリックする。

コマンドラインが、Editorにロードされる。

## 関連トピック

- 「カスタムコマンドラインを、ファイルに保存」
- 「SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定」

### カスタムコマンドラインを、ファイルに保存

作成したカスタムコマンドラインをファイルに保存して、後で同じプロジェクトや別のプロジェクトで使用したり、他のユーザーとシェアすることもできます。

#### コマンドラインをファイルに保存するには:

1. Pre-Post Generation Step Editorで、コマンドラインを書く。
2. 完成した後に、Save Asをクリックする。

Save Asダイアログボックスが開く。

3. コマンドラインを保存したいフォルダまで移動して、名前を付け、Saveを押す。

コマンドラインは、.wcmdlineファイルとして保存され、いつでも再利用することができます。

## 関連トピック

- 「ファクトリコマンドラインや、カスタムコマンドラインのロード」
- 「SoundBankのプリジェネレーション、ポストジェネレーションのステップの設定」

### ログに表示されるメッセージの管理

SoundBankやコンバージョンの、生成中のログに表示される警告やメッセージを、以後は表示させたくない、または表示させる必要がない場合も考えられます。不要な警告やメッセージを、SoundBank Logに表示させないためには、Log Ignore（ログで無視）リストに追加します。また、Log Ignoreリストからそのメッセージタイプを削除すると、再びログに表示されます。Log Ignoreリストは、Project Settingsダイアログボックスにおいて、プロジェクトレベルで管理されます。



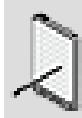
#### 注記

ログ中のメッセージを直接右クリックして、Add Message(s) to Ignore Listを選択して、リストにメッセージを追加することもできます。詳しくは「[Log Ignoreリストに、メッセージを追加する](#)」を参照。

### Log Ignoreリストにメッセージを追加するには:

1. メニューバーで、Project > Project Settingsをクリックする。  
Project Settingsダイアログボックスが開く。
2. Logsタブに切り替える。
3. Log Ignore Listに追加するメッセージを選択する。
4. OKをクリックして、Project Settingsダイアログボックスを閉じる。

次回ログを生成する時に選択されたメッセージタイプがWwiseに無視され、ログに表示されない。



#### 注記

Log Ignoreリストからメッセージタイプを除くには、そのメッセージタイプのIgnoreチェックマークを、外すだけです。

### メッセージの重大度 (Severity) を変更するには:

1. メニューバーで、Project > Project Settingsをクリックする。  
Project Settingsダイアログボックスが開く。
2. Logsタブに切り替える。
3. Severity列のドロップダウン機能を使い、重大度を変更する。
4. OKをクリックして、Project Settingsダイアログボックスを閉じる。

次回ログを生成する時に、選択した重大度が適用されます。

### ログに表示されるメッセージ数を制限するには:

1. メニューバーで、Project > Project Settingsをクリックする。  
Project Settingsダイアログボックスが開く。
2. Logsタブに切り替える。
3. Limit number of messages displayed to:オプションを選択して、ログに表示させる最大メッセージ数を入力する。なお、この制限数は、ログの全てのメッセージタイプに適用される。
4. OKをクリックして、Project Settingsダイアログボックスを閉じる。

### 関連トピック

- 「[プロジェクトのGeneral Settingsの設定](#)」

- ・「プロジェクトのSoundBankの設定」
- ・「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」
- ・「外部ソースのインプット・アウトプット場所を指定」

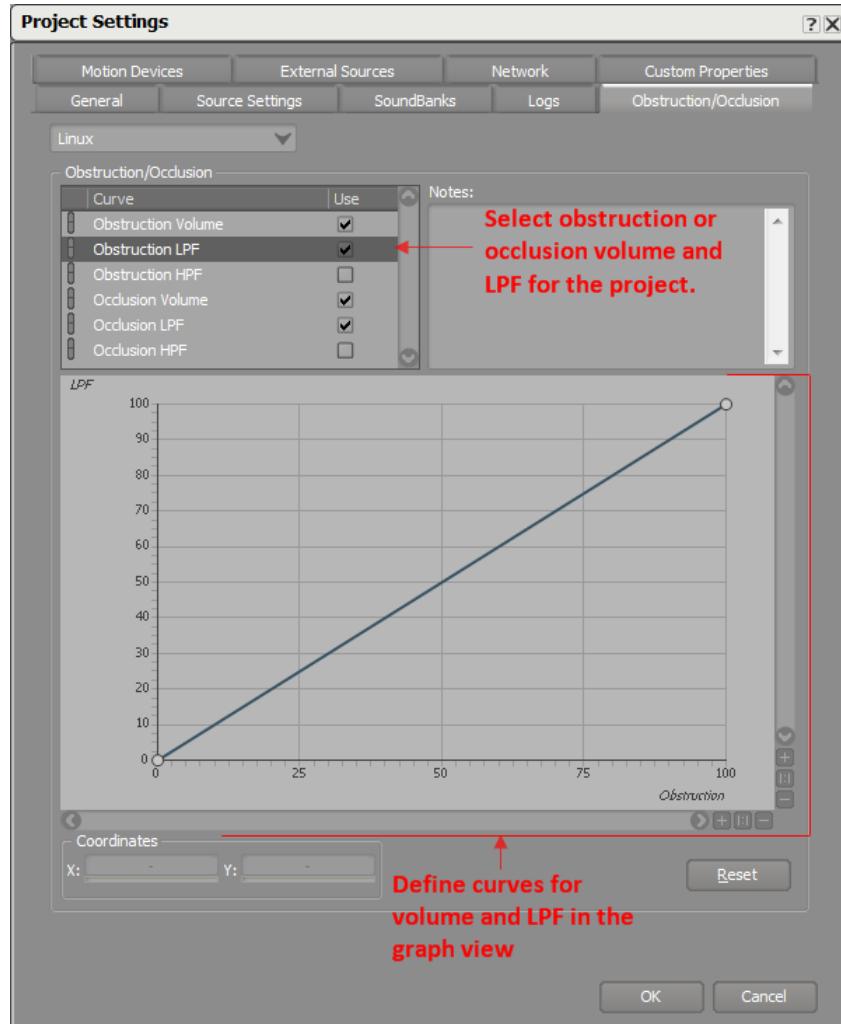
### プロジェクトのオブストラクションやオクルージョンの、カーブ設定

WwiseのProject SettingsダイアログボックスのObstruction/Occlusionタブで、サウンドオブジェクトのオブストラクション設定とオクルージョン設定を、プロジェクトが対象とするプラットフォームごとに行います。オブストラクションは、ゲーム中にジオメトリにある壁や柱などの物体が、サウンドソースとリスナーの間の空間を部分的に塞ぐ時に起きる状況です。。オクルージョンは、ゲームのジオメトリにある物体が、サウンドソースとリスナーの間の空間を完全に塞ぐ時に起きる状況です。

ゲーム開発者は、ゲーム内でオブストラクションやオクルージョンの条件がおきる可能性のあるジオメトリをプログラミングします。Wwiseはオブストラクションやオクルージョンのレベルを計算しません。物理的な計算をゲーム側で行い、その結果を以下の関数に送ります: [SetObjectObstructionAndOcclusion\(\)](#) .

開発者がオブストラクションやオクルージョンをどうプログラミングするかについては、Wwise SDKドキュメンテーションの「[Obstruction and Occlusion in Environments](#)」を参照してください。

Wwiseでは、ゲーム中にサウンドオブジェクトがオブストラクションやオクルージョンの影響を受けた時に、適用するボリュームカーブやLPFカーブをプラットフォームごとに設定できます。設定値は相対値であり、ゲームオブジェクトに既に設定されたボリューム値やLPF値に対して、付加されます。



オブストラクションやオクルージョンのカーブをさらに細かく複雑に設定するには、カーブセグメントの形状を調整します。カーブセグメントとは、カーブ上の任意の2点のコントロールポイントの間の部分を指します。カーブ形状として、直線、定数、対数関数、指数関数、S型カーブなど、様々な選択肢があります。

オブストラクションやオクルージョンの詳細情報については「[学習コーナー: オブストラクションとオクルージョンについて](#)」を参照してください。

**プロジェクトのオブストラクションやオクルージョンを設定するには:**

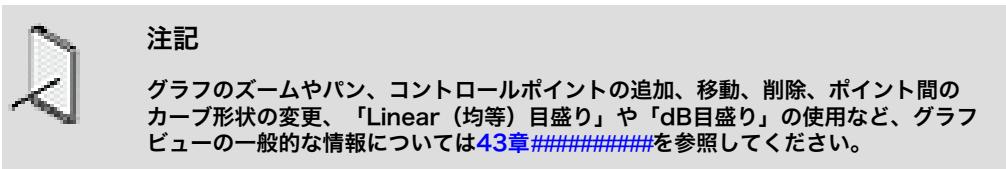
1. 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - メニューバーで、Project > Project Settingsをクリックする。
  - [Shift]+[K]を押す。
2. Obstruction/Occlusionタブに切り替える。
3. プラットフォームリストから、これからオブストラクションやオクルージョンのカーブを設定するプラットフォームを選択する。

4. 選択したプラットフォームのオブストラクションやオクルージョンに、別のボリュームやLPFを設定するには、リンクインジケータを右クリックして、ショートカットメニューから、Unlinkを選択する。



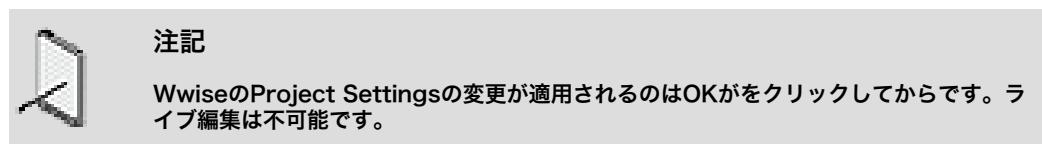
リンクインジケータがオレンジ色に変わり、このアンリンク（リンクを解除）されたプロパティ設定が、その選択したプラットフォームに限り、使用される。

5. ゲームオブジェクトのオブストラクションやオクルージョンの、ボリュームやLPFのカーブを設定するには、以下の操作を行う。
  - ・チェックボックスUseを選択して、対象カーブをハイライト表示させる。
  - ・カーブ上に新規ポイントを作成するには、カーブ上でダブルクリックする。
  - ・カーブ上のポイントを削除するには、ポイントを選択して、Deleteを押す。
  - ・設定したポイントを全て削除して、最初からやり直すには、Resetをクリックする。



6. カーブにコメントを追加するには、Notesボックスをクリックして、コメントを入力する。
7. 完了後に、OKをクリックする。

Project Settingsダイアログボックスが閉じ、設定したオブストラクションやオクルージョンのプロパティカーブが保存される。



## 関連トピック

- ・「グラフビューのズームとパン」
- ・「グラフビューの単位の設定」
- ・「コントロールポイント間の、カーブ形状の指定」
- ・「新規プロジェクトの作成」
- ・「プロジェクトのトラブルシューティング」
- ・「プロジェクトのデフォルトユーザー設定」
- ・「プロジェクトのGeneral Settingsの設定」
- ・「プロジェクトのSoundBankの設定」
- ・「ログに表示されるメッセージの管理」

- ・「外部ソースのインプット・アウトプット場所を指定」

### 外部ソースのインプット・アウトプット場所を指定

プラグインExternal Sourceを使用する場合、まず、External Sources Listファイルの場所を指定する必要があります。このファイルは、以下の情報を指定する、非常にシンプルなXMLファイルです。

- ・Wwiseで作成したテンプレートに関連付けが可能な、外部オーディオファイルの場所
- ・各ファイルのコンバージョンに使用する、コンバージョン設定

また、コンバージョンしたソースを保存するフォルダを指定して、ランタイムにそれらをWwiseサウンドエンジンが利用できるようにします。

### 外部オーディオソースのインプットパス・アウトプットパスを指定するには:

1. メニューバーで、Project > Project Settingsをクリックする。

Project Settingsダイアログボックスが開く。

2. External Sourcesタブに切り替える。
3. グループボックス**Input Path**で、External Sources Listファイルが入ったフォルダの場所を指定するために、以下のいずれかを行う。
  - ・External Sources Listをクリックして、External Sources Listファイルの場所を示すパスを入力する。
  - ・いずれかのゲームプラットフォームのBrowseボタン [...]をクリックして、そのプラットフォームのExternal Sources Listファイルが含まれるフォルダまで移動してOpenをクリックする。
4. 上記3を、プロジェクトで有効な全てのプラットフォームで繰り返す。
5. グループボックス**Output Path**で、コンバージョンした外部ソースを保存するフォルダを指定するために、以下のいずれかを行う。
  - ・External Sources Output Folderリストをクリックして、オーディオファイルを保存する場所のパスを入力する。
  - ・いずれかのゲームプラットフォームのBrowseボタン [...]をクリックして、そのプラットフォームのコンバージョン後のオーディオファイルを保存するフォルダまで移動してOKをクリックする。
6. 上記5を、全てのプラットフォームで繰り返す。
7. OKをクリックして、Project Settingsダイアログボックスを閉じる。

### 関連トピック

- ・「プロジェクトのGeneral Settingsの設定」
- ・「プロジェクトのSoundBankの設定」
- ・「ログに表示されるメッセージの管理」
- ・「プロジェクトのオブストラクションやオクルージョンの、カーブ設定」

- ・「プロジェクトのデフォルトユーザー設定」

## ネットワークポートの指定

オーサリングアプリケーションWwiseをゲームに接続して、プロファイリング機能やリアルタイム編集機能を使う時、Wwiseとゲームの間のコミュニケーションは、ゲームとオーサリングアプリケーションのために開いたネットワークポートを使用するネットワークで行われます。オーサリングアプリケーション用に開いて使用するポートは、WwiseのProject Settingsダイアログボックスの、Networkタブからカスタム設定できます。ゲーム用に開いて使用するポートは、ゲームがWwiseコミュニケーションモジュールを初期化する時にカスタム設定できます。

ポート番号は予め設定する（固定）か、OSに自動的に選定させます（ダイナミック）。どちらにするかは、ポートを開くのがオーサリングアプリケーションなのか、ゲームなのかによって決まります。



### 注意

ポートをProject Settingsで変更する場合、必ずWwiseコミュニケーションモジュールを初期化する、ゲーム側のコードの中の値と同じ値に、正確に変更して下さい（具体的には、`AK::Comm::Init()` にパスされる構造のports.uDiscoveryBroadcast部分）。

## Game Discovery Broadcast ポート

このポートは、ゲーム内のコミュニケーションモジュールによって開かれ、オーサリングアプリケーション側が接続可能なゲームを探す時にネットワーク上でブロードキャストするメッセージを聞き取るために使われます。オーサリングアプリケーションがこのポートに対してGame Discoveryメッセージをブロードキャストするので、ゲームとWwiseプロジェクトで同じポートを使用することが重要です。ゲーム側とオーサリングアプリケーション側の両方でこのポートを認識する必要があるため、ダイナミック（0設定）にできません。



### 注意

ポート番号は1から65535まで自由に選べますが、ゲームと同じコンピュータやゲーム機で稼働する他のアプリケーションと、コンフリクトを起こさないことを、確認して下さい。

## Game Discovery Response ポート

このポートは、オーサリングアプリケーションWwiseによって開かれます。ここからゲームがGame Discoveryメッセージに反応します。このポートの番号を設定する時に、以下の重要事項を考慮して下さい。

- ・ポートを0に設定することで、オーサリングアプリケーションはダイナミックポート番号（別名エフェメラルポート番号）をリクエストします。つまり、特定のポート番号を使用せずに、OSが自動的にポート番号を1つ選びます。デフォルトで、このダイナミックポート番号が設定され、他のアプリケーションとのコンフリクトを回避するために、ダイナミックポート番号が推奨されます。
- ・固定ポート番号（ダイナミックまたはエフェメラルポート番号以外）を使用する必要がある場合は、オーサリングアプリケーションと同じコンピュータで稼働するゲームに接続する予定がない場合を除き、Game Discovery Broadcast Portと同じポート番号を使わないでください。

- ポート番号は0から65,535まで自由に選べますが、ゲームと同じコンピュータやゲーム機で稼働する他のアプリケーションと、コンフリクトを起こさないことを、確認して下さい。

### ゲームのコミュニケーションポートのカスタム設定

前述の通り、Game Discovery Broadcast Portはコミュニケーションモジュールによってゲーム内で開かれます。コミュニケーションモジュールによってゲーム内で開かれるポートは他に3つあり、初期化中にカスタマイズされます。これらのポートはオーサリングアプリケーションで露出されないので、詳細については[Wwise SDK ドキュメンテーション](#)の「Initializing Communications」を参照してください。

コミュニケーションポートをカスタム設定するには:

- 以下のいずれかを行い、Project Settingsダイアログボックスを開く。
  - メニューバーで、Project > Project Settingsをクリックする。
  - [Shift]+[K]を押す。
- Networkタブに切り替える。
- 必要に応じてポート番号を変更する。
- OKをクリックする。

Project Settingsダイアログボックスが閉じる。これで、新規ポート番号がゲームとのコミュニケーションに使われる。

### 関連トピック

- 「ローカルまたはリモートで、ゲーム機に接続する」
- 「リモート接続の問題のトラブルシューティング」

## プロジェクトのデフォルトユーザー設定

Wwiseでは、Default User Settingsダイアログボックスで、プロジェクトの様々なデフォルトユーザー設定を設定できます。デフォルトで設定した値は、プロジェクトのプロパティ設定のスタート地点として適しています。プロジェクト開発の早い段階でデフォルト設定を決めれば、プロジェクトで作成した全オブジェクトに適用できるので、効率的です。

プロジェクト開発の最中にオブジェクトのデフォルト設定を変更すると、変更後に作成されたオブジェクトだけに変更が適用されます。1つのオブジェクトタイプ、例えばSFXオブジェクトのデフォルトプロパティを設定して、その後に別のオブジェクトタイプ、例えばボイスオブジェクトの設定に変更できるので、ワークフローの効率化に役立ちます。

これらのデフォルト設定は、現在のプロジェクトディレクトリにある.wsettingsファイルに保存されます。デフォルト設定はユーザー別に設定できるので、プロジェクト内の違う部分で作業をしているユーザー達が、それぞれ異なるデフォルト設定を設定できます。

## プロジェクト内のオブジェクトのデフォルトユーザー設定を設定するには:

1. 以下のいずれかを行い、Default User Settingsダイアログボックスを開く。
  - プロジェクトメニューで、Default Settingsを選択する。
  - [Shift]+[D]を押す。
2. サウンドオブジェクトやミュージックオブジェクトのデフォルトルーティング設定は、以下の操作を行う。

RoutingグループボックスでAudio OutputのBrowseボタン(...)をクリックする。

Project Explorer - Browserが開く。

プロジェクトのサウンドオブジェクトやミュージックオブジェクトのデフォルトバスとするAudio Busを選択する。

OKをクリックする。

新規に作成するサウンドオブジェクトやミュージックオブジェクトが、全て自動的にこのAudio Busにルーティングされます。

3. 全ての新規オブジェクトのデフォルトボリュームを設定するには、テキストフィールドVoice Volumeで値を指定する。
4. プロジェクトのデフォルトのコンバージョン設定ShareSetをオーバーライドするには、以下の操作を行う：

グループボックスDefault Conversion Settingsで、チェックボックスOverride Project Settingsを選択する。

Browseボタン(...)をクリックする。

Project Explorer - Browserが開く。

Conversion Settings階層の中を移動して、プロジェクトのデフォルトコンバージョン設定として使用するShareSetを選択する。

OKをクリックする。

新規に作成する最上位の親オブジェクトが、全て自動的にこのコンバージョン設定ShareSetを使用する。

5. 完了後に、OKをクリックする。

Default Settingsダイアログボックスが閉じ、設定したデフォルトユーザー設定が保存される。

## 関連トピック

- 「[Project Settingsの設定](#)」
- 「[ユーザープリファレンスの設定](#)」

## プロジェクトのトラブルシューティング

インテグレティレポートを生成して、プラットフォーム、オーディオファイル、SoundBank、プラグインなどの問題に関する情報を見て、プロジェクトのステータスを確認できます。このレポートを検証することで、プロジェクトの以下のような問題に対処できます。

- ・欠如しているメディアファイル
- ・特定のランゲージバージョンなど、欠如しているソース
- ・プラグイン問題
- ・SoundBankに欠如しているオブジェクト
- ・SoundBankのレンダリング問題やエフェクトバイパス問題
- ・SoundBankのレンダリングやRPC

インテグレティレポートはプロジェクト問題を提示するほか、問題に関する以下のような情報も提供します。

- ・Platform - 問題が発生したプラットフォーム
- ・Type - 影響を受けるオブジェクトの種類
- ・Object name - 影響を受けるオブジェクトやエレメントの名前
- ・Status - 問題の内容
- ・Comments/Suggestions - 問題への対処に関する情報や提案
- ・Hierarchy - 影響を受けるオブジェクトの階層内の場所

### インテグレティレポートの生成

Wwiseプロジェクトの開発中にいつでも、インテグレティレポートを生成できます。SoundBankの生成前や、ソースコントロールを使用している場合は、大幅に変更したWork Unitをチェックインする前などに、インテグレティレポートを生成すると良いでしょう。

特定の問題を探す場合、生成する情報の種類を指定してレポートにフィルタをかけると、見やすくなります。レポート内容は、以下から選ぶことも全てを表示させることもできます。

- ・Audio files and sources (オーディオファイル、オーディオソース)
- ・Hierarchies (階層)
- ・References (レファレンス)
- ・Optimizations (最適化)

レポートを生成してから、記載された問題を確認した上で、問題をダブルクリックして提案されたフィックスを実行します。例えば、“No group assigned to the Switch container (Switch Containerにグループがアサインされていない)”というステータスマッセージをダブルクリックすると、Property Editorが開くので、Switch ContainerにSwitch GroupまたはStateやState Groupをアサインできます。

### インテグレティレポートを生成するには:

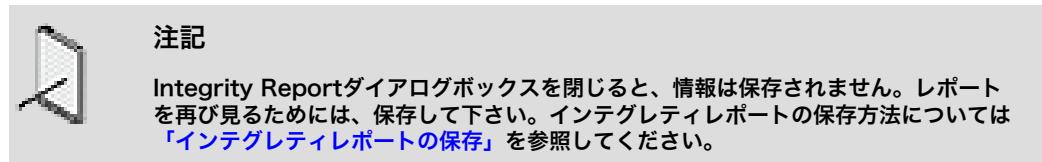
1. Wwiseメニューバーで、Views > Integrity Reportをクリックする。

- Integrity Reportが開く。
2. インテグレティレポートに含めるプラットフォーム問題を決めるために、[Platform Manager](#) で定義したプラットフォームを、グループボックス[Platforms](#) で1つ以上選択する。
  3. インテグレティレポートに含めるランゲージを決めるために、グループボックス[Languages](#)で、[Languages](#)オプションを選択して、該当するランゲージを選択する。
  4. フィルタをかけて問題のタイプを限定するには、以下から1つ以上、選択する:
    - Audio files and sources (オーディオファイル、オーディオソース)
    - Hierarchies (階層)
    - Broken References
    - Optimizations (最適化)
5. **Generate**をクリックする。

生成プロセスのステータスが、プログレスバーで表示される。

6. Wwiseインテグレティレポートの生成が完了したら、**Close**をクリックする。

Integrity Report- Completedダイアログボックスが閉じ、インテグレティレポートが表示される。



7. 問題をダブルクリックすると対応するダイアログボックスが開くので、問題のステータスをフィックスまたは編集する。

#### インテグレティレポートで発見できる問題

インテグレティレポートで検知できるプロジェクト問題とその対処方法を、下表に示します。

Issue	Information/Suggestion
Original audio file not found. オリジナルのオーディオファイルがない。	以下のいずれかを行う: <ul style="list-style-type: none"> <li>・不足しているファイルを、Originalsフォルダにドラッグする。</li> <li>・Audio File Importerを使って、不足しているファイルをインポートする。</li> </ul>
Audio source is missing. オーディオソースがない。	メッセージをダブルクリックして、SFXの場合はProject ExplorerのAudioタブで、ボイスの場合はContents Editorで、不足しているソースを確認する。
Source Plug-in not supported on platform. ソースプラグインにプラットフォームが未対応。	メッセージをダブルクリックして、Contents Editorを開き、現在のプラットフォームが対応するソースプラグインを選択する。
Source Plug-in not installed.	ソースプラグインをインストールする。

Issue	Information/Suggestion
ソースプラグインがインストールされていない。	
Missing language sources. ランゲージソースがない。	Audio File Importerを使って、不足しているランゲージソースをインポートする。
Selected codec does not support this source's channel configuration. 選択したコーデックが、本ソースのチャンネルコンフィギュレーションに未対応。	コンバージョン前にWwiseが自動的にソースをダウンミックスする。コーデック（オーディオフォーマット）やチャンネル設定を変更するには、メッセージをダブルクリックする。
Audio FX plug-in not supported on platform. オーディオFXプラグインにプラットフォームが未対応。	メッセージをダブルクリックして、Property Editorを開き、現在のプラットフォームが対応するエフェクトプラグインを選択する。
Audio FX Plug-in not installed. オーディオFXプラグインがインストールされていない。	不足しているオーディオエフェクトのプラグインをインストールする。
Render effect has been applied to an "In use" source that is not an audio file. レンダーエフェクトがオーディオファイルではない"In Use"ソースに適用された。	ソースプラグインにレンダーエフェクトを適用できない。メッセージをダブルクリックして、以下のいずれかを行う。 Property Editorで、レンダーエフェクトのオプションを外す。 Contents Editorで、オーディオファイルを"In Use"とするオーディオソースを選択する。
The selected effect must be rendered. 選択したエフェクトがレンダリングをする。	メッセージをダブルクリックして、Property Editorを開き、以下のいずれかを行う。 エフェクトのレンダリングを行う。 使用するエフェクトを変更する。
The effect could be rendered to save CPU, since no parameters will change in game. ゲームでパラメータが変更しないので、エフェクトをレンダリングしても、CPU負荷を削減できる。	メッセージをダブルクリックして、Property Editorを開き、チェックボックスRenderを選択する。
The effect on this object has been bypassed. The render effect will be ignored. 本オブジェクトでエフェクトがバイパスされた。レンダーエフェクトは無視される。	両方の操作をサポートすることは不可能なので、プロジェクトにおけるレンダーエフェクトの重要性を検証する。レンダーエフェクトを適用する場合は、バイパスエフェクトを無効にする。
A bypass effect property has been assigned to the RTPC for this object. The render effect will be ignored. 本オブジェクトのRTPCにバイパスエフェクトプロパティが設定された。レンダーエフェクトは無視される。	両方の操作をサポートすることは不可能なので、レンダーエフェクト操作とRTPC主導のバイパスエフェクトの、双方のプロジェクトにおける重要性を検証する。レンダーエフェクトを適用する場合は、そのオブジェクトのRTPCバイパスエフェクトを無効にする。
An RTPC has been applied to the effect for this object. The render effect will be ignored. 本オブジェクトのエフェクトにRTPCが適用された。レンダーエフェクトは無視される。	両方の操作をサポートすることは不可能なので、適用されたRTPCとレンダーエフェクトのどちらがプロジェクトにとってより重要なを検証する。レンダーエフェクトを適用する場合は、RTPCを無効にする。
Streamed XMA files do not support region loops. All user-defined loop regions will be ignored. ストリーミングXMAファイルがリージョンループに対応していない。ユー	以下のいずれかを行う： <ul style="list-style-type: none"><li>メッセージをダブルクリックして、Property Editorを開き、チェックボックスLoopを外す。</li><li>Conversion Settingダイアログボックスで、ファイルのオーディオフォーマットを変える。</li></ul>

## Wwiseのプロジェクトについて

---

Issue	Information/Suggestion
ザー定義によるループリージョンが全て無視される。	
XMA files cannot have a loop setting that exceeds 254. XMAファイルに254を超えるループを設定できない。	以下のいずれかを行う: <ul style="list-style-type: none"> <li>メッセージをダブルクリックしてProperty Editorを開き、テキストボックスNo. of loopsで、ループ数を減らす。</li> <li>Conversion Settingダイアログボックスで、ファイルのオーディオフォーマットを変える。</li> </ul>
In memory XMA files of over 8 Mb are not supported. 8Mbを超えるインメモリXMAファイルに未対応。	以下のいずれかを行う: <ul style="list-style-type: none"> <li>サウンドをストリーミングする。</li> <li>サンプルレートまたは圧縮品質を下げる。</li> </ul>
Seek table required for this virtual voice behavior. 本バーチャルボイス動作はシークテーブルを要する。	メッセージをダブルクリックしてConversion Settingダイアログボックスを開き、以下のいずれかを行う。 <ul style="list-style-type: none"> <li>Vorbis Encoder Parametersダイアログボックスの、シークテーブルを有効にする。</li> <li>ファイルフォーマットを変更する。</li> </ul>
Seek table not required for this virtual voice behavior. 本バーチャルボイス動作はシークテーブルが不要である。	シークテーブルを無効にしてメモリ消費を下げることが可能。メッセージをダブルクリックしてConversion Settingダイアログボックスを開き、Vorbis Encoder Parametersダイアログボックスの、シークテーブルを無効にする。
Seek table required for music objects in Vorbis format. Vorbisフォーマットのミュージックオブジェクトがシークテーブルを要する。	メッセージをダブルクリックしてConversion Settingダイアログボックスを開き、以下のいずれかを行う。 <p>Vorbis Encoder Parametersダイアログボックスの、シークテーブルを有効にする。</p> <p>ファイルフォーマットを変更する。</p>
Child object in this container is not assigned to a switch. 本コンテナの子オブジェクトがSwitchにアサインされていない。	メッセージをダブルクリックして、Project Explorerと、関連するProperty Editorを開き、以下のいずれかを行う。 <p>アサインされていないオブジェクトを削除する。</p> <p>オブジェクトをSwitchにアサインする。</p>
No group assigned to the Switch container. Switch Containerにグループがアサインされていない。	ダブルクリックして、Property Editorを開き、Switch ContainerにSwitch GroupまたはStateやState Groupをアサインする。
No objects assigned to a switch in Switch container. Switch Container内のSwitchに、オブジェクトがアサインされていない。	ダブルクリックして、Contents Editorを開き、Switch ContainerのSwitchにオブジェクトをアサインする。
No group is assigned to the Music Switch container. Music Switch Containerに、グループがアサインされていない。	ダブルクリックして、Property Editorを開き、Music Switch ContainerにSwitch GroupまたはStateやState Groupをアサインする。
No music objects assigned to a switch in this container. 本コンテナのSwitchに、ミュージックオブジェクトがアサインされていない。	ダブルクリックして、Music Switch Association Editorを開き、Switchにミュージックオブジェクトをアサインする。
The playlist of the Sequence container has no assigned objects. Sequence Containerのプレイリストに、オブジェクトがアサインされていない。	ダブルクリックして、Contents Editorを開き、プレイリストにオブジェクトをアサインする。
The playlist of the Music Playlist container has no assigned objects.	ダブルクリックして、Music Playlist Editorを開き、プレイリストにミュージックオブジェクトをアサインする。

## Wwiseのプロジェクトについて

---

Issue	Information/Suggestion
Music Playlist Containerのプレイリストにオブジェクトが、アサインされていない。	
A blend track in the blend container has no assigned objects. Blend Containerのブレンドトラックに、オブジェクトがアサインされていない。	ダブルクリックして、Blend Track Editorを開き、以下のいずれかを行う。 ブレンドトラックにオブジェクトを追加する。 ブレンドトラックを削除する。
A music track in the Music segment contains no audio files. Music Segmentのミュージックトラックに、オーディオファイルが入っていない。	ダブルクリックして、Music Segment Editorを開き、以下のいずれかを行う。 ミュージックトラックにクリップを追加する。 ミュージックトラックを削除する。
The SoundBank refers to objects that no longer exist. SoundBankが参照するオブジェクトが、既に存在しない。	そのイベントまたはオブジェクトをSoundBankから削除する。
No object assigned to event action. Event Actionにオブジェクトがアサインされていない。	メッセージをダブルクリックして、Event Editorを開き、以下のいずれかを行う。 これらのアクションにオブジェクトをアサインする。 Event Actionを削除する。
No object assigned to path in dialogue event. ダイアログイベントのパスに、オブジェクトがアサインされていない。	メッセージをダブルクリックして、Dialogue Event Editorを開き、以下のいずれかを行う。 パスにオブジェクトをアサインする。 パスを削除する。
Some child objects in this sample accurate container are not using the same output properties as the parent container. 本サンプルアキュレートコンテナのいくつかの子オブジェクトが、親コンテナと同じアウトプットプロパティを使用していない。	親コンテナがサンプルアキュレートでない可能性がある。ダブルクリックして、Schematic Viewを開き、このサンプルアキュレートコンテナの子オブジェクトのアウトプットオーバーライド設定を削除する。
Some child objects in this sample accurate container are not using the same effect properties as the parent container. 本サンプルアキュレートコンテナのいくつかの子オブジェクトが、親コンテナと同じエフェクトプロパティを使用していない。	親コンテナがサンプルアキュレートでない可能性がある。ダブルクリックして、Schematic Viewを開き、このサンプルアキュレートコンテナの子オブジェクトのエフェクトオーバーライド設定を削除する。
Some child objects in this sample accurate container are not using the same advanced settings properties as the parent container. 本サンプルアキュレートコンテナのいくつかの子オブジェクトが、親コンテナと同じ詳細設定プロパティを使用していない。	親コンテナがサンプルアキュレートでない可能性がある。ダブルクリックして、Schematic Viewを開き、このサンプルアキュレートコンテナの子オブジェクトの詳細設定のオーバーライドを削除する。
Some containers in this sample accurate container are not sample accurate.	親コンテナがサンプルアキュレートでない可能性がある。ダブルクリックして、Project Explorerを開き、このサンプルアキュレートコンテナの子コンテナのうち、サンプルアキュレートでないものをサンプルアキュレートに変更する。

Issue	Information/Suggestion
本サンプルアキュレートコンテナのいくつかのコンテナが、サンプルアキュレートでない。	
This sample accurate container has a child switch container. 本サンプルアキュレートコンテナに子 Switch Containerがある。	親コンテナがサンプルアキュレートでない可能性がある。ダブルクリックして、Project Explorerを開き、Switch Containerを削除する。
An audio source in this sample accurate container does not have the same number of channels as other audio sources in the container. 本サンプルアキュレートコンテナ内に、他とはチャンネル数の違うオーディオソースが1つ入っている。	親コンテナがサンプルアキュレートでない可能性がある。本コンテナのオーディオソースが全て、同じチャンネル数となることを確認する。
Some audio files are shorter than 0.05 seconds. These cannot be used for crossfade transitions. 0.05秒より短いオーディオファイルがある。これらはクロスフェードによるトランジションに使用できない。	ダブルクリックして、Project Explorerを開き、クロスフェードトランジションには短すぎるオーディオファイルの関連するサウンドオブジェクトを探して、以下のいずれかを行う。 より長い別のオーディオファイルを、サウンドオブジェクトの上までドラッグ＆ドロップする。 親オブジェクトのProperty Editorで、別の種類のトランジションを選択する。
Switch containers with more than one object assigned to a switch do not crossfade correctly. 複数のオブジェクトが1つのSwitchにアサインされているSwitch Containerが、正常にクロスフェードしない。	ダブルクリックして、Project Explorerを開き、関連するProperty Editorにおいて、クロスフェードトランジションのSwitchに、オブジェクトが1つだけ、アサインされることを確認する。
A container with a trigger rate duration of less than 0.021 seconds may not play as expected. トリガーレートの長さが0.021秒未満のコンテナは、計画通りに再生されない可能性がある。	ダブルクリックして、Property Editorを開き、関連するProperty Editorにおいて、Triggerレート時間を0.021秒より長く設定する。
This source only contains an LFE channel. 本ソースに含まれるのは、LFEチャンネルのみ。	
This source contains an LFE channel. 本ソースにLFEチャンネルが含まれる。	

## インテグレティレポートの保存

問題を解決する時の参考資料として、またプロジェクトで発生した問題の記録として、インテグレティレポートを保存すると良いでしょう。

### インテグレティレポートを保存するには:

1. Integrity Reportで、Save Reportをクリックする。

Save Asダイアログボックスが開く。

2. レポートの保存先フォルダまで移動する。
3. デフォルトのレポート名を、レポート内容が分かる名前に変更して、Saveをクリックする。

指定した場所に、インテグレティレポートがテキストファイルとして保存される。

## 学習コーナー: オブストラクションとオクルージョンについて

オブストラクションやオクルージョンは、サウンドソースとリスナーの間の空間が塞がれた時に発生する状況です。オブストラクションは部分的に塞がれた状況で、オクルージョンは完全に塞がれた状況です。これらの条件はゲーム開発者が設定して、ゲームのジオメトリをプログラミングで定義します。Wwiseはオブストラクションやオクルージョンのレベルを計算しません。物理的な計算をゲーム側で行い、その結果を以下の関数を使って、サウンドエンジンに送ります: [SetObjectObstructionAndOcclusion\(\)](#)。

Wwiseでは、ゲーム中にサウンドオブジェクトがオブストラクションやオクルージョンの影響を受けた時に適用する、ボリュームやLPFを設定するだけです。設定値は相対値であり、ゲームオブジェクトに既に設定されたボリューム値やLPF値に対して、付加されます。

これらの条件の実装において、オブストラクションとオクルージョンに、率 (%) をアサインして、よりリアルにします。この割合は、ゲームエンジンから来るリアルタイムのポジション情報に基づき変化して、塞がれたエリアを出入りするゲームオブジェクトの動きを反映します。例えば、スパイのキャラクターが美術館の中の大きな画像の間を動くとします。オブストラクション率を、スパイが画像の後ろを通る時の位置によって変化させることもできます。同様に、オクルージョンの割合を、完全に閉塞した用具入れからスパイがゆっくりと忍び出る時に、変化させることもできます。

### 注記

 オブストラクション率 (%) やオクルージョン率 (%) の計算は、サウンドエンジンが担当します。Wwiseでは数値を特定しません。

オブストラクションやオクルージョンがいつ発生するのかを、より良く理解するために、以下のトピックも参照してください。

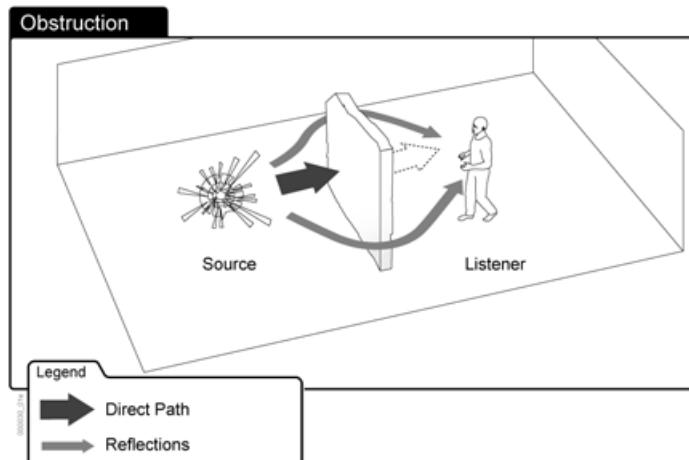
- ・「[オブストラクションについて](#)」
- ・「[オクルージョンについて](#)」

### オブストラクションについて

オブストラクションは、ゲーム中にジオメトリにある壁や柱などの物体が、サウンドソースとリスナーの間の空間を部分的に塞ぐ時に起きる状況です。例えば、スパイゲームでプレイヤーキャラクターが柱の影に隠れても、前方の銃声は聞こえます。

オブストラクションが起きると、リスナーに音の反響 (Reflections) がはつきりと聞こえますが、音の直接パス (Direct Path) が妨害されます。この時の音の修正は、ボリューム減少かLPF増加、または両方を適用することで実現します。

オブストラクションの状況を、下図に示します。

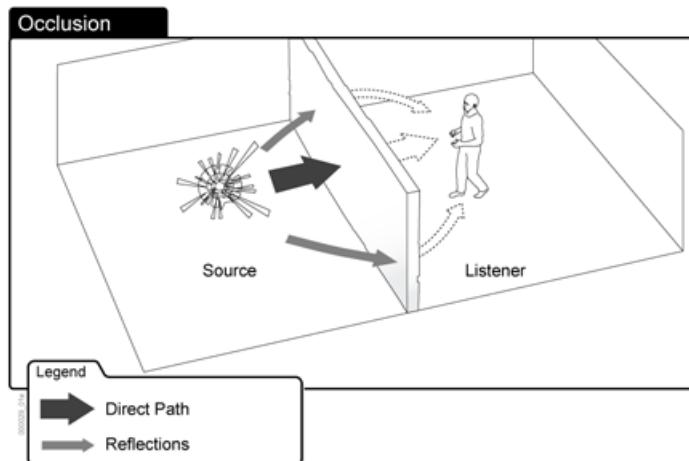


### オクルージョンについて

オクルージョンは、ゲームのジオメトリにある物体が、サウンドソースとリスナーの間の空間を完全に塞ぐ時に起きる状況です。例えば、スパイゲームで隣の部屋の銃とプレイヤーキャラクターの間に壁があるのにも関わらず、発砲音が壁越しに聞こえます。

オクルージョンが起きると、音の直接パス (Direct Path) と反響 (Reflections) パスの両方が妨害されます。オブストラクションと同様に、音のボリュームとLPF の両方を調整できます。

オクルージョンの状況を、下図に示します。



## プロジェクトマネジメントに関する、コツとベストプラクティス

プロジェクトを効率的に管理するための、コツやベストプラクティスについて、以下のセクションを参照してください。

### インテグレティレポートの活用

プロジェクトの様々な課題に対処するために定期的にインテグレティレポートを生成することは大事ですが、特に以下の段階では特に推奨されます。

- SoundBanks生成前 - オーディオやモーションをSoundBankにパッケージングする前にプロジェクト内の問題を解決するために。
- 大幅な変更をWork Unitにチェックインする前 - 変更点によってプロジェクトに問題が発生しないかを確認するために。

### デフォルト設定の活用

デフォルト設定を活用して、プロジェクトのルーティングやボリュームを「一括編集」できます。どのデフォルト設定値を変更しても、変更後のデフォルト値が、新しく追加するオブジェクトに適用されます。つまり、これから加えるオブジェクトに合わせて、デフォルト値を設定することができます。

### オブストラクションカーブやオクルージョンカーブの設定

グラフのカーブを必ず直線の組み合わせとして、プロジェクトのCPU負荷やメモリ使用量を抑えるように心がけてください。グラフはできるだけシンプルな形からスタートして、やむを得ない場合のみカスタマイズします。

### 大きなプロジェクトを、Work Unitで分割

大きなプロジェクトを数人で作業する場合、Work Unitを使ってプロジェクトを細かく分けることができます。Work Unitとは、プロジェクト内の特定の部分やエレメントに関連した情報を含む、固有のXMLファイルです。Work Unitを使って、プロジェクトの様々なエレメントを整理し管理します。チームとして作業を進める時にソースコントロールシステムで全てのWork Unitを管理することで、複数のメンバーによる同時作業がしやすくなります。また、プロジェクトをWork Unitで分けた後に一部のWork Unitをプロジェクトからアンロードして、自分が作業をするものだけを維持することもできます。Work Unitを部分的にアンロードする主な理由は、プロジェクトのロード時間の短縮、メモリ使用量の削減、そしてWwiseの全体的なパフォーマンスの向上です。プロジェクト自体が非常に大きい場合、Work Unitを1つ以上アンロードすることで、パフォーマンスが大幅に改善されます。

---

## 第4章 プラットフォームの管理

概要 .....	92
Platform Managerを使用する .....	92

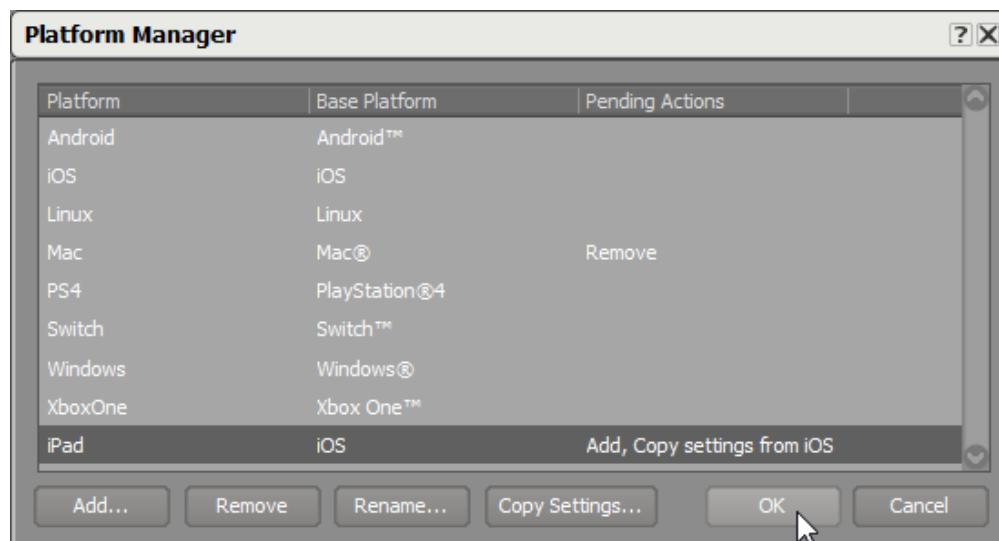
## 概要

一般的にゲームは複数のプラットフォーム向けにリリースされます。そのため、プロジェクトの対象となるプラットフォームを指定できることが不可欠です。WwiseではこれをPlatform Managerで行います。

WwiseメニューでProject > Platform Manager...を選択してPlatform Managerを起動します。デフォルトショートカットは、Shift+Alt+Pです。

## Platform Managerを使用する

以下に示すPlatform Managerで、ゲームの対象プラットフォームを定義できます。プラットフォームはプラットフォーム名と、ベースとなるSDKで成立しています。リストにあるSDKに対して、必要な数のプラットフォームを自由に設定できます。



### Platform Manager

Wwiseが提供するベースプラットフォームのリストで、Wwiseで使う主要SDKを網羅していますが、プロジェクトに関連する各種ベースプラットフォームに合わせてプラットフォームを作成することもできます。また、1つのベースプラットフォームを使って複数のプラットフォームを追加して、様々な差分に対応することもできます。

例えば、プロジェクトでiPhoneとiPadの両プラットフォームの開発を進めているとします。基本的に両者ともiOSプラットフォームですが、厳密には同じではありません。異なるプロジェクトタイプの管理を効率的に行えるように、iOSベースのプラットフォームを2つ作成して、iPad専用とiPhone専用に分けることができます。

### 関連トピック

- 「[プラットフォームの追加、削除、コピー](#)」

## プラットフォームの追加、削除、コピー

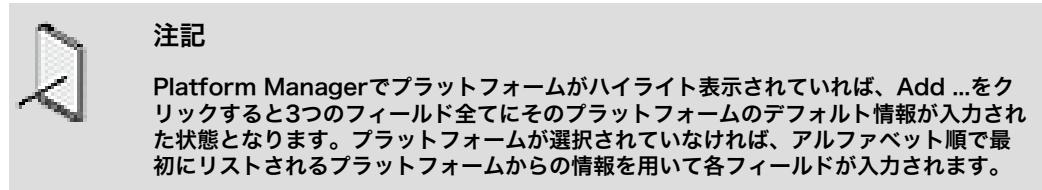
WwiseメニューバーでProject > Platform Manager...を選択する。

Platform Managerが表示される。

### プラットフォームを追加するには

1. Add...をクリックする。

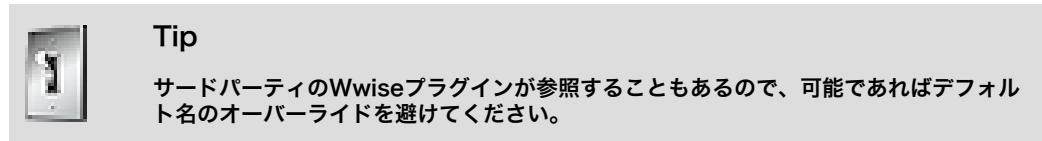
Add Platformダイアログボックスが開く。



2. SDKのリストからプラットフォームのBase Platformを選択する。

Nameフィールドが適宜更新される。

3. 必要であれば、Override Default Nameを選択する。Nameフィールドが編集可能となるので、新しいプラットフォーム名を設定できる。英数字（スペース不可、アンダースコア可能）を使い、既定のプラットフォーム名とは異なる名前にする必要がある。



4. または、既存プラットフォームをCopy settings fromで選択する。
5. 新しいプラットフォームの情報を確認して、OKをクリックしてAdd Platformダイアログを閉じる。

新しいプラットフォーム名、ベースプラットフォーム、今後のアクションであるPending Actions (Addと、場合によってはCopy settings from...) がAdd Platformダイアログの新しい最後の行の、それぞれの列に表示される。



6. OKをクリックして追加した新しいプラットフォームを加える。

このプラットフォーム変更はやり直せない旨の警告メッセージが表示される。

7. YesをクリックしてPending Actionsのアクションを実行する。

プロジェクトのリロード後にプラットフォームへの変更、特にパス名の変更を確認するよう別の警告メッセージが表示される。

8. OKをクリックしてプロジェクトをリロードする。

一旦プロジェクトが閉じ、新しいプラットフォーム設定でリロードされる。プロジェクトのサイズによっては、数秒かかることがある。

#### プラットフォームを削除するには

1. 削除するプラットフォームをリストで選択する。
2. Removeをクリックする。
3. 選択中のプラットフォームのPending Actions列にRemoveと表示される。



4. OKをクリックして選択したプラットフォームの削除を実行する。

このプラットフォーム変更はやり直せない旨の警告メッセージが表示される。

5. YesをクリックしてPending Actionsのアクションを実行する。

プロジェクトのリロード後にプラットフォームへの変更、特にパス名の変更を確認するよう別の警告メッセージが表示される。

6. OKをクリックしてプロジェクトをリロードする。

一旦プロジェクトが閉じ、新しいプラットフォーム設定でリロードされる。プロジェクトのサイズによっては、数秒かかることがある。

#### プラットフォームの設定をコピーするには

既にあるプラットフォームの専用設定を別のプラットフォームにコピーすることができます。

両プラットフォームの設定の大部分が同じであれば、この機能で作業時間を大幅に短縮できます。

1. Copy settings...をクリックする。

Copy Platform Settingsダイアログボックスが開く。

2. Fromリストでソースとなるプラットフォームを選択する。
3. Toリストでデスティネーションとなるプラットフォームを選択する。
4. OKをクリックして追加した新しいプラットフォームを加える。

このプラットフォーム変更はやり直せない旨の警告メッセージが表示される。

5. YesをクリックしてPending Actionsのアクションを実行する。

プロジェクトのリロード後にプラットフォームへの変更、特にパス名の変更を確認するよう別の警告メッセージが表示される。

6. OKをクリックしてプロジェクトをリロードする。

一旦プロジェクトが閉じ、新しいソースからターゲットにコピーされるプラットフォーム用の設定：でリロードされる。プロジェクトのサイズによっては、数秒かかることがある。



**注意**

変更を元に戻したい時のために、必ずプロジェクトの最新バックアップをとってください。



**警告**

この操作を元に戻すことはできず、ClipboardにあるWwise関連のコンテンツや、Undo（元に戻す）履歴は、消去されます。

## 関連トピック

- ・ [「あるプラットフォームの設定を、他のプラットフォームにコピーする」](#)

---

## 第5章 ランゲージの管理

概要 .....	97
Language Managerを使う .....	97

## 概要

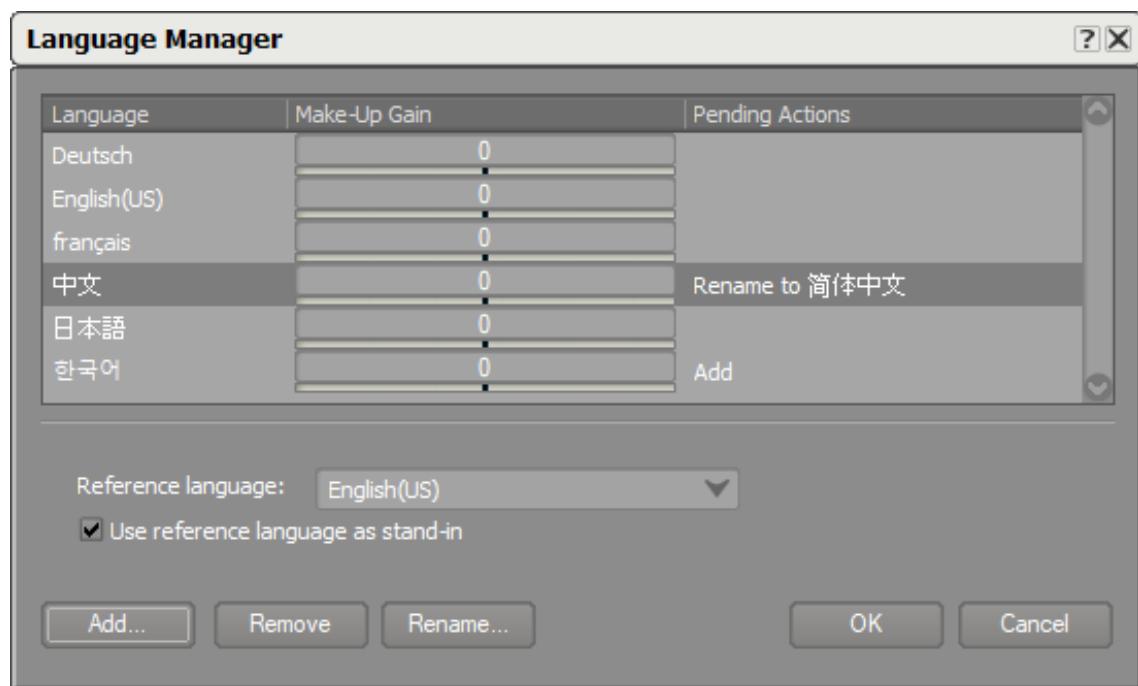
一般的にゲームは複数のランゲージ向けにリリースされます。そのため、プロジェクトの対象となる複数のランゲージを指定できることが不可欠です。WwiseではこれをLanguage Managerで行います。

WwiseメニューでProject > Languages...を選択してLanguage Managerを起動します。デフォルトショートカットはShift+J。

## Language Managerを使う

ローカリゼーション対象のランゲージ（言語）が決定したら、プロジェクト用にこれらのランゲージの作成を開始します。プロジェクトで使う全てのランゲージが、ツールバーのランゲージセレクタや、ボイスオブジェクトのContents Editorに表示されます。ランゲージリストを作成する時点で、全てのランゲージファイルが準備できているわけではないので、準備中のランゲージの代わりに使えるレファレンスランゲージ（Reference language）として、言語を1つ、選択します。また、レファレンスランゲージに設定したコンバージョン設定を、インポートする他のランゲージファイルのデフォルトのコンバージョン設定とすることもできます。

Language Managerで、プロジェクトのランゲージを選択して、レファレンスランゲージとして使う言語を指定します。



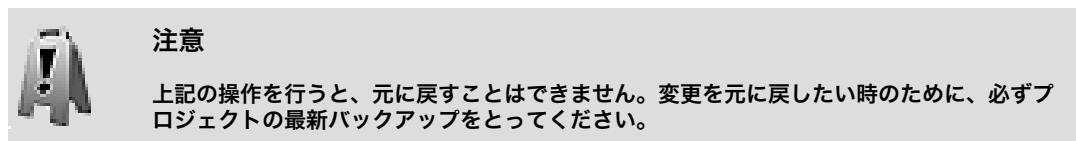
### プロジェクトランゲージを設定する

プロジェクトで使うランゲージは、Language Managerで設定します。ランゲージを設定する時に、ランゲージファイルのボリュームオフセットも設定できます。

WwiseメニューでProject > Languages...を選択する。Language Managerが表示される。

ここで、以下を行えます:

- ランゲージを追加する
- ランゲージを削除する
- ランゲージ名を変更する
- レファレンスランゲージを設定する

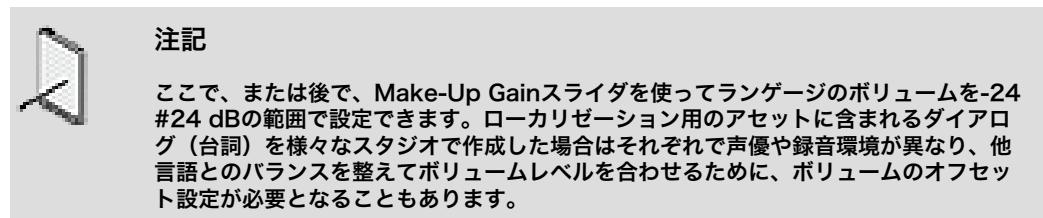
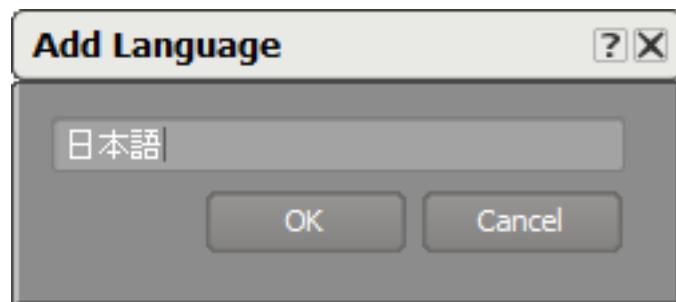


### ランゲージを追加するには

1. Add...をクリックする。

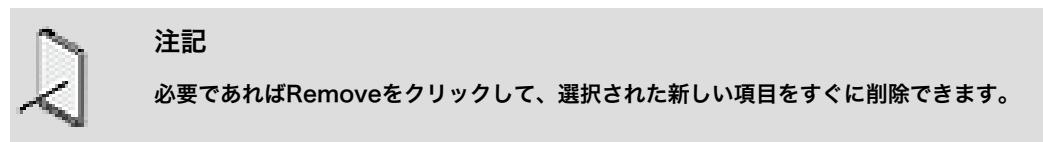
Add Language ダイアログボックスが開く。

2. ランゲージ名を、OSが対応している文字を使って入力する。



3. 新しいランゲージを確認して、OKをクリックしてAdd Languageダイアログを閉じる。

新しいランゲージと次のアクション ("Add") がLanguage Managerの新しく追加された最後の行に、それぞれの列に表示される。



4. OKをクリックして追加した新しいランゲージを加える。

メッセージボックスが表示され、Wwiseクリップボードの内容と、Undo (元に戻す) 機能の履歴が消去される、という警告が表示される。

5. YesをクリックしてPending Actionsのアクションを実行する。

ランゲージ変更に伴い、コードを検証して、特にGeneratedSoundBanksとOriginalsのパスを確認するよう、別の警告メッセージが表示される。

6. OKをクリックして、設定変更を続ける。

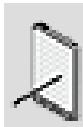
Process Logが開き、指定したアクション成功のメッセージが表示される。プロジェクトのLanguage Selectorリストがアップデートされ、新しく追加したランゲージも表示される。

**ランゲージを削除するには**

1. 削除するランゲージをリストで選択する。

2. Removeをクリックする。

3. 選択中のランゲージのPending Actions列にRemoveと表示される。



**注記**

Removeアクションをキャンセルするには、Language Managerを閉じるかCancelをクリックします。

4. OKをクリックして選択したランゲージの削除を実行する。

このランゲージ変更はやり直せない旨の警告メッセージが表示される。

5. YesをクリックしてPending Actionsのアクションを実行する。

ランゲージの変更を検証して、特にGeneratedSoundBanksとOriginalsのパス名の変更を確認するよう、別の警告メッセージが表示される。

6. OKをクリックして、設定変更を続ける。

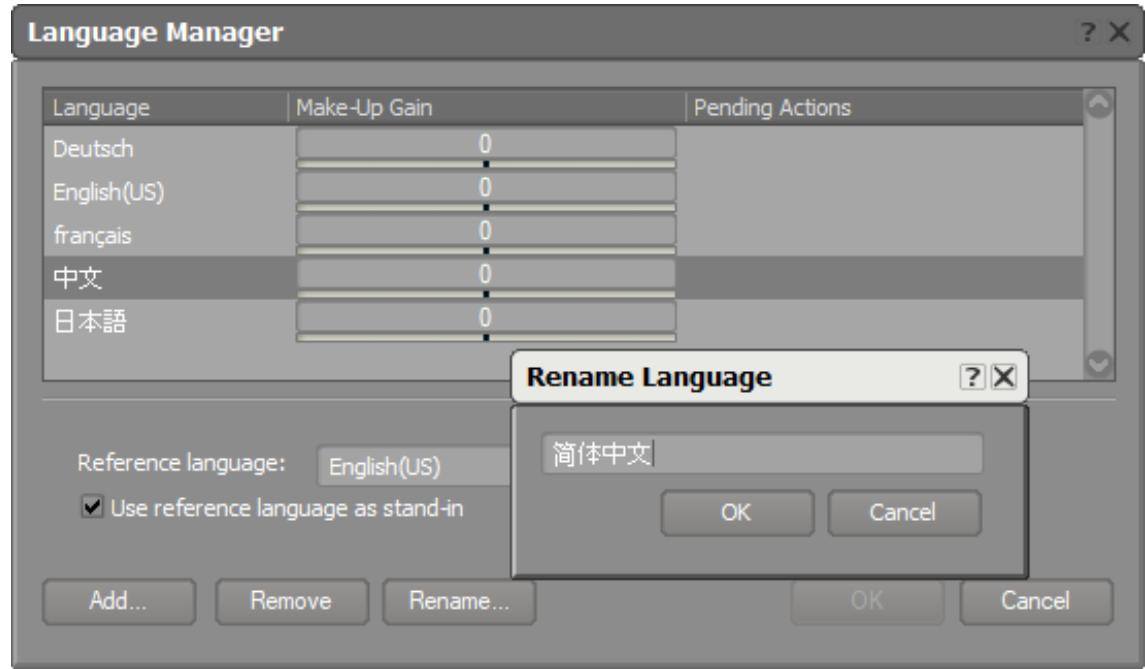
Process Logが開き、指定したアクション成功のメッセージが表示される。プロジェクトのLanguage Selectorリストがアップデートされ、指定したランゲージが削除される。

**ランゲージ名を変更するには**

1. Rename...をクリックする。

Rename Languageダイアログが開く。

2. 新しいランゲージ名を、OSが対応している文字を使って入力する。



- 新しいランゲージを確認して、OKをクリックしてRename Languageダイアログを閉じる。

新しいランゲージ名と次のアクション（"Rename 'language'"）が選択したランゲージの、それぞれの列に表示される。

- OKをクリックして追加した新しいランゲージを加える。

メッセージボックスが表示され、Wwiseクリップボードの内容と、Undo（元に戻す）機能の履歴が消去される、という警告が表示される。

- YesをクリックしてPending Actionsのアクションを実行する。

ランゲージ変更に伴い、コードを検証して、特にGeneratedSoundBanksとOriginalsのパスを確認するよう、別の警告メッセージが表示される。

- OKをクリックして、設定変更を続ける。

Process Logが開き、指定したアクション成功のメッセージ（またはエラー）が表示される。プロジェクトのLanguage Selectorリストがアップデートされ、ランゲージの新しい名前も表示される。

#### レファレンスランゲージを変更するには

- Reference languageリストでレファレンスランゲージを選択する。
- Use reference language as stand-inのオプションを選択することもできる。これを選択すると、オーディオファイルのローカライズしたバージョンがない場合は、そのランゲージのSoundBankに、レファレンスランゲージのオーディオファイルが代わりに入れられる。
- 新しいレファレンスランゲージと、代替として使用するかどうかのステータスを確認して、OKをクリックする。

メッセージボックスが表示され、Wwiseクリップボードの内容と、Undo（元に戻す）機能の履歴が消去される、という警告が表示される。

4. YesをクリックしてPending Actionsのアクションを実行する。

ランゲージ変更に伴い、コードを検証して、特にGeneratedSoundBanksとOriginalsのパスを確認するよう、別の警告メッセージが表示される。

5. OKをクリックして、設定変更を続ける。

Process Logが開き、指定したアクション成功のメッセージ（またはエラー）が表示される。

## 関連トピック

- [「プロジェクトのローカリゼーション作業」](#)

---

## 第6章 Workgroupの活用

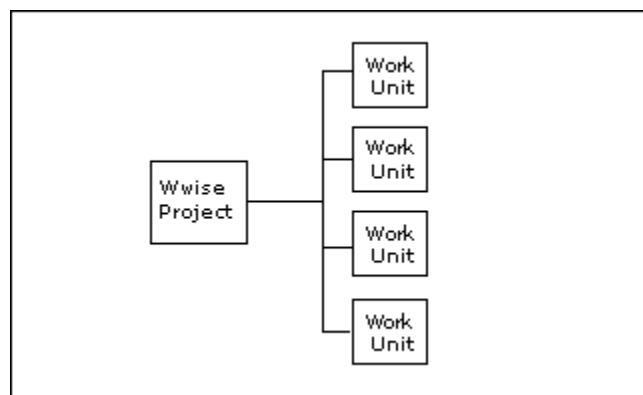
概要 .....	103
プロジェクトをWork Unitで分割 .....	106
プロジェクトファイルのステータス確認 .....	114
Wwiseとソースコントロールシステムを使う .....	116
プロジェクトの不整合の解消 .....	116
Workgroupプラグインを使ったプロジェクトファイルの管理 .....	118
Workgroupに関する、コツとベストプラクティス .....	131

## 概要

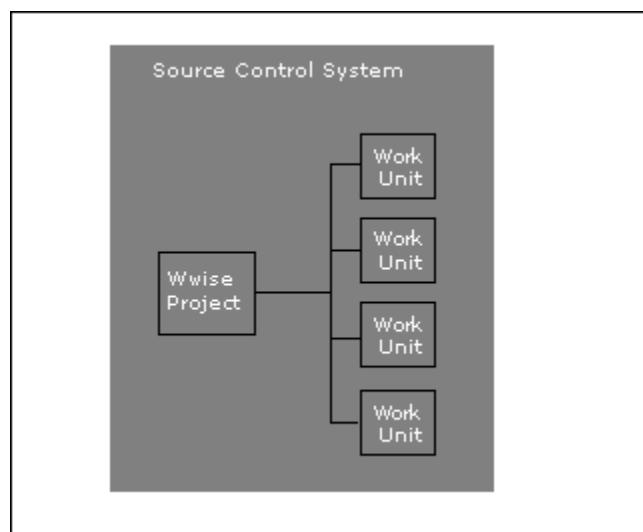
今日のゲーム環境において、ゲームが急速に複雑化し、市場に迅速にリリースする必要性が高まる中、サウンドデザイナー、ミュージックコンポーザー、オーディオインテグレーター、オーディオプログラマーなどが、同じプロジェクトに参加して共同で作業をできることが不可欠です。WwiseのWorkgroupを利用すれば、これが可能です。

1つのゲームに対して、Wwiseで使うプロジェクトは1件だけです。

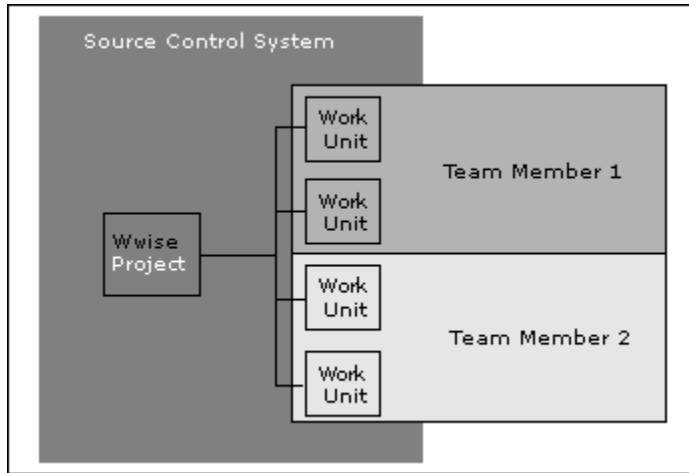
同じプロジェクトで複数の関係者が効率的に作業するには、プロジェクトを小分けにする必要があります。Wwiseでは、Work Unitという単位に分けます。



Work Unitのファイルは、ユーザーが選択したソースコントロールシステムで管理できます。



これにより作業チームの各メンバーが、プロジェクトの同じ部分または異なる部分の作業を、併行して行えます。



一般的に、メンバーがそれぞれプロジェクトの別の部分で作業すれば、マージ段階で頻出する困難な問題を回避できます。しかし、同じWork Unitを2人以上のメンバーが同時に作業することが不可欠な場合も考えられます。このような状況では、ファイルをソースコントロールシステムにチェックバックする時に、マージのコンフリクトが発生する可能性が高くなります。マージコンフリクトに対処する最善の方法については、使用するソースコントロールのドキュメンテーションを参照してください。

Wwiseはソースコントロール管理システムではありませんが、Wwiseのオープンアーキテクチャを利用して、採用したソースコントロールシステムを簡単に実装できます。その結果、プロジェクトのアセット管理やソースコントロール機能の多くを、直接Wwiseから実行できるようになります。Wwiseのソースコントロールプラグインについては「[Workgroupプラグインを使ったプロジェクトファイルの管理](#)」を参照してください。



### 注記

ソースコントロールシステムのWorkgroupプラグインを作成するには、ソースコントロールシステム側でAPIを使ったサードパーティインテグレーションに対応していなければなりません。なお、Wwiseには、PerforceとSubversionのプラグインがインストールされています。

## Work Unitとは

WwiseのWorkgroupの基礎となるのが、Work Unitです。Work Unitとは、プロジェクト内の特定の部分やエレメントに関連した情報を含む、固有のXMLファイルです。Work Unitを使って、プロジェクトの様々なエレメントを整理し管理します。チームとして作業を進める時にソースコントロールシステムで全てのWork Unitを管理することで、複数のメンバーによる同時作業がしやすくなります。

プロジェクトを作成すると、Wwiseの以下のエレメントごとに、Default Work Unitが作成されます。

- Actor-Mixer Hierarchy
- Audio Devices
- Attenuations

- Control Surface Sessions
- Conversion Settings
- Dynamic Dialogue
- エフェクト
- Events
- Game Parameter
- Interactive Music Hierarchy
- Master-Mixer Hierarchy
- Mixing Sessions
- Modulators
- プリセット
- Queries
- SoundBank
- Soundcaster Sessions
- States
- Switches
- Trigger

Default Work Unitは、プロジェクトディレクトリの該当フォルダの中に作成されます。それぞれ“Default Work Unit.wwu.”という名前が付きます。オブジェクト、Event、Stateなどのプロジェクトエレメントを作成する前にそれぞれのWork Unitを準備しなくてもよいように、Default Work Unitが作成されます。

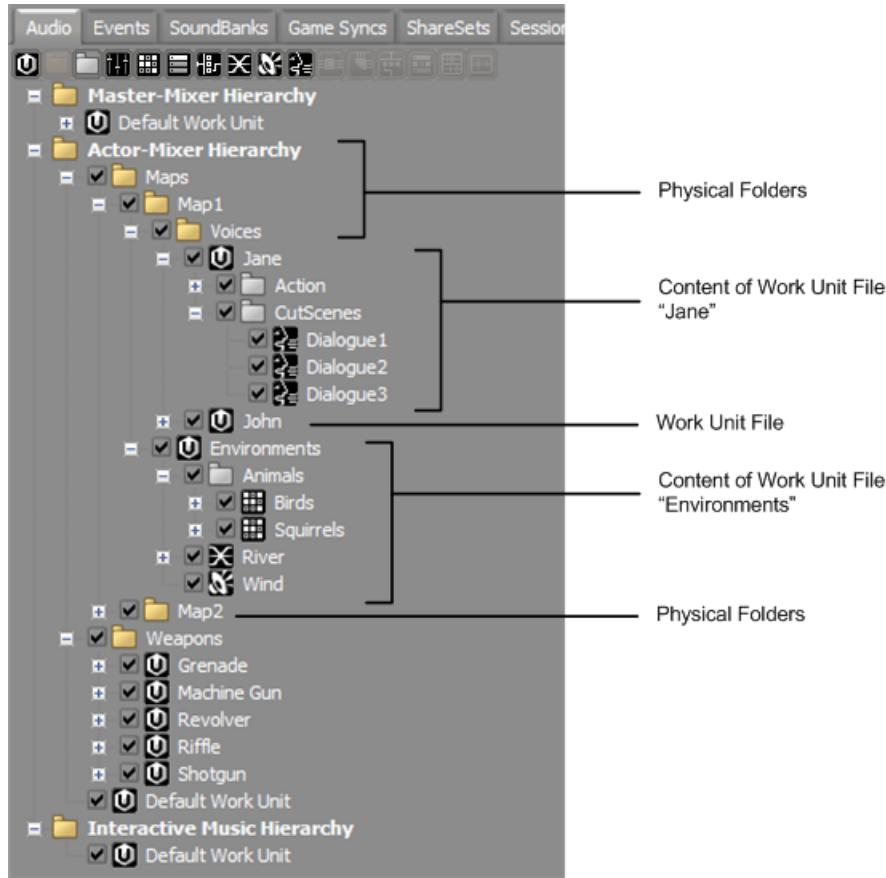
プロジェクトが大きくなり始めた時や、プロジェクトチームのメンバーが増え始めた時は、プロジェクトの各エレメントを新しいWork Unitに分割できます。例えば、State用のWork Unitとして、StatesLevel1、StatesLevel2、StatesLevel3という3つのWork Unitを作成することができます。

新規Work Unitを作成する場合は、Default Work Unitに何も入っていない状態で残すことができます。Default Work Unitは重要なプロジェクトファイルなので、名前変更や削除をしてはいけません。もし名前変更や削除を行った場合は、次にプロジェクトを開いた時に、Wwiseが再度、作成します。

Work Unitはディスク上のフィジカルサブフォルダに入れて整理することもできます。フィジカルフォルダの構成が、Project Explorerで複製されます。

新規Work Unitは、プリセットとMaster-Mixer Hierarchyの2つのプロジェクトエレメント以外で、自由に作成できます。Master-Mixer Hierarchyのバスは、必ずDefault Work Unitの中に残しておかなければなりません。

下図は、Work Unitを作成して並べ替えることで、Actor-Mixer Hierarchyの中のサウンド構造を整理した例です。



Project Explorerのタブを切り替えて、Work Unitのコンテンツの作成や管理ができます。Work Unitを使ってプロジェクトを整理する方法については「[プロジェクトをWork Unitで分割](#)」を参照してください。

## プロジェクトをWork Unitで分割

プロジェクトを分ける前に、これからの中の作業の効率を高めてコンフリクトを最小限に抑えるために、各種エレメントの最良の分割方法を決定します。プロジェクトの分割方法はいくつもありますが、対象プロジェクトに最も適した方法を検討する時間を、充分にとることが重要です。

Work Unitを作成してから、サウンド構造、イベント、SoundBankなどを該当するWork Unitにドラッグして、作業内容を分割します。また、プロジェクトの自分のバージョンからいくつかのWork Unitをアンロードして、自分自身が担当するものだけを残せば、プロジェクトのロード時間やメモリ使用量を減らせます。

プロジェクト内のWork Unit管理には、以下のタスクが伴います。

- 「[プロジェクトのWork Unitの作成](#)」
- 「[Work Unitに、プロジェクトエレメントをアサイン](#)」
- 「[プロジェクトのWork Unitの、ロードとアンロード](#)」

## プロジェクトのWork Unitの作成

プロジェクトの以下のエレメント用に、新規Work Unitを作成できます。

- Actor-Mixer Hierarchyのサウンド構造
- Interactive Music Hierarchyのミュージック構造
- アクションイベント、ダイアログイベント
- SoundBank
- Switch、State、Game Parameter、Trigger
- Effect ShareSet、Attenuation ShareSet
- Soundcasterセッション、Mixing Deskセッション
- Queries

Work Unitを最初に作成した時のプロジェクトエレメント以外のものを、そのWork Unitに入れることはできません。例えば、イベント用のWork Unitにはイベントのみ、エフェクト用のWork Unitにはエフェクトのみを入れることができます。

ソースコントロールシステムを使ってプロジェクトファイルを管理する場合、Default Work Unitを含む全てのWork Unitを、ソースコントロールシステムのセントラルレポジトリまたはデボに追加できます。Workgroupプラグインを使用している時にWork Unitを作成すると、そのファイルをレポジトリに追加するようプロンプトが表示されます。

### Work Unitを作成するには:

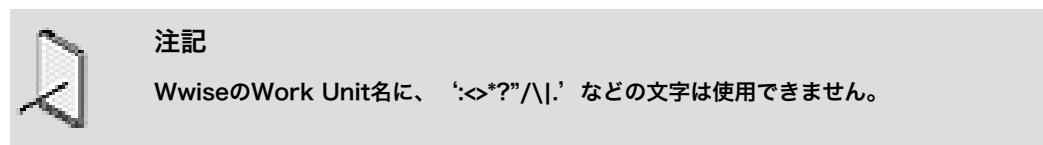
1. Project Explorerで、1つのフィジカルフォルダを右クリックする（例：`<e0>Actor-Mixer Hierarchy</e0>`、`<e1>Interactive Music Hierarchy</e1>`）。

ショートカットメニューが表示される。

2. **New Child > Work Unit**をクリックする。

Work Unitダイアログボックスが開く。

3. Nameフィールドに、Work Unitの名前を入力する。



4. OKをクリックする。

新規Work Unitが作成される。



## 関連トピック

- 「Work Unitに、プロジェクトエレメントをアサイン」
- 「ユーザーが作成したWork Unit名の変更」
- 「ユーザーが作成したWork Unitの削除」
- 「Workgroupに関する、コツとベストプラクティス」

## Work Unitに、プロジェクトエレメントをアサイン

プロジェクトのサウンド構造、イベント、SoundBankなど全てのエレメントは、自動的に該当するDefault Work Unitにアサインされます。新規Work Unitを作成した後、プロジェクト内の様々なエレメントをこれら新規Work Unitに振り分けることができます。エレメントをアサインしたいWork Unitにドラッグして入れることで、簡単にアサインできます。



### 注記

プロジェクトエレメントを新規Work Unitに移動したり既存のWork Unit間で移動させる時は、プロジェクトチームの他のメンバーがそのプロジェクトエレメントを使って作業をしていないことを、事前に確認してください。

## Work Unitにプロジェクトエレメントをアサインするには:

- Project Explorerで1つのタブをクリックする。（タブとは、例えばAudio、Events、SoundBanks、Game Syncs、ShareSets、Sessions、Queriesなど。）
- Default Work Unitにあるエレメントを1つ以上ドラッグして、自分で作成したWork Unitに入れる。

エレメントが新規Work Unitにアサインがされる。

## 関連トピック

- 「プロジェクトのWork Unitの作成」
- 「Workgroupに関する、コツとベストプラクティス」
- 「フィジカルフォルダの管理」
- 「Nested Work Units」
- 「プロジェクトのWork Unitの、ロードとアンロード」
- 「ユーザーが作成したWork Unit名の変更」
- 「ユーザーが作成したWork Unitの削除」

## フィジカルフォルダの管理

Work UnitはActor-Mixer Hierarchyなど特定のオブジェクトカテゴリーに関連するルートフォルダ内にあるディスク上のPhysical Folderに整理できます。何百というWork Unitがあり、1つのフォルダに一律に入れたくない場合などは、特に便利です。

フィジカルフォルダと仮想フォルダは、本質が全く異なります。フィジカルフォルダはディスク上の実際のディレクトリを表すのに対し、仮想フォルダはそれが入っ

ているWork Unitのファイルの一部です。両者ともコンテンツの整理に使いますが、フィジカルフォルダはWork Unitの先祖であり、仮想フォルダはWork Unitの子孫です。

フィジカルフォルダを、Project Explorer内で直接作成したい時は、他のフィジカルフォルダのコンテキストメニューを利用するか、Project Explorerのツールバー ボタンを利用します。

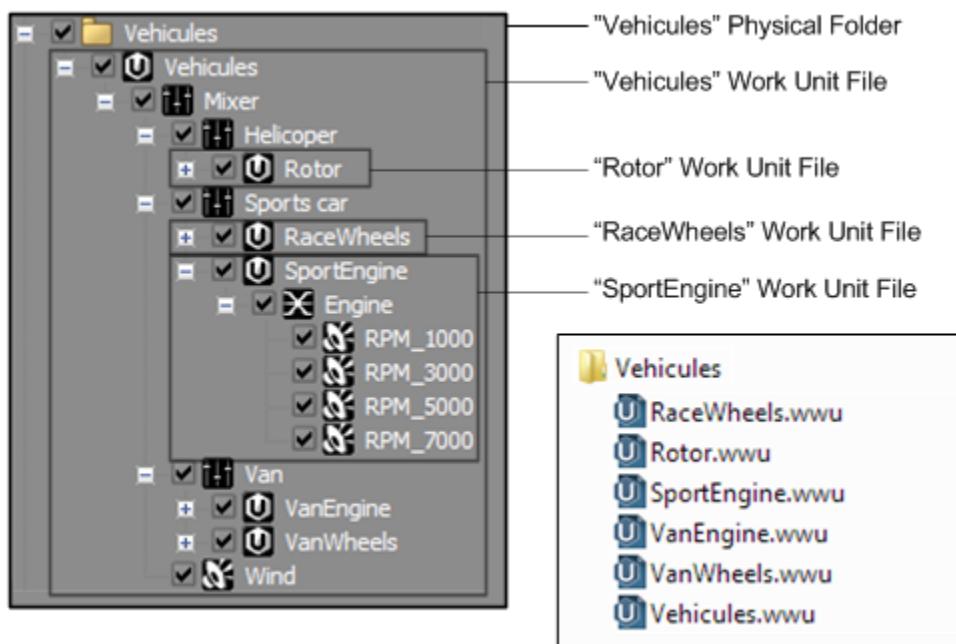
フィジカルフォルダに入れることができるのは、他のフィジカルフォルダとWork Unitファイルだけです。

### Nested Work Units

Work Unitを他のWork Unitの中にネスト化できます。ネスト化すると、Work Unitのコンテンツは親のWork Unitではなくネスト化したWork Unitに委任されます。これをを利用して、1つの共通Actor-Mixerオブジェクトの下に複数のWork Unitを置くことで、ネスト化したWork Unitに共通の動作やミキシングプロパティを設定できます。

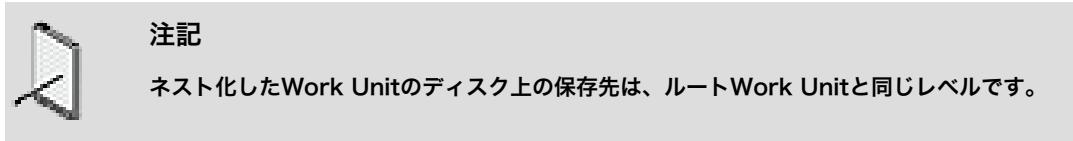
Work Unitをネスト化することによりファイルをさらに細かく分別できるので、マージ時のコンフリクトの可能性を減らせます。

ネスト化したWork Unitは、それが入れられたフィジカルフォルダの中に保存されます。Work Unit名は、1つのフィジカルフォルダ内で必ず固有の名前とします。



ネスト化したWork Unitを作成できるのは、以下の種類のオブジェクトの下です。

- Physical folders
- Work Unit
- Virtual folders
- Actor-Mixers

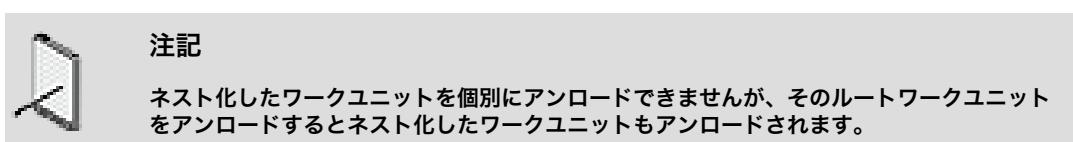
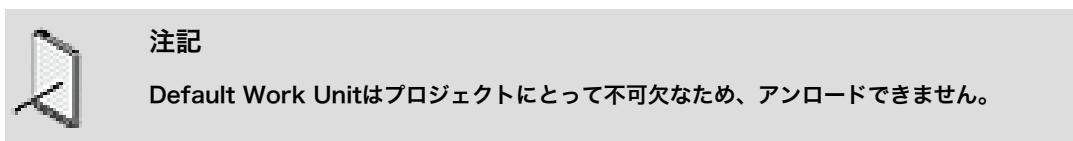


### プロジェクトのWork Unitの、ロードとアンロード

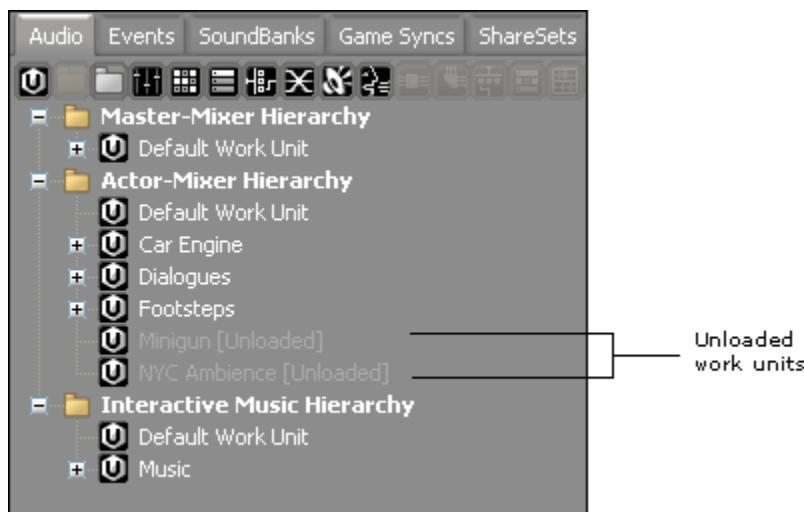
プロジェクトを複数のWork Unitに分割した後は、自分が作業するものだけを残して、他のWork Unitをアンロードできます。プロジェクトの一部のWork Unitをアンロードする主な目的は、プロジェクトのロード時間の削減、メモリ使用量の削減、そして全体のパフォーマンスの改善です。プロジェクト自体が非常に大きい場合、Work Unitを1つ以上アンロードすることで、パフォーマンスが大幅に改善されます。

プロジェクトからアンロードできるのは、ユーザーが作成した以下の種類のWork Unitです。

- Actor-Mixer Hierarchy Work Units
- Interactive Music Hierarchy Work Units
- Event Work Units



プロジェクトからアンロードしたWork Unitは、Project Explorerで、影表示となります。再度プロジェクトにロードするまで、そのWork Unitでは作業できません。



ロードしたWork Unitに入っているオブジェクトやイベントで、引き続き作業ができます。ただし、これらのオブジェクトやイベントの中には、アンロードされたオブジェクトやイベントへのレファレンスを含んでいるものがあるかもしれません。例えば、以下のプロジェクトエレメントの中に、アンロードされたオブジェクトやイベントへのレファレンスが含まれることがあります。

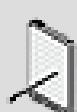
- Dialogue Events
- Events
- Music Stingers
- Music Transitions
- Queries
- SoundBank
- Soundcaster and Mixing Sessions

その場合、アンロードされたエレメントは[Unloaded]とマークされ、黄色ハイライトで表示されます。

Event Actions				
No.	PF	Actions	Objects	Scope
1	>>	<input checked="" type="checkbox"/>	Play	[Unloaded] Minigun_Barrel_Start Game object
2	>>	<input checked="" type="checkbox"/>	Play	[Unloaded] Fire Game object
3	>>	<input checked="" type="checkbox"/>	Play	[Unloaded] Fire_Sub Game object
4	>>	<input checked="" type="checkbox"/>	Play	[Unloaded] Shells Game object
5	>>			

引き続きオブジェクトの追加や削除、プロパティ値の変更、SoundBank生成などが可能ですが、Wwiseでこれらのオブジェクトやイベントを再生することはできません。

ユーザーがプロジェクトからWork Unitをアンロードしても、すぐにアンロードされません。アンロードするWork Unitをユーザーが選択した後にプロジェクトを閉じた時に初めて、プロジェクトからアンロードされます。次回プロジェクトを開く時、アンロードしたこれらのWork UnitをWwiseはロードしません。アンロードしたWork Unitの中のコンテンツをアクセスする必要がある場合は、いつでも簡単にプロジェクトに再ロードできます。また、アンロードしたWork Unit内のオブジェクトやイベントに影響するようなプロジェクト変更を実行した場合は、それらのWork Unitをプロジェクトに再ロードする時に、変更点が、Project Load Logに表示されます。



### 注記

Work Unitをプロジェクトからアンロードすると、使用されていないソースファイルと、プロジェクトにロードしていないソースファイルを、Wwise上で区別できません。その結果、全てのWork Unitを再ロードしてWwiseで完全な検証を実行するまで、プロジェクトで使用するファイルとしてWwise側で把握できないものは、File ManagerのUsage列で、Unknown (不明) とマークされます。

## プロジェクトからWork Unitをアンロードするには:

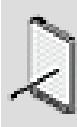
1. Project Explorerで、アンロードするWork Unit（単数または複数）を選択する。
2. 選択したWork Unitを右クリックして、メニューで、Unload Work Unit (at next project load)（プロジェクトの次回ロード時にWork Unitをアンロードする）を選択する。
3. Ctrl+Sを押してプロジェクトを保存する。
4. Project > Project Nameをクリックして、プロジェクトを再度開く。

プロジェクトが開く時、事前にアンロードしたWork Unitはロードされません。

## プロジェクトにWork Unitを再ロードするには:

1. Project Explorerで、アンロードされたWork Unitのうちプロジェクトに再ロードするもの（単数または複数）を選択する。
2. 選択したWork Unitを右クリックして、メニューで、Load Work Unit を選択する。

Work Unitがプロジェクトに再ロードされます。



### 注記

アンロードしたWork Unit内のオブジェクトやイベントに影響するようなプロジェクト変更を実行した場合は、それらのWork Unitをプロジェクトに再ロードする時に、変更点が、Project Load Logに表示されます。

## 関連トピック

- ・「プロジェクトのWork Unitの作成」
- ・「Work Unitに、プロジェクトエレメントをアサイン」
- ・「ユーザーが作成したWork Unit名の変更」
- ・「ユーザーが作成したWork Unitの削除」

## ユーザーが作成したWork Unit名の変更

プロジェクトで作成したWork Unit名を変更する必要が出てくることもあります。Work Unit名は、Project Explorerや、File Managerで変更できます。Windows ExplorerやMac FinderでWork Unitの名前を変更したり削除したりすると、インテグレティ上のエラーやプロジェクトデータの損失が発生するおそれがあるので、避けてください。



### 注意

Default Work Unitsはクリティカルなプロジェクトファイルであり、名前変更や削除をしてはいけません。もし名前変更や削除を行った場合は、次にプロジェクトを開いた時にWwiseが再度、作成します。

## Work Unit名を変更するには:

1. Project Explorerで、F2を押すか、コンテキストメニューRenameを使う。

Work Unit名の上に編集可能なテキストボックスが表示される。

2. 編集ボックスに新しい名前を入力し、Enterを押す。

ダイアログWorkgroup Operationが表示される。

3. 実行される操作を確認して、Continueをクリックする。

ソースコントロールプラグインを使用中であれば、操作が完了したことを確認すること。

4. OKをクリックして、Process Logを閉じる。

### 関連トピック

- [「Work Unitとは」](#)
- [「プロジェクトのWork Unitの作成」](#)
- [「Work Unitに、プロジェクトエレメントをアサイン」](#)
- [「ユーザーが作成したWork Unitの削除」](#)

### ユーザーが作成したWork Unitの削除

作成したWork Unitがプロジェクトで不要となり、削除する必要が出てくることもあります。Work Unitは、Project Explorerや、File Managerで削除できます。Windows ExplorerやMac FinderでWork Unitを削除すると、インテグレティ上のエラーやプロジェクトデータの損失が発生するおそれがあるので、避けてください。

ソースコントロールシステムからワークユニットファイルを削除すると、そのワークユニットのエレメントがプロジェクトで不在となるため、インテグレティエラーが起きることがあります。プロジェクトを開くと、これらのエラーと共にフィックス方法の詳細が、必要に応じて表示されます。プロジェクトの不整合については[「プロジェクトの不整合の解消」](#)を参照してください。



#### 注意

Default Work Unitsはクリティカルなプロジェクトファイルであり、名前変更や削除をしてはいけません。もし名前変更や削除を行った場合は、次にプロジェクトを開いた時にWwiseが再度、作成します。

## Work Unitを削除するには

1. 以下のいずれかを行う:

- Project Explorerで、削除するWork Unitを選択して、Deleteを押すか、コンテキストメニューのDelete機能を使う。
  - ダイアログWorkgroup Operationが表示される。

2. 実行される操作を確認して、Continueをクリックする。

ソースコントロールプラグインを使用中であれば、操作が完了したことを確認すること。

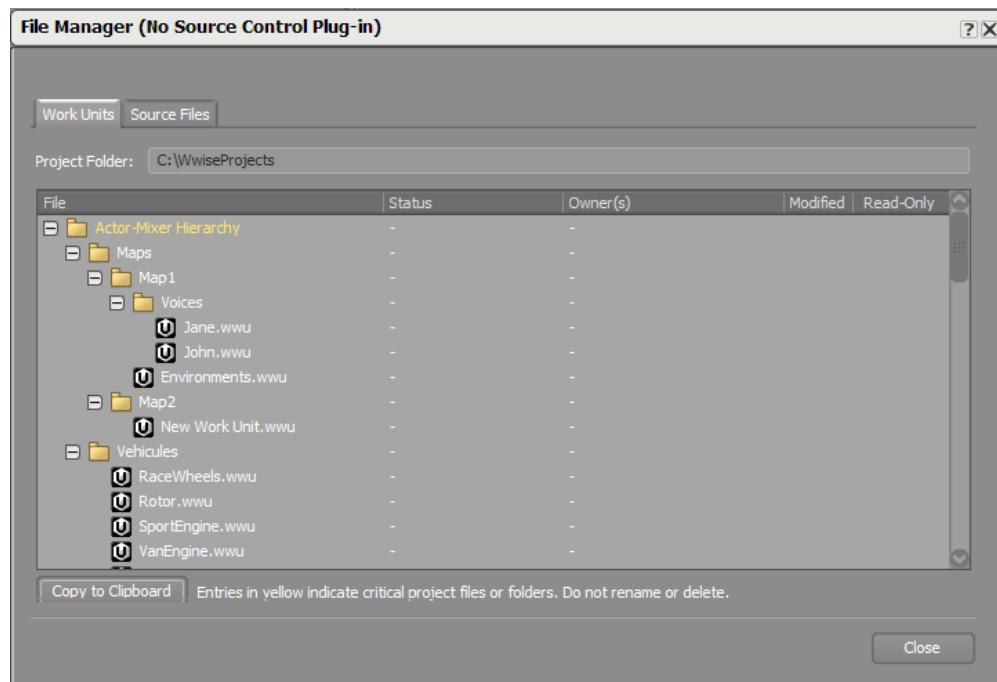
3. OKをクリックして、Process Logを閉じる。

## 関連トピック

- ・「Work Unitとは」
- ・「プロジェクトのWork Unitの作成」
- ・「Work Unitに、プロジェクトエレメントをアサイン」
- ・「ユーザーが作成したWork Unit名の変更」

## プロジェクトファイルのステータス確認

プロジェクトが大きくなると、プロジェクト内の様々なファイルをトラッキングする必要が出てきます。プロジェクト内の各オーディオファイルや各Work Unitの情報が、プロジェクトファイルの情報と共にFile Managerに表示されます。File Managerを使ってプロジェクトファイルやソースファイルのステータスとオーナーを確認すると共に、プロジェクト内のオーディオソースが特定のソースオーディオファイルを使っているかを判断できます。

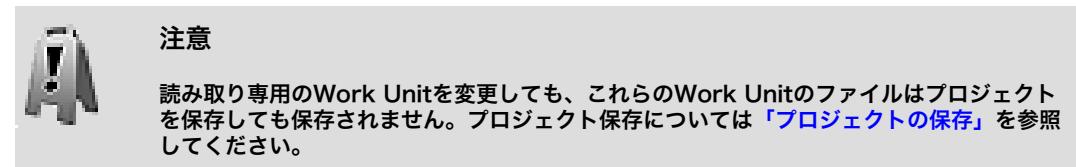
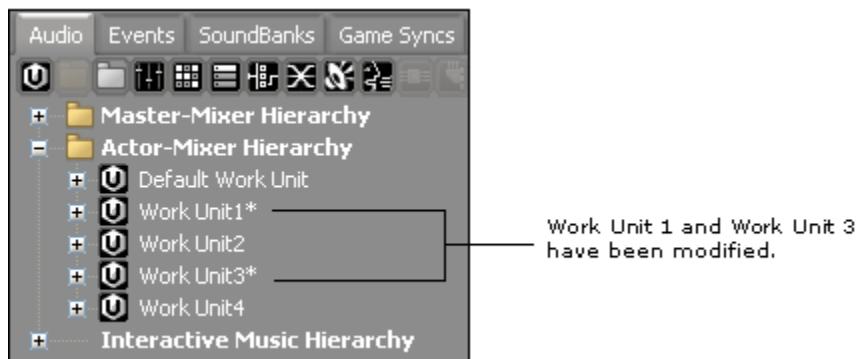


Work Unitが変更されたり読み取り専用タグが付けられると、プロジェクトファイルの該当コラムにチェックマークが付きます。このようなフィードバック情報を活用して、変更したファイルや変更できないファイルをトラッキングできます。

また、プロジェクトが使用しているファイルの情報が、File Managerに表示されるので、不要なファイルを定期的にクリーンアップできます。ただし、Work Unit

をプロジェクトからアンロードすると、使用されないソースファイルと、プロジェクトにロードしていないソースファイルを、Wwise上で区別できません。この場合、プロジェクトで使用していない様に見えるソースファイルは、Wwiseが完全な検証を実行するまで、Usage列で、Unknown（不明）とマークされます。

どのファイルが変更されたかのフィードバックが、WwiseのProject Explorerでも直接、表示されます。プロジェクトの表示と同様に、どのWork Unitが変更され保存を要するかが、\*（アステリスク）で示されます。



### プロジェクトファイルのステータスを確認するには:

1. 以下のいずれかを行う:

- メニューバーで、Project > File Manager...をクリックする。
- [Shift]+[F1]を押す。

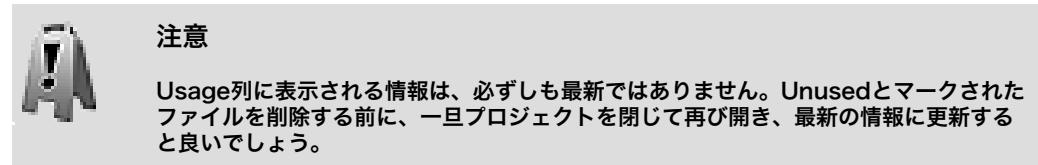
File Managerダイアログボックスが開く。

2. 各ファイルに関して、以下を確認する。

- 変更されたか。
- 読み取り専用か。

3. Source Filesタブに切り替えて各ソースファイルに関して、以下を確認する。

- 変更されたか。
- 読み取り専用か。
- プロジェクトの1つ以上のオーディオソースによって使用されているか。



4. OKをクリックして、Project File Statusダイアログボックスを閉じる。

## 関連トピック

- ・「プロジェクトの保存」
- ・「プロジェクトの不整合の解消」
- ・「Workgroupに関する、コツとベストプラクティス」

## Wwiseとソースコントロールシステムを使う

プロジェクトの開発環境でアセットや他のプロジェクトファイルを効率的に管理するために、既にソースコントロールシステムを使用しているかもしれません。Work UnitなどWwiseプロジェクトの全てのファイルがXMLベースなので、これらのファイルも同じソースコントロールシステムで簡単に管理できます。

プロジェクトの以下のファイルを、ソースコントロールシステムで管理できます。

- ・ Wwiseプロジェクトファイル - .wprojファイル。
- ・ WwiseWork Unit - .wwuファイル、Default Work Unitも含む。
- ・ Originalsフォルダ - Wwiseにインポートしたオリジナルサウンドファイルが入ったフォルダ。
- ・ 生成したSoundBank - プラットフォームやランゲージ別に生成したSoundBankファイル。



### 注意

プロジェクトディレクトリに存在する.cacheフォルダは、Wwiseの作業用ローカルフォルダです。.cacheフォルダの中身をソースコントロールシステムに追加すると、Wwiseが予期せぬ動作を起こすことがあるので避けてください。

ゲーム開発中にいつでもプロジェクトファイル (WPROJj) 、Work Unitファイル (WWU) 、オーディオファイルのステータスをFile Managerで確認できます。また、PerforceやSubversionなどのWorkgroupプラグインを使う場合、ソースコントロール機能をWwiseから直接、実行できます。WwiseでのWorkgroupプラグインの使い方については「[Workgroupプラグインを使ったプロジェクトファイルの管理](#)」を参照してください。

複数のメンバーが参加するWorkgroupによる作業でソースコントロールシステムを使ってプロジェクトファイルを管理する場合は、他人も同じプロジェクトで作業をしているので、マージコンフリクトの解消が必要になることがあります。頻繁に作業結果を同期してマージすることが重要で、常にチームメンバー同士がお互いの作業内容を共有できるようにしてください。ベストプラクティスのリストが「[Workgroupに関する、コツとベストプラクティス](#)」にあります。

## プロジェクトの不整合の解消

数人が同じプロジェクトで作業をする場合、特定ファイルを変更したため、プロジェクトファイルエラーや不整合が発生することがあります。このような問題の影響を抑えるために、プロジェクトファイルを開く度に、Wwiseは以下2種類のプロジェクト検証を行います。

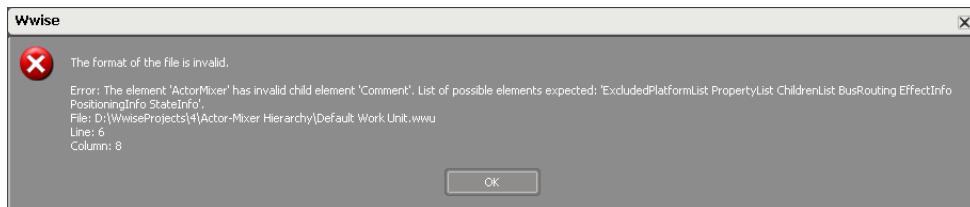
- ・ XMLシンタックスエラーやスキーマエラーの検証

- ・プロジェクト不整合の検証

検証することでエラーを解消し、更なるプロジェクト問題を防止することができます。

### XMLシンタックスエラー、スキーマエラー

ソースコントロールシステムでプロジェクトファイルをチェックインまたはマージする時に、コンフリクト解消のためにXMLコード自体の更新が必要となることがあります。更新中にXMLシンタックスエラーやスキーマエラーが発生した場合、プロジェクトファイルが無効となり、Wwiseでプロジェクトを開けなくなります。メッセージボックスが開き、エラーの内容とエラーの具体的なファイル名や場所が表示されます。

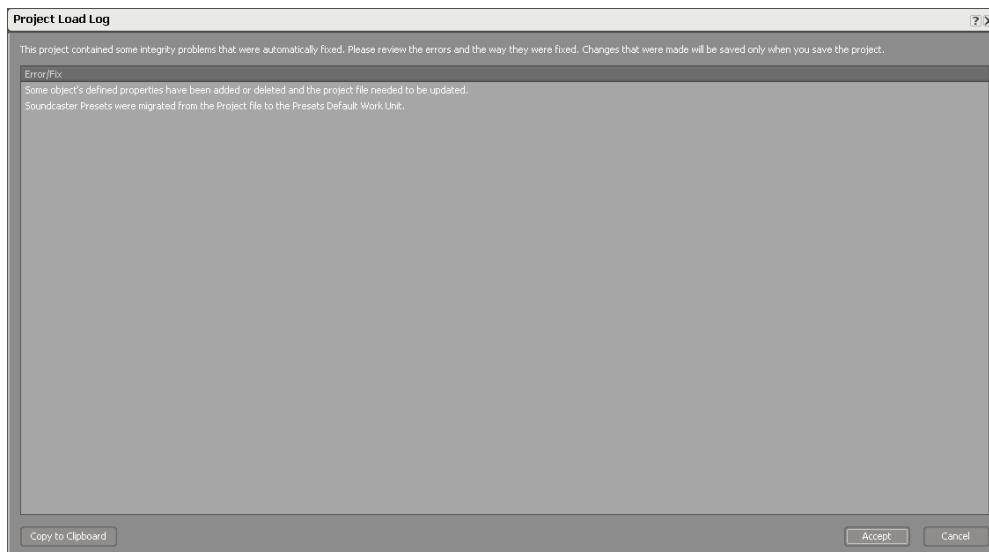


問題を解消するには、エラーに表示された各Work Unitファイルを見直す必要があります。

### プロジェクト不整合の問題

プロジェクトファイルにXMLシンタックスエラーがなければ、次にWwiseがプロジェクトに不整合や問題がないかを検証します。例えば、Work Unit Statesの中の1つのStateが削除されたにもかかわらず、サウンド構造やミュージック構造のWork Unitの中にあるオブジェクトがそのStateを使っている場合、プロジェクト不整合となります。

Wwiseがプロジェクト問題を検知すると、各問題の詳細と共に、それをフィックスする方法を必要に応じて表示します。



問題が検知された場合の対処方法として、フィックス案を全て受け入れるか、拒否してプロジェクトの旧バージョンに戻すか、拒否してXMLファイル内で不整合をフィックスする方法があります。

### プロジェクトの不整合を解消するには:

1. Wwiseで、プロジェクトを開く。

プロジェクトの不整合が検出された場合、Project Load Logが開き、プロジェクトの全ての問題のリストと共に、必要に応じてWwiseの提案するフィックス案が表示されます。

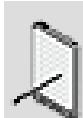
2. 以下のいずれかを行う:

- Acceptをクリックして、フィックス案を受け入れる。

Wwiseが、不整合をフィックスしてからプロジェクトを開く。フィックス案を受け入れる場合は1つずつ確認して、手作業によるフィックスが必要かどうかを判断する。

- Cancelをクリックして、フィックス案を受け入れない。

Wwiseがプロジェクトを閉じる。プロジェクトの旧バージョンに戻すか、不整合をXMLファイル内でフィックスする。



#### 注記

Wwiseが実行したプロジェクトのフィックスは、プロジェクトを保存するまで保存されません。

## Workgroupプラグインを使ったプロジェクトファイルの管理

Wwiseはオープンソースであるため、Workgroupプラグインを作成してソースコントロールソフトを簡単に実装できます。このWorkgroupプラグインがWwiseとソースコントロールソフトの間のリンクとなるので、Wwiseから直接ファイルを管理し、チェックインやチェックアウトなどのソースコントロール機能を実行できます。

ソースコントロールの操作はシステムによって異なるので、具体的な機能やワークフローはシステムによって違います。

Perforce®とSubversionという2種類のソフトウェア構成管理(SCM / Software Configuration Management)システムをWwiseすぐに使えるように、それぞれのWorkgroupプラグインのサンプルが同梱されています。PerforceとSubversionは、コピー・編集・マージ方式を採用し、ロック・編集・アンロック方式と異なります。この方式では、Workgroupの各メンバーがレポジトリまたはデポを読み込み、クライアントワークスペース、つまりファイルやプロジェクトの自分専用の作業用コピーを入れたディレクトリを、自分のワークステーション上に作成します。チームメンバーは各自の個人的コピーを編集するので、全員が同時に作業できます。準備ができた時点で、各メンバーが個人的コピーをマージさせ、新しいファイナルバージョンができます。マージ段階でコンフリクトが発生した場合はバージョンコントロールシステムが補助しますが、最終的に問題を正しく解消するのは、各メンバーの責任です。

他のソースコントロール管理システムを使用する場合、WwiseのWorkgroupプラグインを自分で作成できます。Workgroupプラグインの作成とインテグレーションについては、SDKドキュメンテーションの[How to Create a Source Control Plug-in DLL](#)を参照してください。

ソースコントロールプラグインを使うと、複数のソースコントロール機能をWwiseから直接、利用できます。例えば、プラグインPerforceを使うと、以下のソースコントロール操作をWwiseから実行できます。

- **Get latest version** デポから最新バージョン入手して、自分の作業コピーを更新。
- **Submit changes** ペンディング中のファイルを、処理するために、Perforceサーバに送信。
- **Check out** デポファイルの作業用コピーをクライアントワークスペースに作成。
- **Lock** 他のクライアントがサーバ上に、ファイルの作業用コピーをサブミットできないように、そのファイルをロック。
- **Unlock** 他のクライアントがサーバ上に、ファイルの作業用コピーをサブミットできるように、そのファイルをアンロック。
- **Mark for add** ファイルをデポに追加するために、ファイルをadd（追加）とマークする。ファイルがチェンジリストに追加されるので、デポにサブミットする必要がある。
- **Mark for delete** で、ファイルをデポから削除する。ファイルがチェンジリストに追加されるので、削除するためにデポにサブミットする必要がある。
- **Move** ディレクトリOriginalsにあるフォルダ間でソースファイルを移動する。
- **Rename** デポにあるファイル名の変更。
- **Revert changes** クライアントワークスペースで行ったファイルへの変更を取り消す。
- **Resolve** 1つのファイルの2つのバージョンの差分を調整。
- **Diff** クライアントワークスペースにあるファイルを、デポにあるファイルと比較。
- **File History** ファイルの変更履歴を表示。



複数のソースコントロールコマンドをWwiseからアクセスできるほか、Project Explorerに表示される特別なアイコンオーバーレイで、Work Unitファイルのステータスが簡単に確認できます。例として下表に、Perforce Workgroupプラグインで使用するアイコンオーバーレイを示します。

Overlay Icon	名前	内容
	Normal (not checked out)	ファイルが最新版であり、誰もチェックアウトしていない。
	Marked for add Moved	ファイルが、Perforceデポ追加のためにマークされたか、ステータスmove/addとして移動された。

Overlay Icon	名前	内容
	Checked out (Open for Edit)	ファイルが、Perforceデポからチェックアウトされている。
	Checked out by another user	ファイルが、Workgroupの他のユーザーによってチェックアウトされている。
	Concurrently checked out	Workgroupで自分以外にも、ファイルをチェックアウトしているユーザーがいる。
	Outdated (not latest revision of file)	ファイルの新しいバージョンがサーバにある。最新版が入手可能である。



### 注記

"Checked out by another user" "Concurrently Checked out" "Outdated" アイコンは、Perforce Workgroupプラグインだけにあります。

## Perforce/Subversionの対応バージョン

どちらのWorkgroupプラグインを採用する場合でも、使用するPerforceまたはSubversionのバージョンが、プラグインと互換性があることを、以下の情報で確認してください。

### Perforce

Perforce Workgroupプラグインは、Perforce SDK version 2015.1を使用して構築されていますが、Perforceサーバのほぼ全てのバージョンに対応するはずです。もし使用するPerforceのバージョンとWorkgroupプラグインの互換性がない場合は、Wwise SDKのソースを使ってそのバージョンに合うプラグインを再構築できます。Perforceプラグインについては、[Wwise SDKドキュメンテーション](#)を参照してください。

### Subversion

Subversion Workgroupプラグインは、Subversion version 1.8.5を使用して構築されています。もし使用するSubversionのバージョンとWorkgroupプラグインの互換性がない場合は、Wwise SDKのソースを使ってそのバージョンに合うプラグインを再構築できます。Subversionプラグインの再構築については、Wwise SDKドキュメンテーションを参照してください。



### 注記

Subversionのbinフォルダは、デフォルトでC:\Program Files\Subversion\binにあり、システムのPATH環境変数の中にある必要があります。通常はSubversionインストーラが、PATHに追加しますが、例えばSubversionを上記ZIPファイルからマニュアルインストールした場合は、binフォルダをマニュアルでPATHに追加する必要があります。



### Tip

Perforce、Subversion、その他のPlatform SDKアップデートは、弊社SDK Documentationの[What's New](#)に記載されます。

Workgroupプラグインを使ったアセット管理法を、以下のセクションで説明します。主にPerforce Workgroupプラグインについての記載ですが、Subversionの場合も同様のワークフローとなります。

- ・「Perforceで、プロジェクトファイルをセットアップ」
- ・「Perforceにファイルを追加」
- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

### Perforceで、プロジェクトファイルをセットアップ

Perforce Workgroupプラグインを利用する前に、ユーザーまたはシステム管理者が以下を行なう必要があります。

- ・ソースコントロール管理ソフトのクライアントバージョンを、Workgroupのメンバー全員のワークステーションにインストールしてセットアップする。
- ・クライアントワークスペースを、Workgroupのメンバー全員のワークステーションに設定する。
- ・プロジェクトのファイルの最新バージョンをデポから入手して、クライアントワークスペースに保存する。
- ・Wwiseで、Workgroupプラグインを選択して設定する。

Perforceのインストールや設定については、システム管理者に問い合わせるか、Perforceドキュメンテーションを参照してください。

Workgroupプラグインの選択や設定は、Project Settingsダイアログボックスで行います。WwiseのWorkgroupプラグインの選択や設定については「ソースコントロールプラグインの、コンフィギュレーション」を参照してください。

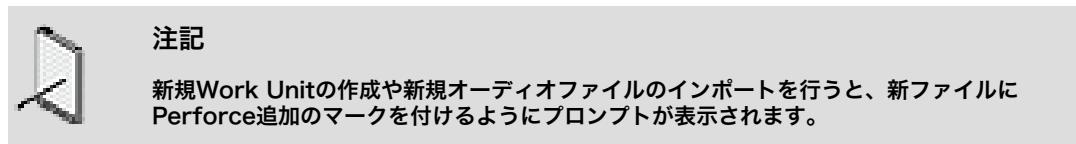
### 関連トピック

- ・「Perforceにファイルを追加」
- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

## Perforceにファイルを追加

プロジェクトで作業を始める前に、プロジェクトのファイルやオーディオソースを、デポまたはレポジトリに追加します。Perforceにファイルを追加するには、以下の2つの手順を行います。

- ・ チェンジリストにファイルを追加するには:。
- ・ 変更をデポにサブミットするには。



### チェンジリストにファイルを追加するには:

1. メニューバーで、Project > File Manager...をクリックする。  
File Managerが開く。
2. 以下のタブのいずれかに切り替える。
  - ・ Work Unitsタブは、PerforceにWork Unitを追加する場合。
  - ・ Sourcesタブは、Perforceにオーディオファイルを追加する場合。
3. Perforceに追加するファイル（単数または複数）を選択する。
4. 選択したファイルを右クリックして、メニューで、Mark for addを選択する。

Process Logダイアログボックスが開き、選択されたファイルがチェンジリストに追加され“opened for add”となり、デポにサブミットする準備ができることを表示する。

5. OKをクリックして、Process Logを閉じる。

ファイルのステータスが“add”になる。

6. Closeをクリックして、File Managerを閉じる。

Project Explorerで、プロジェクトファイルに、オーバーレイアイコンMarked for add（追加用にマーク）が付く。

この時点ではまだ、チェンジリストにファイルを追加しただけである。デポに新しいファイルをチェックインするには、チェンジリストをサブミットする必要がある。ファイルのデポへのサブミットについては「[Perforceデポに変更をサブミット](#)」を参照してください。

7. プロジェクトを保存する。

### 関連トピック

- ・ 「[Perforceデポのファイルの、チェックアウトと編集](#)」
- ・ 「[Perforceデポに変更をサブミット](#)」
- ・ 「[Perforceデポのファイルと比較](#)」
- ・ 「[Perforceで、Originalsフォルダ内のソースファイルの移動](#)」

- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

## Perforceデポのファイルの、チェックアウトと編集

プロジェクトのファイルを編集する前に、ファイルをチェックアウトしてファイルの最新バージョンを確実に自分のクライアントワークスペースに入れます。Perforceでファイルをチェックアウトすると、そのファイルは書き込み可能となります。つまり、ファイルへの変更を保存できます。

Perforceのデフォルトのチェックアウトは排他的チェックアウトでないため、同じファイルを使ってチームの他のメンバーが同時に作業をしているかもしれません。マージコンフリクトが起きると解消に時間と労力を要する場合があるので、まめに連絡を取り合い、問題発生を避けてください。

ファイルをチェックアウトして編集するには:

1. メニューバーで、Project > File Manager...をクリックする。

File Managerが開く。

2. 以下のタブのいずれかに切り替える。

- Work Unitsタブは、PerforceにWork Unitを追加する場合。
- Sourcesタブは、Perforceにオーディオファイルを追加する場合。

3. 開いて編集するファイル（単数または複数）を選択する。

4. 選択したファイルの1つを右クリックして、メニューで、Check outを選択する。

Process Logダイアログボックスが開き、そのファイルがチェックアウトされ“opened for edit”となったことを表示する。

5. OKをクリックして、Process Logを閉じる。

ファイルのステータスが“edit”になる。

6. Closeをクリックして、File Managerを閉じる。

Project Explorerで、プロジェクトファイルに、オーバーレイアイコンChecked out（チェックアウト中）が付く。

## 関連トピック

- ・「Perforceにファイルを追加」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

## Perforceデポに変更をサブミット

ファイルで作業が終了した場合や、変更点をチームの他のメンバーがアクセスできるようにするには、そのファイルをデポにサブミットします。自分が加えた変更の詳細情報を一緒に提出し、自他ともにファイル変更の範囲を完全に理解することが非常に重要です。このコメント提出は、コンフリクトに対応する際に役立ちます。

ファイルをサブミットしてPerforceのデポに戻すと、変更点がデポにあるファイルにマージされます。コンフリクトがある場合は、それを解消するまでそのファイルをデポにサブミットできません。

### 変更をデポにサブミットするには

1. メニューバーで、Project > File Manager...をクリックする。

File Managerが開く。

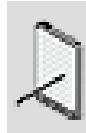
2. Submit Changesボタンをクリックしてから、以下のいずれかのオプションをクリックする。
  - Work Units - Work Unitだけをチェックインするためにサーバに送る場合。
  - Sources - ソースファイルだけをチェックインするためにサーバに送る場合。
  - All - ペンディング中の全ファイル（Work Unitやソースファイル）を、チェックインするためにサーバに送る場合。

Submit Changesダイアログボックスが開く。

3. サーバにチェックインしたくないファイルがある場合は、それらのファイルを選択から外し、OKをクリックする。
4. 選択したファイルの1つを右クリックして、メニューで、Submit Changesを選択する。

Descriptionダイアログボックスが開く。

5. ファイルに加えた変更の詳細を入力する。



### 注記

変更をサブミットする時に、Descriptionに日本語や中国語などの全角文字を使う場合は、P4システムの環境変数であるP4CHARSETをutf8に設定する。

6. 変更をサブミットした直後に引き続きファイル編集をしたい場合は、Check out file(s) after submit（サブミット後にファイルをチェックアウトする）オプションを選択する。
7. OKをクリックする。

Process Logダイアログボックスが開き、ファイルがチェックインされ“submitted”となったことを表示する。

8. OKをクリックして、Process Logを閉じる。

ファイルのステータスが“normal”になる。

9. Closeをクリックして、File Managerを閉じる。

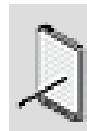
**Check out file(s) after submit**オプションを選択していない限り、Project Explorerで、プロジェクトファイルに、オーバーレイアイコンChecked in（チェックイン中）が付く。

### 個別ファイルをデポにサブミットするには:

1. メニューバーで、Project > File Manager...をクリックする。  
File Managerが開く。
2. 以下のタブのいずれかに切り替える。
  - Work Unitsタブは、PerforceにWork Unitを追加する場合。
  - Sourcesタブは、Perforceにオーディオファイルを追加する場合。
3. サブミットするファイル（単数または複数）を選択する。
4. 選択したファイルの1つを右クリックして、メニューで、Submit Changesを選択する。

Descriptionダイアログボックスが開く。

5. ファイルに加えた変更の詳細を入力する。



#### 注記

変更をサブミットする時に、Descriptionに日本語や中国語などの全角文字を使う場合は、P4システムの環境変数であるP4CHARSETをutf8に設定する。

6. 変更をサブミットした直後に引き続きファイル編集をしたい場合は、Check out file(s) after submit（サブミット後にファイルをチェックアウトする）オプションを選択する。
7. OKをクリックする。

Process Logダイアログボックスが開き、ファイルがチェックインされ“submitted”となったことを表示する。

8. OKをクリックして、Process Logを閉じる。

ファイルのステータスが“normal”になる。

9. Closeをクリックして、File Managerを閉じる。

**Check out file(s) after submit**オプションを選択していない限り、Project Explorerで、プロジェクトファイルに、オーバーレイアイコンChecked in（チェックイン中）が付きます。



#### Tip

Project ExplorerでWork Unitを右クリックして、Workgroup > Submit Changesを選択することで、Perforceデポに変更をサブミットすることもできます。

### 関連トピック

- 「Perforceにファイルを追加」

- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

### Perforceデポのファイルと比較

クライアントワークスペースにあるファイルを、デポにあるファイルの最新バージョンと比較 (Diff) して、両者の差分を確認できます。Diff操作でファイルを比較する前に自分のプロジェクトを保存して、差分確認にファイルの最新バージョンが使われるようになります。

ファイルを、Perforceのデポにあるファイルと比較するには:

1. メニューバーで、Project > File Manager...をクリックする。  
File Managerが開く。
2. 比較するWork Unitのファイルを右クリックして、メニューで、Diffを選択する。  
両ファイルの差分がPerforceのデフォルトDiffビューアに表示される。
3. 両ファイルの差分を確認する。
4. 差分の確認後、Perforceを閉じてWwiseに戻る。

### 関連トピック

- ・「Perforceにファイルを追加」
- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

### Perforceで、Originalsフォルダ内のソースファイルの移動

プロジェクトの、Originalsフォルダに入っているソースファイルを移動できますが、その前に他のチームメンバーがそれらのファイルを使用していないことを確認してください。他人がファイルをチェックアウト中にファイルを移動してしまうと、その人がファイル変更をデポにサブミットする時に問題が発生します。

Perforceでファイルを移動する場合、まず削除するために現在のファイルが開かれ、移動先にファイルの正確なコピーが作成されます。これらのファイルはデポで自動的に更新されないので、移動を完了してから、チェンジリストをデポにサブミットする必要があります。ファイルを移動する前に、ファイルをデポにチェックインする必要があります。



#### 注記

Originalsフォルダにあるソースファイルを移動する前に、プロジェクトを保存する必要があります。

ディレクトリOriginalsのフォルダ間で、ファイルを移動するには:

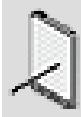
1. 現在のプロジェクトを保存する。
2. 以下のいずれかを行う:
  - メニューバーで、Project > File Manager...をクリックする。
  - [Shift]+[F1]を押す。

File Managerが開く。

3. Source Filesタブに切り替える。
4. 移動するファイル（単数または複数）を選択する。
5. 選択したファイルを右クリックして、メニューで、Moveを選択する。

Browse For Folderダイアログボックスが開く。

6. フォルダ層を移動して、ファイルの移動先となるフォルダを選択する。



#### 注記

新規フォルダを作成する場合は、親フォルダを選択して、Make New Folderボタンをクリックして、フォルダ名を指定します。

7. OKをクリックして、ファイルを新規フォルダに移動する。

Process Logダイアログボックスが開き、ファイルの新しいコピーがブランチコマンドを使って指定したフォルダに追加され、元ファイルが削除のため開かれた（Opened for delete）ことが表示される。

8. OKをクリックして、Process Logダイアログボックスを閉じる。
9. リストから、ブランチと削除の両エントリを選択する。
- 10選択したエントリを右クリックして、メニューで、Submit Changesを選択する。

Descriptionダイアログボックスが開く。

- 11内容を分かりやすく入力して、OKをクリックする。

Process Logが開き、ペンドィング中の両操作がデポにサブミットされたことが表示される。

- 12OKをクリックして、Process Logを閉じる。

これで、ファイルは指定したフォルダに属す。

## 関連トピック

- 「Perforceにファイルを追加」

- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

### Perforce使用中に、ファイル名を変更

プロジェクトのファイル名を変更できますが、その前に他のチームメンバーがそのファイルを使用していないことを確認してください。他人がファイルをチェックアウト中にファイル名を変更してしまうと、その人がファイル変更をデポにサブミットする時に問題が発生します。

Perforceでファイル名を変更する場合、まず削除するために現在のファイルが開かれた後、ファイルの正確なコピーが作成され、新しいファイル名が付けられます。これらのファイルはデポで自動的に更新されないので、ファイル名変更を完了してから、チェンジリストをデポにサブミットする必要があります。ファイル名を変更する前に、ファイルをデポにチェックインする必要があります。



### プロジェクトファイルの名前を変更するには:

1. メニューバーで、Project > File Manager...をクリックする。

File Managerが開く。

2. 変更するWork Unitファイル名を右クリックして、メニューで、Renameを選択する。

Rename Fileダイアログボックスが開く。

3. Nameフィールドに新しいファイル名を入力する。

4. OKをクリックする。

Process Logダイアログボックスが開き、ファイルのコピーが、ブランチコマンドを使って新しい名前で作成され、旧ファイルが削除のために開かれたことが表示される。

5. OKをクリックして、Process Logを閉じる。

旧ファイルのステータスが“delete”になり、ステータスが“branch”である新ファイルが作成される。

6. 新旧の両ファイルを選択する。

7. 右クリックして、メニューで、Submit Changesを選択する。

Descriptionダイアログボックスが開く。

8. 変更の内容を分かりやすく入力して、OKをクリックする。

Process Logが開き、元ファイルが削除され新ファイル名のファイルが追加されたことが表示される。

9. OKをクリックして、Process Logを閉じる。

10 Closeをクリックして、File Managerを閉じる。

Projectダイアログボックスが開き、プロジェクトに外部から変更が加えられたためプロジェクトの最新バージョンを再ロードするよう、プロンプトが表示される。

11 Yesをクリックする。

プロジェクトの最新バージョンがロードされ、ファイル名が変更されている。

## 関連トピック

- ・「Perforceにファイルを追加」
- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイルをプロジェクトから削除」
- ・「Perforce使用中に、プロジェクトを保存」

## Perforce使用中に、ファイルをプロジェクトから削除

プロジェクトで不要となったファイルは、削除できます。Perforceでファイルを削除する場合、まず削除するファイルを開き、次にチェンジリストをデポにサブミットします。ファイルを削除する前に必ずプロジェクトを保存して、前回の保存以降に加えた変更の紛失を防止してください。



### プロジェクトからファイルを削除するには:

1. メニューバーで、Project > File Manager...をクリックする。

File Managerが開く。

2. 削除するファイル（単数または複数）を選択する。

3. 選択したプロジェクトファイルの1つを右クリックして、メニューで、Mark for Delete を選択する。

ファイル削除の確認メッセージが表示される。

4. Yesをクリックする。

Process Logダイアログボックスが開き、ファイル（単数または複数）が削除のために開かれたことを示す。

5. OKをクリックして、Process Logを閉じる。

ファイルのステータスが“delete”になる。

6. Closeをクリックして、File Managerを閉じる。

Projectダイアログボックスが開き、プロジェクトに外部から変更が加えられたためプロジェクトの最新バージョンを再ロードするよう、プロンプトが表示される。

7. Yesをクリックする。

プロジェクトの最新バージョンがロードされ、ファイルが削除されている。

この時点ではまだ、チェンジリストにファイルを追加して削除するためにマークしただけである。デポからファイルを削除するには、チェンジリストをサブミットする必要がある。

## 関連トピック

- ・「Perforceにファイルを追加」
- ・「Perforceデポのファイルの、チェックアウトと編集」
- ・「Perforceデポに変更をサブミット」
- ・「Perforceデポのファイルと比較」
- ・「Perforceで、Originalsフォルダ内のソースファイルの移動」
- ・「Perforce使用中に、ファイル名を変更」
- ・「Perforce使用中に、プロジェクトを保存」

## Perforce使用中に、プロジェクトを保存

変更したいファイルを全てチェックアウトしてある限り、問題なくプロジェクトを保存できます。もしチェックアウトしていないファイルに変更を加えてしまった場合は、Wwiseのプロンプトが表示され、プロジェクト全体を保存する前にそのファイルをチェックアウトすることを求められます。

### プロジェクトを保存するには:

1. 以下のいずれかを行う:

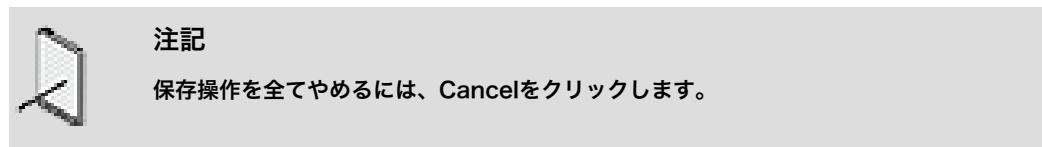
- ・メニューバーで、Project > Saveをクリックする。
- ・Ctrl+Sを押す。

Wwiseが、プロジェクトの全ファイルを保存しようとする。チェックアウトしていないファイルに変更を加えていた場合、ダイアログPending Source Control Operationsが開く。

2. Check out modified filesオプションを選択して、変更したファイルをチェックアウトして、Wwiseが保存できるようにする。

### 3. OKをクリックする。

必要なファイルをWwiseがチェックアウトして、プロジェクトを保存する。



## 関連トピック

- [「Perforceにファイルを追加」](#)
- [「Perforceデポのファイルの、チェックアウトと編集」](#)
- [「Perforceデポに変更をサブミット」](#)
- [「Perforceデポのファイルと比較」](#)
- [「Perforceで、Originalsフォルダ内のソースファイルの移動」](#)
- [「Perforce使用中に、ファイル名を変更」](#)
- [「Perforce使用中に、ファイルをプロジェクトから削除」](#)

## Workgroupに関する、コツとベストプラクティス

Wwiseでソースコントロールシステムを採用する前に、以下のセクションに目を通して、オーディオ開発プロセス全体を通してチーム作業やプロジェクトファイルを効率的に管理するための、コツやベストプラクティスを参照してください。

### プロジェクト計画の段階で

- プロジェクトを小さいWork Unitに分けます。プロジェクトの規模が大きくプロジェクトデータを全て複数のDefault Work Unitに入れてしまうと、Wwiseのレスポンスタイムが遅くなる上に1人が変更を加える度に他のチームメンバーもマージする必要があり、マージ問題やコンフリクトの危険性が大幅に増えてしまいます。Work Unitでプロジェクトを小規模なグループに分けることで、全員の作業効率が上がり、情報へのアクセスも速くなり、各自が別の部分で作業できるので、マージ問題の可能性が減少します。
- グローバル（全体にかかる）Work Unitの責任者を決める。Master-Mixer HierarchyやPresetなど一部のプロジェクトエレメントは、追加したWork Unitにさらに分割することはできません。このようなプロジェクトエレメントの変更を1人の担当者に限定するか、少なくとも全ての変更を把握している担当者を1人確保すると良いでしょう。

### 基本的なワークフロー

- プロジェクト全体に影響するグローバルなプロジェクトエレメントを、効率的に管理する。グローバルなプロジェクトエレメントであるStateやGame Parameterに対して名前変更や削除を行ってしまうと、プロジェクト内の他の多数のオブジェクト、例えばそのエレメントを使用するサウンドオブジェクトやコンテナなどが、全て変更されるかもしれませんので、注意してください。変更を保存してチェックインしてしまうと、他人が作業をしている多数のプロジェクトファイルにも影響するかもしれません。このような変更による影響範囲を限定す

るために、できるだけ早い段階でグローバルなプロジェクトエレメントを設定して、以後は変更しないように心がけてください。初期設定の後に変更する必要がある場合は、以下を行ってください。

- ・他のチームメンバーに、グローバルエレメントが変更されたことを警告する。
- ・チームメンバー全員に、自分の変更をチェックインしてもらう。
- ・自分のファイルをチェックインする。
- ・チームメンバー全員に、各自のプロジェクトファイルを更新してもらう。
- ・以上の処理を行えば、他のチームメンバーが更新するだけで新ファイルを入手でき、マージする必要がない。
- ・ファイルのステータスを確認する。Work Unitで作業を開始する前に、File Managerを見て、読み取り専用のファイルを確認してください。読み取り専用のWork Unitを変更しても、プロジェクト内でそのファイルを保存できません。
- ・ローカルファイルを定期的にバックアップする。中央レポジトリにあるファイルは定期的なバックアップ計画の対象かもしれません、ローカルマシンにあるコピーは対象外です。データ紛失を防ぐために定期的に自分のプロジェクトファイルをバックアップすると良い習慣であり、特にファイルに大幅な変更を加えた場合バックアップが大切です。
- ・チェックイン前にインテグレティレポ。Work Unitをチェックインする前にインテグレティレポートを生成して、プロジェクトエラーがないかを確認すると良いでしょう。エラーがあった場合は、解決してから、Work Unitをチェックインします。

### ファイルの同期

- ・新しいワークセッションを始める前に同期する。新しいワークセッションを開始する前に自分のプロジェクトファイルをサーバと同期し、最新の変更が手元にあることを確認してください。
- ・同期前にWwiseを閉じる。ファイルをサーバとシンクする前に、Wwiseを閉じて情報の紛失を防止してください。Wwiseが開いた状態だと、メモリにそのコピーが残ります。同期すると自分のディスク上のファイルが変更されますが、プロジェクトの自分の旧バージョンがメモリに残ります。現在開いているプロジェクトを保存する時に、ディスクに保存された他人による変更を上書きしてしまうかもしれません。

これはWorkgroupプラグインを使用しない場合に限ります。Workgroupプラグインを使用中に同期すると、プロジェクトの最新バージョンを再ロードするよう自動的にプロンプトが表示されます。

### ファイルのチェックイン

- ・サブミットを頻繁に行う。大きな変更を行い他のチームメンバーに影響する場合は、頻繁にファイルをサーバにサブミットして他のメンバーがその変更メリットを取り入れられるようにします。時間が開きすぎると、コンフリクトの確率も高くなります。小規模での的を絞った変更をサブミットすれば、必要に応じて簡単にプロジェクトの旧バージョンに戻すことができ、コンフリクト発生の際の解決も容易です。
- ・役に立つコメントを添付する。ファイルをコミットまたはチェックインする時に、どのような変更を加えたのかが明確に分かる説明を添付してください。

## ソースコントロールシステム

- ソースコントロールシステムの理解を深める。Wwiseプロジェクトのファイル管理にソースコントロールシステムを採用する前に、その機能を充分に理解しておくべきです。ソースコントロールシステムの細かい仕組みを知ることで、問題を回避して効率的で有力な制作パイプラインを構築できます。

## プロジェクトの不整合

- WwiseプロジェクトのXML構造の理解を深めるファイルをマージする前に、WwiseプロジェクトのファイルのXML構造を理解する時間を確保してください。場合によってXMLをアップデートする必要があります。理解不足はプロジェクトの崩壊を引き起こしかねません。もしXMLを編集した場合は、ファイルをソースコントロールシステムにチェックバックする前に、必ずプロジェクトをWwiseで開いてください。XMLの変更した部分が有効で予定通りの内容であることを、確認できます。

## Communication

- 頻繁でオープンなコミュニケーションを推進する。どのようなWorkgroup環境においても、他のチームメンバーとオープンな情報交換を頻繁に行なうことが重要です。オープンなコミュニケーションを頻繁に行なうことでコンフリクトが減り、ファイルマージの時間を短縮できるので、制作パイプラインの効率化につながります。例えば、他のチームメンバーに影響するようなWork Unit変更を実行する前に、各自の変更をチェックインして、連絡するまで作業の同期や再開を保留するように依頼します。

## 使用中ファイル

- File ManagerのUsage列で、Unusedとマークされたファイルを削除する前に、プロジェクトを一旦閉じて再度開き、この情報が最新であることを確認する習慣をつけると良いでしょう。

## SoundBank

- プロジェクトに変更がある度にSoundBankを編集しなくてもよいように、SoundBankのセットアップを、Work Unitやフォルダを使って再現します。WwiseがSoundBankのエレメントと、プロジェクトのエレメントの間のアクティブループを維持するので、これらのWork UnitをSoundBankに追加すれば、SoundBankが自動的に更新されるので、編集が一切不要となります。

---

## 第7章 メディアファイルの活用

概要 .....	135
インポート処理 .....	135
メディアファイルのインポート .....	143
メディアファイルの置換 .....	160
ファイルインポート関連の問題への対処 .....	162
Originalsフォルダのメディアファイルの並べ替え .....	163
キャッシュクリア .....	164
外部エディタでオーディオファイルを編集 .....	166
プラグインを使ってソースを作成する .....	167
メディアファイル管理に関する、コツとベストプラクティス .....	168

## 概要

典型的なゲームのアセット数は数千件にものぼり、プロジェクトのアセットを効率的に管理することが重要となります。Wwiseでは、以下の2種類のアセットを、根本的なレベルで区別します。

- Wwiseにインポートした、オリジナルのアセット
- 多様なプラットフォーム向けに作成した、アセットの各種バージョン

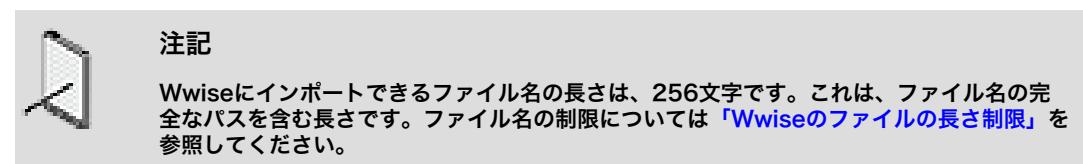
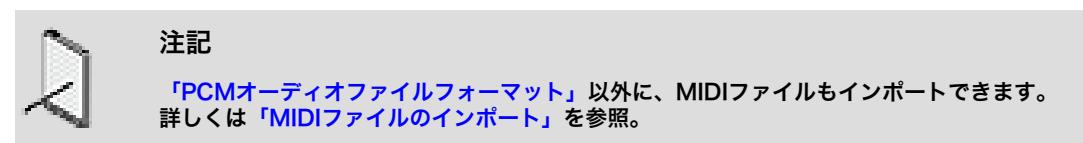
Wwiseでは、この2種類のアセットをプロジェクトフォルダの別の場所に保存することで、別々に管理します。オリジナルアセットは、プロジェクトのOriginals フォルダに保存されます。通常、インポートしたアセットはチーム内の数人がシェアするので、このOriginalsフォルダはネットワーク上のどこに置くこともでき、既に使用中のソースコントロールシステムで簡単に管理できます。

他のアセットバージョンはユーザーのプロジェクトのキャッシュフォルダでローカルに保存されます。このため、各ユーザーは自分のプラットフォームバージョンを自分で管理し、様々なコンバージョン設定の実験ができます。

ファイルは2種類とも、Wwise上で再生できます。コンバージョン前のオリジナルファイルは、Transport ControlやSoundcasterで、Original機能を有効にすると、再生できます。Original機能が有効でないと、コンバージョン済みのファイルがあれば、Wwiseはそれを再生しようとします。ただし、フォーマットによってコンバージョン後のファイルをWwiseが再生できないこともあるので注意してください。あるプラットフォーム向けにオーディオファイルをコンバージョンすると、そのプラットフォームの具体的なハードウェア要件に合わせてファイルが変換されます。つまり、Wwiseを実行しているプラットフォームがそのファイルタイプをサポートしない場合は、コンバージョン後のファイルをWwiseで再生できな可能性があります。

## インポート処理

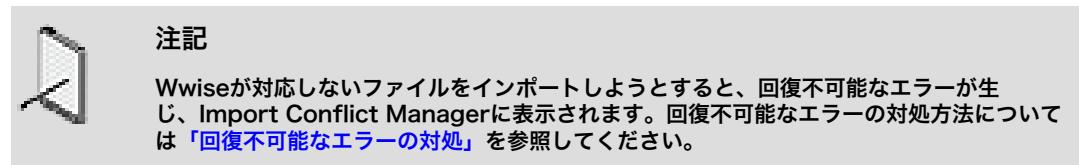
Wwiseのプロジェクトにファイルをインポートする前に、Wwiseが対応しているファイルの種類と、インポートしたメディアファイルをどこに保存するのかを、確認してください。



### 対応しているメディアファイルは？

Wwiseが対応しているメディアファイルの種類については、以下を確認してください。

- 「PCMオーディオファイルフォーマット」
- 「チャンネルコンフィギュレーション」
- 「サンプルレート」
- 「ビット深度」



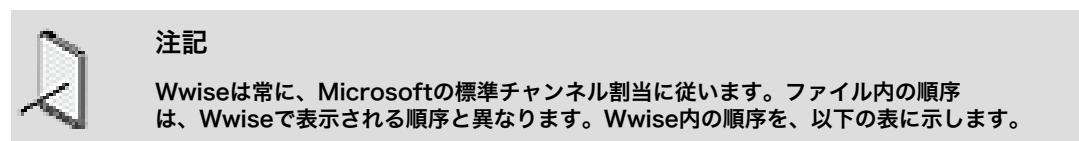
### PCMオーディオファイルフォーマット

オーディオフォーマットがWaveform (WAV) やAmbisonics (AMB) であるファイルをWwiseにインポートできます。Wwiseはファイルヘッダを読み、適切なチャンネルコンフィギュレーションを適用できます。

### チャンネルコンフィギュレーション

複数オーディオチャンネルのメディアファイルをWwiseにインポートできます。

下表は、Wwiseが対応している全てのソースチャンネルコンフィギュレーションと、そのコンフィギュレーションに使用可能なチャンネル、そして特記がない限りWwiseがデフォルトで使用するコンフィギュレーションを示したものです。



チャンネル名 - 略式名とフル名	推奨される角度
L Left	22° -30°
R Right	22° -30°
C Center	0°
SL Surround Left	90° -110°
SR Surround Right	90° -110°
BL Back Left	135° -150°
BR Back Right	135° -150°
LFE Low-Frequency Effects	N/A

チャンネル	Wwiseが使用するチャンネルコンフィギュレーション
0.1	LFE
1.0	C
1.1	C + LFE
2.0	L + R
2.1	L + R + LFE
3.0	L + R + C
3.1	L + R + C + LFE
4.0	L + R + SL + SR

チャンネル	Wwiseが使用するチャンネルコンフィギュレーション
4.1	L + R + SL + SR + LFE
5.0	L + R + C + SL + SR
5.1	L + R + C + SL + SR + LFE
6.0	L + R + BL + BR + SL + SR
6.1	L + R + BL + BR + SL + SR + LFE
7.0	L + R + C + BL + BR + SL + SR
7.1	L + R + C + BL + BR + SL + SR + LFE
7.1.2	L + R + C + BL + BR + SL + SR + HFL + HFR + LFE
7.1.4	L + R + C + BL + BR + SL + SR + HFL + HFR + HBL + HBR + LFE
9.1	L + R + C + SL + SR + HFL + HFR + HBL + HBR + LFE
10.1	L + R + C + SL + SR + T + HFL + HFR + HBL + HBR + LFE
11.1	L + R + C + SL + SR + T + HFL + HFC + HFR + HBL + HBR + LFE
13.1	L + R + C + BL + BR + SL + SR + T + HFL + HFC + HFR + HBL + HBR + LFE

Wwiseにインポートする際に具体的なマルチチャンネルコンフィギュレーションを維持したい場合は、WAVヘッダのフォーマットチャンクWAVEFORMATEXTENSIBLEで、定義する必要があります。なお、WAVEFORMATEXTENSIBLEの、チャンネルマスクで定義したチャンネルにWwiseが対応しない場合は、対応する値に自動的に変更されます。例えば、レフトチャンネルを定義しライトチャンネルを定義しないコンフィギュレーションの場合は、レフトチャンネル（0x01）がセンターチャンネル（0x04）に変更され、またサイドチャンネルを定義しリアチャンネルを定義しないコンフィギュレーションの場合は、サイドチャンネル（0x100、0x200）をリアチャンネル（0x10、0x20）に変更します。

なお、WwiseのアプリケーションであるMulti-Channel Creatorを使えば、モノオーディオファイルやステレオオーディオファイルを組み合わせて、1つのマルチチャンネルファイルにすることができます。この独立したユーティリティはWwiseパッケージの一部としてインストールされ、Wwiseアプリケーションと同じディレクトリにあります。Multi-Channel Creatorを使うと、どのようなマルチチャンネルコンフィギュレーションのファイルでも作成できるので、Wwiseプロジェクトにインポートできます。また、このツールで特定のモノファイルを0.1コンポーネントとしてフラグを立て、独立したLFEチャンネルとしてWwiseにインポートすることも可能です。

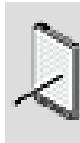
### サンプルレート

Wwiseにインポートできるメディアファイルの最大サンプルレートは、96kHzです。

### ビット深度

Wwiseにインポートできるメディアファイルのビット深度は、以下の通りです。

- 16 Bit
- 24 Bit



## 注記

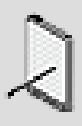
32ビットのフロートPCMファイルをWwiseにインポートできますが、最大コンバージョンフォーマットは24ビットです。またTransport Controlで再生する場合はファイルが16ビットにコンバージョンされます。

## インポート時に、Wwiseのオブジェクトを作成

インポート予定のファイルにWwiseが対応することを確認した後に、そのファイルでどのようなオブジェクトを作成するのかを決めます。

WwiseはSound SFXオブジェクト、Sound Voiceオブジェクト、Music Trackオブジェクトを区別します。これにより、例えば各種ランゲージに翻訳されるVoiceオブジェクトなどをWwiseで別々に扱えます。さらにContainerやMusic Segmentなど全体構造をつくるオブジェクトを使って、基礎となるオブジェクトをまとめたり整理したりできます。Wwiseの様々なオブジェクトタイプを見分けられるように、異なるアイコンで表示されます。

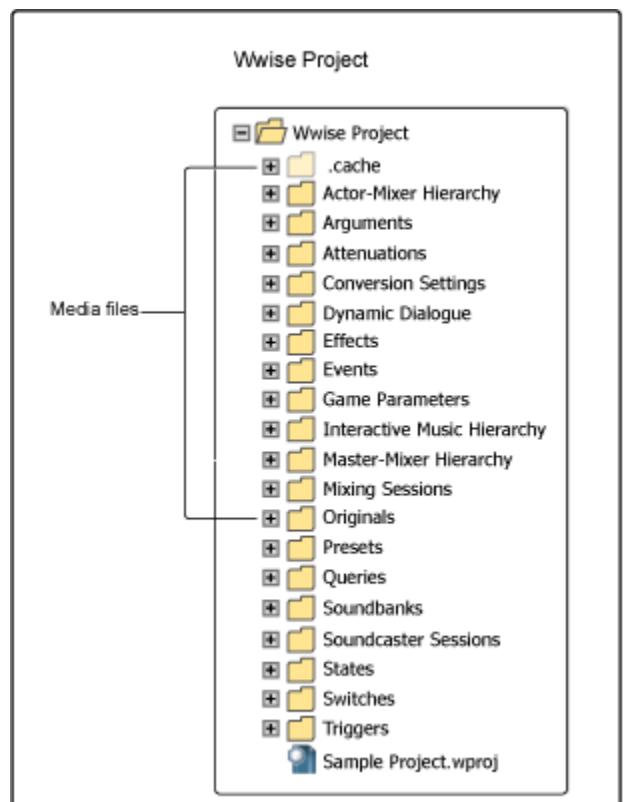
アイコン	内容
	SFX - Actor-Mixer Hierarchyにメディアファイルをインポートする時のデフォルトオブジェクトタイプ。
	Voice - ゲームの複数のランゲージバージョン用に翻訳される可能性の高いダイアログのために作成される。
	<p></p> <p><b>Tip</b></p> <p>Reference Language用にWAVファイルをActor-Mixer Hierarchyにドラッグする時に、Import As:リストの値をデフォルトのSound SFXではなく自動的にSound Voiceにするには、CtrlとShiftを押しながらドラッグします。</p>
	<p>Random Container - Actor-Mixer Hierarchyにフォルダをドラッグすると、デフォルトで作成される。ただしプロンプト表示されるAudio File ImporterのObject Type/Actionリストで別項目を選択すれば、フォルダを変更できる:</p> <ul style="list-style-type: none"> <li>• No object</li> <li>• Virtual Folder</li> <li>• Actor-Mixer</li> <li>• Sequence Container</li> <li>• Switch Container</li> <li>• Blend Container</li> <li>• Sound SFX</li> </ul>
	<p></p> <p><b>Tip</b></p> <p>Ctrlキーを押しながら、フォルダをProject ExplorerのActor-Mixer Hierarchyにドラッグすると、フォルダの全ての.wavファイル用に、サウンドSFXオブジェクトを作り、Random Containerを自動的に作成します。</p>
	Music Track - Interactive Music HierarchyのMusic Segmentの下にソースファイル (WAVまたはAMB) をドラッグすると作成される。
	<p></p> <p><b>Tip</b></p> <p>Music Trackオブジェクトを自動的に作成するには、Project ExplorerでCtrlキーを押しながらWAVファイルをInteractive Music HierarchyのMusic Segment上にドラッグします。</p>
	Music Segment - Interactive Music Hierarchyにフォルダやソースファイル (WAVまたはAMB) をドラッグすると、デフォルトで作成される。

アイコン	内容
	<p>ただしプロンプト表示されるAudio File Importerの<b>Object Type/Action</b>リストで別項目を選択すれば、フォルダを変更できる:</p> <ul style="list-style-type: none"> <li>• No object</li> <li>• Virtual Folder</li> <li>• Music Switch Container</li> <li>• Music Playlist Container</li> </ul> <div style="background-color: #f0f0f0; padding: 10px;">  <p><b>注記</b></p> <p>既存Music SegmentにWAVファイルをドラッグすると、ドラッグした全てのWAVファイルに対してMusic Trackが作成されます。</p> </div> <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;">  <p><b>Tip</b></p> <p>Music Segmentオブジェクトを自動的に作成するには、Ctrlキーを押しながら.wavファイルをProject ExplorerのInteractive Music Hierarchyにドラッグします。</p> </div>

## メディアファイルの構造

Wwiseプロジェクトのメディアファイルの保存方法を理解することは、チームで作業している場合は、特に役に立ちます。メディアファイルは、2つのフォルダに保存されます。

- Originals folder
- .cache フォルダ

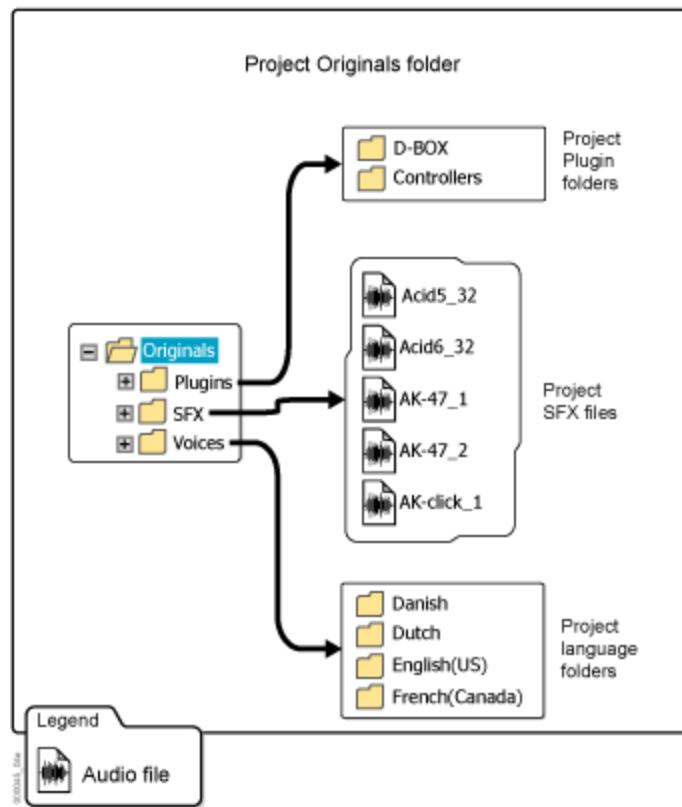


## Originalsフォルダ

Wwiseにメディアファイルをインポートすると、プロジェクトのOriginalsフォルダにコピーされ、変換や変更を受けない状態で残されます。一般的に、このフォルダはソースコントロールの下の、プロジェクト関係者全員がアクセスできる場所に保存されます。プロジェクトを作成する時に、Originalsフォルダの場所が決められますが、Project Settingsダイアログボックスで変更できます。このオリジナルファイルはTransport ControlやSoundcasterで**Original**を選択すると再生できます。

Originalsフォルダは、さらに以下のサブフォルダに分けられます。

- Plugins
- SFX
- Voices



Wwiseにインポートしたメディアファイルは全て、Wwiseで効率的に管理できるように、種類別に分けられます。

プロジェクトが大きい場合、これらのフォルダに入っているファイルをさらにサブフォルダに分けて整理することもできます。その際は、ファイルをインポートする前に、以下の制限があることに注意してください。

- サブフォルダに入れるメディアファイルをWwiseへインポートする時は、サブフォルダの中から行う必要があります。これによって、Wwiseはメディアファイル

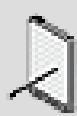
ルのフルパスとファイル名を使ってファイルを保存して、各メディアファイルの保存場所を管理します。

- ファイルパスは必ず、固有パスとします。つまり、あるフォルダ内にあるメディアファイルは、必ず全て異なるファイル名とします。ただし、保存先のフォルダが別であれば、2つのメディアファイルが同じファイル名を持つことも可能です。

既にプロジェクトにメディアファイルをインポートしてある場合、フォルダ間でのソースメディアファイルの移動も可能です。ソースメディアファイルの移動については「[Originalsフォルダのメディアファイルの並べ替え](#)」を参照してください。

### Cacheフォルダ

キャッシュフォルダはデフォルトでWwiseプロジェクト内に格納され、開発中のゲームが対応するプラットフォーム用に変換した、各プラットフォーム向けメディアファイルが入れられます。変換されたこれらのファイルは、WwiseのTransport Controlや、Soundcasterで、Originalボタンが選択されていない時に再生されます。



#### 注記

キャッシュフォルダの場所を、Project Settingsダイアログボックスで変更できます。プロジェクトのキャッシュフォルダの場所の変更については「[Cacheフォルダの設定](#)」を参照してください。

ただし、フォーマットによってコンバージョン後のファイルをWwiseが再生できないこともあるので注意してください。あるプラットフォーム向けにオーディオファイルをコンバージョンすると、そのプラットフォームの具体的なハードウェア要件に合わせてファイルが変換されます。つまり、Wwiseを実行しているプラットフォームがそのファイルタイプをサポートしない場合は、コンバージョン後のファイルをWwiseで再生できな可能性があります。

コンバージョン後のファイルをWwiseでテストするには、以下を行えます:

- PCM - Windowsプラットフォームを選択する。
- ADPCM - Windowsプラットフォームを選択する。
- Vorbis - Windowsプラットフォームを選択する。



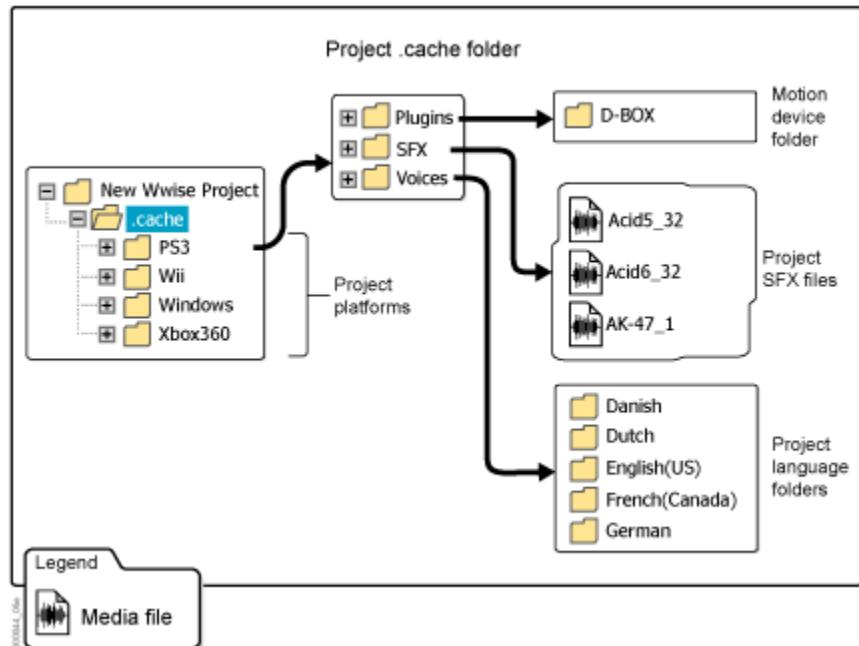
#### 注意

ソースコントロールシステムにこのフォルダを含めてはいけない。

全てのプラットフォームフォルダは、さらに以下のサブフォルダに分けられます。

- Plugins
- SFX
- Voices

Wwiseがユーザーに代わってこれらのフォルダの中身を管理するので、これらのフォルダ内を直接、変更を加える必要はありません。



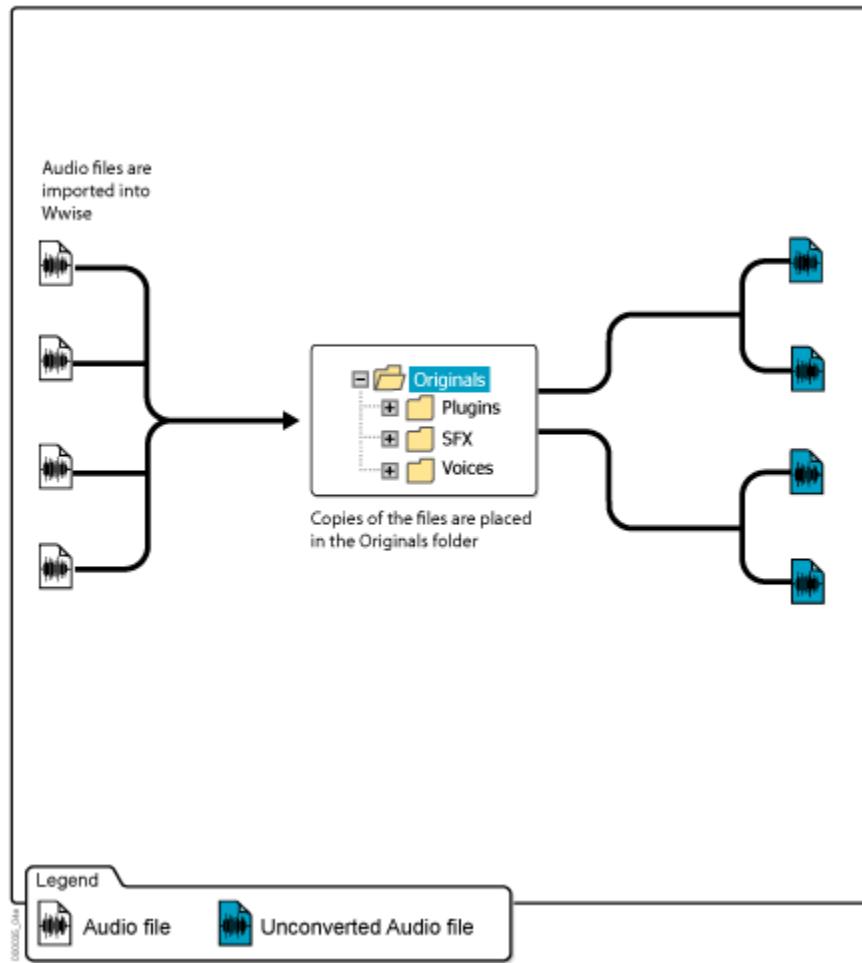
## インポート処理の内容

インポート処理中に、Wwiseで複数の操作が実行されます。なお、メディアファイルのコンバージョンは、別途行われます。プラットフォーム別のメディアファイルのコンバージョンについては「[オーディオファイルをコンバージョンする](#)」を参照してください。

インポート処理で、以下の操作が行われます。

- Wwiseでオリジナルのメディアファイルを検証し、プロジェクトのOriginals フォルダにコピーする。
- メディアファイルのオーディオソースが作成される。
- ソースを含むサウンドオブジェクトまたはミュージックオブジェクトがWwiseで作成され、Project Explorerの、Audioタブの該当する階層内に表示される。

Wwiseプロジェクトにメディアファイルをインポートした時の処理を、下図に示します。



## メディアファイルのインポート

WwiseにSFXやボイスのメディアファイルをインポートするタイミングや理由は、そのメディアファイルの使用目的によって異なります。一般的に、以下の段階などでファイルをインポートします。

- メディアファイルを、プロジェクトの最初に、またはファイルが準備された時に、プロジェクトに取り込む時。
- 既にインポートしたメディアファイルを置き換える時。例えば、プロジェクト初期にプレースホルダや仮のファイルを用いた場合に、それを置き換える時。
- ローカリゼーション用に、プロジェクトにランゲージ別メディアファイルを取り込む時。ローカリゼーションについては「[プロジェクトのローカリゼーション作業](#)」を参照してください。

Wwiseでは、**Audio File Importer**を使ってメディアファイルをプロジェクトにインポートするか、Quick Importショートカットを使ってオーディオファイルをインポートします。

## Quick Import

Wwiseにメディアファイルをインポートするためのショートカットを、下表に示します。

To	ショートカット
SFXメディアファイルのインポート	Wwiseにファイルをドラッグする。
ボイスメディアファイルのインポート	Shiftを押しながら、Wwiseにファイルをドラッグする。
Audio File Importerを開けずに、SFXメディアファイルのインポート	Ctrlを押しながら、Wwiseにファイルをドラッグする。
Audio File Importerを開かずに、Import Sound Voiceメディアファイルをインポートする。Importer	Ctrl+Shiftを押しながら、Wwiseにファイルをドラッグする。 このショートカットでランゲージをインポートする場合は、必ずLanguage Selectorでレファレンスランゲージが選択されていることを確認すること。
新しいソースとして、Contents Editorへのメディアファイルのインポート	Contents Editorにファイルをドラッグする。



### Tip

プロジェクトに数千件のボイスアセットがあり表計算ソフトで管理している場合は、対応するSound VoiceオブジェクトをVoice Asset Importerを使ってWwiseで簡単に作成できます。Wwiseにボイスアセットをインポートする方法については「[テキストファイルからボイスアセットをインポート](#)」を参照してください。

## Wwiseのファイルの長さ制限

Wwiseにインポートできるファイルのファイル名の長さは、最大256文字です。このファイル名の長さ制限はWindows側の制限で、ファイル名のパス全体の長さが対象です。例えば、ディレクトリのルートにあるファイル名は、最大252文字まで可能です。一方、フォルダ内にあるファイルの場合は、フォルダ名の長さも考慮する必要があります。ファイル名の長さ制限は、Wwiseでメディアファイルのインポートやコンバージョンを行う際に問題となることがあります。

制限一杯のファイル名、またはそれに近いファイル名のメディアファイルをインポートすると、インポート問題が発生することがあります。ファイル名が長すぎると、以下のエラーの1つが表示されます。

- Not a valid wav file. (有効な.wavファイルでない)
- Error copying the file to the originals folder. (ファイルをOriginalsフォルダにコピーする際にエラーが発生)

制限一杯またはそれに近いファイルのインポートを、Wwiseが許可することができますが、コンバージョンを行う際に問題となることがあります。この場合は、以下のエラーが表示されます。

- Can't open source or output file. (ソースまたは出力ファイルを開けない)

このような問題が起きた場合は、ファイル名を短くするか、ファイル名パス全体の長さを短くする必要があります。

## SFX用のメディアファイルのインポート

Audio File Importerで、メディアファイルをSFXオブジェクトとして指定してインポートすることが可能です。ファイルをインポートするとProject ExplorerのAudioタブで選択した場所にSFXオブジェクトが作成されます。どのHierarchyにインポートするかによって、オブジェクトはサウンドオブジェクトまたはミュージックオブジェクトとなります。

### SFXメディアファイルをインポートするには:

1. 以下のいずれかを行う:
  - Wwiseのメニューで、Project > Import Audio Filesをクリックする。
  - Actor-Mixer HierarchyまたはInteractive Music Hierarchyでオブジェクトを右クリックして、メニューで、Import Audio Filesを選択する。

Audio File Importerが開く。

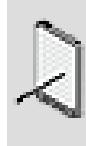
2. Add FilesまたはAdd Foldersをクリックする。

Open Fileダイアログボックスが開く。

3. インポートするメディアファイルの場所まで移動する。
4. ファイルまたはフォルダを選択して、OpenまたはSelect Folderをクリックする。

選択したファイルまたはフォルダが、Audio File Importerにロードされる。

5. メディアファイルをサウンドオブジェクトとしてインポートするには、Import as Sound SFXオプションを選択する。
6. オブジェクトを作成するWwise内の場所を変更するには、[...]をクリックして、階層内の新しい場所まで移動する。

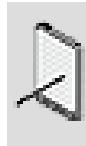


#### 注記

メディアファイルをミュージックオブジェクトとしてインポートするには、そのミュージックオブジェクトを作成したいInteractive Music Hierarchy内の場所まで移動する。この時、Import as SFX and Voiceオプションは、選択できない。

7. Importをクリックする。

Importingダイアログボックスが開き、ファイルインポートプロセスの進行を確認できる。



#### 注記

メディアファイルにエラーやコンフリクトがある場合は、Import Conflict Managerが開く。コンフリクトの対処方法については「[ファイルインポート関連の問題への対処](#)」を参照。



#### 注記

ソースコントロールを使用していれば、インポートしたファイルをソースコントロールシステムに追加するよう、プロンプトが表示される。

## 関連トピック

- 「ボイスオーバー用のメディアファイルのインポート」
- 「Wwiseのファイルの長さ制限」
- 「Quick Import」
- 「メディアファイルの置換」
- 「ファイルインポート関連の問題への対処」
- 「Originalsフォルダのメディアファイルの並べ替え」
- 「キャッシュクリア」
- 「外部エディタでオーディオファイルを編集」

## メディアファイルの入ったフォルダのインポート

メディアファイルが入っているフォルダ構造（単数または複数）をプロジェクトに直接インポートすることも可能です。フォルダ構造をインポートすると、以下が起きます。

- オプションで指定すれば、インポートする構造の中の各フォルダに対応するコンテナオブジェクトを、Wwise側で作成する。
- メディアファイルをOriginalフォルダにコピーする時は、Wwise側でフォルダ構造をコピーする。

## プロジェクトにフォルダ構造をインポートするには:

### 1. 以下のいずれかを行う:

- Wwiseのメニューで、Project > Import Audio Filesをクリックする。
- Project Explorer階層にあるオブジェクトを右クリックして、メニューで、Import Audio Filesを選択する。

Audio File Importerが開く。

### 2. Add Foldersをクリックする。

Select Folderダイアログボックスが開く。

### 3. インポートするメディアファイルの場所まで移動する。

### 4. フォルダを選択して、Select Folderをクリックする。

選択したフォルダが、Audio File Importerにロードされる。

### 5. Object Type/Action列から、インポートするフォルダごとに、作成するオブジェクトタイプを選択する。



#### Tip

Audio File Importerで、複数のフォルダ項目を選択してObject Type/Actionを選択すると、全項目を同時に変更できます。

### 6. Importをクリックする。

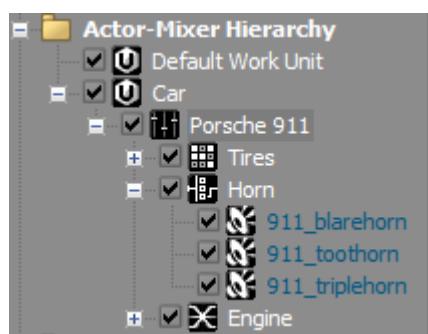
Importingダイアログボックスが開き、ファイルインポートプロセスの進行を確認できる。

## テンプレートを使用したメディアファイルのインポート

フォルダからメディアファイルをインポートする際、インポートするフォルダやメディアファイルで新規Wwiseオブジェクトを作成するために、Audio File Importerがテンプレートとして使用するオブジェクトを、プロジェクト内で指定しておくことができます。これはメディアファイルだけが異なる、複数の類似する構造を作成する時に便利です。

典型的な例が、互いに構造が似ているものの中のメディアファイルが異なる、様々なメーカー・型の車の構造です。ワークフローを以下に示します。

1. プロジェクトに複数の構造をインポートする際にテンプレートとして使用する、最初の構造を作成する。この構造は、コンテナやサウンドオブジェクトを組み合わせた集合体となる。



この構造でよければ、次の車のテンプレートとして使える。

2. 今Wwiseで作成した車構造に合わせて、同様のフォルダ構造をコンピュータ上に作成して自分のメディアファイルをここに入れる。



これで、インポートしやすい構造の準備が整う。

3. ProjectメニューまたはコンテキストメニューでImport Audio Files...をクリックする。デフォルトショートカットはShift+I。

Audio File Importerが開く。

4. Add Folders...をクリックする。

ダイアログExplorerが開く。

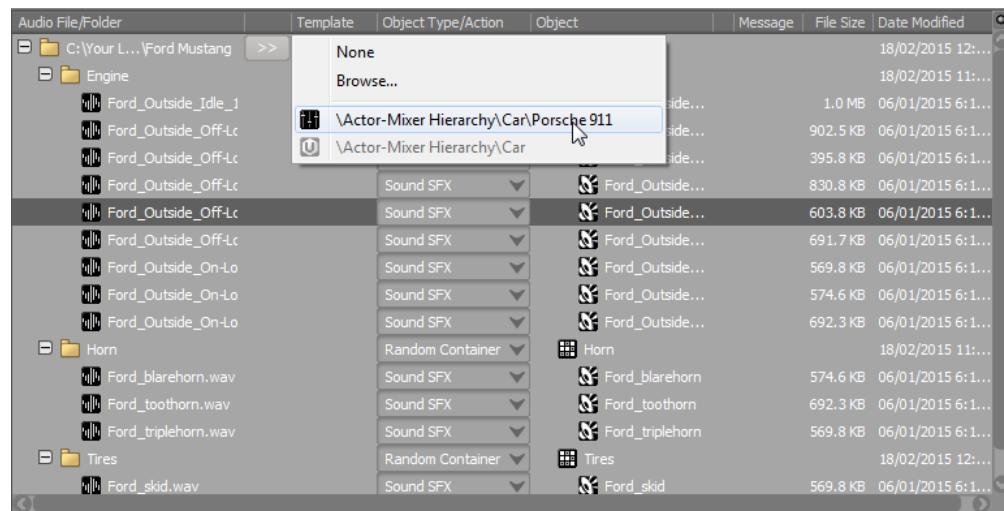
5. インポートするフォルダ（先ほど作成した車構造の親フォルダ）まで移動してクリックする。

選択したフォルダの名前が Folderフィールドに表示される。

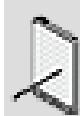
6. Select Folderをクリックする。

**Audio File Importer**が再び前面表示となり、ローカルフォルダ階層が**Audio File/Folder**パネルに表示され、この構造内にある全てのメディアファイルも表示される。

- Template列の左にあるセレクタ[>>]をクリックして、最初のステップで作成した構造まで移動する。これがテンプレートとなる。なお、[>>]ボタンで、最近テンプレートとして使用したオブジェクトのリストも表示される。



この時点では、インポートするメディアファイルやフォルダと、テンプレート中のオブジェクトを、Wwiseが自動的にマッチさせようとする。マッチするフォルダはテンプレート側で対応するフォルダと同じ種類のコンテナに設定され、マッチしないフォルダはデフォルトでRandom Containerに設定される。



#### 注記

インポートする際のマッチ度合を変更するには、**Audio File Importer**の左下にある**Template match mode**リストから選択します。Match allモードは特殊なアルゴリズムを使いインポートしたファイルに一番近いテンプレートオブジェクトを見つけるのに対し、Perfect match onlyモードはインポートしたファイルやフォルダとテンプレートオブジェクトの名前が完全に一致する場合のみマッチさせます。インポートしたものとテンプレートのマッチがない場合は、**Message**列にNo matching template foundと表示されます。



#### Tip

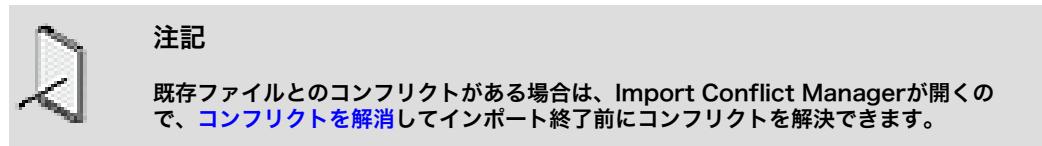
テンプレートマッチのアルゴリズムは、名前の比較に基づいています。このためベストプラクティスとして、フォルダ階層の名前をテンプレート階層の名前と似たものにします。



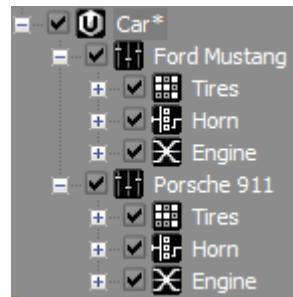
#### 注記

テンプレートとインポートした構造が上手くマッチしない場合、[>>]ボタンをクリックしてから、Noneを選択して、テンプレートを完全に削除できる。また、Object Type/Action列で別のオブジェクトタイプを選択すれば、テンプレートとの関係を解除できる。

- Importをクリックする。



Audio File Importerが閉じProject ExplorerのHierarchy内に新しい構造が表示される。



### ボイスオーバー用のメディアファイルのインポート

ダイアログやボイスオーバー（ナレーションなど）のメディアファイルをインポートする時に、このファイルをボイスオブジェクトとして指定して、Wwiseがローカリゼーション段階でこれらのオブジェクトを認識できるようにします。

プロジェクトのローカリゼーション用のボイスファイルをインポートすることも考えられます。ローカリゼーション用ファイルをインポートすると、プロジェクトのランゲージフォルダに保存され、その言語用の新しいソースがWwiseで作成されます。ただし、事前にLanguage Managerを使って、プロジェクトで使用するランゲージを設定しておく必要があります。プロジェクトランゲージを設定してから、ランゲージソース用のメディアファイルをインポートします。プロジェクトのランゲージ作業については「[プロジェクトのローカリゼーション作業](#)」を参照してください。



メディアファイルをボイスオブジェクトとして、プロジェクトにインポートするには:

1. 以下のいずれかを行う:

- Wwiseのメニューで、Project > Import Audio Filesをクリックする。
- Actor-Mixer HierarchyのサウンドボイスまたはWork Unitを右クリックして、メニューでImport Audio Filesを選択する。

Audio File Importerが開く。

2. Add FilesまたはAdd Foldersをクリックする。

Open Fileダイアログボックスが開く。

3. インポートするメディアファイルの場所まで移動する。

4. これらのファイルを選択して、Openをクリックする。

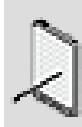
選択したファイルが、Audio File Importerにロードされる。

5. Import as Sound Voiceオプションを選択する。

6. オブジェクトを作成するWwise内の場所を変更するには、 [...]をクリックして、階層内の新しい場所まで移動する。

7. Importをクリックする。

Importingダイアログボックスが開き、ファイルインポート処理の進行を確認できる。



#### 注記

メディアファイルにエラーやコンフリクトがある場合は、Import Conflict Managerが開きます。コンフリクトの対処方法については「[ファイルインポート関連の問題への対処](#)」を参照。

## 関連トピック

- [5章 ランゲージの管理](#)
- [「SFX用のメディアファイルのインポート」](#)
- [「Wwiseのファイルの長さ制限」](#)
- [「サウンドボイスのメディアファイルの置換」](#)
- [「SFXファイルの置換」](#)
- [「ファイルインポート関連の問題への対処」](#)
- [「Quick Import」](#)
- [「Originalsフォルダのメディアファイルの並べ替え」](#)
- [「キャッシュクリア」](#)
- [「外部エディタでオーディオファイルを編集」](#)

## タブ区切りText FileからMedia Fileをインポート

タブ区切り (tab-delimited、別名tab-separated) テキストファイルから、多数のメディアファイルをインポートすることができます。タブ区切りテキストファイル (拡張子TXTまたはTSV) は、Microsoft Excelなどの表計算アプリケーションで生成できます。またANSI、UTF-8、UCS-2でエンコーディングしたファイルをインポートできます。



#### 重いタブ区切りのインポートに時間がかかることも

何万個というメディアファイルをインポートしようとすると、それなりの時間がかかるかもしないことに、注意してください。できるだけ早く終わらせるには、ファイルの大がかりなインポートはSSDドライブ付きのマシンで行い、ディスクに負荷がかかるようなアプリ

ケーション、例えばソースコントロール同期、クラウド同期、また場合によってはウィルス対策サービスなども、事前に終了させておきます。

タブ区切りファイルで設定できるエレメントを以下に示します：

- WAV、AMB、MIDIファイルをインポート
- ファイルを保存するオブジェクト構造
- Voice VolumeやOutput Busなどの、プロパティ値やレファレンス
- 作成するイベント

タブ区切りテキストファイルの最初の行に、その下に続く行の各列の内容を定義します。列の順番は、インポート処理に影響しません。

次の表は、タブで区切られた可能性のあるテキストファイルの列見出しをすべて一覧表示して説明しています。



### Absolute Path または オブジェクトのGUIDを取得する

**Absolute path** と **GUID** はタブ区切りファイルのオブジェクトを指定するために使われます。オブジェクトの絶対パスやGUIDを取得するには、Shiftキーを押しながらオブジェクトを右クリックしてコンテキストメニューを表示します。Copy path(s) to clipboardまたはCopy GUID(s) to clipboardを選択します。

Header	Content
Audio File	<p>インポートするWAVファイルまたはMIDIファイルのパスを定義するこのファイルが既に <b>Originals</b> フォルダになければ、コピーされる。パスは、絶対パス、または相対パスとする。</p> <ul style="list-style-type: none"> <li>• <b>Absolute</b>：オーディオファイルネーム自体を含むフルパス 例: <i>C:\MyWaves\MyFolder\Originals\MySound.wav</i></li> <li>• <b>Relative</b>：タブ区切りファイルのディレクトリへの相対パス。 例: <i>..\..\MyWaves\MyFolder\Originals\MySound.wav</i></li> </ul> <p>コンテナを作成する場合は、この列は空白のままでよい。</p>
Audio File	<p>base64でエンコードされた、インポートするWAVオーディオファイルデータを定義する。ターゲットファイルパスを指定して、base64でエンコードしたデータと、垂直線 (   ) で分ける必要がある。ターゲットファイルのパスは、<b>Originals</b> フォルダに対する相対パスとして、インポートタイプフォルダの下で、抽出される。</p> <p>例: <i>MySounds/MySound.wav/</i> <i>Uk1GRuRcAABXQVZFZm10IBAA...</i></p>
Object Path	<p>作成するオブジェクトのパスと名前を定義する。パスにバックスラッシュを使用する。パスは、絶対パス、または相対パスとする。</p> <ul style="list-style-type: none"> <li>• <b>絶対</b>: バックスラッシュとオブジェクトカテゴリから始めるフルパス。 例: <i>\Actor-Mixer Hierarchy\MyWorkUnit\MyVirtualFolder\MySound</i></li> </ul> <div style="background-color: #e0e0e0; padding: 5px;"> <b>注記</b> 絶対パスで指定された Work Unit が存在しない場合、インポートは失敗します。     </div> <ul style="list-style-type: none"> <li>• <b>相対</b>: Audio File Importerダイアログで指定したインポートデストイネーションに対する相対パス。</li> </ul>

Header	Content		
	<p>例: <code>MyVirtualFolder\MySound</code></p> <p>この列が空白の場合は、オブジェクト名として、インポートするオーディオファイルの名前が付けられ、Audio File Importerダイアログで指定したデスティネーションにインポートされる。オブジェクトの名前の前に、<code>&lt;&gt;</code> の間に書き込むことによって、パス内の違うオブジェクトのタイプを指定することができます。タイプが指定されない場合、インポートプロセスが仮想フォルダにデフォルト設定されます。可能なオブジェクトタイプの一覧は、以下で説明するオブジェクトタイプ列と同じです。</p> <p>この列が空白の場合は、オブジェクト名として、インポートするオーディオファイルの名前が付けられ、Audio File Importerダイアログで指定したデスティネーションにインポートされる。</p> <p><b>Object Path「オブジェクトパス」例:# \&lt;Actor-Mixer&gt;Barrel\&lt;Sound SFX&gt;Minigun_Barrel_Stop</b></p>		
Object Type	<p>オブジェクトタイプを指定する。一般的なオブジェクトタイプ:</p> <ul style="list-style-type: none"> <li>• Virtual Folder</li> <li>• Actor-Mixer Hierarchy オブジェクト: <ul style="list-style-type: none"> <li>• Actor-Mixer</li> <li>• Random Container</li> <li>• Sequence Container</li> <li>• Switch Container</li> <li>• Blend Container</li> <li>• Sound SFX</li> <li>• Sound Voice</li> </ul> </li> <li>• Interactive Music Hierarchy オブジェクト: <ul style="list-style-type: none"> <li>• Music Switch Container</li> <li>• Music Playlist Container</li> <li>• Music Segment</li> <li>• Music Track</li> </ul> </li> </ul> <p>この列が空白の場合は、オブジェクトパス内で指定しない限り、Object Type が Virtual Folder であるとみなされます。</p> <div data-bbox="760 1284 838 1543" data-label="Image"> </div> <div data-bbox="899 1305 948 1330" data-label="Section-Header"> <h4>注記</h4> </div> <div data-bbox="899 1343 1408 1529" data-label="Text"> <p>Audio File Importerを使う場合、1回のインポート処理で Actor-Mixer HierarchyのオブジェクトとInteractive Music Hierarchyのオブジェクトを同時に作成することはできません。1回のインポート処理で Sound SFX オブジェクトと Sound Voice オブジェクトを同時に作成することはできない。ただし、Wwise Authoring APIと -TabDelimitedImport コマンドラインの方法を使って、Actor-Mixer Hierarchyオブジェクトや Interactive Music Hierarchyオブジェクトをインポートすることは可能です。</p> </div> <tr> <td data-bbox="352 1543 752 1911">           Property[PropertyName]            または、  <code>@PropertyName</code>  <b>例:</b>  <ul style="list-style-type: none"> <li>• <code>Property[Voice Volume]</code></li> <li>• <code>Property[Volume]</code></li> <li>• <code>@Voice Volume</code></li> <li>• <code>@Volume</code></li> </ul> </td><td data-bbox="752 1543 1434 1911"> <p>列のヘッダに、設定するプロパティを指定する。プロパティの名前として、RPCタブ、Multi Editor、List Viewなどで表示されるものを使う。</p> <p>使用可能なプロパティについては、<a href="#">Wwise Objects Reference</a>を参照してください。</p> <p><b>一般的なプロパティ名:</b></p> <ul style="list-style-type: none"> <li>• Volume</li> <li>• Pitch</li> <li>• Lowpass</li> </ul> </td></tr>	Property[PropertyName] または、 <code>@PropertyName</code> <b>例:</b> <ul style="list-style-type: none"> <li>• <code>Property[Voice Volume]</code></li> <li>• <code>Property[Volume]</code></li> <li>• <code>@Voice Volume</code></li> <li>• <code>@Volume</code></li> </ul>	<p>列のヘッダに、設定するプロパティを指定する。プロパティの名前として、RPCタブ、Multi Editor、List Viewなどで表示されるものを使う。</p> <p>使用可能なプロパティについては、<a href="#">Wwise Objects Reference</a>を参照してください。</p> <p><b>一般的なプロパティ名:</b></p> <ul style="list-style-type: none"> <li>• Volume</li> <li>• Pitch</li> <li>• Lowpass</li> </ul>
Property[PropertyName] または、 <code>@PropertyName</code> <b>例:</b> <ul style="list-style-type: none"> <li>• <code>Property[Voice Volume]</code></li> <li>• <code>Property[Volume]</code></li> <li>• <code>@Voice Volume</code></li> <li>• <code>@Volume</code></li> </ul>	<p>列のヘッダに、設定するプロパティを指定する。プロパティの名前として、RPCタブ、Multi Editor、List Viewなどで表示されるものを使う。</p> <p>使用可能なプロパティについては、<a href="#">Wwise Objects Reference</a>を参照してください。</p> <p><b>一般的なプロパティ名:</b></p> <ul style="list-style-type: none"> <li>• Volume</li> <li>• Pitch</li> <li>• Lowpass</li> </ul>		

Header	Content
	<ul style="list-style-type: none"> <li>• Highpass</li> </ul> <p><b>一般的なプロパティ表示名:</b></p> <ul style="list-style-type: none"> <li>• Voice Volume</li> <li>• Voice Pitch</li> <li>• Voice Low-pass Filter</li> <li>• Voice High-pass Filter</li> </ul> <p>各行に、作成されるオブジェクトに設定するプロパティ値を指定する。</p> <p>列挙値は、列挙名に対する整数値を指定する必要があり、これは、.wwuファイルやWObjects.xmlにある。</p> <p>この列が空白の場合は、数値が設定されない。</p>
Reference[ReferenceName] または、 @ReferenceName または、 ~ReferenceName (deprecated)	<p>列のヘッダに、設定するレファレンスを指定する。レファレンスネーム（レファレンス名）とは、レファレンス名か、List ViewやMulti-Editorに表示される表示名の、どちらかです。</p> <p>使用可能なレファレンスについては、<a href="#">Wwise Objects Reference</a>を参照してください。</p> <p><b>一般的なレファレンス表示名:</b></p> <ul style="list-style-type: none"> <li>• Output Bus</li> <li>• User Auxiliary Send 0</li> <li>• User Auxiliary Send 1</li> <li>• User Auxiliary Send 2</li> <li>• User Auxiliary Send 3</li> <li>• Conversion Settings</li> <li>• Effect 0</li> <li>• Effect 1</li> <li>• Effect 2</li> <li>• Effect 3</li> <li>• Attenuation</li> <li>• MIDI Target</li> </ul> <p>各行に、以下のいずれかを指定する。</p> <ul style="list-style-type: none"> <li>• <b>絶対パス:</b> プロジェクト内のオブジェクトへの絶対パス。</li> </ul> <p>例: \Master-Mixer Hierarchy\Default Work Unit \Master Audio Bus\Sub Bus</p> <ul style="list-style-type: none"> <li>• <b>GUID:</b> オブジェクトのGUID。WWUファイル内にあるか、オブジェクトのショートカットメニューを開いてShiftキーを押してオーサリングツール内で直接見つけることができます。</li> </ul> <p>例: {10EC26E3-03EB-4470-8B30-4344B6F90438}</p> <p>この列が空白の場合は、レファレンスが設定されない。</p>
Notes	作成するオブジェクトの備考やコメントを設定する。
Audio Source Notes	作成するオブジェクトに関連付けられたオーディオソースファイルのメモやコメントを定義する。
Event	<p>インポートするオブジェクト用に作成されるイベントの名前とパスを定義する。</p> <p>パスは、絶対パス、または相対パスとする：</p>

Header	Content
	<ul style="list-style-type: none"> <li>絶対パス: \Events\ で始め、Work Unit名を続ける、Eventへの絶対パス。 例: \Events\MyWorkUnit\PlayEvent</li> <li>相対パス: EventのDefault Work Unitへの相対パス。 例: MyVirtualFolder\PlayEvent</li> </ul> <p>Actionタイプを指定するには、イベント名の最後に @ActionName を付ける。指定のない場合は、PlayActionが使われる。</p> <p>Action Type 「アクションタイプ」 の例：</p> <ul style="list-style-type: none"> <li>MyEvent@Play</li> <li>MyEvent@Stop</li> <li>MyEvent@Pause</li> <li>MyEvent@Resume</li> <li>MyEvent@Break</li> <li>MyEvent@Seek</li> </ul> <p>パスエレメントがまだデスティネーションに存在しない場合は、Virtual Folderが作成される。</p> <p>指定したイベントが既にデスティネーションにある場合は、既存イベントに新しいEvent Actionが追加される。</p> <div style="background-color: #f0f0f0; padding: 5px;">  <b>Tip</b>            1つのオブジェクトに対して複数のイベントを作成するには、複数のEvent列を追加します。         </div>
Dialogue Event 「ダイアログイベント」	<p>インポートするオブジェクト用に作成されるDialogue Eventの名前とパスを定義する。パスは、絶対パス、または相対パスとする：</p> <ul style="list-style-type: none"> <li>絶対パス: \Dynamic Dialogues\ で始め、Work Unit名を続ける、Dialogue Eventへの絶対パス。 例: \Dynamic Dialogues\MyWorkUnit\DialogueEvent</li> <li>相対パス: Dialogue EventのDefault Work Unitへの相対パス。 例: MyVirtualFolder\DialogueEvent</li> </ul> <p>Dialogue Event 列の主な目的は、Dialogue Event Editorで行われるよう、Dialogue Event パスを作成したり修正することです。パスを指定するには、Dialogue Event名の最後に @Path: を付け、続けて「.」(ピリオド) で連結したSwitch構造またはState構造を付けます。</p> <p>パスのProbabilityとWeightの値(0から100)を指定することができます。そうする場合には、/ を追加して Probability:### あるいは Weight:### を入力して下さい。詳しくは「<a href="#">パスの ProbabilityとWeightを設定する</a>」を参照。パスを指定しない場合、ProbabilityやWeight引数の使用は何もしません。</p> <div style="background-color: #f0f0f0; padding: 5px;">  <b>Tip</b>            1つのオブジェクトに対して複数のイベントを作成するには、複数のEvent列を追加します。         </div> <p><b>Dialogue Event 「ダイアログイベント」 例:# virtualFolder \Unit_Under_Attack@Path:&lt;Switch Group:MyVirtFolder \FS_Type&gt;Running.&lt;State Group:Hostile&gt;Gang.&lt;Switch</b></p>

Header	Content
	<p><i>Group:VirtualFolder\Location&gt;Alley/Probability:80/ Weight:20</i></p>
Switch Assigntion	<p>Switch Container に関連付けられる Switch または State Group を Actor-Mixer Hierarchy の中だけで定義してください。また、Switch Container の子がどの関連グループから Switch または State に割り当たられるかを定義してください。</p> <p>&lt;&gt; の間で Switch や State Group を設定するには、グループタイプとグループ名を、コロンで区切って指定する。アサインするには、&gt; の後に子を指定できます。詳細は以下の例を参照。</p> <div style="background-color: #f0f0f0; padding: 10px;">  <p><b>Tip</b> パス内の Switch や State Group の名前の前に Virtual Folder 階層を指定することができます。</p> </div> <div style="background-color: #f0f0f0; padding: 10px;">  <p><b>Tip</b> Switch Container に関連付けるグループの名前の前に Virtual Folder 階層を指定することができます。</p> </div> <div style="background-color: #f0f0f0; padding: 10px;">  <p><b>Tip</b> 一つのオブジェクトに対して複数の Switch 割り当てを作成する場合、Switch Assigntion 列を複数回、追加してください。</p> </div> <p><b>Switch Assigntion 例:</b># &lt;Switch Group:YourSwitchGroupName&gt;YourSwitchName , &lt;Switch Group:SS_FS_Type&gt;Jump</p>

### タブ区切りファイルをインポートするには：

1. 以下の操作でオーディオファイルインポーターを開きます:
  - Project Explorerでデスティネーション（インポート先）を選択し、ショートカットメニューで、Import Audio Filesを選択する。（Shift+I）
  - プロジェクトメニューで Import Audio Files...を選択する
 Audio File Importerが開く。
2. Import Tab Delimited...ボタンをクリックする。
3. タブ区切りファイルを選択して、OKをクリックする。
4. Audio File Importerリストでインポート内容を確認する。
5. Importをクリックする。

### タブ区切りの例

次の表は（まだ違いがありますが、Sample Projectから詳細を引用）タブ区切りワークシートの例です。確かにすべて網羅しているわけではありませんが、Object Type 列と Object Path で指定されたオブジェクトタイプ、絶対パスと相対パスなど、同じものをインポートするいくつかの異なった方法による複数の可能なインポートを示します。

**1つのパス内で複数オブジェクトの1ライン作成：**わずか1行を使って、パス構造全体が複数のコンテナで作成できる方法を示しています。Object Path で指定され

たObject Typesのそれぞれは、Object Type列のエントリで別々の行に入力されている可能性があります。

Audio File	Object Path	Property [Voice Pitch]	Reference [Output Bus]	Event
C:\My Audio Files\ Minigun\ MnGn_Barrel_Down.wav	\Actor-Mixer Hierarchy\Default Work Unit\<Actor-Mixer>Minigun\<Blend Container>Barrel\<Sound SFX>Minigun_Barrel_Stop	50	\Master-Mixer Hierarchy\Default Work Unit\Master Audio Bus\Environmental SFX\Guns	\Events\Minigun\Pay_Minigun@Play

**Switch Group と Default SwitchがアサインされたSwitch の作成:** Switchを追加して、次にそれにSwitch GroupとSwitch Valueをアサインする方法を示しています。

Object Path	Object Type	Switch Assigntation
\Actor-Mixer Hierarchy\My Default Work Unit\<Actor-Mixer>Minigun\<Switch Container>Shells\A	Sound SFX	<Switch Group:SS_Surface>Dirt
\Actor-Mixer Hierarchy\My Default Work Unit\<Actor-Mixer>Minigun\<Switch Container>Shells\B	Sound SFX	<Switch Group:SS_Surface>Wood_Solid

**Dialogue EventsでSound Voices の追加 :** Sound Voicesが、関連するDialogue Event やパスで追加される方法を示しています。

Audio File	Object Path	Object Type	Dialogue Event 「ダイアログイベント」
..\..\My Voice Recordings\English(US)\UNA-GG-AL_01.wav	\Actor-Mixer Hierarchy\Dialogues\Captain_A\UNA-GG-AL_01	Sound Voice	\Dynamic Dialogue\Default Work Unit\Unit_Under_Attack@Path:<State Group:Unit>Unit_A.<Switch Group:FS_Type>Running.<State Group:Hostile>Gang.<State Group:Location>* Probability:80 Weight:20
..\..\My Voice Recordings\English(US)\UNA-HS-AL_01.wav	\Actor-Mixer Hierarchy\Dialogues\Captain_A\UNA-HS-AL_01	Sound Voice	Unit_Under_Attack@Path:<State Group:Unit>Unit_A.<Switch Group:FS_Type>Running.<State Group:Hostile>DefuseBomb.<State Group:Location>Alley Probability:20 Weight:20
..\..\My Voice Recordings\English(US)\UNA-GG-HG_01.wav	[D2F87A0D-7D5C-429DB13D-03478450C2A2]	Sound Voice	Unit_Under_Attack@Path:<State Group:Unit>Unit_A.<Switch Group:FS_Type>Running.<State Group:Hostile>Gang.<State Group:Location>Hangar

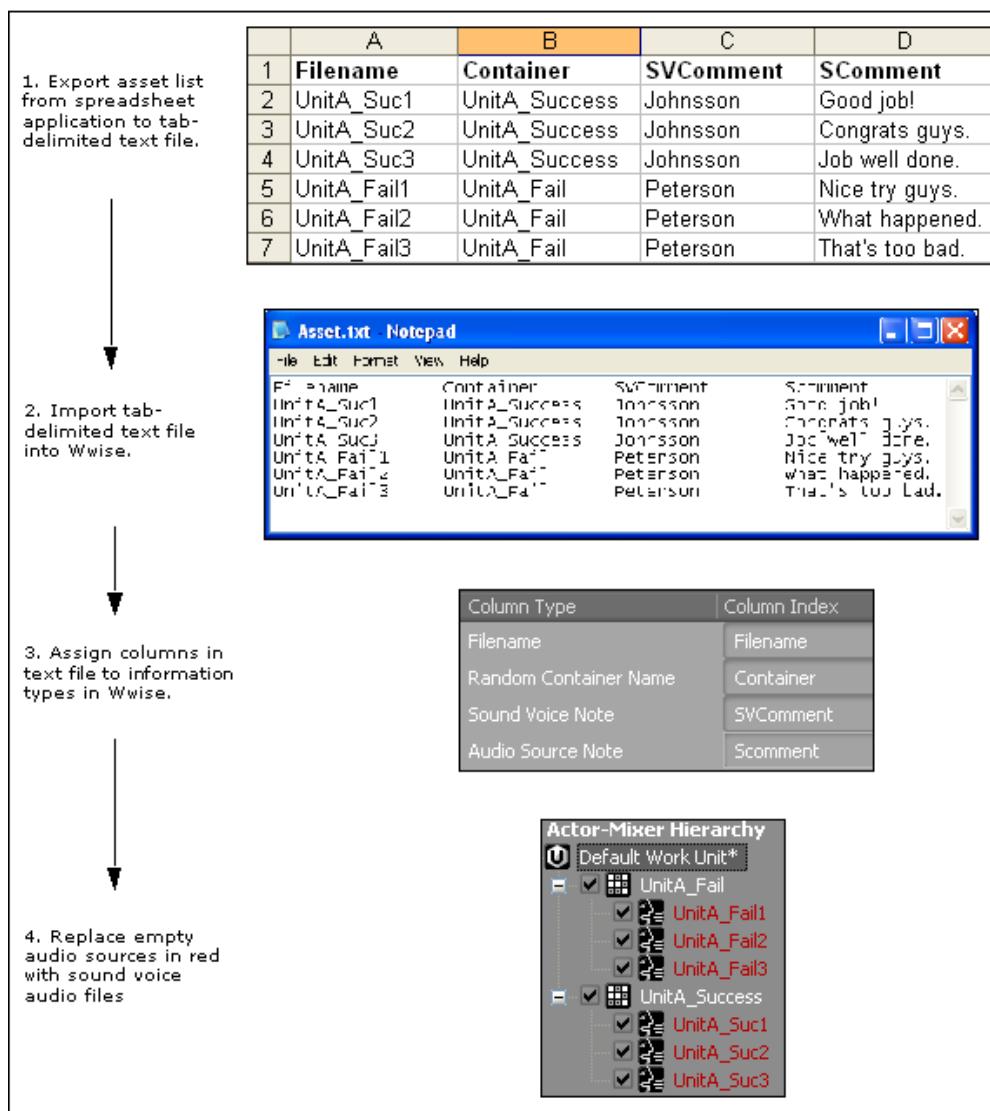
### テキストファイルからボイスアセットをインポート

ゲームのボイスアセットが数千件にものぼることが珍しくない中、Wwiseでボイスアセットを効率的に作成することが重要となります。一般的に、これらのアセットはMicrosoft Excelなど外部プログラムで管理されるため、特定の情報をタブ区切りテキストファイルからインポートしてWwiseプロジェクトで再現できるよう

になっています。これで制作プロセスが速くなるだけでなく、エラー率も減らせます。



下図は、表計算ソフトのボイスアセットリストをWwiseに取り込むワークフローを示したものです。

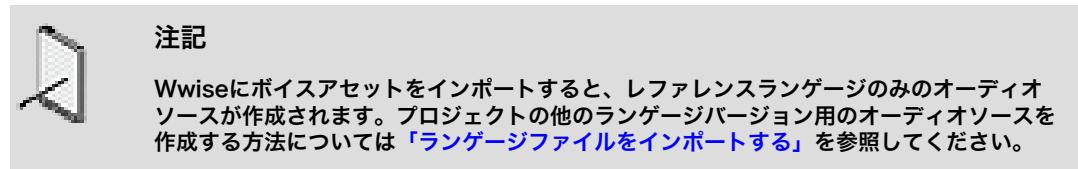


テキストファイルにアセットに関する様々な情報が記載されているかもしれません。ファイルのインポート時にWwiseで使用する情報は、以下の4種類だけです。

- **Filenames** - ファイル名であり、サウンドボイスオブジェクトと、そのオーディオソースを作成するために使う。最初はオーディオソースに何も入っていないが、同名のオーディオファイルへのレファレンスが含まれる。つまり、他

のオーディオファイルと同様の方法で、置換できる。Sound Voiceオブジェクトのメディアファイルの置換については「[サウンドボイスのメディアファイルの置換](#)」を参照。

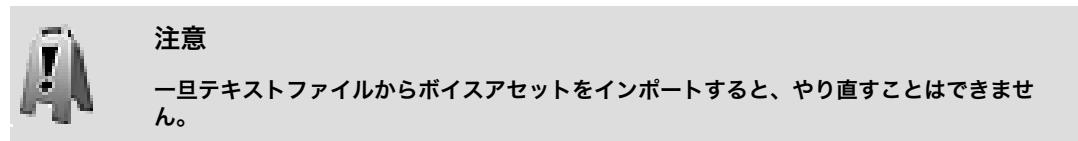
- **Random container names** - Random Container名であり、テキストファイルにこれが含まれる場合、そのサウンドボイスオブジェクトの親Random Containerを作成する時に使う。
- **Sound voice notes** - サウンドボイスのメモであり、サウンドボイスオブジェクトのNotesフィールドに追加される情報。例えば、このダイアログを話すキャラクター名など。
- **Audio source notes** - オーディオソースのメモであり、オーディオソースのNotesフィールドに追加される情報。例えば、実際に使われる台詞など。



テキストファイルの中の情報の順番や長さは重要でなく、そのファイルをインポートする前に、テキストファイルのどの列をインポートするのかを事前に指定します。Filenameだけが必須の情報であり、Sound Voiceオブジェクトやオーディオソースを作成するために使われます。前述の4種類の情報にマッピングされる情報以外の列がテキストファイルにある場合は、Wwiseで無視されます。

各列が正しく割り当てられることを確認するために、作成されるオブジェクトやインポートされる情報のプレビューが表示されます。

Voice Assets Preview			
Sound Voice	Random Container	Sound Note	Source Note
UnitA_Suc1	UnitA_Success	Johnsson	Good job!
UnitA_Suc2	UnitA_Success	Johnsson	Congrats guys.
UnitA_Suc3	UnitA_Success	Johnsson	Job well done.
UnitA_Fail1	UnitA_Fail	Peterson	Nice try guys.
...	...	...	...



テキストファイルからボイスアセットをインポートするには:

1. ProjectメニューでImport Voice Assetsを選択する。

Voice Asset Importerが開く。

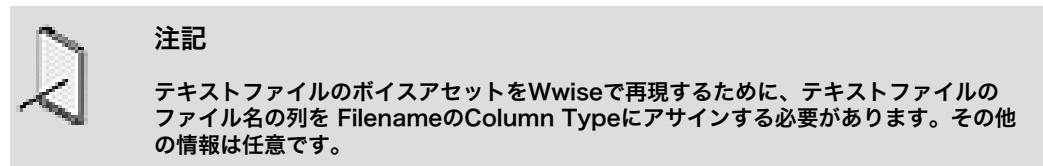
2. File to importフィールド横のBrowseボタン (...) をクリックする。

Openダイアログボックスが開く。

3. インポートするテキストファイルを選択して、Openをクリックする。

ファイルの場所が**File to import**フィールドに表示される。

4. もしテキストファイルにヘッダ行があり、これを列の割振りに使用する場合は、チェックボックス**Use header**を選択する。
5. **Header row**スピナーで、テキストファイルのどの行にヘッダ情報があるかを指定する。
6. **Start import at row**スピナーで、Wwiseがボイスアセット情報を読み始める行を指定する。
7. **Column Index**リストで、以下のColumn Typeに該当するテキストファイル中の列を選択する：

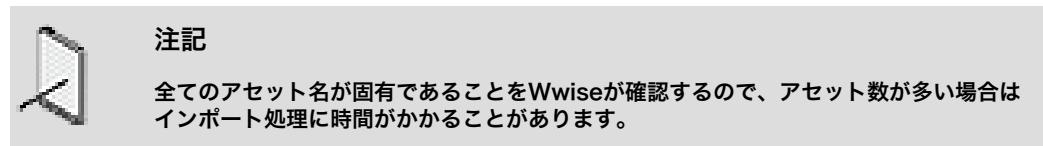


- **Filename**
  - **Random Container Name**
  - **Sound Voice Note**
  - **Audio Source Note**
8. 表Voice Assets Previewの情報を確認して、Wwiseが求める情報の種類に合わせてテキストファイルの行が正しく割振られているかを、確認する。
  9. 新しいSound Voiceアセットを作成する場所を変更するには、以下を行う：
    - Import destinationテキストボックス横の**Browse...**ボタンをクリックする。
    - Project Explorer - Browserで、プロジェクト階層の中の新しい場所を選択する。
    - OKをクリックする。

**Import destination**テキストフィールドに新しいパスが表示される。

10 Importをクリックする。

Sound VoiceオブジェクトがWwise内に作成され、対応するオーディオソースや、該当する場合はRandom Containerも同時に作成される。インポート処理中に問題が発生した場合は、WwiseのImport Completedダイアログボックスに表示される。



## 関連トピック

- 「サウンドボイスのメディアファイルの置換」
- 「ボイスオーバー用のメディアファイルのインポート」
- 「SFX用のメディアファイルのインポート」

- ・「ファイルインポート関連の問題への対処」
- ・「ランゲージファイルをインポートする」

## メディアファイルの置換

プレースホルダとして仮のファイルを使用する場合があります。このような時は、インポートしてあるメディアファイルを置換する必要があります。置換するには、Audio File Importer、またはショートカット機能Quick Replaceを使用します。

### Quick Replace

既にインポートしてあるメディアファイルの置換に使えるショートカットを、下表に示します。

To...	Use this shortcut:
既存SFXファイルを置換して、新規サウンドオブジェクトを作成するには。	ファイルをドラッグ＆ドロップする。
いSFXファイルを置換するには。	Altを押しながら、ファイルをドラッグする。
既存ボイスファイルを置換して、新規サウンドオブジェクトを作成するには。	Shiftを押しながら、ファイルをドラッグする。
ボイスファイルを置換するには。	Alt+Shiftを押しながら、ファイルをドラッグする。 このショートカットでランゲージを置換する場合は、必ずLanguage Selectorでレファレンスランゲージが選択されていることを確認する。

### SFXファイルの置換

Audio File Importerを使い、プロジェクトに既にあるSFXファイルを置換できます。

#### 既存SFXメディアファイルを置換するには:

1. 以下のいずれかを行う:

- ・Wwiseのメニューで、Project > Import Audio Filesをクリックする。
- ・アクターミキサー階層にあるオブジェクトを右クリックして、メニューで、Import Audio Filesを選択する。

Audio File Importerが開く。

2. AddまたはAdd Foldersをクリックする。

Open Fileダイアログボックスが開く。

3. インポートするメディアファイルの場所まで移動する。

4. これらのファイルを選択して、Openをクリックする。

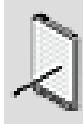
選択したファイルが、Audio File Importerにロードされる。

5. グループボックスImport Modeで、Replace audio filesを選択する。

6. グループボックスObject Typeで、Import as Sound SFXを選択する。

## 7. Importをクリックする。

Importingダイアログボックスが開き、ファイルインポート処理の進行を確認できる。



### 注記

メディアファイルにエラーやコンフリクトがある場合は、Import Conflict Managerが開きます。コンフリクトの対処方法については「[ファイルインポート関連の問題への対処](#)」を参照。

## 関連トピック

- 「[メディアファイルのインポート](#)」
- 「[サウンドボイスのメディアファイルの置換](#)」
- 「[ファイルインポート関連の問題への対処](#)」
- 「[Originalsフォルダのメディアファイルの並べ替え](#)」
- 「[キャッシュクリア](#)」
- 「[外部エディタでオーディオファイルを編集](#)」

## サウンドボイスのメディアファイルの置換

Audio File Importerの、Replace audio filesモードを使って、プロジェクトにある既存のボイスメディアファイルを置き換えることができます。このモードにはコンバージョン設定のオプションがありません。プロジェクトのローカリゼーションを行う場合は、Localize Languagesモードを使用する必要があります。ローカリゼーションについては「[プロジェクトのローカリゼーション作業](#)」を参照してください。

## プロジェクトのボイスメディアファイルを置換するには:

### 1. 以下のいずれかを行う:

- Wwiseのメニューで、Project > Import Audio Filesをクリックする。
- Actor-Mixer HierarchyのサウンドボイスまたはWork Unitを右クリックして、メニューでImport Audio Filesを選択する。

Audio File Importerが開く。

### 2. Addをクリックする。

Openダイアログボックスが開く。

### 3. インポートするメディアファイルの場所まで移動する。

### 4. これらのファイルを選択して、Openをクリックする。

選択したファイルが、Audio File Importerにロードされる。

### 5. グループボックスImport Modeで、Replace audio filesを選択する。

### 6. グループボックスObject Typeで、Import as Sound Voiceを選択する。

### 7. Destination languageリストから、ランゲージを1つ選択する。

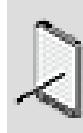


#### 注記

プロジェクトトランゲージリストはLanguage Managerで作成します。プロジェクトトランゲージについては[5章#####](#)を参照してください。

### 8. Importをクリックする。

Importingダイアログボックスが開き、ファイルインポート処理の進行を確認できる。



#### 注記

メディアファイルにエラーやコンフリクトがある場合は、Import Conflict Managerが開きます。コンフリクトの対処方法については「[ファイルインポート関連の問題への対処](#)」を参照。

## 関連トピック

- 「メディアファイルのインポート」
- 「テキストファイルからボイスアセットをインポート」
- 「SFXファイルの置換」
- 「ファイルインポート関連の問題への対処」
- 「Originalsフォルダのメディアファイルの並べ替え」
- 「キャッシュクリア」
- 「外部エディタでオーディオファイルを編集」

## ファイルインポート関連の問題への対処

メディアファイルをWwiseプロジェクトにインポートする際に、エラーが発生することがあります。発生する可能性のある2種類のエラーを、以下に示します。

- 「回復可能なエラーの対処」 はConflict Managerで解消できる
- 「回復不可能なエラーの対処」 はWwiseで解消できない

### 回復可能なエラーの対処

既にWwiseにあるファイルをインポートしようとすると、Audio File ImporterでReplace Modeを選択していない限り、Conflict Managerが開きます。目的に合わせて、以下の3種類のオプションから選択してください。

- Replace - 既存のオーディオファイルを、インポートするファイルで置換。
- Use Existing - オーディオソースに現在リンクされているファイルを、継続して使用。
- Cancel - インポート操作をキャンセル。

個別にオプションを選択することも、エラーリストの全てのファイルにオプションを適用することも可能です。

### かい回復可能なエラーのファイルを置換するには:

1. ウィンドウConflict Managerで、以下を行う:

全ファイルを置換するには、Set All Toエリアの、Replaceをクリックする。

ファイルを個別に置換するには、Error Listの、Replaceをクリックする。

### 2. Importをクリックする。

インポートリストにあるファイルが、指定した場所にインポートされる。

### 回復不可能なエラーの対処

Conflict Managerで回復不可能なエラーが発覚した場合、該当ファイルのファイルインポート操作をキャンセルしてください。

回復不可能なエラーはWwiseで解消できませんが、発生する可能性のある回復不可能なエラーの種類を知っておくと便利です。問題によってはWwise外で解消して、再度インポートを試すことができます。

回復不可能なエラーとその対策案を、下表に示します。Wwiseが対応するファイルについては「[PCMオーディオファイルフォーマット](#)」を参照してください。

Cause	Suggestion
Unsupported file format 未対応のファイルフォーマット	メディアファイルのフォーマットを波形オーディオフォーマット (.wav) に変更する。
Sample rate beyond specified range サンプルレートが指定範囲外	オーディオファイルのサンプルレートを1kHz-96kHzの範囲に変更する。
Bit rate beyond specified range ビットレートが指定範囲外	オーディオファイルのビットレートを16 bitまたは24 bitに変更する。
Unsupported input channel configuration 未対応の入力チャンネルコンフィギュレーション	オーディオチャンネル数を0.1~7.1のチャネルコンフィギュレーションに減らすか、このチャンネルコンフィギュレーションに対応するフォーマットにメディアファイルを変更する。
Audio file not found オーディオファイルがない	サウンドボイスオブジェクトに関して、レンジスランゲージのメディアファイルはない。
Audio file name is over 256 characters オーディオファイル名が256文字を超える	メディアファイルのファイル名を256文字未満に変更する。

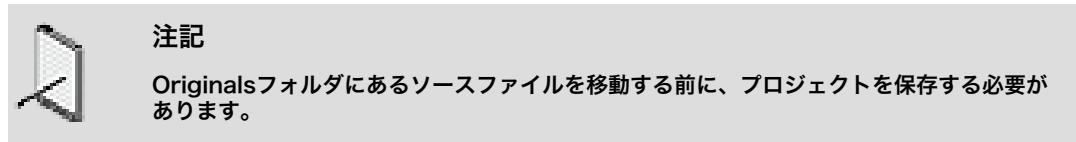
### 関連トピック

- 「[PCMオーディオファイルフォーマット](#)」
- 「[回復可能なエラーの対処](#)」
- 「[キャッシングクリア](#)」
- 「[外部エディタでオーディオファイルを編集](#)」

## Originalsフォルダのメディアファイルの並べ替え

プロジェクトのソースメディアのファイル数が増えた時は、サブフォルダに分けて整理できます。ただし、ソースファイルのフルパスとファイル名をWwise側で管理

する必要があるので、ファイルの移動は必ずWwise上で、File Managerから行ってください。



ディレクトリOriginalsのフォルダ間で、ファイルを移動するには:

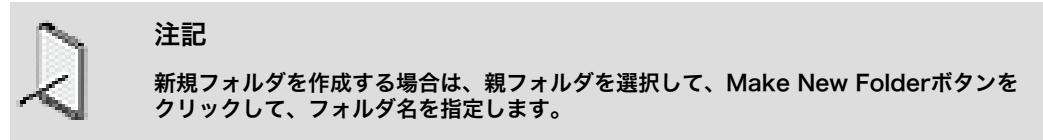
1. 現在のプロジェクトを保存する。
2. 以下のいずれかを行う:
  - メニューバーで、Project > File Manager...をクリックする。
  - [Shift]+[F1]を押す。

File Managerが開く。

3. Source Filesタブに切り替える。
4. 移動するファイル（単数または複数）を選択する。
5. 選択したファイルを右クリックして、メニューで、Moveを選択する。

Browse For Folderダイアログボックスが開く。

6. フォルダ層を移動して、ファイルの移動先となるフォルダを選択する。



7. OKをクリックして、ファイルを新規フォルダに移動する。

Process Logダイアログボックスが開き、移動した各ファイルの情報が表示される。

8. OKをクリックして、Process Logダイアログボックスを閉じる。

### 関連トピック

- 「[メディアファイルのインポート](#)」
- 「[メディアファイルの置換](#)」
- 「[ファイルインポート関連の問題への対処](#)」
- 「[キャッシュクリア](#)」
- 「[外部エディタでオーディオファイルを編集](#)」

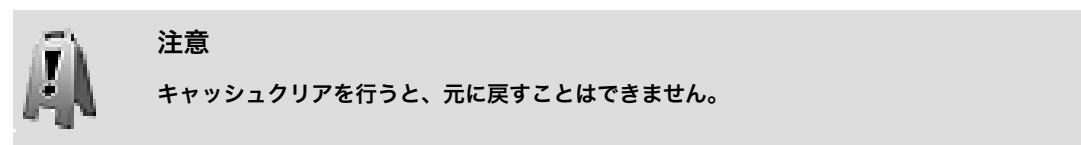
## キャッシュクリア

コンバージョン後のメディアファイルを効率的に管理するために、定期的にキャッシュフォルダをクリアして、不要なファイル、古いファイル、問題のあるファイルなどを削除する必要があります。キャッシュクリアの範囲を設定して、メディアファイルを限定的にキャッシュからクリアすることもできます。キャッシュクリアすると良い場合を、以下に示します。

- Wwiseオブジェクトを削除した後 - オブジェクトのコンバージョン済みメディアファイルやオーファン（孤立）ファイルは不要であるが、オブジェクトとの関連性が続き、キャッシュフォルダからマニュアルでクリアするまで残る。この場合、これらだけを削除することができる。
- プラットフォームコンバージョンを行ったため、低品質となった可能性がある場合 - プラットフォームの、コンバージョン後のファイルをフラッシュしてやり直したい。
- ランゲージファイルの最終バージョンが納品された時 - 新しいコンバージョン後ファイルだけを、キャッシュに残したい。

このような状況では、以下に従い、限定的にキャッシュフォルダをクリアできます。

- **Converted files** — クリアしたいコンバージョン後のファイルの種類を指定。キャッシュにある全てのコンバージョン後ファイルをクリアできる他、オーファン（孤立）ファイルのみのクリアも可能。
- **Platforms** - どのプラットフォームのファイルをクリアするかを指定。
- **Languages** - どのランゲージのファイルをクリアするかを指定。



### プロジェクトキャッシュをクリアするには:

1. Project メニューで、Clear Audio File Cacheを選択する。

Clear Audio File Cacheダイアログボックスが開く。

2. グループボックスAudio Filesで、以下のオプションを1つ選択する。

**Only orphan files**

**All converted files**

3. グループボックスPlatformsで、以下のオプションを1つ選択する。

- Current platform
- All platforms

4. サウンドボイスオブジェクトの場合は、グループボックスLanguagesで、以下のオプションを1つ選択する。

- Current language
- All languages

5. OKをクリックする。

Clearing Audio File Cacheダイアログボックスが開き、クリア処理の状況を確認できる。

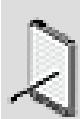
クリア処理が問題なく終了すると、指定したコンバージョン後のメディアファイルが、キャッシュから削除される。

## 関連トピック

- 「メディアファイルのインポート」
- 「メディアファイルの置換」
- 「ファイルインポート関連の問題への対処」
- 「Originalsフォルダのメディアファイルの並べ替え」
- 「外部エディタでオーディオファイルを編集」

## 外部エディタでオーディオファイルを編集

ゲームプロジェクトで使うオーディオファイルが準備できるまでに、オーディオエディタ（（単数または複数）が使用されます。また、プロジェクトにこれらのオーディオファイルをインテグレートした後も、さらに編集を加える場合も考えられます。例えば、長過ぎる台詞の最後を落としたり、口笛の曲に音符を追加したりすることもあるでしょう。Wwiseに既にインポートしたオーディオファイルであっても、それを選択して、ユーザーの好きなオーディオエディタで直接開くことができます。つまり、プロジェクトに既に実装したファイルの微調整が可能になります。



### 注記

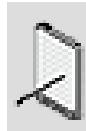
Wwiseで外部エディタを使用する前に、User Preferencesでエディタを指定する必要があります。詳しくは「[外部オーディオエディタの選択](#)」を参照してください。

必要な許可を得ている場合は、インポートしたオーディオファイルをProject Explorerで選択して、自分が選んだ外部オーディオファイルエディタで直接開けます。ファイル許可については「[Wwiseとソースコントロールシステムを使う](#)」を参照してください。

### 外部エディタでオーディオファイルを編集するには:

1. Project ExplorerのAudioタブで、サウンドオブジェクトを右クリックする。

ショートカットメニューが表示される。



### 注記

Contents Editorにあるソースや、コンテナやアクターミキサーを右クリックして、同時に複数のオーディオファイルを編集することもできます。

2. Ctrl-Eを押すか、コンテキストメニューで、Edit in External Editorを選択する。

使用可能なエディタリストが表示される。

3. オーディオファイルの編集に使用するエディタを選択する。

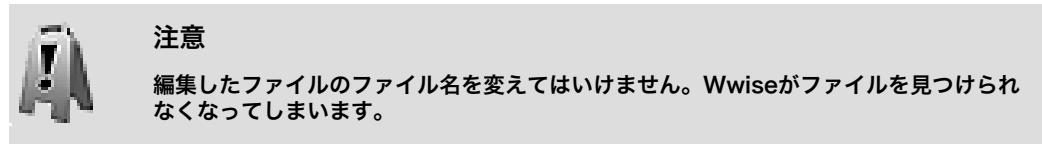
オブジェクトのオーディオファイルが、選択した外部エディタで開く。

4. ファイルの編集が終了したら、以下のパスのいずれかで、エディタに保存する。

サウンドは、[ProjectName]\Originals\SFX\[Filename.wav]の下に保存

ボイスは、[ProjectName]\Originals\Voices\[Language]\[Filename.wav]の下に保存

編集済みのメディアファイルを、Wwiseで再生できる。



## 関連トピック

- 「メディアファイルのインポート」
- 「メディアファイルの置換」
- 「ファイルインポート関連の問題への対処」
- 「Originalsフォルダのメディアファイルの並べ替え」
- 「キャッシュクリア」

## プラグインを使ってソースを作成する

Wwiseのオープンアーキテクチャを利用すれば、ソースプラグインを作成してオーディオを強化できます。Wwiseのオープンアーキテクチャを利用すれば、ソースプラグインを作成してオーディオやモーションを強化できます。シンセサイザやフィジカルモデリングのプラグインをWwiseに簡単にインテグレートして、サウンドやモーションのオブジェクトを作成するために使えます。また、プラグインのプロパティを調整して様々なサウンドエフェクトやモーションエフェクトをつくり出すことも可能です。

Wwiseには、様々なソースプラグインが同梱されています。

### 提供中のソースプラグイン

- Wwise Audio Input
- Wwise External Source
- Wwise Silence
- Wwise Sine
- Wwise Tone Generator
- Wwise Motion Generator
- SoundSeed Air - Wind\*
- SoundSeed Air - Woosh\*

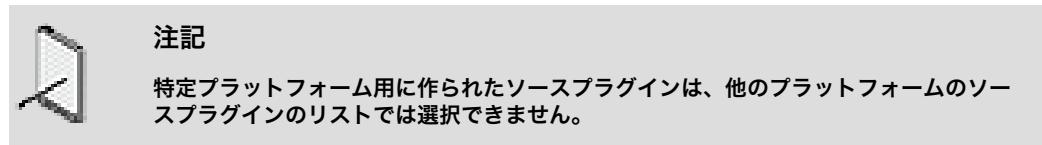
\* SoundSeed Airを使ったゲームの開発、実装、販売を計画している場合は、別途ライセンス購入が必要となります。詳しくは、 Audiokinetic社の営業担当: [sales@audiokinetic.com](mailto:sales@audiokinetic.com) までご連絡下さい。

### ソースプラグインを追加するには:

- Property Editorに、オブジェクトを1つロードする。

2. Contents Editorで、Add Sourceをクリックする。

Sourceメニューが表示され、使用可能なソースプラグインのリストが表示される。



3. 追加するソースプラグインを選択する。

ソースがオブジェクトに追加され、Contents Editorで新しい項目として表示されます。

4. ソースプラグインをダブルクリックすると、Source Plug-in Property Editorで全プロパティのリストが表示される。

5. 必要に応じて、プロパティを編集する。

### 関連トピック

- 「Contents Editorにオブジェクトを追加する」
- 「Contents Editorのオブジェクトやソースを試聴する」

## メディアファイル管理に関する、コツとベストプラクティス

オーディオ開発プロセス全体を通してプロジェクトのメディアファイルを効率的に管理するためのコツやベストプラクティスについて、以下のセクションで説明します。

### キャッシュをクリアする

スペース節約と効率的なファイル管理のためには、定期的なキャッシュクリアが必要です。以下の状況では、キャッシュクリアを行ってください。

- ・オブジェクトの削除後
- ・File Managerでファイルの削除後

### File Managerを使う

プロジェクトのメディアファイルを管理するために、File Managerの、Source Filesタブに表示される情報を利用することもできます。File Managerが、プロジェクトのOriginalsフォルダの情報を表示するので、必要に応じて、使用しないファイルをプロジェクトから削除できます。ただし、File Managerでオーディオファイルを削除しても、それに関連したオーディオキャッシュにあるコンバージョン後ファイルはクリアされません。これらを削除するには、Clear Audio Cacheダイアログボックスにある、Only orphan filesオプションを選択して、マニュアルで削除する必要があります。File Managerについては「[プロジェクトファイルのステータス確認](#)」を参照してください。

---

## 第8章 Actor-Mixer Hierarchyの構築

概要 .....	170
プロジェクト階層内のプロパティについて .....	174
Actor-Mixer Hierarchyの構築 .....	182
ランダム化したプロパティ値で、再生を強化 .....	187
Actor-Mixer Hierarchyに関する、コツとベストプラクティス .....	188

## 概要

典型的なゲームでは、アセットが数千件にものぼることが、珍しくありません。これらのアセットを、パソコン上のその他のファイルと同様に整理できることが大事です。Wwiseではプロジェクトのアセットを特別なグループ分けで整理することで、アセットを使った作業をしやすくしています。いくつかのオブジェクトをまとめて親子の関係を成立させることで、階層構造が構築されます。この構造で、プロジェクトのアセットを整理できるだけでなく、そのグループ全体のプロパティや動作を定義することも可能です。プロパティや動作の共通するオブジェクトを同じグループにまとめると便利です。操作できる範囲が広がり、時間の節約と開発プロセスの効率化にもつながります。

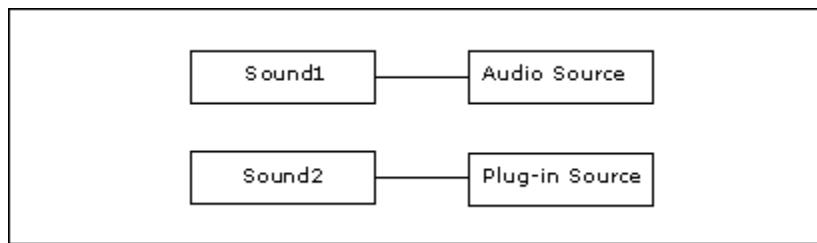
以下に示すオブジェクトタイプを組み合わせて、アセットをまとめ、プロジェクトのActor-Mixer Hierarchyを構築します。

- ・「サウンドオブジェクトとは？」
- ・「Actor-Mixer Hierarchyの中で、コンテナやアクターミキサーを使う」

### サウンドオブジェクトとは？

Actor-Mixer Hierarchyの基盤を成すのがサウンドオブジェクトです。サウンドオブジェクトは、プロジェクト用に作成した個別のオーディオアセットを表します。なお、各サウンドオブジェクトに1つのソースが含まれ、このソースがゲーム中に再生される実際のオーディオコンテンツを決定します。

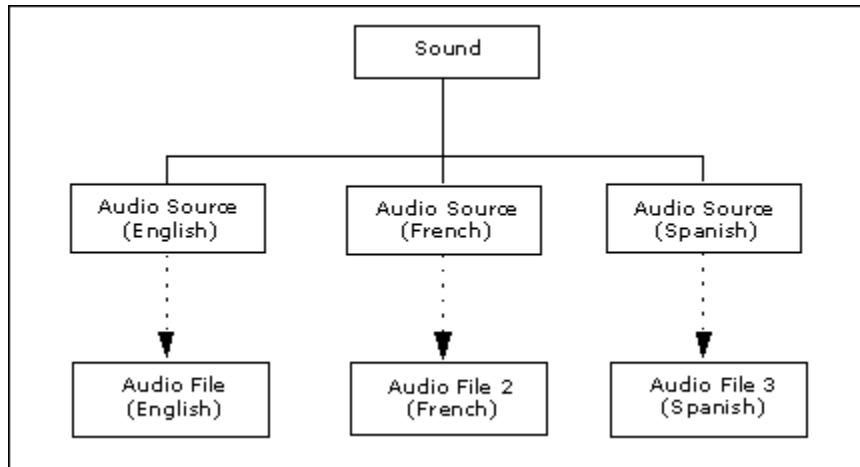
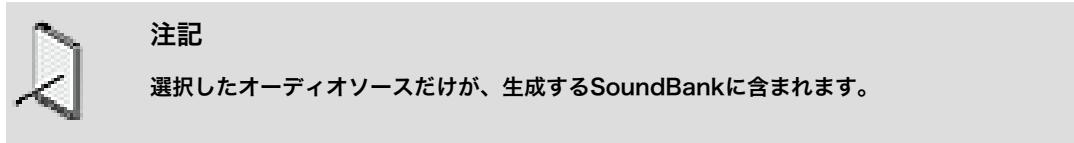
Wwiseでは、オーディオソースとプラグインソースの2種類のソースが使われます。



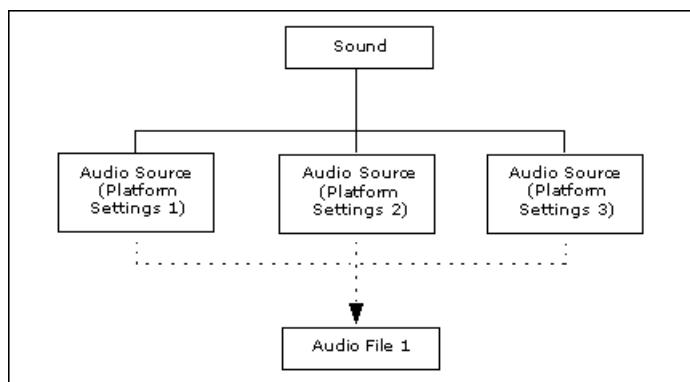
最も一般的なソースが、オーディオソースです。オーディオソースが、インポートしたオーディオファイルとサウンドオブジェクトの間の別レイヤとなります。プロジェクトにインポートされるオーディオファイルにリンクされている。



操作性と柔軟性を高めて、多言語やマルチプラットフォームのゲーム開発に対応するため、Wwiseのサウンドオブジェクトに複数のソースを入れることもできます。これらのオーディオソースは様々なオーディオファイルにリンクできるので、可能なサウンドを次々に試したいときや、同じサウンドを表すために複数のランゲージバージョンを用意する場合などに便利です。



また、プラットフォームごとに、サンプルレートなどのコンバージョン設定を変える場合は、1つのオーディオファイルに複数のオーディオソースをリンクできます。

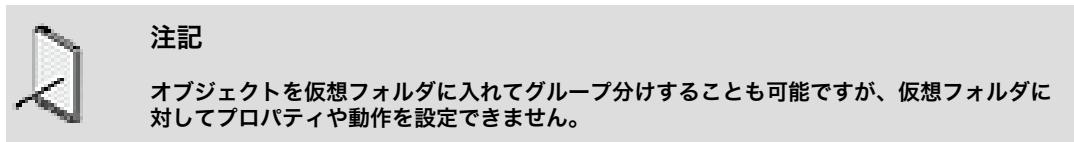


### オブジェクトをまとめて、Actor-Mixer Hierarchyを作成

様々なオブジェクトをグループにまとめて、そのグループ全体のプロパティや動作を定義することができます。オブジェクトをグループ分けすることで、プロジェクトの階層構造が構築されます。

Wwiseでは、オブジェクトをグループにまとめるために、以下の2種類のオブジェクトを利用します。

- ・「[コンテナの種類](#)」：オブジェクトをグループにまとめて、主にランダム、シーケンス、Switchなど特定の動作に従って再生させるために使う。
- ・「[Actor-Mixer Hierarchyの中で、コンテナやアクターミキサーを使う](#)」：オブジェクトをグループにまとめ、主にボリュームやピッチなど全体的なプロパティを指定するために使う。



ゲーム全体の構造の中で、コンテナとアクターミキサーの役割が異なるため、各オブジェクトのプロパティ、動作、再生方法などに関して設定できる範囲がそれぞれ異なります。下表は、コンテナとアクターミキサーの違いをまとめたものです。

提供する機能	Container	Actor-Mixer
プロパティ値の設定	✓	✓
オブジェクトの動作の設定	✓	#
オブジェクトの再生	✓	#

### コンテナの種類

ゲームにあるオーディオは様々な動作をするため、使用するコンテナをランダム、シーケンス、Switch、ブレンドの4種類から、目的に合わせて選べます。ゲーム内のサウンドの再生方法を決めるための設定内容が、コンテナの種類によって異なります。

アイコン	内容	内容
	Random Container	ランダムに再生されるオブジェクトやコンテナ（単数または複数）のグループ。
	Sequence Container	特定の順番またはプレイリストに従って再生されるオブジェクトやコンテナ（単数または複数）のグループ。
	Switch Container	ゲーム内のあるエレメントに対して存在するいくつかの選択肢に対応して、一連のSwitchに整理された、オブジェクト、コンテナのグループ。ゲームでどのSwitchが有効であるかによって、再生するオブジェクトが決まる。
	Blend Container	同時に再生されるオブジェクトやコンテナ（単数または複数）のグループ。このコンテナ内の様々なオブジェクトは、RPC使用によるGame Parameter値へのオブジェクトプロパティのマッピングが実行されるブレンドトラックにグループ分けが可能。Game Parameter値に基づいて、ブレンドトラック内のオブジェクトの間にクロスフェードを適用することも可能。

### Actor-Mixer Hierarchyの中で、コンテナやアクターミキサーを使う

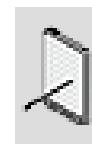
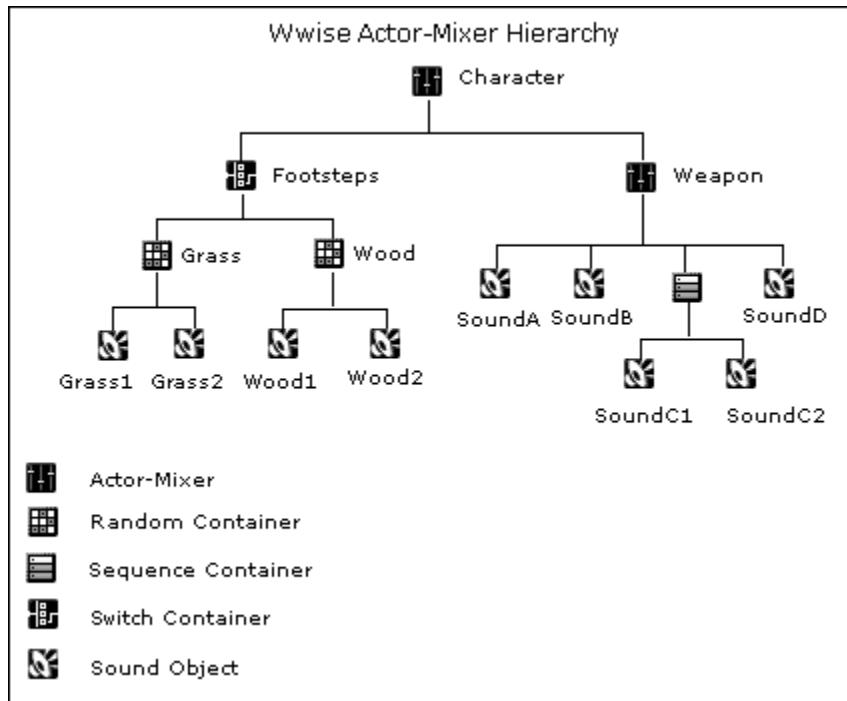
コンテナもアクターミキサーも、プロジェクト階層の各種アセットをグループにまとめるために使いますが、採用するレベルや使用目的が異なります。

コンテナは、Actor-Mixer Hierarchyの第2レベルに位置し、親オブジェクトにも子オブジェクトになります。コンテナの中にサウンド、コンテナなどを入れて、グループを形成します。さらに、コンテナの中にコンテナを入れてネスト化することで、多様な結果を実現させてリアルな状況を表現することも可能です。

アクターミキサーもまた、プロジェクト階層内のオブジェクトをグループにまとめるために使用しますが、コンテナよりも1段上のレベルに位置します。つまり、

アクターミキサーがコンテナの親となることがあっても、逆はありません。アクターミキサーにサウンド、コンテナ、他のアクターミキサーなどを無制限に入れ、親として設定できます。多数のWwiseオブジェクトを1つのグループにまとめて、全体に共通するプロパティを適用したい時に、アクターミキサーを使います。

ゲームの1人のキャラクターに関するサウンドアセットをグループにまとめるために、コンテナやアクターミキサーを活用した例を、下図に示します。

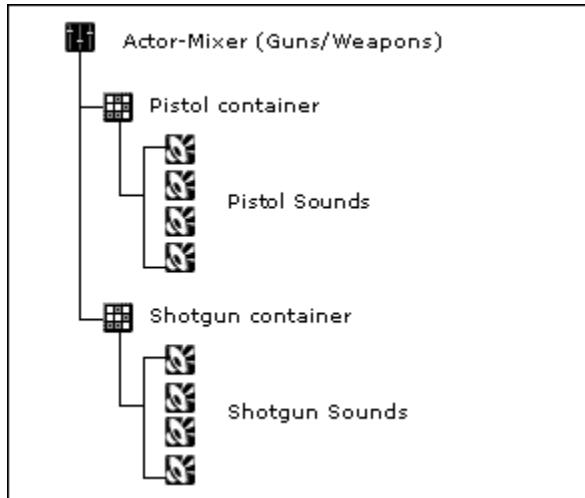


### 注記

モーションアセットの場合も、同じ階層レベルにコンテナやアクターミキサーを配置して階層を作り、グループにまとめることができます。

## アセット整理の実践例

例えば、ファーストパーソンシューターゲームで、7種類のウェポンを使うとします。そして1つのウェポンを表す複数のサウンドに、同様のプロパティを設定したいとします。その場合、そのウェポンに関連する全てのサウンドをまとめて、1つのSequence ContainerやRandom Containerに入れます。次に、多数のウェポンのそれぞれのコンテナを、1つのSwitch Containerにまとめて入れて、全ウェポンのボリュームやピッチなどのプロパティを、まとめてコントロールします。



アセットの構造は、ゲームデザインに基づき、プロジェクトの初期段階で構築します。その時に考慮すべきプロジェクトの他のエレメントとして、Work Unit、ルーティング、ゲームシンクなどがあります。プロジェクト全体を意識することで、様々なオブジェクトを効率的にグループ分けすることができます。オブジェクトをグループ分けする際の工夫については、Actor-Mixer Hierarchyの構築に関するコツとベストプラクティスの「[Actor-Mixer Hierarchy](#)で、オブジェクトをグループにまとめる」を参照してください。

ルーティング構造やWork Unitの設定、ゲームシンクの作成などについては、以下のセクションを参照してください。

- [6章Workgroupの活用](#)
- [「概要」](#)
- [17章Stateの活用](#)
- [18章Switchの使い方](#)
- [19章RTPCの使い方](#)
- [20章Triggerの使い方](#)
- [21章ダイナミックダイアログの、StateやState Groupの活用](#)

## プロジェクト階層内のプロパティについて

ゲームの構造は複雑な設計となるため、各種プロパティが階層内のオブジェクトにどのように適用されるのかを、理解しておく必要があります。Wwiseで設定するオブジェクトのプロパティには、以下の2種類があります。

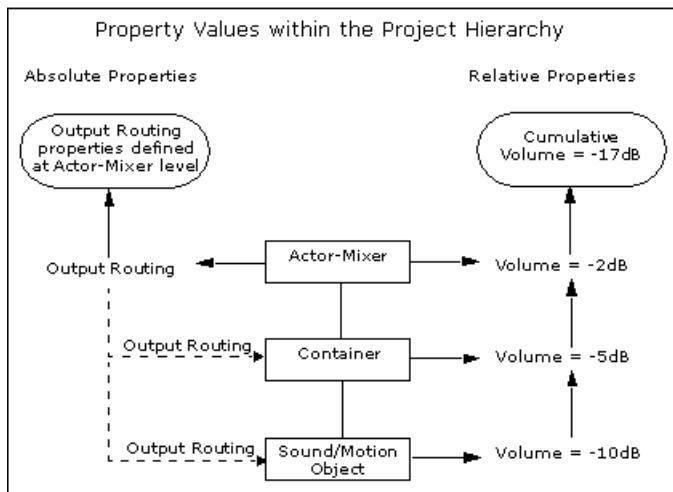
- **Relative properties:** 階層レベルごとに決定する、累積されるプロパティで、ピッチやボリュームなどがこれに該当する。相対プロパティの数値の合計が、最終的なプロパティ値となる。なお、全ての相対プロパティは、累計値が最小リミット、または最大リミットに達した時点で制限されることに注意すること。プロパティの最小値と最大値を、以下に示す。
  - Volume: (-200 to +200) 単位dB
  - Pitch: (-2,400 to 2,400) 単位cents

- Low-pass filter: (0 to 100) 単位%
- High-pass filter: (0 to 100) 単位%



- **Absolute properties:** 一般的には最上位で定義されるプロパティで、その下の子オブジェクトに受け継がれる。例えば、オブジェクトのアウトプットルーティングなど。設定は階層の最上位で設定するものの、実際にはその下の各レベルで適用される。また、絶対プロパティを階層中の各レベルでオーバーライドすることが可能。

この2種類のプロパティ値がプロジェクト階層に対してどのように適用されるかを、下図で示します。



プロジェクトにアセットをインポートして、各種オブジェクトを活用してアセットをロジカル構造に整理できたら、各アセットのプロパティや動作を設定します。

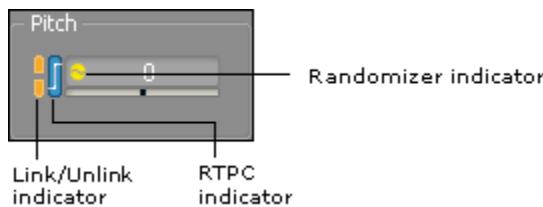
Wwiseのオブジェクトのプロパティや動作を個別に設定して、そのオブジェクトを定義して他と差別化します。ボリュームやピッチなどのプロパティは、オブジェクトの特徴を定義します。一方、動作 (behavior) 設定とは、ループ再生やストリーミング再生など、オブジェクトの再生方法を決めるものです。

また、階層内の全ての種類のオブジェクトに対して、まとめてプロパティや動作を設定することも可能です。こうすることで、複数のオブジェクトが入ったグループのプロパティを全体的にコントロールして、グループ内のどのオブジェクトを、どのような順番で再生するのかを定義します。

さらに、対応している特別なエフェクトを特定のオブジェクトに適用し、目標とするユニークなエクスペリエンスをつくり出すこともできます。you are aiming for. なお、階層の最上位でエフェクトを適用すると、実際にはその下の各レベルでエフェクトが適用されます。

## プロパティのインジケータ

Property Editorや、Contents Editorに表示されるプロパティ値に、1つ以上のインジケータ（表示）が付くことがあります。これらのインジケータは、そのプロパティ値がアクティブな他のプラットフォームにリンク（Link）しているのか、そのプロパティ値がRPCでGame Parameterに結び付けているのか、そのプロパティ値にRandomizerが適用されているのか、などを示します。



状況に応じて使われる各種インジケータを、下表に示します。

Indicator	名前	内容
	Link	このプロパティ値が、他のアクティブなゲームプラットフォームの設定とリンクしていることを示す。
	Unlink	このプロパティ値が、他のアクティブなゲームプラットフォームの設定とリンクせず、独立して設定された値であることを示す。
	Partial Unlink	このプロパティ値が、現在のプラットフォームでは他とリンクしているが、それ以外の1つ以上のアクティブなプラットフォームにおいて、設定がリンクしていないことを示す。
	RPC - On	このプロパティ値が、ゲーム内のパラメータ値にRPCで結び付けていることを示す。
	RPC - Off	このプロパティ値が、ゲーム内のパラメータ値に結び付いていないことを示す。
	State - On	Randomizerエフェクトが適用されたプロパティ値。
	State - Off	Randomizerエフェクトが適用されたプロパティ値。
	Randomizer - On	Randomizerエフェクトが適用されたプロパティ値。
	Randomizer - Off	Randomizerエフェクトが適用されていないプロパティ値。

リンク機能やRandomizer機能を有効・無効にするには、対象のプロパティ値を右クリックします。RPC機能は、Property EditorのRPCタブで設定します。

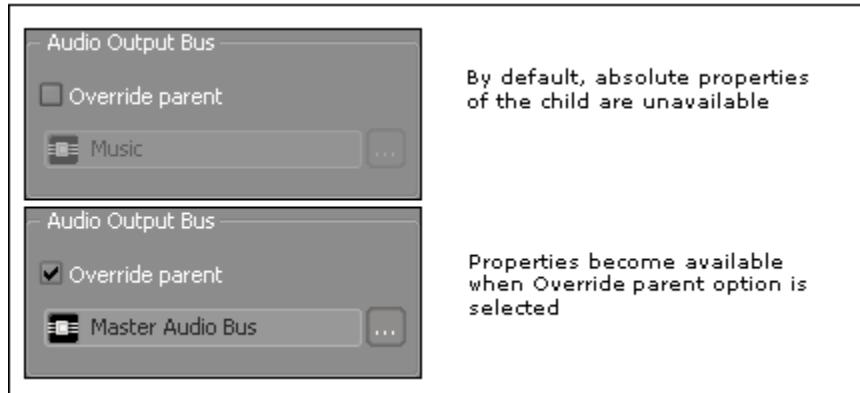
プロパティ値のリンク・アンリンク、RPCの使用、プロパティ値のランダム化などについては、以下のセクションを参照してください。

- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「Game Parameterを使って、プロパティ値をコントロールする」
- 「ランダム化したプロパティ値で、再生を強化」

## 絶対プロパティの設定

絶対プロパティ（Absolute property）の設定は、親の下にある全ての子オブジェクトに自動的に継承されるので、階層の中の最上位で親オブジェクトに対して設定

します。なお、あるオブジェクトに別のプロパティを指定する必要がある場合は、親のプロパティ設定をオーバーライドして新しく設定できます。デフォルトとして子オブジェクトに対して絶対プロパティを設定できませんが、Override parentオプションを選択すると、設定可能となります。



階層にあるオブジェクトの絶対プロパティとして、以下を設定できます。

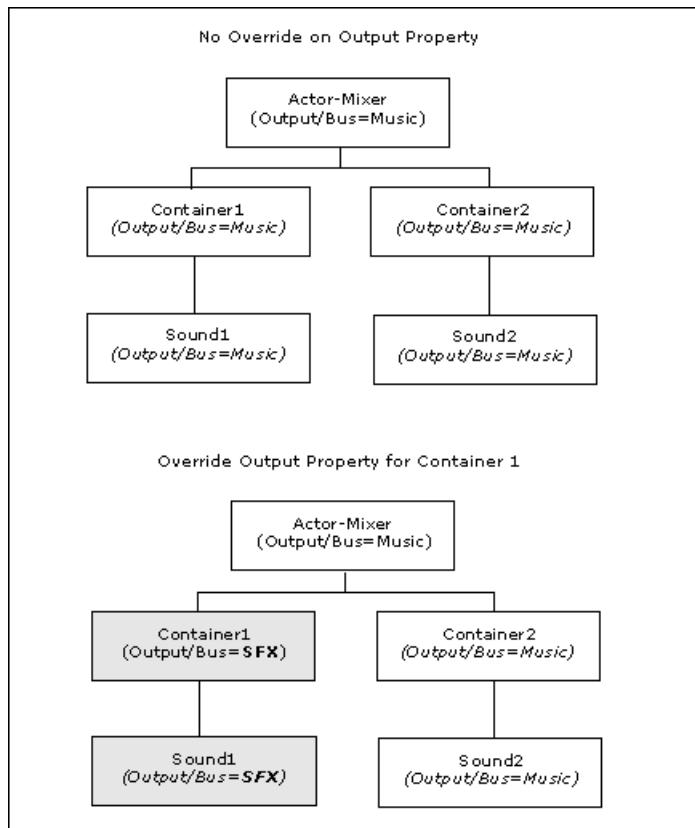
- 「オブジェクトのアウトプットルーティングの指定」



### 親のプロパティのオーバーライド

あるオブジェクトの絶対プロパティを変える必要がある場合は、親のプロパティ設定をオーバーライドして、設定を変えられます。なお、あるオブジェクトで設定をオーバーライドした場合、そのオブジェクトの子オブジェクトでも設定がオーバーライドされます。

オーバライドコマンドを実行した場合、下図の通り、そのオブジェクトとその下にある全てのオブジェクトのプロパティ設定が、変更されます。

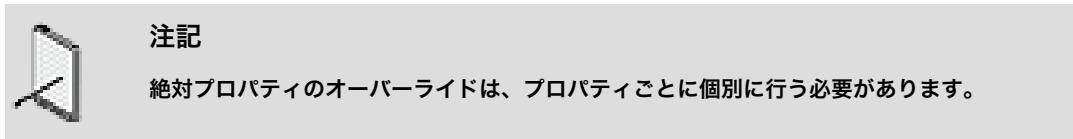


### 親のプロパティをオーバーライドするには

1. Property Editorに、子オブジェクトを1つロードする。
2. 変更したい絶対プロパティの、Override parentオプションを選択する。

そのグループ内のプロパティが有効になる。

3. 必要に応じてプロパティを変更する。



### 関連トピック

- 「絶対プロパティの設定」
- 「オブジェクトのアウトプットルーティングの指定」
- 「相対プロパティのボリューム、ピッチ、LPF、HPFの設定」

### 相対プロパティのボリューム、ピッチ、LPF、HPFの設定

相対プロパティ (Relative property) は、階層に存在するオブジェクトごとに設定します。相対プロパティは絶対プロパティと違い累積されるので、親のプロパティ値が子のプロパティ値に加算されます。

相対プロパティ値を個別にランダム化して、ゲーム内のリアリズムを高めることができます。プロパティ値のランダム化については「[ランダム化したプロパティ値で、再生を強化](#)」を参照してください。

オブジェクトの種類やプラットフォームによって、設定できない相対プロパティがあります。オブジェクトやプラットフォームごとに設定できる相対プロパティを、下表に示します。

オブジェクト、プラットフォーム	Volume	Pitch	LPFとHPF
サウンドオブジェクト	✓	✓	✓
コンテナ（全）	✓	✓	✓
Actor-Mixer	✓	✓	✓
Music track	✓		✓
Music Segment	✓		✓
ミュージックコンテナ（全）	✓		✓

### オブジェクトの相対プロパティを定義するには

- Property Editorに、オブジェクトを1つロードする。
- 以下のプロパティに関して、数値を入力するか、プロパティのスライダをドラッグして、値を設定する。
  - Volume
  - Pitch
  - LPF
  - HPF



#### 注記

複数のオブジェクトの相対プロパティを同時に編集するには、Project Explorerで複数オブジェクトを選択して、選択を右クリックして、Show in Multi Editorを選択します。

### 関連トピック

- 「[絶対プロパティの設定](#)」
- 「[ローパスとハイパスフィルタ値と対応するカットオフ周波数を関連付ける](#)」
- 「[ランダム化したプロパティ値で、再生を強化](#)」
- 「[オブジェクトの再生方法の設定](#)」

### ローパスとハイパスフィルタ値と対応するカットオフ周波数を関連付ける

ランタイムのCPU使用を効率化して、LPFとHPFのモデルをVolume、Pitch, LFE（階層を通して相対的）と同じモデルにするために、ローパスフィルタのプロパティを0から100%の間でノーマライズしてあります。

### LPF値に対応するカットオフ周波数の例

あるサウンド構造のサウンドとコンテナ:

- ・サウンドのLPF = 15%
- ・コンテナのLPF= 30%

このシナリオでは、ランタイムに適用される最終的なLPFは45%であり、カットオフ周波数は1922 Hzとなります。

実際のカットオフ周波数値をWwiseのローパスフィルタで使った場合は、おおよそ次のような状況になります:

- ・サウンドのLPF = 13,500Hz
- ・コンテナのLPF = 7000 Hz

この事例では、ランタイムに最終的なカットオフ周波数を定義するのでは、不明確です。

### WwiseのLPFとHPF値カットオフ周波数

Wwiseの0から100までのLPF・HPF値と、それに対応する実際のカットオフ周波数 (Hz) のおおまかな対比表を、以下に示します。これはWwiseのLPFとHPFの相対値システムを、参考用に単純化したもので、実際のカットオフ周波数はこれと多少異なるかもしれません。

LPF値 :	カットオフ周波数 (Hz) :	HPF値 :
0	20,000	100
1	19,567	99
2	19,133	98
3	18,700	97
4	18,267	96
5	17,833	95
6	17,400	94
7	16,967	93
8	16,533	92
9	16,100	91
10	15,667	90
11	15,233	89
12	14,800	88
13	14,367	87
14	13,933	86
15	13,500	85
16	13,067	84
17	12,633	83
18	12,200	82
19	11,767	81
20	11,333	80
21	10,900	79
22	10,467	78
23	10,033	77
24	9,600	76

LPF値：	カットオフ周波数 (Hz)：	HPF値：
25	9,167	75
26	8,733	74
27	8,300	73
28	7,867	72
29	7,433	71
30	7,000	70
31	6,422	69
32	5,892	68
33	5,405	67
34	4,959	66
35	4,550	65
36	4,174	64
37	3,829	63
38	3,513	62
39	3,223	61
40	2,957	60
41	2,713	59
42	2,489	58
43	2,283	57
44	2,095	56
45	1,922	55
46	1,763	54
47	1,618	53
48	1,484	52
49	1,361	51
50	1,249	50
51	1,146	49
52	1,051	48
53	964	47
54	885	46
55	812	45
56	745	44
57	683	43
58	627	42
59	575	41
60	528	40
61	484	39
62	444	38
63	407	37
64	374	36
65	343	35
66	315	34
67	289	33
68	265	32

LPF値：	カットオフ周波数 (Hz)：	HPF値：
69	243	31
70	223	30
71	204	29
72	188	28
73	172	27
74	158	26
75	145	25
76	133	24
77	122	23
78	112	22
79	103	21
80	94	20
81	86	19
82	79	18
83	73	17
84	67	16
85	61	15
86	56	14
87	51	13
88	47	12
89	43	11
90	40	10
91	36	9
92	33	8
93	31	7
94	28	6
95	26	5
96	24	4
97	22	3
98	20	2
99	18	1
100	17	0

## Actor-Mixer Hierarchyの構築

新規プロジェクトを設定してから、Project Explorerの、Audioタブで各種アセットの構造を作成します。ミュージック以外のアセットを含む構造が、Actor-Mixer Hierarchyです。このアクターミキサー階層にオブジェクトを追加して、階層内の様々なレベルでグループにまとめることで、相互関係を成立させます。ここで追加できるオブジェクトの種類を、下表に示します。

Object	アイコン	内容
Sounds	#	オーディオアセットを表すWwiseオブジェクトで、オーディオソースが含まれる。以下の通り、2種類のサウンドオブジェクトがある。
		Sound SFX - サウンドエフェクトのオブジェクト。

Object	アイコン	内容
		Sound Voice - サウンドボイスのオブジェクト。
Containers	#	オブジェクト、または他のコンテナを含むオブジェクトグループで、特定の動作設定に従って再生される。コンテナ内の子オブジェクトに適用されるプロパティを設定できる。以下の通り、4種類のコンテナがある。
		Random Containers - ランダムに再生されるオブジェクト（単数または複数）のグループ。
		Sequence Container - 特定のプレイリストに従って再生されるオブジェクトやコンテナ（単数または複数）のグループ。
		Blend Container - 同時に再生されるオブジェクトコンテナ（単数または複数）のグループ。Blend Container内のサウンドやコンテナは、RPC使用によるGame Parameter値へのサウンドプロパティのマッピングが実行されるブレンドトラックにグループ分けが可能。Game Parameter値に基づいて、ブレンドトラック内のオブジェクト間にクロスフェードを適用することも可能。
		Switch Containers - ゲームの特定エレメントに対して存在する様々な選択肢に対応するSwitchやStateに整理された、オブジェクトやコンテナ（単数または複数）のグループ。
Actor-Mixers		他のオブジェクトを入れられる、上位オブジェクト。アクターミキサーに設定されたプロパティは、その下にあるオブジェクトのプロパティに適用される。 また、アクターミキサー内のオブジェクトを、仮想フォルダでグループにまとめることも可能。
Virtual Folders		他のオブジェクトを入れられる、上位エレメント。仮想フォルダは、コンテナ、サウンドなどの子オブジェクトとすることはできない。
Work Unit		複数の人が同時に同じプロジェクトの作業ができるように、プロジェクトを分割するために使う上位エレメント。ワークユニットの中に、プロジェクト階層内のアセットを入れるほか、ステート、エフェクト、サウンドバンクなど他のWwiseエレメントも入れられる。
Physical Folders		プロジェクトで使用する他の物理フォルダやWork Unitを含めることができる、高レベルにあるエレメント。物理フォルダを、コンテナ、またはサウンドの子オブジェクトにすることはできない。

### Actor-Mixer Hierarchyへ、オブジェクトを追加

階層構築のスタート地点は、Work Unitです。新しい階層はDefault Work Unitから直接、作成し始めることができます。また既に作成されたWork Unitにオブジェクトを追加し始める方法も、チーム作業の場合によくとられます。Work UnitやWorkgroupについては[6章Workgroupの活用](#)を参照してください。

実際に階層を構築するには、以下のいずれかを行います。

- ・プロジェクト構造を設定してから、構造にオーディオファイルをインポートする。
- ・オーディオファイルをインポートしてから、プロジェクト構造を作成してオーディオファイルを整理する。

オーディオファイルのインポートと、これによってアクターミキサー階層の新規オブジェクトが作成される方法については[7章メディアファイルの活用](#)を参照してください。

## Project Explorerのツールバーで、子オブジェクトを作成するには:

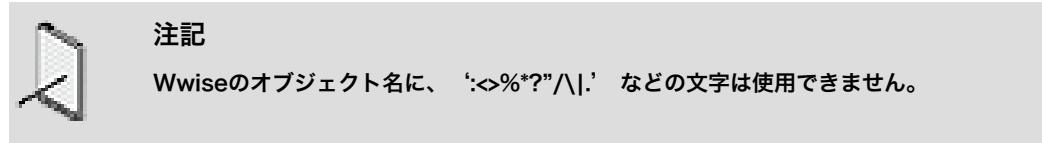
1. Project Explorerの、Audioタブで、作成するオブジェクトを入れるWork Unitを選択する。

Project Explorerのツールバーで、複数のアイコンが有効になる。

2. リストで、追加したいオブジェクトのアイコンをクリックする。

アクターミキサー階層の選択したワークユニットの下に、そのオブジェクトが追加されます。

3. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



これで、階層に他のオブジェクトを追加する準備が完了する。作業前にオブジェクト同士の関係を充分に理解することで、適切に整理することができる。また、プロジェクトの後の段階において、時間の大幅な節約にもつながる。

## Actor-Mixer Hierarchyの中に、子オブジェクトを作成するには:

1. Project Explorerの、Audioタブで、作成するオブジェクトを入れるWork Unitを右クリックする。
2. ショートカットメニューで、New Childを選択する。

サブメニューに、追加できるオブジェクトのリストが表示される。

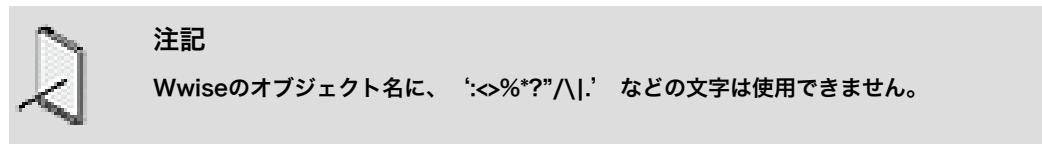
この階層レベルで追加できるオブジェクトの種類は、以下の通りである。

- Virtual Folder
- Actor-Mixer
- Switch Container
- Random Container
- Sequence Container
- Blend Container
- Sound SFX
- Sound Voice

3. リストで、追加したいオブジェクトをクリックする。

アクターミキサー階層の選択したワークユニットの下に、そのオブジェクトが追加されます。

4. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



これで、階層に他のオブジェクトを追加する準備が完了する。作業前にオブジェクト同士の関係を充分に理解することで、適切に整理することができる。また、プロジェクトの後の段階において、時間の大幅な節約にもつながる。

## 関連トピック

- ・「親オブジェクトの追加」
- ・「オブジェクトのコピー＆ペースト」
- ・「オブジェクトの移動」
- ・「オブジェクトのカット、削除」
- ・「プロジェクトのWork Unitの作成」
- ・「Work Unitに、プロジェクトエレメントをアサイン」

### 親オブジェクトの追加

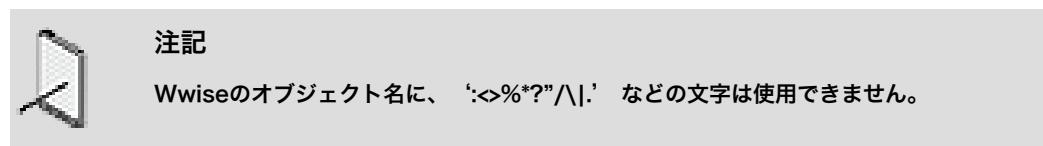
ワークユニットの最初のオブジェクトを追加すれば、続いて他のオブジェクトを Actor-Mixer Hierarchy に追加して、相互に親子関係を持たせることができます。親オブジェクトは他のオブジェクトを含むので、親オブジェクトを作成してから、既存のオブジェクトを移動して中に入れることができます。親子関係を設定するメリットは、親のプロパティを変更して動作設定を変えると、下の子オブジェクトにも適用されることです。Actor-Mixer Hierarchy のプロパティについては「[プロジェクト階層内のプロパティについて](#)」を参照してください。

Project Explorer のツールバーで、親オブジェクトを作成するには:

1. Project Explorer の、Audio タブで、親オブジェクトを追加するオブジェクトを選択する。
2. Shift キーを押して、選択したオブジェクトの親として追加できるオブジェクトのアイコンを表示させる。
3. リストで、追加したいオブジェクトのアイコンをクリックする。

アクターミキサー階層の選択したオブジェクトの親として、オブジェクトが追加されます。

4. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



親オブジェクトを作成するには:

1. Project Explorer の、Audio タブで、親を作成するオブジェクトを右クリックする。
2. ショートカットメニューで New Parent を選択する。

サブメニューに追加できるオブジェクトのリストが表示される。

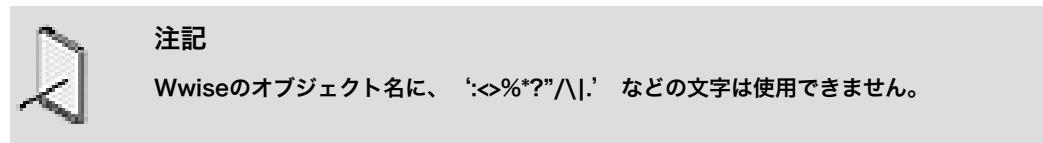
選択したオブジェクトの階層レベルによって、以下を新しい親オブジェクトとして追加できる。

- Switch Container
- Random Container
- Sequence Container
- Blend Container
- Virtual Folder
- Actor-Mixer

3. リストで、追加したいオブジェクトをクリックする。

アクターミキサー階層に新しい親オブジェクトが追加され、そのオブジェクトの種類に従いデフォルト名が付けられます。

4. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



## 関連トピック

- [「Actor-Mixer Hierarchyへ、オブジェクトを追加」](#)
- [「オブジェクトのコピー &ペースト」](#)
- [「オブジェクトの移動」](#)
- [「オブジェクトのカット、削除」](#)
- [「プロジェクトのWork Unitの作成」](#)
- [「Work Unitに、プロジェクトエレメントをアサイン」](#)

## Actor-Mixer Hierarchyのオブジェクトの管理

アクターミキサー階層ではショートカットメニューとWindowsの標準ショートカットを使って、オブジェクトの名前変更、カット、コピー、ペーストなどができます。なお、階層の構造を変える際は、以下に注意してください。

### オブジェクトの移動

Action	Results
オブジェクトの移動	階層内でオブジェクトの場所を変えると、移動先の親のプロパティや動作が、そのオブジェクトに適用される。 イベントと結び付いたオブジェクトを移動すると、そのオブジェクトのイベントへの結び付きが維持される。

### オブジェクトのコピー&ペースト

Action	Results
オブジェクトのコピー	オブジェクトをコピーして新しい場所にペーストすると、ペースト先の親のプロパティや動作がそのオブジェクトに適用される。また、その子オブジェクトにも適用される。

Action	Results
	イベントと結び付いたオブジェクトをコピーすると、新しいオブジェクトにイベントが結び付かない。

### オブジェクトのカット、削除

Action	Results
オブジェクトのカット、削除	<p>オブジェクトをカットまたは削除すると、その子オブジェクトも削除される。</p> <p>関連するコンバージョン済みオーディオファイルは、削除されない。関連するオブジェクトが存在しない、コンバージョン済みのオーディオファイルは、オーファン（Orphan / 孤立）ファイルと呼ばれる。オーファンファイルを削除するには、オーディオキャッシュのクリアが必要となる。オーファン（孤立）オーディオファイルの削除については「<a href="#">キャッシュクリア</a>」を参照してください。</p> <p>イベントと結びついたオブジェクトを削除またはカットすると、そのイベントのオブジェクトが1つ不足する結果となる。</p>

## ランダム化したプロパティ値で、再生を強化

オブジェクトがゲーム中に再生される度にユニークな感じを出すために、Wwiseで設定できる様々なプロパティ値にRandomizer機能を適用することができます。Randomizerは、オブジェクトが再生される度に、事前に指定した範囲内の数値を選択して、そのオブジェクトのプロパティ値を毎回、変化させます。

Randomizerを使うには、以下を行います。

- 「[Randomizerの有効化と無効化](#)」
- 「[Randomizerのプロパティ編集](#)」

### Randomizerの有効化と無効化

Randomizerアイコンの付いたプロパティは、全てRandomizer機能を適用できます。Randomizerはデフォルトで無効にしてありますが、欲しいエフェクトを出すために、簡単に有効にしてプロパティを変えられます。

### Randomizerを有効にするには:

1. Randomizerアイコンがあるテキストボックスまたはスライダを、右クリックする。

ショートカットメニューが表示される。

2. メニューで、Enable Randomizerを選択する。

Randomizerビューが表示され、テキストボックスのRandomizerアイコンが黄色になる。



#### Tip

Randomizerアイコンをダブルクリックして、Randomizerビューを開くこともできます。

## 関連トピック

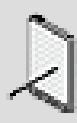
- 「ランダム化したプロパティ値で、再生を強化」
- 「Randomizerのプロパティ編集」

## Randomizerのプロパティ編集

Randomizerは、プロパティの数値として可能な範囲を設定するだけなので、要求されるのは最小値（Min）と最大値（Max）だけです。ここで設定する値は絶対値ではなく、設定されたプロパティ値をオフセットするために使います。

### Randomizerのプロパティを編集するには:

1. プロパティのRandomizerアイコンをダブルクリックする。  
Randomizerビューが表示される。
2. テキストボックスMinに、そのプロパティの最小値として設定する数値を入力する。この最小値（Min）が、既存のプロパティ値に対するオフセットとして適用される。
3. テキストボックスMaxに、そのプロパティの最大値として設定する数値を入力する。この最大値（Max）が、既存のプロパティ値に対するオフセットとして適用される。



#### 注記

複数のオブジェクトのRandomizer機能のプロパティを同時に編集するには、Project Explorerで複数オブジェクトを選択して、選択を右クリックして、Show in Multi Editorを選択します。

## 関連トピック

- 「ランダム化したプロパティ値で、再生を強化」
- 「Randomizerの有効化と無効化」

## Actor-Mixer Hierarchyに関する、コツとベストプラクティス

Wwiseで階層構造を作成する機能は、非常に柔軟性に優れています。プロジェクトの初期段階で一貫した分かりやすい方針を立てることは、以後の時間節約や効率性につながります。オーディオプロジェクトの進め方には様々な手法がありますが、ゲームで最高の結果を達成するために参考となるコンセプトを、以下に紹介します。

### Actor-Mixer Hierarchyで、オブジェクトをグループにまとめる

階層の構築を始める前に、オーサリング時間を短縮するだけでなく、プロジェクトのメモリ負荷を抑えるためにも、オブジェクトの最適な整理方法を充分に検討する必要があります。計画内容にもありますが、プロジェクトの様々なオブジェクトを効率的にグループにまとめるための提案を、以下に示します。

Actor-Mixer Hierarchyで設定したポジショニングやRTPCなどのプロパティは、全ての子オブジェクトに継承されるので、メモリやCPUを節約する究極の手段と

もいえます。多数のサウンドオブジェクトを整理する時は、アクターミキサー内のグループ分けについて、以下の点を検討して下さい。

- ・プロパティ設定の処理が一回で済むように、共通設定をシェアさせる
- ・オーバーライド設定によってオブジェクトごとに処理する状況ができるだけ回避するために、オーバーライド数を制限する。



#### 注意

アクターミキサーのレベルでエフェクトを適用すると、適用方法としては効率的であっても、CPUは節約できません。アクターミキサーのレベルでエフェクトを設定すると、そのエフェクトのインスタンスが全ての子オブジェクトに適用されるためです。オブジェクトごとにエフェクトがリアルタイムで処理されるため、CPU負荷がかなり増加します。

メモリを効率的に使用するには、状況に応じてアクターミキサー内でオブジェクトをグループにまとめて、以下のプロパティをシェアさせます。

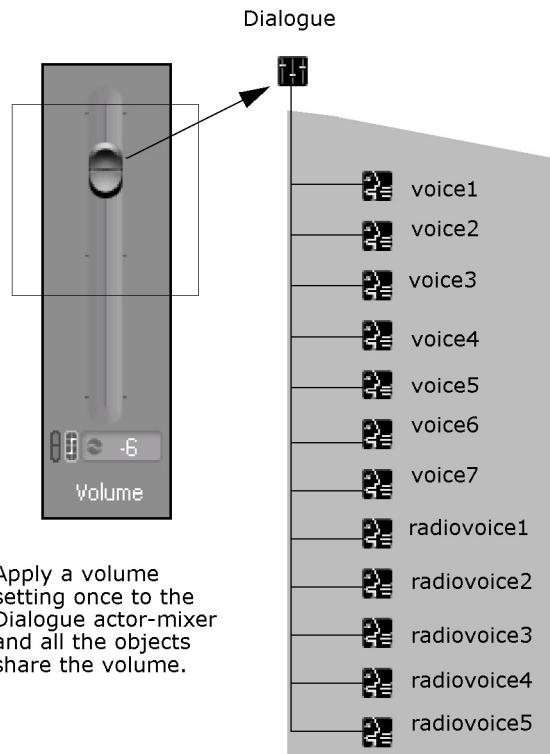
- ・Positioning
- ・RTPCs
- ・States
- ・Randomizers

例えば、サウンドが10個入ったアクターミキサーにおいて、サウンドポジションの設定を3Dにする場合を考えます。サウンド1つ1つで、Override parentのオプションを選択して、3D設定することができます。しかしこれでは、アクターミキサー自体を3Dポジショニングに設定した時と比べてランタイムのメモリ使用量が10倍になります。また、アクターミキサーに入ったサウンドのいくつかを2Dに設定する場合、アクターミキサー自体をゲーム定義による3Dポジショニングに設定した上で、そのうちのいくつかのサウンドを2Dポジションにした方が、メモリ使用量が最適化されます。2Dサウンドによるメモリ負担増はないので、2Dポジションのサウンドで、アクターミキサーをオーバーライドして2Dを適用します。

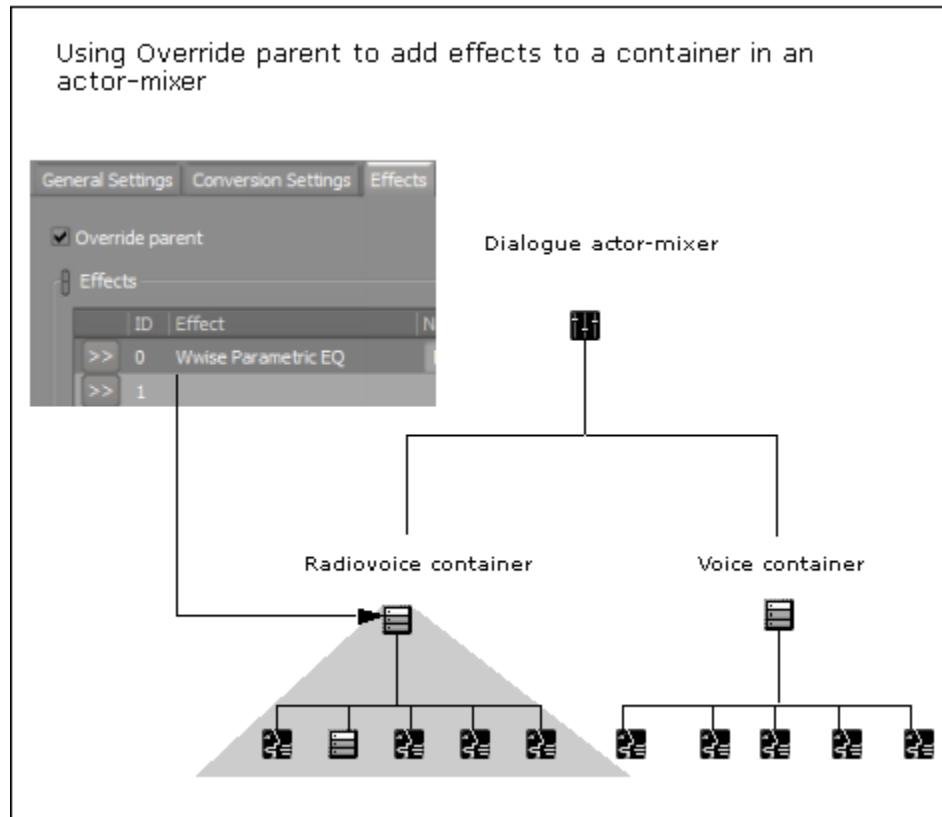
共通するプロパティをアクターミキサー全体に対して設定するのが一般的には最良ですが、状況によってはコンテナに対して共通プロパティを設定した方がメモリ消費を抑えられます。例えば、Random Containerに入った足音のように特定のコンテナのオブジェクトだけにポジショニングを設定したい状況では、ポジショニングのプロパティ設定をコンテナに対して設定した方が、親のアクターミキサーに対して設定するよりもメモリを節約できます。ただし、アクターミキサーの構造に入った全オブジェクトにポジショニングのプロパティを共有させたいのであれば、アクターミキサーに対して設定します。

#### 事例: オブジェクトをグループにまとめる

それでは上記のコンセプトを利用して、オブジェクトを効率的にアクターミキサーにまとめる方法について考えます。今回は、ゲームで使うダイアログアセットで作業をします。アセットは、キャラクターボイス（voice）として使うものと、無線通話（radio voice）に使うものに分かれます。全てのダイアログアセットに同じボリュームを適用するなど、共通して設定できるプロパティがあるため、まず、Dialogueというアクターミキサーにまとめます。



上図の通り、アクターミキサーのボリュームを設定した後に、無線通話のダイアログだけにパラメトリックEQエフェクトを追加します。この場合、無線ボイスを1つずつ編集して、個別にアクターミキサーの設定をオーバーライドさせてエフェクトを追加することができます。しかし、無線ボイスのファイルを1つのコンテナにまとめて、そのコンテナに対してアクターミキサー設定のオーバーライドとエフェクトの追加を適用させた方が、効率的です。



### リアルタイムミキシングと、オブジェクトのプロパティ

ゲームや、Game Simulator機能に接続した状態で、以下の相対プロパティを、Wwiseでリアルタイムで調整することができます。

- Volume
- Pitch
- LPF
- Center %
- RTPC値
- ステート切り替え、スイッチ切り替え
- Trigger
- 減衰コントロール
- 一部のオーディオやソースエフェクトの、プラグイン設定

 **注記**  
基本的なルールとして、ゲームパラメータにマッピングできるプロパティは、全てゲーム中にリアルタイムに変更できます。

ただし、これを実行するには、調整したいプロパティを持つオブジェクトを、Transport ControlまたはSoundcasterにロードする必要があります。オブジェクトがロードされていないと、そのオブジェクトがサウンドエンジンに登録されないため、変更を加えても有効になりません。Actor-MixerはTransport

Controlにロードできませんが、Actor-Mixerの子オブジェクトをロードすればその親オブジェクトがサウンドエンジンに登録されます。一旦オブジェクトを登録すると、ゲームとの接続が切れるまで、登録された状態が続きます。



### 注記

Transport Controlで、あるオブジェクトにピンを付けると、そのオブジェクトのピンを外すまで、他のオブジェクトをロードできないので、注意してください。ただし、Soundcasterにオブジェクトをロードした場合は、サウンドエンジンにそのオブジェクトが登録されます。

## 相対プロパティとパフォーマンス

ピッチなど、Wwiseの一部の相対プロパティは、プラットフォームによって、パフォーマンスに影響することがあります。Wwiseでは、ピッチをサンプルレートで管理します。したがってサウンドにピッチを適用するとファイルのリサンプル処理が発生し、CPU負荷が増加します。

---

## 第9章 アウトプットバス構造の構築

概要 .....	194
バスのプロパティ設定 .....	198
バス階層の構築例 .....	202

## 概要

バスはプロジェクト内のオブジェクトをミキシングして最終的なアウトプットを決めるためにグループにまとめるところで、詳しくは「[概要](#)」を参照してください。ただしプロジェクトの仕上段階に入る前から、様々なバスの役割やプロジェクトの全体構造の中でのバスの役目を理解しておくことが大切です。

プロジェクトのアセット構造を整理してアセットを戦略的に管理するように、プロジェクトのアウトプット側の整理も必要です。アウトプットバスをグループ分けしてマスター・ミキサー階層という階層構造に入れることで、バスの相対プロパティ、ステート、RTPCを定義して、プロジェクトのルーティングに適用するエフェクトも設定できます。

ミキシングプロセスを効率化するために、プロジェクトにとって最適なルーティング方法について、充分に検討してください。例えば、アンビエントミュージックや、銃音などのサウンドを、それぞれに対応するバスにルーティングすることで、ゲームのオーディオミキシングを、シンプルにできます。

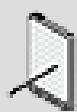
最初にMaster-Mixer Hierarchyのデフォルト構造を理解する必要があります。中にマスターバスがあり、それぞれの役割や階層の形式を以下に示します:

- 「[Master Audio Bus階層](#)」
- 「[Secondary Bus Hierarchy](#)」

### Master Audio Bus階層

Master Audio Bus階層は、複数のアウトプットバスからなる構造で、プロジェクトのサウンドとミュージックは、この階層にルーティングされます。この階層を構成する、機能の異なる3つのレベルを、以下に示します。

- **マスター・オーディオ・バス (Master Audio Bus)** — 階層の最上位エレメントであり、オーディオの最終的なアウトプットを決定する。作成したオーディオ・バスの名前を変更したり、削除したりできますが、移動はできません。マスター・オーディオ・バスに対して、エフェクトを適用できる。
- **Audio busses:** マスター Audio Busの下でいくつかのグループにまとめることができる任意のバス（単数または複数）で、サウンドミックスを整理したり完成させたりするために使う。これらのバスは名前変更、移動、削除が可能であり、エフェクトも適用できる。
- **Auxiliary busses:** Auxiliary BusやAudio Busの下でいくつかのグループにまとめることができる、任意のバス（単数または複数）。Audio Busと同様に、これらのバスの名前変更、移動、コピー、削除が可能であり、Auxiliary Busに対するエフェクトも適用できる。プロジェクトのどの位置に存在するサウンドオブジェクトも、AUXバスに送り、ボリューム、チャンネルコンフィギュレーション、ポジショニング、RTPCなどを調整したり、エフェクト、ステート、ミキサー・プラグインなどを適用したりできる。Auxiliary Busでは、ダッキング、ボイス、HDRミックスの調整はできない。



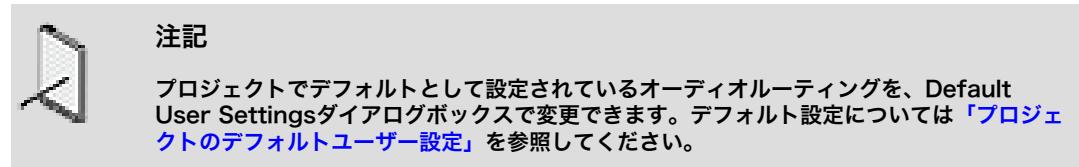
#### 注記

Audio Busを、Auxiliary Busの子に設定することはできません。子として設定できるのは、他のAuxiliary Busに限定されます。

インターフェース上で分かりやすいように、Audio BusやAuxiliary Busは、以下のアイコンで表されます。

アイコン	内容
	Audio Bus
	Auxiliary Bus

デフォルトとして、アクターミキサー階層のサウンドは、マスターオーディオバスにルーティングされます。ただし、アウトプット構造を作り上げる過程で、作成した様々なバスに、各種オブジェクトをルーティングできます。



## Secondary Bus Hierarchy

セカンダリバス階層 (Secondary Bus Hierarchy) は、メインアウトプット (テレビやスピーカー) 以外のアウトプットに送られるコンテンツをミックスするための、Audio Busの構造です。あなたのプロジェクトで使う様々な種類のアウトプットに合わせて、セカンダリオーディオバスを必要な数だけ作成できます。セカンダリアウトプットの一例として、ゲームコントローラのスピーカー、チャット用ヘッドフォン、DVRバイパスアウトプットなどがあります。メインのアウトプット構造と同じように、セカンダリ構造に設定できる子バスやAUXバスの数に、制限はありません。

Master Secondary Bus階層にルーティングしたサウンドを、セカンダリアウトプットに送る2種類の方式を、以下に示します。

- 対象サウンドの **Output Bus** のプロパティをセカンダリバス階層にあるバスに直接、設定する方式。通常は1つのセカンダリアウトプットインスタンスだけに結び付いているサウンドなどに向いています。例えば、プレイヤーが発射するガンショット、テニスラケットの打音、PDAの音、ゲームプレイのフィードバックなど。
- 対象サウンドをマスター オーディオバス階層のバスにルーティングしてからセカンダリバス階層のAUXバスへ送る、ユーザーセンドまたはゲームセンドの設定を追加する方式。これは同じサウンドが複数のアウトプットとテレビから同時に聞こえてくる場合に向き、例えばスパイカメラやアナウンスなどが考えられます。

バス階層は、ミキシングの構造を定義するだけです。そのミックスを具体的なアウトプットと関連付けるには、対応するマスターバス側で、適切な**Audio Device**シェアセットを選択します。Wwiseで提供されている「[組み込みオーディオデバイス](#)」のリストを確認してください。

ゲーム中に複数のインスタンスが存在するかもしれないプレイヤー関連のアウトプット (ゲームコントローラやヘッドフォンなど) の場合は、それに関連するミ

キシング階層が、プレイヤーごとにコピーされることを、知つておく必要があります。プロジェクトで定義するのは、特定のアウトプットタイプ用のミキシングレシピのようなものだけです。実際にどのコピー構造にルーティングされるかは、プログラマーの設定するリスナーとゲームオブジェクトの関連付けで決まります。

デフォルトで、オーサリングアプリケーションが全てのサウンドをメインのサウンドカードにルーティングするので、自分のサウンドデザインを聞くことができます。一方、テスト用に様々なハードウェアデバイスを選択することも可能で、Audioメニューで???ウインドウを表示させて選択します。

### Master-Mixer Hierarchyにバスを追加する

オーディオルーティング構造を作成するには、Master Audio Busに複数のAudio Busを追加します。また、ほかのアウトプット用にミキシングするために、別のマスターバスも作成できます。Motionを使う場合は、モーションデバイス用のマスターバスも作成して、全てのサウンド（モーションは低周波オーディオだけ）を、その階層のバスにルーティングするようにします。Master Busの下に子バスを1つ作成できたら、ルーティング構造を構築するために設定する親バスや子バスに個数の制限はありません。

### Master-Mixer Hierarchyに子バスを追加するには:

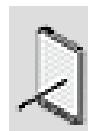
1. Project ExplorerのAudioタブで、新しい子バスを作成したいバスを、右クリックする。
2. メニューで、以下のオプションを1つ選択する。
  - New Child > Audio Bus
  - New Child > Auxiliary Bus

Master-Mixer Hierarchyに新規の子バスが追加される。

### Master-Mixer Hierarchyで親バスを作成するには:

1. Project Explorerの、Audioタブで、新しく作成する親バス（New parent）の、子となるバスを1つ右クリックする。
2. メニューで、以下のオプションを1つ選択する。
  - New Parent > Audio Bus
  - New Parent > Auxiliary Bus

Master-Mixer Hierarchyに新規の親バスが追加される。



#### 注記

Auxバスは、階層の最上位になることができません。そのため、New Parent > Auxiliary Busという設定が、この場合は指定できなくなります。

### Master-Mixer Hierarchyに新しいマスターバスを追加するには:

1. Project ExplorerのAudioタブで、Master-Mixer HierarchyのDefault Work Unitを右クリックする。

2. メニューで、以下のオプションを1つ選択する。

- New Child > Audio Bus

Master-Mixer Hierarchyに新規のマスターバスが追加される。

## 関連トピック

- 「Master-Mixer Hierarchyにあるバスの移動」
- 「Master-Mixer Hierarchyにあるバスの削除」

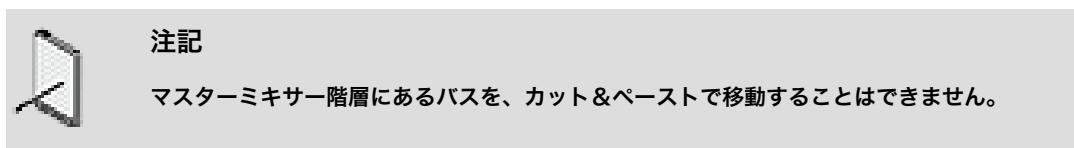
### Master-Mixer Hierarchyにあるバスの移動

複数のバスを追加した後に、1つのバスの位置を変えて、バス同士の関係を変更できます。

Master-Mixer Hierarchy内で、バスを移動するには:

1. バスを移動先までドラッグする。

このバスと、その下にある全ての子が、新しい場所に移動される。場所が変わることで、移動したバスは、新しく親となったバスのプロパティの影響を受ける。



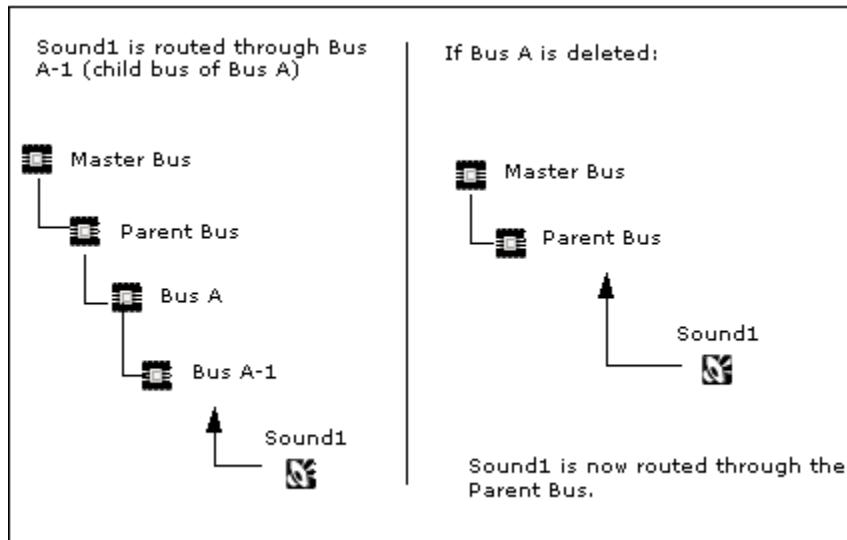
## 関連トピック

- 「Master-Mixer Hierarchyにバスを追加する」
- 「Master-Mixer Hierarchyにあるバスの削除」

### Master-Mixer Hierarchyにあるバスの削除

誤って作成したバスや不要となったバスは、削除できます。バスを削除すると、その子バスも全て削除されます。削除したバスにルーティングされていたオブジェクトは全て、階層の次の親オブジェクトに改めてアサインされます。ルーティング設定が変更されたオブジェクトに設定されていたプロパティのオーバーライド設定は、そのまま維持されます。

バスを削除した時に、そのバスにルーティングされていたサウンドオブジェクトへの影響を下図に示します。



### Master-Mixer Hierarchyにあるバスを削除するには:

- Project Explorerの、Audioタブで、削除するバスを1つ選択して、以下のいずれかを行う。
  - Deleteキーを押す。
  - このバスのショートカットメニューで、Delete Selectionを選択する。
- このバスとその子が全て削除され、削除されたバスにルーティングされていたオブジェクトは全て親バスにルーティングされます。

### 関連トピック

- 「Master-Mixer Hierarchyにバスを追加する」
- 「Master-Mixer Hierarchyにあるバスの移動」

## バスのプロパティ設定

バスのプロパティ設定を利用して、ゲームのオーディオやモーションに対してグローバル（プロジェクト全体を対象）に変更を適用できます。バスのプロパティ設定で、以下を行えます。

- 「既存のサウンドから、モーションを生成」
- 「バスの相対プロパティを設定する」
- 「信号をダッキングする」
- 「ゲームミュージックを、プレイヤーのミュージックに置き換える」

バスは制御設定の最終レベルであるため、変更を行うとその下にある全てのオブジェクトが影響を受けます。

オブジェクトの場合と同様に、バスに対してもエフェクトの適用、RPCの使用、Stateのアサイン、詳細プロパティの設定ができます。詳しくは、以下のセクションを参照してください。

- 「バスにエフェクトを適用」
- 「Game Parameterを使って、プロパティ値をコントロールする」
- 「オブジェクトやバスにStateをアサインする」
- 12章Priorityの管理

### バスの相対プロパティを設定する

相対プロパティは、階層のバスごとに設定します。相対プロパティは累積されるので、1つのバスのプロパティ値はその下にある子オブジェクトに加算されます。

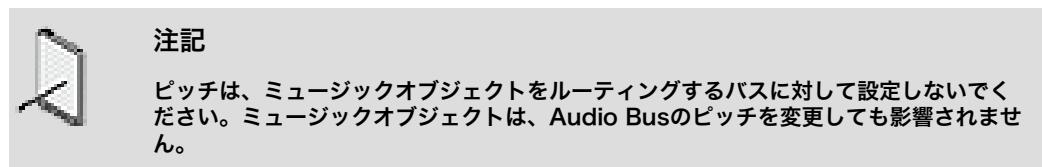
バスに設定できる3つの相対プロパティは、以下の通りです。

- Bus Volume:** バスに直接適用される減衰。
- Voice Volume:** バスの中で再生中のオーディオオブジェクトに適用される減衰。
- Voice Pitch:** バスの中で再生中のオーディオオブジェクトの再生スピード。

Multi Editorで、複数のバスの相対プロパティを同時に編集できます。Project ExplorerのAudioタブで複数のバスを選択してから、ショートカットメニューでShow in Multi Editorを選択します。また、ゲームに接続した状態で、リアルタイムにこれらのプロパティの微調整も行えます。

### バスの相対プロパティを設定するには

- Property Editorにバスを1つロードする。
- 数値を入力するか該当スライダをドラッグして、相対プロパティ値を設定する。



### 関連トピック

- 「バスにエフェクトを適用」
- 「信号をダッキングする」
- 「ゲームミュージックを、プレイヤーのミュージックに置き換える」
- 「Master-Mixer Consoleを使って、ミックスを最終調整する」

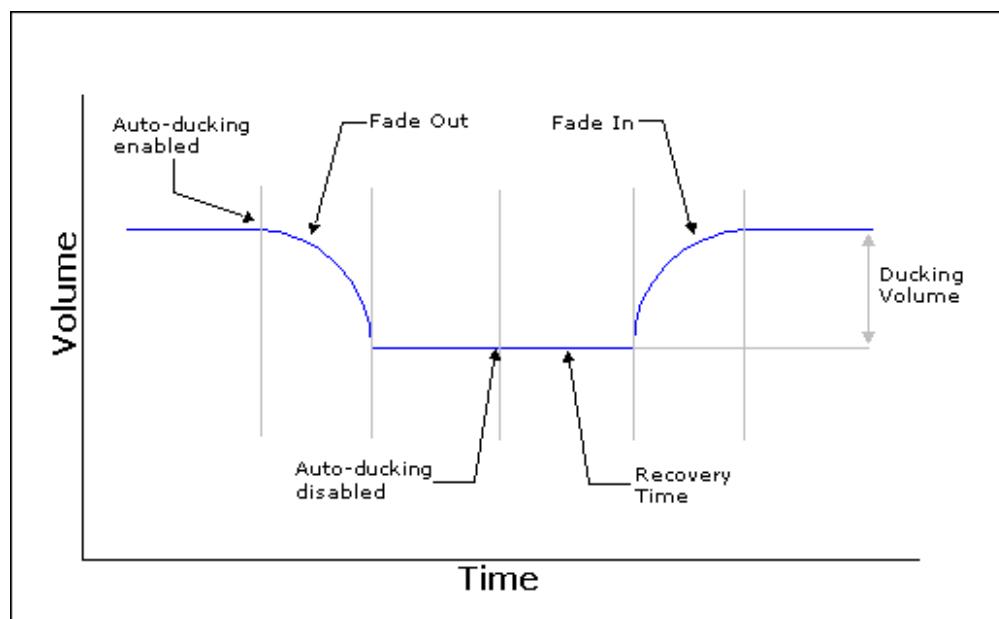
### 信号をダッキングする

ゲーム中に、タイミングによって特定のオブジェクトを他より目立たせたい状況もあります。例えば、キャラクターがゲーム中に話している時はミュージックの音量を下げるかもしれません。ダッキング (Ducking) 機能を使えば、ある信号が他の信号と比較した時に重要かどうかを設定できます。ダッキングを使うと、1つのバスが目立つように、同じタイミングで別のバスを通る全てのオブジェクトのボリュームレベルが自動的に下がります。

信号のダッキングの仕方をコントロールするには、以下のプロパティや動作設定を調節します。

- Ducking volume
- Fade out
- Fade in
- Curve shape
- Recovery time
- Maximum ducking volume

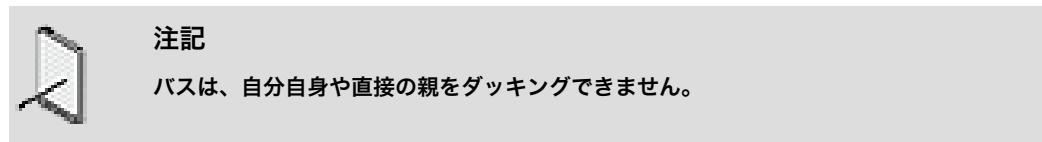
プロパティ値や動作設定がダッキングされる信号に与える影響を、下図に示します。



### 信号をダッキングするには:

1. Property Editorにバスを1つロードする。
2. グループボックスAuto-duckingのInsertをクリックする。

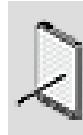
Project Explorer - Browserが開き、ダッキングできる全てのバスのリストが表示される。



3. 現在のバスが信号を受信した時にダッキングさせるバスを1つ、リストから選択する。
4. OKをクリックする。

選択したバスがDucking listに追加される。

5. ダッキングされるバスのプロパティや動作を定義するために、以下の設定を調節する。
  - **Volume** - 現在のバスが信号を受信した時の、選択したバスのボリューム減少量。
  - **Fade Out** - 元のボリュームからダッキングしたボリュームまでの、フェードアウトにかかる時間。
  - **Fade In** - 元のボリュームに戻るまでの、フェードインにかかる時間。
  - **Curve** - フェードアウトやフェードインの仕方を決めるカーブ形状。
6. テキストボックス**Recovery time**に、現在のバスの信号の終了から、ダッキングされた複数の信号のフェードイン開始までの時間を指定する。
7. テキストボックス**Maximum ducking volume**に、現在のバスが1つ以上のバスによってダッキングされてボリュームが減衰する時の減衰最大量を指定する。



### 注記

ダッキングは、Auxiliary Busで使用できません。

## 関連トピック

- 「バスにエフェクトを適用」
- 「バスの相対プロパティを設定する」
- 「ゲームミュージックを、プレイヤーのミュージックに置き換える」
- 「Master-Mixer Consoleを使って、ミックスを最終調整する」

## ゲームミュージックを、プレイヤーのミュージックに置き換える

Xbox One™、iOS、Android™、PlayStation® 4の各プラットフォームで、プレイヤーはゲームのミュージックを自分の音楽に置き換えられます。どのプラットフォームの場合も、ユーザのミュージックがスタートした時点でミュートする全てのバスの**Mute for Background Music**オプションを有効にしておく必要があります。複数のバスを選択でき、ミュージックバス以外も選択できます。

### プラットフォームのバックグラウンドミュージックオプションに、バスをアサインするには:

1. Property Editorに、Audio Busを1つロードする。
2. **Mute for background music**オプションを選択する。

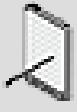
ユーザがコンソールのミュージックプレイヤを通して自分のミュージックをスタートさせると、このバスはミュートされる。

### プラットフォーム別の設定

**Mute for background music**を有効にした時の動作はプラットフォームによって多少異なります。また、サウンドエンジン初期化の時点で追加プログラミングが必要となります。

- **Android** : Mute/Unmuteアクションは、ユーザがミュージックプレイヤーのアプリケーションからゲームに切り替えた時に限り実行されます。つまり、ユーザのミュージックが独自に終了した場合はUnmuteアクションがありません。
- **iOS** : サウンドエンジンの初期化設定でAudioSessionのMixOtherフラグが設定されている場合は、Mute/Unmuteアクションはユーザがミュージックプレイヤーのアプリケーションからゲームに切り替えた時に限り実行されます。つまり、ユーザのミュージックが独自に終了した場合はUnmuteアクションがありません。iOS 8以降でAVAudioSessionCategoryAmbientカテゴリを使用している場合、全てのアプリケーションによるオーディオ中断でゲームミュージックのミュートやアンミュートが実行されます。
- **Xbox One 及び PS4** : メーカが追加したDVR機能を使い、ユーザが自分のゲームを録画して公開できる。そこで、ゲームオーディオの一部でありユーザーが置き換えることのできるゲーム音楽の著作権について、法的な問題が発生します。ゲームスタジオ側は音楽をゲームに使う権利を有しているものの、いかなる形式であれエンドユーザーが配信する権利はありません。このため、TCRは、バックグラウンドミュージックの録音を禁止しています。対策としてCPU負荷が最も低いのは、ゲームの音楽を他と分けてミックスすることです。これは、セカンドリアウトプット機能で行います。

Xbox OneとPS4は、録音してはいけない音を送るためのバーチャルデバイスを提供しています。このDVR-bypass Audio Device用に別のミックスを設定するには、新しいマスターバスを作成します。次に、作成した新しいバスのAudio Deviceプロパティを変更して、DVR-bypassデバイスにポイントするようにします。これで、音を通常通り新しいバスや子バスにルーティングできます。詳細は「[セカンドリアウトプットを理解する](#)」を参照してください。



### PS4に関する重要事項

Mute for Background Music機能を使う場合は、セカンドリアウトプットをコードで追加する必要があります（[AddOutput](#) を使用）。この出力が使われていない場合は、ユーザー音楽がスタートしたことをPS4システムは通知しません。

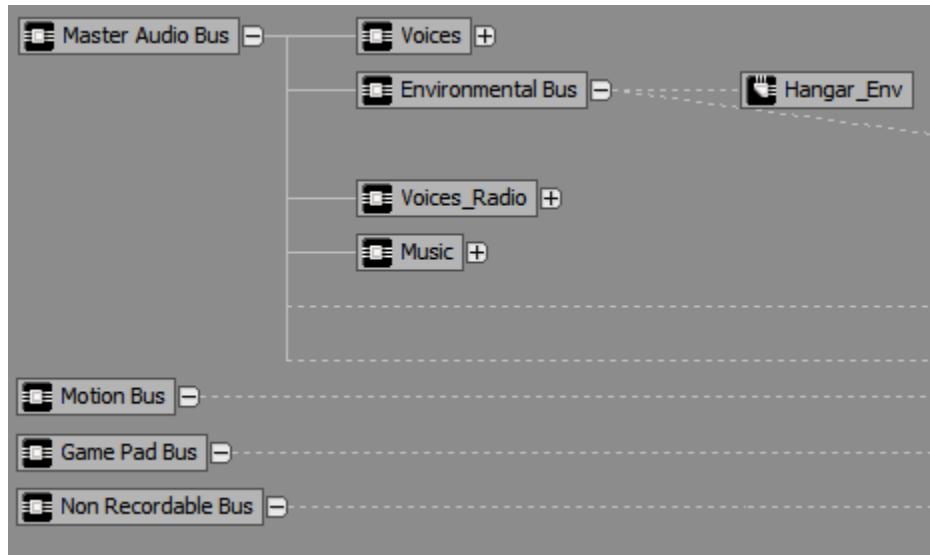
## 関連トピック

- 「[バスにエフェクトを適用](#)」
- 「[バスの相対プロパティを設定する](#)」
- 「[信号をダッキングする](#)」
- 「[Master-Mixer Consoleを使って、ミックスを最終調整する](#)」
- 「[オブジェクトのアウトプットルーティングの指定](#)」
- 「[組み込みオーディオデバイス](#)」
- [セカンドリアウトプットをインテグレートする](#)

## バス階層の構築例

Wwiseで、バス階層の構造を柔軟に設計できます。つまり、プロジェクトのサウンドを整理する上で、絶対に正しい構造というものはありません。とはいえ、どのようなアセットや要件の組み合わせが入っているプロジェクトでも一般的に共通するような方法を、以下の簡単な事例で紹介します。

以下のスクリーンショットはSchematic Viewのもので、このプロジェクトでは、マスター・オーディオバスの下に4つのオーディオバスがあり、セカンダリマスター・バスが3つあります。



Master Audio Busには、以下のバスがあります:

- **Environmental Bus** - プレイヤーに聞こえてくるであろう様々なサウンドを環境的な要因（リバーブなど）によって分類したバスで、例えば砂利、木板、セメントの地面別にフットステップ（プレイヤーや他のキャラクターの足音）を分けます。
- **Music**: 全てのミュージックをまとめたバスで、ゲーム中の特定場面でプレイする時のミュージックや、ゲーム外のUIメニューを操作中のミュージックなどが含まれます。
- **Voices**: キャラクターダイアログのほぼ全てをまとめたバス。
- **Voices\_Radio** - このゲームではダイアログが頻繁に発生し、無線での会話を表現する特殊設定が必要なボイスも多いため、Voiceバスとは別にこのバスも用意しました。ボイスの整理の仕方として、Voiceバスの下に子バスとして設定する方法もあります。この2つの方法は考え方として似ていますが、求められるサウンドアウトプットを実現するためのミキシングを考慮すると、別のバスをMaster Audio Busのすぐ下に定義する方が効率的となりました。

広大な航空機ハンガー内にいる時のサウンドを調整しやすくするために、Auxiliary Busとして**Hangar\_Env**も設定しました。これを使えば、ゲームシーンがハンガー内に移動した場合に、サウンドをマニュアル操作またはゲームコールでこのバスに送り、ハンガー環境で聞こえるであろうオープンエコーを連想させるリバーブを適用できます。

次に、追加されたマスター・バスがあり、それぞれ送り先が別のセカンダリ・アウトプットです:

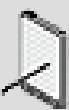
- **Motion Bus** - 全てのモーションシグナル（別名ランブル）を受信するバスで、適切なコントローラにそれをアウトプットします。この例では子バスがありませんが、モーションでも複雑なミキシングを設定することができます。

- **Non Recordable Bus** - システムによっては、DVR (Digital Video Recorder) に録音されない別のアウトプットがあります。プレイヤーによる録音で公に配布してはならない著作権のある音楽などに、合理的です。音（または音楽）をこのバスにルーティングすると、録音されません。
- **Game Pad Bus** - コントローラのスピーカーのことで、ここからアウトプットされる音を、プレイヤーは直接に、または少なくともはつきりと、聞こえます。キャラクターの頭が壁にぶつかる衝突音などに理想的なスピーカーです。



### Tip

>同じサウンドを同時に、セカンダリバスとオーディオバスの両方に送ることは推奨されません！どのシステムも多少のレイテンシが発生します。ほんの数ミリ秒であっても、同じサウンドが遅れて別々の出力源から聞こえてくると耳障りです。



### 注記

ミュージックファイルなどのサウンドオブジェクトから、モーションデータを生成できます。既存のサウンドからモーションを生成する方法については「[既存のサウンドから、モーションを生成](#)」を参照してください。

**audio**kinetic

## パート III. サウンドやモーションでゲームを強化する



---

10. オブジェクトの再生方法の設定 .....	207
概要 .....	208
オブジェクトの再生方法の設定 .....	208
Random Container、Sequence Containerの再生方法の設定 .....	213
Switch Containerの中身と、再生方法の設定 .....	225
Blend Containerの中身と、再生方法の設定 .....	230
オブジェクト再生に関する、コツとベストプラクティス .....	241
11. Positioningの定義 .....	243
Positioning定義の概要 .....	244
WwiseのPositioningを理解する .....	245
2Dオブジェクトの活用 .....	247
3Dオブジェクトの活用 .....	249
BusにPositioningを適用 .....	252
3Dポジショニングのイメージ図 .....	253
距離で変化する減衰設定 .....	265
アニメーションパスを利用した、空間ポジショニング .....	278
センタースピーカーの、オーディオ信号のルーティング .....	289
Positioningに関する、コツとベストプラクティス .....	289
チャンネルコンフィギュレーションを理解する .....	296
スピーカーとヘッドフォンのパンニングルール .....	298
Ambisonicsを使う .....	300
12. Priorityの管理 .....	306
概要 .....	307
オブジェクトのプライオリティを理解する .....	309
オブジェクトの再生インスタンスの制限 .....	312
再生Priorityの設定 .....	316
低ボリュームのオブジェクトの管理 .....	318
プライオリティに関する、コツとベストプラクティス .....	320
13. エフェクトの活用 .....	322
概要 .....	323
エフェクトの使用 .....	323
環境音響を実現する、エフェクトの使用 .....	331
エフェクトに関する、コツとベストプラクティス .....	335
14. モーションの活用 .....	336
概要 .....	337
ゲームのモーションを作成 .....	339
モーション用に、アウトプット構造の構築 .....	345
モーションに関する、コツとベストプラクティス .....	346

---

## 第10章 オブジェクトの再生方法の設定

概要 .....	208
オブジェクトの再生方法の設定 .....	208
Random Container、Sequence Containerの再生方法の設定 .....	213
Switch Containerの中身と、再生方法の設定 .....	225
Blend Containerの中身と、再生方法の設定 .....	230
オブジェクト再生に関する、コツとベストプラクティス .....	241

## 概要

再生方法は、ゲームの状況に応じて変える必要があります。様々なシナリオに対応するために、Wwiseではアクターミキサー階層のオブジェクトの再生方法を、個別に設定することができます。また、再生をさらに自由に幅広くコントロールするために、複数のオブジェクトを、以下の種類のコンテナにまとめ、再生させることができます。

- ・「Random Containerの作成」
- ・「Sequence Containerの作成」
- ・「Switch Containerの中身と、再生方法の設定」
- ・「Blend Containerの作成」

ここで使うコンテナの種類によって、中に入れたオブジェクトの再生方法が決まります。具体的には、Random Containerはコンテナ内のオブジェクトをランダムに再生、Sequence Containerはコンテナ内のオブジェクトをプレイリストに従って再生、Switch Containerはコンテナ内のオブジェクトをゲーム内のSwitchまたはStateに基づいて再生、Blend Containerはブレンド設定やRTPCを利用しながらコンテナ内の全オブジェクトを同時に再生します。

異なる種類のコンテナを組み合わせれば、ゲームの様々なシーンに対応した再生方法を効率的かつクリエイティブに管理できるため、最終的にゲームのエクスペリエンスを強化します。

## オブジェクトの再生方法の設定

プロジェクト中にあるオブジェクトには、いくつかの動作（Behavior）が設定されます。オブジェクトの動作設定で、オブジェクトが呼ばれた時の再生回数や、メモリに保存するのかDVD、CD、ハードドライブなどから直接ストリーミングするのか、などが決まります。

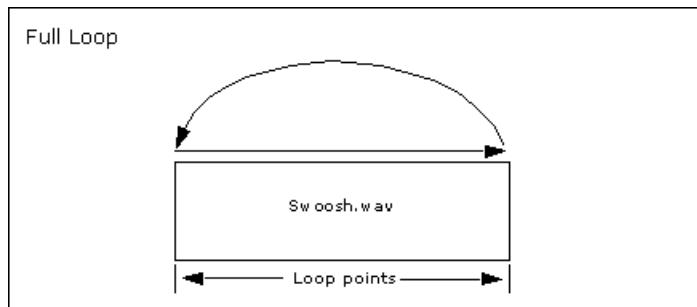
オブジェクトの動作として、以下を設定できます。

- ・「ループの設定」
- ・「メディアのストリーミング」

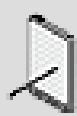
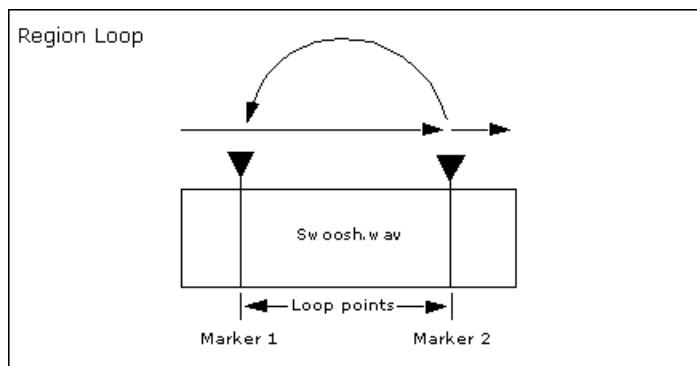
### ループの設定

オブジェクトはデフォルトとして、最初から最後まで1回再生されます。しかし場合によって、オブジェクトを2回以上再生させたいこともあります。そこで、Loop（ループ再生）を設定します。オブジェクトをループさせる時は、まずループの回数を設定します。なおループ再生は、無限にループさせるオプションと、ループ回数を指定するオプションがあります。

ループはデフォルトでオブジェクト全体が再生されますが、オーディオファイルにリージョン（範囲）を指定するマーカーを設定するとオブジェクトの指定された部分だけがループされます。一方、下図のようなFull Loop（完全ループ）の場合は、オブジェクト全体が最初から最後まで再生されます。一旦オブジェクトが最後まで再生されると、また最初に戻って再生されます。その後、指定したループ回数に到達するまで繰り返し再生されます。



また、下図のようなRegion Loopの場合は、マーカーで指定したリージョンだけがループ再生されます。まず、オブジェクトの最初からループリージョンの最後まで再生されます。その後、マークされたリージョンの先頭に戻りループリージョンを再生し、指定したループ回数に到達するまでリージョンの再生を繰り返します。リージョンのループが終了すると、そのオブジェクトの残りの部分が再生されます。ループ回数が無限の場合は、オブジェクトの残りの部分が再生されることはありません。Wwiseが対応できるループリージョンは1つだけなので、オーディオファイル中に設定されたリージョンが1つだけであることを、確認してください。



### 注記

リージョンマーカー（またはAuditionのスタートサンプル、エンドサンプル）は、Adobe® Audition®やSound Forge®など、サードパーティアプリケーションで作成する必要があります。なお、これら以外のサードパーティアプリケーションが使えることもありますが、Wwiseの正式な対応ではありません。

### サウンドをループさせるには:

1. Property Editorに、オブジェクトを1つロードする。
2. Loopオプションを選択する。  
Loopの選択肢が選択可能となる。
3. 以下のオプションを1つ選択する。

**Infinite**（無限）は、オブジェクト全体、またはファイル内で設定したループリージョンを、無限に繰り返す。

**No. of Loops**（ループ回数）は、オブジェクト全体、またはファイル内で設定したループリージョンを、設定した回数だけ再生する。

4. No. of Loopsオプションを選択した場合は、オブジェクトまたはループリージョンを再生させる回数を入力する。

### 関連トピック

- 「オブジェクトの再生方法の設定」
- 「メディアのストリーミング」

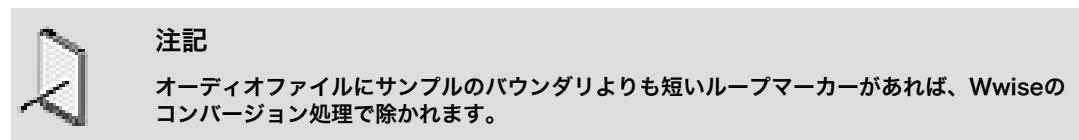
### 圧縮されたオーディオファイルのループ

ADPCMなどの圧縮ファイルフォーマットでは、ファイル長さやリージョンマークーがサンプルバウンダリの位置になる必要があるため、コンバージョン処理でゼロベースのサンプルが、パディングされます。この余分なパディングが、サウンドをループさせる時に問題となることがあります。問題を回避するために、Wwiseのプロンプトが表示されるので、ループ(Loop)するサウンドを再コンバージョンして、以下を確実にします。

- ファイルの終わりでパディングが発生しない。
- ループのリージョンマークーがサンプルバウンダリと合っている。

Wwiseで再コンバージョンを行うと、圧縮形式の要件にファイルが適合するよう、ピッチシフトを使います。ループでサンプルアキュレートの状態が維持され、ファイルのサンプルレートが変わりません。

なお、ループするXMAファイルがWindowsとXbox Oneで多少、音が変わることに気付くかもしれません。これは、Xbox Oneではハードウェアを使ってファイルをデコードするので、ループポイント間でショートフェードを適用できるためです。一方PCでは、XMAエンコーダを使って別の.wavファイルが生成されます。



### ソースエディタを使う

ソースエディタを使って、以下のオブジェクトタイプを編集できます。

- オーディオソース (.wavファイルのカプセル化)
- ソースプラグイン (Wwise Silence、Wwise Sine、Wwise Tone Generatorなど)

### オーディオソースの編集

ソースエディタで、.wavファイルを破壊することなく、以下の操作を行えます。

- ソースの始まりや終わりのトリム
- フェードイン、フェードアウトの適用
- ループポイントの設定
- スムーズなループのための、クロスフェード設定

これらの操作は全て、ゲーム中にリアルタイムではなく、オフラインで処理します。操作はプロジェクトのWork Unitに保存され、「元に戻す」と「やり直す」が可能です。

### オーディオソースのコンテンツをトリムする

.wavファイルのコンテンツをトリムすると、WAVデータを節約できます。トリム操作は、.wavファイルの最初や最後のサイレンスを削除する時に使います。コンテンツのトリムは、ソース素材のコンバージョン時に行われる、非破壊的な処理です。



#### 注記

トリム、ループ、フェードなどの編集機能は、Music Segmentやミュージッククリップのオブジェクトの一部として提供されるため、Interactive Music Hierarchyのオーディオソースオブジェクトには適用できません。

### ソースをトリムするには:

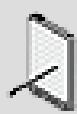
1. Trim Start (波形の左下コーナーにある、青の四角ハンドル) を、右にドラッグする。
2. Trim End (波形の右下コーナーにある、青の四角ハンドル) を、左にドラッグする。

### オーディオソースのループ設定

ループプロパティを有効にしたサウンドオブジェクトに、オーディオソースが入っている時に、ループポイントが使われます。.wavファイルにループポイントがあれば、エディタでループポイントを移動するまで、これがデフォルトとして使われます。エディタでループポイントを変更した時点で、ファイルのループポイントは使用されません。

ループポイントが使用されるのは、以下の条件に限られます。

- **Override file loop point** (ファイルのループポイントをオーバーライド) が有効である。
- 親サウンドオブジェクトが、ループしている。

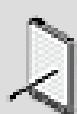


#### 注記

Source Editorでループポイントを設定または変更しても、オリジナルの.wavファイルは変更されません。これは非破壊的な処理です。ループポイントは、プロジェクトに保存されます。

### クロスフェードの長さの編集

ループ時にクリック音が聞こえる場合は、オーディオソースのクロスフェード長さを設定することで、ループの動作をスムーズにします。



#### 注記

ループ時のクリックやポップを回避するために、常にPCMデータのゼロ交差の位置にループポイントを配置することが、推奨されます。

### クロスフェード付きの、ループポイントを調整するには:

1. Loop Start (波形の左上コーナーにある、緑のハンドル) を、右にドラッグする。

2. Loop End（波形の右上コーナーにある、赤のハンドル）を、左にドラッグする。
3. クロスフェード長さを指定する。

### フェードイン、フェードアウトの編集

ソースエディタでフェードハンドル（青の三角）をドラッグして、フェードインやフェードアウトを作成できます。フェードはコンバージョン後のファイルに適用され、ソースを再生する時の追加処理は不要です。

### フェードイン、フェードアウトを作成するには:

1. Fade-inハンドル（波形の左上コーナーにある、青の三角ハンドル）を、右にドラッグする。
2. Fade-outハンドル（波形の右上コーナーにある、青の三角ハンドル）を、左にドラッグする。

### オーディオソースのインテグリティ問題の解決

外部エディタを使って、オーディオソースのプロパティや.wavファイルを編集した場合に、オーディオソースに対して選択した処理が、適用されない、または完全には履行されないことがあります。

### インテグリティ問題を引き起こす、シナリオ例

1. Wwiseに.wavファイルをインポートして、サウンドオブジェクトとオーディオソースオブジェクトを作成する。
2. Source Editorで、.wavファイルの終わりをトリムする。
3. 外部WAVエディタで、.wavファイルの終わりの部分を、エンドトリム位置よりも大きく、削除する。

Wwiseのトリム処理が、.wavファイルの、既に存在しない部分に対する処理となる。これは、オーディオソースのインテグリティ問題として検知される。

### オーディオソースのインテグリティ問題の検知

オーディオソースのインテグリティ問題は、Wwiseの以下の場所で確認できます。

- Source Editorに、「！」付きの黄色い三角マークで表示される。
- オーディオソースのテスト中に、Integrity Reportで確認。
- SoundBank生成時に行われる、オーディオソースのコンバージョン中に確認。

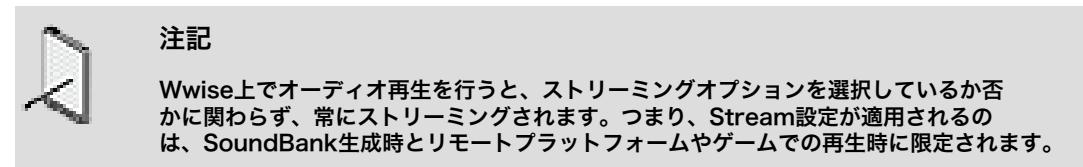
### オーディオソースのインテグリティ問題の修復

オーディオソースのインテグリティ問題を修復するには、以下のいずれかを行います:

- サポートされなかった操作に関して、オーディオソースをマニュアル修復する。
- Audio Source Editorの、黄色の三角マークをクリックする。
- オーディオソースのリストのある画面で、オーディオソースオブジェクト、またはサウンドオブジェクトを右クリックして、Repair Audio Source Integrity Issuesを選択する。

### メディアのストリーミング

オブジェクトは、メモリから再生するものと、ハードドライブやCD、DVDなどからストリーミング (Streaming) するものに分ることができます。また、ディスクやハードドライブからストリーミングするメディアのオプション設定として、再生ディレイを回避するために小さなバッファを作成して、残りのファイルをフェッチするのに必要なレイテンシ時間をカバーすることもできます。この時、ハードドライブ、CD、DVDなどメディアソース側の要件に従ってバッファのサイズを指定します。ストリーミングするファイルの残りの部分と異なり、この小さなバッファ分はSoundBankに保存され、ゲーム中の適切なタイミングでSoundBankと共にメモリにロードされます。



### サウンドをストリーミングするには:

1. Property Editorに、オブジェクトを1つロードする。
2. Streamオプションを選択する。

Streamの選択肢が選択可能となる。

3. オブジェクトがトリガーされた時に全く遅延なく再生するには、Zero Latencyを選択する。ゼロレイテンシを達成するためには、データファイルの最初の一定部分をメモリに保存して、メディアからファイルの残りをフェッチするまでのディレイをカバーする必要がある。
4. テキストボックスPrefetch lengthに、フェッチするためにメモリに保存するデータファイルの長さを、ミリ秒単位で入力する。

### 関連トピック

- 「[オブジェクトの再生方法の設定](#)」
- 「[ループの設定](#)」

## Random Container、Sequence Containerの再生方法の設定

Random ContainerやSequence Containerを使えば、ゲームの再生方法を、様々な方法で整理できます。Sequence Containerは、指定した順序に従つたいくつかのオブジェクトを、一貫して再生する一方で、Random Containerは、複数のオブジェクトからランダムに1つだけ選んで再生し、変化のあるサウンドスケープをつくり出すことでプレイヤーのエクスペリエンスを高めます。

Random ContainerやSequence Containerを使う際は、以下のタスクを行います。

- 「[Random Containerの作成](#)」

- ・「Sequence Containerの作成」
- ・「プレイリストの作成」

### Random Containerの作成

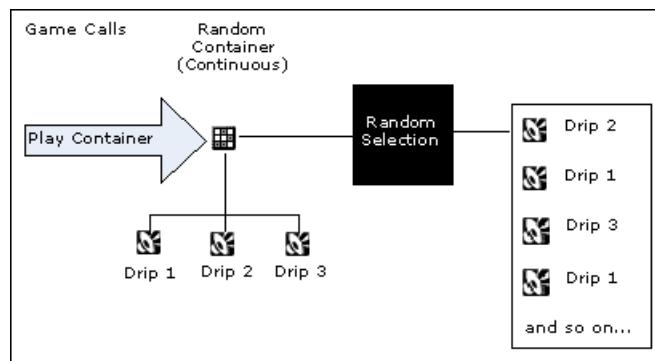
複数のオブジェクトの中から、ランダムに選んで再生するような状況には、Random Containerを使います。例えば、ファイティングゲームのアタック、ヒット、リアクションのサウンドを、全てRandom Containerで再生させることができます。

Wwiseには2種類のランダム再生があり、標準のランダム選択ではコンテナ内のオブジェクトの再生確率が全て等しいのに対し、シャッフル選択では一度再生されたオブジェクトが選択群から外されます。

また、Random Containerの中にあるオブジェクトのウェイト付けも可能です。オブジェクトに付けられたウェイトによって、そのオブジェクトが選択されて再生される確率が増加または減少します。あるオブジェクトに設定されるウェイト付けの数値は、コンテナ内のウェイト付け数値の合計と比較されます。つまり、オブジェクトに設定された数値は、全オブジェクトの数値の合計数と同じ回数を再生した場合の、再生される可能性となります。例えば、コンテナ内のオブジェクトが2つあり、それぞれ1と100のウェイト付けが設定されている場合は、1番目のオブジェクトの再生確率が1/101となり、2番目のオブジェクトの再生確率が100/101となります。

### Random Containerの活用例

例えば、ゲームの洞窟の場面を考えます。洞窟という環境の雰囲気を出すために、バックグラウンドに水滴が垂れるサウンドを追加します。この場合、様々な水滴サウンドをまとめて入れたRandom Containerを作成します。キャラクターが洞窟内にいる時は継続してこのサウンドを再生させたいので、コンテナの再生モードをContinuous（継続）として、ループをInfinite（無限）に設定します。限られた数のサウンドをランダムに再生させることで、リアル感が出ます。



### Random Containerを作成するには:

1. Project Explorerで、Actor-Mixer Hierarchyにある以下のオブジェクトを1つ選択して、右クリックする。
  - Work Unit

- Virtual Folder
- Actor-Mixer
- Blend Container
- Random Container
- Sequence Container
- Switch Container

2. ショートカットメニューで、**New Child > Random Container**を選択する。

新しいRandom Containerが作成され、Actor-Mixer Hierarchyでハイライト表示されます。

3. 新しいRandom Container名を入力して、Enterを押す。
4. 新しいRandom Containerをダブルクリックして、そのProperty EditorとContents Editorを開く。
5. Random Containerにオブジェクトを入れるために、Project ExplorerからContents Editorにドラッグする。
6. グループボックス**Random**で、以下のオプションを1つ選択する。
  - Standardでは、オブジェクト群が維持される。1度再生されたオブジェクトも、これから再生される可能性のあるオブジェクトのリストから外されないため、再び再生されることがある。
  - Shuffle（シャッフル）では、再生されたオブジェクトがオブジェクト群から外される。全てのオブジェクトが再生されるまで、再び再生されない。
7. 同じオブジェクトが続けて再生されるのを避けるには、**Avoid repeating last x played**（最後のx個は繰り返さない）オプションを選択する。

このオプションを選択した場合の動作は、StandardモードとShuffleモードで異なる。

Standardモード中は、再生するオブジェクトを完全にランダムに選択するものの、最後に再生したx個のオブジェクトを、再生リストから外す。

Shuffleモード中は、リストをリセットする時に、最後に再生したx個のオブジェクトをリストから外す。

8. テキストボックス**Avoid repeating last x played**に、同じオブジェクトを再び再生するまでの間に、他のオブジェクトを何個、再生させるかを、入力する。
9. Contents Editorで、コンテナ内の各オブジェクトのウェイト付けをする。

ウェイト付けで、オブジェクトの優先順位が決まります。あるオブジェクトに設定した数値は、全体の合計数の中でそのオブジェクトが再生される可能性のある回数を表します。

## 関連トピック

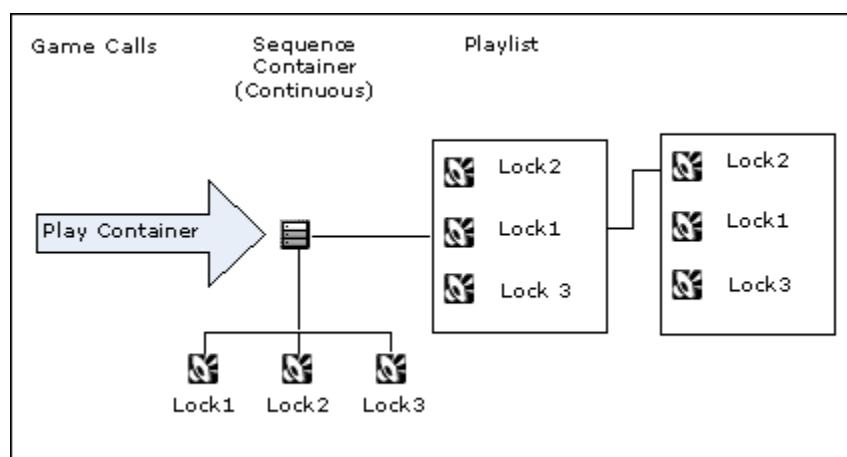
- 「コンテナのScopeの設定」
- 「コンテナ内のオブジェクトの再生モードの設定」
- 「Sequence Containerの作成」

## Sequence Containerの作成

複数のオブジェクトを特定の順番に従って再生させるような状況では、Sequence Containerを使います。Sequence Containerは、指定されたプレイリストの順番に従って中のオブジェクトを再生します。あるキャラクターの台詞（ダイアログ）などをSequence Containerで再生できます。例えばキャラクターが「Hello」という前に「Goodbye」と言わないように、順番を指定できます。

## Sequence Containerの活用例

ファーストパーソンシューティングゲームの事例を考えます。ゲーム中にプレイヤーがボタンを押して巨大な鉄扉を開き、多数の解錠の仕掛けが続く場面があるとします。この場合、様々な解錠サウンドをまとめたSequence Containerを作成します。そしてこれらのサウンドを論理的な順序に並べたプレイリストを作成します。扉の解錠が進むについて解錠サウンドを次から次に鳴らすので、コンテナの再生モードはContinuous（継続）とします。



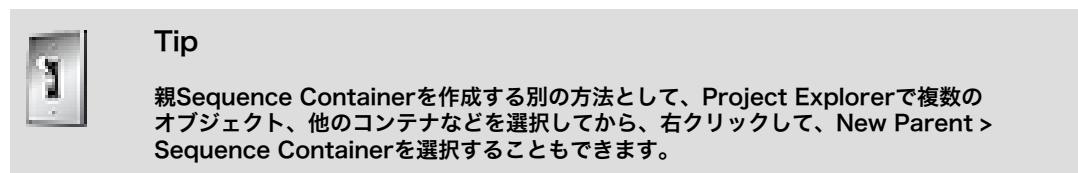
## Sequence Containerを作成するには:

1. Project Explorerで、Actor-Mixer Hierarchyにある以下のオブジェクトを1つ選択して、右クリックする。
  - Work Unit
  - Virtual Folder
  - Actor-Mixer
  - Blend Container
  - Random Container
  - Sequence Container
  - Switch Container
2. ショートカットメニューで、New Child > Sequence Containerを選択する。

新しいSequence Containerが作成され、Actor-Mixer Hierarchyでハイライト表示されます。

3. 新しいSequence Container名を入力して、Enterを押す。

4. 新しいSequence Containerをダブルクリックして、そのProperty EditorやContents Editorを開く。
5. Sequence Containerにオブジェクトを入れるために、Project ExplorerからContents Editorにドラッグする。
6. At end of playlist (プレイリスト終了後) の動作を設定するために、以下のオプションを1つ選択する。
  - Restart (再スタート) は、プレイリストの最終オブジェクトを再生した後に、リストの元々の順番に従い、再び最初から最後まで再生する。
  - Play in reverse order (逆順に再生) は、プレイリストの最終オブジェクトを再生した後に、リストを最後から最初まで逆順に再生する。

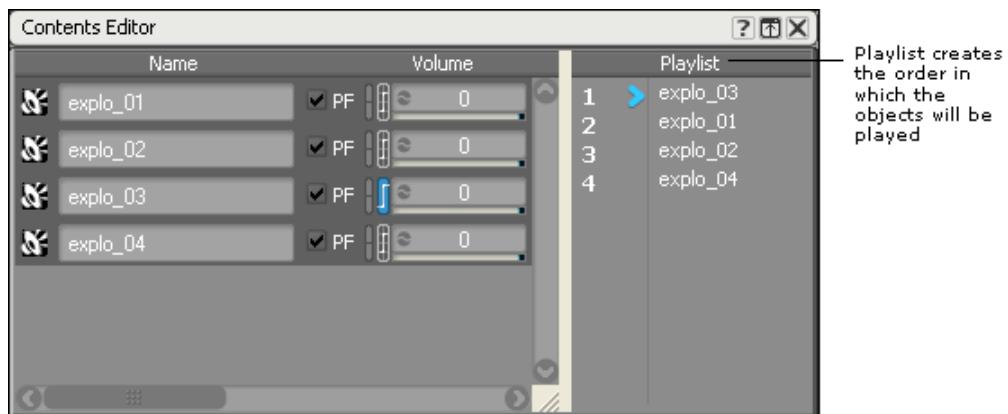


### 関連トピック

- 「プレイリストの作成」
- 「コンテナ内のオブジェクトの再生モードの設定」
- 「コンテナのScopeの設定」
- 「Random Containerの作成」

### プレイリストの作成

Sequence Containerを作成すると、Contents Editorに、Playlistペインが追加される。ここでプレイリストを作成して、コンテナ内のオブジェクトの再生順序を設定できます。



自由にプレイリストでオブジェクトの追加、削除、並べ替えをしながら、調整を重ねて完成させます。また、作成中にいつでもプレイリストを再生できます。プレイリストの再生については「[プレイリストの再生](#)」を参照してください。



### 注記

Sequence Containerは、プレイリストが作成されるまで再生できません。

### プレイリストにオブジェクトを追加、削除

プレイリスト (Playlist) は簡単に作成でき、Contents Editorの、Playlistペインでオブジェクト、他のコンテナなどを追加したり削除します。早く作業するためには、複数のオブジェクトを選択して、一斉に追加または削除することも可能です。

多くの場合、Sequence Containerの中にあるオブジェクトを使ってプレイリストを作成することになります。しかし、Project ExplorerのAudioタブから直接、オブジェクトをプレイリストにドラッグすることも可能です。この操作で、オブジェクトが現在の場所からSequence Containerに移動します。なお、オブジェクトの移動ではなくコピーを作成する場合は、Ctrlキーを押しながら、Project ExplorerのAudioタブからオブジェクトをプレイリストにドラッグします。

### プレイリストのオブジェクトを追加、削除するには:

1. Property Editorに、Sequence Containerを1つロードする。

コンテナ内のオブジェクトが、Contents Editorに表示される。

2. プレイリストにオブジェクトを追加するには、ContentsペインからPlaylistペインに、オブジェクトをドラッグする。

オブジェクトがプレイリストに追加される。

3. プレイリストからオブジェクトを削除するには、Playlistペインで削除するオブジェクトをクリックする。

4. Deleteキーを押す。

オブジェクトがプレイリストから外されるが、Contentsペインには残る。

### 関連トピック

- ・「[プレイリストの作成](#)」
- ・「[プレイリストのオブジェクトの順番の変更](#)」
- ・「[プレイリストの再生](#)」

### プレイリストのオブジェクトの順番の変更

プレイリスト (Playlist) を作成する時に、オブジェクトの順番を変えながら、聞こえ方や感じ方の違いを試せます。なお、プレイリストにコンテナが入っている場合、それらのコンテナ内にネスト化されたオブジェクトの順番は変えられません。

### プレイリストのオブジェクトの順番を変更するには:

1. Contents EditorのPlaylistペインで、移動したいオブジェクト（単数または複数）を選択する。

2. プレイリストの新しい位置に、オブジェクトをドラッグする。

オブジェクトが新しい位置に移動する。

### 関連トピック

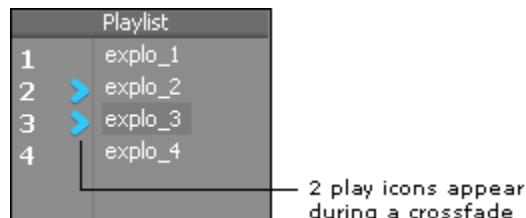
- 「[プレイリストの作成](#)」
- 「[プレイリストにオブジェクトを追加、削除](#)」
- 「[プレイリストの再生](#)」

### プレイリストの再生

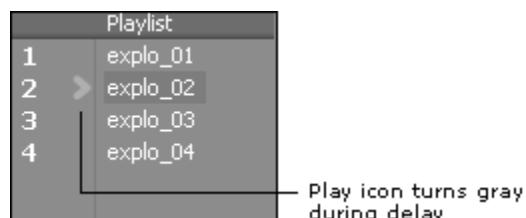
Sequence Containerを再生するには、まずプレイリスト (Playlist) を作成する必要があります。再生中に、プレイリストの青い再生アイコンが再生中のオブジェクトを示します。



オブジェクト間にクロスフェードが設定されている場合は、クロスフェード中、プレイリストに2つの再生アイコンが表示されます。



オブジェクト間でディレイが設定されている場合は、ディレイ中は再生アイコンが青からグレーに変わります。



### プレイリストを再生するには:

1. Property Editorに、Sequence Containerを1つロードする。
2. Transport Controlの、Playアイコンをクリックする。

指定された順番で、Wwiseがプレイリストを再生する。

## 関連トピック

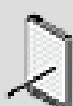
- ・「プレイリストの作成」
- ・「プレイリストにオブジェクトを追加、削除」
- ・「プレイリストのオブジェクトの順番の変更」

## コンテナ内のオブジェクトの再生モードの設定

Random ContainerやSequence Containerに複数のオブジェクトが入るため、その再生モード（Playback mode）を指定する必要があります。Wwiseで設定できる再生モードは、以下の通りです。

- ・ **Step** - コンテナが再生される度に、コンテナ内のオブジェクトを1つだけ再生する。
- ・ **Continuous** - コンテナが再生される度に、コンテナ内の全オブジェクトを一通り再生する。

再生モードをContinuous（継続）とした場合は、オブジェクトをループさせるオプションや、コンテナ内の様々なオブジェクト間にトランジションを設定するオプションもあります。



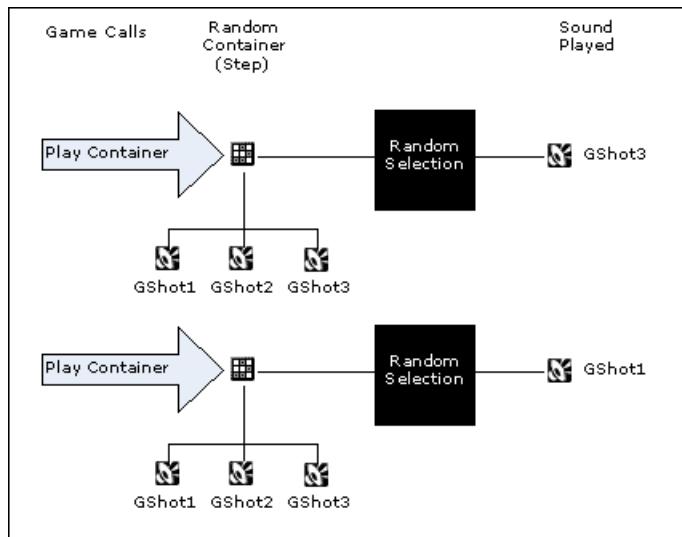
クロスフェードトランジションは、Blendと Switch Containerには適用されません。

Random Container や SequenceContainerとは異なり、Blend Container と Switch Container は、常に変化する可能性のある Game Parameter に関連付けられているため、期間が不明です。そのため、クロスフェード（Xfade）トランジションは、Random ContainerまたはSwitch Containerで設定され、子Blend Containerまたは子Switch Containerでは機能しません。

### コンテナ内のオブジェクトを、1つ再生

場面によって、コンテナ（Container）を呼び出した時に、コンテナのオブジェクトを、1つだけ、再生させることができます。例えば、拳銃を1回発砲した時に1つのサウンドだけを再生する場面や、キャラクターが1回話す時に台詞を1つだけ言わせる場面などです。このような状況で設定する再生モードが、Stepモードです。

拳銃のサウンドをStepモードのRandom Containerに入れ、拳銃を発砲した時の流れを、下図に示します。



コンテナ内のオブジェクトを、1つ再生するには:

1. Property Editorに、Random ContainerまたはSequence Containerを1つロードする。
2. グループボックスPlay Modeで、Stepオプションを選択する。

コンテナが再生される度に、1つのオブジェクトだけが再生される。

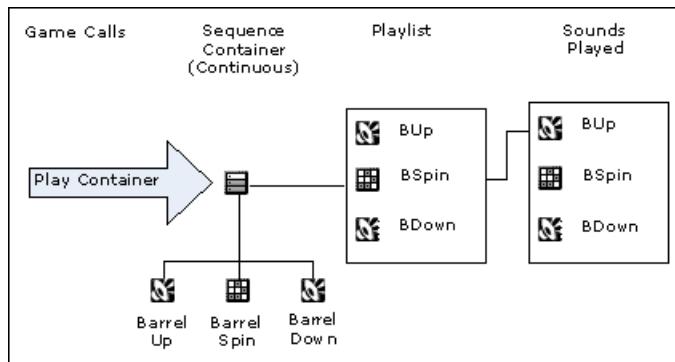
## 関連トピック

- 「コンテナ内のオブジェクトを、全て再生」
- 「Random Container、Sequence Containerの再生方法の設定」
- 「Random Containerの作成」
- 「Sequence Containerの作成」
- 「プレイリストの作成」
- 「コンテナのScopeの設定」

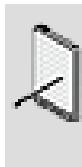
## コンテナ内のオブジェクトを、全て再生

別の場面では、コンテナ (Container) を呼び出した時に、コンテナ内の全てのオブジェクトを再生させることができます。例えば、ゲームで特殊な銃を発砲して、一連のサウンドを立て続けに再生させる場面などです。このような状況で設定する再生モードが、Continuousモードです。

Continuousモードに設定したSequence Containerを使い、特殊な銃が発砲された時に、一連のサウンドをWwiseが次々と再生する流れを、下図に示します。



Continuousモードではコンテナ内のオブジェクトを全て再生するため、ループを設定できるほか、オブジェクト間に様々なトランジションを設定できます。



### 注記

Random Containerのモードが、Continuousに設定されている場合、リストが1回、完全に再生されるまで、何回も繰り返されるオブジェクトもあります。これは、そのコンテナに入っている各オブジェクトに設定されたウェイト付けに起因します。

### コンテナ内のオブジェクトを、全て再生するには:

1. Property Editorに、Random ContainerまたはSequence Containerを1つロードする。
2. グループボックスPlay Modeで、Continuousオプションを選択する。

Continuousの選択肢が選択可能となる。

3. Sequence Containerにおいて、コンテナが再生される度にプレイリストを頭に戻すには、**Always reset playlist**（常にプレイリストをリセットする）オプションを選択する。このオプションの選択を外すと、コンテナは前回ストップした位置から、つまりプレイリストの次のオブジェクトの最初から、再生を続ける。
4. コンテナの全てのコンテンツをループさせるには、Loopオプションを選択する。

Loopの選択肢が選択可能となる。

5. 以下のオプションを1つ選択する。
  - **Infinite**（無限）を選択すると、コンテナが無限に繰り返される。
  - **No. of Loops**（ループ回数）を選択すると、設定した回数だけコンテナが再生される。
6. No. of Loopsオプションを選択した場合は、コンテナを再生させる回数を、入力する。
7. **Transitions**を選択すると、プレイリストのオブジェクト間にトランジションが適用される。

Transitionの選択肢が選択可能となる。

8. Typeリストの以下のオプションから、1つ選択する。

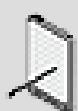
- Xfade (amp) を選択すると、一定振幅を用いたクロスフェードが、2つのオブジェクト間に追加される。



- Xfade (power) ) を選択すると、定電力を用いたクロスフェードが、2つのオブジェクト間に追加される。



- Delayを選択すると、2つのオブジェクト間にサイレンスが追加される。
- Sample Accurateを選択すると、オブジェクト間のレイテンシのない、シームレスなトランジションとなる。なお、オーディオフォーマットのうち、VorbisやXMAはサンプルアキュレートなトランジションとして信頼できるコーデックではない。
- Trigger rate を選択すると、コンテナ内のオブジェクトのTrigger率を指定できる。これは、速い銃撃音の表現などに便利である。

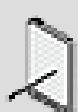


### 注記

トランジションタイプとしてXfade、Sample accurate、またはTrigger rateを選択する場合は、いくつかの制限や制約がある。代表的なものを以下に示します:

- Blend ContainerやSwitch Containerでは、クロスフェード (Xfade) を使えません。Random ContainerやSequence Containerのクロスフェードは、子のBlend ContainerやSwitch Containerが再生されると、失敗します。
- ストップアクションのフェードアウトは、XfadeやTrigger rateのトランジションを使ったコンテナに伝播しません。コンテナに対してStopEventをコールすると、フェードアウト値が再生中の現在のセグメントよりも長い場合は、そのセグメントが終了するとサウンドが急に途切れます。対策として、Set Voice Volumeアクションを使ってフェードを作成して、Stopアクションをディレイさせます。

9. テキストボックスDurationに、クロスフェード、ディレイ、またはTrigger率の時間の長さを入力する。



### 注記

サンプルアキュレートのトランジションでは、Durationオプションを選択できません。

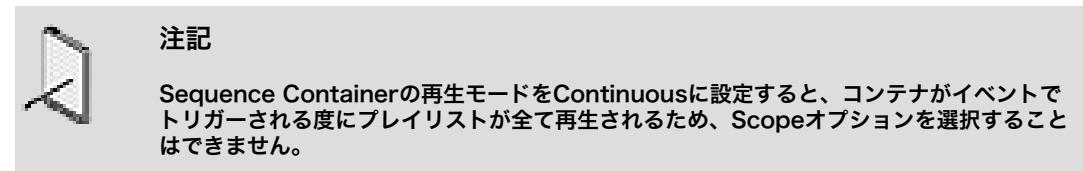
## 関連トピック

- 「コンテナ内のオブジェクトを、1つ再生」
- 「Random Container、Sequence Containerの再生方法の設定」
- 「Random Containerの作成」
- 「Sequence Containerの作成」
- 「プレイリストの作成」
- 「コンテナのScopeの設定」

## コンテナのScopeの設定

同じコンテナに異なるゲームオブジェクトを複数入れることもあるので、このコンテナのゲームで使用する全てのインスタンスを1つのオブジェクトとして扱うのか、各インスタンスを個別に扱うのかを決める必要があります。WwiseではこのコンセプトをコンテナのScopeと呼びます。スコープ設定として、以下のどちらかを選びます。

- **Global** - コンテナの、ゲームで使う全てのインスタンスを1つのオブジェクトとして扱うため、ゲームオブジェクト同士のサウンドやボイスオブジェクトの繰り返しを回避できる。
- **Game object** - コンテナの各インスタンスを個別の要素として扱う。このため、ゲームオブジェクト同士でサウンドがシェアされない。



### コンテナのScopeの設定例

ファーストパーソンのロールプレイングゲームの例を考えます。10人のガードに、同じ30個の台詞をシェアさせます。まず、30個のサウンドボイスオブジェクトをRandom Containerに入れ、ShuffleとStepに設定します。次に1つのコンテナをガード10人用に使い、コンテナのスコープをGlobalに設定することで、守衛が他の守衛と同じ台詞を繰り返す可能性をなくします。このコンセプトは、ゲームで複数のオブジェクトがシェアするようなあらゆるコンテナに適用できます。

### コンテナのスコープを設定するには:

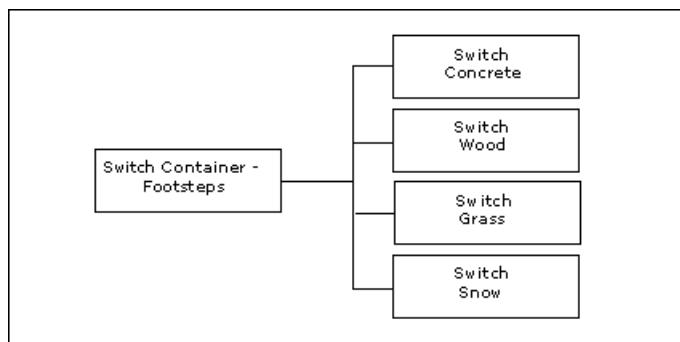
1. Property Editorに、Random ContainerまたはSequence Containerを1つロードする。
2. グループボックスScopeで、以下のオプションを1つ選択する。
  - **Global**は、コンテナの、ゲームで使うインスタンスを全て1つのオブジェクトとして扱う。このため、ゲームオブジェクト同士のサウンドやボイスオブジェクトの繰り返しを回避できる。
  - **Game object**は、コンテナの各インスタンスを個別の要素として扱う。このため、ゲームオブジェクト同士でサウンドがシェアされない。

### 関連トピック

- 「Random Container、Sequence Containerの再生方法の設定」
- 「Random Containerの作成」
- 「Sequence Containerの作成」
- 「プレイリストの作成」
- 「コンテナ内のオブジェクトの再生モードの設定」

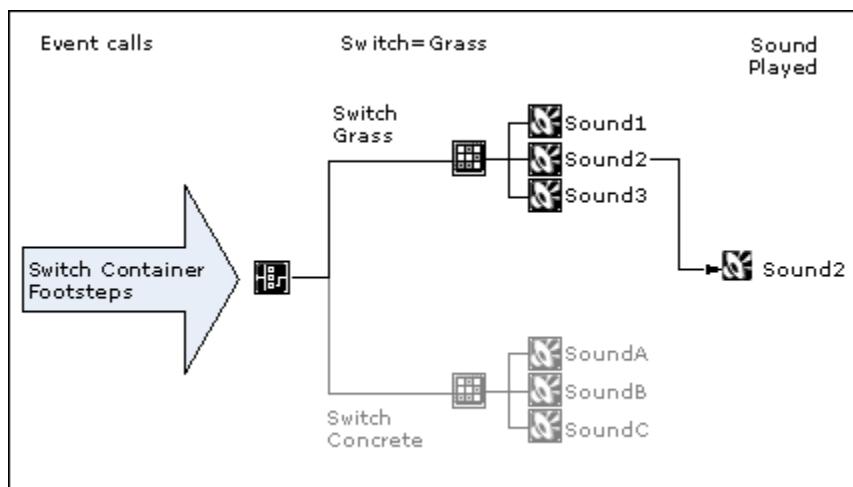
## Switch Containerの中身と、再生方法の設定

Switch Containerを使って、ゲームに存在する様々な選択肢に従い、オブジェクトをグループにまとめることができます。選択肢1つに対して、SwitchまたはStateを1つ、Switch Containerに入れます。例えば、キャラクターが歩く様々な地面の種類に対応するために、Switch Containerを作成できます。この場合、コンクリート、床板、草、雪など、ゲーム中にキャラクターが遭遇するかもしれない、全ての地面素材に対応するSwitchを、コンテナに入れます。



1つのSwitchまたはStateの中に、その選択肢に関連した複数のオブジェクトを入れます。例えば、コンクリートの足音サウンドは全てConcreteSwitchにまとめて入れ、床板の足音サウンドは全てWoodSwitchにまとめて入れます。ゲームがSwitch Containerを呼び出すと、Wwise側で、現在どのSwitch、またはStateが有効かを確認してから、どのコンテナまたはサウンドを再生するかを判断します。

下図は、イベントによって、Switch ContainerFootstepsが呼び出された時の流れを示します。このコンテナには、キャラクターがゲーム中に横切る様々な地面のサウンドが、まとめてあります。今回の例では、GrassとConcreteの2つのSwitchが入っています。下図では、イベントがSwitch Containerを呼び出した時に、キャラクターは草の上を歩いているため（SwitchはGrass）、草の足音サウンドが再生されます。Switch内の足音サウンドはRandom Containerでまとめてあるので、キャラクターが同じ地面を踏む度に、異なるサウンドが再生されます。



## Switch Containerの種類の設定

Switch Containerを作成する際に、そのコンテナが基づくのがStateか、Switchか、またはRTPCかを決める必要があります。なお、SwitchやStateは、Switch Container Property Editorで選択できますが、RTPCをSwitch Groupにリンクさせるには、Switch Group Property Editorで行います。RTPCをSwitchに結びつける方法については「[Switchに、Game Parameter値をマッピングする](#)」を参照してください。

Switchの種類を決めてから、コンテナにSwitch GroupまたはStateやState Groupをアサインする必要があります。これによって、オブジェクトをアサインする対象のSwitch、State、RTPCが決まります。

コンテナにステートグループやスイッチグループをアサインしたり、RTPCをスイッチとして使う前に、これらのステートグループ、スイッチグループ、RTPCを作成する必要があります。作成方法については、以下を参照してください。

- ・「[Stateの活用](#)」
- ・「[Switchの使い方](#)」
- ・「[Game Parameterの作成](#)」

## Switch Containerの種類を設定するには:

1. Property Editorに、Switch Containerを1つロードする。
2. グループボックス**Switch Type**で、以下のオプションを1つ選択する。
  - ・ Switchは、ゲームのスイッチに基づくコンテナとなる。
  - ・ Stateは、ゲームのStateに基づくコンテナとなる。
3. Groupリストから、コンテナにアサインするSwitchまたはStateのグループを選択する。  
Contents EditorのAssigned Objectsペインに、グループ内のSwitchまたはStateが表示される。
4. ゲームでSwitchまたはStateを特定できない場合に、デフォルトとして再生するSwitchまたはStateを、Default Switch/Stateリストから選択する。

## 関連トピック

- ・「[Switch Containerの中身と、再生方法の設定](#)」
- ・「[Switch Containerの再生方法の設定](#)」
- ・「[SwitchやStateの中身の管理](#)」
- ・「[Switch Container内のオブジェクトの再生方法の設定](#)」

## Switch Containerの再生方法の設定

ゲーム中は、いつでもSwitchやStateが切り替わる可能性があるので、サウンドオブジェクトやMotion FXオブジェクトも即時に切り替えるのか、あるいはSwitch Containerが次に再生される時に初めて切り替えるのかを決める必要があります。Wwiseで設定できる再生モードは、以下の通りです。

- Step
- Continuous

Stepオプションは、足音サウンドなど一回だけのサウンドに適しています。一方、Continuousオプションは、スノーボードのサウンドなど、常にループするオブジェクトなどに使うと便利です。

### Switch Containerの再生モードを設定するには:

1. Property Editorに、Switch Containerを1つロードする。
2. グループボックス **Play Mode** で、以下のオプションを1つ選択する。
  - Stepは、再生中にSwitchが変更されたか否かに関わらず、新しい再生イベントがトリガーされた時だけ、新しいオブジェクトを再生する。
  - Continuousは、新しいSwitchまたはStateが検知されたらすぐに、新しいオブジェクトを再生する。Continuousを選択すると、再生するオブジェクトを変更するための新しい再生イベントは必要ない。

### 関連トピック

- 「Switch Containerの中身と、再生方法の設定」
- 「Switch Containerの種類の設定」
- 「SwitchやStateの中身の管理」
- 「Switch Container内のオブジェクトの再生方法の設定」

### SwitchやStateの中身の管理

Switch Container内のオブジェクトは、必ずSwitch (Switch) やState (State) にアサインします。オブジェクトをSwitchまたはStateにアサインする際に、以下が必要なことがあります。

- 「SwitchやStateのオブジェクトの追加、削除」
- 「SwitchやStateのオブジェクトの追加、削除」
- 「SwitchやStateのオブジェクト移動」

### SwitchやStateのオブジェクトの追加、削除

Switch (Switch) にオブジェクトをアサインするのは簡単で、Contents Editor のAssigned Objectsペインに、オブジェクト、他のコンテナを追加するだけです。早く作業するために、CtrlまたはShiftを押しながら複数のオブジェクトをクリックして、一斉に追加または削除することも可能です。なお、Switchに複数のオブジェクトをアサインすると、Wwiseでの再生時やゲームのランタイムに、全て同時に再生されます。

多くの場合、Switch Containerの中にあるオブジェクトをアサインしますが、Project ExplorerのAudioタブから直接、オブジェクトをSwitchまたはStateに、ドラッグすることも可能です。この操作で、オブジェクトが現在の場所からSwitch Containerに移動します。なお、オブジェクトの移動ではなくコピーを作

成する場合は、Ctrlキーを押しながら、Project ExplorerのAudioタブからオブジェクトをAssigned ObjectsペインのSwitchまたはState (State) にドラッグします。

### SwitchやStateにオブジェクトをアサイン、削除するには:

1. Property Editorに、Switch Containerを1つロードする。

コンテナ内のオブジェクトが、Contents Editorに表示される。

2. オブジェクトをSwitchまたはStateにアサインするには、Contentsペインにあるオブジェクトを、Assigned ObjectsペインのSwitchまたはStateにドラッグする。

オブジェクトがSwitchまたはStateに追加される。



#### 注記

オブジェクトを、SwitchまたはStateのタイトル上に直接ドロップする必要があります。

3. SwitchまたはStateからオブジェクトを削除するには、Assigned Objectsペインで削除するオブジェクトをクリックする。

4. Deleteキーを押す。

オブジェクトがSwitchまたはStateから外されるが、Contentsペインには残る。

### 関連トピック

- ・「[SwitchやStateのオブジェクト移動](#)」
- ・「[Switch Containerの中身と、再生方法の設定](#)」
- ・「[Switch Containerの種類の設定](#)」
- ・「[Switch Containerの再生方法の設定](#)」
- ・「[Switch Container内のオブジェクトの再生方法の設定](#)」

### SwitchやStateのオブジェクト移動

オブジェクトを間違ったSwitch (Switch) またはState (State) にアサインしてしまった時や、単純にオブジェクトを移動したい時は、いつでもできます。



#### Tip

複数のオブジェクトを同時に移動するには、Ctrlを押しながら各オブジェクトをクリックして、新しい場所までドラッグします。

### Switch間やState間で、オブジェクトを移動するには:

1. Contents EditorのAssigned Objects ペインで、移動するオブジェクト（単数または複数）を選択する。
2. 新しいSwitchまたはStateまで、オブジェクトをドラッグする。

オブジェクトが新しいSwitchまたはStateにアサインされる。

### 関連トピック

- 「SwitchやStateのオブジェクトの追加、削除」
- 「Switch Containerの中身と、再生方法の設定」
- 「Switch Containerの種類の設定」
- 「Switch Containerの再生方法の設定」
- 「Switch Container内のオブジェクトの再生方法の設定」

### Switch Container内のオブジェクトの再生方法の設定

SwitchやStateはゲーム中に頻繁に切り替わるので、その時にSwitch Containerのオブジェクトがどう反応するべきかを、設定する必要があります。オブジェクトの再生方法を決めるために、以下の項目を設定します。

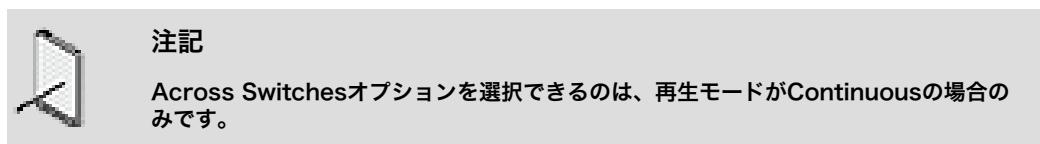
- **Play** - Switch Containerがトリガーされる度に毎回オブジェクトを再生するのか、SwitchまたはStateが切り替わった時だけに再生するのか。
- **Across Switches** - 2つ以上のSwitchに入っているオブジェクトの再生中に、新しいSwitchまたはStateがトリガーされた時、そのオブジェクトを再生し続けるのか。
- **Fade In** - 新しいSwitchまたはStateがトリガーされた時、新しいオブジェクトをフェードインさせるのか。
- **Fade Out** - 新しいSwitchまたはStateがトリガーされた時、再生中のオブジェクトをフェードアウトさせるのか。

### Switch Containerのオブジェクトの再生方法を設定するには:

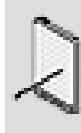
1. Property Editorに、Switch Containerを1つロードする。

コンテナ内のオブジェクトが、Contents Editorに表示される。

2. SwitchまたはStateが切り替わった最初の時だけ、オブジェクトを再生するには、Play列で、**1st only**オプションを選択する。なお、この1st onlyオプションを選択しないと、Switch Containerがゲームにトリガーされる度に、オブジェクトが再生される。
3. SwitchまたはStateが切り替わる時に、ソース（出所）とデスティネーション（行き先）の両方のSwitchまたはStateに入っているオブジェクトを、強制的に再生し続けるには、Across Switches列で、**Continue to play**オプションを選択する。なお、Continue to playオプションを選択しないと、オブジェクトの再生は停止し、再び最初から再生される。



4. SwitchまたはStateが切り替わる時に、新しいオブジェクトをフェードインする場合は、その時間の長さをテキストボックスFade-Inに入力する。



#### 注記

Fade-Inオプションを選択できるのは、再生モードがContinuousの場合のみです。

5. SwitchまたはStateが切り替わる時に、再生中のオブジェクトをフェードアウトさせる場合は、その時間の長さをテキストボックスFade Outに入力する。

#### 関連トピック

- ・[「Switch Containerの中身と、再生方法の設定」](#)
- ・[「Switch Containerの種類の設定」](#)
- ・[「Switch Containerの再生方法の設定」](#)
- ・[「SwitchやStateの中身の管理」](#)

## Blend Containerの中身と、再生方法の設定

Blend Containerを使って、複数のオブジェクトをまとめて、同時に聞こえるように設定できます。また、Blend Containerのブレンドトラックに複数のオブジェクトをまとめることで、それらにRTPCカーブ（単数または複数）を簡単に設定することもできます。さらに、オブジェクト同士がRTPCの変化に応じてクロスフェードする様子を調整できます。これを調整することで、ゲーム内のパラメータが変化すると、オブジェクトもスムーズに切り替わります。

Blend Containerを使って、以下の操作が可能です。

- ・[「Blend Containerの作成」](#)
- ・[「ブレンドトラックを使いこなす」](#)
- ・[「Crossfadeの管理」](#)
- ・[「Blend Containerの中身の試聴」](#)

#### Blend Containerの作成

Blend Containerを使うと、ゲームの状況に応じて、互いに関連性のある複数のオブジェクトを同時に再生させ、複雑な組み合わせを表現することができます。Blend Containerで複数のオブジェクトを柔軟にまとめることができます。Blend Containerを再生すると、中にあるオブジェクトが全て、同時に再生されます。

Blend Containerには、以下のオブジェクトを入れることができます:

- ・ Blend Container
- ・ Random Container
- ・ Sequence Containers
- ・ Sounds
- ・ Switch Container

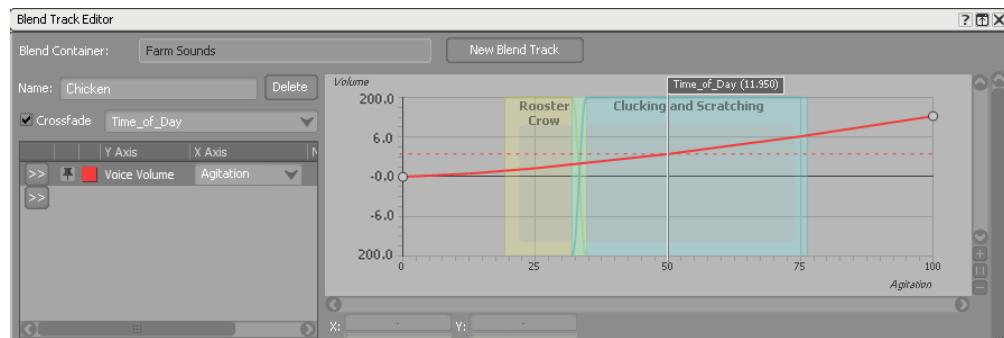
- Voices

### Blend Containerの活用例-1

ゲーム中に農場の場面があるとします。プレイヤーにリアルな農場の中を歩き回る感覚を提供しなければなりません。そこで雰囲気を出すために、様々な家畜サウンド（牛のモー、鶏のコッコ、馬のヒヒーンという鳴き声）を用意して、Farm Sound（農場音）というBlend Containerにまとめて入れて、同時に聞こえるようにします。

### Blend Containerの活用例-2

次に、もう少し高度な農場場面を作ります。プレイヤーが農場の門をくぐる時に聞かせたいのは、幅広いサウンドであって、動物の雑音ではありません。これを実現するには、ブレンドトラックの中のサウンドを整理して、「時計」などのGame Parameterに基づきサウンドをクロスフェードさせる方法があります。例えばChicken（鶏）というブレンドトラックを作り、早朝に雄鶏のコケコッコー、日中にコッコという鳴き声や地面をつつく音、そして夜は静寂、というサウンドを表現します。さらに、サウンドの特徴をGame Parameterに対応させるRTPCも設定できます。例えば、プレイヤーが剣を振り回す時に鶏を興奮させたいとします。それには、Agitation（動搖）というRTPCを作成して、Agitationのレベルが上昇するとブレンドトラックChickenのボリュームが増加するようなRTPCカーブを設定できます。



### ブレンドトラックを使いこなす

ブレンドトラック（Blend tracks）は、Blend Containerの中にあり、オブジェクトやそのプロパティ値をまとめるために使います。ブレンドトラックに多数のオブジェクトを入れることができ、全てを同時に再生させることも、RTPCを使ってGame Parameterに応じて再生させることもできます。

ブレンドトラックを使うには、以下を行う必要があります:

- 「[ブレンドトラックの作成](#)」
- 「[ブレンドトラックのオブジェクトの追加、削除](#)」

### ブレンドトラックの作成

Blend Containerのオブジェクトを整理するには、まず必要なブレンドトラック（Blend tracks）を作成してブレンドトラック名を入力します。

## 新しいBlend Trackを作成するには:

1. Property Editorに、Blend Containerを1つロードする。
2. Edit Blend Tracksをクリックする。

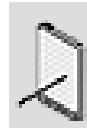
Blend Track Editorが開く。

3. New Blend Trackをクリックする。
4. Blend Trackの名前を入力して、Enterを押す。

Blend Track Editorに新規Blend Trackが表示される。

5. 必要に応じてこの処理を繰り返し、新規Blend Trackを作成する。

作成したBlend Trackは、Blend Track Editorに表示されるほか、Contents EditorのBlend Tracks部分にも表示される。



### 注記

Blend Trackを削除するには、Blend Trackを選択してDeleteを押します。

## 関連トピック

- ・「Blend Trackのオブジェクトの追加、削除」
- ・「Blend TrackのRTCP Curveの設定」
- ・「Blend TrackのRTCP Curveの編集」
- ・「Blend TrackのRTCP Curveの表示」

### Blend Trackのオブジェクトの追加、削除

Blend Trackは、オブジェクトを入れるまで、空の状態です。Blend Containerに入っている各Blend Trackに、最大128個のオブジェクトを入れることができます。なお、1つのオブジェクトが同時に複数のBlend Trackに存在することも可能で、1つのBlend Trackに同じオブジェクトを複数回、入れることも可能です。Blend Trackのオブジェクトの追加や削除は、Contents Editorから行います。

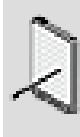


### Tip

Blend Track内のオブジェクトの順番は、クロスフェード時のオブジェクトの配置を左右するため、重要です。クロスフェードについては「[Crossfadeの管理](#)」を参照してください。

## Blend Containerの中のBlend Trackに、オブジェクトを追加するには:

1. Property Editorに、Blend Containerを1つロードする。
2. Blend Trackの中身を追加するには、Contents EditorまたはProject Explorerからオブジェクトをドラッグして、Blend Track(Blend Track)リストにあるBlend Trackに入れる。



### 注記

Blend Trackからオブジェクトを削除するには、オブジェクトを選択してDeleteを押します。この操作でBlend Trackリストからオブジェクトが削除されますが、Contents Editorからは削除されません。

## 関連トピック

- 「Blend TrackのRTPCカーブの設定」
- 「Blend TrackのRTPCカーブの編集」
- 「Blend TrackのRTPCカーブの表示」

### Blend TrackのRTPCカーブの設定

Blend TrackにRTPCカーブを設定する方法は、以下の2通りです。

- Blend Container自体にカーブを設定
- Blend Container内の各Blend Trackに、個別にカーブを設定

Blend Container自体にRTPCを設定すると、このRTPCがContainer内にある全てのオブジェクトに均等に適用されます。RTPCの作成や活用方法については[「概要」](#)を参照してください。

一方、Containerの各Blend Trackに個別にRTPCを設定することも可能です。この方法では、限定されたオブジェクトに対してRTPCを設定できます。例えばレーシングゲームの衝突サウンドを全て、1つのBlend Trackに入れます。次にこのBlend Trackに対して、衝撃強度に応じてボリュームを変化させるRTPCを設定すると、衝突サウンドが発生する度に、ボリュームがこのRTPCによって変化します。

### Blend TrackにRTPCカーブを設定するには:

- Blend Container Property Editorで、Edit Blend Tracksをクリックして、Blend Track Editorを開く。



### Tip

Ctrl+Shift+Tを押して、Blend Track Editorをアクセスすることもできます。

- 既に作成してあるBlend Trackが表示される。
- RTPCカーブを付加するBlend Trackの、Property Selectorボタンを押す。

設定可能なサウンドプロパティのリストが表示される。

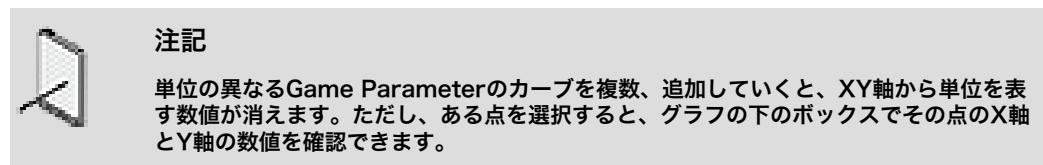
- RTPCに応じて変化させるサウンドプロパティをクリックする。

グラフビューにRTPCカーブが表示される。

- X軸のリストから、Wwiseのプロパティに対してアサインするGame Parameterをクリックする。

グラフビューのX軸に、Game Parameterが表示される。

5. グラフに他のRTPCカーブを設定するには、上記ステップを繰り返す。



### 関連トピック

- 「ブレンドトラックのオブジェクトの追加、削除」
- 「ブレンドトラックのRTPCカーブの編集」
- 「ブレンドトラックのRTPCカーブの表示」

#### ブレンドトラックのRTPCカーブの編集

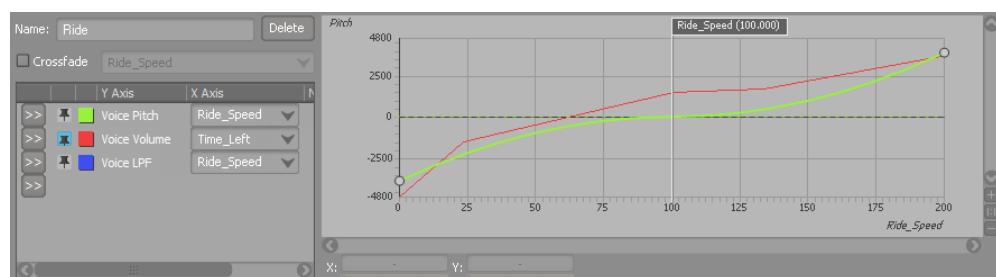
ブレンドトラックにRTPCカーブを単数または複数、設定した後は、他のRTPCカーブと同様に編集できます。例えば、ポイントの追加や移動、カーブセグメントの設定などが可能です。詳細については「[Game Parameterを使って、プロパティ値をコントロールする](#)」を参照してください。

### 関連トピック

- 「ブレンドトラックのオブジェクトの追加、削除」
- 「ブレンドトラックのRTPCカーブの設定」
- 「ブレンドトラックのRTPCカーブの表示」

#### ブレンドトラックのRTPCカーブの表示

Blend Containerの各ブレンドトラックに、複数のRTPCカーブが設定されていることもあります。複数のカーブを同時に編集するには、Blend Track Editorを使います。



RTPCリストでRTPCカーブを選択すると、グラフでハイライト表示され編集可能となります。RTPCカーブの編集方法については「[Game Parameterを使って、プロパティ値をコントロールする](#)」を参照してください。

### 関連トピック

- 「ブレンドトラックのオブジェクトの追加、削除」
- 「ブレンドトラックのRTPCカーブの設定」
- 「ブレンドトラックのRTPCカーブの編集」

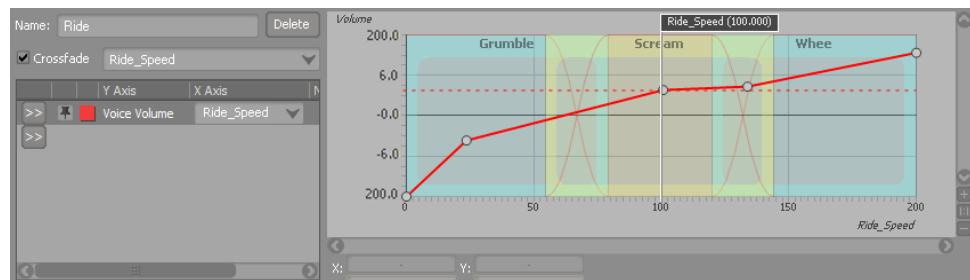
## Crossfadeの管理

Blend Containerを使うと、複数のオブジェクトを同時に再生できます。この時、ブレンドトラックにクロスフェードを適用すると、オブジェクトの聞こえ方や感じ方が変わります。クロスフェードを管理しやすくするために、ブレンドトラックのオブジェクトがブロック（長方形）として表示されます。ブロックはグラフに表示され、X軸が変化するGame Parameterを表します。ブロックのX軸の位置が、そのサウンドが聞こえるゲームパラメータ値、またはモーションエフェクトを感じとれるゲームパラメータ値を示します。

ブロック同士をオーバーラップさせてクロスフェードをつくり出すと、オブジェクト間の移り変わりがスムーズになります。また、Blend ContainerのブレンドトラックにおいてクロスフェードとRTPCカーブを組み合わせることで、オブジェクトのプロパティを調整できます。

### クロスフェードの活用例

例えば、ジェットコースターのシミュレーションゲームを考えます。スピードが時速0kmから200kmの範囲で変化するジェットコースターを、プレイヤーが作るゲームです。初歩的な方法として、ジェットコースターの乗客サウンドにRTPCを設定したBlend Containerを用意すれば、ジェットコースターが加速するにつれボリュームを増加できます。ここで乗客サウンドとして、Grumble（退屈した不満の声）、Whee（興奮した歓声）、Scream（恐怖の叫び）などを多数そろえたとします。これらのサウンドをBlend Containerに入れ、ジェットコースターのスピードに合わせてサウンドの種類を選びます。ジェットコースターのスピードが時速40kmに到達した時点で、興奮した歓聲音を再生し始めることができます。プレイヤーは、色々な乗客サウンドを、次第に大きくなるボリュームで聞くことができます。

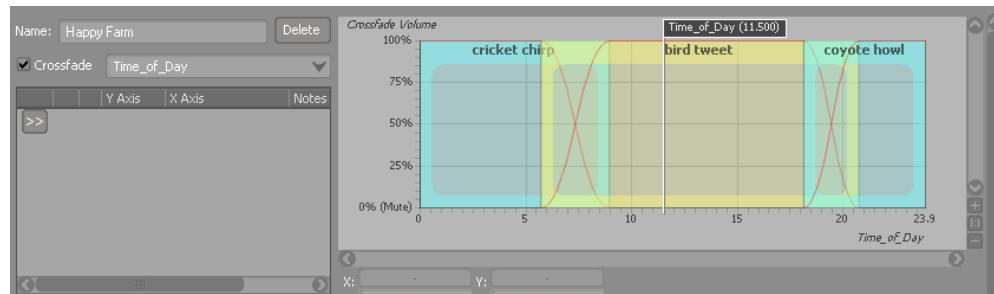


### クロスフェードの利用

Blend Containerでクロスフェード（Crossfade）を使うには、まずクロスフェードにチェックを入れます。次に、各ブロックの大きさを調整します。Game Parameterが、ブロックがX軸に接するポイントまで到達すると、そのオブジェクトが聞こえてきます。ブロック同士をオーバーラップさせると、クロスフェードが発生します。

クロスフェードを使って、Blend Container内のオブジェクトの再生を整理できます。Blend Containerのオブジェクトは全てが同時に再生され、RTPCのマッピングに基づいて聞こえる、または感じ取れるので、クロスフェードを活用すればオブジェクト同士がスムーズに移り変わる空間が作れます。例えば、ゲーム内の時計と

連動するRTPCがあれば、コオロギの音や鳥の鳴き声のクロスフェードをオーバーラップさせ、夜が明ける様子を表現できます。



### クロスフェード機能を使うには:

- Blend Track Editorで、クロスフェードを付加するブレンドトラックを選択する。

ブレンドトラックがアクティブになる。

- Crossfadingオプションを選択する。

Game Parameterリストが選択可能となる。

- このブレンドトラックの、クロスフェードに適用するGame Parameterを選択する。

ブレンドトラックで各オブジェクトがブロックで表示され、Game Parameterの単位がX軸に表示される。

ブレンドトラックのオブジェクトは、色の異なるブロックとしてグラフ上に個別に表示される。ブロックが並ぶ順番は、Contents editorのブレンドトラック列に表示されるオブジェクトの順番と同じである。

- ブロックの大きさや位置を変更するには、ブロックを選択して左辺または右辺を横にドラッグする。

ブロックで表示されたオブジェクトが、X軸に沿って再生される。

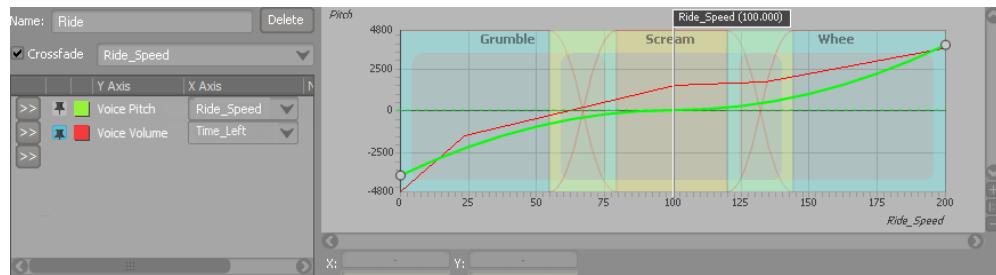
### 関連トピック

- 「Crossfadeの管理」
- 「クロスフェードとRTPCを使う」
- 「クロスフェードのモードの種類」

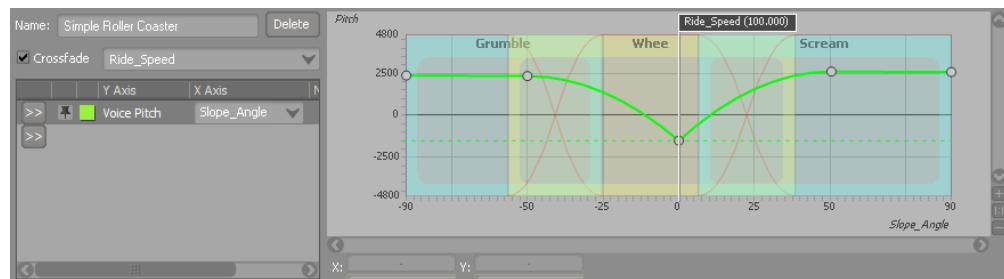
### クロスフェードとRTPCを使う

ブレンドトラックで、RTPCカーブの能力とクロスフェード (Crossfade) の柔軟性を組み合わせることができます。Blend Containerのオブジェクト間のクロスフェードをGame Parameterで変化させながら、それに重ねてオブジェクトの RTPCカーブも設定できるのです。

## オブジェクトの再生方法の設定



なお、1つのブレンドトラックにおいて、クロスフェードとRTPCカーブを別々のパラメータに合わせることができます。例えば、ジェットコースターのシミュレーションゲームで、RTPCカーブを使って車両の傾斜がきつくなるにつれ乗客サウンドのピッチを増加させます。しかし同時に、スピードに基づくクロスフェードをブレンドトラックに設定して、再生するサウンド自体をスピードに応じて切り替えることができます。その結果、ジェットコースターが速くなるにつれ聞こえてくるサウンドが変わり、などのサウンドが再生中であっても、車両の傾きがきつくなるほど、ピッチが高まります。



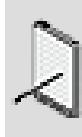
## 関連トピック

- 「クロスフェードの利用」
- 「Crossfadeの管理」
- 「クロスフェードのモードの種類」
- 「クロスフェードトランジションの注意事項」

## クロスフェードのモードの種類

目的に合わせて、Wwiseで様々な方法のクロスフェード (Crossfade) を適用できます。ブレンドトラックのオブジェクト同士のクロスフェードを調整するには、以下の3種類のモードから選択します:

- **None:** オーバーラップするブロックの境界線を超えた瞬間に、RTPCのプロパティが最大値から最小値に、または最小値から最大値に、切り替わる。
- **Automatic (default):** フェードの長さが、隣接するブロックとのオーバーラップ幅によって決まる。隣にブロックがなければ、フェードしない。
- **Manual:** フェードイン終了とフェードアウト開始の各ポイントを、移動して決める。なお、フェードイン開始とフェードアウト終了の各ポイントは、必ずオーバーラップするブロック外側の下の角となる。



### 注記

クロスフェードのカーブ形状を変更することもできます。カーブ形状の設定については「[コントロールポイント間の、カーブ形状の指定](#)」を参照してください。

## クロスフェードモードを選択するには:

1. ブロックの端を右クリックして、モードを1つクリックする。

## 関連トピック

- 「[クロスフェードの利用](#)」
- 「[Crossfadeの管理](#)」
- 「[クロスフェードとRTPCを使う](#)」
- 「[クロスフェードトランジションの注意事項](#)」

## クロスフェードトランジションの注意事項

クロスフェードトランジションをRandom ContainerやSequence Containerで使う時に、特に注意すべき点を以下に示します。

- オーディオファイルの長さ
  - オーディオファイルの長さは、0.2秒以上なければなりません。
  - クロスフェード時間は、最低0.1秒です。
- クロスフェード時間とオーディオファイル長さの関係
  - サウンドAからサウンドBにクロスフェードする場合に、サウンドエンジンが許容できる最大クロスフェード時間は、オーディオファイルAの長さの1/2です。クロスフェード時間が許容できる最大時間よりも長いと、クロスフェード時間がフェードアウトしていくオーディオファイルの長さの半分になるよう自動的に調整されます。



### 注記

コンテナにクロスフェード時間の長すぎるオーディオファイルがあっても、Wwiseでそれを制限したり表示したりすることはできません。クロスフェード時間の調整が必要であれば、ランタイムにサウンドエンジンが行います。

- ピッチとクロスフェード
  - コンテナのピッチ値をRTPCで設定したり、コンテナの再生中にSet Pitch Event Actionがトリガーされると、サウンド間でクロスフェードが適用された時に予期しない結果が発生するかもしれません。
- ソースプラグインとクロスフェード
  - クロスフェードをソースプラグインに適用すると、ソースの終わりが判断できずにクロスフェードが無視されることがあります。例えば、Sineで生成したソースの長さがRTPCに依存している場合などに、これが起きます。このような時はクロスフェードが無視され、クロスフェードのないトランジションとなります。
- Switch Containerとクロスフェード

- Switch ContainerがSequence Containerの子である場合は、SwitchにアサインしたWwiseオブジェクトの数によってクロスフェードトランジション適用の仕方が変わります。
- 2つのボイス
  - サウンドエンジンは、クロスフェード時に2つの別々のボイスを使います。
- バーチャルボイスとクロスフェード
  - バーチャルボイスの動作設定であるPlay from BeginningやResumeは、設定の意図からしてボリュームスレッショルド（閾値）より小さくなつた時やPlayback Limitを超えたとき、サウンド長さに影響しますが、クロスフェード時間のメカニズムはこれを考慮できません。
  - ボイスは、スレッショルド以下のボリュームになると、バーチャルボイスに変わるかもしれません。どのようなサウンドでも、スレッショルドに対して比較するボリュームは、そのサウンドの全オーディオチャンネルの実際の有効ボリュームです。つまりActor-Mixer Hierarchyの様々なボリューム、フェードトランジション、インタラクティブミュージックのトランジション、RTPC、State、ポジショニング、減衰などの結果です。
  - フェードトランジションの作用も、サウンドの有効ボリュームを計算する上で影響します。ランダム、シーケンス、Blend Containerの中のクロスフェードトランジションでは、フェードしていくサウンドが一定期間、ボリュームスレッショルド以下になる可能性が高くなります。スレッショルド以下の動作設定が「From Beginning」や「Resume」であれば、実際の長さが、コンテナのロジックで予想する長さよりも、長くなります。結果として、予期しない動作が生じます。さらに悪い状況として、スレッショルド以下でフェードアウトしたサウンドが、あたかも（つまり「バーチャル」には）再生停止となり、永遠に再生し続けてしまいます。その結果、引き続き他のサウンドをシーケンス再生することを、コンテナがやめてしまふかもしれません。

結論として、このようなタイプのバーチャルボイスを、クロスフェードトランジションを使ったコンテナに使うのは避けるべきです。このようなコンテナでバーチャルボイスを使いたい場合は、動作設定をPlay from elapsed timeにします。

### 関連トピック

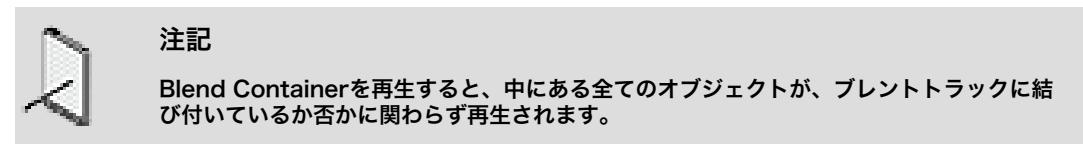
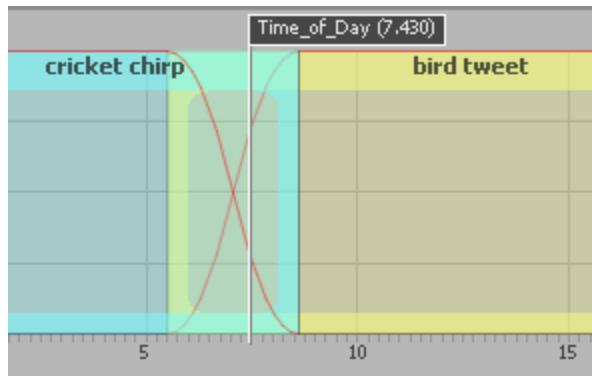
- 「クロスフェードの利用」
- 「Crossfadeの管理」
- 「クロスフェードのモードの種類」
- 「クロスフェードとRTPCを使う」

### Blend Containerの中身の試聴

Blend Containerを使えば複雑でリアルなサウンドスケープをつくり出せるので、作成中にテストしてください。Transport ControlとBlend Track Editorを使って、Blend Containerの中身を試聴できます。また、その場でクロスフェード幅の変更や、RTPCカーブのポイント追加など、各種の調整も可能です。

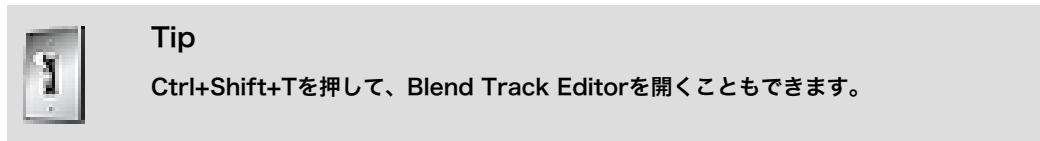
Blend Track Editorに表示されるカーソルを移動して、Blend Containerの再生中にGame Parameterを変更できます。オブジェクトを既にWwiseのプロパティ値に

マッピングしてあるので、Game Parameter値を変更すると、自動的にオブジェクトのプロパティ値も変わります。これは、ゲーム中にGame Parameterが変わった時の状況をシミュレーションするものなので、設定したプロパティのマッピングがゲーム中に効果的に作動するかを、検証できます。



Blend Containerのコンテンツを再生するには:

1. Transport Controlに、ブレンドトラックを1つロードする。
2. Blend Container Property Editorで、Editをクリックして、Blend Track Editorを開く。



Blend Containerの各ブレンドが表示される。

3. Playアイコンをクリックする。

Blend Containerを再生しながらカーソルを使ってGame Parameter値を変更すれば、その変更にオブジェクトが反応する様子を聞く、または感じることができます。



### 関連トピック

- ・「[Blend Containerの作成](#)」
- ・「[ブレンドトラックを使いこなす](#)」

- ・「ブレンドトラックのオブジェクトの追加、削除」

## オブジェクト再生に関する、コツとベストプラクティス

プロジェクト階層の各種オブジェクトの再生方法（Playback behaviors）を決定する前に、以下のセクションに目を通して、求められる効果を達成するために役立つ、コツやベストプラクティスを参照してください。

### サウンドのストリーミング

基本的なルールとして、メモリに保存するには大きすぎるサウンド、例えばミュージックやアンビエントルームトーンなど、また、ボイスも大量にある場合は、ストリーミングします。しかし、どの方法が最適かを判断する前に、様々な条件を考慮する必要があります。

プロジェクトの開始段階で開発チームと話し合い、以下を決めてください。

- ・オーディオ用に使えるストリーミング数
- ・各ストリームの帯域幅
- ・各ストリームを、他と共有するか否か
- ・同時に使えるストリーミング数
- ・プラットフォームのハードドライブからストリーミングするのか、ディスクから直接するのか、それともリモートサーバからするのか

以上の情報を踏まえた上でサウンドのサンプルレートや圧縮形式を考慮して、必要な計算をすれば、そのサウンドをストリーミングするのか否かを的確に判断できます。

### ストリーミングと、プリフェッチ

同時ストリーミング数が多いほど、必要なプリフェッチ時間が増えます。まずは短いプリフェッチ時間にして、プロジェクトが進行する中で同時ストリーミングの数が増え始めれば、必要に応じて長くします。

### Random Container

Continuous設定のRandom Containerを、Shuffleモードで使用する目的は、サウンドの繰り返しを防ぐためであり、スコープをグローバルに設定したRandom Container内では、サウンドの繰り返しを避けるようにします。しかし、コンテナ内のサウンド数が少なく、このコンテナが再生されるインスタンス数が多い場合は、ある程度の繰り返しが発生するかもしれません。このような状況を避けるには、コンテナ内のサウンドオブジェクト数を、再生インスタンス数の2倍以上にします。例えば、あるRandom Containerの再生インスタンスが3個あれば、コンテナに最低6個のサウンドオブジェクトを入れて、繰り返しを回避します。

### Blend Container

- ・CPU負荷 - Blend Containerを再生すると、その中のオブジェクトが全て同時に再生されるので、多大なメモリ負荷につながる可能性があることに注意してください。なお、CPU負荷を削減するためにバーチャルボイス設定を利用するこ

も可能ですが、クロスフェードが短く頻繁な場合は、不具合が発生するかもしれません。バーチャルボイスの使い方については「ボリューム閾値とバーチャルボイス」を参照してください。

- Switch Containerを使うか、Blend Containerを使うか - ゲーム中にRPCを設定したSwitch ContainerやBlend Containerを使うと、両者とも同じような結果になることがあります。しかし、この2つの機能の仕組みはかなり違うので、ゲームの要件に従い選択してください。
- 違う時に違うオブジェクトを再生させたい場合は、クロスフェードが重要でなければ、Switch Containerを使う。
- 全てのオブジェクトを毎回、再生させ、クロスフェードも必要な場合は、Blend Containerを使う。

---

## 第11章 Positioningの定義

Positioning定義の概要 .....	244
WwiseのPositioningを理解する .....	245
2Dオブジェクトの活用 .....	247
3Dオブジェクトの活用 .....	249
BusにPositioningを適用 .....	252
3Dポジショニングのイメージ図 .....	253
距離で変化する減衰設定 .....	265
アニメーションパスを利用した、空間ポジショニング .....	278
センタースピーカーの、オーディオ信号のルーティング .....	289
Positioningに関する、コツとベストプラクティス .....	289
チャンネルコンフィギュレーションを理解する .....	296
スピーカーとヘッドフォンのパンニングルール .....	298
Ambisonicsを使う .....	300

## Positioning定義の概要

ゲームのサウンド、ミュージック、モーションなどのPositioning（ポジショニング）やPropagation（伝搬）はプレイヤーをゲームに引込み、没入感を強化させる重要な役割を担うので、ゲームで設定できる多様なサウンドエフェクトやモーションエフェクトの使い方を理解することが大事です。

典型的なゲームでは、以下に示すサウンドエフェクトやモーションエフェクトを、組み合わせて使います。

- ・局所的なアンビエントサウンド - サウンドエミッタが1つの場所から動かない。局所的なアンビエントサウンドの例として、大きい機械や噴水などがある。
- ・非局所的なアンビエントサウンド - サウンドエミッタは動くが、特定のゲームオブジェクトに添付されていない。非局所的なアンビエントサウンドの例として、アンビエント（背景）の鳥の鳴き声や虫の音などがある。
- ・移動するオブジェクトのサウンド - サウンドエミッタは特定のゲームオブジェクトと共に動く。移動するオブジェクトサウンドの例として、吠える犬や怒鳴る守衛など、ゲームのキャラクターや動物がトリガーするサウンドがある。
- ・ゲームインターフェースのサウンド - ゲームインターフェースの特定エレメントなど、画面上の定位置のアイテムに関連するサウンド。ゲームインターフェースのサウンドの例として、ヘッドアップディスプレイ（HUD）のパーツ、メニュー音（ボタン音、項目移動の音）、ファーストパーソンシューティングゲームの銃などがある。

Wwiseの強力で柔軟なポジショニング設定ツールを活用すれば、このような各種サウンドタイプを実現させ、ゲーマーの期待に応えるようなエクスペリエンスを生み出すことができます。

### ポジショニングの活用例（Part 1）

ファーストパーソンのステルスゲームの例を考えます。ゲーム中に特殊部隊が遠く離れた火山島までたどり着き、隊員の1人をテロリスト達から救出する場面があるとします。非常に危険なミッションのため、救出隊員はチームとして協力して一緒に行動しなければなりません。隊員達は敵地のジャングルを静かに前進するにつれ、以下のサウンドやモーションを体験します。

- ・メインキャラクターの足音
- ・敵のジャングル基地で灯される、数々のトーチ
- ・小屋の中で話し合うテロリスト達
- ・頭上を飛ぶ蚊
- ・本部から発信される最新情報
- ・ミッションに挑む隊員同士の、声をひそめたやりとり
- ・ミッション遂行に成功した後に、基地を破壊するために仕掛けた爆弾の爆発
- ・止むことのない島の火山の地響き
- ・インタラクティブミュージック

これら全てのサウンドは、ポジショニングや伝搬の詳細設定が異なります。それぞれのサウンドをWwiseで管理する方法を、種類別に次のセクションで説明します。

## WwiseのPositioningを理解する

Wwiseは、ゲームの様々なサウンド、ミュージック、モーションオブジェクトのリアルなPositioning（ポジショニング）やPropagation（伝搬）をシミュレーションするための、多様なツールをそろえています。Wwiseの各種オブジェクトにポジショニングを設定する方法を、以下のセクションで説明します。

- 「サウンドオブジェクトやミュージックオブジェクトのPositioning設定」
- 「1つのゲームオブジェクトに対し、複数のポジションの作成」
- 「モーションオブジェクトのポジショニング」
- 「プロジェクト階層内で、ポジショニングオプションの設定」
- 「3Dポジショニングでマルチチャンネルソースを扱う」

### サウンドオブジェクトやミュージックオブジェクトのPositioning設定

基本の設定としてサウンドオブジェクトやミュージックオブジェクトは、2D positioning (2Dポジショニング) または3D positioning (3Dポジショニング) を選べます。

この2種類のポジショニング設定の主な違いは、ソースチャンネルのアウトプットスピーカーへのマッピング方式です。2Dポジショニングではソースチャンネルが互いにリンクしているのに対し、3Dポジショニングでは入力チャンネルをサラウンド環境のどのスピーカーにアウトプットすることも可能です。ポジショニング方式は、Positioning TypeにRPCを設定することでランタイムに変更できます。

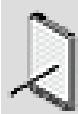
2Dのサウンドオブジェクトやミュージックオブジェクトの場合は、パンニングで各チャンネルのボリュームのバランスをとり、サウンドオブジェクトまたはミュージックオブジェクトがサラウンド環境で別々のスピーカーから聞こえるようにできます。

一方、3Dのサウンドオブジェクトの場合は、サウンドのポジショニング設定で考慮すべき条件は、サウンドエミッタの3D空間における実際のポジションと、リスナーからの距離の2点です。この2つの条件に対応するためのWwiseのツールを以下に示します。

- **Attenuation（減衰）** - サウンドエミッタがリスナーから遠ざかる時や向こうを向く時に、オーディオ信号が自然と弱まる様子をシミュレーションする。
- **Spatialization（空間配置）** - サウンドオブジェクトやミュージックオブジェクトの実際の位置やポジショニングを決定する。

さらに柔軟に設定できるように、空間配置に使う情報を以下から選択できます。

- **User-defined（ユーザーが定義する）** - Wwiseの専用アニメーションパスを使い事前に定義したポジショニング情報。
- **Game-defined（ゲームが定義する）** - ゲーム内のサウンドエミッタの実際のポジションを元に計算したポジショニング情報。

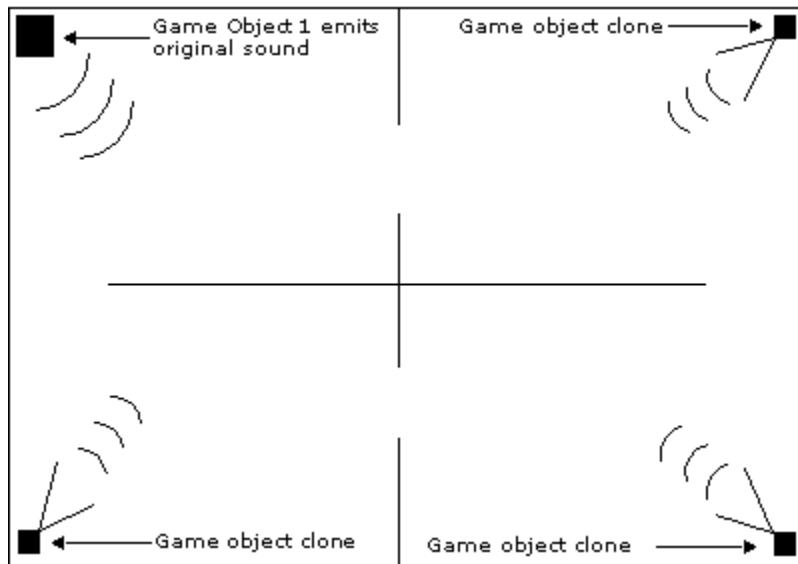


#### 注記

Game-definedやUser-definedのポジショニング情報はサラウンド環境内の空間配置のほか、減衰のための距離計算にも利用します。

## 1つのゲームオブジェクトに対し、複数のポジションの作成

ゲームの状況によって、リアルなサウンドスケープを生み出すには単一のサウンドインスタンスでは不充分となることも考えられ、その例が複数のスピーカーのPAシステムなどです。この時、サウンドとそれに対応するゲームオブジェクトを複製せずに対応するには、サウンドをエミット（出す）ゲームオブジェクトのクローンを作成して、ゲーム内の別の場所に配置することができます。PAシステムの例で言えば、オリジナルのサウンドをエミットするゲームオブジェクトのクローンを複数、作成して、下図のようにPAシステムの各スピーカーの場所に配置します。



クローンは、オリジナルのサウンドの正確な複製サウンドを出し、クローンのプロパティ、動作、減衰設定などはオリジナルのサウンドと同じです。なお、オリジナルのサウンドと複数のクローンが同時に再生されるので、それぞれのボリュームを全て足すのか、全サウンドの累積ボリュームをオリジナルのサウンドの最大ボリュームレベルで再生するのかを、選べます。

クローンを使うと、サウンドが单一のポジションで発生する場合とは違うスプレッドや減衰となる可能性が高いので、オリジナルのサウンドのスプレッド設定と減衰設定に気をつけてください。特に、オリジナルのサウンドとクローンの、複数のサウンドが同時にプレイヤーに聞こえる状況では、注意が必要です。クローンサウンドがオリジナルのサウンドに追加されると、合算されたサウンドのボリュームが大きすぎて、クリッピングが発生することがあります。

このようにゲームオブジェクトのクローンを使える他の事例を、以下に示します。

- 廊下の壁に、多数のトーチ（たいまつ）が並ぶ時。
- 池や滝などの大型の変形したオブジェクトがあり、単純な半径減衰が適さない時。
- 壁が部分的に崩れ落ち、壁の向こう側にあるサウンドが露出された時。クローンした複数のゲームオブジェクトを、それぞれ特定の場所に配置することで、部分的なオクルージョンやオブストラクションの表現となりうる。

ゲームオブジェクトのクローンを作成して改めて配置するのは、Wwise SDKで行います。1つのゲームオブジェクトに対して複数のポジションを設定する方法については、Wwise SDKドキュメンテーションの「[Sound Engine Integration Walkthrough > Integrate Wwise Elements into Your Game > Integrating 3D Positions > Integration Details - 3D Positions](#)」を参照してください。

### モーションオブジェクトのポジショニング

ゲームコントローラーなど、高度でないデバイスでは、モーターを使った3D環境のシミュレーションができないため、ポジショニング (Positioning) は問題となりません。ただし、減衰を利用して、プレイヤーから離れるモーション信号の強度を徐々に減らすことはできます。

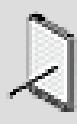
### プロジェクト階層内で、ポジショニングオプションの設定

オブジェクトのポジショニング (Positioning) 設定をするにあたり、階層内の各オブジェクトごとにポジショニングを設定できることを忘れないでください。ポジショニングは絶対プロパティとして扱われるため、最上位のオブジェクトのポジショニングは、自動的に子オブジェクトに継承されます。しかしこれらの設定をオーバーライドさせ、特定のオブジェクトのポジショニングをカスタム設定することも可能です。階層内でポジショニングがどう作用するかについては「[プロジェクト階層内のプロパティについて](#)」を参照してください。

### 3Dポジショニングでマルチチャンネルソースを扱う

マルチチャンネルのサウンドをモノにダウンミックスしなくとも、モノソースと同様にPositioning Editor を使って空間配置することができます。

各スピーカーに対してソースのチャンネルをどう配分するかを計算するためには、[listener](#)の周りに半円形の「バーチャルエミッタ」が設定されます。半円形の範囲は、スプレッドの設定で決まります（100%は完全な円形、0%はリスナー正面のポイント）。この半円形がチャンネルの数だけ均等に分割され、ソースの元のチャンネルがそれぞれ、半円内の該当エリアにアサインされます。例えば、ソースがステレオの場合は、円の左半分にあるバーチャルエミッタは全て、ソースの左チャンネルを使い、右半分にあるものは全て、ソースの右チャンネルを使うことになります。その後にスピーカー毎に、各バーチャルエミッタの実際の配分が計算されます。このようにして、3Dポジショニングにおいてもマルチチャンネルのイメージがある程度維持されます。



#### 注記

スプレッドを0に設定するとバーチャルエミッタが全てリスナー正面の1点に配置されることになり、これは全てのチャンネルをモノにダウンミックスしてから3Dスペースに配置する場合と似ています。

## 2Dオブジェクトの活用

2Dサウンドや2Dミュージックのチャンネルは、リスナーやゲームオブジェクトのポジションやオリエンテーション（向き）に関わらず、デフォルトでアウトプットバスの該当チャンネルに直接アサインされます。ただし、2D Pannerを利用して、サウンドオブジェクトやミュージックオブジェクトのチャンネル同士のボリューム

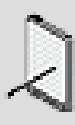
バランスを調整することで、アウトプットバスの各チャンネルから異なるボリュームで聞こえるようにできます。

2D Pannerに2次元のグラフビューが表示され、車のパンナーをコントロールする「パン」や「フェード」をシミュレーションするXY座標があります。点音源を示す黒点をグラフビュー内で自由にドラッグして、アウトプットバスの各チャンネルへのパワールーティングを変更できます。

実際の出力配分はアウトプットバスのコンフィギュレーションで決まります。例えば、アウトプットバスがステレオであれば、Y座標の影響は全くありません。なお、7.1コンフィギュレーションの場合のサイドスピーカーは、中央に黒点がある時に最大レベルに達します。

キャラクターのボイスなど、ゲームプレイに不可欠のサウンドは、確実に聞こえるようにセンタースピーカーに信号をルーティングすることができます。Center %の指定で、センタースピーカーを通る信号の量を決定できます。Center %コントロール機能については「[センタースピーカーの、オーディオ信号のルーティング](#)」を参照してください。

2Dのサウンドは、ゲーム中の様々な目的に使え、メニュー音やキャラクター音などがその一例です。また、CPUやメモリへの負荷が少ないため、メモリやCPUが不足する状況では、2Dサウンドの採用を検討しても良いでしょう。



### 注記

2D Pannerはアンビソニックスのアウトプットに全く影響しません。

## モーションオブジェクトの2Dポジショニング

ゲームコントローラーなど、高度でないデバイスでは、モーターを使った3D環境のシミュレーションができないため、パンニングなしの2Dポジショニング（2D positioning）を使います。ただし状況によって、ソースとの距離に応じてモーション信号の強度を弱めることも考えられます。この場合は、モーションオブジェクトのポジショニングを3Dに設定してから、減衰設定を利用することができます。

## 2Dオブジェクトのチャンネルを定義するには:

1. Property Editorに、最上位オブジェクトを1つロードする。
2. Positioningタブに切り替える。



### 注記

このオブジェクトが最上位のオブジェクトでない場合、Positioningオプションを設定する前にOverride parentオプションを選択する必要があります。

3. Enable Positioningを有効にする。

ポジショニングリストが有効になります。

4. 2Dオプションを選択する。

2Dポジショニング機能が設定可能となる。

5. 信号をデフォルトのフロントレフト、フロントライト以外のスピーカーやモーターにパンさせるには、Enable Pannerチェックボックスを選択する。
6. **Edit**をクリックして、2D Position Editorを開く。
7. 点音源の位置を示す黒点を、2D面上で自由にドラッグする。

指定された位置をシミュレーションするために、各チャンネルのボリュームとインテンシティが調整される。



### 関連トピック

- ・「3Dオブジェクトの活用」
- ・「センタースピーカーの、オーディオ信号のルーティング」

## 3Dオブジェクトの活用

3Dサウンドや3Dミュージックのオブジェクトでは、どの入力チャンネルを、サラウンド環境のどのスピーカーにアウトプットするのかを自由に設定できるので、リスナーとの関係に応じたオブジェクトの動きを簡単にシミュレーションできます。この強化されたサラウンド体験を達成するために、Wwiseで空間配置 (spatial positioning) と減衰の両方を使用します。

### 空間配置 (Spatial positioning)

空間配置 (Spatialization) は、ゲームの3D環境における、オブジェクトの実際の場所、つまりポジション (Positioning) を指定します。空間配置を設定すると様々なスピーカーからサウンドが聞こえ、ゲーム内をソースが動く様子が表現されます。なお空間配置を使用しない場合は、サウンドオブジェクトやミュージックオブジェクトはソースの動きに関わらず、元のチャンネルコンフィギュレーションに基づいて再生されます。

対象サウンドの種類によって、空間配置の方法を、以下の2種類から選択できます。

- ・ **User-defined** (ユーザーが定義する) - Wwiseの専用アニメーションパスを使い事前に定義したポジショニング情報。
- ・ **Game-defined** (ゲームが定義する) - ポジショニング情報を、ゲーム中のオブジェクトの実際のポジションによって、定義する。

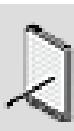
ゲーム中の移動するサウンドオブジェクトやミュージックオブジェクトの大部分が、Game-defined spatializationを利用することになりますが、User-defined spatializationの使い方も多数あり、例えば鳥や虫など固定されていないアンビエントサウンドに利用できます。

## Attenuation

空間配置に加えて、3Dのオブジェクトに減衰を設定できます。減衰設定とは、リスナーから信号が遠ざかるにつれ、自然と信号が弱まる状況をシミュレーションするものです。Wwiseでは様々なカーブを使って、ボリュームやローパスフィルタなどのプロパティ値を、具体的な距離の数値にマッピングします。これらのカーブで、オブジェクトの距離に基づいた洗練されたロールオフを設定できます。また、さらにリアルにするには、リスナーに対するオブジェクトの向きに応じてサウンドを減衰させる、サウンドコーンを設定することもできます。3Dオブジェクトの減衰設定については「[距離で変化する減衰設定](#)」を参照してください。

## Positioning オプションの組み合わせ

ポジショニングの様々なオプションを組み合わせて、ゲームのリッチで多様なオーディオ環境を作成できます。状況に応じて空間配置だけを使用したり、減衰だけを使用したり、また両方を使用したりします。



### 注記

ゲームプレイにとって重要な3Dサウンドは、必ず聞こえるようにセンタースピーカーへ信号をルーティングする方法もあります。Center %の指定で、センタースピーカーを通る信号の量を決定できます。Center %コントロール機能については「[センタースピーカーの、オーディオ信号のルーティング](#)」を参照してください。

## モーションオブジェクトの3Dポジショニング

モーション（バイブルーション）機能付きのデバイスの中には、3Dポジショニング（3D positioning）に対応していないものもあります。例えば、ゲームコントローラーのモーターは、モーターが少なく動作が限定されているため、3D spatialization（空間配置）情報を活用できません。このため、高度でないモーションデバイス向けには、2Dポジショニングをパンニングなしで、利用してください。ただし状況によって、ソースとの距離に応じてモーション信号の強度を弱めることも考えられます。この場合は、モーションオブジェクトのポジショニングを3Dに設定してから、空間配置を設定せずに減衰設定を利用することができます。

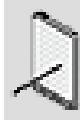
## Wwise 3D の特徴

Wwiseで3Dポジショニングがどのように計算されるかについて、以下のリストを参照してください。細かい仕組みが分からなくてもいいのでオーサリングツールの設定を始めたいと思う方は、最初のリストを飛ばしてその次のリストを見てください。

### 3D ポジショニングの計算の仕組み:

1. バーチャルポジションを決めるのは、スプレッドです（詳しくは[こちらの "スプレッドとフォーカス" のグラフ](#) を参照）。複数のインプットチャンネルがあれば、スプレッドの弧をそれで分割します（例えばステレオではバーチャルポジションの半分が左、残り半分が右にアサインされます）。
2. バーチャルポジションごとにVBAP ゲインが計算されます。
3. 全てのバーチャルポジションのゲインを、以下のように統合します：

- アウトプットチャンネルごとに、そこに貢献する各インプットチャンネルのゲインの2乗の合計を計算します。
- 次に、インプットチャンネルの数で割ります。
- 次に、その商の平方根を出します。

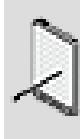


### 注記

特に`spread == 0`であれば、全てのバーチャルポジションが同じ位置となります。その場合は当然、インプットチャンネルのダウンミックスが発生します。

## 3Dオブジェクトのプロパティを定義するには:

- Property Editorに、最上位の親オブジェクトを1つロードする。
- Positioningタブに切り替える。



### 注記

このオブジェクトが最上位のオブジェクトでない場合、Positioningオプションを設定する前にOverride parentオプションを選択する必要があります。

- Enable Positioningを有効にする。

ポジショニングリストが有効になります。

- 3Dオプションを選択する。

3Dポジショニング機能が設定可能となる。

- グループボックスPosition Sourceで、以下のオプションを1つ選択して、空間配置に使う情報を決める。
  - User-definedは、アニメーションパスを使い、Wwiseで事前に定義した、空間ポジショニング情報
  - Game-definedは、ゲームがリアルタイムで計算する、空間ポジショニング情報



### 注記

User-definedを選んだ場合は、Editボタンをクリックして、3D Position Editorを開き、対象となるオブジェクトのアニメーションパスを設定します。アニメーションパスの作成については「[アニメーションパスを利用した、空間ポジショニング](#)」を参照してください。

- Position SourceをUser-definedとした場合、アニメーションパスのポジションをリスナーの向きにロックさせるには、チェックボックスFollow Listener Orientation（リスナーの向きを追う）をチェックする。このオプションを選択すると、リスナーの向きに関わらず、音が常に同じスピーカーから聞こえてくる。逆にこのオプションを選択しないと、リスナーがパスから独立して移動する。つまり、リスナーが向きを変えると、サウンドが違うスピーカーから聞こえてくる。このオプションについては「[リスナーの向きを追うアニメーションパスの作成](#)」を参照。
- Position SourceをGame-definedとした場合、ゲームのフレームごとにソースのポジション情報を更新するには、チェックボックスUpdate at each

**frame** (フレームごとに更新) を選択する。逆にこのオプションを無効になると、ソースの再生が終わるまで、ポジションが静止した状態になる。

8. 3Dゲーム空間内の動きをシミュレーションできるようにスペシャルポジショニング情報を計算するには、**Spatialization Mode** ドロップメニューで**Position**、または**Position + Orientation**を選択する。ここで**None**を選択すると、オブジェクトが、ソースの移動に関わらず、元のチャンネルコンフィギュレーションに従い再生される。

## 関連トピック

- 「距離で変化する減衰設定」
- 「アニメーションパスを利用した、空間ポジショニング」
- 「センタースピーカーの、オーディオ信号のルーティング」

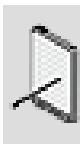
## BusにPositioningを適用

オブジェクトに適用するのと同じように、Audio BussとAuxiliary Bussに配置を簡単に適用できます。バスをProperty Editorにロードして、サウンドオブジェクトと同じように2Dまたは3Dの配置を適用するだけです。。この操作については「[2Dオブジェクトの活用](#)」と「[3Dオブジェクトの活用](#)」を参照してください。

バスとサウンドオブジェクトの両方を配置できるため、オーディオデザイナーはサウンドの伝播をより詳細に制御することができます。追加された複雑さは必ずしも必要ではありませんが、以下のような特定のシナリオでは、位置決めバスは非常に便利です：

- Virtual Surveillance または Loud-Speaker System：ゲーム内のシステムは、プレイヤーが隠れたマイクを使って他のプレイヤーを探れるようにします。オーディオはキャプチャされ、バーチャルラウドスピーカーからマップのもう一方の端に再送信されます。lo-fiエフェクトを適用すると、サブミックスは、あたかもラウドスピーカーから放射されているかのように、空間的に音が出ます。
- Acoustic Portals：空間オーディオを重視したバーチャルリアリティゲームを制作しています。錯覚を維持するには、現実的かつ説得力のある音響現象をシミュレートすることが最も重要です。リスナーが隣接する部屋から来る残響音を聞くことができるようになります。部屋のすべての音が混合され、リバーブユニットに送られ、エフェクトの出力が空間的に配置され、あたかも玄関を通り抜けているかのように配置されます。
- Early Reflections と Other Effects：音声ごとにEffectを適用するのは面倒ですが、共有バスのEffectとして適用するにはあまりにも幅があり、便利ではありません。実際には、Effectを適用できるサウンドエミッタごとにバスのインスタンスを作成したいでしょう。空間化はサブミックスのダウンストリームで行われるか、またはプラグインがそれを処理します。これは、Wwise Reflectがサウンドエミッタごとにリスナーの位置に関連して独自の初期反射セットを生成するために使用する戦略です。
- Clustering Sounds to Reduce Overhead：複雑なRTPCシステムによってなされる、かなりの数のコンポーネントサウンドで構成された車両音があります。コンポーネントは車両周辺のさまざまな物理的な場所に由来しているため、サウンドエミッタには個別のゲームオブジェクトを使用します。しかし、個々のエミッタごとにEffectと空間化を適用するのはコストがかかります。そのため、リ

スナーが適切に近くにいる場合にのみこれを実行します。リスナーが十分に離れていて、角度の微妙な変化が目立たない場合は、サウンドのグループ全体に処理と空間化を適用する前に、すべてのサウンドをミックスする必要があります。



### 注記

マスター・オーディオバスや、マスター・セカンダリバスは、ポジションを設定できません。

## 3Dポジショニングのイメージ図

3Dポジショニングとは、ソース（どのようなチャンネルコンフィギュレーションでも可）をバス（どのようなチャンネルコンフィギュレーションでも可）にパンニングする処理のことで、ゲーム中のエミッタやリスナー・オブジェクトの相対的なポジションや向き（orientation）に基づいて行います。これらのオブジェクトのポジションは、WwiseサウンドエンジンAPI経由でゲーム側が設定します。距離に基づくSpreadやFocusのプロパティがパンニング方式を左右するほか、ゲームによって動くポジションや向きも影響します。Wwiseの3Dポジショニングのアルゴリズムは、点音源とそれを取り囲む、方向性を持つマルチチャンネルアンビエンスの間を、Spreadを使って正しくモーフィングできるように設計されています。

3Dポジショニングのアルゴリズムを簡素化すると、以下のような流れです：

1. バーチャルソースをリスナー周りに配置する。
2. 各入力チャンネルに対応するバーチャルソースのグループを、Focusに従って集約する（標準コンフィギュレーションのみ）。
3. バーチャルソースを、エミッタのリスナーに対する相対的な向きに従って、回転させる（Spatializationモードでは、Position + Orientationだけに設定する）。
4. さらに、バーチャルソースをリスナーに対するエミッタの相対的なポジションに向けて、Spreadに従い、集約する。
5. 各バーチャルソースを、出力コンフィギュレーションのチャンネルに対してパンニングさせて、出力（power）をノーマライズする。

SpreadやFocusの詳細は、[オブジェクトプロパティの減衰カーブ](#)のページを参照してください。

標準的なマルチチャンネルソースの各チャンネルを、平面を均等に分割することでバーチャルソースのグループに対してアサインします。ソース側のハイトチャンネルには対応していないので、落とされます。

### チャンネルコンフィギュレーションのタイプ

Wwiseは、[標準](#)と[アンビソニック](#)のチャンネルコンフィギュレーションのネーティブサポートを提供しています。両者の主な違いは、上記の手順5に関わるパンニング計算方法で、両者ではスペーシャルオーディオの表現方法が異なるためです。また、バーチャルソースのレイアウト方法も異なり、Focusの唯一の存在理由が、個別の入力チャンネルから個別の出力チャンネルへの均一的でない「漏れ」の対策であるため、Focusは標準コンフィギュレーションに限り適用されます。ただ

し、Spreadや相対的なエミッタ・リスナー回転に関するバーチャルソースの動作は、コンフィギュレーションのタイプに関わらず同一です。

標準コンフィギュレーションのソースをアンビソニックスのバスにパンニングしたり ("encode to ambisonics") 、その逆を行ったり ("decode from ambisonics") できます。

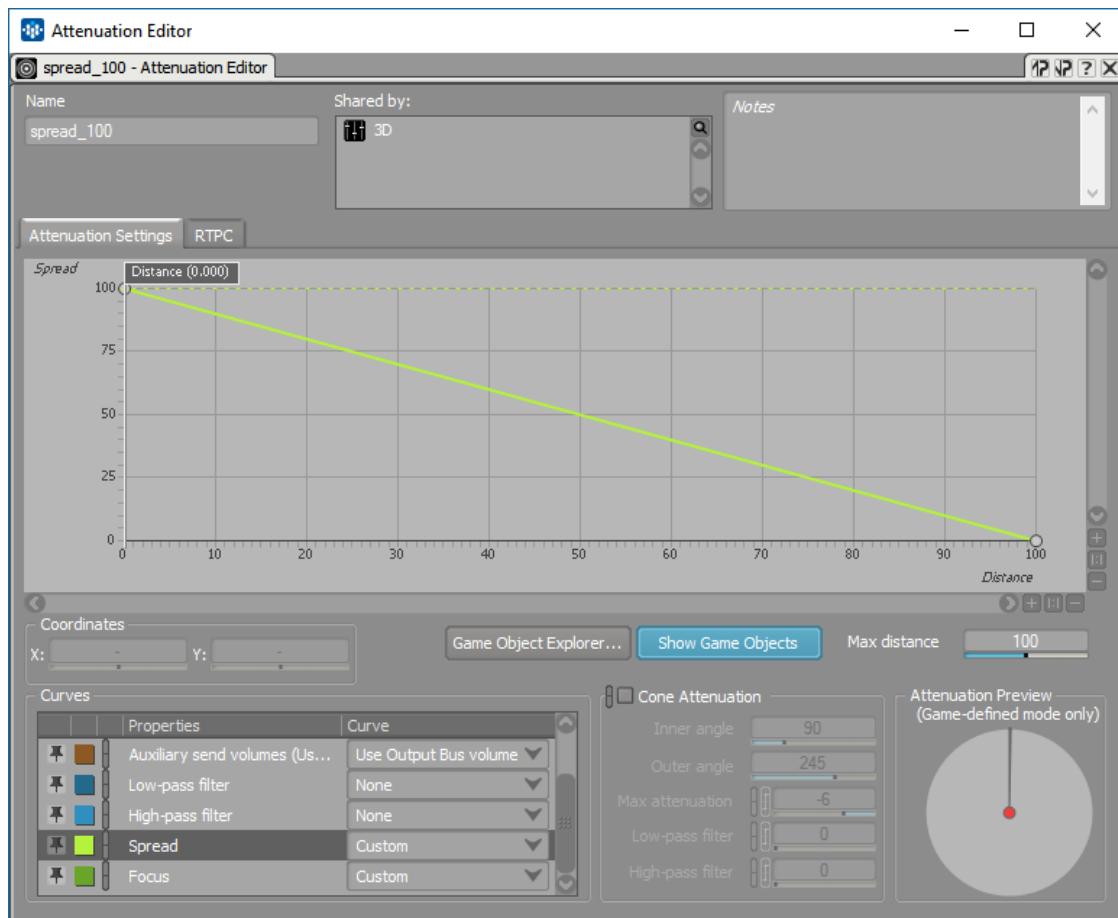
### 例

次ページ以降の事例で、上記アルゴリズムの様々な面のイメージを説明します。

- 「Spreadの効果」
- 「向き (orientation) の効果」
- 「Focus」
- 「平面のソースを、平面のチャンネルコンフィグレーションにパンニングする」
- 「3Dコンフィグレーションにパンニングする」

### Spreadの効果

下図スクリーンショットのように、SpreadのAttenuationカーブが、距離が多い時に0%（点音源）であり、距離が少ない時に100%（完全に包囲）であると仮定します。



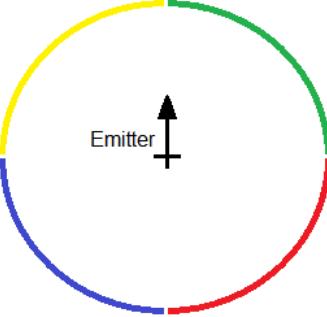
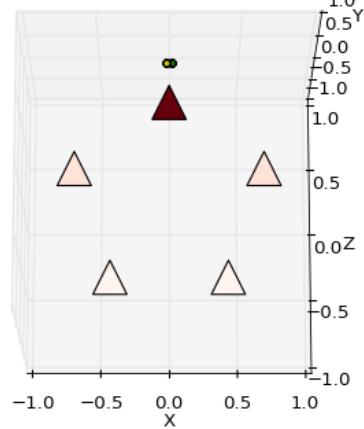
以下の事例で使うSpreadカーブ

下表のソースのチャンネルコンフィギュレーションは、4.0です。これで、入力チャンネルをバーチャルソースに振り分けた時の配分が、はつきりと分かります（FL: 黄、FR: 緑、SL: 青、SR: 赤）。

2列目の「エミッタに対するリスナーの相対的ポジション」には、エミッタであるゲームオブジェクトが北向きの矢印で表され、ゲームオブジェクトの向きがゲームによって北向きに設定されたことを表しています。リスナーであるゲームオブジェクトも北を指し、向きはエミッタであるゲームオブジェクトに向いています。4つの図でリスナーの位置が3つの異なる距離にあり、結果としてSpread値も0、30、70、100%の4種類となります。

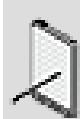
3列目の「バーチャルソースの配置と出力バスのチャンネルのエネルギー」には、その効果がバーチャルソースの配置（色付きの小さい円）として表れています。また、三角は出力バスのチャンネルを表し、センターチャンネルが上にあります。さらに三角の色が、パンニングからくるエネルギー量を示します。赤が濃いほどエネルギーが大きいことを意味します。

この例は5.0chコンフィギュレーションを使い、角度を $+/-72^\circ$ 、 $+/-144^\circ$ としていますが、5.x chのスピーカーコンフィギュレーションのデフォルト角度は、 $+/-90^\circ$ 、 $+/-135^\circ$ です。ただ、どのようなコンフィギュレーションでもアンビソニックスさえも、バーチャルソースの配置は全く同じなので表現することができます。

エミッタとリスナーの向きが同一で、エミッタがリスナーに向いている時の、Spread値の効果		
Spread	エミッタに対するリスナーの相対的ポジション	バーチャルソースの配置と出力バスのチャンネルのエネルギー
0% Spread	 	

エミッタとリスナーの向きが同一で、エミッタがリスナーに向いている時の、Spread値の効果		
Spread	エミッタに対するリスナーの相対的ポジション	バーチャルソースの配置と出力バスのチャンネルのエネルギー
30% Spread		
70% Spread		
100% Spread		

### 向き (orientation) の効果



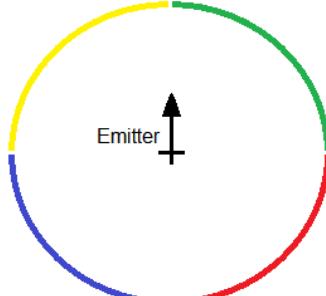
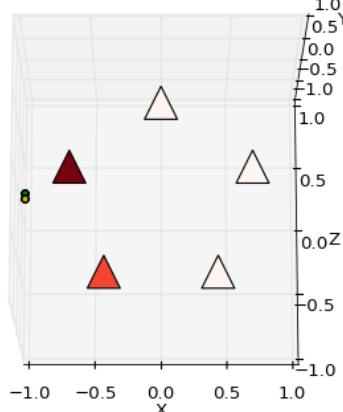
#### 注記

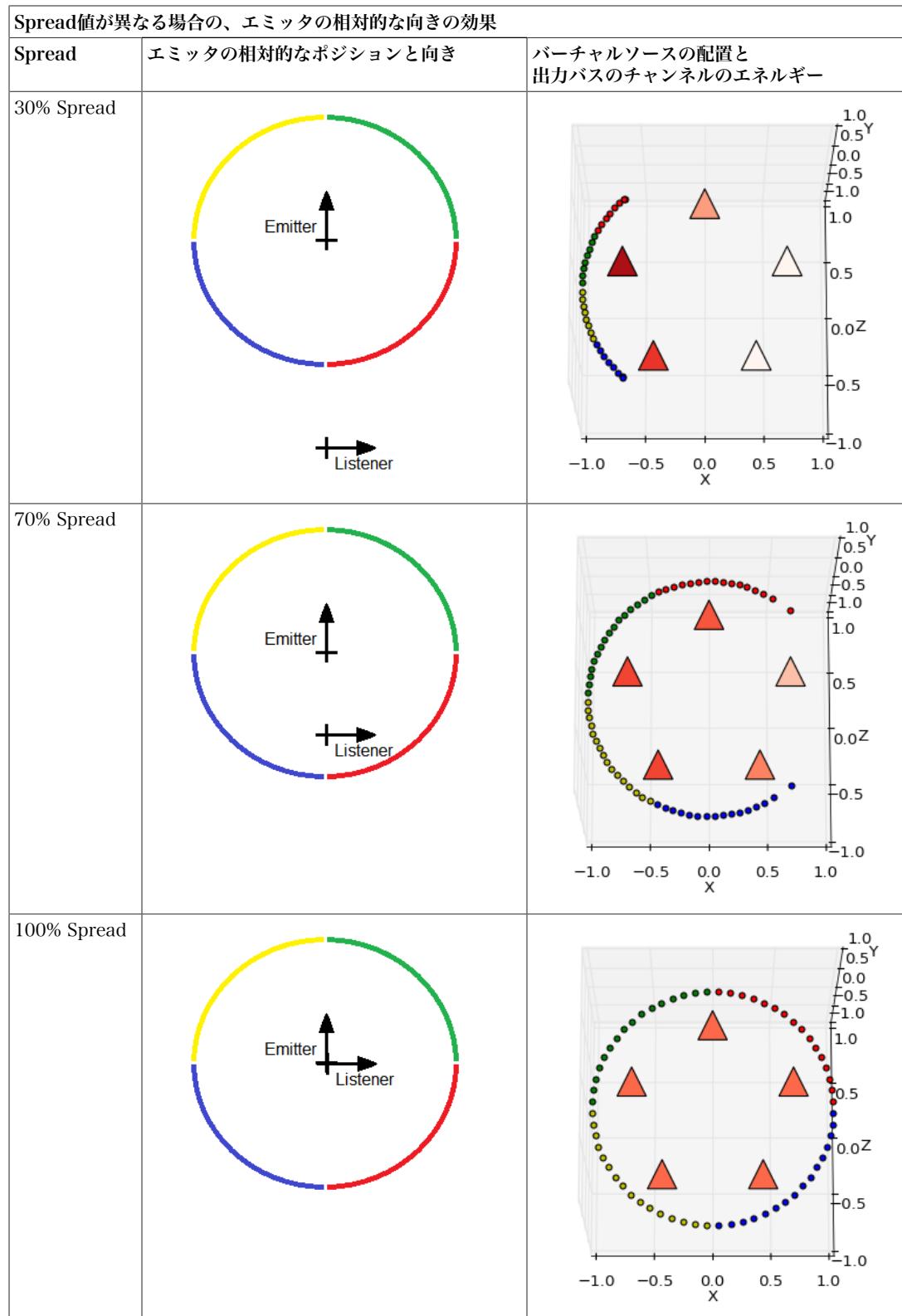
この事例の内容などについては[最初の事例](#)を参照してください。

下記のエミッタオリエンテーションの影響は、Spatialization Modeを**Position + Orientation**に設定した場合に限り該当します。以下は、**Position + Orientation**の使用と、**Position Spatialization Mode**の使用の比較検討です。

次の例は前例と似ていますが、リスナーが東向きです。これは、2つの異なる向きの動作を同時に説明できる例です：

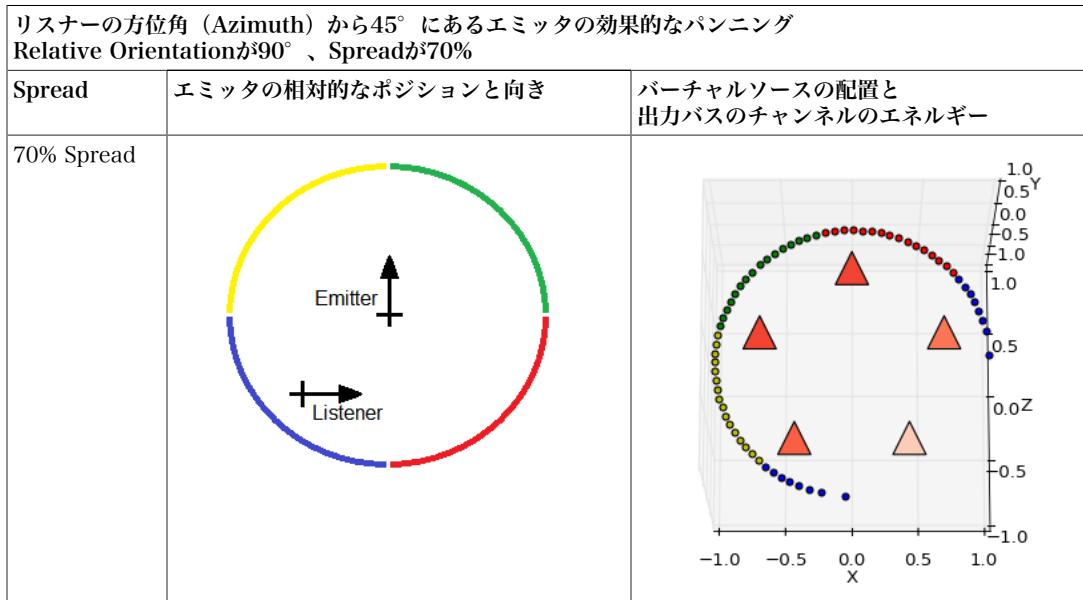
- **エミッタの相対的なポジション**: エミッタはリスナーの左側に位置しているので、Spreadによる収縮が左側で起きます。これが一番良く分かるのは、最初の0% Spreadの図です。
- **エミッタの相対的な向き**: エミッタの相対的な向きがリスナーから反時計回りに90°ずれているので、バーチャルソースも同じように回転させます。これが一番良くわかるのは、4番目の100% Spreadの図で、入力チャンネルを表す色が回転されています。もちろんソースがモノであれば、全く影響ありません。

Spread値が異なる場合の、エミッタの相対的な向きの効果		
Spread	エミッタの相対的なポジションと向き	バーチャルソースの配置と出力バスのチャンネルのエネルギー
0% Spread	 	



エミッタの相対的なポジションと、エミッタの相対的な向きの違いを明確にするために、エミッタの相対的なポジションが $45^\circ$ 、エミッタの相対的な向きが $90^\circ$ 、Spreadが70%の場合の結果を下図に示します。これは、リスナーがエミッタ

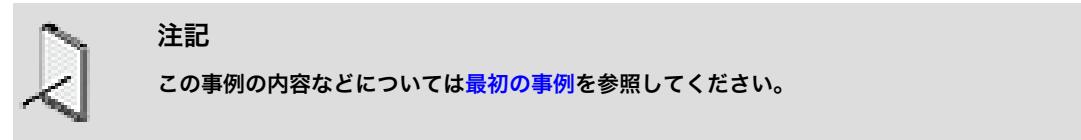
夕の音場に南西から突入して、その向きが東向きである場合を表しています。緑と赤のエミッタチャンネルがリスナーのおおよそフロントレフトとフロントライトに位置して、青のチャンネルが周囲を囲み始めている様子が分かります。



上記の例から、エミッタオリエンテーションが影響するのはマルチチャンネルソースだけだということが、分かります。オリエンテーションによって変わる回転は、マルチチャンネルソースが音場、つまりソニック要素がパンニングされてマルチチャンネル素材にレンダリングされたアンビエンスを表している場合に、最も効果を発揮します。ランタイムに、Wwiseがリスナーやエミッタのオリエンテーションに基づいて、この素材を回転させて、アンビエンスを3Dワールドのジオメトリに、結果的に結び付けます。

ただし場合によっては、マルチチャンネルファイルが音場を表しているのではなく、単にスペシャル的な広がりの印象を与えるためにオーサリングされた音であります。ステレオ音でよくあるケースです。この場合、スペシャル的な広がりは元のチャンネル分けが最適で、回転しようとすると逆効果です。このような種類のコンテンツでは、Spatialization Mode Positionでエミッタオリエンテーションの機能を外すことができます。

## Focus



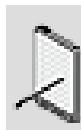
Focusパラメータは、各入力チャンネルのバーチャルソースを1つにまとめて、それから回転やスプレッドを適用します。個別のマルチチャンネルファイルをパンニングする時に、入力チャンネルから出力チャンネルへのリーク（漏れ）を最小限に抑えるために使います。下図は、前述のSpread 70%の例を示しています。Focusを使わないと、例えば、赤の入力チャンネルがCenterチャンネル、Front Rightチャンネル、さらにはFront Leftチャンネルにまでリークしてしまうことが分かります。

Focusが80%の場合は入力チャンネルがまとめられ、赤の入力チャンネルが貢献するのはCenterチャンネルとFront Right チャンネルであって、それも先ほどよりは弱く、さらにFront Leftチャンネルには貢献しないのが分かります。とはいっても、Focusが100%であっても、入力チャンネルは完全に揃っていない限り、必ず2以上の出力チャンネルに貢献することを良く理解しておくべきです。

この問題はアンビソニックスのパンニングでは存在せず、アンビソニックスの設計上、均一とはいえ、既に出力チャンネル全般にリークしています。このため、ソース側のチャンネルコンフィギュレーションがアンビソニックスの時は、Focusを無視します。

リスナーの方位角（Azimuth）から45° にあるエミッタの効果的なパンニング Relative Orientationが90° 、Spreadが70%			
Spread	Focus	エミッタの相対的なポジションと向き	バーチャルソースの配置と出力バスのチャンネルのエネルギー
70% Spread	0% Focus		
70% Spread	80% Focus		

### 平面のソースを、平面のチャンネルコンフィグレーションにパンニングする



#### 注記

この事例の内容などについては[最初の事例](#)を参照してください。

ここまで、平面のソース（4.0、ハイトチャンネルなし）と、平面の出力チャンネルコンフィギュレーション（5.0）を見てきました。同じ内容で、今度は奥行も含

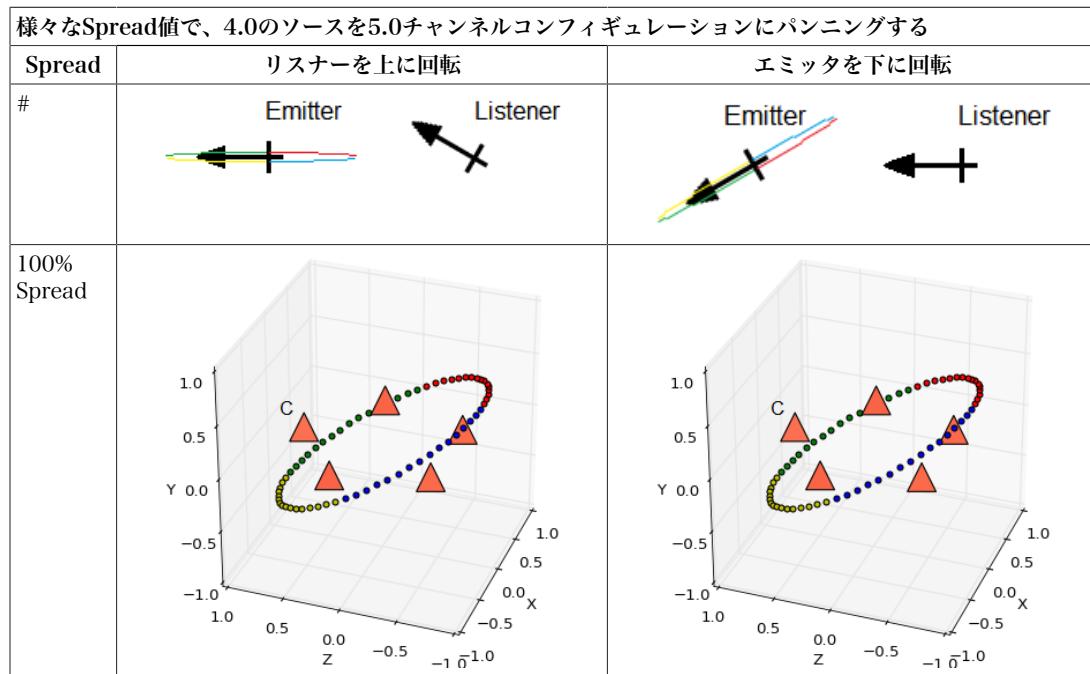
めて検討します。なお、バーチャルソースの座標をスピーカーの平面に投影させることで、パンニングを計算します。

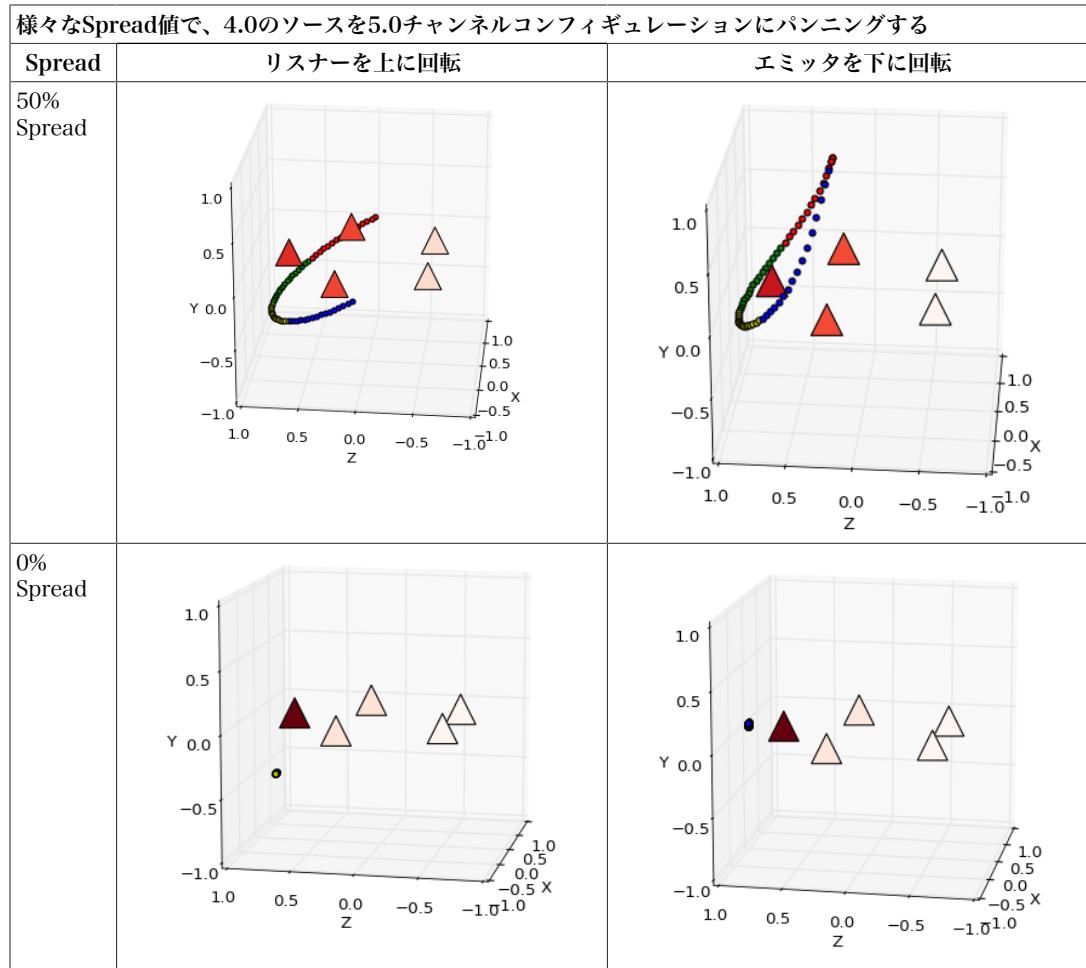
下表の中央の列の「リスナーを上に回転」は、リスナーが上（上空） $30^\circ$  を向いた時の様子を示しています。リスナーが回転することで影響を受けるのは：

- エミッタ音場の相対的な向き (Spreadが100%の場合)
- 点音源の入射角 (Spreadが0%の場合)

点音源 (Spread 0%) が最終的にCenterスピーカーよりも下となるのは、このためです。

右列の「エミッタを下に回転」は、リスナーではなくエミッタを $30^\circ$  下に回転させた様子です。これで影響を受けるのはエミッタ音場の相対的な向きだけで、点音源の入射角は影響を受けません。結果として、0% Spreadでは点音源がリスナーの真正面にきて、中央列のようにリスナーの下ではありません。バーチャルソースが、Spread 100%から0%へとモーフィングする様子が違うのに注目してください。





### 3Dコンフィグレーションにパンニングする



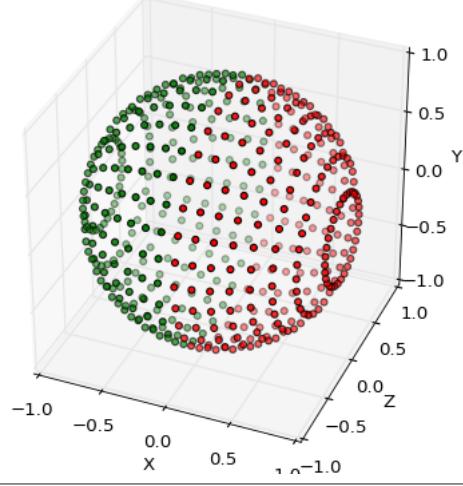
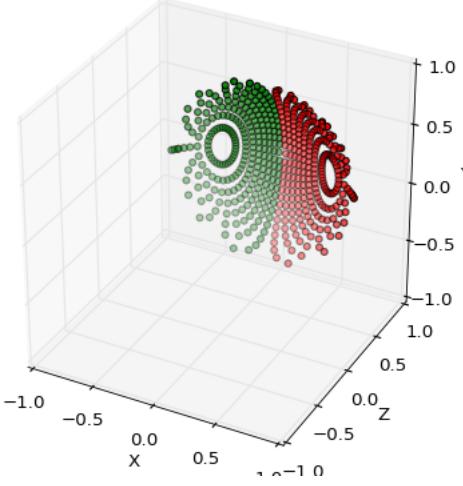
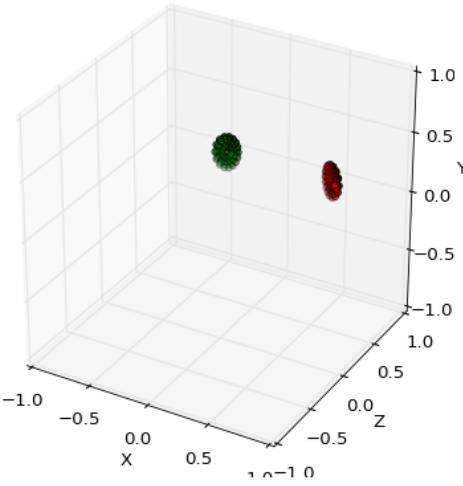
#### 注記

この事例の内容などについては[最初の事例](#)を参照してください。

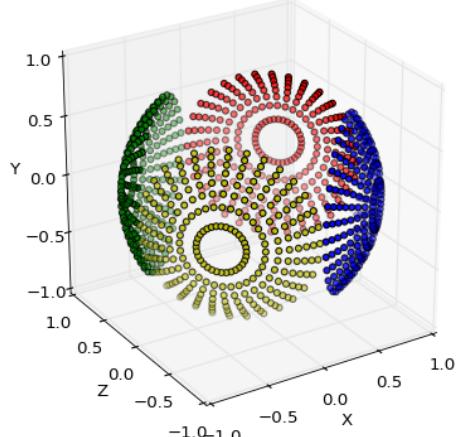
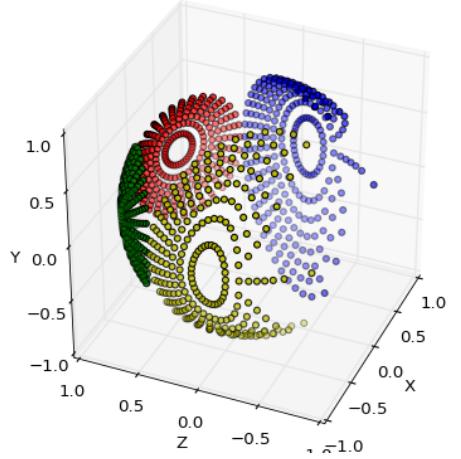
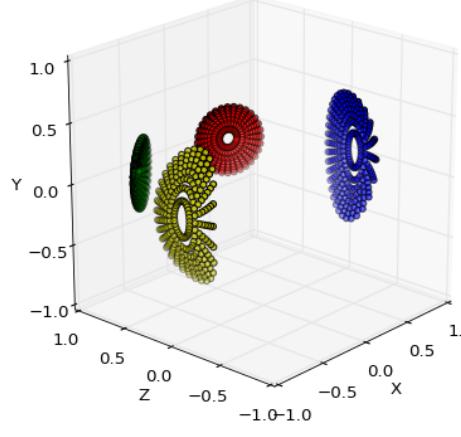
3Dチャンネルコンフィグレーションに対してパンニングする時、つまり上下からくる音を表現できるコンフィギュレーション（7.1.4やアンビソニックスなど）の場合、バーチャルソースを球冠（spherical cap）の形として処理します。下図は、SpreadやFocusの異なるステレオや4.0のソースを示します。

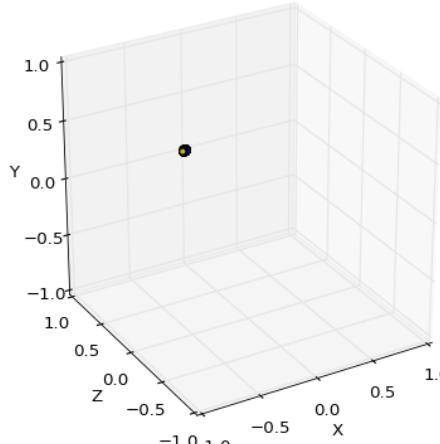
なお前述の通り、3Dポジショニングでパンニングに活用する入力チャンネルは最大7（平面上のチャンネル）なので、それ以外は落とされます。

逆にアンビソニックスでは、厳密に各チャンネルをバーチャルソースのグループにアサインするのではなく、全体のスペーシャル情報を維持します。バーチャルソースの変化（回転、収縮）は、標準コンフィギュレーションと同じ変化なので、下図のイメージ図の内容が、色分け以外は同じです。

ステレオや4.0のソースを3Dコンフィギュレーション（ここには表示されていない）にパンニングする 様々なSpread値やFocus値			
Source config	Spread	Focus	結果
Stereo	100	0	
Stereo	30	0	
Stereo	30	80	

## Positioningの定義

ステレオや4.0のソースを3Dコンフィギュレーション（ここには表示されていない）にパンニングする 様々なSpread値やFocus値			
Source config	Spread	Focus	結果
4.0	100	0	
4.0	70	0	
4.0	70	50	

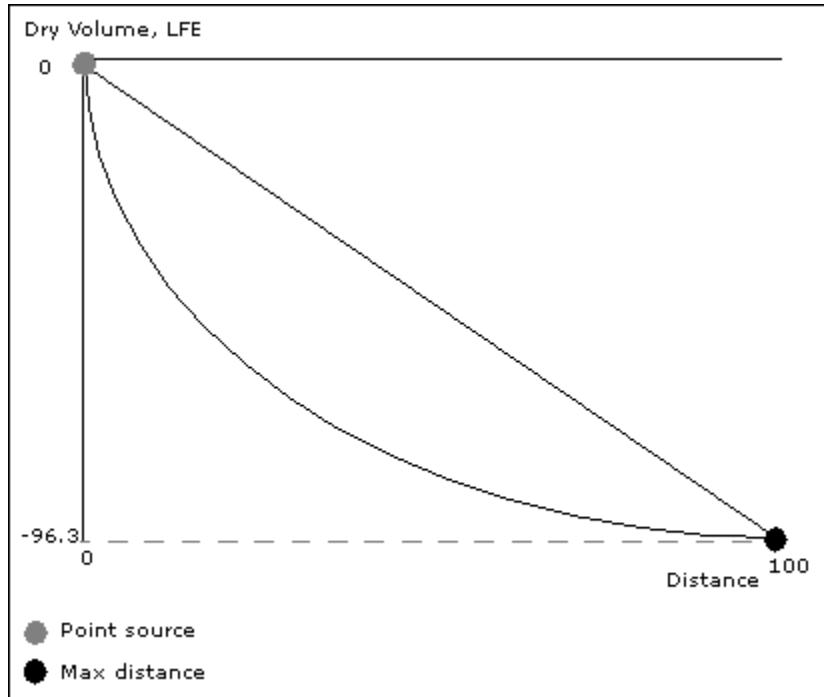
ステレオや4.0のソースを3Dコンフィギュレーション（ここには表示されていない）にパンニングする 様々なSpread値やFocus値			
Source config	Spread	Focus	結果
4.0	0	-	

## 距離で変化する減衰設定

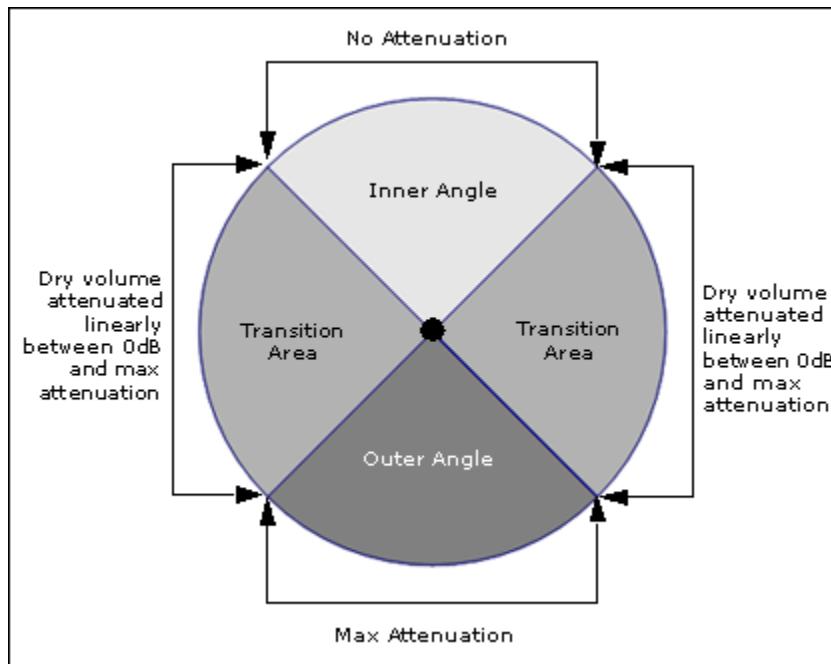
ゲーム中にソースがマイクから離れる時に、自然なロールオフをシミュレーションするには、減衰プロパティを活用します。Wwiseの減衰は、以下の2種類のプロパティ値に基づいています。

- **Distance attenuation** - 距離減衰の意であり、ソースとリスナーの間の距離を元に、信号の強度に影響を与える。
- **Cone attenuation** - コーン減衰の意であり、リスナーに対するゲームオブジェクトの向きを元に、信号の強度に影響を与える。

距離に応じた減衰を、いくつかのカーブを使って設定します。これらのカーブで、Wwiseのプロパティ値であるボリュームなどを、距離にマッピングします。カーブ上の各ポイントのプロパティ値を設定することで、オブジェクトがリスナーから離れていく時のボリュームなどの減衰を、コントロールできます。



コーン減衰は、ソースの前後左右のエリアを角度で定義して設定します。ソース周辺にこれらのエリアを定義することで、オブジェクトのゲーム中の向きに基づいたオブジェクトの減衰をシミュレーションできます。



エフェクトと同様に減衰に関しても、減衰プロパティのインスタンスを作成して、ShareSetでそれらを複数のオブジェクトにシェアさせることができます。

## 減衰インスタンスの管理

減衰インスタンス（Attenuation instances）は、減衰に関するプロパティをまとめたものです。ゲーム中の各種オブジェクトの多くは、同じ減衰プロパティとなるので、減衰インスタンスを作成した上で、Attenuation ShareSetを使って複数のオブジェクトにシェアさせることができます。

ShareSetとは、複数のオブジェクトを登録できる、インスタンスプロパティをまとめたものです。ShareSetに変更を加えると、そのShareSetに登録している全てのオブジェクトに影響します。ShareSetを利用するメリットは、個別のオブジェクトごとに減衰プロパティを変更する必要がないことと、ゲームの貴重なメモリを節約できる点です。

ただし状況によって、減衰設定をシェアさせたくないこともあります。この場合は、カスタマイズした減衰インスタンスを作成して、固有のプロパティ値を設定します。

プロジェクトのAttenuation ShareSetを管理するために、以下のタスクを実行できます。

- ・「[Attenuation ShareSetの作成](#)」
- ・「[Attenuation ShareSetの削除](#)」

### Attenuation ShareSetの作成

オブジェクトの減衰プロパティを設定する前に、まずAttenuation ShareSetを作成する必要があります。Attenuation ShareSetは減衰プロパティ設定をまとめたものです。ShareSetは、同じような減衰プロパティを必要とする全てのオブジェクトに適用できます。

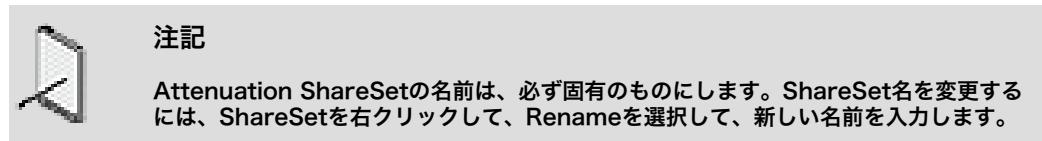
### Attenuation ShareSetを、Project Explorerで作成するには:

1. Project Explorerで、ShareSetsタブに切り替える。
2. Attenuationsセクションで、以下のいずれかを行う。
  - ・ Work UnitまたはVirtual Folderを選択して、Project Explorerツールバーの、Attenuationアイコンをクリックする。
  - ・ Work UnitまたはVirtual Folderを右クリックして、ショートカットメニューで、New child > Attenuationを選択する。

新規ShareSetが、選択したWork UnitまたはVirtual Folderの子として、階層に表示される。

3. 新規ShareSetの名前を入力して、Enterを押す。

新規ShareSetが、Attenuations階層に表示される。



## Attenuation ShareSetを、Property Editorで作成するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Positioningタブに切り替える。
3. セレクタボタン(>>)をクリックして、メニューで、Newを選択する。



### Tip

必ず"2D"ではなく"3D"を選択して、Attenuationを有効にしてセレクタで選択できる  
ようにしてください。

New Attenuationダイアログボックスが開く。

4. 減衰ShareSetを作成する、ワークユニットを選択する。
5. ShareSet名を入力して、OKをクリックする。

新規ShareSetが作成され、現在のオブジェクトに適用される。

## 関連トピック

- [「Attenuation ShareSetの削除」](#)
- [「Attenuation ShareSetを、オブジェクトに適用」](#)
- [「Attenuation ShareSetを、カスタムインスタンスに変換」](#)

## Attenuation ShareSetの削除

不要となったAttenuation ShareSetは、削除できます。なお、削除する前に、そのShareSetを使うオブジェクトがないことを確認してください。ShareSetを削除した場合は、登録している全てのオブジェクトから、自動的に削除されます。



### 注記

Attenuation EditorのShared byボックスに、現在のAttenuation ShareSetを使用する全てのオブジェクトのリストが表示されます。

## Attenuation ShareSetを、削除するには:

1. Project Explorerで、ShareSetsタブに切り替える。
2. Attenuationセクションで、削除するShareSetをクリックする。
3. Deleteキーを押す。

ShareSetが削除され、このShareSetに登録された全てのオブジェクトから、削除される。

## 関連トピック

- [「Attenuation ShareSetの作成」](#)
- [「Attenuation ShareSetを、オブジェクトに適用」](#)
- [「Attenuation ShareSetを、カスタムインスタンスに変換」](#)

### 減衰インスタンスをオブジェクトに適用

プロジェクトのAttenuation ShareSetが作成できたら、オブジェクトに適用し始めます。なお、このプロパティの影響を受けるオブジェクトが1つか複数かによって、カスタマイズした減衰インスタンスを使うのか、シェアされる減衰インスタンスを使うのかが、決まります。

減衰インスタンスを適用する際に、以下を行なうことも可能です。

- ・「Attenuation ShareSetを、オブジェクトに適用」
- ・「Attenuation ShareSetを、カスタムインスタンスに変換」

#### Attenuation ShareSetを、オブジェクトに適用

オブジェクトを減衰させるには、オブジェクトにAttenuation ShareSetを適用します。同じShareSetに複数のオブジェクトを登録できるので、変更が必要となった時に、時間を大幅に節約できます。

#### Attenuation ShareSetを、オブジェクトに適用するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Positioningタブに切り替える。
3. グループボックスAttenuationで、セレクタボタン(>>)をクリックする。

Attenuation ShareSetリストが表示される。

4. 適用させるAttenuation ShareSetを選択する。

該当テキストボックスにShareSet名が表示され、ShareSetがオブジェクトに適用されたことを示す。



#### 関連トピック

- ・「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- ・「Attenuation ShareSetの作成」
- ・「Attenuation ShareSetの削除」
- ・「Attenuation ShareSetを、カスタムインスタンスに変換」

#### Attenuation ShareSetを、カスタムインスタンスに変換

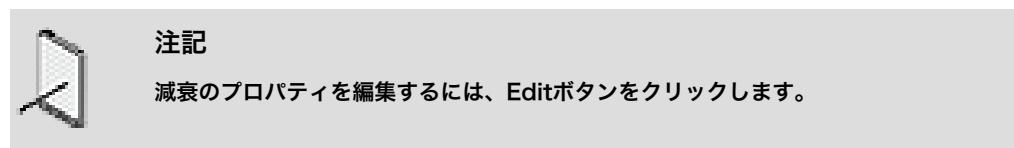
WwiseのAttenuationインスタンスは、以下の2種類に分類できます:

- ・カスタム減衰インスタンスは、階層にある1つのオブジェクトに適用します。カスタムインスタンスのプロパティを変更すると、この1つのオブジェクトだけが影響を受けます。
- ・Attenuation ShareSetsはプロジェクト階層にある複数のオブジェクトに適用します。ShareSetのプロパティを変更すると、そのShareSetを使用する全てのオブジェクトが、影響を受けます。

減衰インスタンスを作成すると、デフォルトで全てShareSetとなります。ただし、目的に応じて、Attenuation ShareSetをカスタム減衰インスタンスに変換できます。変更した後に加えた修正は、減衰インスタンスに登録された1つのオブジェクトだけに適用されます。

減衰インスタンスを、オブジェクトに適用するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Positioningタブに切り替える。
3. Modeリストから、以下のオプションを1つ選択する。
  - Use ShareSetsは、現在のオブジェクトに、ShareSetを適用する。
  - Define customは、現在のオブジェクトに、カスタム減衰インスタンスを適用する。



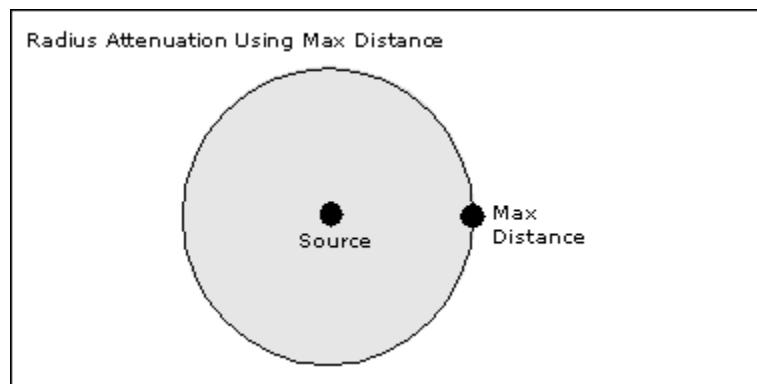
### 関連トピック

- [「Attenuation ShareSetの作成」](#)
- [「Attenuation ShareSetの削除」](#)
- [「Attenuation ShareSetを、オブジェクトに適用」](#)

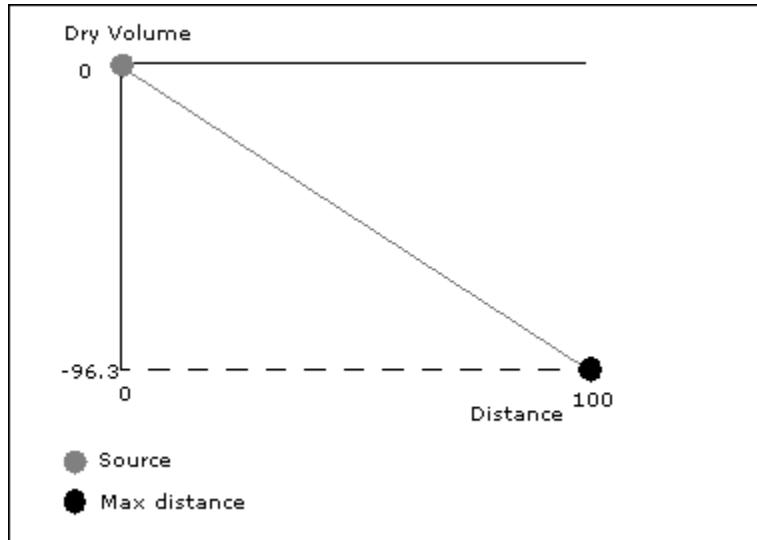
### オブジェクトプロパティの減衰カーブ

Wwiseのオブジェクトを減衰させるには、Wwiseで設定するボリュームやローパスフィルタなど特定のプロパティと、ゲーム中のリスナーからポイントソースまでの距離に基づく関係を定義する、いくつかのカーブを作成します。

どのカーブにもMax distance値が設定され、これでオブジェクトの最大減衰が起きるポイントが定義されます。サウンドは全方向性ソースから発散されるので、Max distance値はソースの周りの球の半径です。

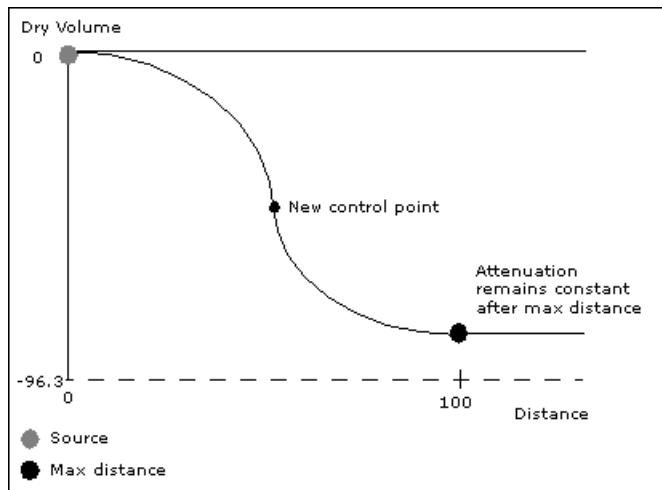


デフォルトで、オブジェクトのソースから最大距離まで、線形補間を使って減衰が適用されます。



多くの場合はこれで充分ですが、より高度なカーブが必要となるオブジェクトもあります。減衰カーブにコントロールポイントを追加すると、コントロール範囲が広がります。コントロールポイントで減衰カーブを分けて、オブジェクトの減衰をより細かく設定できます。

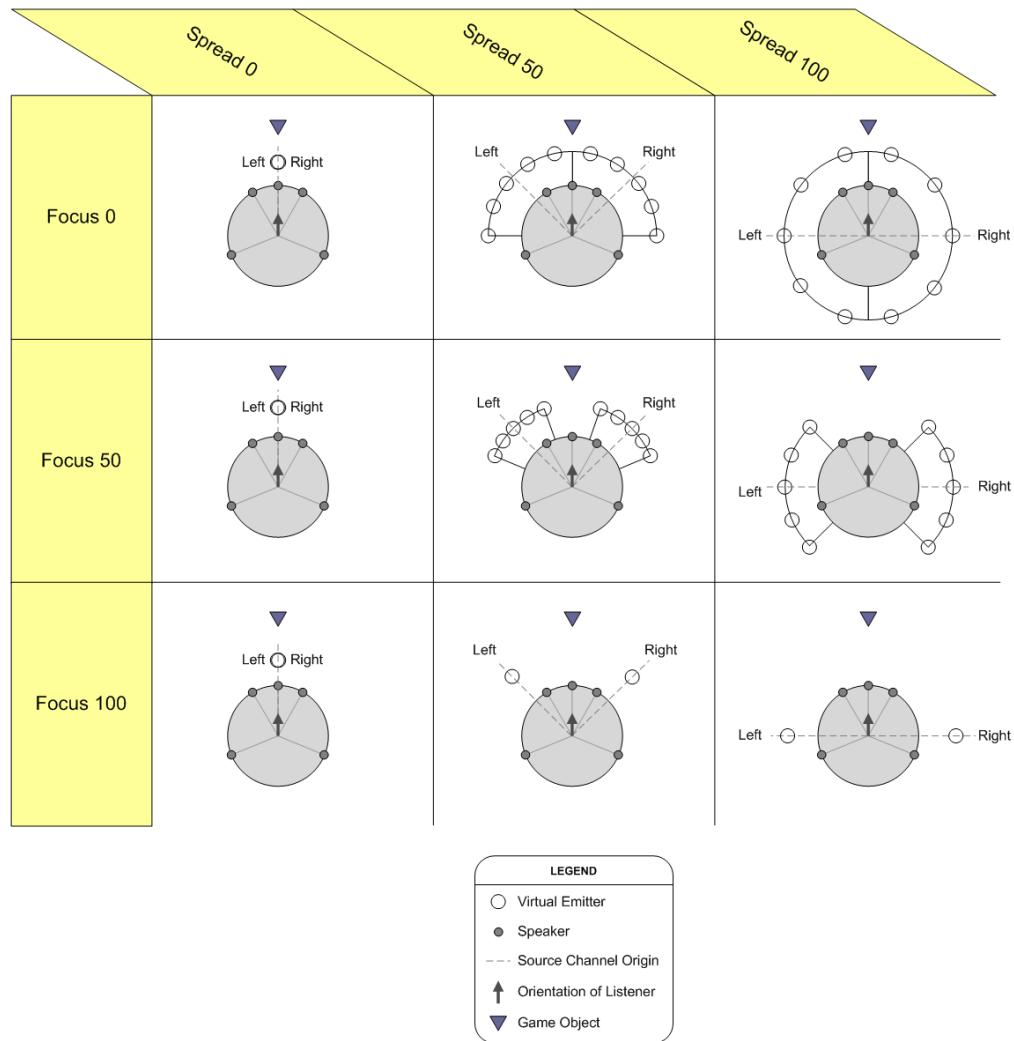
カーブをさらに細かく複雑に設定するには、カーブセグメントの形状を調整します。カーブセグメントとは、カーブ上の任意の2点のコントロールポイントの間の部分を指します。カーブ形状として、直線、定数、対数関数、指数関数、S型カーブなど、様々な選択肢があります。カーブ形状の指定などグラフ上での作業に仕方については[43章グラフィビューについて](#)を参照してください。



Wwiseで設定する以下のプロパティに関して、減衰カーブを設定できます。

- **Output Bus volume:** オーディオアウトプットバスにルーティングされる信号の、減衰または振幅。
- **Auxiliary Send Volumes:** Game-defined、またはUser-defined Auxiliary Busに送られる信号の、減衰または振幅。

- **Low-pass filter:** 指定値に基づいて高周波を減衰させる、リカーシブ（再帰）フィルタ。ローパスフィルタの単位は、適用するローパスフィルタ率を表し、「0」はローパスフィルタなし（信号に影響なし）を表し、「100」は最大減衰を表す。
- **High-pass filter:** 指定値に基づいて低周波を減衰させる、リカーシブ（再帰）フィルタ。ハイパスフィルタの単位は、適用するハイパスフィルタ率を表し、「0」はハイパスフィルタなし（信号に影響なし）を表し、「100」は最大減衰を表す。
- **Spread:** オーディオやモーションが隣のスピーカーやモーターに広がる量または割合を指定する機能で、サウンドやモーションが距離に合わせて、スプレッド値が低い時は点音源、高い値では完全に拡散した伝搬へと変わる。スプレッド値が「0」の時は、スピーカーの隣に位置するエミットソースのチャンネルが、そのスピーカーだけで再生されることを示す。一方「100」の時は、そのエミットソースのチャンネルが拡散され、全てのスピーカーから聞こえる（または感じられる）。
- **Focus:** Spread値によって生じた複数のバーチャルエミッタを集約するために使う値（%）。そFocusが0%のときは、バーチャルエミッタに変更はないが、数値が高くなると、それぞれのバーチャルポイントがソースチャンネル元の周りに近づいていく。



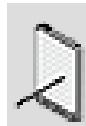
なお、減衰プロパティ値は相対値であるため、減衰値は、そのオブジェクトの既存プロパティ値に追加されます。

### 減衰カーブを設定するには:

- Property Editorにオブジェクトをロードして、Positioningタブに切り替える。
- AttenuationでEditをクリックする。

Attenuation Editorが開き、選択した減衰インスタンスのプロパティ設定が表示される。

- テキストボックスMax distanceで、サウンドが最大減衰に到達する、ソースポイントからの距離を指定する。

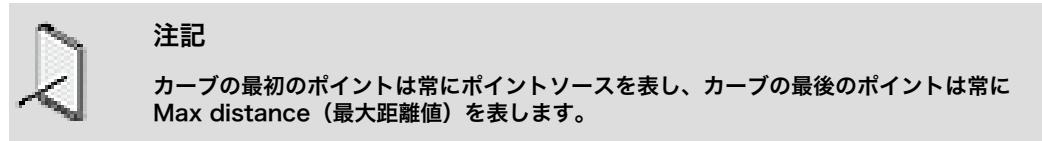


#### 注記

Max distance（最大距離値）に達した後は、減衰設定は一定となります。

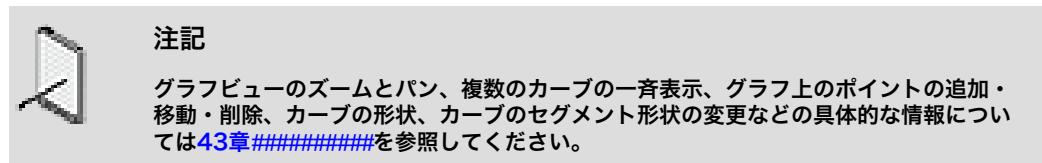
4. グループボックス**Curves**で、リストから**Output Bus Volume**のカーブを選択する。

**Output Bus Volume**のデフォルトカーブが、グラフビューに表示される。



5. **Output Bus Volume**の減衰カーブを操作するには、以下のいずれかを行う：

- ・カーブ上にポイントを追加する。
- ・ポイントを別の場所にドラッグするか、XY座標のボックスに具体的な数値を入力する。
- ・各カーブセグメントの形状を決める。



6. 残りのカーブに関して、**Curve**リストで、以下のオプションを1つ選択する。

- ・**None**は、該当プロパティに減衰カーブを適用させない。Noneを選択すると、該当するプロパティが減衰せず、全開の強度で維持される。
- ・**Use Output Bus Volume**は、該当プロパティに、同じ**Output Bus Volume**カーブを適用させる。なお、このオプションを選択できるのは、**Auxiliary send volumes** カーブだけである。
- ・**Custom**は、該当プロパティのために、カスタマイズした減衰カーブを作成する。

## 関連トピック

- ・「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- ・「3Dオブジェクトの活用」
- ・「距離で変化する減衰設定」
- ・「オブジェクトプロパティの減衰カーブ」
- ・「コーン型境界線による、向きのシミュレーション」
- ・「センタースピーカーの、オーディオ信号のルーティング」
- ・「グラフビューのズームとパン」
- ・「コントロールポイントの追加」
- ・「グラフビューの単位の設定」
- ・「コントロールポイント間の、カーブ形状の指定」

## コーン型境界線による、向きのシミュレーション

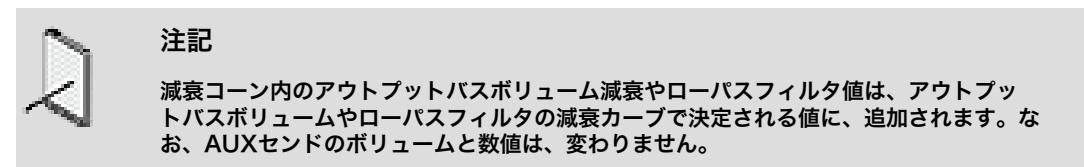
Wwiseにおけるサウンドは、デフォルトで全方向性ソースから発散されます。しかし実際のサウンドは、一般的に何らかの方向性を有します。Wwiseでサウンドの方

向性をシミュレーションするために、サウンドコーンを利用します。サウンドコーンで様々な角度を設定することで、サウンドが特定方向に伝搬する様子をシミュレーションできます。設定した角度の外部にリスナーが移動すると、アウトプットバスボリューム（ドライコンポーネントとして作用する）が減衰します。3D game-defined positioningでは、ゲームのライブデータを使うため、サウンドコーン（Sound cone）の方角が最終的にはゲームオブジェクトの向きによってコントロールされます。

サウンドコーン内のエリアは、以下のアングル（角度）で決定されます。

- **Inner angle** - 内側の角度のことであり、アウトプットバスボリュームの減衰エフェクトや、ローパスフィルタのエフェクトが発生しないエリアを定義する角度。
- **Outer angle** - 外側の角度のことであり、アウトプットバスボリュームの減衰エフェクトとローパスフィルタのエフェクトが、最大レベルで維持されるエリアを定義する角度。

トランジションエリア、つまりインナーアングルとアウターイングルの境界線の間のエリアは、アウトプットバスボリュームのロールオフが起きます。インナーアングルの境界線では減衰が発生せず、アウターイングルの境界線では最大減衰値に到達するので、両者の間で線形補間を使いボリュームを減衰させます。なお、アウターイングルで決まるエリアの中では、ボリューム減衰は常に最大減衰値と等しくなります。



**Cone Attenuation**プロパティ横の**Attenuation Preview**にサウンドコーンの様々なエリアが図示されます。ユーザーがインナーアングルやアウターイングルの角度を変えると、この図も自動的に更新されます。

### モーションの方向性のシミュレーション

モーションは全方向性ソースから発散されますが、状況によってモーションの伝搬を特定の方向（Direction）へ向けてシミュレーションさせることも考えられます。このような場合は、コーン減衰（Cone attenuation）プロパティを利用します。サウンドコーンをモーションに適用すると、サウンドの場合と同じ作用を及ぼしますが、ローパスフィルタの設定はモーションオブジェクトに対して影響しません。サウンドコーンは、リスナーの位置とソースの向きとの関係に基づいて、設定した角度に応じてモーションを減衰させます。

### サウンドの方向性を、サウンドコーンでシミュレーションするには:

1. Property Editorにオブジェクトをロードして、Positioningタブに切り替える。
2. グループボックスAttenuationで、Editを選択する。

Attenuation Editorが開き、選択した減衰インスタンスのプロパティ設定が表示される。

### 3. Cone Attenuationオプションを選択する。

サウンドコーン機能が設定可能となる。

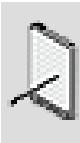
4. テキストボックス**Inner angle**で、アウトプットバスボリュームを減衰させないエリアを決める角度を指定する。
5. テキストボックス**Outer angle**で、アウトプットバスのボリューム減衰とローパスフィルタの効果を最大レベルで維持させるエリアを決める角度を指定する。



#### 注記

インアーアングルとアウターアングルの間のエリアは、トランジションエリアと呼ばれます。このエリアのアウトプットバスボリュームは、減衰の起きないインアーアングルの境界線と、最大減衰値に到達するアウターアングルの境界線の間を、線形補間を用いて減衰させます。

6. テキストボックス**Max attenuation**で、オブジェクトがアウターアングルの範囲内に入った時にアウトプットバスボリュームを減衰させる量を設定する。
7. インアーアングルとアウターアングルの間にあるサウンドに、ローパスフィルタを適用するには、テキストボックス**Low pass filter**に、数値を入力する。ローパスフィルタはリカーシブ（再帰）フィルタで、指定した数値に基づいて高周波を減衰させる。



#### 注記

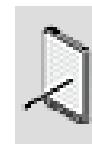
ローパスフィルタの単位は、適用するローパスフィルタ率を表し、「0」はローパスフィルタなし（信号に影響なし）を表し、「100」は最大減衰を表す。

## 関連トピック

- ・「3Dオブジェクトの活用」
- ・「距離で変化する減衰設定」
- ・「オブジェクトプロパティの減衰カーブ」
- ・「センタースピーカーの、オーディオ信号のルーティング」

## サウンドの減衰設定のプレビュー

サウンドの減衰（Attenuation）プロパティが設定できたら、設定内容をWwiseで直接、プレビューできます。サウンドの減衰半径内をプレイヤーが移動する時に、何が聞こえてくるのかが、大まかに確認できます。Wwiseで直接、サウンドの減衰をプレビューできるため、ゲームにオーディオを全く実装していない状態であっても、設定の微調整や工夫が可能となります。



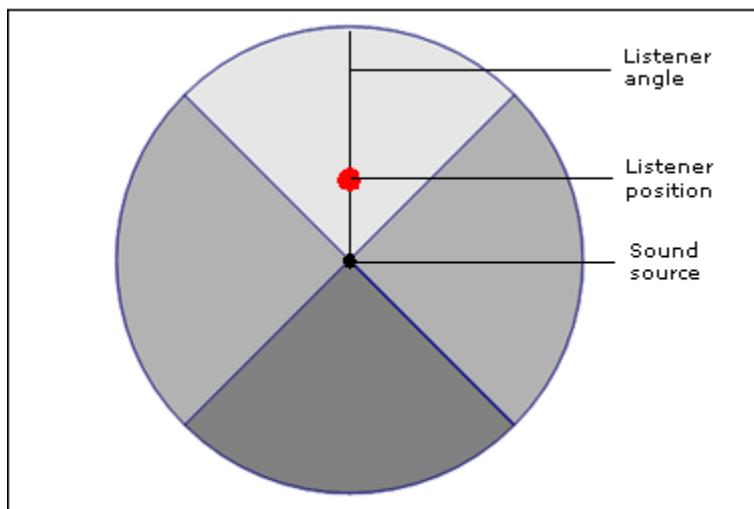
#### 注記

減衰設定のプレビュー表示が可能なのは、3D Game-defined positioningを使用するサウンドに限定されます。

減衰設定をプレビューするには、サウンドの減衰半径内でリスナーの位置を変えられなければなりません。これを、Attenuation Editorにある**Attenuation**

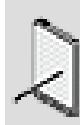
Preview機能で行います。Attenuation Previewはサウンドの減衰半径を図式化した画面で、半径はMax distance（最大距離値）によって決まります。サウンドのソースが、常に円の中心にあります。このサウンドソースに対するリスナーの位置関係を、以下の2種類の操作機能で変更します。

- **赤い点** - サウンドソースに対するリスナーの実際の位置を表す。この赤い点は減衰半径内を自由に動かせる。中心から赤い点を遠ざけると、サウンドのボリュームが次第に小さくなり、円の外周で最大減衰に到達する。つまり、リスナーの位置を指定することで、サウンドソースからリスナーまでの距離が自動的に設定される。
- **黒い細線** - サウンドソースに対するリスナーの角度を表す。デフォルトでリスナーはサウンドソースの真正面（ $0^\circ$  の角度）にあるが、黒い細線を円くドラッグして角度を変更できる。角度を  $360^\circ$ （ $-180^\circ$  から  $180^\circ$  まで）完全に移動させることで、サウンドソースの周りを動く様子をシミュレーションできる。



再生中に角度やポジションを操作して、サウンドに対してリスナーが移動する様子をシミュレーションできます。

リスナーのポジションや角度をリセットするには、Ctrlを押しながら減衰半径の中をクリックするか、Transport ControlのResetセレクタボタン(>>)をクリックして、Reset Positionを選択します。



### 注記

グラフビューの、Distanceカーソルをドラッグすることで、サウンドに対するリスナーの位置を変えることもできます。

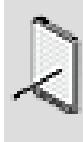
**サウンドの減衰設定をプレビューするには:**

1. Property Editorに3D Game-definedポジショニングを使ったオブジェクトをロードして、Positioningタブに切り替える。
2. AttenuationでEditをクリックする。

Attenuation Editorが開く。

3. Playアイコンをクリックして、サウンドを再生する。
4. リスナーのポジションを変更するには、赤い丸をドラッグして、減衰半径内を自由に移動する。

リスナーが、サウンドソースの周りを動いているかのように聞こえる。減衰レベルは、ソースとの距離の増減に従い、変化する。



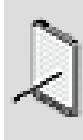
#### 注記

リスナーのポジションを指定することで、サウンドソースからリスナーまでの距離が自動的に設定されます。また、グラフィューのDistanceカーソルをドラッグすることで、リスナーをサウンドソースに近づけることや遠ざけることができます。

5. リスナーの角度だけを変更して距離を変更しない場合は、以下のいずれかを行う。

- ・減衰半径内の任意の場所をクリックする。
- ・黒の細線を、円形にドラッグする。

リスナーが、サウンドソースの周りを動いているかのように聞こえる。また、減衰コーンが使われていれば、リスナーがサウンドコーンの別のエリアへ移動すると、減衰も変化する。なお、角度を変更する時に、距離は変わらない。その結果、サウンドの距離に基づいた減衰は発生しない。



#### 注記

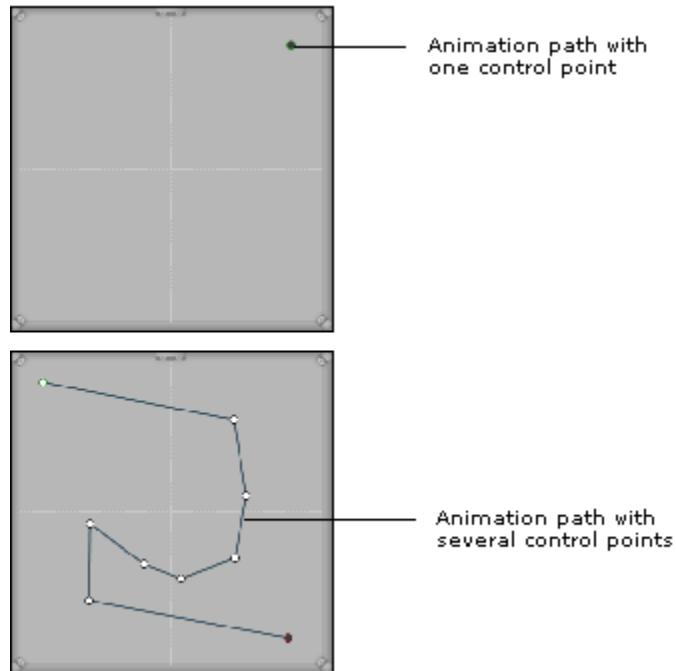
リスナーのポジションや角度をリセットするには、Ctrlを押しながら減衰半径の中をクリックするか、Transport ControlのResetセレクタボタン (>>) をクリックして、Reset Positionを選択します。

## アニメーションパスを利用した、空間ポジショニング

個別のプレイヤーに特化されたエクスペリエンスをプレイヤーに提供したい場合は、User-defined poistioningを使って3Dのオブジェクトの空間ポジション (Spatial positioning) を決めることができます。Wwiseでポジショニングを事前に設定すると、リスナーの位置や向きに関わらず、以下のエクスペリエンスが提供されます。

- ・サウンドが同じスピーカーから聞こえてくる。
- ・モーション（バイブルーション）が同じモーターから伝わってくる。

ポジショニング情報は、アニメーションパス (Animation path) を使って設定します。アニメーションパスは、ある時点のソースの場所を定義する、コントロールポイント（単数または複数）から成ります。複数のポイントを作成すると、オブジェクトが、時間の経過と共にパス上を動きます（アニメートされます）。



### User-defined Spatial Positioningの活用例

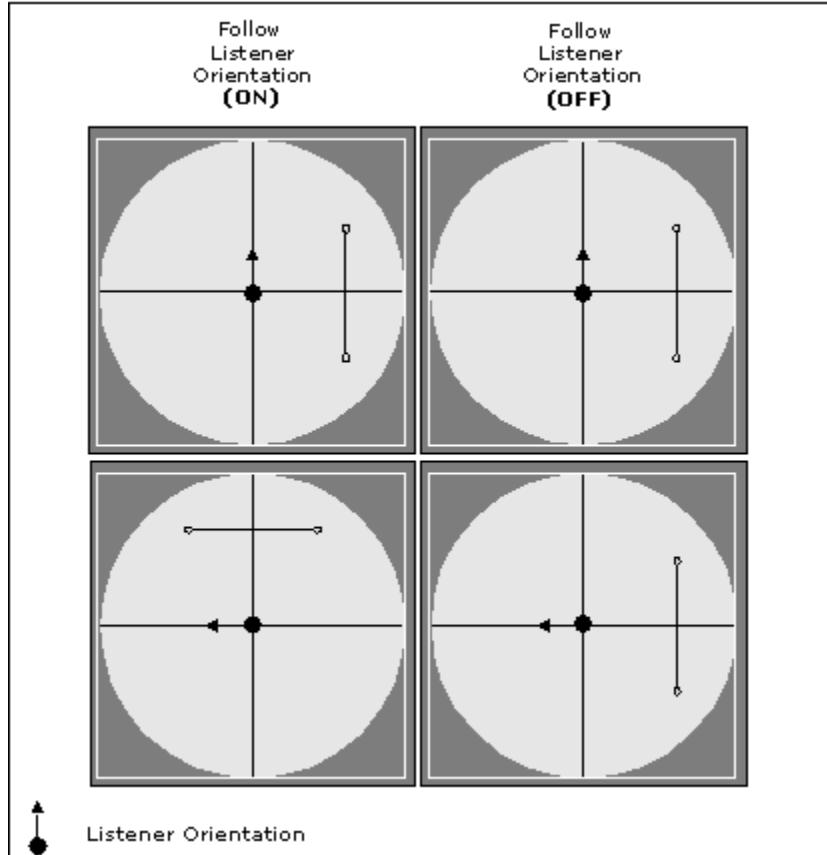
例えば、船上を舞台とするロールプレイングゲームを考えます。アンビエンスの一部として、カモメの鳴き声を使います。カモメが、船の周りを飛び回る様子を表現します。これを実現するには、ポジショニングを3D User-definedに設定してから、サウンドパスを何種類か作成して、鳥が船の周りを飛ぶパターンをシミュレーションします。また、Attenuation ShareSetを採用してサウンドを減衰させれば、カモメのサウンドがさらにリアルになります。

User-defined ポジショニングの使い方は多数あり、例えばジャングルの中の虫、鳥、猿など、固定されていないアンビエントサウンドに利用できます。

### リスナーの向きを追うアニメーションパスの作成

User-defined ポジショニングを使う場合のオプションとして、ゲームのリスナーの向きに合わせて、アニメーションパスの位置をロックすることができます。両者をロックすると、リスナーの向きに関わらず、常に同じスピーカーからサウンドが聞こえてきます。一方ロックしないと、リスナーがパスとは独立して動きます。つまり、リスナーが向きを変えると、サウンドが違うスピーカーから聞こえてくる。

下図は、リスナーの向きが変わる時のアニメーションパスの様子を、Follow Listener Orientationオプションがオンとオフの場合で、示します。



Follow Listener Orientationオプションは、オブジェクトのProperty EditorのPositioningタブで、設定します。Wwiseオーサリングツール内でリスナーの向きを変更することはできないため、レベルエディタまたはゲームでのSoundFrame実装の違いのみを聞くことができます。

### アニメーションパスの設定

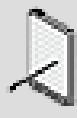
オブジェクトのポジショニングを設定するには、1つのソースポイントまたは複数のコントロールポイント（単数または複数）を使ってパスを作成します。パスを作成すると、オブジェクトがそのパス上を指定された時間をかけて動いて（アニメーションされて）いきます。さらに効果を高めて柔軟にするには、1つのオブジェクトにパスをいくつも設定してそれぞれの時間の長さを変えて、順番通りまたは全くランダムな順番で再生させます。

アニメーションパス（Animation path）は、グラフビューとタイムラインを使って作成します。グラフやタイムラインのポイントの移動と削除や、ズームとパンなどの具体的な情報については[43章グラフビューについて](#)を参照してください。

### アニメーションパスの作成

オブジェクトのポジションを設定する前に、アニメーションパスを作成する必要があります。アニメーションパスは、コントロールポイント（単数または複数）で成立します。複数のポイントを使うと、アニメーションパスが作成され、オブジェクトがパスに沿って指定した時間をかけて移動します。

パスは、グラフビューで作成します。グラフビューには、Max distance（最大距離値）に対する割合（%）として、コントロールポイントの位置が表示されます。つまり、各コントロールポイントの数値は、グラフのどの象限にあるかによって、0から+/- 100の値となります。



### 注記

初めてパスを作成すると、Wwiseが自動的にパス名をObjectName\_01\_Pathとします。このパス名はいつでも、内容が分かるパス名に変更できます。ただし、パス名は必ず固有名前とします。

### アニメーションパスを作成するには:

1. Position Editor (3D User-defined)で、Newをクリックする。

新規アニメーションパスがリストに追加され、グラフビューとタイムラインの両方に、コントロールポイントが1つ追加される。

2. テキストボックスPath Nameに、内容が分かるパス名を入力する。
3. パスの最初のポイントを、グラフビューの任意の場所までドラッグして、配置する。
4. ポイントを追加するには、グラフビューでダブルクリックする。

グラフビューのポイントが、自動的に結ばれて、アニメーションパスが形成される。



### Tip

ポイントの位置を微調整するには、グラフビューでポイントをドラッグするか、X座標とY座標のテキストボックスに座標を直接、入力します。また、ポイント間のタイミングを変更するには、タイムライン上で移動します。

## 関連トピック

- 「パスのポイントの、ポジションのランダム化」
- 「パスの長さ（時間）の変更」
- 「グラフビューで、減衰半径とコーンを表示」
- 「アニメーションパスのリストの順番の変更」
- 「アニメーションパスの削除」
- 「アニメーションパスの再生方法の設定」

### パスのポイントの、ポジションのランダム化

ゲームのリアル感を高めて繰り返しを避けるために、パスにあるポイントのポジションを簡単にランダム化（Randomize）できます。ポイントの水平方向と垂直方法の数値として可能な範囲を、ポイントごとに指定するだけです。ランタイムに、Wwiseがこの範囲からランダムに数値を選択して、コントロールポイントの実際のポジションとして指定します。



### 注記

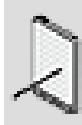
グラフビューにおける各コントロールポイントの位置は、Max distance（最大距離値）に対する割合（%）として、測定されます。Random Rangeで設定した値で、ポイントのX、Y、Z座標を変更するため、同じ単位で設定します。

パスのポイントの、ポジションをランダム化するには：

1. Position Editor (3D User-defined)で、パスリストから、パスを1つ選択する。
2. グループボックスRandom Rangeで、以下の2つのプロパティ値を指定する。

**Horizontal** - ポイントのXY座標（水平ポジション）に対するオフセット値を入力することで、そのポイントで可能な数値の範囲を指定する。ランタイムにWwiseがこの範囲から数値を選択することで、コントロールポイントの水平位置が決まる。

**Vertical** - ポイントのZ座標（垂直ポジション）に対するオフセット値を入力することで、そのポイントで可能な数値の範囲を指定する。ランタイムにWwiseがこの範囲から数値を選択することで、コントロールポイントの垂直位置が決まる。



### 注記

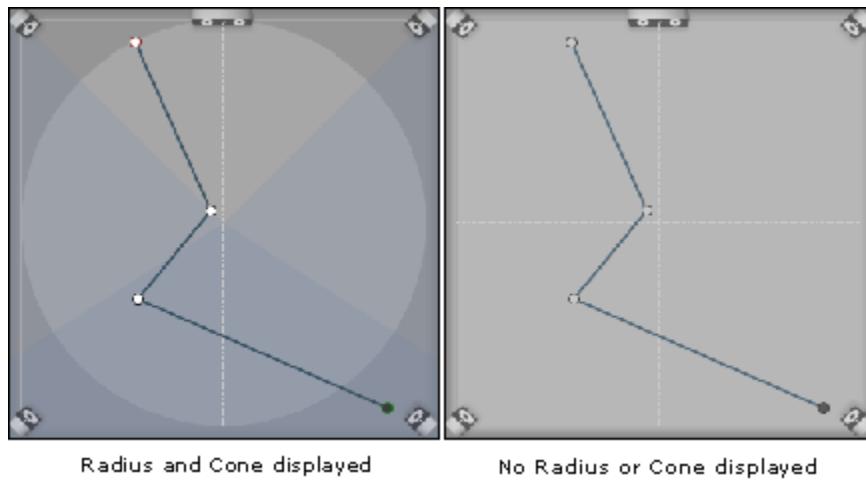
各ポイントの垂直位置（Z値）のデフォルトは、0です。

## 関連トピック

- ・「アニメーションパスの作成」
- ・「パスの長さ（時間）の変更」
- ・「グラフビューで、減衰半径とコーンを表示」
- ・「アニメーションパスのリストの順番の変更」
- ・「アニメーションパスの削除」
- ・「アニメーションパスの再生方法の設定」

### グラフビューで、減衰半径とコーンを表示

信号がリスナーから遠ざかる時の自然なロールオフをシミュレーションするために減衰を使う場合は、グラフビューでパスを設定する時に減衰の各エリアを視覚的に確認できるように、Max distance（最大距離）半径とサウンドコーン（Sound cone）を参考情報として表示できます。



グラフィューの表示を、変更するには:

1. グループボックスDisplay Optionで、以下のオプションを選択する。

Show Radiusは、減衰のMax distance（最大距離）半径を、グラフィューに表示させる。

Show Coneは、コーンの角度（インナーアングルとアウターアングル）を、グラフィューに表示させる。



## 関連トピック

- 「アニメーションパスの作成」
- 「パスのポイントの、ポジションのランダム化」
- 「アニメーションパスのリストの順番の変更」
- 「アニメーションパスの削除」
- 「アニメーションパスの再生方法の設定」

### パスの長さ（時間）の変更

作成したパスの時間の長さを、パス（Path）ごとに設定できます。パスの時間を変えるには、タイムライン（Timeline）の長さを変える必要があります。なおタイムライン上のポイント間が均等ではない（Non-linearモード）場合は、ポイントの間隔を維持してタイムラインをストレッチするのか、ポイントを現在の位置に残したままタイムラインを伸ばすのかを、指定します。

アニメーションパスの時間を、変更するには:

1. Pathリストで、時間を変更したいパスをクリックする。

## 2. Configure Timelineボタンをクリックする。

Configure Timelineダイアログボックスが開く。

### 3. テキストボックスLengthで、選択したパスの新しい時間を指定する。

### 4. 既にタイムラインにコントロールポイントがあり、Non-linearモードであれば、以下のオプションを1つ選択する。

- **Stretch proportionally**は、既存の複数のコントロールポイントの、タイムライン上の位置関係を維持しながら、位置を変える。
- **Preserve key values**は、既存のコントロールポイントのタイムライン上の位置を、変えない。つまり、新しいタイムラインの長さより後のコントロールポイントは、削除される。

### 5. OKをクリックする。

アニメーションパスの長さが更新される。またタイムラインに、新しい時間設定が反映される。

## 関連トピック

- 「[アニメーションパスの作成](#)」
- 「[パスのポイントの、ポジションのランダム化](#)」
- 「[アニメーションパスのリストの順番の変更](#)」
- 「[アニメーションパスの削除](#)」
- 「[アニメーションパスの再生方法の設定](#)」

## アニメーションパスのリストの順番の変更

アニメーションパス (Animation path) リストのパスを特定の順序で表示させたい、つまり再生させたい場合は、順番を変更できます。

## パスリストの順番を、変更するには:

1. Pathリストで、移動するパスをクリックする。
2. 選択したパスを、移動先までドラッグする。



### Tip

パスリストの中でパスを移動すると、リストでパスが挿入される位置が、赤いラインで表示されます。

## 関連トピック

- 「[アニメーションパスの作成](#)」
- 「[パスのポイントの、ポジションのランダム化](#)」
- 「[パスの長さ（時間）の変更](#)」
- 「[アニメーションパスの削除](#)」
- 「[アニメーションパスの再生方法の設定](#)」

### アニメーションパスの削除

不要となったパスは、リストから削除できます。

#### アニメーションパスを、削除するには:

1. Pathリストで、削除するパスをクリックする。
2. Deleteをクリックする。

確認メッセージボックスが表示されます。

3. OKをクリックする。

リストから、パスが削除される。

### 関連トピック

- 「[アニメーションパスの作成](#)」
- 「[パスのポイントの、ポジションのランダム化](#)」
- 「[パスの長さ（時間）の変更](#)」
- 「[アニメーションパスのリストの順番の変更](#)」
- 「[アニメーションパスの再生方法の設定](#)」

### アニメーションパスの再生方法の設定

複数のアニメーションパス (Animation paths) を作成した後は、その再生方法を決めます。コンテナと同様に、作成したパスをランダムに再生するか、パスリストで指定した順番通りに再生するかを、選べます。また、全てのパスを次から次へと再生させるのか、オブジェクトが再生された時にパスを1つだけ再生させるのかも、選べます。

#### アニメーションパスの、再生方法を設定するには:

1. Position Editor (3D User-defined)で、再生方法を、以下のオプションを1つ選択する。

**Sequence**は、リストの最初から最後まで、順にパスを再生する。

**Random**は、全てのパスを再生し終わるまで、ランダムな順番でパスを再生する。

2. グループボックスPlay Mode で、以下のオプションを1つ選択する。

- **Continuous**は、全てのパスを再生し終わるまで、パスを次から次へと再生する。
- **Step**は、オブジェクトが再生される度に、リストのパスを1つだけ、再生する。

3. Continuousを選択すると、オプションとして、LoopやTransition timeを、選択できる。

アニメーションパスのリスト全体を無限に再生するには、Loopオプションを選択する。

1つのパスが終わり、次のパスが始まるまでの間に、線形（リニア）トランジションを追加するには、Transition timeオプションを選択した上で、トランジションにかける時間を、該当するテキストボックスで指定する。

4. Stepを選択すると、オプションとして、Pick new path when sound startsを、選択できる。

Pick new path when sound startsオプションを選択すると、新しいサウンドが再生される度に、そのサウンドが再生イベントでトリガーされたか否かに関わらず、強制的にWwiseで新しいパスを使用する。このオプションは、Continuousに設定したコンテナの中の各サウンドに、違うパスを使用したい時に、特に便利である。

## 関連トピック

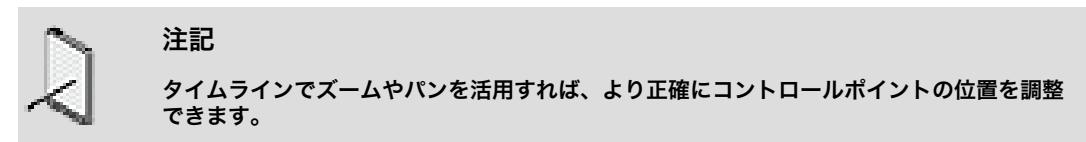
- 「アニメーションパスの作成」
- 「パスのポイントの、ポジションのランダム化」
- 「パスの長さ（時間）の変更」
- 「アニメーションパスのリストの順番の変更」
- 「アニメーションパスの削除」

## タイムラインのポイントの設定

アニメーションパス（Animation paths）を作成する時に、サウンドがパス上を移動するのに時間を設定する必要があります。Position Editor (3D User-defined) タイムラインを使って、コントロールポイントが時間軸上のどの位置にあたるのかを、指定します。



タイムラインのコントロールポイントを動かして、オブジェクトが、パスのポイント間を移動する時のスピードを設定します。ポイントの移動と削除はできますが、タイムラインに直接ポイントを追加することはできません。コントロールポイントを追加できるのは、グラフビューだけです。同時に複数のポイントを選択して、一斉に移動または削除することも可能です。



## タイムラインモードの変更

Position Editor (3D User-defined)のタイムラインのモードは、以下の2種類があります。

- **Linear** - コントロールポイントが、タイムラインに沿って均等な間隔で自動的に配置される。

- **Non-linear** - コントロールポイントが、タイムライン上の自由な位置に配置される。

Linear (直線) モードでは、オブジェクトがパスに沿って移動するにつれ、あるキーから次のキーへと、一定の動作をつくり出します。一方、Non-linear (非直線) モードでは、パス上のポイントからポイントへとオブジェクトが移動する時のスピードを、早めたり遅くすることができます。なお、デフォルトのタイムラインモードはLinearです。

Configure Timelineボタンをクリックすると、タイムラインの特定動作をさらに細かく設定できます。詳しくは「[ポジショニングのタイムラインの設定](#)」を参照してください。

### タイムラインモードを変更するには:

1. Position Editor (3D User-defined)で、以下のいずれかを行う。
  - Linearオプションの選択を外して、タイムラインモードを、Non-linearに設定する。
  - Linearオプションを選択して、タイムラインモードを、Linearに設定する。

### タイムラインのコントロールポイントの選択

タイムラインのコントロールポイント (Control points) を移動または削除する前に、それを選択する必要があります。一度に選択できるコントロールポイントは、1つ、または複数、または全てです。

### タイムラインにある、1つのコントロールポイントを選択するには:

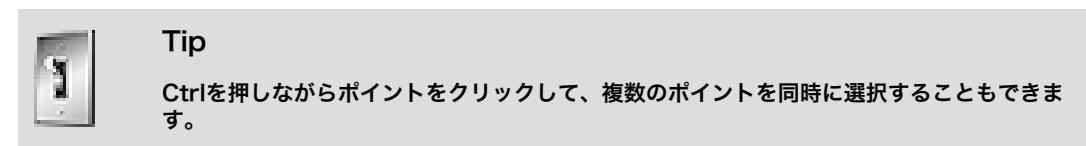
1. タイムラインで、コントロールポイントを1つ、クリックして選択する。

選択されたコントロールポイントが黒色になる。

### タイムラインにある、複数のコントロールポイントを選択するには:

1. タイムラインで、選択する複数のポイントの上をドラッグして、四角で囲みます。

選択されたポイントが黒色になる。



### タイムラインにある、全てのコントロールポイントを選択するには:

1. タイムラインの任意の場所をクリックして、アクティブにする。
2. **Ctrl+A**を押す。

タイムライン上の全てのポイントが選択され、黒色になる。

## 関連トピック

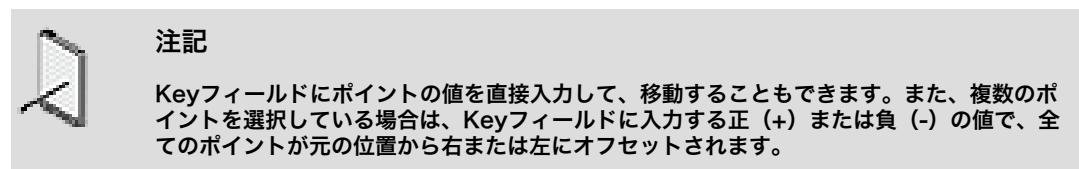
- ・「タイムラインのコントロールポイントの移動」
- ・「タイムラインのコントロールポイントの削除」

### タイムラインのコントロールポイントの移動

タイムラインがNon-linearモードの場合、コントロールポイント (Control points) を移動してポイントとポイントの間の時間を変更できます。その結果、オブジェクトがそのポイントの間に移動する時に、加速や減速する効果があります。なお、コントロールポイントを、その前後にあるコントロールポイントを超えて移動させることはできません。

#### タイムラインで、コントロールポイントを移動するには:

1. タイムラインで、コントロールポイントを単数または複数を選択する。  
選択されたコントロールポイントが黒色になる。
2. ポイントを、タイムラインの新しい位置までドラッグする。



## 関連トピック

- ・「タイムラインのコントロールポイントの選択」
- ・「タイムラインのコントロールポイントの削除」

### タイムラインのコントロールポイントの削除

タイムラインのコントロールポイント (Control points) は、最初のポイント以外を削除できます。

#### カーブのコントロールポイントを削除するには:

1. タイムラインで、ポイントを単数または複数を選択する。  
選択されたコントロールポイントが黒色になる。
  2. Deleteキーを押す。
- コントロールポイントが、タイムラインとグラフビューの両方から削除される。

## 関連トピック

- ・「タイムラインのコントロールポイントの選択」
- ・「タイムラインのコントロールポイントの移動」

## センタースピーカーの、オーディオ信号のルーティング

ゲームプレイに不可欠のサウンドは、プレイヤーに確実に聞こえるようにします。そのためには、2Dまたは3Dのオブジェクトを、センタースピーカーにルーティング (Routing) させる率を自由に設定できます。例えば、レーシングコースにおけるドライバーやアナウンサーの声を全て、センタースピーカーに通し、ゲームの他のサウンドがいかに大きくなろうとも、確実に聞き取れるようにします。

### Center %の設定

Center %で設定した割合に応じて、センタースピーカーにルーティングされる信号の量を、下表に示します。

設定	Center %
0%	ファンタムセンターイメージが作成され、センタースピーカーに信号が送られない。
1-99%	センタースピーカーを通る信号の量を変える。Center %の設定値が上がると、フロントレフトとフロントライトのスピーカーレベルが下がる。
100%	ディスクリートセンターイメージが作成され、全信号（フルシグナル）がセンタースピーカーに送られる。

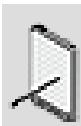


#### 注記

信号が他のスピーカーにスプレッドされる分量は、それぞれのスピーカーとソースの位置関係によって決まります。

### センタースピーカーに、オーディオ信号をルーティングするには:

- Property Editorに、最上位の親オブジェクトを1つロードする。
- Positioningタブに切り替える。



#### 注記

このオブジェクトが最上位のオブジェクトでない場合、Positioningオプションを設定する前にOverride parentオプションを選択する必要があります。

- テキストボックスCenter %で、センタースピーカーを通すボリューム率 (%) を指定する。

### 関連トピック

- 「2Dオブジェクトの活用」
- 「3Dオブジェクトの活用」
- 「距離で変化する減衰設定」
- 「アニメーションパスを利用した、空間ポジショニング」

## Positioningに関する、コツとベストプラクティス

Wwiseでオブジェクトのポジショニングを設定する前に、以下のセクションに目を通して、ゲームの各種オブジェクトのポジショニングを効率的に管理するための事例、コツ、ベストプラクティスなどを参照してください。

## ポジショニングの活用例 (Part 2)

Wwiseで設定できる様々なポジショニングオプションが理解できたところで、前述のファーストパーソン「[ポジショニングの活用例 \(Part 1\)](#)」のサウンドとモーションエフェクトのポジショニングを設定するために、各オプションの使い方を検討します。

- **足音 - ファーストパーソンゲーム**では、メインキャラクターの足音サウンドを常にカメラに添付します。このサウンドは動きや減衰がないので、基本的な2Dポジショニングが適しています。しかし、他の隊員達の足音サウンドは隊員の動きに合わせるために、サウンドを隊員のゲームオブジェクトに添付する必要があります。この場合、空間配置(spatialization)付き3D Game-definedポジショニングが適していますが、減衰は不要です。
- **敵のジャングル基地で灯される、数々のトーチ** - これらのサウンドは、トーチのゲームオブジェクトに添付します。トーチは固定ですが、サウンドエミッタの場所やマイクからの距離がプレイヤーの位置によって変化します。このようなサウンドのシミュレーションには、空間配置付き3D Game-definedポジショニングと、空間配置と減衰の両方を使います。
- **小屋で話し合うテロリスト** - これらのサウンドは、ゲーム環境を自由に動き回るテロリストのゲームオブジェクトに添付します。このようなサウンドのシミュレーションには、空間配置付き3D Game-definedポジショニングと、空間配置と減衰の両方を使います。
- **頭上を飛ぶ蚊** - この例では、蚊が飛ぶ音が聞こえるものの、見えてきません。サウンドエミッタが3D空間を動く必要があり、実際のゲームオブジェクトは存在しない状況では、3D User-defined positioningが適しています。ランダムに再生される複数のサウンドパスに、空間配置と減衰の両方を適用することで、非常にリアルな虫のサウンドを表現できます。
- **本部からの最新情報** - 本部からの無線通信は、特定のゲームオブジェクトに関係なく、サラウンド環境を動き回らないので、2Dポジショニングが適しています。また、通信内容はミッション遂行に不可欠なので、全てのサウンドまたは一部分をセンタースピーカーにルーティングするのも良いでしょう。
- **ミッションに挑む隊員同士の声をひそめたやり** - 隊員達の小声は、各隊員のゲームオブジェクトに添付されるので、3D Game-definedポジショニングが適しています。また、隊員同士の位置が相互に変化するため、何らかの空間配置が必要となりますが、全員がチームとして協力し合うので、お互いのコミュニケーションに減衰を設定する必要はありません。なお、隊員同士のコミュニケーションはミッション遂行の要となるため、全てのサウンドまたは一部分を、センタースピーカーにルーティングするのも良いでしょう。
- **ミッション遂行に成功した後、基地を破壊するために仕掛けた爆弾の爆発** - 隊員達は、爆発の音と衝撃を感じ取ることになります。これを表現する爆発のサウンドやモーションオブジェクトは、爆弾のゲームオブジェクトに添付します。爆弾の位置は固定されていますが、サウンドおよびモーションエミッタの場所とリスナーからの距離がプレイヤーの位置によって変化します。このようなエフェクトのシミュレーションには、3D Game-definedポジショニングと、空間配置と減衰の両方を使います。
- **止むことのない、島の火山の地響き** - 島にいる限り、火山の地響きエフェクトが常に聞こえ、かつ感じられます。サウンドオブジェクトとモーションオブジェクトの両方を、ゲームオブジェクト "Island (島)" に添付するのが一般的です。あ

る程度の減衰を設定すれば、プレイヤー達が島に接近するにつれ地響きをより大きく、またはより強くできます。なお、このサウンドとモーションは動かないので、空間配置は必ずしも必要ありません。

- ・ インタラクティブミュージック - ミュージックは特定のゲームオブジェクトと関連せず、サラウンド環境を動き回る必要もないため、2Dポジショニングが適しています。例えば、フロントスピーカーとリアスピーカーのバランスをとるために、一部のミュージックトラックをパンニングさせます。

下表は、この例の様々なサウンドを作成するために使用できるポジショニングオプションをまとめたものです。

サウンド	2D		3D			
#	パンニングなし	パンニング有り	Game-Defined	User-Defined	Spatialization	Attenuation
隊員の足音	✓	#	#	#	#	#
トーチ	#	#	✓	#	✓	✓
テロリストの会話	#	#	✓	#	✓	✓
蚊の羽音	#	#	#	✓	#	#
本部からの無線通信	✓	#	#	#	#	#
隊員同士のコミュニケーション	#	#	✓	#	✓	#
爆発	#	#	✓	#	✓	✓
火山の地響き	#	#	✓	#	#	✓
インタラクティブミュージック	#	✓	#	#	#	#

この事例は、Wwiseの様々なオプションを活用して、様々な種類のポジショニングや伝搬を作りあげる1つの方法を紹介するものです。実際には、具体的なエフェクト、作成するゲーム、達成したいスペシャルエフェクトなどの諸条件によって、選択するオプションが決まります。

## パフォーマンスの最適化

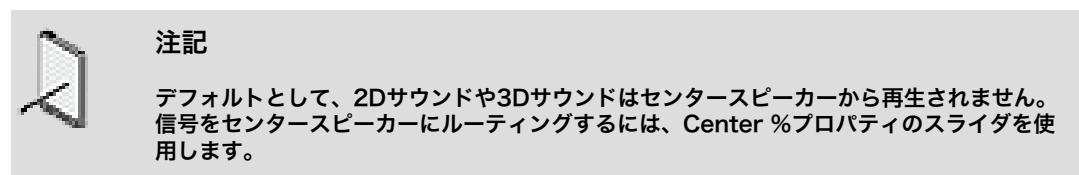
- ・ スプレッド減衰カーブを使わない場合は、モノサウンドを使う - スプレッドカーブを使ってオーディオ信号を広げる予定がなければ、パフォーマンス最適化のためにモノサウンドを使用してください。スプレッドを使わないと、ステレオサウンドの全入力チャンネルが同じポジションにマッピングされ、ダイナミックにレンダリングしなければならないのに対し、モノサウンドを使えば、この処理がオフラインで行われ、ゲームプレイ中のCPU負荷となりません。
- ・ Attenuation Editorのカーブの再利用、カーブ数の削減で、パフォーマンスを改善する - Attenuation Editorでカーブを追加すればするほど、処理能力とメモリに負担がかかります。パフォーマンス向上のため、Output Bus Volumeカーブ (AUXセンドボリューム用) を再利用する、またはカーブを全く使わないという方法があります。
- ・ カーブのポイント（頂点）数を制限して、直線カーブセグメントを使い、パフォーマンスを改善する - カーブにポイントを追加すればするほど、またカーブの形状が複雑になればなるほど、処理能力やメモリに負担がかかります。多くの

場合、カーブにポイントを2、3点設定して直線セグメントを使えば、必要な減衰効果を得られます。

- ShareSetを使い、減衰プロパティの設定をシェアさせる - ゲームの各種オブジェクトの中に、減衰プロパティが似ているものがいくつかあれば、プロパティ設定をShareSetでシェアさせることができます。減衰プロパティ設定をシェアすると、メモリを節約でき、減衰プロパティを変更する時間も短縮できます。
- Positioning TypeにRTPCを用いて、同じような役目には、サウンドを再利用する。例えば、プレイヤーの足音を2Dに設定して、敵の足音を3Dに設定した上で、同じサウンド階層を使えます。これで、デフォルトプールのメモリを大幅に節約できます。

### 具体的なポジショニングシナリオの概要

それでは、具体的なシナリオを検討しながら、Wwiseの様々なポジショニング設定や減衰設定の活用方法を説明します。



#### Scenario 1

- Positioning (配置) : 3D
- Attenuation (減衰) : None
- Spatialization (空間配置) : Off

結果: この設定の結果、2Dサウンドと同じポジショニングとなります。

#### Scenario 2

- Positioning (配置) : 3D
- Attenuation (減衰) : 単純な直線型カーブ



- Spatialization (空間配置) : Off

結果: この設定の結果、サウンドソースからリスナーが離れるにつれ次第に静かになるサウンド（減衰）となりますが、サウンドの位置はオリジナルのサウンドアセットで示される位置と全く同じとなり、ポジショニングやローテーション（空間配置）が一切、適用されません。

#### Scenario 3

- Positioning (配置) : 3D
- Attenuation (減衰) : None

- Spatialization (空間配置) : On

結果: この設定の結果、特定の場所から発信されるサウンドとなります (空間配置)、サウンドソースからリスナーがどれだけ離れても、ボリュームは一切減衰しません (減衰)。

#### Scenario 4

- Positioning (配置) : 3D
- Attenuation (減衰) : 単純な直線型カーブ



- Spread: 単純な直線型カーブ



- Spatialization (空間配置) : On
- Sound source: Mono

結果: この設定の結果は、以下となります。

- リスナーがサウンドソースから遠いと、サウンドが特定の場所から発信され (空間配置)、ボリュームが下げられ (減衰)、主に1つのスピーカーで再生されます (スプレッド)。
- リスナーがサウンドソースに近いと、サウンドが特定の場所から発信され (空間配置)、ほぼフルボリュームとなり (減衰)、ほぼ均等に両スピーカーに配分されます (スプレッド)。

#### Scenario 5

- Positioning (配置) : 3D
- Attenuation (減衰) : 単純な直線型カーブ



- Spread: 単純な直線型カーブ

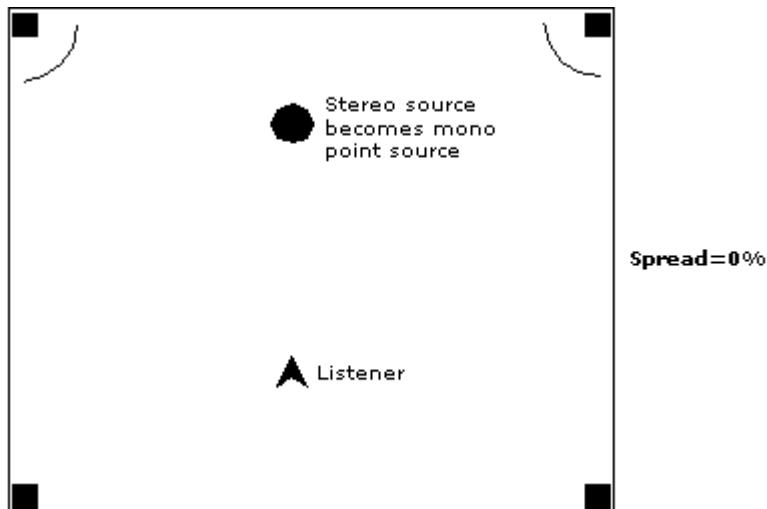


- Spatialization (空間配置) : On
- Sound source: Stereo

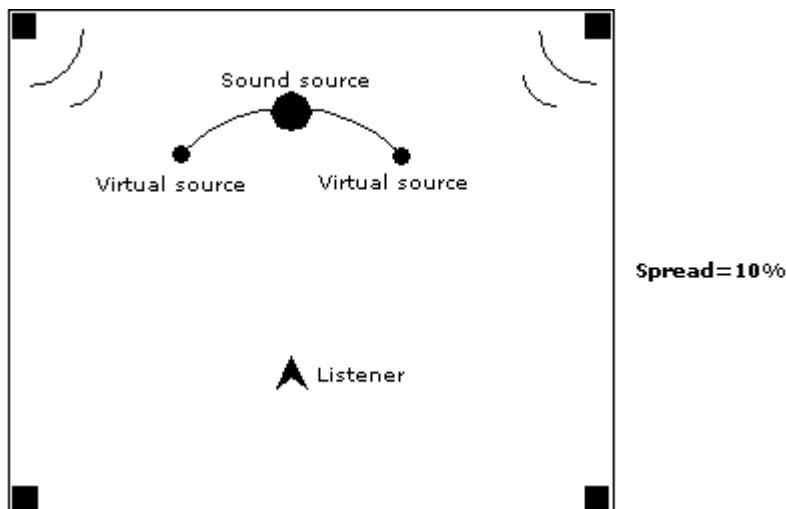
結果: この設定の結果は、以下となります。

- リスナーがサウンドソースから遠いと、サウンドが特定の場所から発信され (空間配置)、ボリュームが小さくなります (減衰)。スプレッドなしで空間配置が

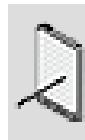
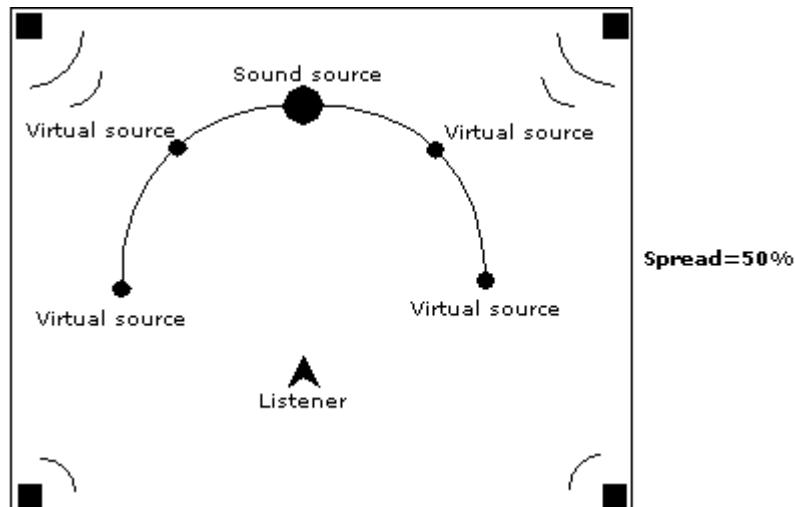
設定されたステレオソースの場合、両チャンネルがフォールドダウンされ、モノの「ポイントソース」となります。スプレッドがない時は、CPUに効率的なモノファイルを使った方が良いのは、このためです。



- スプレッドを使用すると、オリジナルのソースのオフセットとして新しい「バーチャルソース」が設定されます。例えば、スプレッド値が低ければ、計算に基づき、実際のポジションの左右にバーチャルソースに設定し、通常のスプレッドなしのサウンドと全く同じ方法で、わずかに違うポジションから、バーチャルソースの貢献分がスピーカーに追加されます。



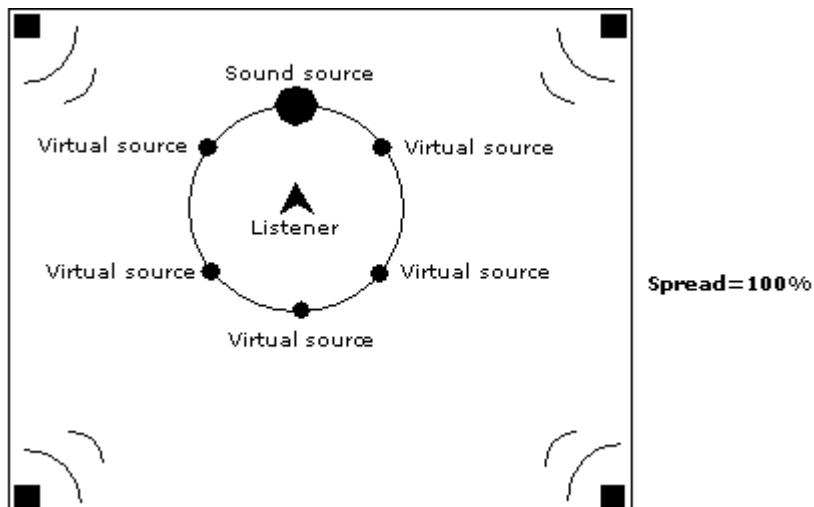
スプレッド値が上昇するにつれ、リスナーの周りにより大きい円弧を描くため、バーチャルソースの数が増えます。当然、バーチャルソースのパワーを実際のソースよりも低くして、パワー合計を一定に保ちます。



#### 注記

これらのソースはボリューム計算のためだけに使われ、実際に新しいサウンドは再生されていません。

- リスナーがサウンドソースに近いと、サウンドが特定の場所から発信され（空間配置）、ほぼフルボリュームとなります（減衰）。高いスプレッド値と組み合わせると、サウンドがあらゆる方向からきます。ステレオサウンドの左右のチャンネルは、別々にスプレッドされます。



- 次に、距離が「0」という特別な状況を考えます。Wwiseの全ての空間配置の計算（及びコーン減衰）は、角度に基づいています。距離が「0」であると、リスナーの向きが前なのか左か右なのか、Wwiseは判断できません。ゲーム中にこのような場面が起きることを避けてください。実際にゲームプレイ中にこの状況が発生してしまうと、Wwiseは単純にステレオサウンドのモノバージョンを作成して、範囲外のボリューム計算を回避します。コーン減衰にも、同じ理論が適用されます。リスナーの向きが不明の場合は、減衰が全く起きていないものとして扱われます。コーンLPFに関しても、同様です。

## チャンネルコンフィギュレーションを理解する

ソース素材、アウトプット機器、バス設定などに応じて、Wwiseパイプラインは様々なレベルでマルチチャンネルコンフィギュレーションに対応しています。なおプラットフォームによって、条件が変わります。



### ダウンミキシング

様々なチャンネルコンフィギュレーションを取り扱う中で最適なパフォーマンスを確保するには、Wwiseのダウンミキシングのアプローチを理解することが重要です。詳しくは[付録A](#)を参照してください。

Wwiseが対応するチャンネルコンフィギュレーションと、それぞれのチャンネル名、スピーカーの推奨角度、オーダー（順序）を、以下に示します。

#### 0.1 (mono)

項目	内容
Low Frequency Effects (LFE)	N/A

#### 1.0 (mono)

項目	内容
Center (C)	0°

#### 2.0 (stereo)

項目	内容
Left (L)	22° -30°
Right (R)	22° -30°

#### 3.0

項目	内容
Left (L)	22° -30°
Right (R)	22° -30°
Center (C)	0°

#### 4.0

項目	内容
Left (L)	22° -30°
Right (R)	22° -30°
Surround Left (SL)	90° -110°
Surround Right (SR)	90° -110°

#### 5.1

項目	内容
Left (L)	22° -30°
Right (R)	22° -30°
Center (C)	0°
Surround Left (SL)	90° -110°
Surround Right (SR)	90° -110°
Low Frequency Effects (LFE)	N/A

7.1	
項目	内容
Left (L)	22° -30°
Right (R)	22° -30°
Center (C)	0°
Surround Left (SL)	90° -110°
Surround Right (SR)	90° -110°
Back Left (BL)	135° -150°
Back Right (BR)	135° -150°
Low Frequency Effects (LFE)	N/A

WwiseはHeightスピーカーを使った標準チャンネルコンフィギュレーションの拡張にも対応しています。提供される各チャンネルの定義とバスで提供される標準コンフィギュレーションの説明を、以下に示します。

Heightチャンネルで拡張された標準コンフィギュレーション	
項目	内容
Left (L)	22° -30°
Right (R)	22° -30°
Center (C)	0°
Surround Left (SL)	90° -110°
Surround Right (SR)	90° -110°
Back Center (BC)	180°
Back Left (BL)	135° -150°
Back Right (BR)	135° -150°
Top (T)	真上
Height Front Left (HFL)	Lと同じで、かつ水平線より上
Height Front Center (HFC)	Cと同じで、かつ水平線より上
Height Front Right (HFR)	Rと同じで、かつ水平線より上
Height Back Left (HBL)	BL（またはSL）と同じで、かつ水平線より上
Height Back Center (HBC)	BCと同じで、かつ水平線より上
Height Back Right (HBR)	BR（またはSR）と同じで、かつ水平線より上
Low Frequency Effects (LFE)	N/A

Height拡張Busの利用できるコンフィギュレーション	
Dolby™ Bus Configurations	
	注記 Wwiseが対応するDolbyコンフィギュレーションとして、5.1サラウンドや7.1サラウンドのコンフィギュレーションにHeightチャンネルを追加したものも含まれます。表示名の第3桁がHeightチャンネルのチャンネル数を表します。
Dolby 5.1.2	L + R + C + SL + SR + HFL + HFR + LFE
Dolby 7.1.2	L + R + C + BL + BR + SL + SR + HFL + HFR + LFE
Dolby 7.1.4	L + R + C + BL + BR + SL + SR + HFL + HFR + HBL + HBR + LFE
Auro Bus Configurations	

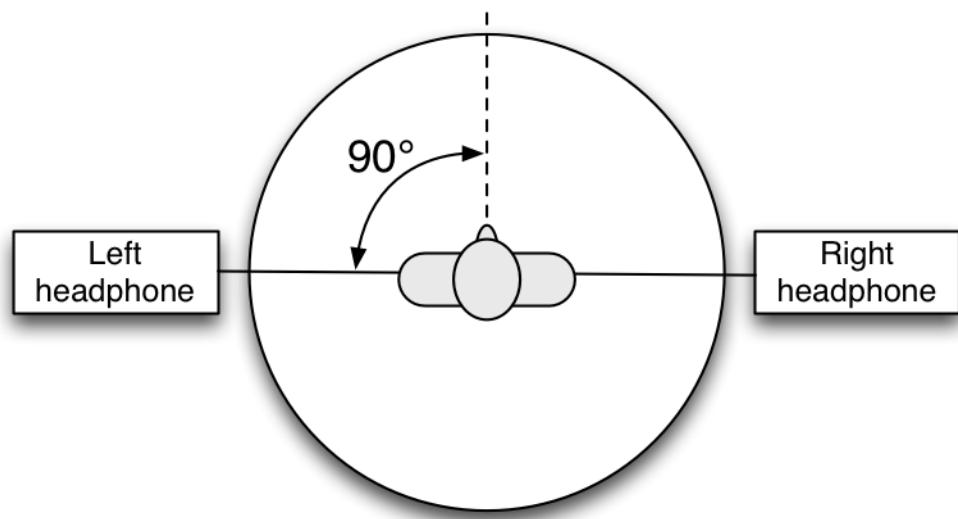
Height拡張Busの利用できるコンフィギュレーション	
	注記
<p>Wwiseが対応するAuroコンフィギュレーションとして、5.1サラウンドや7.1サラウンドのコンフィギュレーションにHeightチャンネルを追加したものも含まれます。表示名の第1桁で表すチャンネル総数が増え、Heightチャンネル分である4から6チャンネルまでが追加されると、9.1から11.1チャンネルと表されます。Auro 13.1は7.1サラウンドコンフィギュレーションにHeightを6チャンネル追加したものです。</p>	
Auro 9.1	L + R + C + SL + SR + HFL + HFR + HBL + HBR + LFE
Auro 10.1	L + R + C + SL + SR + T + HFL + HFR + HBL + HBR + LFE
Auro 11.1	L + R + C + SL + SR + T + HFL + HFC + HFR + HBL + HBR + LFE
Auro 13.1	L + R + C + BL + BR + SL + SR + T + HFL + HFC + HFR + HBL + HBR + LFE

Ambisonics Busの利用できるコンフィギュレーション	
	注記
<p>Ambisonicsはスピーカー設定とは独立したB-formatサウンドフィールド（音場）を使います。以下の表示はB-formatの方向性チャンネルを示し、スピーカーチャンネルではありません。この表示方法はSN3DノーマライゼーションのACNチャンネルオーダーに基づいていますが、MaxNノーマライゼーションのFuMaオーダーと同様に、数字ではなく文字を使用します。詳細は「<a href="#">アンビソニックスコンポーネントのオーダー</a>」を参照。</p>	
Ambisonics 1-1	W-Y-Z-X
Ambisonics 2-2	W-Y-Z-X-V-T-R-S-U
Ambisonics 3-3	W-Y-Z-X-V-T-R-S-U-Q-O-M-K-L-N-P

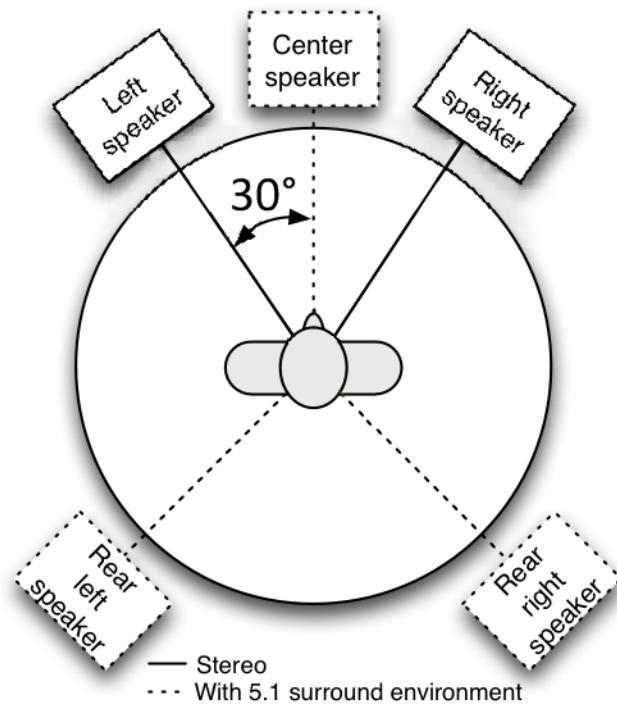
## スピーカーとヘッドフォンのパンニングルール

Wwiseにはヘッドフォン用とスピーカー用の2種類のパンニングルールがあります。デフォルトで全プラットフォームでスピーカーのパンニングルールを採用しますが、例外としてポータブルコンソール（Android、iOSなど）でヘッドフォンのパンニングルールを採用します。2つのモードは微妙に違い、聞く環境に応じてリアルで正確なオーディオを提供するために設定されています。パンニングルールの設定はWwise上で試聴できるほか、ランタイムにゲーム内で設定することも可能です。

**Headphone panning rule**



**Loudspeaker panning rule**



2つのモードを試聴するには、以下を行います:

- メニューバーで、Audio > Stereo Channel Configuration (Speakers) をクリックする。

- メニューバーで、Audio > Stereo Channel Configuration (Headphones) をクリックする。

**注記**

ゲーム内でパンニングルールを設定するには、SDKドキュメンテーションの [AK::SoundEngine::SetPanningRule](#) を参照してください。

## Ambisonicsを使う

Ambisonicsはリスナーの上下リージョンに加えて水平面もカバーするサラウンドサウンドテクニックです。B-formatサウンドフィールド（音場）表現と球面調和関数を使い、スピーカー設定から独立しています。これを使うと、リスナーと共にサウンドを回しやすくなります。球体のサウンド表現が必要な場面、例えばアンビエントサウンドを提供する時やVRを展開する時などに、アンビソニックスが効果的な選択肢となります。

**Tip**

アンビソニックスにご興味があれば、このヘルプの以下の説明以外にも、[Ambisonics in Wwise: Overview](#) というブログ記事も参考にしてください。アンビソニックスの細かい理論や、WwiseでどのようにVRに活用できるかなどが描かれています。また、Wwise Sample Project の中にもアンビソニックスの簡単な例がいくつか含まれていますので、[こちらの説明](#) を参照してください。.

Wwiseを使って簡単にアンビソニックスファイルをインポートしたり、他のコンフィギュレーションからアンビソニックスフォーマットへサブミックスしたりできるので、増えつつあるアンビソニックス対応プラットフォームで再生できるアンビソニックスアウトプットの実現が可能です。

### アンビソニックスファイルを使うには:

- アンビソニックスB-formatで録音したファイル、つまりWAVやAMBとして保存できるファイルを、他のオーディオファイルと同様にインポートする。詳細は「[メディアファイルのインポート](#)」を参照。

**注記**

Wwiseでアンビソニックスファイル（AMBまたはWAV）をインポートする時は、maxNノーマライゼーションの通常のFuMaオーダーを使っているものとみなします。

- 他のサウンドオブジェクトの場合と同様に、適切なAudio BusをOutput Busに指定する。詳細は「[オブジェクトのアウトプットルーティングの指定](#)」を参照。

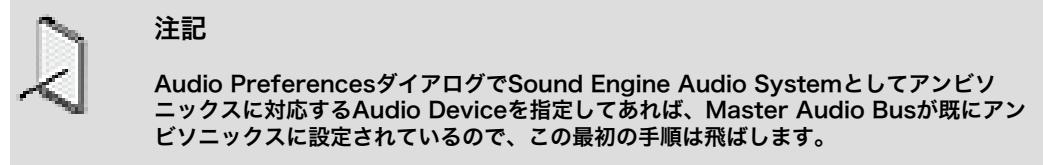
Wwiseが必要に応じてチャンネルをミックスする。詳細は「[アンビソニックスのサブミックス](#)」を参照。

### アンビソニックスをアウトプットするには:

- Audio Busに以下のアンビソニックスチャンネルコンフィギュレーションを設定する:

- Ambisonics 1-1 (第1オーダー、4チャンネル)
  - Ambisonics 2-2 (第2オーダー、9チャンネル)
  - Ambisonics 3-3 (第3オーダー、16チャンネル)
- チャンネルコンフィギュレーションの詳細は[利用できるAmbisonicsコンフィギュレーション表](#)を参照。

上記で設定したAudio Busを経由してアウトプットされるソースファイルを、Wwiseがミックスしてアンビソニックスとしてアウトプットする。



2. 最終的なアウトプットデバイスとして、アンビソニックスのチャンネルコンフィギュレーションに対応するデバイス、例えば3Dバイノーラルシステムなどを指定する。

アンビソニックス対応アウトプットではない場合は、Wwiseがそのチャンネルアウトプットを自動的に適切な標準アウトプットにサブミックスする。詳細は[「アンビソニックスのサブミックス」](#)を参照。

### Wwiseエンジンが、最終アウトプットとしてHOAアウトプットを出力するように設定する

上記の通り、Channel Configurationの設定がAmbisonics #-#となっているバスがある場合、Wwiseはアンビソニック信号とHOA (higher-order ambisonic) 信号を出します。これらの信号は、アンビソニックバスの親バスのチャンネルコンフィギュレーションが標準コンフィギュレーションに設定されていると、それに合わせてデコードされ、ステレオや5.1などの標準マルチチャンネル信号に変換されます。

Master Audio Busは最終ポイントのコンフィギュレーションを継承しますが、これはプラットフォームによって異なり、通常は自動的に検知されます。現時点では、多くのプラットフォームで許容されるアウトプットコンフィギュレーションが限定されています。典型的なのは、ステレオ、5.1、7.1などです。つまり、一般的にMaster Audio Busは標準的なチャンネルコンフィギュレーションなので、子のアンビソニックバスも強制的にこのコンフィギュレーションにデコードされ、WwiseはHOAを直接アウトプットしません。

ただし、カスタマイズしたAudio Deviceプラグインを書いて、アンビソニックコンフィギュレーションを宣言させることも可能です。この場合、Master Audio Busのチャンネルコンフィギュレーションがアンビソニックになり、WwiseはHOA信号をデコードせずに維持します。

別の方法として、WwiseにHOAを有するファイルにダンプさせるだけで充分であれば、単純に[Wwise Recorder](#)をアンビソニックバスにインサートできます。生成されるファイルはAMBフォーマット (FuMa) となり、アンビソニックオーダーは、Recorderをホストするバスと同じになります。

## 関連トピック

- 「アンビソニックスコンポーネントのオーダー」
- 「Ambisonicsをプラグインで使う」
- 「アンビソニックスのサブミックス」
- 「アンビソニックスの制約事項」

### アンビソニックスコンポーネントのオーダー

アンビソニックスコンポーネントの主要なオーダーフォーマットは、Furse-Malham（別名FuMa）とAmbisonics Channel Number（別名ACN）の2つです。下図の通り、前者はアルファベットを使って各グループ内のオーダー（順序）をアルファベット順に表示し、まず最初にW（オムニ）チャンネルがあり、次にその右下、左下、下中央と続き、さらに次のオーダーに移り最初にRがあり、次にその右、左、更にその右側と左側に続き、そして次のオーダーに移り同様のパターンをたどります。一方、後者は非常に分かりやすい「左から右へ」の数字を使ったオーダーをたどります。

一番上のモノブロックから第1、第2、第3オーダーのベースを示すこの図を見ると分かる通り、各アンビソニックスオーダーはピラミッド形状であり、それぞれ4チャンネル、9チャンネル、16チャンネルあります。

Ambisonic Order	FuMa	ACN	
Mono	W	0	
1 <sup>st</sup> -order full sphere	Y Z X	1 2 3	
2 <sup>nd</sup> -order full sphere	V T R S U	4 5 6 7 8	
3 <sup>rd</sup> -order full sphere	Q O M K L N P	9 10 11 12 13 14 15	
<i>Read alphabetically in groupings of three orders (W,X,Y,Z)(R,S,T,U,V)(K,L,M...)</i>		<i>Read numerically (0,1,2,3,4,5,6...)</i>	

### Wwiseにおけるアンビソニックスのオーダー

Wwise内のアンビソニックスファイルのオーダーは、最初のインポートやその後のエクスポートとは異なります。Wwiseは、インポートされるアンビソニックスファイルが従来のFuMaオーダーに従うものと予測しながら、一旦Wwiseに取り込んだ時点で新しいACNオーダーとSN3Dノーマライゼーションにファイルを変換します。しかしFuMaのアルファベット割当は維持されます。つまり上図にあるFuMaオーダーは左から右に読まれ、W Y Z X V T R S U Q O M K L N Pとなります。オーサリングツールのSource Editorにあるメーターやグラフビューでは、このハイブリッド型ACNオーダー表示が採用されています。

このおかげで、カスタム化プラグインの開発にWwiseで変換したACNオーダーが対応できます。これにもかかわらず、アンビソニックス対応バスでWwise Recorderプラグインを使って録音されたファイルには、FuMaオーダーが使われます（結果としてWwiseにインポートできるファイルが作成されます）。

## 関連トピック

- ・「Ambisonicsをプラグインで使う」
- ・「アンビソニックスのサブミックス」
- ・「アンビソニックスの制約事項」

### Ambisonicsをプラグインで使う

WwiseプラグインAPIでアンビソニックスのチャンネルコンフィギュレーションは `AK_ChannelConfigType_Ambisonic` として露出されます。その結果、デベロッパは カスタム化プラグインの開発においてアンビソニックベッドを活用できます。

### AmbisonicsをEffectで使う

Wwiseと一緒に提供されるEffectのうちチャンネルを独立処理しているWwise Compressor、Wwise Guitar Distortion、Wwise Parametric EQなどのプラグインは、アンビソニックスに対応しています。



一方、リバーブのチャンネル処理は異なります。リバーブでは一般的にインプットチャンネルをダウンミックスしてリバーブをかけてから、概ね非相関化 (de-correlated) された信号をミックスすることでリバーブ処理後のアウトプットをアップミックスして、アウトプットチャンネルに送ります。Wwise RoomVerb と Wwise Convolution Reverb だけがアンビソニックスのボイスやバスに対応しています。それらはW (オムニ) チャンネルを取り、それを残響させ、無相関信号を無指向性チャネルおよび指向性チャネルに混合することによって、それをアップミックスします。Wwise Matrix Reverbをアンビソニックスのボイスやバスで使うと、初期化が失敗してしまいます。

## 関連トピック

- ・「アンビソニックスコンポーネントのオーダー」
- ・「アンビソニックスのサブミックス」
- ・「アンビソニックスの制約事項」

### アンビソニックスの制約事項

アンビソニックスはリスナーに向かってくる空間音響 (spatial audio) を表現します。このため、シーン内にあるオブジェクトの正確な表現ができません。そして、このようなオブジェクト周辺を移動したり周りで動いたりする表現が難しくなります。サウンドイベントが常に奥の方から発信されるように聞こえてしまいます。

これはWwise内のアンビソニックス動作に2つの重大な影響を与えます。

- Attenuationカーブを設定すると、設定対象のゲームオブジェクトの位置から遠ざかる時にサウンドフィールド全体のボリュームが下がります。この場合、サウンドフィールドを出入りしようとするときに状況が複雑になります。

- サウンドフィールドの片側に移動しても、その方向に向いているソースが反対方向に向いているソースよりも大きく聞こえません。

### アンビソニックスとVBAPの比較

Wwiseに実装された3Dサウンドのパンニングアルゴリズムは広く使われるVBAPアルゴリズムをデフォルトとし、全体のエネルギー消費を一定に保ちつつサウンド精度を最大限にします。しかし同時にエネルギー・プレッドの変化が制限されてしまいます。つまり、バーチャルソースがラウドスピーカーとアラインしているとエネルギー・プレッドが最小限となり、ラウドスピーカーの弧の中心（標準コンフィギュレーションの場合）や三角の中心（Heightチャンネルのあるコンフィギュレーションの場合）で最大限となります。

逆にアンビソニックスのエネルギー・プレッドはソース方向やラウドスピーカー配置に関わらず、常に一定です。アンビソニックスオーダーが低ければ低いほど、スプレッドが大きくなります。このためアンビソニックスの第1オーダーは非常に不正確であり、アンビソニックスの第3オーダーは比較的正確であるものの、VBAPと比べると正確性に欠けます。

美的観点から、モノソースをアンビソニックスベッドにエンコードできます。

### 関連トピック

- 「アンビソニックスコンポーネントのオーダー」
- 「Ambisonicsをプラグインで使う」
- 「アンビソニックスのサブミックス」

### アンビソニックスのサブミックス

アンビソニックスのソースポジショニングは、ミキシングの瞬間のゲームオブジェクトやリスナーのポジショニングに基づいて決まり、これは他のチャンネルコンフィギュレーションと変わりません。このためアンビソニックス設定のサウンドオブジェクトを他のコンフィギュレーションとサブミックスできるものの、その場合は空間情報が失われるかもしれません。例えばサブミックスが7.1であると、アンビソニックスAudio Busにミックスした時にHeight情報がすべて失われます。同様にサブミックスが7.1.4であれば、元々リスナーの耳より下に配置されているオブジェクトが耳の高さに聞こえてしまいます。以上の理由で、サブミックスにもアンビソニックスを使うことが推奨されます。空間配置の精度がそれほど重要でなければ、より低いオーダーを使っても良いでしょう。

### ルーティング動作

アンビソニックスに入りするオーディオは、各種コンフィギュレーションのルーティングに従ってWwiseで自動的に変換されます。アウトプットは、使われているチャンネルコンフィギュレーションや、ポジショニングタイプが2Dか3Dかによって変わります。これらの要因で決まるルーティング処理の一般的な情報を、下表に示します。

Routing	2D Positioning Behavior	3D Positioning Behavior
標準コンフィギュレーションから、アンビソニックスへ	信号がアンビソニックスにエンコードされる。	リスナーのポジションや向きに対するゲームオブジェクトの相対位置に

Routing	2D Positioning Behavior	3D Positioning Behavior
	<p>モノソースは直接、W（オムニ）チャンネルにエンコードされる。</p> <p>マルチチャンネルソースの場合は、各チャンネルを、そのチャンネルに基づいた入射角を有するモノソースとして扱う。例えばステレオソースの場合は、左チャンネルが左45°にあるモノソースとしてエンコードされ、右チャンネルが右45°にあるモノソースとしてエンコードされる。全てのインプットチャンネルの角度を以下に示す（時計回りを正とする）：</p> <ul style="list-style-type: none"> <li>• L: -45°</li> <li>• R: 45°</li> <li>• C: 0°</li> <li>• SL: -90°</li> <li>• SR: 90°</li> <li>• BL: -135°</li> <li>• BR: 135°</li> <li>• Heightチャンネルは高さ +45° とする。</li> </ul> <p>入射角の決まったソースのエンコード方法は分かりやすい。</p>	<p>基づいた入射角を使い、信号がアンビソニックスにエンコードされる。</p> <p>マルチチャンネルファイル、スプレッド、フォーカスは標準コンフィギュレーションの場合と全く同様に扱われ、様々なインプットチャンネルを表すバーチャルソースは、ゲームオブジェクト、リスナー、フォーカス、スプレッドに基づいてリスナー周りに配置されて適宜エンコードされる。スプレッドやフォーカスの詳細は「<a href="#">オブジェクトプロパティの減衰カーブ</a>」を参照。</p>
アンビソニックスから、アンビソニックスへ	スピーカーの直接割当が行われる。インプットのオーダーの方が高ければ、余分なチャンネルが捨てられる。インプットのオーダーの方が低ければ、アウトプットの余分なチャンネルは無音で残される。	ゲームオブジェクトやリスナーの相対的な向きに従い、アンビソニックベッドを回す。インプットとアウトプットのオーダーが異なる場合は、小さいオーダーを使って回転を計算し、高いオーダーを無音にする。
アンビソニックスから、標準コンフィギュレーションへ	アンビソニックス信号を、ゲームで指定されたスピーカー位置を使ってアウトプットコンフィギュレーションにデコードする。スピーカー位置の設定方法は <a href="#">SetSpeakerAngles()</a> (in AK::SoundEngine) を参照。	ゲームオブジェクトやリスナーの相対的な向きに従い、アンビソニックベッドを回してから、2Dポジショニングの場合のようにアウトプットの標準コンフィギュレーションにデコードする。



### 注意

2Dモノソースをフロントのまっすぐなソースとして扱わずに、W（オムニ）チャンネルにエンコードすると、異常再生が発生します。2Dモノソースをアンビソニックバスにルーティングして、そのバスが標準コンフィギュレーションのバスにルーティングされている場合は、モノソースが全てのアウトプットチャンネルで再生される結果となります。これに対して、通常はWwiseで2DモノソースはCチャンネルやL/Rチャンネルだけに送られます。

### 関連トピック

- 「アンビソニックスコンポーネントのオーダー」
- 「Ambisonicsをプラグインで使う」
- 「アンビソニックスの制約事項」

---

## 第12章 Priorityの管理

概要 .....	307
オブジェクトのプライオリティを理解する .....	309
オブジェクトの再生インスタンスの制限 .....	312
再生Priorityの設定 .....	316
低ボリュームのオブジェクトの管理 .....	318
プライオリティに関する、コツとベストプラクティス .....	320

## 概要

ゲーム中に、多くのオブジェクトなどが同時に再生される状況が起きる可能性は常にあり、中にはプロジェクトチームが設定したオブジェクトの制限数を超えることもあります。再生されるサオブジェクトの数を効率的に管理するには、同時にいくつまで再生を許可するのか、またどれを優先するべきかを、決める必要があります。

Wwiseでは、ゲーム中に同時に再生するオブジェクトを判断するための主なPriority（優先順位）として、以下の3種類があります。

- **Playback Limit**（再生制限） - 各種インスタンスの、同時に再生できる数の制限。（バーチャルボイスはこれに含まれません。）
- **Playback priority**（再生プライオリティ） - あるオブジェクトの別のオブジェクトと比べた時の優先度。
- **Volume threshold**（ボリューム閾値） - この閾値未満ではの各種オブジェクトが再生されない、一定のボリュームレベル。

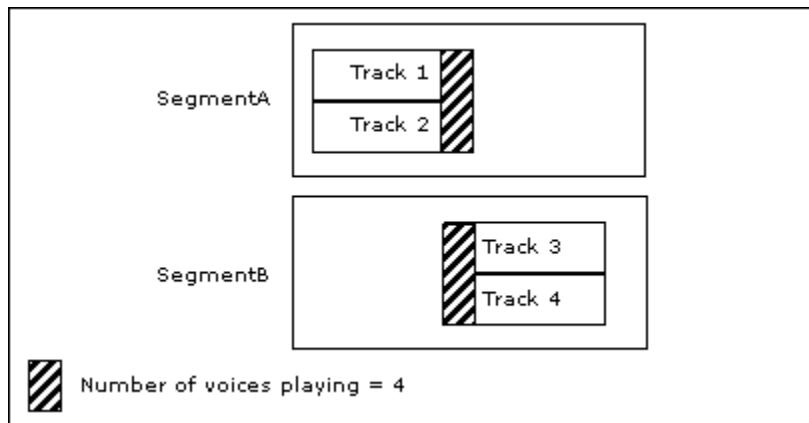
制限を設け、プライオリティを設定して、最低ボリュームレベルを指定することで、ゲームの多数のオブジェクトを効率的かつクリエイティブに管理でき、与えられたメモリ制限を守れます。

### 注記

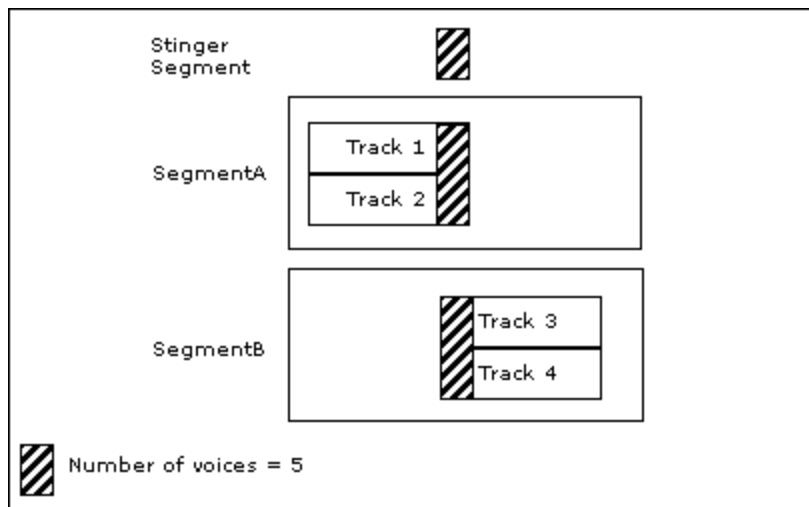
上記の3つのテクニックに加えて、プログラマーがサウンドエンジンのメモリプールの1つ、または複数に対して、メモリ閾値を設定することができます。Memory Threshold（メモリ閾値）設定を有効にすると、使用中のメモリ率が指定した閾値より低いかどうかを、サウンドエンジンが定期的にチェックします。もし閾値を超えた場合は、サウンドエンジンがプライオリティの低いサウンドを順番にドロップし（落し）、プライオリティが高いサウンド用にスペースを開けます。Memory Thresholdについては、[Wwise SDKドキュメンテーション](#)を参照してください。

## 再生制限とプライオリティの活用例

例えば、非常に帯域幅が制限されたゲーム用に、ミュージックを作曲する場合を考えます。同時に再生できるボイスは最高4つ、と指示されたとします。この場合、最上位の親オブジェクトの再生制限（Playback limit）を「4」に設定します。1つのMusic Segmentに入るトラック数を2つ以下としたので、トランジション中に2つのトラックでそれぞれプレエントリとポストエグジットの部分が再生されても、2つのボイスしか再生されません。このようにすれば、同時に2つのセグメントが再生されるとしても、再生制限数の「4」以下となります。

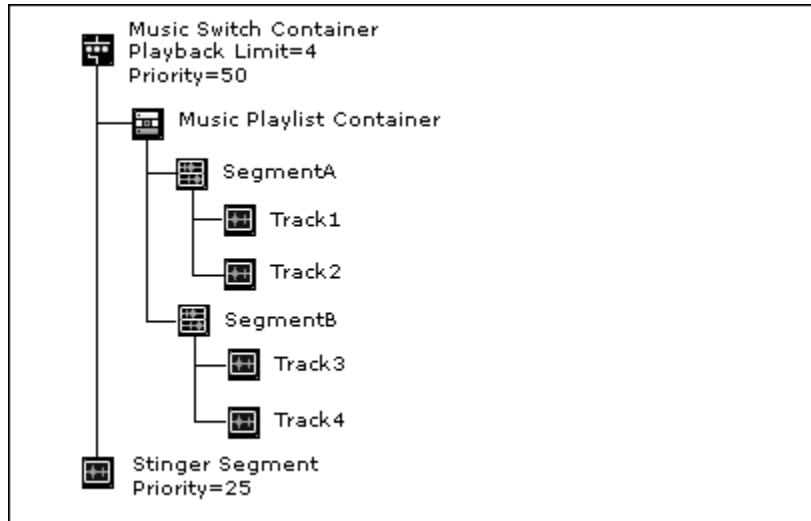


しかし、トランジション中にゲームがステインガーをトリガーすると、最大ストリーム数「4」を超えててしまいます。



ボイス数が再生制限を超えると、Wwiseが各ミュージックオブジェクトの再生プライオリティ (Playback priority) を確認して、どれを停止するのかを判断します。この例では、通常のミュージックが再生され続けるように、ステインガーセグメントのプライオリティを低く設定することが予想されます。もし今回のオブジェクトが5つとも、同じプライオリティであれば、キルする（消す）対象を最新インスタンスとするのか、最古インスタンスとするのかを、選択できます。

参考として、今回の例で考えられるInteractive Music Hierarchy(ミュージック階層)の構造、Playback Limit (再生制限)、Priority (プライオリティ) を、下図に示します。



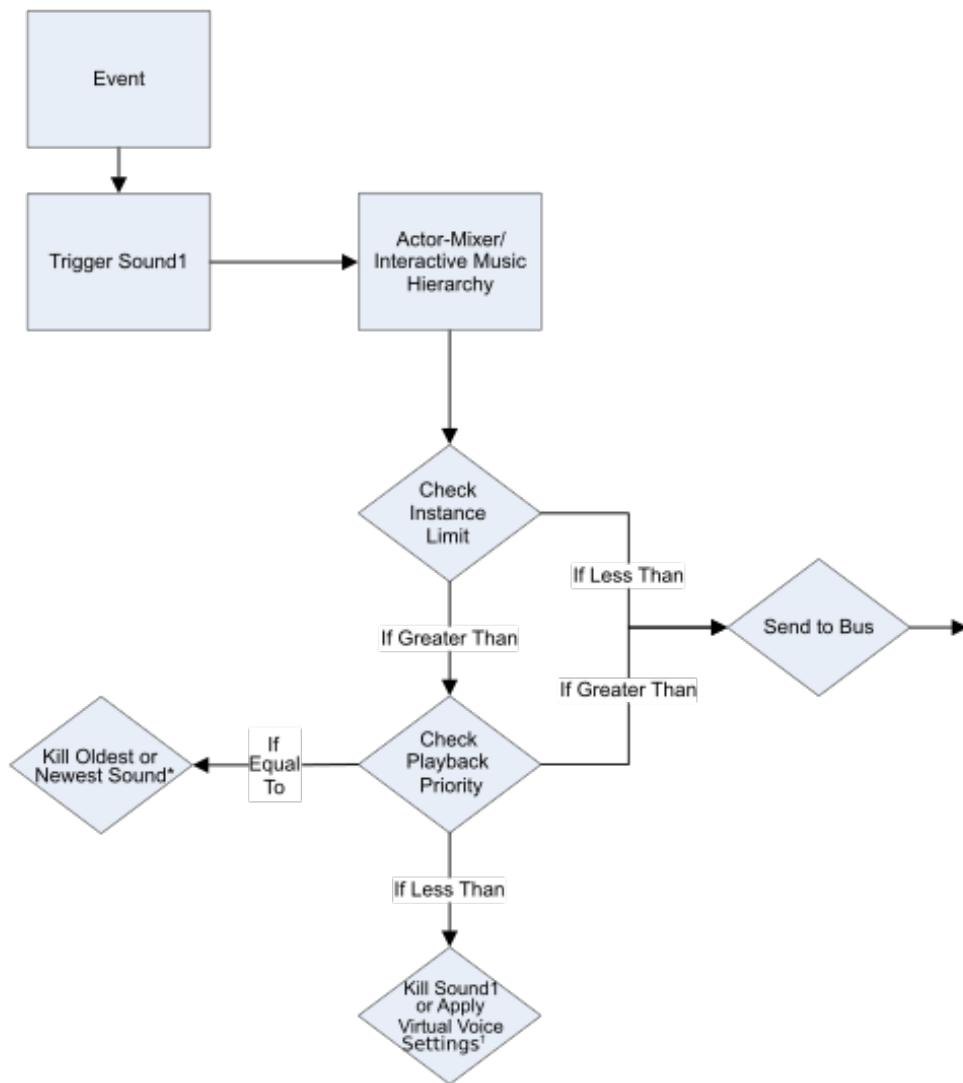
ミュージックオブジェクトやInteractive Music Hierarchyの構築については23章Interactive Music Hierarchyの構築を参照してください。

## オブジェクトのプライオリティを理解する

ゲーム中のある時点で、どのオブジェクトが再生されるのかは、Advanced playback settings（再生の詳細設定）で決定します。再生の詳細設定をWwiseでは2つの異なるレベルで設定し、Actor-Mixer HierarchyやInteractive Music Hierarchyではオブジェクトレベルで、そしてMaster-Mixer Hierarchyではバスレベルで設定します。Wwise内の別々のレベルで設定が行われているため、オブジェクトが再生される前に2つの異なる処理を通過する必要があります。

第1の処理は、アクターミキサーまたはインタラクティブミュージックのレベルで行われます。このレベルでオブジェクトのAdvanced Settingsを設定すると、グローバルまたはゲームオブジェクトごとの上限が設定されます。制限値に到達すると、設定されたプライオリティによって、マスター ミキサー階層のバスのレベルに送られるオブジェクトが決まります。

以下のフローチャートは、Actor-Mixer Hierarchyのどのサウンドオブジェクトやモーションオブジェクトが、またはInteractive Music Hierarchyのどのミュージックオブジェクトが、ゲームオブジェクトごとに再生されるのかをWwiseが判断する様子を示します：実際には、どのオブジェクトがバスに送られるかを最終的に決定する前に、別の親オブジェクト（それぞれ独自のリミットがあってもよい）を通して数回の反復が存在するかもしれません。

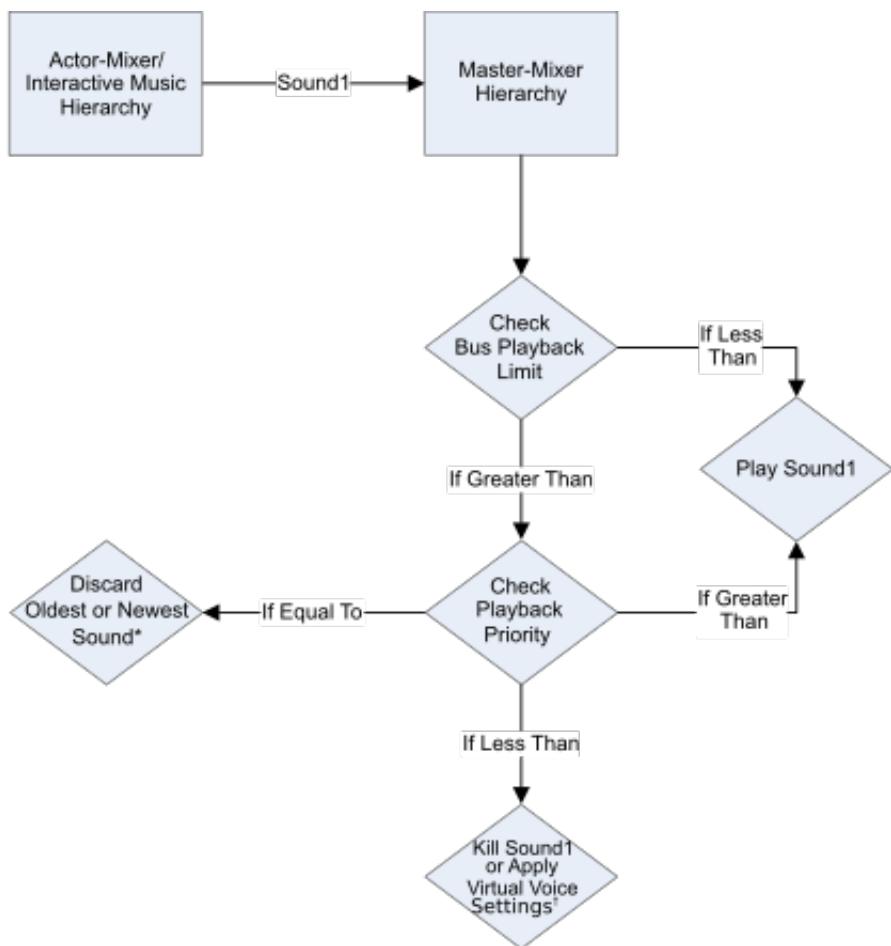


\* You decide whether oldest or newest sound is killed.

† You decide whether to kill the sound or to apply the applicable Virtual Voice Settings.

Actor-Mixer HierarchyまたはInteractive Music Hierarchy で新しいオブジェクトがキルされず、バーチャルボイスに送られなければ、次にMaster-Mixer Hierarchyの処理に進みます。このレベルでは、バス構造内にボイスリミット制限に基づいておなじようなプロセスを経ます。

以下のフローチャートで、バスを通してアウトプットされるオブジェクトをWwise が判断する様子を示します。



\* You decide whether oldest or newest sound is killed.

<sup>†</sup> You decide whether to kill the sound or to apply the applicable Virtual Voice Settings.

この時点では、グローバルプロジェクトのボイス制限数は、Max Voice Instances、Project SettingsのGeneralタブで設定し、キルされるサウンドや、個々のサウンドのバーチャルボイス設定を元にバーチャルボイスとなるサウンドが、判断されます。



### 注記

Property EditorのAdvanced Settingsタブにあるプロパティは、全て絶対プロパティです。絶対プロパティや親プロパティのオーバーライドについては「[絶対プロパティの設定](#)」を参照してください。

### ボリューム閾値とバーチャルボイス

Wwiseで再生するサウンド、ミュージック、モーションの各種オブジェクトを決めるための条件として、Playback LimitやPlayback Priority以外に、Volume Thresholdがあります。ボリュームレベルが、指定されたボリューム閾値に到達した時、またはサウンドの数が再生制限で指定した制限数を超えた時、そのオブジェクトは以下のいずれかとなります。

- 再生し続ける。

- キルされる（消される）。
- バーチャルボイスリストに移動する。

バーチャルボイスリストとは、サウンドエンジンが、オブジェクトの特定パラメータをモニターするものの、全く処理をしない、バーチャル環境です。このオプションを選択すると、各種オブジェクトのボリュームレベルに基づいて、フィジカルボイスからバーチャルボイスへと移動したり、逆に戻されたりします。つまり、そのオブジェクトのボリュームがボリューム閾値以上に戻った場合や、再生されているサウンド数が同時再生サウンド数の制限未満になると、バーチャルとなったオブジェクトが自動的にフィジカルボイスに戻されます。ボリューム閾値の設定については「[プロジェクトのボリューム閾値の設定](#)」を参照してください。また、低ボリュームのオブジェクトの動作の設定については「[低ボリュームのオブジェクトの管理](#)」を参照してください。

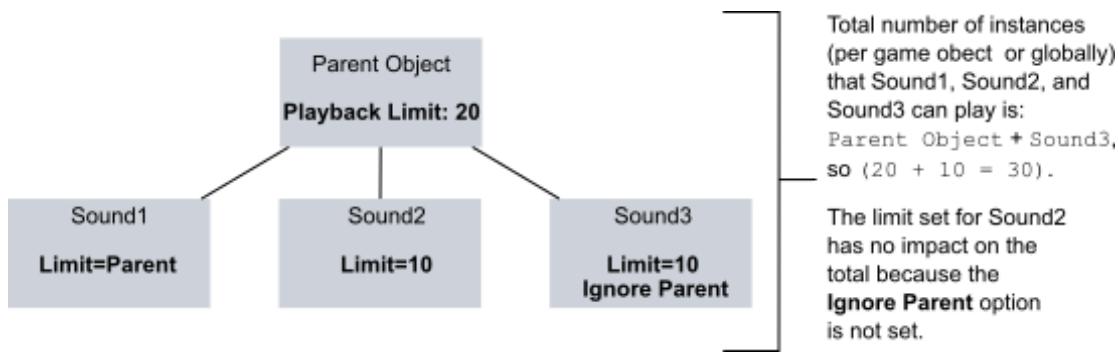
## オブジェクトの再生インスタンスの制限

ゲームのリソース制限やデザイン上の制約に対応するためには、ゲームのどの時点においても、オブジェクトなどの再生を最適化する必要があります。プロジェクトに存在する多様なオブジェクトの同時再生を管理するには、以下の手法があります：

- ゲームオブジェクトごと、または、グローバル（全体）で1つ以上のActor-Mixer Hierarchy または Interactive Music Hierarchy オブジェクト、その子たちに対して、再生できるインスタンス数を制限する；
- バスを通過するインスタンス総数を制限する
- ゲーム全体のインスタンス総数を制限する

制限に到達すると、このオブジェクトのプライオリティ設定に従って、制限に到達した場合の対応として選択した動作に基づきどれを捨てるのかをWwiseが判断します。複数のオブジェクトが同じプライオリティの場合は、捨てるべきインスタンスとして再生中の最新または最古のインスタンスを選択できます。

アクターミキサーまたはインタラクティブミュージックのレベルで再生制限を設定すると、同じ構造の中で再生できるインスタンス数を、グローバルまたはゲームオブジェクトごとにコントロールすることになります。もし子オブジェクトが親レベルで設定した再生制限を無視すると、制限階層の一部ではなくなります。その代わりに、再生するインスタンスの総数はそれらの合計になります。つまり、制限が親で「20」、子で「10」であれば、可能なインスタンスの最大数は「30」です。しかし、子が **Ignore parent** に設定されていなければ、インスタンスの最大数は親と同様に20です。以下のイラストに見られうように、親が無視されるかどうかにかかわらず、制限は子オブジェクトに配置されます。これは特定のオブジェクトとその子孫のための最大インスタンス数の細かな再生制御を可能にします。



一方、Master-Mixerレベルで再生制限を設定すると、バスを同時に通過できるサウンド、ミュージック、モーションの各種インスタンスの合計数を指定することになります。オブジェクトのプライオリティはアクターミキサーまたはインタラクティブミュージックのレベルで既に個別に指定されているので、バスの再生プライオリティ設定はありません。

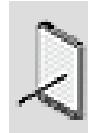
プロジェクトレベル、つまりゲームレベルで再生制限を設定すると、同時にアクティブとなる各種インスタンスの合計数を指定することになります。各サウンドのAdvanced Settingsで個別に設定したVirtual Voiceの動作が、適用されます。

### 再生制限を設定するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Advanced Settingsタブに切り替える。
3. グループボックスPlayback Limitで、**Limit sound instances to** オプションを選択する。

再生制限のオプションが選択可能となる。

4. 該当するテキストボックスに、1つのゲームオブジェクトで同時に再生できる最大インスタンス数を入力する。
5. Playback Limit (再生制限) のスコープを設定するために、以下のオプションを1つ選択する。
  - Globalは、グローバルに制限を適用する。
  - Per game objectは、ゲームオブジェクトごとに制限を適用する。



#### 注記

バスでは、制限が常にグローバルなので、このオプションは選択できません。

6. Playback Limit (再生制限) に到達した場合の処理として、以下のオプションを1つ選択する。
  - Kill voiceは、最低プライオリティのインスタンスの再生を停止する。
  - Use virtual settingsは、事前に定義したそのサウンドのバーチャルボイス動作を実行する。サウンドバーチャル動作とは Continue to play (再生を続ける)、Kill voice (キルする)、Send to virtual voice (バーチャルボイスに送る)、Kill if finite else virtual (無限でなければキル、そうでなければバーチャル) です。

7. 再生制限に到達したものの、最低プライオリティのオブジェクトが複数ある場合の処理として、When limit is reached and priority is equal（制限に到達してプライオリティが等しい時）リストから、以下のオプションを1つ選択する。
- **Discard oldest instance**は、再生中のインスタンスの中で最低プライオリティ、かつ最古のものを停止させる。
  - **Discard newest instance**は、再生中のインスタンスの中で最低プライオリティ、かつ最新のものを停止させる。
8. プロジェクトつまりゲームの制限は、WwiseのProject Settingsでプラットフォームごとに編集できる。またサウンドエンジンが正しく初期化されれば、Wwise SDK APIを使ってゲームから変更することもできる。

ボイス制限のシステムを使う時に、以下を知っておくと良い。

- バーチャルボイスは、有効なボイスとして数えられない。
- バーチャルボイス設定が**Continue to play**であるサウンドが原因で、制限を超える可能性がある。
- バーチャルボイスに送られたサウンドは、通常、フィジカルボイス（バーチャルではない）に反応してPlay from beginning（最初から再生）、Play from elapsed time（経過した時間から再生），or Resume（再開）に設定されます。しかし、Kill if finite else virtual オプションの場合、バーチャルボイスに送られたサウンドは、いつもフィジカルボイスに反応してPlay from elapsed timeに設定されます。

例えば、以下の例を検討します。

- バスで設定された制限は、同時に最大4つのサウンド。
- このバスのWhen limit is reachedは、Use virtual voice settingsに設定されている。
- ボリューム閾値は、-60 dB
- 現在のフレームで、このバスで、以下の8つのボイスが、再生されている。

サウンド名	Priority	Volume	バーチャル動作の設定
Sound_1	100	0 dB	Go Virtual
Sound_2	90	-90 dB	Go Virtual
Sound_3	80	-90 dB	Continue to play
Sound_4	70	0 dB	Go Virtual
Sound_5	60	0 dB	Kill voice
Sound_6	50	0 dB	Kill voice
Sound_7	40	0 dB	Go Virtual
Sound_8	30	0 dB	Continue to play

結果を下表に示します。

サウンド名	結果	理由
Sound_1	再生 (1/4)	最高プライオリティであり、かつボリューム閾値より上。
Sound_2	バーチャルになる	ボリュームが閾値未満。
Sound_3	再生 (2/4)	ボリューム閾値より下でも、再生し続ける。

サウンド名	結果	理由
Sound_4	再生 (3/4)	-
Sound_5	再生 (4/4)	-
Sound_6	キルされる	制限超過であり、より高いプライオリティのサウンドが4つ、既に再生中。
Sound_7	バーチャルになる	制限超過であり、より高いプライオリティのサウンドが4つ、既に再生中。
Sound_8	再生 (5/4)*	* バーチャルやキルにできないサウンドで、より高いプライオリティが再生中の4つのサウンドを置換えることができない特別な状況であり、制限超過しても再生される。

## FAQ

Q: 設定した制限が守られていないようです。なぜか。

A: そのサウンドのバーチャル動作が、互換性の理由からデフォルト設定である**Continue to play**に設定されていないか確認してください。この設定のサウンドは全て、制限を超えても落されず、バーチャル化されません（ただし、制限を超えた時の動作が、明確に“Kill”とされている場合を除く）。制限超過に**Use virtual voice settings**の設定を使い始めた時点で、CPU節約のために落せる、またはバーチャル化できるサウンドは全て、この設定とする必要があります。

選択されたオブジェクト階層内にあるサウンドオブジェクトにのみ制限が適用されることにも注意してください。もし、子オブジェクトが **Ignore Parent** オプションを有効にすると青芸階層の一部ではなくなり、親の制限によってコントロールできなくなります。

Q: 大体の時は制限が守られているようだが、プロファイラを見ると、**Continue to play**に設定されていないサウンドによって、指定した制限数を時々超えることがある。

A: サウンドがバーチャルになる時、そのサウンドが数ミリ秒間（1オーディオフレーム分）アクティブの状態で残り、フェードアウトすることでクリッキングを防止します。例えば、制限が「1」で1つのサウンドが既に再生中であれば、より高いプライオリティの別のサウンドが入ってくると、最初のサウンドがバーチャルになる過程でフェードアウトするため、アクティブボイス数が一時的に「2」となります。

Q: なぜ“Send to virtual voice”を、制限を超えた時のデフォルトとしないのか。なぜ全てのサウンドのデフォルトが“Continue to play”なのか。

A: ソースコーデックによってこのような問題があるため、デフォルトで全てが問題なく動くことを保障するために、デフォルトに採用していません。例えば、Vorbisで**Play from elapsed time**のバーチャル動作を正しく実行するには、シークテーブルが必要です。

## 関連トピック

- ・「[再生Priorityの設定](#)」
- ・「[低ボリュームのオブジェクトの管理](#)」

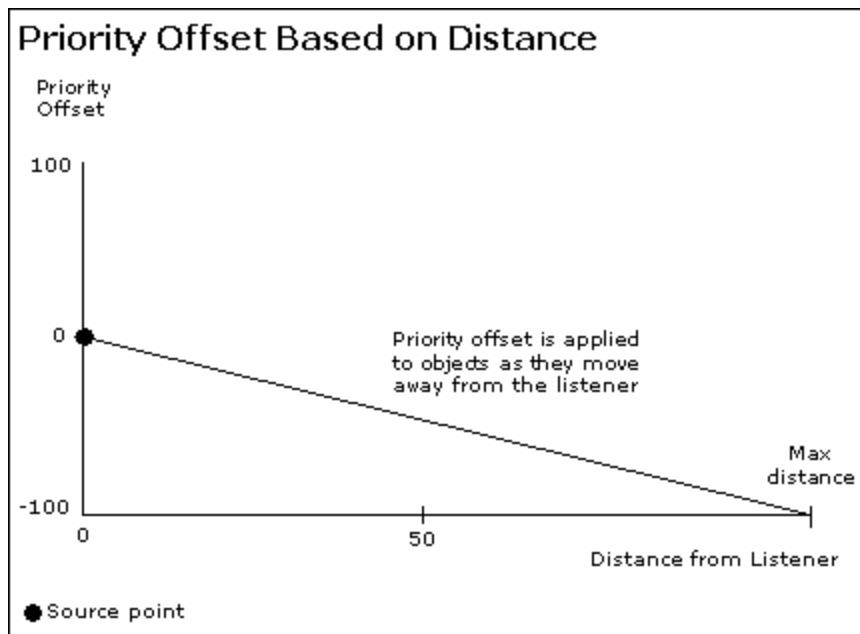
## 再生Priorityの設定

ゲームオブジェクトまたはバスのレベルで同時に再生できるオブジェクトの制限数に到達すると、Wwiseは各オブジェクトのPriority（優先順位）、つまり相対的な重要度を利用して、どれを再生するのかを判断します。

各種オブジェクトのプライオリティは1から100までの標準的な数値単位で設定し、最低プライオリティは「1」、最高プライオリティは「100」とします。複数のオブジェクトが同じプライオリティの場合は、再生中の最新と最古のどちらのインスタンスを停止するのかを選べます。

### 距離に基づくOffset priority設定の利用

リスナーからオブジェクトまでの距離に基づき、再生プライオリティを変更することができます。Wwiseでは、Attenuation Editorで設定したMax distance（最大距離値）を使いプライオリティをオフセットします。適用するオフセット量は、リスナーに対するオブジェクトの相対的な位置で決まります。オフセットが全く適用されないソースポイントと、指定したオフセット値を全て適用する減衰の最大距離の間で、Wwiseが線形補間を行います。



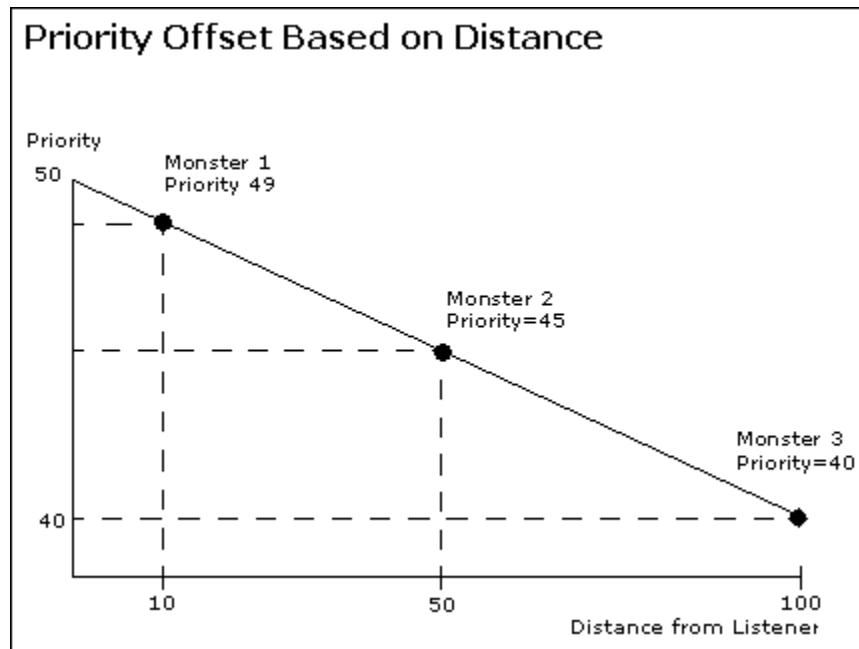
### 距離に基づくプライオリティの活用例

例えば、Monster1、Monster2、Monster3の3つの異なるサウンドオブジェクトがある場合を考えます。どのサウンドオブジェクトもPriorityを50、Offset priorityを-10に設定します。またAttenuation EditorでMax distance（最大距離）を100メートルに設定します。ゲーム中は、3匹のモンスターからリスナーまでの距離がそれぞれ変化します。今回は例として、リスナーからモンスター達までの距離が以下であると仮定します。

- Monster 1: 10 meters

- Monster 2: 50 meters
- Monster 3: 100 meters

3匹のモンスターからリスナーまでの距離はそれぞれ異なり、距離に基づくプライオリティオフセットが適用されているので、各サウンドのプライオリティが異なってきます。リスナーからの距離によってサウンドのプライオリティが影響される様子を、以下のグラフに示します。



Playback priorityはPlayback limit（再生制限）と一緒に使われ、再生制限を超えるとWwiseがこれらのサウンドのそれぞれのプライオリティを確認して、どれを再生するかを判断します。

### オブジェクトの再生Priorityを定義するには:

1. Property Editorに、最上位の親オブジェクトを1つロードする。



2. Advanced Settingsタブに切り替える。
3. テキストボックスPriorityにオブジェクトのプライオリティ、つまり相対的な重要度として、1から100までの数値を入力するが、100は最高プライオリティを表す。
4. リスナーからオブジェクトまでの距離に基づいて再生プライオリティをオフセットするには、Offset priority by x at max distance（最大距離でxだけプライオリティをオフセットする）を選択する。
5. テキストボックスに、最大距離でプライオリティをオフセットさせる数値を指定する。

## 関連トピック

- 「オブジェクトの再生インスタンスの制限」
- 「低ボリュームのオブジェクトの管理」

## 低ボリュームのオブジェクトの管理

多数のオブジェクトが同時に再生される時に、パフォーマンスの最適レベルを維持する必要があります。これを達成する便利な方法として、一定のボリュームレベルより低いオブジェクトにフラグを立てて、貴重な処理能力やメモリの消費を防ぎます。聞こえなくなったこれらのオブジェクトは、再生し続けることもできますが、完全に停止させる、またはバーチャルボイスリスト（Virtual voice list）のキューに入れることも可能です。

バーチャルボイスリストとは、サウンドエンジンが、オブジェクトの特定パラメータをモニターするものの、全く処理をしない、バーチャル環境です。オブジェクトをバーチャルボイスリストに入れるメリットは、そのボリュームレベルによって、オブジェクトがフィジカルボイスとバーチャルボイスの間を行き来できることです。ボリューム閾値未満にボリュームが下がると、バーチャルボイスリストにそのオブジェクトが追加され、処理されなくなります。オブジェクトが減衰の最大距離の半径内に移動するなどしてボリュームレベルが上がると、そのオブジェクトがバーチャルボイスからフィジカルボイスへと移り、オーディオまたはモーションが再びサウンドエンジンによって処理されます。なお、低ボリュームのオブジェクトに、バーチャルボイスを利用する一定のメリットがある一方、フィジカルボイスに戻る時のわずかなディレイの発生や、サンプルアキュレート性の欠如など、一定の限界があり、全てのオブジェクトに適しているわけではありません。

さらに柔軟に細かく操作できるように、オブジェクトがバーチャルボイスからフィジカルボイスに戻る時の再生の動作も、設定できます。



## 低ボリュームのサウンド管理の活用例

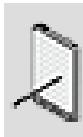
例えば、ファーストパーソンシューティングゲームで、メインキャラクターが複雑な廊下を移動中の場合を考えます。彼は敵の無線機を拾ったので、敵の通信内容を盗み聞きできます。このような状況で同時に再生されるサウンドは多数あり、キャラクターの足音、暗い廊下を照らす数々のトーチの燃える音、無線機から聞こえてくるサウンドなどがあります。それぞれのサウンドの特徴や要件は異なります。このため、ボリュームが変更する時の処理も異なります。

サウンド	処理	Wwiseオプション名
足音	一般的に足音サウンドは短い単発のサウンドなので、ボリューム閾値より低いボリュームの時は、サウンドをキルして貴重なオーディオ処理能力を節約しても良いでしょう。	Kill voice
無線通信	無線機のボリュームは、ゲームのキャラクターがいつでも変えられるので、無線通信の継続性、サンプルアキュ	Continue to play

サウンド	処理	Wwiseオプション名
	レート性、そして即時対応が求められます。サウンドのボリュームがボリューム閾値より低くなった時も、サウンドの再生を継続することができます。これにより、プレイヤーが敵のコミュニケーションを再び聞きたいと思ってボリュームを上げた時、すぐにサウンプルアキュレートなレスポンスを生成できます。	
トーチ	トーチのサウンドは比較的短く、常にループされます。このサウンドは一定の継続性を要するものの、無線通信などの正確性は求められません。サウンドのボリュームがボリューム閾値より低くなった時は、サウンドをバーチャルボイスリストに送ります。サウンドエンジンは、ボリュームレベルをモニターするものの、ボリューム閾値を再び超えるまでオーディオ処理を一切行いません。	Send to virtual voice

## 低ボリュームのオブジェクトを管理するには:

- Property Editorに、オブジェクトを1つロードする。



### 注記

このオブジェクトが最上位のオブジェクトでない場合、Volume Thresholdオプションを設定する前にOverride parentオプションを選択する必要があります。

- Advanced Settingsタブに切り替える。
- Volume Thresholdより低くなった時のオブジェクトの動作を指定するには、Virtual Voice グループのVirtual voice behaviorリストから以下のオプションを1つ選択する:
  - Continue to playは、そのオブジェクトが聞こえなくなる、または感じられなくなるにも関わらず、それを再生し続ける。サンプルアキュレートな状態を維持するには、唯一の選択肢である。
  - Kill voice : オブジェクトの再生を停止する。
  - Send to virtual voiceは、そのオブジェクトを、バーチャルボイスリストに送り、サウンドエンジンが特定パラメータをモニターするものの、全く処理をしない。
- Send to virtual voiceオプションを選択した場合は、オブジェクトがバーチャルボイスからフィジカルボイスに戻る時の動作を指定する必要がある。指定するには、On return to physical voice (フィジカルボイスに戻る時) リストから、以下のオプションを1つ選択する:
  - Play from beginningは、オブジェクトを最初から再生する。オブジェクトのループカウントもリセットされる。
  - Play from elapsed timeは、まるでオブジェクトが停止しなかったかのように、再生を続ける。
  - Resumeは、フィジカルボイスからバーチャルボイスリストにオブジェクトが移動する時に一時停止して、フィジカルボイスに戻る時に、再生を再開する。



### 注記

フィジカルボイスに戻るミュージックオブジェクトは、常にPlay from elapsed time (経過した時間から再生) オプションで再生されます。

## 関連トピック

- ・「オブジェクトの再生インスタンスの制限」
- ・「再生Priorityの設定」

# プライオリティに関する、コツとベストプラクティス

オブジェクトの詳細設定を決める前に、以下のセクションに目を通して、求められる効果を達成するために役立つコツやベストプラクティスを参照してください。

## Playback Priority

ミュージックオブジェクトのプライオリティを高くする - 多くの場合、ゲームのミュージックオブジェクトは、他のオブジェクトよりも、高いプライオリティとするべきです。そうすることで、多数のオブジェクトがトリガーされて再生制限を超えた場合でも、音楽が確実に継続して再生されます。

## Volume Threshold

短いサウンドSFXは、ボリューム閾値より低くなった時はキルする - バーチャルボイスリストに送ってしまうと、以下のいずれかが起きてしまうので、一般的に、ボリューム閾値より低くなった短いサウンドFXは、“Kill voice”オプションを適用すべきです。

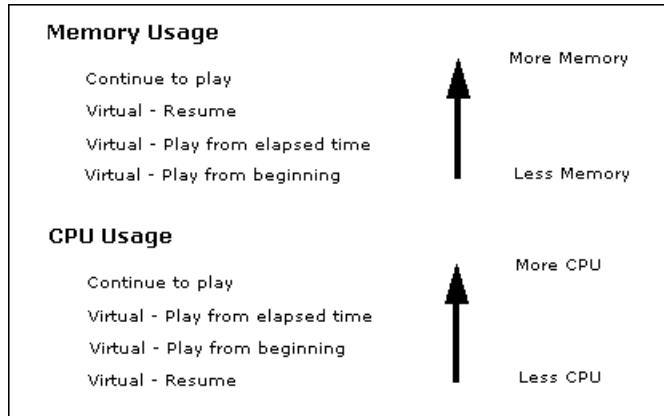
- ・ Play from beginning、またはResumeを選択すると、サウンドの元の目的とは異なる状況で再生されてしまう可能性が高い。
- ・ IPlay from elapsed timeを選択すると、バーチャルボイスとして終わってしまう可能性の高いサウンドに、CPUが無駄に使われてしまう。

## バーチャルボイス

バーチャルボイスのオプションを選択する時に、バーチャルボイス設定によるメモリ消費やCPU負荷に注意してください。

- ・ Play from beginning - メモリやCPUの使用量が低いオプションだが、ストリーミングのサウンドの場合、バーチャルボイスから戻る時にディレイが起きる可能性がある。
- ・ Play from elapsed time - ある程度のCPUとメモリが節約できるオプションだが、ストリーミングのサウンドの場合、バーチャルボイスから戻る時にディレイが起きる可能性がある。
- ・ Resume - より少ないCPUと、より多いメモリを使用するオプションであり、これはサウンドがバーチャルボイスから戻る時のために、メモリバッファを留保しておくからである。なお、サウンドが範囲内に戻らないためバーチャルボイスから戻ることがないと、ゲーム終了までメモリバッファがメモリに残ってしまう。この状況が多くのサウンドで発生すると、時間の経過と共にメモリバッファが蓄積され、ゲームでかなりのメモリ量を使用し続ける可能性がある。

バーチャルボイスの各オプションと“Continue to Play”オプションの、メモリ消費とCPU消費を比較した関係を、下図に示します。



ただし、サウンドのオーディオフォーマット、サンプルレート、ストリーミング設定など諸条件を考慮すると、上記オプションの差異は、無視できる程度であること、非常に大きな差となることも考えられるので、注意が必要です。

常に計算される	バーチャルの時は計算されない
<ul style="list-style-type: none"> <li>ポジショニング計算、ボイスのプライオリティ、そしてボリュームは、ボイスをフィジカルボイスに戻す判断基準として必要なため、引き続き計算される。</li> <li>Play from elapsed timeモードの場合は、バーチャルの時も、ソースプラグインが引き続き計算される（ソースプラグイン内をサウンドエンジンがシークできないため）。</li> </ul>	<ul style="list-style-type: none"> <li>コーデック解凍。</li> <li>オーディオデータの移動、コピー、ミキシング。</li> <li>LPF、サンプルレートコンバージョン、ピッチシフト、Actor-Mixer Hierarchyに挿入されたエフェクト。</li> <li>I/O reads。</li> </ul>

- 注記**  
多くの場合、ボイスがバーチャルになるとオーディオデータのバッファが解放されます  
が、Resume動作の時は解放されません。
- 注記**  
Master-Mixer Hierarchyのバスに対するエフェクトは、このバスを経由して再生される  
フィジカルボイスが1つ以上ある時に限り、処理されます。

### ミュージックオブジェクトに、インスタンス制限を設ける

ミュージックオブジェクトにインスタンス制限を設ける場合、制限数を低く設定しそぎると、いくつかのクリップが再生されないためにミュージックが割れたり切れたりすることがあるので、注意してください。Music Segment間でフェードやSwitchを行う時に、各セグメントで、1つ以上のクリップのプレエントリ部分やポストエグジット部分が再生され、さらにステインガーセグメントが再生されることもあり、簡単にインスタンス数が増えてしまうことがあります。

---

## 第13章 エフェクトの活用

概要 .....	323
エフェクトの使用 .....	323
環境音響を実現する、エフェクトの使用 .....	331
エフェクトに関する、コツとベストプラクティス .....	335

## 概要

ゲームのサウンドやミュージック、モーションなどを強化するには、プロジェクト階層にあるオブジェクトに1つ以上のエフェクト (Effects) を適用する方法があります。Wwiseはオープンアーキテクチャであるため、簡単に自分でエフェクトプラグインを作成して定義できます。多様なエフェクトを活用することで、ゲームのサウンドやミュージック、モーションなどが活き活きとし、よりリアルになります。さらに、Wwiseにいくつかのエフェクトプラグインやファクトリシェアセットが同梱されているので、この作業に利用してください。

エフェクトのレンダリングについては「[エフェクトのレンダリング](#)」を参照してください。また、環境音響を再現するためのエフェクトについては「[環境音響を実現する、エフェクトの使用](#)」を参照してください。

## エフェクトの使用

エフェクト (Effects) は、複数のプロパティをまとめた、インスタンスと呼ばれるもので構成されます。設定したインスタンスをShareSetとして使えば、プロジェクトの複数のオブジェクトに適用できます。ShareSetを利用するメリットは、個別のオブジェクトごとにエフェクトのプロパティを変更する必要がないことです。また、目的に応じて、特定のオブジェクトに合わせたカスタムインスタンスを設定することもできます。

ShareSetのエフェクトのプロパティを変更すると、そのShareSetを使用する全てのオブジェクトが影響を受けます。

プロジェクトのエフェクトを管理するためのタスクを、以下に示します。

- ・「[エフェクトShareSetの作成](#)」
- ・「[エフェクトShareSetの削除](#)」
- ・「[オブジェクトにエフェクトを適用する](#)」
- ・「[バスにエフェクトを適用](#)」

### エフェクトShareSetの作成

一般的に、階層のオブジェクトやバスにエフェクトを適用する時は、ShareSetとして適用します。Wwiseにデフォルトや設定済みのエフェクトShareSetsが同梱されていますが、実際には自分でもShareSetを作成して追加することになるでしょう。各エフェクトShareSetによって、エフェクトの異なるバージョンが成立し、これをプロジェクトの様々なオブジェクトに適用します。

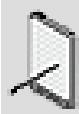
WwiseでエフェクトShareSetを作成できるのは、以下の2ヶ所です。

- Project Explorer
- Property Editor

### Project ExplorerでエフェクトShareSetを作成するには:

1. Project Explorerで、ShareSetsタブに切り替える。

2. Effectsセクションで、シェアセットを作成するWork Unitを、右クリックする。  
ショートカットメニューが表示される。
3. New Childをクリックする。  
設定可能なエフェクトプラグインのリストが表示される。
4. 作成するエフェクトのタイプを選択する。  
新しいインスタンスが、そのWork Unitの子として階層内に表示されます。
5. デフォルトのインスタンス名を、新しいShareSetの内容が分かる名前に置き換えて、Enterを押す。  
ShareSetの新しい名前と、角括弧内にエフェクトのタイプが、エフェクト階層に表示される。

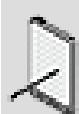


#### 注記

ShareSetの名前は、必ず固有のものとし、文字、数字、アンダースコアのみを使用してください。また、必ず英数字で始めます。ShareSet名を変更するには、ShareSetを右クリックしてRenameを選択して、新しい名前を入力します。ShareSetをゲームに実装した後は、変更すると追加のプログラミングが必要となるため、必要不可欠でなければ避けてください。

### Property EditorでエフェクトShareSetを作成するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Effectsタブに切り替える。



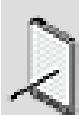
#### 注記

このオブジェクトが最上位オブジェクトでない場合、Effectオプションを選択する前にOverride parentオプションを選択する必要があります。

3. 表Effectsで、セレクタボタン (>>)をクリックする。  
選択可能なエフェクトのリストが表示される。
4. 適用するエフェクトタイプの上まで、マウスのカーソルを移動する。  
該当するエフェクトShareSetのリストが表示される。
5. Newをクリックする。

New Effectダイアログボックスが開く。

6. Effect ShareSetを作成するWork Unitを選択する。
7. ShareSetの名前を入力する。



#### 注記

ShareSetの名前は、必ず固有のものとし、文字、数字、アンダースコアのみを使用してください。また、必ず英数字で始めます。ShareSet名を変更するには、ShareSetを右クリックしてRenameを選択して、新しい名前を入力します。

## 8. OKをクリックする。

新しいShareSetが作成され、選択したオブジェクトに適用される。

### 関連トピック

- 「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- 「エフェクトShareSetの削除」
- 「オブジェクトにエフェクトを適用する」
- 「エフェクトShareSetを、カスタムインスタンスに変換する」

### エフェクトShareSetの削除

不要となったShareSetは削除できます。なお、削除する前に、そのShareSetを使うオブジェクトがないことを確認してください。ShareSetを削除すると、登録している全てのオブジェクトから自動的に削除されます。

#### ShareSetを削除するには:

- Project Explorerで、ShareSetsタブに切り替える。
- Effectsセクションで、削除するShareSetをクリックする。
- Deleteキーを押す。

ShareSetが削除され、それが適用されていたオブジェクトからも削除されます。

### 関連トピック

- 「エフェクトShareSetの作成」
- 「オブジェクトにエフェクトを適用する」

### オブジェクトにエフェクトを適用する

エフェクトを設定すると、プロジェクトのオブジェクトがよりリアルで多様になります。エフェクトShareSetを作成すれば、簡単に編集して、プロジェクトのオブジェクトに適用できます。また、エフェクトShareSetの適用対象のオブジェクトが1つであれば、カスタムエフェクトインスタンスに変換することもできます。

階層にあるオブジェクト1つに対して、最大4種類のエフェクトを適用できます。連続する複数のエフェクトを適用すると、リストに表示された順番でエフェクトが処理されます。エフェクトを適用する順番によってオブジェクトの最終的な結果が変わってくるので、この順番が重要です。また、CPUにかなりの負荷がかかるエフェクトもあるので、その場合は効率的な適用方法を充分に検討してください。エフェクトを適用する際のコツについては「[エフェクトに関する、コツとベストプラクティス](#)」を参照してください。

エフェクトを適用する際に、以下を実行することもできます。

- 「オブジェクトに、エフェクトShareSetを適用する」
- 「エフェクトShareSetを、カスタムインスタンスに変換する」

- ・「エフェクトのバイパス」
- ・「エフェクトのレンダリング」
- ・「エフェクトのプロパティ編集」
- ・「エフェクトの順番変更」

### オブジェクトに、エフェクトShareSetを適用する

ShareSet (ShareSets) を作成した後は、プロジェクト階層のどのオブジェクトにも適用できます。



#### Tip

時間に基づくエフェクト (Wwise RoomVerb、Delayなど) をミュージックオブジェクトに適用してしまうと、このオブジェクトで設定済みの、時間に基づくプロパティや動作と干渉することがあるので、避けた方が良いでしょう。干渉を避けるには、エフェクトをオブジェクトではなくマスターミキサーレベルで適用します。詳しくは「[バスにエフェクトを適用](#)」を参照してください。

### オブジェクトにエフェクトShareSetを適用するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Effectsタブに切り替える。



#### 注記

このオブジェクトが最上位オブジェクトでない場合、Effectオプションを選択する前に Override parentオプションを選択する必要があります。



3. 表Effectsで、セレクタボタン (>>)をクリックする。

オブジェクトに適用できるエフェクトのリストが表示される。

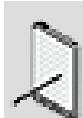
4. 適用するエフェクトタイプの上まで、マウスのカーソルを移動する。

該当するエフェクトShareSetのリストが表示される。

5. 適用するShareSetをクリックする。

ShareSetが、そのオブジェクトに適用されます。

6. オブジェクトに適用するエフェクトを追加するには、上記3~5を繰り返す。



#### 注記

1つのオブジェクトに、最大4種類のエフェクトを適用できます。

### 関連トピック

- ・「[エフェクトのプロパティ編集](#)」
- ・「[エフェクトShareSetを、カスタムインスタンスに変換する](#)」

- ・「エフェクトのバイパス」
- ・「エフェクトのレンダリング」
- ・「エフェクトのプロパティ編集」
- ・「エフェクトの順番変更」

### エフェクトShareSetを、カスタムインスタンスに変換する

Wwiseのエフェクトインスタンス (Effect instances) は、以下の2種類に分類できます。

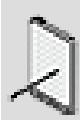
- ・ **Custom effect instances**は、1つのオブジェクトのみに適用される。カスタムインスタンスのプロパティを変更すると、この1つのオブジェクトだけが影響を受けます。
- ・ **ShareSets** は、複数のオブジェクトに適用できる。ShareSetのプロパティを変更すると、そのShareSetを使用する 全てのオブジェクトが、影響を受ける。

デフォルトで、作成した全てのエフェクトインスタンスが、最初はShareSetとなります。1つのオブジェクトだけのために、ShareSetを調整することも可能です。その場合は、Effect ShareSetをカスタムエフェクトインスタンスに変換します。例えば、Funky\_ReverbというShareSetを作成したものの、Speech\_1というサウンドだけに使う、特別バージョンが必要な場合を考えます。まずSpeech\_1にFunky\_Reverbを適用して、それをカスタムインスタンスへと変換してから、調整します。この時の変更は、Speech\_1だけに適用され、Funky\_ReverbShareSetに登録されたオブジェクトには適用されません。

### ShareSetをカスタムインスタンスに変換するには:

1. Property Editorに、エフェクトが適用されているオブジェクトをロードする。
2. Effectsタブに切り替える。
3. 表Effectsで、カスタムインスタンスに変換するエフェクトの、Modeリストより、**Define custom**を選択する。

エフェクトインスタンスの名前に、“Custom”という単語が添付される。これで、このインスタンスを変更しても、影響を受けるのは、このインスタンスを使用する1つのオブジェクトだけである。



#### 注記

ShareSetのデフォルト（カスタム）バージョンをオブジェクトに適用して、カスタムエフェクトインスタンスを作成することもできます。オリジナルのShareSetも引き続き存在し、それに登録されたオブジェクト用に、変更されず残ります。

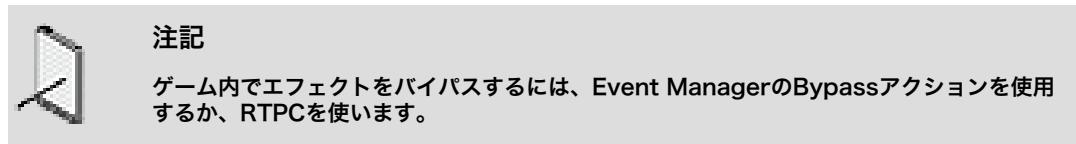
### 関連トピック

- ・「オブジェクトに、エフェクトShareSetを適用する」
- ・「エフェクトのプロパティ編集」

### エフェクトのバイパス

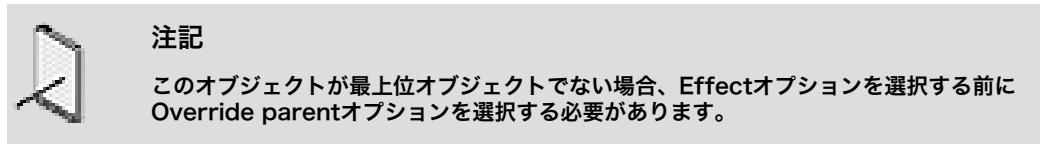
オブジェクトにエフェクトを適用した後でも、エフェクトをバイパス（迂回）することで、元の処理されていないバージョンをいつでも試聴できます。1つのオ

プロジェクトに複数のエフェクトを適用できますが、1つずつバイパスすることができます。また、階層のどのレベルにおいてでも、エフェクトのバイパス (Bypass effect) が可能です。



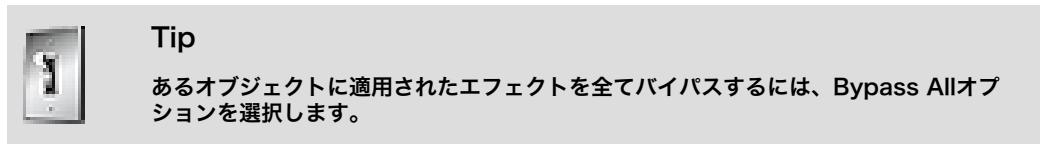
### エフェクトをバイパスするには:

1. Property Editorに、最上位オブジェクトを1つロードする。
2. Effectsタブに切り替える。



3. 表Effectsで、バイパス（迂回）する各エフェクトに対してそれぞれBypassオプションを選択する。

このエフェクトが処理されなくなる。



### 関連トピック

- ・「エフェクトのプロパティ編集」
- ・「エフェクトShareSetを、カスタムインスタンスに変換する」
- ・「エフェクトのレンダリング」
- ・「エフェクトのプロパティ編集」
- ・「エフェクトの順番変更」

### エフェクトのレンダリング

各種オブジェクトに適用されたエフェクト（単数または複数）を、SoundBankにパッケージングする前にレンダリング（Render）することができます。この処理はゲームプレイ中の処理負担を軽くしますが、エフェクトをBypass Effectイベントでバイパスできなくなり、またエフェクトにRTPCカーブを使えなくなります。

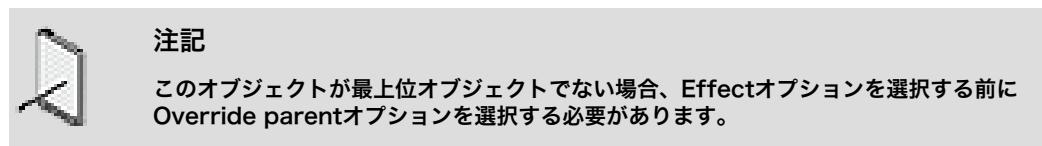
連続する複数のエフェクトのうち、1つのエフェクトをレンダリングすると、それ以前のエフェクトも自動的にレンダリングされます。例えば、サウンドオブジェクトに以下の3種類のエフェクトを適用した場合を考えます。

- ・ Matrix Reverb
- ・ Parametric EQ
- ・ Compressor

ここでParametric EQエフェクトのレンダリングオプションを選択すると、Parametric EQとMatrix Reverbの両方のエフェクトがレンダリングされてから、SoundBankにパッケージングされます。

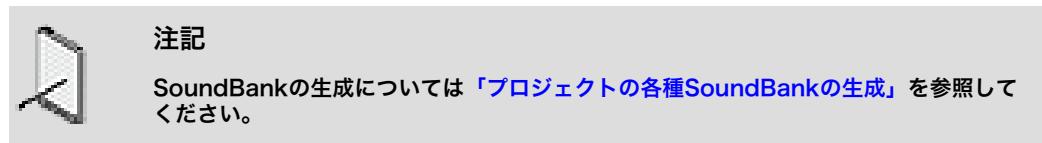
### エフェクトをレンダリングするには:

1. Property Editorに、オブジェクトを1つロードする。
2. Effectsタブに切り替える。



3. 表Effectsで、レンダリングする各エフェクトに対して、それぞれRenderオプションを選択する。

SoundBank生成時に、選択したエフェクトとそれ以前の全てのエフェクトが、オブジェクト上で事前にレンダリングされる。



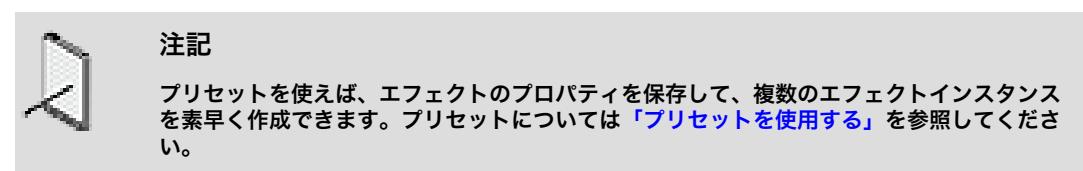
## 関連トピック

- ・「[エフェクトのプロパティ編集](#)」
- ・「[エフェクトShareSetを、カスタムインスタンスに変換する](#)」
- ・「[エフェクトのバイパス](#)」
- ・「[エフェクトのプロパティ編集](#)」
- ・「[エフェクトの順番変更](#)」

### エフェクトのプロパティ編集

全てのエフェクトインスタンス (Effect instances) のプロパティを、目的に合わせて編集できます。特に新しいShareSetやカスタムインスタンスでは、エフェクトを適用する前にプロパティを編集することが多くなります。編集しないと、そのエフェクトのプロパティがデフォルト設定のままで残ってしまいます。また、エフェクトのいくつかのプロパティにRPCを適用すれば、エフェクトインスタンスをさらに強化できます。

なお、ShareSetのプロパティ値を変更すると、そのインスタンスを使う全てのオブジェクトに影響します。一方、カスタムエフェクトインスタンスを編集すると、そのエフェクトが適用される1つのオブジェクトだけが、変更されます。

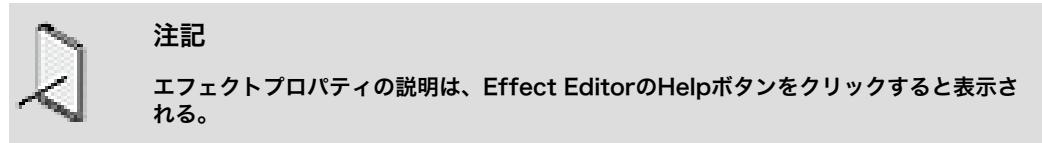


## ShareSetを編集するには:

1. Project ExplorerのShareSetsタブで、編集するShareSetをダブルクリックする。または、Property Editorのグループボックス EffectのEdit...をクリックする。

選択したエフェクトのEffect Editorが開く。

2. プロジェクトで求められるサウンド表現に合わせて、エフェクトのプロパティを設定する。



## カスタムエフェクトインスタンスを編集するには:

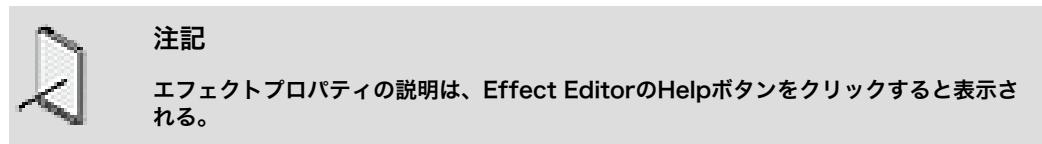
1. Property Editorに、オブジェクトまたはバスを1つロードする。

2. Effectsタブに切り替える。

3. 表Effectsで、編集するカスタムインスタンスの、Edit icon (...)をクリックする。

選択したエフェクトのEffect Editorが開く。

4. プロジェクトで求められるサウンド表現に合わせて、エフェクトのプロパティを設定する。



5. カスタムエフェクトインスタンスのプロパティを元のShareSetのプロパティ値に戻すには、Reset Effect Settingsをクリックする。

## 関連トピック

- ・「エフェクトShareSetを、カスタムインスタンスに変換する」
- ・「オブジェクトに、エフェクトShareSetを適用する」
- ・「エフェクトのバイパス」
- ・「エフェクトのレンダリング」
- ・「エフェクトの順番変更」

## エフェクトの順番変更

複数のエフェクト (Effects) を処理する場合の順番は、オブジェクトの最終的な結果を大きく左右することができます。このため、正しい順番でエフェクトを適用することが重要です。Wwiseは、エフェクト表の順番でエフェクトを処理します。必要に応じて、エフェクトをエフェクトリストの別の場所に移動して、簡単に順番を変更できます。

## エフェクトの順番を変更するには:

1. Property Editorに、オブジェクトまたはバスを1つロードする。
2. Effectsタブに切り替える。
3. 表Effectsで、順番を先または後へ移動するエフェクトを選択する。

その行が、ハイライト表示される。

4. リスト内の移動先まで、エフェクトをドラッグする。

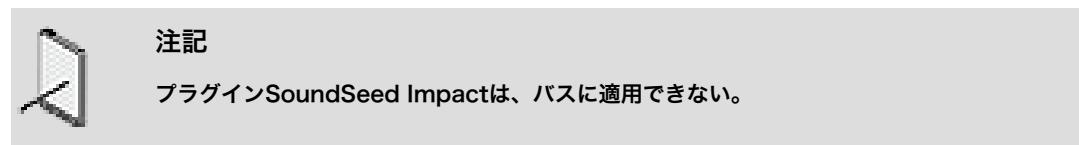
リストでエフェクトが挿入される位置が、赤いラインで表示される。

## 関連トピック

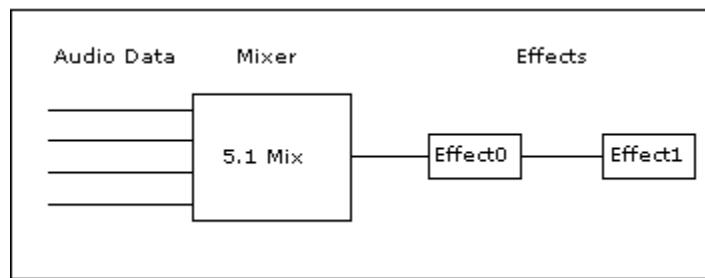
- 「エフェクトShareSetを、カスタムインスタンスに変換する」
- 「オブジェクトに、エフェクトShareSetを適用する」
- 「エフェクトのバイパス」
- 「エフェクトのレンダリング」
- 「エフェクトのプロパティ編集」

## バスにエフェクトを適用

バスにオーディオエフェクト (Audio effects) を適用するのは、各種オブジェクトへの適用と同じく簡単です。バスをProperty Editorにロードして、サウンドオブジェクトにエフェクトを適用する時と同じ手段で、設定するだけです。この操作については「[オブジェクトにエフェクトを適用する](#)」を参照してください。



バスにエフェクトを適用すると、エフェクトが適用される前に、入ってくる全てのオーディオデータをサブミックスします。連続する複数のエフェクトを適用すると、リストに表示された順番でエフェクトが適用されます。



## 環境音響を実現する、エフェクトの使用

Wwiseでは、Audio BusとAuxiliary Busの2種類のバスがあります。AUXバスは、センドシステムと共に、環境音響を実現するために使えます。Auxiliary Busにエ

エフェクト (Effects) を適用して、リバーブを追加するなど、再生中のサウンドが環境の影響を受ける様子を再現できます。また、Auxiliary Busのエフェクトは、ゲームオブジェクトの位置データに従い、ダイナミックに適用することもできます。

プロジェクトのAuxiliary Busでエフェクトを適用する方法を、以下のセクションで説明します。

- ・「Send処理を理解する」
- ・「Game-Defined Auxiliary Sendを、ワークフローに取り込む」

### Send処理を理解する

ゲーム中の環境の音響は、Auxiliary Busで複数のエフェクト (Effects) を合わせて、信号をそのバスに送ることで、シミュレーションされます。作成する Auxiliary Busの数に制限はなく、ゲーム中の各種環境に合わせて作成できます。また、SDKの機能である[SetBusEffect\(\)](#)を使えば、ゲームの中でAuxiliary Busにダイナミックにエフェクトを追加することも可能です。Auxiliary Busにエフェクトを設定した後は、これらのバスに送るオブジェクトを定義する必要があります。

オブジェクトセンドの設定は、User-defined Auxiliary Sendを使って静的に行うのか、Game-Defined Auxiliary Sendを使ってゲーム側でセンドを決定するのかを、選択できます。

### Auxiliary Busを作成するには:

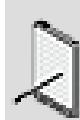
1. Master-Mixer Hierarchyで、Auxiliary Busオブジェクトを作成する。
2. Auxiliary Busを確認する。
3. Auxiliary BusのEffectタブで、環境音響に必要なエフェクトを、追加する。

### User-defined Auxiliary Sendを使って、オブジェクトに環境音響の影響を与えるには:

1. Actor-Mixer Hierarchyのオブジェクトを、確認する。
2. User-defined Auxiliary Sendリストに、Auxiliary Busをドラッグ&ドロップする。
3. Send volumeスライダを使って、Auxiliary Busに送る信号量を定義し、オブジェクトが環境音響にどの程度、参加するのかを、定義する。

### Game-Defined Auxiliary Sendを使って、オブジェクトに環境音響の影響を与えるには

1. Actor-Mixer Hierarchyのオブジェクトを、確認する。
2. Use game-define auxiliary sendsオプションを選択する。
3. ゲームで、関数AK::SoundEngine::SetGameObjectAuxSendValues()を使い、AUXセンドをゲームオブジェクトごとにアサインする。



## 注記

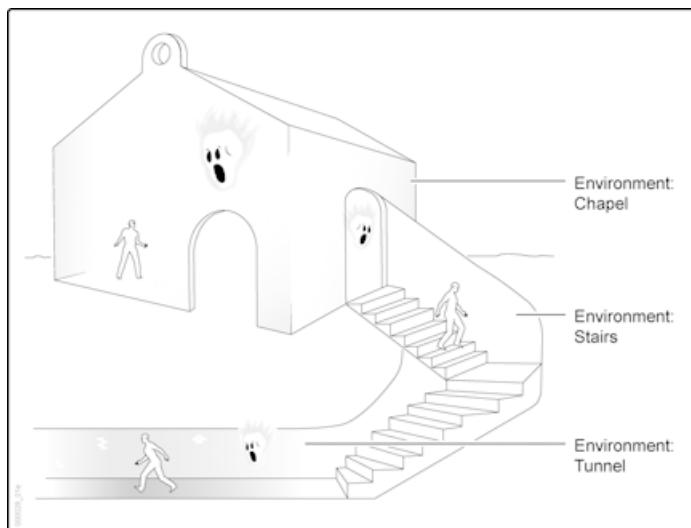
ゲーム内に環境をマッピングする方法については[Wwise SDKドキュメンテーション](#)で、Wwise SDK - Windows > Sound Engine Integration Walkthrough > Integrate Wwise Elements into Your Game > Integrating Environments and Game-defined Auxiliary Sendsを参照してください。また、オーディオデザイナーやオーディオプログラマーは、Integration Demoを見て、AUXセンドの設定例を確認すると良いでしょう。

**事例: AUXセンドを使って環境音響を再現する**

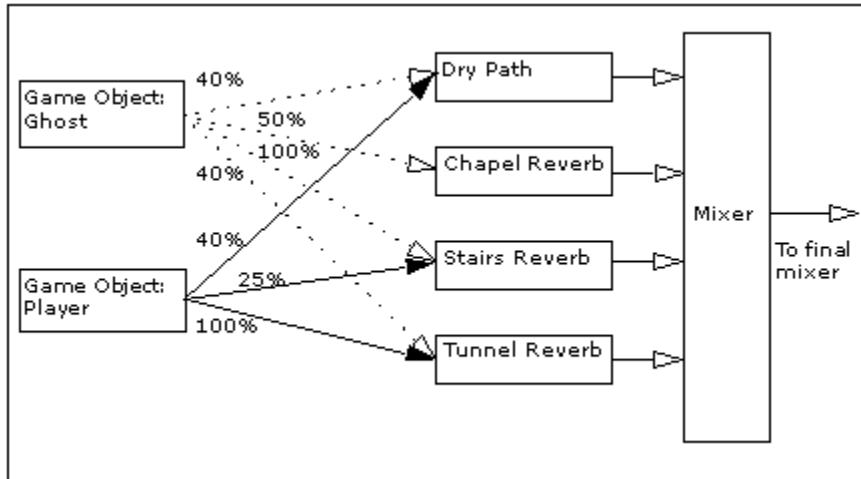
例えば、呪われた墓場を中心に展開するゲームの開発を考えます。ゲームでは沢山の幽霊が出現し、幽霊サウンドがどの環境で聞こえてくるのかによって、幽霊の聞こえ方を変えることとします。ゲームのある場面では、プレイヤーが教会やトンネル、そして両者を接続する階段を探検します。この部分のために、以下のAuxiliary Busを設定します。

- 教会 (Chapel)
- 階段 (Stairs)
- トンネル (Tunnel)

上記の3つの環境に、それぞれ異なるリバーブ (Reverb) を設定します。例えば、トンネルの空間は教会よりもかなり狭く、暗い石壁で囲まれていますから、リバーブは教会と比べて非常に強くなります。Wwiseでは、トンネルの音響に合った高いリバーブレベルと短いディケイ時間のリバーブインスタンスや、教会や階段に合つたりバーブインスタンスを作成できます。作成したAuxiliary Busは、あとから開発者がゲームのジオメトリにマッピングします。その結果、幽霊がトンネルに入ると、教会にいる時よりも幽霊サウンドの響きがより遠くまで聞こえます。



また、WwiseのAUXセンドは、同時に発生する全てのセンドのセンドボリュームをダイナミックに計算することで、ある環境から別の環境へ移動する様子を表現することもできます。今回の事例の、ゲーム中のダイナミックなAuxiliary Busのルーティングとミキシングの様子を、下図に示します。

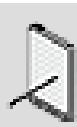


それぞれのサウンドに適用されるセンドの割合は、各ゲームオブジェクトのゲームのジオメトリ内の位置によって決まります。上図の例では、勇敢なプレイヤーが、教会からトンネルに向かって階段を降りているすぐ後ろを、幽霊が追いかけています。トンネルの途中では、プレイヤー（Player）と幽霊（Ghost）が階段（Stairs）環境に100%いますが、同時に教会（Chapel）環境に50%、トンネル（Tunnel）環境に40%います。幽霊のサウンドが、各リバーブインスタンスの適切な割合で処理されます。

### Game-Defined Auxiliary Sendを、ワークフローに取り込む

Wwiseでは、エフェクトでAuxiliary Busを作成して編集して、どのオブジェクトがそれをセンドするのかを設定できます。AUXセンドを、ゲームオブジェクトのポジションに基づいて、プログラムを使ってサウンドに設定できます。Game-Defined Auxiliary Sendを開発チームが実装する方法を、以下に示します。

- ・ サウンドデザイナーが、ゲームの「小部屋」「境界」「洞窟」などの環境に相当するAuxiliary Busオブジェクトを、設定する。
- ・ サウンドデザイナーが、環境音響の影響を受ける全てのサウンドに関して、**Use game-defined auxiliary sends**オプションを選択する。
- ・ ゲーム開発者が、ゲーム内のジオメトリに基づき、環境にAuxiliary Busをマッピングする。
- ・ ゲーム開発者が、リアルタイムで環境の割合（%）を計算した結果を、サウンドエンジンに伝えるメカニズムを作成する。
- ・ サウンドエンジンが、各ゲームオブジェクトがトリガーしたサウンドに適用すべきAuxiliary Busを、計算する。



#### 注記

今回の事例の、ゲーム中のダイナミックなAuxiliary Busのルーティングとミキシングの様子を、下図に示します。また、Wwise SDKの一部としてインストールされている、SoundFrame Test tool (SFTest.exe)を使うこともできます。このツールで、Wwiseをリモート操作してプロジェクトからデータを取得できます。ゲームオブジェクトにイベントを添付して、それを再生し、Edit Sendsボタンを押してGame-Defined Auxiliary Sendを編集します。これによって、Game-definedセンドの様々なレベルをコントロールできます。

## エフェクトに関する、コツとベストプラクティス

Wwiseでエフェクト（Effects）を使う時に、設定できるオプション機能が多数あります。エフェクトを実装する際に効率的な戦略を立てると、サウンドが向上し、リソースも節約できます。プロジェクトでエフェクトを活用する際に、以下の戦略を検討してみてください。

### CPUへの負荷

エフェクトを使うと、必ずCPUに負荷がかかりますが、パワー消費を意識することで、効率的な消費につながります。一般的にMaster-Mixerのレベルでエフェクトを適用すれば、CPU負荷が軽くなります。例えば、ゲームのMaster Audio Busに、Wwise RoomVerbエフェクトを適用すると、ランタイムに処理されるインスタンスが1つだけです。一方、同じエフェクトをオブジェクトレベルで適用すると、同時に何百という数のインスタンス処理が発生するかもしれません。

また、エフェクトをレンダリングすることでも、リアルタイムでの処理が不要となるため、CPUを節約できます。しかし当然、レンダリングされたエフェクトのプロパティは、レンダリング後に変更することができないので、RTPCを適用することはできません。

エフェクトを個別にみると、Delayや、Wwise Parametric EQなどのエフェクトは、CPU負荷が非常に小さい傾向があります。一方、Wwise Compressor、Peak Limiter、Expanderなどのエフェクトは、負荷がやや増えます。リバーブに関しては、Wwise Room Verbを選択すると、リソースを多く使うものの高品質であり、これに対してWwise Matrix Reverbは、品質やパフォーマンスの要件に合わせて、調整できます。

全体的に、最良の戦略は、ゲームプロファイリングツールを使ってプロジェクトをテストすることと言えます。これらを使えば、リアルタイムでエフェクトのCPU負荷を観察できるので、エフェクトをどう使うべきかの判断ができます。プロファイリングについては「[Wwiseにおける、プロファイリングの種類を理解する](#)」を参照してください。

### エフェクトと、ミュージック

時間に基づくエフェクト（例えばWwise RoomVerbやDelayなど）をミュージックオブジェクトに適用してしまうと、このオブジェクトで設定済みの、時間に基づくプロパティや動作と干渉があるので、避けた方が良いでしょう。干渉を避けるには、時間に基づくエフェクトをマスター・ミキサー・レベルで、つまりAudio Busに対して、適用します。

---

## 第14章 モーションの活用

概要 .....	337
ゲームのモーションを作成 .....	339
モーション用に、アウトプット構造の構築 .....	345
モーションに関する、コツとベストプラクティス .....	346

## 概要

プレイヤーをゲームの中へとさらに引き込む別の有力な手段として、モーション（バイブルーション）を使ったフィードバックがあります。

モーション（Motion）を作成してゲームに実装するための、完全なパイプラインを、Wwiseが提供します。既存のオーディオ構築パイプラインと似た、モーション用の総合的なパイプラインソリューションを実装したWwiseを使うことで、以下を実現できます。

- ・洗練されたリアルなモーションエフェクトの作成方法を、短時間で習得できる。
- ・ゲームやサウンドエンジンのパフォーマンスに大きな負荷をかけずに、ゲームにモーションを簡単に実装できる。
- ・オーディオと同じ機能で、モーションを構築し実装できる。
- ・多様なプラットフォームの同類のデバイス向けに、作業量を増やさず、モーションエフェクトを作成できる。
- ・ゲームの要件に合わせて、モーション部分を簡単に追加または削除できる。



### 注記

Wwise Motionは、Macや、iOSプラットフォームでは、利用できません。

オーディオからモーション（Motion）への切り替えをできるだけシンプルにするために、Wwiseではどちらでも、同じワークフロー、方針、そして機能を採用しています。オーディオと同様に、モーションも、階層内で構築し、バスヘアウトプットし、イベントを使ってゲーム中にトリガーします。また、3D環境に配置し、プロジェクトの様々なゲームシンクによって操作し、SoundBankにパッケージングすることも可能です。当然、一番の違いはモーションの再生をスピーカーではなく、ゲームコントローラで行うことです。

## モーションの使い方を理解する

オーディオからモーション（Motion）への切り替えをできるだけシンプルにするために、Wwiseではどちらでも、同じワークフローや、多くの共通する方針や機能を採用しています。オーディオと同様に、モーションも、階層内で構築し、バスヘアウトプットし、イベントを使ってゲーム中にトリガーします。また、3D環境に配置し、プロジェクトの様々なゲームシンクによって操作し、SoundBankにパッケージングすることも可能です。

## 関連トピック

- ・「モーションデバイスの種類」
- ・「モーション生成の方式」

## モーションデバイスの種類

現在、Wwiseが対応しているモーションデバイス（Motion devices）を、以下に示します。

- ・ゲームコントローラ(全プラットフォーム)
- ・PS4 Move®

ほかの種類のデバイスも、サードパーティプラグインを追加すれば可能です。

### モーション生成の方式

最も基本的なレベルで、Wwiseではモーションデータをソースから生成します。このソースは、既にあるメディアファイルか、新規メディアファイルか、シグナルジェネレータプラグインの、いずれかとなります。

ゲームが対応するモーションデバイスを決定した後は、モーションを生成する方式を選択する必要があります。Wwiseでモーションソースを作成する方法は、以下の2つがあります。

- ・既にあるオーディオ信号を使う方法
- ・モーション専用のソースを使う。

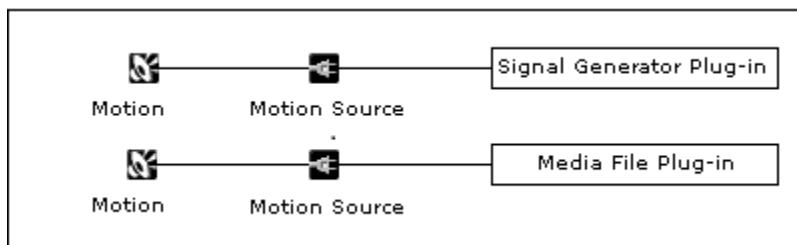
#### 既にあるオーディオ信号から、モーション信号を生成する

モーションを簡単に生成するには、既にあるサウンドを使って、Auxセンドでモーションバスの下にあるAuxバスに送ります。モーションバスは、通常のオーディオバスに、オーディオデバイスとしてモーションデバイスを設定しただけのものです。オーディオ信号が、モーションデバイスプラグインによって、ハードウェアに適した低周波信号に変換されます。元の音に影響はなく、これはほかのセンドのときと同じです。

モーションソースが既存オーディオソースから生成されるため、ゲーム中のオーディオ再生にモーションが関連付いています。つまり、ゲーム中にモーションソースを別のイベントでトリガーする必要はありません。また、モーションソースがオーディオオブジェクトと同じプロパティ、動作、ゲームシンクなどによって左右されることになります。

#### 専用ソースから、モーションを生成

もう1つの方法として、モーションだけのために専用のWwiseオブジェクトを作成して、モーションを生成することができます。モーション用のオブジェクトはほかのサウンドオブジェクトと同じですが、オーディオデータの変換に頼らず、モーション生成専用のソースプラグインを利用します。



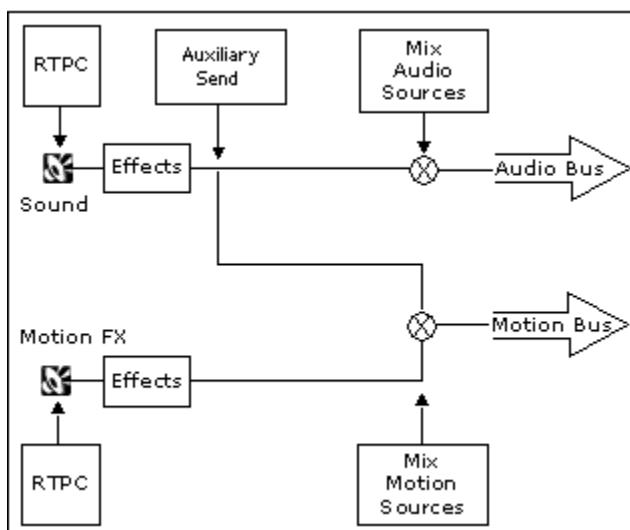
これ以外は、通常の音と同じように動作するオブジェクトです。コンテナやアクターミキサーを使って高度なモーション構造を構築して、モーションエフェクトの

プロパティや動作を設定できます。これらのモーションエフェクトは必ずしもゲームでオーディオに結び付いていないため、ゲーム中にモーションエフェクト自身のイベントによって、いつでもトリガーできます。

当然、状況によって適した方式が異なります。各方式のメリットについては「[ゲームのモーションを作成](#)」を参照してください。

### モーション処理のパイプライン

モーションを処理するパイプラインで、両方の種類のソースから生成されるモーション信号をミックスしてから、ミキシングされたものをモーションバスへ送ります。2種類のモーションソースをWwiseで処理する様子と、モーションパイプラインとオーディオ処理のパイプラインの関係を、下図に示します。



## ゲームのモーションを作成

Wwiseでモーション (Motion) を生成するには、以下の2つの方式があります:

- ・「既存のサウンドから、モーションを生成」
- ・「モーションを生成する専用オブジェクトを作成」

その時の要件や制限によって、どちらの方式を採用するのかを決めます。2つの方式の、それぞれの長所と短所をよく理解した上で、選択してください。

### 既存のサウンドから、モーションを生成

多くのモーションエフェクト (Motion effects) が、オーディオイベントである銃の発砲、道路の振動、爆弾の爆発などにリンクされているので、プロジェクトで既に作成したサウンドやミュージックオブジェクトから、直接モーションデータを生成することができます。この場合の処理は単純で、Wwiseでオーディオ信号から低周波コンポーネントを抽出し、信号のこの部分をリサンプルして、納得のゆくモーションエフェクトにするだけです。この方式は、デザイナーと開発者の作業量が非常に少なく、オーディオと同期したモーションに格好の方式ですが、以下の欠点があります。

- ・作成されるモーションの種類に関する設定範囲が限定される。
- ・オーディオ信号でモーションデータを作成するには、ランタイムにフィルタをかけてリサンプルする必要があり、CPUへの負荷が高い。
- ・モーションがオーディオに関係しない場合に、対応できない。

不利な点があるものの、この方式が向いている状況もあります。例えば、ゲームのモーションのプロトタイプを作成する時に、最適の方式です。プロトタイプが完成して承認されてから、より細かくコントロールできる専用のモーションオブジェクトを作成するのかどうか、決めて良いでしょう。

既存サウンドからモーションを作成するには、以下のタスクを行います:

- ・「[モーションのアウトプットルーティングの指定](#)」
- ・「[オーディオからモーション信号を形成](#)」

なお、既存サウンドからモーションを生成する前に、以下の点に注意してください。

- ・オリジナルのオーディオソースからDCオフセットを取り除くと、モーションアウトプットが変更される。
- ・モーションのアウトプットは、コントローラの物理的なモーターの動力、最高スピード、重さなどが違うので、プラットフォームによって感じ方が変わる。

### モーションのアウトプットルーティングの指定

既存オーディオソースやモーションソースを使ってモーションデータを生成する予定であれば、まず最初に、それをルーティングするモーションバスを指定する必要があります。モーションデータはオーディオデータと異なるので、モーションデータをルーティングする複数のバスからなる、別の階層が必要です。この別の階層をマスター・ミキサー階層に作成するには、新しいマスターバスを作成して、Wwise Motionデバイスをそれにアサインします。モーションバス階層の作成については「[モーション用に、アウトプット構造の構築](#)」を参照してください。

### モーション専用ソースのアウトプットルーティングを指定するには:

1. Property Editorに、最上位オブジェクトを1つロードする。
2. グループボックスOutput Busで、Browse ボタン(...)をクリックする。  
Project Explorer - Browserが開く。
3. モーション対応マスターバスの下のバスを選択する。
4. OKをクリックする。

現在のオブジェクトと、その下にある全ての子オブジェクトから生成されたモーションデータが、選択したバスにルーティングされる。

### モーションのアウトプットルーティングをオーディオソースで指定するには:

1. Property Editorに、最上位オブジェクトを1つロードする。

2. グループボックスUser-defined Auxiliary Sendの、Browseボタン(...)をクリックする。

Project Explorer - Browserが開く。

3. モーション対応マスター・バスの下のAuxバスを選択する。

4. OKをクリックする。

現在のオブジェクトと、その下にある全ての子オブジェクトから生成されたモーションデータが、選択したバスにルーティングされる。

## 関連トピック

- 「[オーディオからモーション信号を形成](#)」

### オーディオからモーション信号を形成

モーションデバイスは低周波しか再現できないので、オーディオソースの高周波を下げたりフィルターをかけたりする必要があるかもしれません。また、出てくる信号を增幅しなければ、適切な強度を感じ取れないかもしれません。オーディオ信号をモーションバスに送信するのにSend to an Auxiliary Busを使うので、センド機能で通常提供されるツールを利用できます。

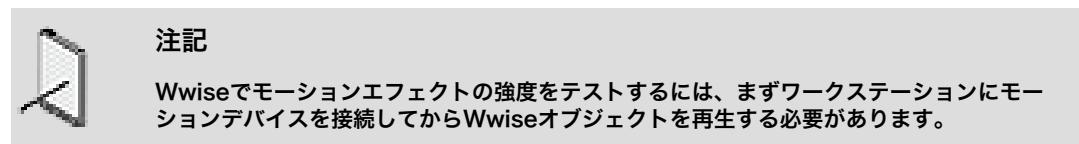
### 高周波をフィルターで除くには:

1. Property Editorに、モーション関係のAuxバスをロードする。
2. グループボックスOutput Busで、Motion Low Passの値を変更して、モーション信号を生成する時に使うカットオフ周波数を設定する。

### 生成したモーション信号をブーストするには、また低減するには:

1. Property Editorに、オーディオオブジェクトを1つロードする。
2. グループボックスUser-defined Auxiliary Send で、Volume の値を変更して、モーション信号を増幅または減衰させる。

さらに、EQなどのエフェクトをAuxバスに適用して、周波数帯を一部ブーストさせたり減少させたりできます。



## 関連トピック

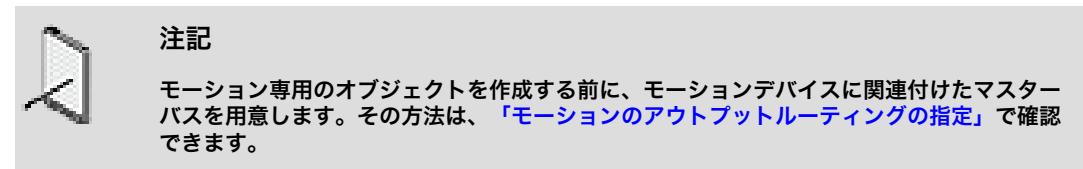
- 「[モーションのアウトプットルーティングの指定](#)」

### モーションを生成する専用オブジェクトを作成

モーションのインテンシティプロファイルをより細かくコントロールするには、プロジェクトでモーションデータだけを生成するオブジェクトを別に設けるのがベ

ストです。それらを、コンテナやアクターミキサーにまとめることができます。サウンドオブジェクトと同様に、メディアファイルにリンクしているソース、またはシグナルジェネレータプラグインを使用するソースを含みます。通常のサウンドオブジェクトとの唯一の違いは、これらのオブジェクトがモーション対応バスに直接ルーティングされることです。また、モーションデータを直接生成する、特別なプラグインソースを使う方が、よい結果が出ることが多いです。現在、Wwiseが提供するモーション専用のソースは、Wwise Motion Generatorだけです。ただし、ほかのプラグインを使うことも可能です。

どのソースを選択するかは、モーションの対象デバイスと、求めているモーションの種類によります。それぞれのソースに、モーション効果を調整するために設定できる複数のプロパティがあります。

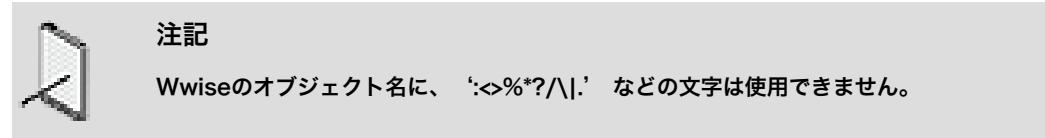


### モーション専用のオブジェクトを作成するには:

1. Project ExplorerのAudioタブで、サウンドオブジェクトを作成するワークユニットを、右クリックする。
2. ショートカットメニューで、New Child > Soundを選択する。

アクターミキサー階層に、新規Soundオブジェクトが追加されます。

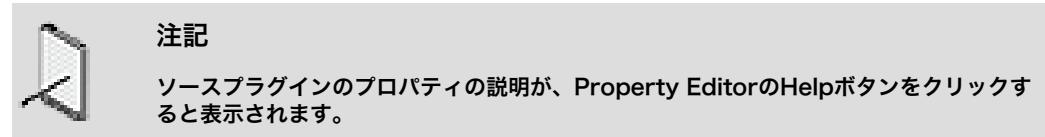
3. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



4. Contents Editorで、Add Source > Wwise Motion Generatorをクリックする。

Wwise Motion Generatorがソースとして追加されます。

5. ソースプラグインをダブルクリックして、全てのプロパティのリストをProperty Editorに表示する。
6. 必要に応じて、プロパティを編集する。



### 関連トピック

- 「[ゲームコントローラ用の、モーションの生成](#)」
- 「[既存のサウンドから、モーションを生成](#)」

## ゲームコントローラ用の、モーションの生成

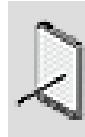
ゲームコントローラは、モーターのスピードを基に、限定された範囲のモーションエフェクトを再現することしかないので、この種類のデバイス用のモーションを生成するために、Wwise Motion Generatorという特別なソースプラグインが設計されました。この特殊なプラグインで、各プラットフォームのコントローラ用に、1つ以上のモーションカーブを作成しカスタマイズできます。モーションカーブはそれぞれ、一定の時間をかけて変化するモーションエフェクトの強さを定義し、オプションでADSRエンベロープコントロールを使いさらに細かくカーブの形状を設定することもできます。

デフォルトで、モーションカーブの強度は、設定時間を通して、モーションをフルに生成する「1」（つまり100%）に設定されています。既存のコントロールポイントを移動し、新しいポイントを追加することで、求められるエフェクトをつくり出してください。

モーションのカーブをさらに細かく複雑に設定するには、カーブセグメントの形状を調整します。カーブセグメントとは、カーブ上の任意の2点のコントロールポイントの間の部分を指します。カーブ形状として、直線、定数、対数関数、指数関数、S型カーブなど、様々な選択肢があります。カーブ形状の指定などグラフ上で の作業に仕方については[43章グラフビューについて](#)を参照してください。

以下の対象に合わせて、それぞれのカーブを作成できます。

- **Default Small** (デフォルト スモール) - モーションジェネレータデバイス（単数または複数）の小型モータによる使用が可能なデフォルトカーブ。
- **Default Large** (デフォルト ラージ) - モーションジェネレータデバイス（単数または複数）の大型モータによる使用が可能なデフォルトカーブ。
- **Default Left Trigger** - 1つまたは複数のモーションジェネレータデバイスの左トリガーモータによる使用が可能なデフォルトカーブ。
- **Default Right Trigger** - 1つまたは複数のモーションジェネレータデバイスの右トリガーモータによる使用が可能なデフォルトカーブ。
- **Xbox On™e Small** - Xbox Oneコントローラの小型モータに割り当てられているカーブ。
- **Xbox One Large** - Xbox Oneコントローラの大型モータに割り当てられているカーブ。
- **Xbox One™ Left Trigger** - Xbox Oneコントローラの左Triggerのモータに割り当てられているカーブ。
- **Xbox One™ Right Trigger** - Xbox Oneコントローラの右トリガーのモータに割り当てられているカーブ。



### 注記

トリガーモータ（デフォルトとXbox One）は、Wwiseオーサリングツールから起動することはできません。PC UWPアプリケーションでのみ使用できます。

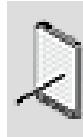
- **PlayStation®4 Small** - PlayStation 4 コントローラの小型モータに割り当てられているカーブ。

- PlayStation®4 Large - PlayStation 4 コントローラの大型モータに割り当てられているカーブ。
- Switch™ Small (Switch スモール) - Switch コントローラの小型モータに割り当てられているカーブ。
- Switch™ Large (Switch™ ラージ) - Switch コントローラの大型モータに割り当てられているカーブ。

ただし、時間を節約するために、カーブをいくつか再利用することも考えられます。また、1つのモーターをソロすることで、そのモーションカーブの微調整がしやすくなります。1つのモーターをソロすると、再生中に他の全てのモーターがミュートされます。

### プラグインWwise Motion Generatorを使い、モーションエフェクトを作成するには:

1. Property Editorに、モーションオブジェクトを1つロードする。
  2. Contents Editorで、Controller motion device タイトルバーで、Add Sourceボタンをクリックする。
  3. 表示されたリストから、Wwise Motion Generatorをクリックする。
- Controller タイトルバーに、モーションソースが追加される。
4. ソースをダブルクリックして、Property Editorにロードする。
  5. Curveリストから、Default Small カーブを選択する。
- カーブが、グラフビューに表示される。
6. Default Small のモーションカーブを、以下の操作で編集する。
    - カーブ上にポイントを追加する。
    - ポイントを別の場所にドラッグするか、XY座標のボックスに具体的な数値を入力する。
    - 各カーブセグメントの形状を決める。



#### 注記

特定のモーターのモーションを微調整するには、リストにある各モーターをソロします。1つのモーターをソロすると、再生中に他の全てのモーターがミュートされます。

7. その他のカーブについて、以下のいずれかを行う。

Usageリストで、Customを選択して、新規モーションカーブを作成する。

Usageリストで、既存のカーブ名を選択して、そのカーブを再利用する。

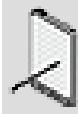
8. テキストボックスPeriodで、1回のモーションカーブサイクルにかかる時間を、指定する。

この設定値で、X軸の長さが決まる。

9. RTPCを使って、モーションエフェクトの長さをリアルタイムで変更したい場合は、テキストボックスPeriod Multiplierに数値を入力する必要がある。Period

344

multiplier値を変更すると、新しい時間の長さに合うように、カーブがストレッチまたは縮小される。



#### 注記

RTPCを使用していない場合は、Period Multiplierの数値をデフォルトの「1」のままにします。

10モーションエフェクトの時間の長さを指定するには、以下のいずれかを選択する：

**One period** - モーションエフェクトの長さとして、Period×Period multiplierと定義する期間 (period) 1つを使う。

**Fixed duration** - モーションエフェクトに固定の長さを指定する。ここで指定した数値によって、モーションエフェクトのカーブを再生する回数が変わる。もしFixed duration値が、Period×Period multiplierの倍数でなければ、最後のカーブサイクルが途中で終わる。

**Envelope** - ADSRエンベロープコントロールを使い、時間の経過に伴うカーブの形状を決める。

#### 関連トピック

- ・ [「既存のサウンドから、モーションを生成」](#)

## モーション用に、アウトプット構造の構築

モーションデータはオーディオデータと似ていますが、一般的にオーディオと違うミキシングが必要です。モーションデバイスに対してアウトプットするには、モーションデータをルーティングできる別のバス階層が必要です。この別の階層をマスターミキサー階層に作成するには、新しいマスター・バスを作成して、Wwise Motionデバイスをそれにアサインします。

#### モーション対応マスター・バスを作成するには:

1. Project Explorerを開く。
2. Audioタブのマスター・ミキサー階層で、Default Work Unitを右クリックする。
3. メニューで、New Childを、そしてAudio Busを選択する。

バスが作成されて、そのプロパティページが開きます。

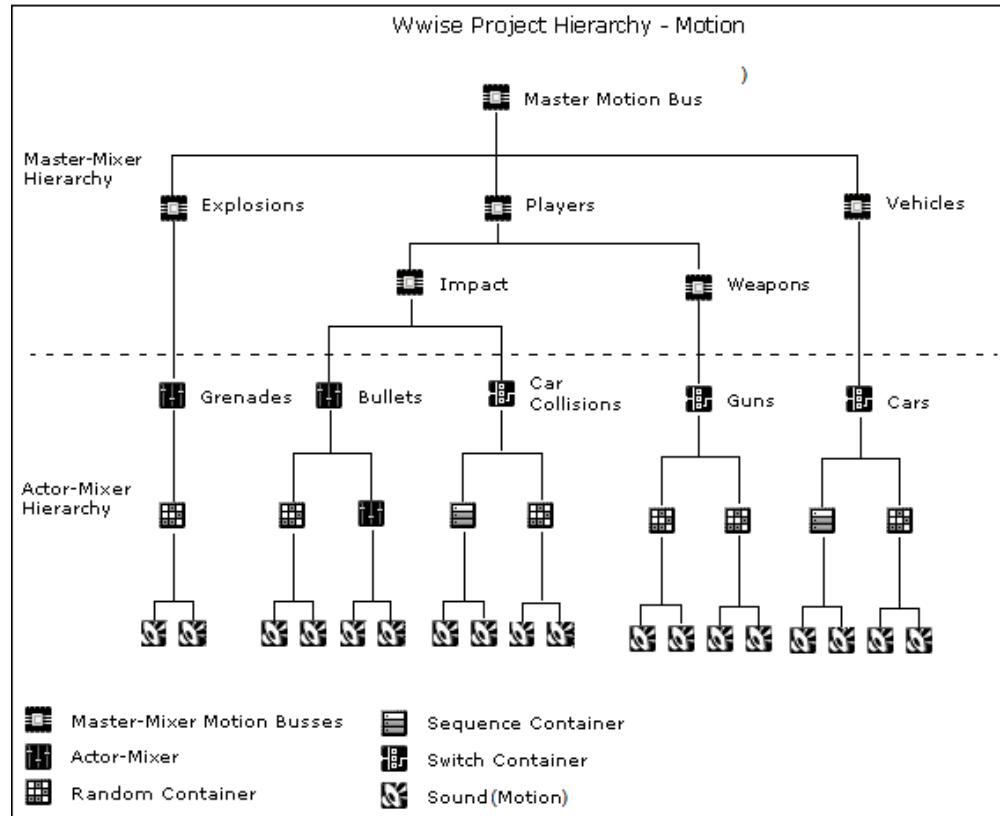
4. Audio Bus Property Editorで、Audio Deviceプロパティの選択ボタン (">>") をクリックする。

使用可能なオーディオデバイス・プラグインが表示されます。

5. リストで、Wwise\_Motionを選択する。

プロジェクト内に作成できるマスター・バスの数に制限はありません。バス構造を構築する時は、モーションバスもオーディオバスも、同じ機能、ルール、条件などが適用されます。モーションソースもオーディオソースも、モーション対応バスに

ルーティングできます。どちらも、信号がモーションバスに付属するオーディオデバイスプラグインによって、モーションデータに変換されます。



プロジェクト用のアウトプット構造の構築については「[概要](#)」を参照してください。

## モーションに関する、コツとベストプラクティス

ゲームのモーションを作成する前に、SDKドキュメンテーションに目を通して、Wwiseでモーションを効率的に管理するための事例、コツ、ベストプラクティスを参照してください。以下の「[モーションデバイスの制約](#)」も確認してみてください。

### モーション関連SDKドキュメンテーション

- セカンダリアウトプットをインテグレートする
- Wwise Motionをインテグレートする

### モーションデバイスの制約

- ゲームコントローラの再生機能には限界があります。ゲームコントローラで再生できるモーションエフェクトが限られているため、多様なモーションエフェクトの違いをプレイヤーが感じ取るのが難しく、特に複数のエフェクトが同時に再生されると、分かりにくくなります。このため、一度に再生できるモーション数を1つに絞るようにすると良いでしょう。

- ・ ゲームコントローラは、高電力型USBポートに接続してください。ゲームコントローラがモーションオブジェクトを再生しない時は、ゲームコントローラが高電力型USBポートに接続されているか、確認してください。USBポートの電源がモーションデバイスを動かすのに不充分だと、OSやデバイスを保護するために、システムがデバイスをアンマウントします。なお、コンピュータの前面にあるUSBポートはモーションデバイスを稼働させるだけの電力を提供できないことが多いので、後面にあるUSBポートに接続してください。

**audio**kinetic

## パート IV. ゲームとのイ ンタラクション



---

15. イベントの管理 .....	350
概要 .....	351
イベントの作成 .....	355
イベントの使い方 .....	362
イベントに関する、コツとベストプラクティス .....	365
16. ダイナミックダイアログの管理 .....	367
概要 .....	368
ダイナミックダイアログシステムを理解する .....	368
ダイアログイベントの作成 .....	372
ダイアログイベントの活用 .....	382
ダイアログイベントに関する、コツとベストプラクティス .....	386
17. Stateの活用 .....	388
概要 .....	389
Stateの活用 .....	390
同じStateやState Groupの、State間のトランジションを設定する .....	393
オブジェクトやバスにStateをアサインする .....	394
Stateに関する、コツとベストプラクティス .....	399
18. Switchの使い方 .....	400
概要 .....	401
Switchの使い方 .....	402
Switchに、Game Parameter値をマッピングする .....	405
Switchに関する、コツとベストプラクティス .....	405
19. RTPCの使い方 .....	407
概要 .....	408
RTPCで使う、Game Parameterの管理 .....	409
Game Parameterを使って、プロパティ値をコントロールする .....	414
LFOの活用 .....	421
エンベロープの活用 .....	422
ゲームオブジェクトの観察 .....	424
RTPCに関する、コツとベストプラクティス .....	425
20. Triggerの使い方 .....	427
概要 .....	428
Triggerの使い方 .....	428
21. ダイナミックダイアログの、StateやState Groupの活用 .....	431
概要 .....	432
StateやState Groupの活用 .....	433

---

## 第15章 イベントの管理

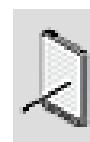
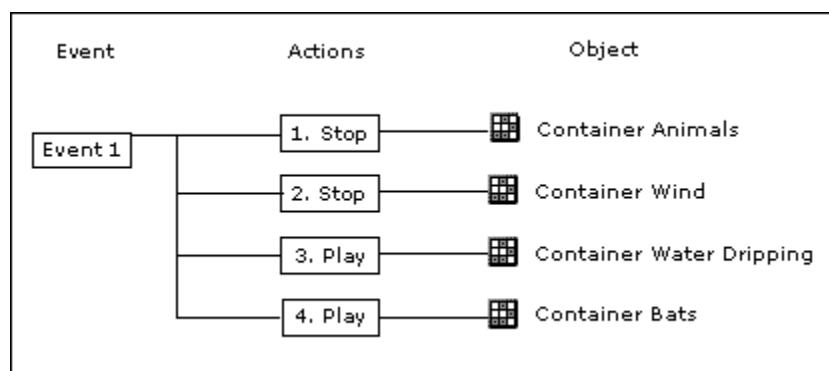
概要 .....	351
イベントの作成 .....	355
イベントの使い方 .....	362
イベントに関する、コツとベストプラクティス .....	365

## 概要

Wwiseでは、ゲーム中のサウンド、ミュージック、ダイアログ、モーションを、イベントで動かします。ゲーム中に、どのサウンド、ミュージック、モーション、またはダイアログ（台詞）を再生するのかは、イベントによって決まります。できるだけ多くの状況に対応するために、Wwiseでは以下の2種類のイベントを使います。

- アクションイベント
- Dialogue Events

“Action” Eventには、プロジェクト階層の様々なサウンド構造、ミュージック構造、モーション構造に適用される、1つ以上のActionが含まれます。選択したアクションによって、Wwiseのオブジェクト（Object）の再生、一時停止、停止などが決まります。例えば、以下のイベントは、ゲームのキャラクターが、風の強い野原から不気味な洞窟に入る場面を表しています。



### 注記

一般的にPlayイベントは、Stopや、場合によってはPauseやResumeなどのアクションイベントと組み合わせます。また、この種のActionは、個別にイベントを作成する代わりに、Wwise SDKを使って、サウンドに対するアクションを実行できます。詳しくは「[プログラムを使った、サウンドの停止、一時停止、再開](#)」を参照してください。

一方、ダイアログイベントは、再生するオブジェクトをダイナミックに決定するための、引数からなる一種のデシジョンツリーを使います。ダイアログイベントについては[16章ダイナミックダイアログの管理](#)を参照してください。

Wwiseでアクションイベントを作成してから、それをゲームエンジンに実装してゲーム中に適宜、呼び出せるようにします。ゲームエンジンはイベントの名前またはIDを使うため、イベントを作成してゲームに実装した後に、イベント内容の構築や微調整をしても、ゲームに再度実装する必要がありません。イベント名やIDを変更しない限り、追加のプログラミングは不要です。よって、様々なオブジェクトを自由に試しながら、オブジェクトの追加や削除を行い、オブジェクト間のトランジションの最終調整ができます。

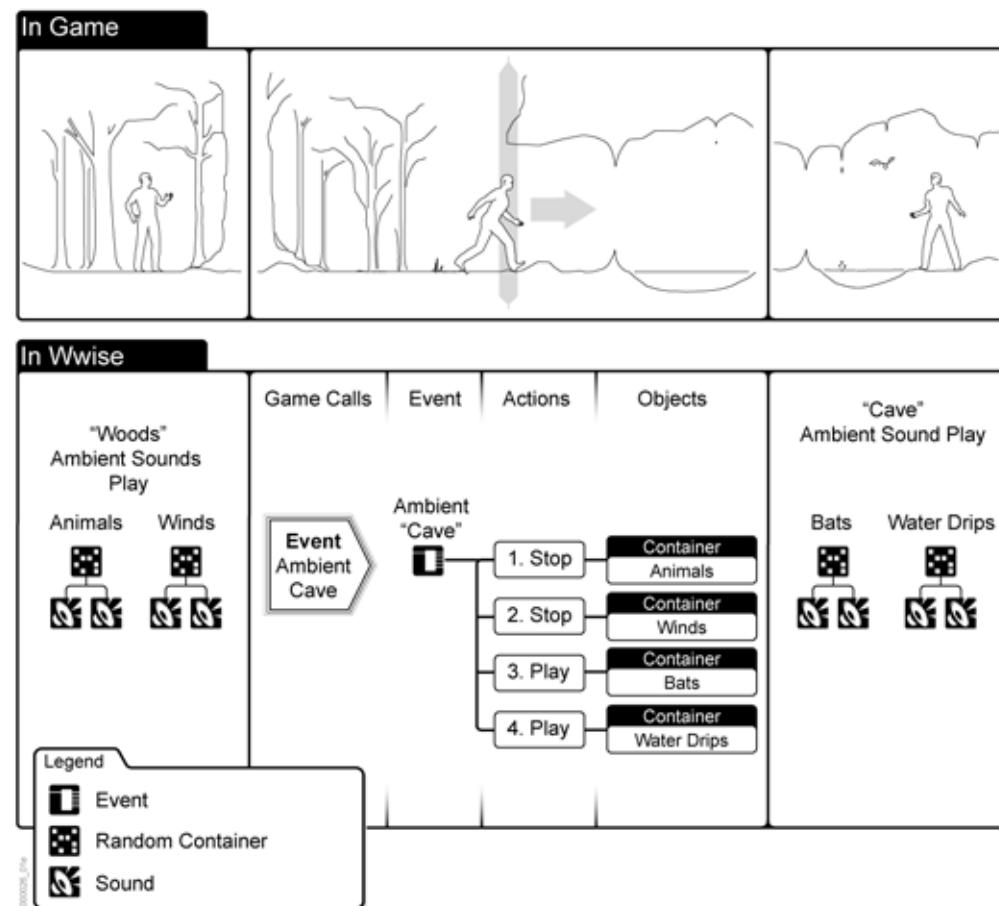
インターフェース上で分かりやすいように、イベントは、以下のアイコンで表されます。

アイコン	内容
	Event

### Eventの活用例

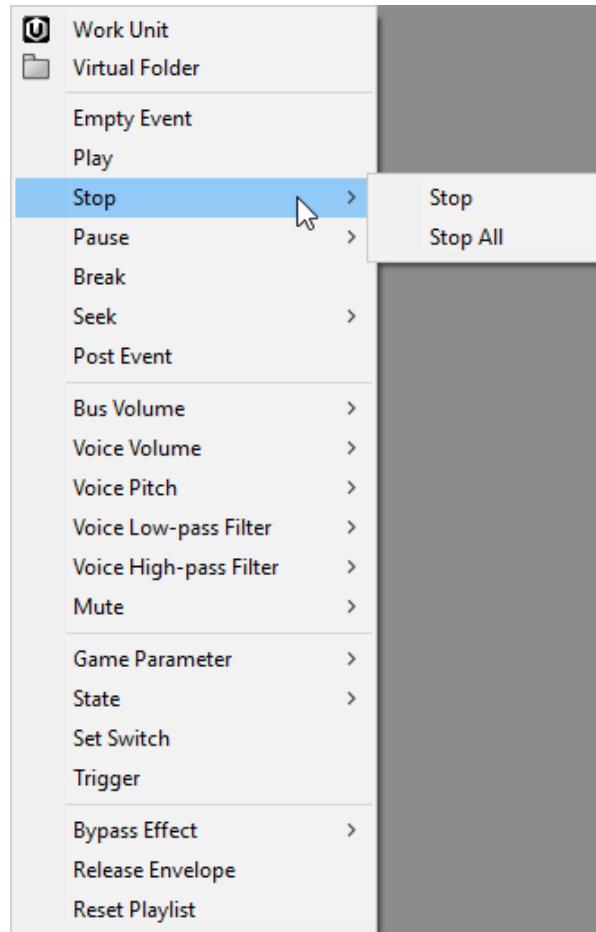
例えば、ファンタジー系ロールプレイングゲームの制作を考えます。ゲームのあるレベルで、ゲームのキャラクターが、森から洞窟に入ります。キャラクターが洞窟に入った瞬間、アンビエントサウンドを変更することになります。プロジェクトの初期段階で、仮のサウンドやプレースホルダサウンドを使って、Wwiseでイベントを作成しておきます。このイベントに、「Woods (森)」のアンビエントサウンドを停止し、「Cave (洞窟)」のアンビエントサウンドを再生する、一連のアクションを入れます。イベントを作成してゲームに実装すると、適切な時にトリガーできます。最初の実装が済むと追加のプログラミングが不要となり、様々なサウンドを試し、アクションの追加や削除を行い、アクションのプロパティを変更しながら、希望のサウンドが得られるまで調整できます。

ゲーム中のアクションとそのイベント、そして再生されるサウンドの関係を、下図に示します。



## Event Actionの種類

Wwiseには、ゲームのサウンド、ミュージック、モーションを動かすための様々なアクションがそろっています。アクションはカテゴリーごとに分類され、各カテゴリーの複数のアクションから選択できます。



各アクションのプロパティでディレイやフェードなどを設定して、これから始まるオブジェクトや終了するオブジェクトの管理を、より細かく行えます。Event Actionの概要を、下表に示します。

 **注記**

下表に示す「概要」は一般的な内容であり、各アクションで設定するスコープを考慮していません。スコープをGame ObjectまたはGlobalに設定することで、ゲーム中にそのアクションをオブジェクトに適用する範囲を、選択します。

Event Action	内容
Empty Event	アクションやイベントが入っていない。
Play	対象オブジェクトを再生する。
Stop > Stop	対象オブジェクトの再生を停止する。
Stop > Stop All	全てのWwiseオブジェクトの再生を停止するが、除外対象を追加できる。
Pause > Pause	対象オブジェクトの再生を、一時停止する。

Event Action	内容
Pause > Pause All	全てのWwiseオブジェクトの再生を一時停止するが、除外対象を追加できる。
Resume > Resume	一時停止した対象オブジェクトの再生を再開する。
Resume > Resume All	一時停止した全てのWwiseオブジェクトの再生を再開するが、除外対象を追加できる。
Break	ループしているサウンド、またはContinuous（継続）コンテナの、再生を停止しながらも、現在のオブジェクト（単数または複数）を最後まで再生する。
Seek > Seek	対象Wwiseオブジェクトの再生ポジションを変更する。現在、再生中でないオブジェクトには、影響しない。
Seek > Seek All	全てのWwiseオブジェクトの再生ポジションを変更するが、除外対象を追加できる。現在、再生中でないオブジェクトには、影響しない。
Post Event	イベントをほかのイベントの中からトリガーする。
#	
Bus Volume > Set Voice Volume	対象バスのボリュームレベルを変更する。
Bus Volume > Reset Volume	対象バスのボリュームを元のレベルに戻す。
Bus Volume > Reset Volume All	全バスのボリュームを、元のレベルに戻すが、除外対象を追加できる。
Voice Volume > Set Voice Volume	対象オブジェクトのボリュームレベルを変更する。
Voice Volume > Reset Volume	対象オブジェクトのボリュームを、元のレベルに戻す。
Voice Volume > Reset Volume All	全オブジェクトのボリュームを、元のレベルに戻すが、除外対象を追加できる。
Voice Pitch > Set Voice Pitch	対象オブジェクトのピッチを変更する。
Voice Pitch > Reset Pitch	対象オブジェクトのピッチを、元の数値に戻す。
Voice Pitch > Reset Voice Pitch All	全てのオブジェクトのピッチを元の数値に戻すが、除外対象を追加できる。
Voice Low-pass Filter > Set Voice Low-pass Filter	対象Wwiseオブジェクトのローパスフィルタ（LPF）適用量を変更する。
Voice Low-pass Filter > Reset Voice Low-pass Filter	対象Wwiseオブジェクトのローパスフィルタ（LPF）適用量を、元の数値に戻す。
Voice Low-pass Filter > Reset Voice Low-pass Filter All	全てのWwiseオブジェクトのローパスフィルタ（LPF）適用量を元の数値に戻すが、除外対象を追加できる。
Voice High-pass Filter > Set Voice High-pass Filter	対象Wwiseオブジェクトのローパスフィルタ（HPF）適用量を変更する。
Voice High-pass Filter > Reset Voice High-pass Filter	対象Wwiseオブジェクトのローパスフィルタ（HPF）適用量を、元の数値に戻す。
Voice High-pass Filter > Reset Voice High-pass Filter All	全てのWwiseオブジェクトのハイパスフィルタ（HPF）適用量を元の数値に戻すが、除外対象を追加できる。
Mute > Mute	対象オブジェクトを無音にする。
Mute > Unmute	対象オブジェクトを「無音状態」以前の、元のボリュームレベルに戻す。
Mute > Unmute All	全てのオブジェクトをサイレンス前の元のボリュームレベルに戻すが、除外対象を追加できる。
#	
Game Parameter > Set Game Parameter	Game Parameter値を変更する。

Event Action	内容
Game Parameter > Reset Game Parameter	Game Parameterを元の数値に戻す。
States > Set State	指定Stateを起動する。
States > Enable State	Disable State Actionを適用した後の対象Wwiseオブジェクトに、該当するStateまたはStatesを再度有効にする。
States > Disable State	対象WwiseオブジェクトのStateまたはStatesを無効にする。
Set Switch	指定Switchを起動する
Trigger	スティンガーを起動させるTriggerを、呼び出す。
#	
Bypass Effect > Enable Bypass	対象Wwiseオブジェクトに適用するエフェクト（単数または複数）を、バイパスさせる。
Bypass Effect > Disable Bypass	エフェクトバイパスを取り消して、対象Wwiseオブジェクトのエフェクト（単数または複数）を、再度、適用する。
Bypass Effect > Reset Bypass Effect	対象オブジェクトのエフェクトバイパス設定を、元の設定に戻す。
Bypass Effect > Reset Bypass Effect All	全てのバイパスエフェクト設定を元の設定に戻すが、除外対象を追加できる。
Release Envelope	対象Wwiseオブジェクトに関連するエンベロープをリリースする。
Reset Playlist	指定したRandom ContainerまたはSequence Containerのプレイリストを初期状態にリセットする。Continuousモードの再生や現在再生中のサウンドには影響しない。

## イベントの作成

ゲームの全てのオブジェクトが、イベントによって動かされます。イベントの作成は、以下の手順を伴います。

- ・「新規イベントの作成」
- ・「イベントに、アクションを追加する」
- ・「Event Actionに、ターゲットをアサインする」
- ・「Event Actionのスコープの設定」
- ・「Event Actionのプロパティの設定」

細かく柔軟にコントロールできるように、イベントで1つのアクションを実行することも、一連のアクションを実行することも可能です。イベントの管理は、Event Editorで行います。

複数プラットフォームを対象にオーサリングする場合は、特定のアクションを1つのプラットフォームから除外（Exclude）することも考えられます。デフォルトで、全てのアクションがイベントに含まれますが、これをプラットフォームごとにカスタマイズできます。複数プラットフォーム向けにオーサリングする方法については「1つのプラットフォームで、プロジェクトのエレメントを除外する」を参照してください。

チームでプロジェクト作業を進める場合、複数のイベントを別々のWork Unitに分けることで、各メンバーが同時に違うイベントの作業をできます。Work Unitを使った作業については「[プロジェクトをWork Unitで分割](#)」を参照してください。

### 新規イベントの作成

新規イベントを作成するには、以下の方法があります。

- アクションやターゲットの入っていない、空のイベントを作成する。
- ある特定のアクションが入ったイベントを作成する。
- 1つのアクションと、1つのターゲットが入ったイベントを作成する。



#### Tip

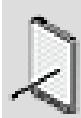
Eventビューアで右クリックして、イベントを作成することもできます。Eventビューアでイベントを追加する時は、そのイベントを特定のWork Unitにアサインする必要があります。

### 空のイベントを作成するには:

- Project Explorerで、Eventsタブに切り替える。
- 以下のいずれかを行う:
  - ワークユニット（または仮想フォルダ）を選択して、Project Explorerのツールバーの、Eventアイコンをクリックする。
  - Work Unitまたは仮想フォルダを右クリックして、ショートカットメニューで、New Child > Empty Event を選択する。

Project Explorerで選択したWork Unit（または仮想フォルダ）に、新規イベントが作成されます。

- デフォルトのイベント名を、内容が分かる名前に変更する。



#### 注記

Event名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

### アクション付きのイベントを作成するには:

- Project Explorerで、Eventsタブに切り替える。
- イベントを追加するWork Unit（または仮想フォルダ）を、右クリックする。
- ショートカットメニューで、New Childを選択して、Event Actionリストを表示する。
- Actionリストから、ActionカテゴリーまたはActionを1つ選択する。前者の場合は、サブメニューに複数のActionが表示されるので、1つ選択する。

選択したアクションの入った新規イベントが、Project Explorerで選択したワークユニット内に、作成されます。

- デフォルトのイベント名を、内容が分かる名前に変更する。

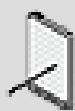


### 注記

Event名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

アクションとターゲットの両方を含む、イベント（単数または複数）を作成するには：

1. Project Explorerの、Audioタブで、イベントに入れるターゲットを1つ以上選択して、それを右クリックする。  
ショートカットメニューが表示される。
2. 以下のオプションを1つ選択する。
  - New Eventは、選択したオブジェクトの入ったイベントを作成する場合。
  - New Events (One event per object)は、選択したオブジェクトごとに、イベントを1つ作成する場合。
  - New Event (Single event for all objects)は、選択した全てのオブジェクトが入った、1つのイベントを作成する場合。
3. Actionリストから、ActionカテゴリーまたはActionを1つ選択する。前者の場合は、サブメニューに複数のActionが表示されるので、1つ選択する。  
選択したアクション（単数）とオブジェクト（単数または複数）が入った1つ以上のイベントが、Event Editorで作成される。
4. Nameフィールドでデフォルトのイベント名を、内容が分かる名前に変更する。



### 注記

Event名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

## 関連トピック

- 「イベントに、アクションを追加する」
- 「Event Actionに、ターゲットをアサインする」
- 「Event Actionのスコープの設定」
- 「Event Actionのプロパティの設定」

### イベントに、アクションを追加する

イベントに入れるアクションを設定する必要があります。1つのイベントに、1つ以上のアクションを入れることができます。

Eventに、Actionを追加するには：

1. Event Editorで、Add >>ボタンを選択する。  
Actionリストが表示される。

2. Actionリストから、ActionまたはActionカテゴリーを1つ選択する。前者の場合は、サブメニューに複数のActionが表示されるので、1つ選択する。

選択したActionが、イベントに追加される。

イベントアクションにターゲットをアサインすることも、イベントにアクションをさらに追加することもできます。

ほかにも、以下のようにEventにActionを追加する方法がいくつかあります：

- 1つ以上のエレメントをProject Explorerから、Event EditorのEvent Actionsペインの空いたスペースにドロップできます。エレメントに適したActionとなりますが、これは自由に変更できます。
- Event EditorのEvent Actionsペインの空いたスペースでショートカットメニューを開いて、New Actionオプションを選択できます。これで、「Eventに、Actionを追加するに」で説明したAdd >>ボタンをクリックした時と同じオプションが、表示されます。
- 一般的なコピー&ペーストのショートカットやショートカットメニューのオプションを使って、同時に1つまたは複数のActionをコピー&ペーストできます。

## 関連トピック

- 「Event Actionの種類」
- 「新規イベントの作成」
- 「Event Actionに、ターゲットをアサインする」
- 「Event Actionのスコープの設定」
- 「Event Actionのプロパティの設定」
- 「イベントの再生」
- 「イベントの使い方」
- 「イベントの名前変更」
- 「イベントのアクションを削除する」

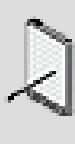
## Event Actionに、ターゲットをアサインする

一般的にEvent Actionは、特定のオブジェクト、構造、またはゲームシンクにアサインする必要があります。対象ターゲットのないアクションが1つ以上入っているイベントを、オーファン（孤立）イベントと呼びます。オーファンEventはEvent ViewerのOrphansタブに表示されます。また、プロジェクトのインテグレティレポートを生成する時にも表示されます。

イベント内のオブジェクトのステータスが分かるように、以下の通り、オブジェクト名が色分けされます。

- 白 - 含まれるオブジェクト。（現在のプラットフォーム内）
- グレー - 含まれないオブジェクト。（現在のプラットフォーム内）
- 赤 - 対象オブジェクトがないEvent Actionや現在のプロジェクトにないオブジェクト。

- ・ 黄 - 現在のプロジェクトに、今はロードされていないオブジェクト。



### 注記

Trigger、Set Switch、Set Stateなど、一部のアクションにミュージックオブジェクトをアサインする場合は、そのアクションが発生するミュージックオブジェクト内で既に指定されているポイントが原因で、アクションの結果にディレイが生じることがあるので注意してください。

## Event Actionに、ターゲットをアサインするには:

1. Event Editorで、ターゲットをアサインするActionを選択する。
2. Browseをクリックする。

Project Explorer - Browserが表示される。

3. 階層内を検索して、Actionにアサインするオブジェクトを選択する。
4. OKをクリックする。

オブジェクトが、選択したActionにアサインされる。



### Tip

オブジェクトをProject Explorerから、Event ActionsリストにあるActionにドラッグすることで、オブジェクトをEvent Actionにアサインすることもできます。

## 関連トピック

- ・ 「新規イベントの作成」
- ・ 「イベントに、アクションを追加する」
- ・ 「Event Actionのスコープの設定」
- ・ 「Event Actionのプロパティの設定」
- ・ 「イベントの再生」

## Event Actionのスコープの設定

イベントを作成する時に、各Actionのスコープ（範囲）を設定する必要があります。スコープによって、そのActionをゲーム中のオブジェクトに適用する範囲が決まります。全てのゲームオブジェクトに適用するグローバルスコープと、あるイベントをトリガーした特定ゲームオブジェクトに限定するスコープがあります。また、スコープの設定を選択できるアクションと、既にスコープが決まっているアクションがあります。

例えば、プレイヤーがゲーム本編を出てメニュー画面に入る時のイベントについて考えます。このイベントによって、“Enter\_Menu”サウンドが再生され、プレイヤーに関連するサウンドが全て一時停止して、ステートが“Menu”になります。続いて“Menu”ステートによってMaster Audio Busのボリュームが20 dB下げられ、“Music”Audio Busのボリュームが20 dB上げられます。

上記の各Event Actionのスコープを、下表に示します。

Event Action	Scope	Comments
Play > Menu_Enter	Game Object	Playイベントは、必ず1つのゲームオブジェクトでトリガーするため、スコープをGame Objectとする。
Pause All	GlobalまたはGame Object	今回のシナリオで、イベントを呼び出すゲームオブジェクトはプレイヤーである。プレイヤー関連のサウンドを一時停止することで、メニュー閲覧中に余計な音を排除できる。つまり、ここでGlobal Scopeを指定することは可能だが、この状況でGame Object Scopeとすれば、プレイヤーは引き続きゲームの他のサウンドを聞ける。
Set State > Menu	Global	<p>Set Stateのスコープは常にGlobalであり、理由はこのステートがプロジェクトのどこに出てきても適用できるからである。今回のシナリオでは、Master Audio Busと"Music" Audio Busに、"Menu" ステートを設定してある。前者のボリュームは下がり、後者のボリュームは上がる。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <b>注記</b>            この例はScopeの説明用であって、オーディオデザインの選択肢として勧めているわけではありません。実際には、両方のバスでミキシングするのであれば、ボリューム変更の適用に非常に短いディレイが伴うことがあります。何らかのクリック音やバズ音が発生する可能性があります。さらに、"Music" Audio Busのボリュームを一定に保つことは、このバスにリニア型エフェクトがある場合は確実ではありません。ただしその場合には、Bypass Effect Actionを使うこともできます。         </div>

## イベントのスコープを設定するには:

1. Scopeリストから、以下のオプションを1つ選択する。

- Game objectで、イベントアクションを、そのイベントをトリガーしたゲームオブジェクトに適用する。
- Globalで、Event Actionを、全ゲームオブジェクトに適用する。

## 関連トピック

- 「新規イベントの作成」
- 「イベントに、アクションを追加する」
- 「Event Actionに、ターゲットをアサインする」
- 「Event Actionのプロパティの設定」
- 「イベントの再生」

## Event Actionのプロパティの設定

Event Actionごとに様々な関連プロパティを設定して、ゲームのサウンド、ミュージック、モーションをさらに細かく調整できます。

Actionによって設定できるプロパティが異なりますが、プロパティは、以下のカテゴリーに分かれます。

- Delays
- Transitions
- Bypass effect properties

- Volume, Pitch, LPF, Game Parameter, Seek, State, or Switch settings.



### 注記

Trigger、Set State、Set Switchなど、一部のEvent Actionで、ミュージックオブジェクトが関連するものに、ディレイプロパティを設定する場合は、そのActionが発生するミュージックオブジェクト内で既に指定されているポイントが原因で、実際のディレイが指定したディレイよりも長くなることがあるので注意してください。

## EventのActionプロパティを設定するには:

- Event Editorで、Event Actionsペインからアクションを選択する。

選択したアクションに関連するプロパティが、Action Propertiesペイン（右側）に表示されます。

- 必要に応じて、関連プロパティ値を指定する。

## 関連トピック

- 「新規イベントの作成」
- 「イベントに、アクションを追加する」
- 「Event Actionに、ターゲットをアサインする」
- 「Event Actionのスコープの設定」
- 「イベントの再生」

## イベントの再生

制作過程において、いつでも、イベントを試聴できます。



### 注記

モーションを試すには、対象モーションデバイスをワークステーションに接続する必要があります。

## イベントを再生するには:

- 以下のいずれかを行う:
  - Event Viewerで、イベントを1つ選択する。
  - Event Editorに、イベントを1つロードする。

このイベントが、Transport Controlにロードされる。

- Transport ControlのPlayアイコンをクリックする。

イベントが再生される。



### 注記

Soundcaster（サウンドキャスター）を使ってイベントを再生することもできます。Soundcasterの使い方については「[Soundcasterを使って試聴する](#)」を参照。

## 関連トピック

- ・「新規イベントの作成」
- ・「イベントに、アクションを追加する」
- ・「Event Actionに、ターゲットをアサインする」
- ・「Event Actionのスコープの設定」
- ・「Event Actionのプロパティの設定」

## イベントの使い方

イベントは、最初の実装が済むと追加のプログラミングが不要なため、様々なオブジェクトを試す、既存オブジェクトのプロパティを変更、またActionの追加や削除を行い、Actionのプロパティを変更しながら、希望する効果が得られるまで調整できます。

イベントを管理するためのタスクを、以下に示します。

- ・「イベントの名前変更」
- ・「イベントのアクションを削除する」
- ・「Event Actionにアサインしたターゲットを置換する」
- ・「イベントの削除」

### イベントの名前変更

イベントが作成されると、Wwiseが自動的に名前を付けます。このイベント名を、内容が分かるものに変更する習慣をつけると良いでしょう。イベントの名前は必ず固有のものとし、文字、数字、アンダースコアのみを使用してください。また、名前は必ず文字またはアンダースコアで始めます。

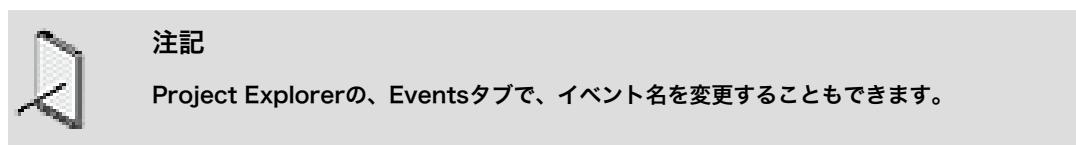
イベントをゲームに実装した後に名前を変更すると、追加のプログラミングが必要となるため、必要不可欠でなければ避けてください。

#### イベント名を変更するには:

1. Editorで、Name フィールドをクリックする。

  イベント名がハイライト表示される。

2. イベントの新しい名前を入力する。



## 関連トピック

- ・「新規イベントの作成」
- ・「イベントのアクションを削除する」

- ・「Event Actionにアサインしたターゲットを置換する」
- ・「イベントの削除」
- ・「イベントに関する、コツとベストプラクティス」

### イベントのアクションを削除する

実験しながらイベントを構築する過程で、イベントの1つ以上のActionを、削除する必要があるかもしれません。イベント名を変更しない限り、Actionを削除しても追加のプログラミングは不要です。

#### イベントから、Actionを削除するには:

1. Event Editorで、イベントから削除するActionを選択する。
2. Removeボタンをクリックする。

イベントから、そのアクションが削除されます。



#### Tip

イベントの中のActionを選択してDeleteキーを押すか、ActionのショートカットメニューでDeleteを選択して、そのActionを削除することもできます。

### 関連トピック

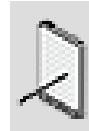
- ・「イベントに、アクションを追加する」
- ・「イベントの名前変更」
- ・「Event Actionにアサインしたターゲットを置換する」
- ・「イベントの削除」

### Event Actionにアサインしたターゲットを置換する

設定したターゲットのオブジェクトを、他のものに入れ替えながら、ゲームに合うものを見つけることができます。Eventをゲームに実装した後でも、様々なサウンド、ミュージック、モーション、Actionなどを使って自由に試すことができます。

#### Actionにアサインしたターゲットのオブジェクトを、置換するには:

1. Event Editorで、オブジェクトの置換をするActionを選択する。
2. Browse...をクリックする。



#### 注記

Browse...ボタンはEvent Editorの下部にあるほか、Target列でショートカットメニューを開きSet Targetを選択した時にも、Browseオプションが表示されます。

または、選択したActionのターゲットのショートカットメニューを開く。

Project Explorer - Browserが表示される。

3. 階層内を検索して、Actionにアサインする新しいターゲットを選択する。

#### 4. OKをクリックする。

Actionの新しいターゲットオブジェクトが、Target列に表示される。



#### Tip

Actionタイプに適した別のオブジェクトをProject Explorerから、Event EditorにあるEvent Actionにドラッグすることで、Actionのターゲットを置換することもできます。Actionの新しいターゲットオブジェクトが、Target列に表示されます。

オブジェクトがActionとして適していなければ、ドラッグできないようになっています。

### 関連トピック

- 「Event Actionに、ターゲットをアサインする」
- 「新規イベントの作成」
- 「イベントの削除」

### イベントのオブジェクトを、Schematic Viewで表示する

イベントに含まれる特定のオブジェクトに関するプロジェクト構造、つまりパイプラインを確認するには、Schematic Viewで簡単に表示できます。

### イベントターゲットのパイプラインを、Schematic Viewで表示するには

1. Event Editorで、パイプラインを表示させるターゲットオブジェクトを右クリックする。

ショートカットメニューが表示される。

2. Show in Schematic Viewをクリックする。

Schematic Viewに、サウンド、またはミュージックのパイプラインが表示される。

### 関連トピック

- 「Event Actionにアサインしたターゲットを置換する」
- 「Event Actionに、ターゲットをアサインする」
- 「新規イベントの作成」
- 「イベントの削除」
- 「イベントの再生」

### イベントの削除

不要となったイベントは、削除できます。イベントを削除する前に、他のチームメンバーが、それをプロジェクトの別の部分で使用していないか、または既にSoundBankの1つに含まれていないかを、確認すると良いでしょう。既にSoundBankに入っているイベントを、自分または他のチームメンバーが削除してしまうと、無効なイベントが発生します。Wwiseは、削除されたイベントやその

他の無効なプロジェクトエレメントをSoundBankから自動的に削除しないので、マニュアル操作で削除する必要があります。このようなイベントやオブジェクト構造をSoundBank内で発見しやすいように、Wwiseが、SoundBank Editorの、Addタブで、名前の後に「Missing」という単語を追加します。無効イベントについては「[SoundBankから、プロジェクトエレメントを削除する](#)」を参照してください。

### イベントを削除するには:

#### 1. 以下のいずれかを行う:

- Project Explorerの、Eventsタブで、削除するイベントを右クリックして、ショートカットメニューで、Delete Selectionを選択する。
- Event Viewerで、削除するイベントをクリックして、Deleteキーを押す。

イベントが削除される。



### 関連トピック

- 「新規イベントの作成」
- 「イベントのアクションを削除する」
- 「イベントの名前変更」
- 「SoundBank内のエレメントを検索する」

## イベントに関する、コツとベストプラクティス

プロジェクトのイベントを効率的に管理するためのコツやベストプラクティスについて、以下を参照してください。

### イベントの名前変更

イベントの名前を変更する前に、イベントがゲームでどのように実装されているかを確認してください。イベントの名前を含む文字列や、ヘッダファイル Wwise\_IDs.h にあるIDを含む文字列を使って実装している場合は、イベントの名前を変更すると、イベントを使い続けるために追加のプログラミングが必要となります。

### イベントの整理

プロジェクトのイベントを整理するために、ワークユニットや仮想フォルダを使うことができます。イベントをいくつかのWork Unitに分けると、複数の人が同時にプロジェクトで作業をしている場合に便利です。例えば、各チームメンバーが、自分のイベントWork Unitの中でイベントを作成できます。イベントを複数のWork Unitに分けることで、生産性やワークフローに影響を及ぼすことなく、マージコンフリクトの解消にかかる余計な時間と手間を回避できます。

## ゲームイベントとWwiseイベントの照合

SoundBankの定義ファイル (Definition file) を使い、ゲームに実装済みのイベント、足りないイベント、Wwiseでこれから作成すべきイベントなどを、確認できます。オーディオプログラマーがゲーム側のイベントリストを生成した後、定義ファイルをWwiseにインポートすることができます。ログファイルの情報を使い、ゲーム内のイベントにマッチするWwise側のイベントや、まだ作成できていないイベントを、確認できます。SoundBankの定義ファイルについては「[定義ファイルをインポートして、SoundBankの中身を入れる](#)」を参照してください。

## プログラムを使った、サウンドの停止、一時停止、再開

イベントに関する情報は、サウンドエンジンのデフォルトメモリプール (Default memory pool) に格納されます。デフォルトメモリプール内のスペースを使いすぎないために、Wwise SDKの、[ExecuteActionOnEvent\(\)](#) 関数を使い、サウンドの停止、一時停止、再開をプログラミングすることも可能です。再生と停止の2つのイベントを作成する代わりに、再生イベントを1つ作成して、サウンドを停止するための、ExecuteActionOnEventの呼び出しを付けることができます。サウンドの一時停止と再開が必要な場合は、別々のイベントを作成せずに、同じ関数を使って一時停止と再開のActionを実行できます。これにより、プロジェクトのイベント数を大幅に削減でき、デフォルトメモリプールの貴重なスペースを解放できます。さらにコントロールの幅を広げるには、サウンドの停止、一停止、再開のフェードアウトタイムを、プログラミングで指定することもできます。ExecuteActionOnEvent関数については、Wwise SDKドキュメンテーションを参照してください。

## Set Voice Pitchイベントを使って、ピッチエンベロープを作成

実際の自然界では、多くの自然音が安定した状態、つまり維持すべきレベルに至るまでに、一定の間はピッチが上下、つまり変動します。このエフェクトを出すには、ピッチエンベロープを使います。Wwiseでは、プログラミング済みのピッチエンベロープに対応していませんが、複数の連続する、Set Voice Pitch Actionのイベントを作成して、近い効果を得られます。イベント内の、Set Voice Pitchアクションごとに、Actionプロパティの既定カーブのどれかに従い、一定時間に渡りピッチを調整させます。なお、この一連のActionの最後のActionは当然、Reset Voice Pitchとなり、ゲームオブジェクトを元のStateに戻す必要があるでしょう。

---

## 第16章 ダイナミックダイアログの管理

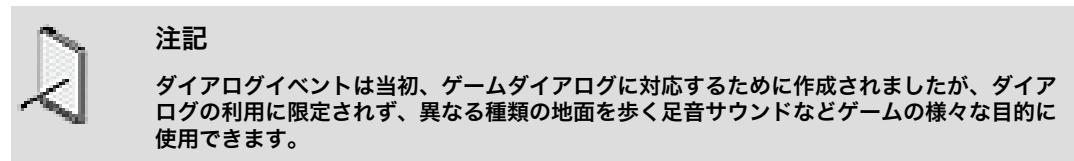
概要 .....	368
ダイナミックダイアログシステムを理解する .....	368
ダイアログイベントの作成 .....	372
ダイアログイベントの活用 .....	382
ダイアログイベントに関する、コツとベストプラクティス .....	386

## 概要

最近のゲームは、リアルタイムストラテジー系、スポーツ系、アドベンチャー系を含め、何らかのダイナミックなオーディオコンポーネントがあり、ゲーム中に起きるActionに応じてオーディオも変化します。ゲームで発生する条件や結果が多岐に渡れば、スクリプトライター オーディオデザイナー達は実に特殊な難題に直面します。全ての条件や結果に対応できるように、あらゆる台詞（ダイアログ）の組み合わせを用意するには、数千個ものアセットや、複雑なSwitch Containerの階層が必要になるかもしれません。これは、メモリへの大きな負担となります。このような状況でも、開発プロセスを合理化して全体のメモリ消費量を削減できるよう、Wwiseはダイナミックダイアログシステムを提供しています。

ダイナミックダイアログシステムは、Wwiseでダイナミックオーディオを構築して管理するための、軽くて効率的な手法です。デシジョンツリー構造に設定した複数のルールを使用して、ゲーム中の任意の時点に再生するダイアログを決定します。

これだけでも強力なシステムですが、Wwise SDKにある追加機能を利用すれば、ランタイムのダイナミックシーケンスを作成できるので、スポーツゲームの実況中継の台詞などに理想的です。Wwiseでは、ダイアログを単語や短いフレーズに分けて、これらをデシジョンツリー構造の特定パスにアサインします。ゲームプレイ中に、指定された単語がサウンドエンジンに送られ、ダイナミックに継ぎ合わされ、自由に流れるサンプルアキュレートな文章が形成されます。また、単語や短いフレーズが別々に存在するため、多様な組み合わせや再利用も可能です。ゲームにダイナミックシーケンスを取り込む方法については[Wwise SDKドキュメンテーション](#)を参照してください。



## ダイナミックダイアログシステムを理解する

Wwiseのダイナミックダイアログというシステムの中心にあるのが、ダイアログイベントであり、どのダイアログを再生するかを決める際のルールや条件のセットを提供します。ダイアログイベントによって、ゲーム内で存在する様々なシナリオ、条件、結果などを再現します。全ての状況に対応できるように、Wwiseでデフォルト条件、つまりフォールバック条件を作成することも可能です。

これらの条件は全て、複数のState GroupやStateを使って定義されます。さらに、State GroupやStateを組み合わせて作ったパスが、ゲームで起きる特定の条件や結果を決めます。このパスを1つずつ、Wwiseが対応するサウンドオブジェクトに結びつけます。ゲーム中にダイアログイベントが呼び出されると、ゲーム側で現在の条件をダイアログイベントの中にある複数のパスと比較します。ゲームにおける現在の条件にマッチする複数のパス、それぞれのパスのモード、Probability（発生確率）、そしてWeight（ウェイト付け）によって、どのダイアログを再生するのか、または全く再生しないのかを決定します。

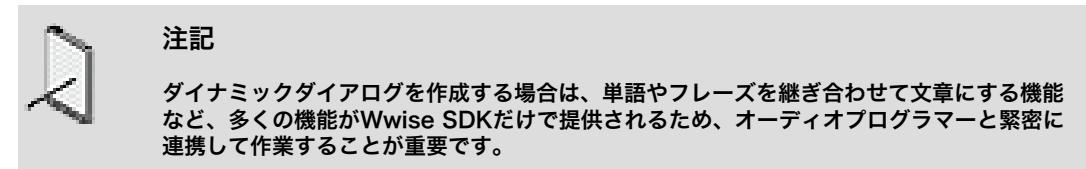
例えば、下記のダイアログイベントには、スポーツゲームの選手の名前に関連したStateやState Groupが入っています。各StateやState Groupの値を組み合わせて、

可能性のある複数のパスや条件が作られます。この例では、実況アナウンサーが選手の姓 (Last) または姓名 (Full) のどちらかを使うという設定です。

Dialogue Event: Name		
Arguments	Player Name	Name Length
Argument Values	Tony Cross	Full
	John Patrick	Last
Argument Paths		Assigned Object
Cross - Full		Cross_Full
Cross - Last		Cross_Last
Patrick - Full		Pat_Full
Patrick - Last		Pat_Last
Player Name - Name Length		He

万が一、ゲームの状況に合うStateがない場合のために、デフォルト、つまりフォールバック用のStateが入ったパスを作ることもできます。フォールバック用のパスには通常、Stateの代わりに1つ以上のStateやState Groupを入れ、汎用性のあるサウンドオブジェクトを設定します。前例の場合、フォールバック用のパスは、プレイヤーの名前の代わりに「He (彼)」というサウンドオブジェクトが設定されています。

ダイアログイベントに対して起こりうる全ての条件を設定した後に、ゲームエンジンに実装します。ゲームがダイアログイベントを呼び出すと、サウンドエンジン側で、ダイアログイベントに合うパスに対応するオーディオオブジェクトを返して、ダイアログイベントを完結させます。次に、そのオーディオオブジェクトをダイナミックシーケンスに挿入して再生するのかどうかを、サウンドエンジンが判断します。オーディオオブジェクトが返される回数と、それをダイナミックシーケンスにインサートする回数は、1:1とは限りません。完結される1件のダイアログイベントの中で、サウンドエンジンが返す1つのオーディオオブジェクトが、必要に応じて何度もダイナミックシーケンスに挿入されることもあります。



ゲームエンジンはダイアログイベントのイベント名を使うので、イベントを作成してゲームへの実装が済めば、イベントの中に入っているコンテンツの構築や微調整を行っても、再度実装をする必要はありません。この方法を取ることで追加のプログラミング作業をせずにStateの追加や削除ができ、様々なサウンドを柔軟に試せます。



### 注意

ダイアログイベントにあるStateやState Groupを、追加、削除、または移動すると、自動的にパスが変更されます。Wwise上の操作は簡単ですが、この種の変更はコードの更新も必要となるため、プログラマーによる対応を要します。

インターフェース上で分かりやすいように、ダイアログイベントは、以下のアイコンで表されます。

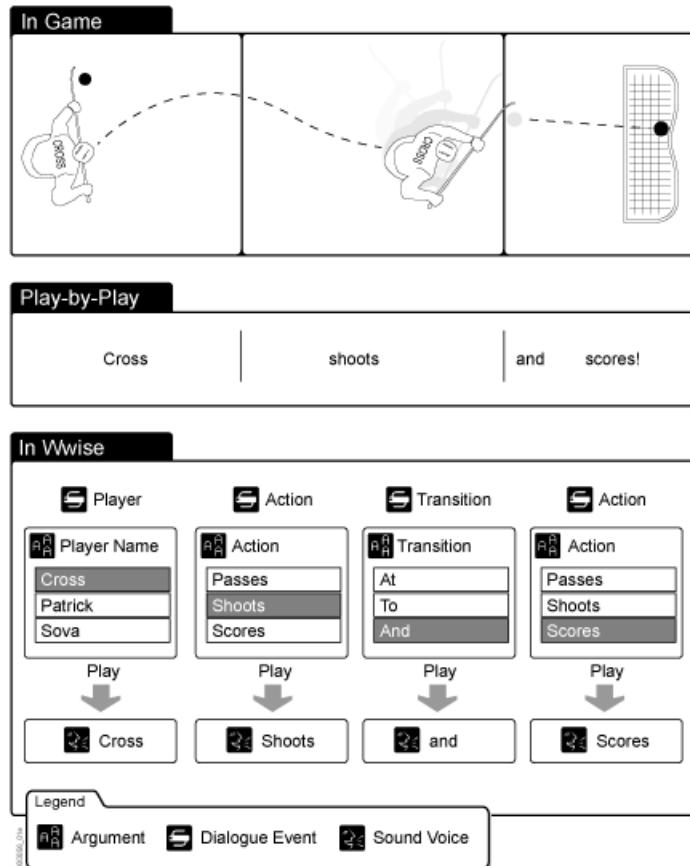
アイコン	内容
	Dialogue event ダイアログイベント

### ダイアログイベントの活用例

例えば、実況中継（Play-by-Play）付きのアイスホッケーゲームの場合を考えます。プレイヤーがシュートして得点した時は、プレイヤーのActionに応じた実況中継が流れるべきです。

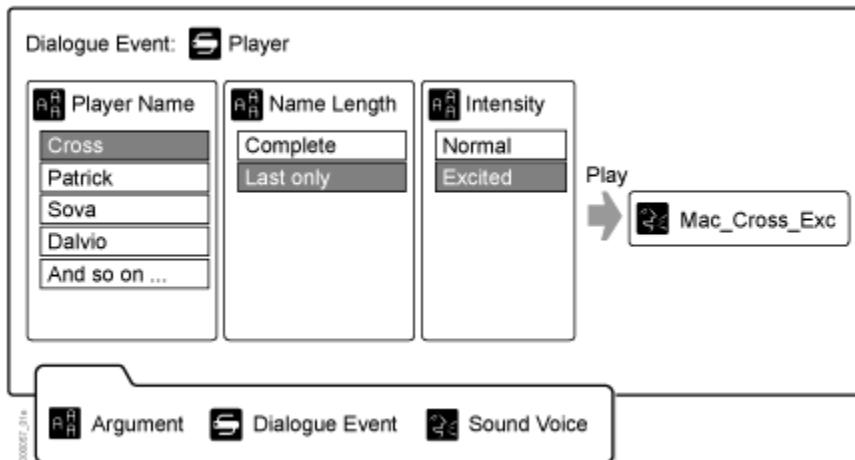
Wwiseで様々な可能性や結末を設定するために、まずダイアログイベントとして、「Player（プレイヤー）」、「Action（アクション）」、「Transition（接続詞）」などを作成します。全てのイベントに、それぞれ、このゲーム用に作成した複数のState GroupやStateを設定します。次に、各条件や結末を定義するパスを作り、それぞれのパスに、適切なボイスオブジェクトを割り当てます。ゲームプレイ中に、ゲーム側が現在のStateとWwiseで定義したパスを照会して再生すべきボイスオブジェクトを判断します。

下図は、「Cross shoots and scores（クロスがシュートし、得点した）」という実況中継を、Wwiseのダイアログイベントが生成する様子を示しています。



上図は、ゲームにおけるStateやState Groupやダイアログイベントの使い方を、簡略化したものです。しかし一般的には、より洗練されたダイアログイベントがゲームに必要で、多数のState GroupやStateが含まれます。Wwiseを使えば、さらにリアルなゲームのダイナミックダイアログをつくり出すために、多様の複雑なシナリオを設定できます。

以下の例は、より洗練されたPlayerダイアログイベントです。このダイアログイベントには、複数のStateやState Groupが設定されています。複数のStateやState Groupがある場合、選択された複数のStateによってパスが成立します。State GroupやStateの全ての組み合わせに対して、パスを1つずつ作成して、パスごとにオブジェクトをアサインできます。ゲーム中に、あるパスのStateが発生すると、そのパスにアサインされているボイスオブジェクトが再生されます。

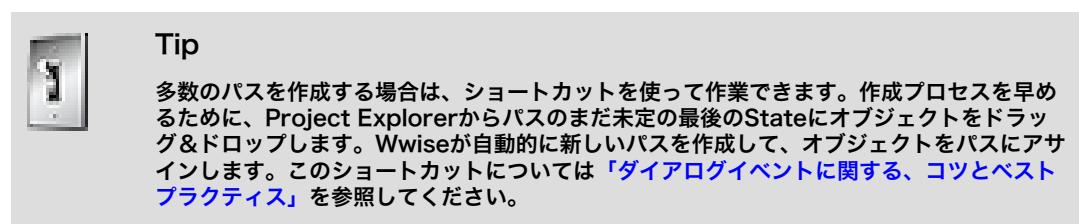
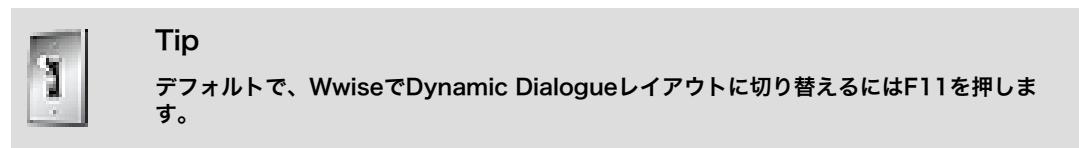


## ダイアログイベントの作成

ゲーム中のダイナミックダイアログのルールは、ダイアログイベントによって決まります。これらのイベントの作成には、以下の手順が伴います。

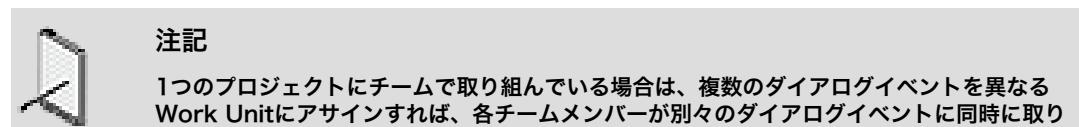
- 「新しいダイアログイベントの作成」
- 「ダイアログイベントを設定する」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスを使って、ゲームの条件を再現する」
- 「パスにオブジェクトをアサインする」

Dialogue Event Editorで、ダイアログイベントを管理できます。



### 新しいダイアログイベントの作成

プロジェクトの初期段階に充分な時間をとり、ゲームを扱いやすいコンポーネントに分けて分析して、ダイナミックオーディオ全体のニーズを把握する習慣をつけることが非常に有効となります。必要な情報が全てそろった時点で、ゲームに要求される全てのダイアログイベントのリストを作成します。リストを作成すると、Wwiseでダイアログイベントを作成しやすくなります。



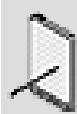
組むことができます。Work Unitを使った作業については「[プロジェクトをWork Unitで分割](#)」を参照してください。

## 新しいダイアログイベントを作成するには:

1. Project Explorerで、Eventsタブに切り替える。
2. Dynamic Dialogueセクションで、以下のいずれかを行う。
  - Work Unit（または仮想フォルダ）を選択して、Project ExplorerツールバーのDialogue Eventアイコンをクリックする。
  - ワークユニット（または仮想フォルダ）を右クリックして、ショートカットメニューでNew Child > Dialogue Eventを選択する。

新規ダイアログイベントが、選択したワークユニット内に作成されます。

3. デフォルトのダイアログイベント名を、内容が分かる名前に変更する。



### 注記

Dialogue Event名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

## 関連トピック

- 「[ダイアログイベントにStateやState Groupを追加する](#)」
- 「[ダイアログイベントを設定する](#)」
- 「[パスを使って、ゲームの条件を再現する](#)」
- 「[Stateを使って、パスを作成する](#)」
- 「[フォールバック用のパスを作成する](#)」
- 「[パスにオブジェクトをアサインする](#)」
- 「[パスのProbabilityとWeightを設定する](#)」
- 「[ダイアログイベントの削除](#)」
- 「[ダイアログイベントの中のオブジェクトの再生](#)」

## ダイアログイベントを設定する

ダイアログイベントごとに、Probability（発生確率）とModeを設定する必要があります。Probabilityを設定することで、ダイアログイベントが、オーディオオブジェクトをサウンドエンジンにサブミットする可能性を操作できるので、再生の可能性をコントロールできます。一旦サブミットされると、サウンドエンジンが、そのオーディオオブジェクトをダイナミックシーケンスにインサートするかどうかを判断します。Probabilityが100%であれば、そのダイアログイベントがゲームに呼び出される度に必ず1つのオーディオオブジェクトをサウンドエンジンにサブミットします。一方、Probabilityが0%であれば、オーディオオブジェクトがサブミットされることはありません。この設定は、ダイアログイベントがトリガーされる度にオーディオが再生されるのを避けたい時などに特に便利です。例えば、ファイティングゲームの試合で、選手同士の接触がある度にアナウンサーがコメントするのを避けるために使えます。このようなシナリオでオーディオが過剰にトリガーされるのを防ぐには、ダイアログイベントのProbabilityの設定を低くします。

モード設定は、ゲームの実際の条件に複数のパスがマッチするような状況で、ダイアログイベントの動作をコントロールするために使います。Modeで選択できるオプションは、Best Match（最も近い）と、Weighted（ウェイト付け）の2つです。

### ダイアログイベントの設定を決めるには

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. ダイアログイベントのProbability値を、0~100に設定する。Probabilityによって、ダイアログイベントがサウンドオブジェクトに対して再生用のオーディオオブジェクトをサブミットする可能性が定義される。



3. ランタイムに、複数の既定パスがトリガーされたStateとマッチする場合に、サウンドエンジンによるパスの選択基準を指定するために、Mode listリストから以下のオプションを1つ選択する。
  - Best Match（ベストマッチ）：ランタイムにトリガーされたステートに最も近い（ベストマッチ）パスを選択する。完全にマッチするものがなければ、ワイルドカード（\*）の最も少ないパスが選択される。
  - Weighted（ウェイト付け）：マッチする複数のパスのそれぞれのウェイト付けに基づき、ランダムに1つ選択する。

### 関連トピック

- 「新しいダイアログイベントの作成」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスを使って、ゲームの条件を再現する」
- 「Stateを使って、パスを作成する」
- 「フォールバック用のパスを作成する」
- 「パスにオブジェクトをアサインする」
- 「パスのProbabilityとWeightを設定する」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

### ダイアログイベントにStateやState Groupを追加する

ダイアログイベントには、ゲームに存在する様々な条件や結果を反映したルールのセットが含まれます。これらのルールは、State GroupやStateを使って定義されます。ダイアログイベントにStateやState Groupを追加すると、該当するStateも追加されます。

### Dialogue EventにState Groupを追加するには

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. Project Explorerで、Game Syncsタブに切り替える。

3. State Groupsセクションで、ダイアログイベントに追加するStateやState Groupを1つ以上選択する。
4. これらのStateやState Groupを、Dialogue Event EditorのState Groupsペインにドラッグする。

StateやState Groupと、該当するStateが、Dialogue Event Editorに表示される。

## 関連トピック

- 「新しいダイアログイベントの作成」
- 「ダイアログイベントを設定する」
- 「パスを使って、ゲームの条件を再現する」
- 「Stateを使って、パスを作成する」
- 「フォールバック用のパスを作成する」
- 「パスにオブジェクトをアサインする」
- 「パスのProbabilityとWeightを設定する」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

### パスを使って、ゲームの条件を再現する

ダイアログイベントのステートグループやステートを設定した後は、ゲームに存在する全ての条件や結末を再現して、ルールを定義する必要があります。様々な State GroupやStateを組み合わせてパスにまとめることで、この作業を進めます。

全ての状況や結末を網羅することは難しいため、具体的なパスが設定されていない状況が発生した時にサウンドエンジンが毎回使用できるフォールバックパス、つまりデフォルトパスを作成するのが一般的です。このフォールバックパスには、StateではなくStateやState Groupを使用します。

パスは、以下の2種類があります。

- 「Stateを使って、パスを作成する」
- 「フォールバック用のパスを作成する」

### Stateを使って、パスを作成する

ゲームの様々な条件や結末を再現するために、ダイアログイベントの複数のステートグループから、ステートを1つずつ選択します。選択したStateで、パスが形成されます。1つのパスに対して、1つの台詞（ダイアログ）を設定すると、ゲーム中にその条件または結末が発生した時に該当するボイスオブジェクトが再生されます。



#### Tip

多数のパスを作成する場合は、ショートカットを使って作業できます。作成プロセスを早めるために、Project Explorerからパスのまだ未定の最後のStateにオブジェクトをドラッグ＆ドロップします。Wwiseが自動的に新しいパスを作成して、オブジェクトをパスにアサインします。このショートカットについては「[ダイアログイベントに関する、コツとベストプラクティス](#)」を参照してください。

## パスを作成するには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. ダイアログイベントの全てのStateやState Groupに関して、それぞれStateをクリックする。

Add Pathが有効になる。

3. Add Pathをクリックして、パスを1つ作成する。

パスが、パスリストに追加される。

4. 引き続き新しいパスを追加するには、別のStateを選択して組み合わせる。



## 関連トピック

- ・「新しいダイアログイベントの作成」
- ・「ダイアログイベントを設定する」
- ・「ダイアログイベントにStateやState Groupを追加する」
- ・「フォールバック用のパスを作成する」
- ・「パスにオブジェクトをアサインする」
- ・「パスのProbabilityとWeightを設定する」
- ・「ダイアログイベントの削除」
- ・「ダイアログイベントの中のオブジェクトの再生」

### フォールバック用のパスを作成する

ゲームに存在する全ての条件や結果を再現することは困難で時間がかかり、コストもかかるので、思いつかなかつた状況や作成が間に合わなかつた状況に対処するための何らかのフォールバック機能が求められます。Wwiseでは、フォールバック用のパスを作成して対処します。フォールバック用パスには、Stateの代わりに1つ以上のStateやState Groupが含まれ、一般的に、使い回せるダイアログ（台詞）が設定されています。例えば、RTSゲームで、プレイヤーの軍隊の最終目的地など、あるオペレーションの結果が不明確な状況を考えます。この状況をカバーするために、Stateの代わりに、1つ以上のStateやState Groupを使ったパスを作成します。このパスに、オペレーションの結論が未定のサウンドオブジェクトをアサインします。軍隊が最終目的地に向かう事例では、例えば「君の軍隊は前進中だ」といったサウンドオブジェクトを設定します。

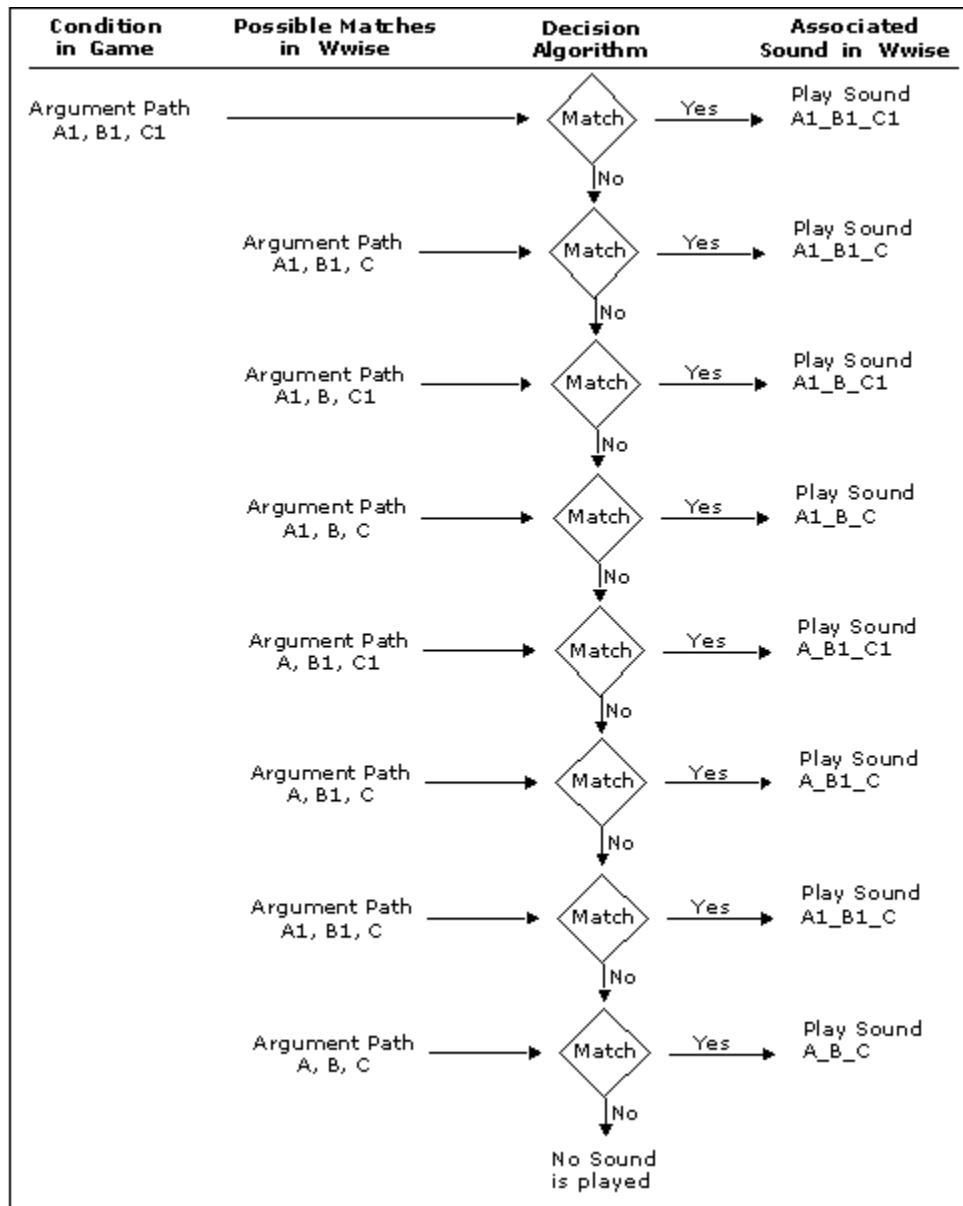
ゲームのダイアログを作成する前に、スクリプトライターがWwiseのフォールバックのアルゴリズムを理解することが重要です。例えば、以下のStateやState GroupとStateから成るダイアログイベントを考えます。

Arguments	A	B	C
Argument paths	A1	B1	C1
	A2	B2	C2
	A3	B3	C3

ゲームがこのダイアログイベントが呼び出すとアルゴリズムは、ゲームの現在の条件（例えば、A1、B1、C1）とマッチするダイアログイベントのパスを1つみつけて再生するサウンドを決めようとします。現在の条件に合うパスが複数ある場合は、Wwiseのダイアログイベントのモードによって、パスとそれに該当するサウンドがどう選ばれるかが最終的に決まります。ダイアログイベントがBest Matchモードの場合は、ゲームの現状とダイアログイベント内で作成されたパスを比較して、正確にマッチするパスを探します。もし正確にマッチするパスがない場合、またはマッチするパスにダイアログが設定されていない場合は、フォールバック用パスがあれば、アルゴリズムがその中から最も近いものを探します。

ダイアログイベントがWeightedモードの場合は、Wwiseがフォールバックパスも含めマッチするものを全て探し出します。次に、マッチした全てのパスのWeight（ウェイト付け）を確認してから、どのパスを選択するのか、どのダイアログ（台詞）を再生するのかを決めます。

Best Match（最も近い）決定のアルゴリズムが、全ての候補を確認して再生するオブジェクトを決定する様子を、下図に示します。



上図の通り、フォールバックがあるかどうかをアルゴリズムが右から左へと探しします。今回の事例では、Cから始め、次にB、そして最後にAを確認します。左端のフォールバックは、他の可能な組み合わせが全てマッチしなかった場合にのみ検討の対象となります。



### 注記

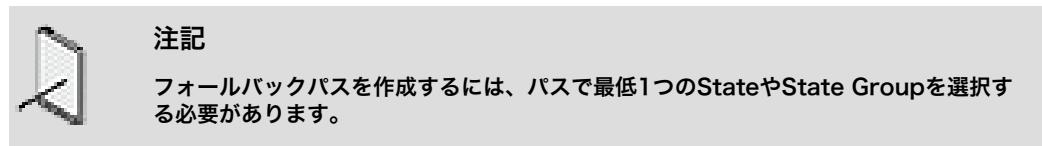
ダイアログイベント用に汎用バスを作成することもできます。この汎用バスはあらゆる状況に対応できるもので、ゲームがどのStateをトリガーしても確実にダイアログが再生されることを確保します。

### フォールバック用のバスを作成するには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。

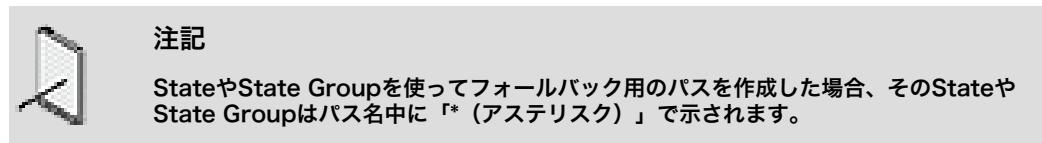
2. ダイアログイベントの全てのStateやState Groupに関して、それぞれStateやState Group、またはStateをクリックする。

Add Pathボタンが有効になる。



3. Add Pathをクリックして、パスを1つ作成する。

パスが、パスリストに追加される。



4. 引き続き新しいパスを追加するには、別のState GroupやStateを選択して組み合わせる。



## 関連トピック

- [「新しいダイアログイベントの作成」](#)
- [「ダイアログイベントを設定する」](#)
- [「ダイアログイベントにStateやState Groupを追加する」](#)
- [「Stateを使って、パスを作成する」](#)
- [「パスにオブジェクトをアサインする」](#)
- [「パスのProbabilityとWeightを設定する」](#)
- [「ダイアログイベントの削除」](#)
- [「ダイアログイベントの中のオブジェクトの再生」](#)

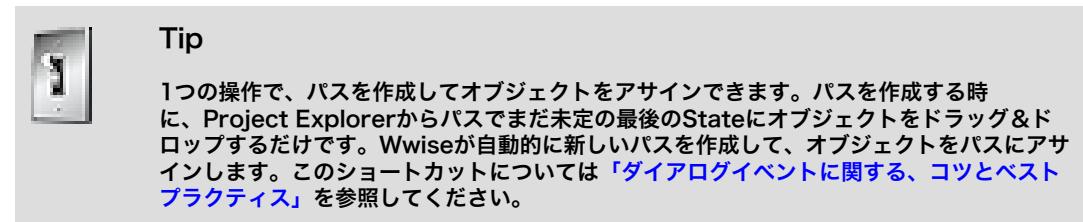
### パスにオブジェクトをアサインする

ダイアログイベントの中に複数のパスを定義できたら、次にパスにオブジェクトを1つずつ設定します。パスにオブジェクトをアサインすることで、そのパスに設定された全てのStateがゲームで満たされた状態の時に再生するオブジェクトが決まります。それぞれの条件に対する結果に多様性をもたらすために、パスにコンテナをアサインすることもできます。

オブジェクトにアサインしたパスは、いつでも変更することができ、その方法は、リストでそのオブジェクトを選択してState Groupペインで新しいパスを選択してからUpdate Pathボタンをクリックするだけです。このボタンは、State Groupsペインで新しく選択したパスに対応するエントリがリストに既にない場合に限り利用できます。

パスにアサインされた各オブジェクトのステータスが分かるように、以下の通り、オブジェクト名が色分けされます。

- ・白 - 含まれるオブジェクト。（現在のプラットフォーム内）
- ・グレー - 含まれないオブジェクト。（現在のプラットフォーム内）
- ・赤 - 現在のプロジェクトで不足しているオブジェクト。
- ・黄 - 現在のプロジェクトに、今はロードされていないオブジェクト。

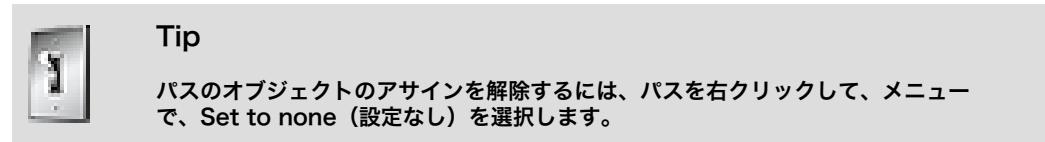


### パスにオブジェクトをアサインするには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. パスリストで、オブジェクトをアサインするパスのブラウズボタン(...)をクリックする。

Project Explorer - Browserが開く。  
3. パスにアサインするコンテナまたはオブジェクトを選択する。  
4. OKをクリックする。

オブジェクトがパスにアサインされ、Object列に表示される。

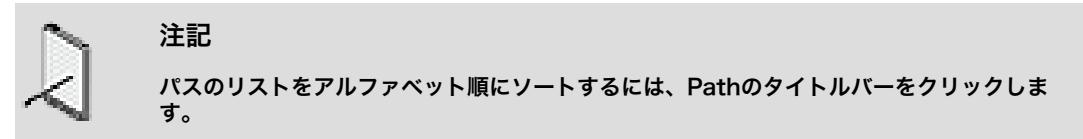


### 関連トピック

- ・「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- ・「新しいダイアログイベントの作成」
- ・「ダイアログイベントを設定する」
- ・「ダイアログイベントにStateやState Groupを追加する」
- ・「パスを使って、ゲームの条件を再現する」
- ・「Stateを使って、パスを作成する」
- ・「フォールバック用のパスを作成する」
- ・「パスのProbabilityとWeightを設定する」
- ・「パスリストにフィルタをかける」
- ・「ダイアログイベントの削除」
- ・「ダイアログイベントの中のオブジェクトの再生」

### パスリストにフィルタをかける

1つのダイアログイベント内に、多数のステートグループやステートがある場合は、パスリストが一杯になってしまうこともあります。リストが長すぎる時は、いくつかあるフィルタオプションを使用して、選択的に表示できます。



### パスリストに、フィルタをかけるには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. Path Filterリストで、以下のオプションを1つ選択する。
  - All (全て) は、作成した全てのパスを表示。
  - Current Selection (現在の選択) は、パスのうち、選択したState GroupやStateを含むものだけを表示。
3. Filterリストで、以下のオプションを1つ選択する。
  - All (全て) は、作成した全てのパスを表示。
  - Assigned object (アサインされたオブジェクト) は、あるオブジェクトに関連するパスのみを表示。
  - Missing (不足) は、プロジェクトから削除されたオブジェクトを設定してあるパスのみを表示。
  - None は、特定オブジェクトに関連しないパスのみを表示。

### 関連トピック

- 「新しいダイアログイベントの作成」
- 「ダイアログイベントを設定する」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスを使って、ゲームの条件を再現する」
- 「Stateを使って、パスを作成する」
- 「フォールバック用のパスを作成する」
- 「パスにオブジェクトをアサインする」
- 「パスのProbabilityとWeightを設定する」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

### パスのProbabilityとWeightを設定する

Wwiseには、ゲーム内のパスの選定と再生を管理するためのツールが他にもあります。ダイアログイベント内のパスごとに、Probability (発生確率) と Weight (ウェイト付け) を設定できます。マッチするパスが複数ある時に、あるパスを選択して該当するオーディオオブジェクトを再生する可能性が、これらのプロパティによって決まります。

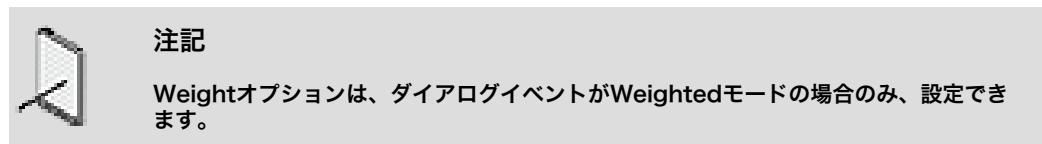
## パスのProbabilityとWeightをアサインするには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. Probabilityスライダを使い、パスのProbabilityを設定する。

Probabilityは、あるパスが選択された時にオーディオが再生される可能性を定義する。最終的な再生の確率は、選択されたパスとダイアログイベントの2つのProbabilityの組み合わせで決まる。

3. Weightスライダを使い、パスにウェイト付けを設定する。

ウェイト付けは、特定のパスを他よりも優先させるために使う。マッチするパスが複数ある場合に、ウェイト付けがより高いパスは選択される可能性が高くなる。マッチするパスの1つのWeightが「100」であれば、Weightが「100」未満のその他のマッチするパスは自動的に除外される。一方、マッチするパスの1つのWeightが「0」であれば、マッチする他の全てのパスのWeightが「0」でない限り除外される。



## 関連トピック

- ・[「新しいダイアログイベントの作成」](#)
- ・[「ダイアログイベントを設定する」](#)
- ・[「ダイアログイベントにStateやState Groupを追加する」](#)
- ・[「パスを使って、ゲームの条件を再現する」](#)
- ・[「Stateを使って、パスを作成する」](#)
- ・[「フォールバック用のパスを作成する」](#)
- ・[「パスリストにフィルタをかける」](#)
- ・[「パスにオブジェクトをアサインする」](#)
- ・[「ダイアログイベントの削除」](#)
- ・[「ダイアログイベントの中のオブジェクトの再生」](#)

## ダイアログイベントの活用

ダイアログイベントを利用してゲームのオーディオを再生させる予定があれば、プロジェクトの初期段階に開発チームと話し合い、ゲームに存在するであろう様々な条件や結果を全て定義することが重要です。

パスは、ダイアログイベントがゲームによって呼び出された時に検証される、ルールのセットなので、StateやState Groupの数や順番の変更など、パスに何らかの変更を加えると、追加のプログラミングが必要となります。最初の段階で、全てのルール、変数、条件などを定義する時間を確保することで、問題を防ぎ追加プログラミングの必要性を最小限に抑えることができます。

ダイアログイベントの管理には、以下のタスクが伴います。

- 「ダイアログイベントの名前変更」
- 「ダイアログイベントのStateやState Groupの並べ替え」
- 「ダイアログイベントからStateやState Groupを削除する」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

### ダイアログイベントの名前変更

ダイアログイベントが作成されると、Wwiseが自動的に名前を付けます。このダイアログイベント名を、内容が分かるものに変更する習慣をつけると良いでしょう。イベントの名前は必ず固有のものとし、文字、数字、アンダースコアのみを使用してください。また、名前は必ず文字またはアンダースコアで始めます。



#### 注意

ダイアログイベントをゲームに実装した後に名前を変更すると、名前変更を組み込むための追加のプログラミングが必要となるため、必要不可欠でなければ避けてください。

### ダイアログイベントの名前を変更するには:

- Project ExplorerのEventsタブで、名前を変更したいダイアログイベントをクリックする。  
イベント名がハイライト表示される。
- イベントの新しい名前を入力する。



#### Tip

Dialogue Event Editorでイベント名を変更することもできます。

### 関連トピック

- 「新しいダイアログイベントの作成」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスにオブジェクトをアサインする」
- 「ダイアログイベントのStateやState Groupの並べ替え」
- 「ダイアログイベントからStateやState Groupを削除する」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

### ダイアログイベントのStateやState Groupの並べ替え

ダイアログイベント内のStateやState Groupの順番によって、各パスの中の値の順番が決まるので、重要となります。StateやState Groupの順番を間違えた場合は、いつでも並べ替えることができます。



### 注意

ダイアログイベントをゲームに実装した後にダイアログイベントのStateやState Groupの順番を変更すると、追加のプログラミングが必要となるため、必要不可欠でなければ避けてください。やむをえずStateやState Groupの並べ替えを行う場合は、オーディオプログラマーに連絡して該当するコード変更を行ってください。

## ダイアログイベントで、ステートグループを並べ替えるには

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. Dialogue Event EditorのState Groupsペインで、StateやState Groupを新しい場所にドラッグする。StateやState Groupを配置しやすいように、赤線が表示される。

StateやState Groupの新しい順番を反映するために、既存のパスが自動的に更新される。

## 関連トピック

- ・「新しいダイアログイベントの作成」
- ・「ダイアログイベントにStateやState Groupを追加する」
- ・「パスにオブジェクトをアサインする」
- ・「ダイアログイベントの名前変更」
- ・「ダイアログイベントからStateやState Groupを削除する」
- ・「ダイアログイベントの削除」
- ・「ダイアログイベントの中のオブジェクトの再生」

## ダイアログイベントからStateやState Groupを削除する

プロジェクト開発の過程で、ダイアログイベントに対して、State Group（単数または複数）の追加や削除ができます。



### 注意

Dialogue Eventをゲームに実装した後にダイアログイベントのState Groupの追加や削除を行うと、追加のプログラミングが必要となるため、必要不可欠でなければ避けてください。やむをえず、StateやState Groupの追加や削除を行う場合は、オーディオプログラマーに連絡して該当するコード変更を行ってください。

## ダイアログイベントからステートグループを削除するには

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. 削除するStateやState Groupを右クリックして、ショートカットメニューで、Remove Columnを選択する。

ステートグループとそのステートが、ダイアログイベントから削除されます。

## 関連トピック

- ・「新しいダイアログイベントの作成」

- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスにオブジェクトをアサインする」
- 「ダイアログイベントの名前変更」
- 「ダイアログイベントのStateやState Groupの並べ替え」
- 「ダイアログイベントの削除」
- 「ダイアログイベントの中のオブジェクトの再生」

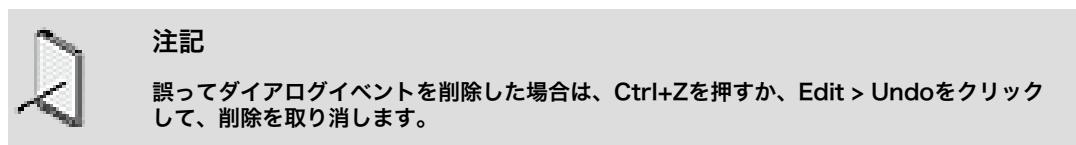
### ダイアログイベントの削除

不要となったダイアログイベントは、削除できます。ダイアログイベントを削除する前に、他のチームメンバーがそれをプロジェクトの別の部分で使用していないか、または既にSoundBankの1つに含まれていないか、確認すると良いでしょう。既にサウンドバンクに入っているダイアログイベントを自分または他人が削除してしまうと、無効なダイアログイベントが発生します。Wwiseは、削除されたダイアログイベントをSoundBankから自動的に削除しないので、マニュアル操作で削除する必要があります。このようなダイアログイベントをWwiseで確認し管理しやすいように、SoundBank Managerでリスト表示されます。無効イベントの削除については「[SoundBank内のエレメントを検索する](#)」を参照してください。また、全ての無効ダイアログイベントのリストをプロジェクトのインテグレティレポートで確認することもできます。インテグレティレポートの使用については「[プロジェクトのトラブルシューティング](#)」を参照してください。

#### イベントを削除するには:

- Project ExplorerのEventsタブで、削除するダイアログイベントを右クリックして、ショートカットメニューで、Delete Selectionを選択する。

ダイアログイベントが削除される。



### 関連トピック

- 「新しいダイアログイベントの作成」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスにオブジェクトをアサインする」
- 「ダイアログイベントの名前変更」
- 「ダイアログイベントのStateやState Groupの並べ替え」
- 「ダイアログイベントからStateやState Groupを削除する」
- 「ダイアログイベントの中のオブジェクトの再生」

### ダイアログイベントの中のオブジェクトの再生

制作過程において、いつでも、ダイアログイベント内のパスにアサインされたオブジェクトを試聴できます。

## ダイアログイベントを再生するには:

1. Dialogue Event Editorに、ダイアログイベントを1つロードする。
2. パスを1つ選択する。

パスにアサインされたサウンドオブジェクトが、自動的にTransport Controlにロードされる。

3. Transport ControlのPlayアイコンをクリックする。

サウンドオブジェクトが再生される。

## 関連トピック

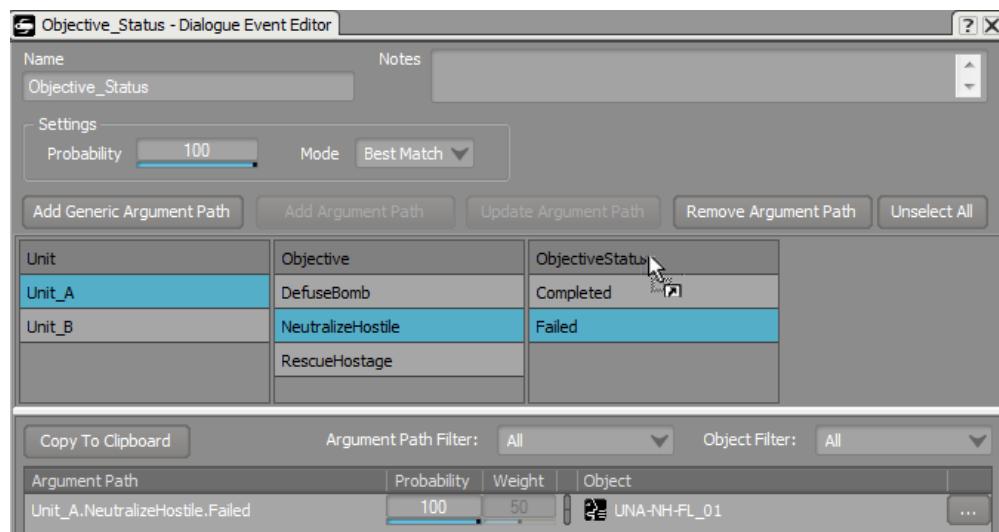
- 「新しいダイアログイベントの作成」
- 「ダイアログイベントにStateやState Groupを追加する」
- 「パスにオブジェクトをアサインする」
- 「ダイアログイベントの名前変更」
- 「ダイアログイベントのStateやState Groupの並べ替え」
- 「ダイアログイベントからStateやState Groupを削除する」
- 「ダイアログイベントの削除」

## ダイアログイベントに関する、コツとベストプラクティス

ダイアログイベントを作成する前に以下のセクションに目を通して、プロジェクトのダイアログイベントを効率的に作成し管理するためのコツやベストプラクティスを参照してください。

### パス作成のショートカット

多数のパスを作成する場合は、ショートカットを使って作業できます。このショートカットで、パスを完成させると同時に対象のサウンドオブジェクトを決定できます。パスの内容を完成させる前に、設定するオブジェクトをProject Explorerから未定の最後のStateやState GroupまたはStateにドロップします。

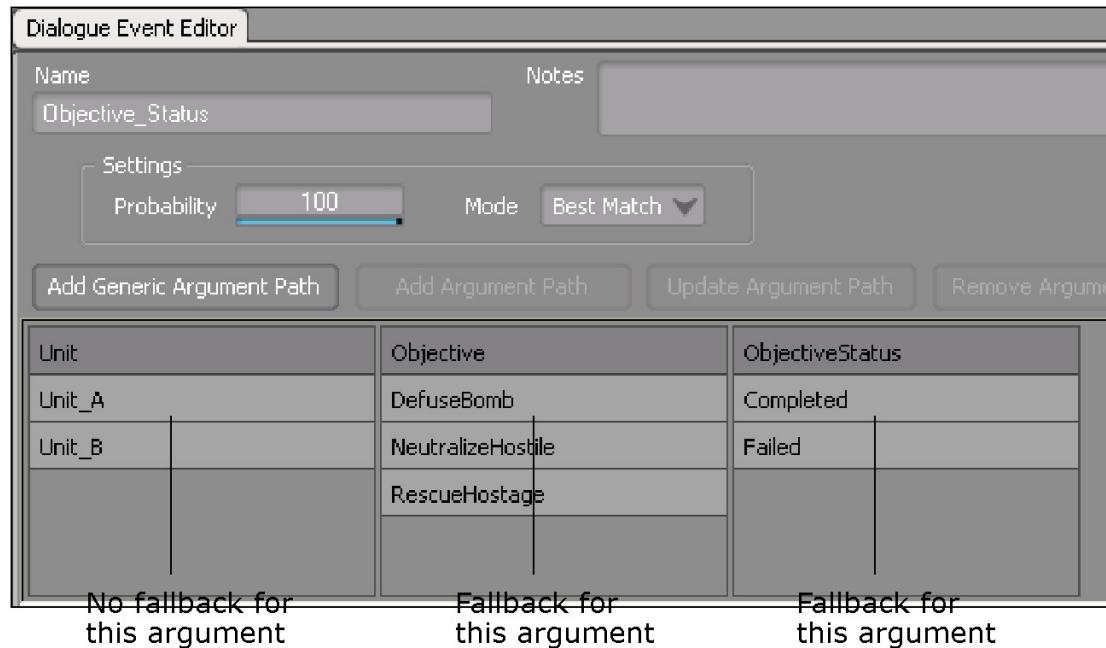


Wwiseで自動的に新しいパスが作成されて、このオブジェクトがアサインされます。既にパスがある場合は、アサインされている現在のオブジェクトが新しいものに置換されます。

### パスの構築のための戦略

- フォールバック用パスを最初に作成すること - 基本的に、ダイアログイベントのフォールバック用パスを全て最初に作成して、あらゆる条件や結果に確実に対応できるようにします。一般的なフォールバックパスを作成した後に、より具体的なパスの作成を開始します。こうすれば、全ての条件を再現する時間がない場合や、開発サイクルの最終段階で新しい条件が追加された場合でも、対応できます。
- 最良のアルゴリズム効率のためにStateやState Groupの順番を工夫すること - フォールバックを使う可能性の低いStateやState Groupは、順番を前にします。アルゴリズムがパスを解くために行うバックトラッキングが減り、ダイナミックダイアログシステム全体の効率が向上します。

以下の例では、StateやState Group 「Unit」 にフォールバックが設定されないため、順番が最初です。このStateやState Groupにフォールバックを用意することは無意味であり、その理由は、Unit AもUnit Bもゲームの具体的なキャラクターに関連するため、一般的な「人」を作成することが不可能なためです。一方、「Objective (目標)」と、「ObjectiveStatus (目標ステータス)」のStateやState Groupに関してはフォールバックを作成する意味があるので、順番がStateやState Group 「Unit」 のあとになります。



---

## 第17章 Stateの活用

概要 .....	389
Stateの活用 .....	390
同じStateやState Groupの、State間のトランジションを設定する .....	393
オブジェクトやバスにStateをアサインする .....	394
Stateに関する、コツとベストプラクティス .....	399

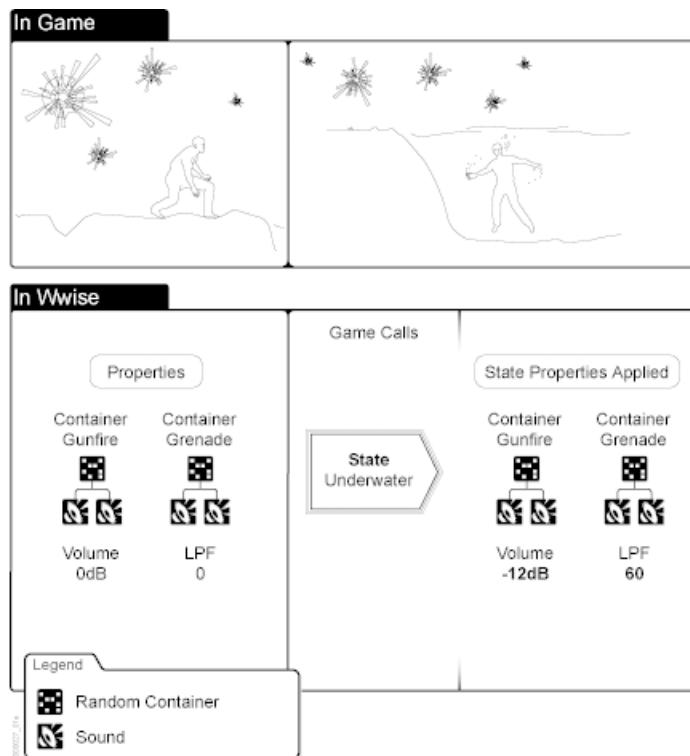
## 概要

ゲームデザイナーは、可能な限り魅惑的なオーディオを、最小限のメモリ、CPU、アセット、ディスクスペースを使って作成しようと、常に努力を重ねています。Stateには、この挑戦に対する効率的でクリエイティブな考え方が導入されています。Stateを使うと、1つのサウンドに対して数種類のミキサースナップショットを作成できるので、ゲーム側の変更に合わせたプロパティ変更をグローバル（全体的）に適用する柔軟性が生まれ、サウンドやミュージックアセットを最適化できます。ゲーム中のアクションに合わせて、サウンドオブジェクトやミュージックオブジェクトのプロパティを変化させることで、アセットを新たに追加することなく、クリエイティブに対応できます。Stateをいつ、どこで活用すれば、最も効率的でクリエイティブな結果が得られるかを、プロジェクトの初期段階で決めてください。

### Stateの活用例

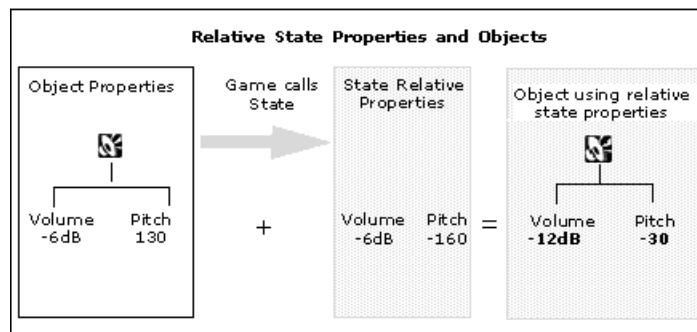
例えば、キャラクターが水中に潜る時のサウンド処理を考えます。この場合、State機能を利用して、再生中の複数サウンドのボリュームやLPFを変更できます。キャラクターが水中にいる時の、銃声音や手榴弾の爆発音を表現するサウンドの変化を、オブジェクトのプロパティ変更によって行います。

下図は、「Underwater (水中)」Stateをゲームが呼び出した時に、「Gunfire (銃声音)」と、「Grenade (手榴弾)」の2つのサウンドオブジェクトのボリュームとLPFのプロパティが変更される様子を示しています。



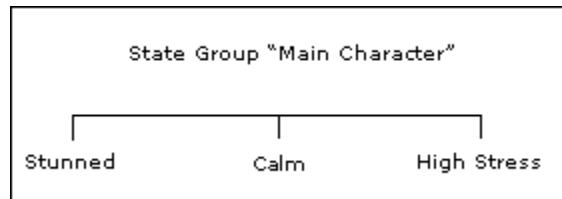
### Stateで設定したプロパティが、どのようにオブジェクトに適用されるのか

Stateプロパティは、常に相対的です。ステートを適用すると、そのオブジェクトのプロパティが、加減されます。



## Stateの活用

StateをWwiseのオブジェクトに適用するには、まずStateをStateやState Groupに入れる必要があります。StateをStateやState Groupに入れて論理的に整理して、Stateを管理しやすくします。例えば、ゲームのメインキャラクターに関する複数の状態、つまりStateを、一緒にまとめると便利です。次にMain CharacterというStateやState Groupを作成して、メインキャラクターに関連する様々なオブジェクトのプロパティに適用するStateをここに追加します。ゲーム中に、メインキャラクターが経験するであろうStateとして、「Stunned（呆然）」「Calm（落ち着き）」「High Stress（ストレス大）」が事前に分かっているとします。これらを同じグループにまとめてからStateごとにプロパティ変更を定義すると、分かりやすいでしょう。



Stateの構造が確立できたら、次にProperty Editorで、各StateやState Groupにオブジェクトを登録して、必要に応じてStateプロパティをカスタマイズします。

インターフェース上で分かりやすいように、State GroupやStateは、以下のアイコンで表されます。

アイコン	内容
	State Group
	State

Stateを使った作業には、以下のタスクが伴います。

- 「State Groupの作成」
- 「Stateの作成」

- ・「同じStateやState Groupの、State間のトランジションを設定する」
- ・「StateやState Groupの削除」

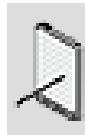
### State Groupの作成

StateやState Groupは以下の2つの場所のどちらでも作成できます。

- Project Explorer
- Property EditorのStatesタブ

Project Explorerで、プロジェクトの新規StateやState Groupを作成するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. Statesセクションで、以下のいずれかを行う。
  - Work Unit（または仮想フォルダ）を選択して、Project Explorerのツールバーの、State Groupアイコンをクリックする。
  - Work UnitまたはVirtual Folderを右クリックして、ショートカットメニューで、New Child > State Groupをクリックする。
- 新しいStateやState Groupが、StateやState Groupのリストに追加される。
3. デフォルトのStateやState Group名を、内容が分かる名前に変更する。



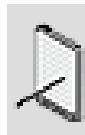
#### 注記

StateやState Groupの名前は、必ず固有のものとし、文字、数字、アンダースコア以外は使用できません。また、名前は必ず文字またはアンダースコアで始めます。

4. 必要に応じて、StateやState Groupを追加する。

Property EditorのStatesタブで、プロジェクトの新規StateやState Groupを作成するには:

1. Property EditorのStatesタブで、Add >>をクリックする。  
Stateのセレクタメニューが開く。
2. New...を選択する。
3. 新規StateやState Groupを保存するWork Unitを選択する。
4. Nameフィールドで、デフォルトのStateやState Group名を、内容が分かる名前に変更する。



#### 注記

StateやState Groupの名前は、必ず固有のものとし、文字、数字、アンダースコア以外は使用できません。また、名前は必ず文字またはアンダースコアで始めます。

5. OKをクリックして、新規StateやState Groupを作成し、これに登録する。

### 関連トピック

- ・「Stateの作成」

- 「同じStateやState Groupの、State間のトランジションを設定する」
- 「StateやState Groupの削除」

## Stateの作成

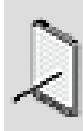
Stateは、ゲームに呼び出されると、ゲームの条件に応じてオブジェクトにプロパティ変更を適用します。あるStateから別のStateへのトランジションについては、State Groupエディタで設定します。

### 新規Stateを作成するには

- Project Explorerで、Game Syncsタブに切り替える。
- Statesセクションで、以下のいずれかを行う。
  - StateやState Groupを1つ選択して、Project Explorerのツールバーの、Stateアイコンをクリックする。
  - StateやState Groupを1つ右クリックして、ショートカットメニューで、New Child > Stateを選択する。

新しいStateが、StateやState Groupに追加されます。

- デフォルトのState名を、内容が分かる名前に変更する。



#### 注記

同じStateやState Groupの中にあるStateの名前は、StateやState Group内で固有のものとし、文字、数字、アンダースコア以外は使用できません。また、名前は必ず文字またはアンダースコアで始めます。

- 必要に応じて、Stateを追加する。

## 関連トピック

- 「State Groupの作成」
- 「同じStateやState Groupの、State間のトランジションを設定する」
- 「StateやState Groupの削除」

## StateやState Groupの削除

StateやState Groupがプロジェクトで不要となり、削除する必要が出てくることもあります。StateやState Groupを削除すると、そのStateやState Groupの中のStateも、全て削除されるので、注意してください。削除されたStateを使用していたオブジェクトやプリセットがある場合は、以後はそのStateを使えなくなります。



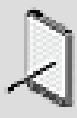
#### Tip

State Groupを削除する前に、それを使用しているオブジェクトを確認するには、State Groupのコンテキストメニューにある、Find All References（全レファレンスを表示）コマンドを使います。

## StateやState Groupを削除するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. States セクションで、削除するStateまたはStateやState Groupを右クリックして、Delete Selectionを選択する。

選択したStateまたはStateやState Groupが、削除される。



### 注記

StateやState Groupを、誤って削除した場合は、Ctrl+Z を押すか、Edit > Undoをクリックして、削除を取り消します。

## 関連トピック

- ・「[State Groupの作成](#)」
- ・「[Stateの作成](#)」

## 同じStateやState Groupの、State間のトランジションを設定する

同じStateやState Groupの中で、State間のトランジションをスムーズに行うために、Stateの切り替え時間を設定できます。トランジション時間の設定には、以下の2つのオプションがあります。

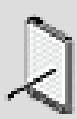
- ・「[StateやState Groupの全Stateの、トランジション設定](#)」は、そのStateやState Groupの中のどのStateの場合も、StateとStateの間のトランジション時間を、同じ時間に設定する場合。
- ・「[StateやState Groupの、State間のトランジションのカスタム設定](#)」は、そのStateやState Groupの、StateとStateの間のトランジション時間を、個別に設定する場合。

### StateやState Groupの全Stateの、トランジション設定

State間で使うトランジション時間を、1つ設定して、あるStateやState Groupの中にある、全てのStateに適用できます。

### 選択したStateやState Groupの、State間のトランジション時間を設定するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. Statesセクションで、StateやState Groupをダブルクリックして、Property Editorにロードする。
3. Default transition timeフィールドで、選択したStateやState Group内の全てのStateに適用する、トランジション時間を設定する。



### 注記

デフォルトのトランジション時間は、カスタム設定のトランジション時間がない場合に使われます。

## 関連トピック

- 「Stateの作成」
- 「StateやState Groupの削除」
- 「StateやState Groupの、State間のトランジションのカスタム設定」

### StateやState Groupの、State間のトランジションのカスタム設定

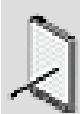
ゲーム内の状況の変化に応じて、よりリアルにするために、StateやState Groupの中の全てのStateに、同じトランジション時間を適用しないことも考えられます。表Custom Transition Timeで、2つのStateを自由に選び、専用のトランジション時間を設定できます。また、カスタム設定したトランジション時間を、両方向で同じ時間とするのかどうかも、指定できます。例えば、「Rain」のStateから、「Snow」のStateに変化する時のトランジション時間を、逆方向でも採用できます。

### 選択したStateやState Groupの、State間のトランジション時間をカスタム設定するには:

- Project Explorerで、Game Syncsタブに切り替える。
- Statesセクションで、StateやState Groupをダブルクリックして、Property Editorにロードする。
- Insertをクリックする。

表Transition Timeに、新しい行が追加される。

- From列で、ソース（元）のStateを選択する。
- Time列で、2つのStateの間に適用する、トランジション時間を設定する。
- To列で、デスティネーション（次）のStateを選択する。
- 逆方向のトランジションにも、同じトランジション時間を設定するには、双方を示すチェックボックスに、チェックを入れる。



#### 注記

カスタム設定の行を追加するには、Insertをクリックし、設定を削除するには、Removeをクリックします。

## 関連トピック

- 「Stateの作成」
- 「StateやState Groupの削除」
- 「StateやState Groupの全Stateの、トランジション設定」

## オブジェクトやバスにStateをアサインする

プロジェクトで使うStateを作成して設定できれば、State GroupやStateをオブジェクトやバスにアサインして、サウンド、ミュージック、モーションをゲームの状況に合わせることができます。また、各種プロパティを個別に調整すれば、サウンド、ミュージック、モーションをさらに差別化できます。

オブジェクトをStateやState Group（単数または複数）に登録すると、そのStateやState Group内にある全てのStateが、自動的にオブジェクトにアサインされます。1つのオブジェクトを、複数のStateやState Groupに登録することができます。アクターミキサーまた、StateやState Groupは、Master-Mixer Hierarchy、Actor-Mixer Hierarchy、Interactive Music Hierarchyの各レベルにおいてアサインできます。

## オブジェクトにState Groupをアサインするには

1. Property Editorに、オブジェクトを1つロードする。
2. Statesタブに切り替える。
3. **Group**セレクタボタン (`<el>></el>`) をクリックして、オブジェクトにアサインするStateやState Groupを選択する。

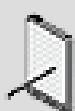
関連するStateがリストに表示されます。これでオブジェクトが選択したStateやState GroupのStateに登録されます。

## 関連トピック

- 「オブジェクトのStateプロパティをカスタマイズする」
- 「ミュージックオブジェクト内の、State切り替えポイントを設定する」

## ミュージックオブジェクト内の、State切り替えポイントを設定する

インタラクティブミュージックの場合は、State切り替えのタイミングを、再生中のミュージックのテンポに合わせることで、スムーズなトランジションを確保できます。これは、そのミュージックオブジェクトのプロパティレベル、またはバスのレベルで設定します。ミュージックオブジェクトやバスの、Property Editorで、登録中のステートグループごとに、ステート変更の最適ポイントを設定します。ポイントとして、例えば、Immediate、Next Cue、Next Beatなどを指定します。ただし、Stateの切り替えに複数のミュージックオブジェクトが関与して、それぞれのState切り替えの設定が異なる場合は、セグメントで設定されている次の機会に、Stateが切り替わります。



### 注記

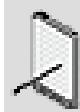
オーディオバスに、アクターミキサー階層のサウンドオブジェクト以外が入っていない場合は、これらの設定が無視され、変更がすぐに適用されます。ただし、ミュージックオブジェクトとサウンドオブジェクトの両方が、Audio Busにルーティングされる場合は、Stateの切り替えはミュージックオブジェクトのState切り替え設定に基づきます。

## ミュージックオブジェクトやバスの、State切り替えポイントを設定するには:

1. Property Editorに、ミュージックオブジェクトまたはバスを1つロードする。
2. Statesタブに切り替える。
3. 登録されているStateやState Groupの1つに対して、以下のオプションのいずれかを、Change occurs at列から選択する。
  - **Immediate:** 直ちに切り替え。トラックのルックアヘッドタイムを設定している場合は、この時間が経過してから、Stateが切り替わる。

- **Next Grid** - 次のグリッドで切り替え。グリッドは、ミュージックオブジェクトをバーチャルに分割する、任意の周波数。
- **Next Bar**: 次のバーで切り替え。
- **Next Beat**: 次のビートで切り替え。
- **Next Cue**: 次のキューで切り替え。キューとは、エントリキュー、エグジットキュー、またはカスタムキューである。
- **Next Custom Cue**: 次のカスタムキューで切り替え。
- **Entry Cue**: エントリキューで切り替え。
- **Exit Cue**: エグジットキューで切り替え。

現在のStateやState GroupのStateの切り替えは全て、指定したポイントで実行される。



#### 注記

1つのStateやState Groupに複数のミュージックオブジェクトを登録した場合は、Stateの切り替えが、全てのオブジェクトに対して、同時に1回、適用されます。Stateの切り替えは、全てのオブジェクトを対象に、最初のトランジションポイントで起きます。

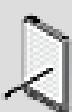
## 関連トピック

- 「オブジェクトのStateプロパティをカスタマイズする」

### オブジェクトのStateプロパティをカスタマイズする

オブジェクトに対してStateやState Groupをアサインした後に、そのオブジェクト用に、各Stateのプロパティをカスタマイズできます。Stateごとに以下のプロパティを変更できます。

- Pitch
- LPF
- HPF
- Volume
- Make-up Gain



#### 注記

オブジェクトやプラットフォームによって、使われないプロパティもあります。例えばPitchプロパティはミュージックオブジェクトに影響せず、Make-up GainはMaster-Mixerのオブジェクトに影響しません。

### オブジェクトのStateプロパティをカスタム設定するには:

1. Property Editorに、オブジェクトを1つロードする。
2. Statesタブに切り替える。
3. 以下のプロパティ値を設定する。
  - Pitch設定で、オブジェクトの再生スピードを、減速または加速する。

- ・>Low-Pass Filter設定で、高周波数を減衰させるリカーシブ（再帰）フィルタを適用する。
- ・High-Pass Filter設定で、低周波数を減衰させるリカーシブ（再帰）フィルタを適用する。
- ・Volume設定で、アウトプットレベル、つまりアウトプット振幅を調整する。

## 関連トピック

- ・「オブジェクトやバスにStateをアサインする」
- ・「State間で、State設定値をコピーする」
- ・「Example: 一時的な難聴のエフェクトを作成する」

### State間で、State設定値をコピーする

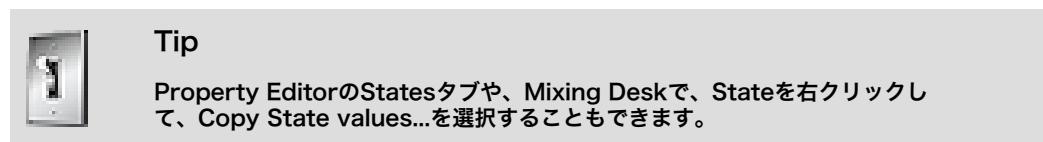
多数のStateで、似たようなプロパティ設定を使う場合は、1度設定を行った後に、そのState設定値を、同じStateやState Group内の他のStateにコピーできます。

Stateの設定値を、同じStateやState Group内の既存Stateや、新規Stateにコピーできます。さらに、ソースのState値によって、どのオブジェクトが影響を受けるのかを指定することもできます。

#### State値をコピーするには:

1. 以下のいずれかのビューで、Copy State Values...ボタンをクリックする。
  - State Property Editor
  - Mixing Desk

Copy States Valuesダイアログボックスが開く。



2. State Groupのセレクタボタン (>>) をクリックして、コピーするカスタムプロパティのStateを含む、StateやState Groupを選択する。
3. Fromのセレクタボタン (>>) をクリックして、コピーするカスタムプロパティのStateを選択する。

StateやState Groupに登録されている全てのオブジェクトが、Affected objectsリストに表示される。



4. Toのセレクタボタン (>>) をクリックして、以下のいずれかを行う。
  - カスタムプロパティを、新しいStateにコピーするには、Newを選択して、新しいState名を設定して、OKをクリックする。

- カスタムプロパティを、既存のStateにコピーするには、リストからStateを選択する。

Wwiseが、Affected objectsリストの各オブジェクトの処理を判断する。

- Use列のチェックボックスで、新しいカスタムState設定を使用するオブジェクトを選択する。
- OKをクリックして、選択したオブジェクトにカスタムState設定を適用する。

## 関連トピック

- 「オブジェクトやバスにStateをアサインする」
- 「オブジェクトのStateプロパティをカスタマイズする」
- 「Example: 一時的な難聴のエフェクトを作成する」

### Example: 一時的な難聴のエフェクトを作成する

制作中のファーストパーソンシューティングゲームで、プレイヤーの近くでフラッシュバングレネードが爆発して、一時的に耳が聞こえなくなる場面があるとします。このような種類のエフェクトを表現するには、全てのサウンドを一時的に変化させる必要があります。

#### 手順

「一時的な難聴」エフェクトを実現するのに最適なのが、Stateの利用です。例えば、"GrenadeFX"というState Groupを作成して、"Stunned"と"Normal"の2つのStateを設定します。"Stunned"StateでPitch、Volume、Low-pass Filter、High-pass Filterの設定値を調整して、「一時的な難聴」エフェクトを作成します。このState Groupをコントロールバス（単数または複数）に対して使えば、SFX、アンビエンス、ミュージックなどを変化させることができます。

さらにリアルにするために、State Group "GrenadeFX"を使うコントロールバスにRTPCも適用して、接近感のエフェクトの強弱を調節します。ステートのプロパティ（ボリューム、ピッチ、HPF、LPF）とRTPCを合わせることで、グレネードとメインキャラクター（マイク）との間隔に応じて、値を増減できます。このエフェクトを実現するには、以下を行います：

- Game Syncsタブで、新しいGame Parameterを、例えば"Grenade Proximity"などと題して作成して、ゲームの内容に合った距離のMin/Max値を設定する。
- State Group "GrenadeFX"を使うコントロールバスの\$1タブを開き、Game Parameter "Grenade Proximity"に1つ以上のプロパティをアサインする。
- グレネード衝撃がキャラクターに近いほど、このエフェクトが激しくなるように、値のマッピングを行う。

StateやRTPC値の微調整は、Soundcasterでインゲームシミュレーションを再現しながら行います。例えば、ミュージック、アンビエントサウンド、ボイス、そしてSFXを再生しながら、StateやState Groupを"Normal"から"Stunned"に切り替えてみます。なお、Stateが切り替わる際のTransition Timeを調整することも忘れないでください。

リアリティを高めるもう1つの方法として、キャラクターのStateがStunnedの時のリバーブテイルエフェクトを、こもらせることもできます。リバーブのプロパティのLPF Cutoff FrequencyにRTPCを使い、Stateの切り替わりに同期させると、リバーブを各種オブジェクトに設定したLPF量に合わせて変化させることができます。

### 関連トピック

- ・「オブジェクトやバスにStateをアサインする」
- ・「オブジェクトのStateプロパティをカスタマイズする」

## Stateに関する、コツとベストプラクティス

Stateを使う時に、以下のセクションに目を通して、開発プロセス全体を通して効率的にStateを使って作業するための、コツやベストプラクティスを参照してください。

### StateやState Groupの削除

Stateをゲームに実装するには、2つの方式があります。Stateアクションを設定してあるイベントを呼び出して実装する方式と、StateやState GroupとState自身を呼び出して実装する方式です。前者の場合は、StateやState Groupを削除すると、呼び出されてもStateが存在しないので、Wwise上で問題が発生します。後者の場合は、StateやState Groupを削除しようとするサウンドデザイナーが、変更内容をオーディオプログラマーに連絡する必要があります。

あるStateが、プロジェクトのどの部分で使用されているのかを確認するには、Find all Referencesコマンドを使用します。

### Stateのプロパティと、CPUやメモリとの関係

ピッチなどWwiseの一部の相対プロパティは、プラットフォームによってパフォーマンスに影響することがあります。Wwiseでは、ピッチをサンプルレートで管理します。よって、オブジェクトにピッチを適用するとファイルのリサンプル処理が発生し、CPU負荷が増加します。

---

## 第18章 Switchの使い方

概要 .....	401
Switchの使い方 .....	402
Switchに、Game Parameter値をマッピングする .....	405
Switchに関する、コツとベストプラクティス .....	405

## 概要

Wwiseのオブジェクトを、階層を使って整理できるほか、Switchを利用して、サウンドオブジェクト、ミュージックオブジェクト、モーションオブジェクトを効率的に整理することも可能です。Switchは、ゲームの特定エレメントに対して存在する全ての選択肢を表し、それぞれの選択肢に対応するオブジェクトを管理するために使います。ゲームのエレメントに対して設定できる選択肢の数は、メインキャラクターが使うウェポンの種類や、様々な天候などに合わせて、決められます。選択候補となるオブジェクトを特定のSwitchにアサインして、ゲーム中に有効化されたSwitchに使い、これらのオブジェクトを再生します。

ランタイムの多様な選択肢を、Switch機能を使って簡単に管理するのに適したゲームの状況やエレメントは多数あり、そのいくつかを以下に紹介します。

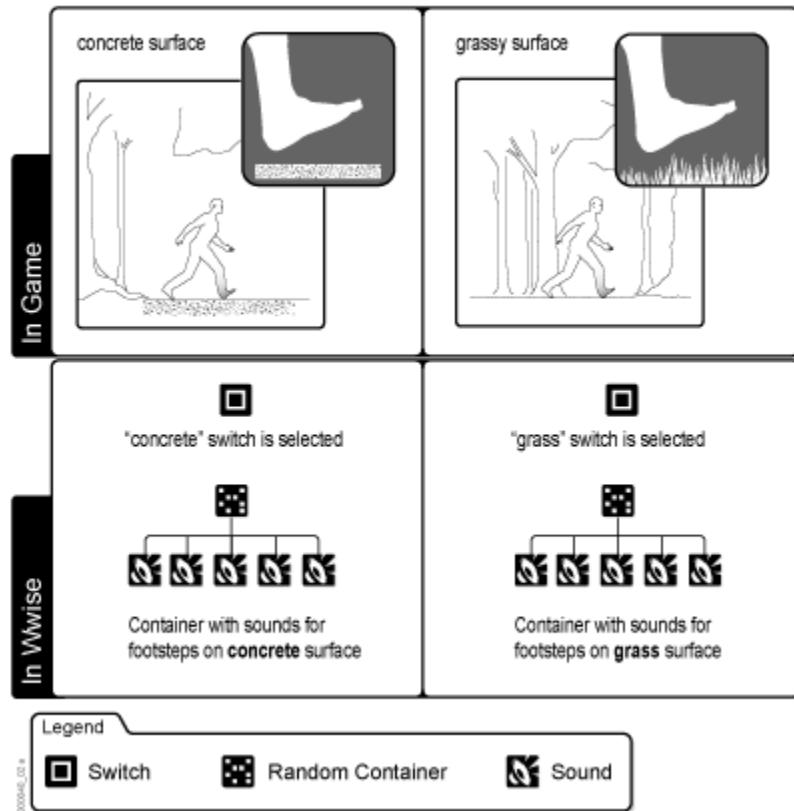
- ・室内状況、地面の種類、屋内外の違いなど、ゲーム設定: 例えば、床板、草、砂利など地面の種類に合わせて、Switchを作成。
- ・ゲームのキャラクター: 例えば、男性キャラクター、または女性キャラクターの、ダイアログ用のSwitchを作成。
- ・天候条件: 例えば、暴風雨、吹雪、しとしと雨、晴天などの、Switchを作成。
- ・悪の世界や、妖精の世界など、ゲーム空間: 例えば、それぞれのワールドに関連する、様々なサウンドのSwitchを作成。
- ・ウェポン: 例えば、ゲームの各種銃器や、レーザービーム、剣など、様々な発火パターンに合わせたSwitchを作成。

どの事例の場合も、まずSwitchを作成してから、それに対応するオブジェクトをアサインします。Switchにアサインしたオブジェクトは、まとめてSwitch Containerに入れます。イベントまたはGame Parameter値によって変化が通知されると、Switch ContainerがSwitchを検証し、正しいオブジェクトが再生されます。

## Switchの活用例

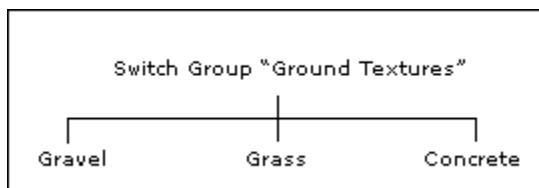
例えば、ファーストパーソンシューティングゲームで、メインキャラクターが次々と変わる環境を歩いて通る状況を考えます。環境に応じて地面も変化するので、コンクリート、芝生、土などの地面素材によって足音も変えます。この場合、それぞれの地面素材に対してSwitchを作成し、各Switchに様々な足音サウンドをアサインします。メインキャラクターがコンクリート面を歩くと「Concrete（コンクリート）」Switchが有効になり、該当するサウンドが再生されます。続いてコンクリート面から芝生面に移動した場合は、「Grass（芝生）」Switchが有効になり、該当するサウンドが再生されます。

下図は、有効になったSwitchによって、再生する足音サウンドが決まる様子を示しています。



## Switchの使い方

WwiseでSwitchを使うには、まずSwitchをSwitch Groupに入れる必要があります。Switch Groupの中に複数のSwitchをまとめて入れることで、ゲームで使えるサウンド、ミュージック、モーションなどの候補オブジェクトを、効率的に管理できます。例えば、キャラクターの様々な足音サウンドを管理するために、「Ground Textures（地面素材）」というSwitch Groupを作ります。次に、ゲーム中に出てくる全ての地面の種類に対して、それぞれのSwitchをWwiseで作成します。既に分かっているゲームの地面の種類に合わせて、「Gravel（砂利）」「Grass（草）」「Concrete（コンクリート）」などのSwitchを追加します。



Switch GroupとSwitchが作成できたら、次にSwitch Containerを作成して、Switchにオブジェクトをアサインします。SwitchやSwitch Containerを使った作業については「[Switch Containerの中身と、再生方法の設定](#)」を参照してください。

インターフェース上で分かりやすいように、SwitchやSwitch Groupは、以下のアイコンで表されます。

アイコン	内容
	Switch Group
	Switch

Switchを使った作業には、以下のタスクが伴います。

- 「Switch Groupの作成」
- 「Switchの作成」
- 「Switch GroupやSwitchを削除する」
- 「Switchに、Game Parameter値をマッピングする」

### Switch Groupの作成

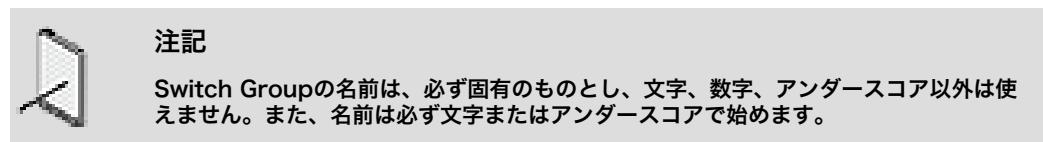
論理的な法則に従ってSwitchをSwitch Groupに入れて整理すると、作業がしやすくなります。Switch Groupは、Project Explorerの、Game Syncsタブで、必要な数だけ作成します。

#### プロジェクトの新規Switch Groupを作成するには:

- Project Explorerで、Game Syncsタブに切り替える。
- Switchesセクションで、以下のいずれかを行う。
  - 仮想フォルダ（またはWork Unit）を選択して、Project Explorerのツールバーの、**Switch Group**アイコンをクリックする。
  - 仮想フォルダ（またはWork Unit）を右クリックして、ショートカットメニューで、**New Child > Switch Group**をクリックする。

新規Switch Groupが、Switch Groupのリストに追加される。

- デフォルトのSwitch Group名を、内容が分かる名前に変更する。



- 必要に応じて、Switch Groupを追加する。

### 関連トピック

- 「Switch GroupやSwitchを削除する」
- 「Switchの作成」

### Switchの作成

選択肢が複数あるゲームエレメント、例えば地面などは、それぞれにWwiseのSwitchが必要です。Switchは、Project Explorerの、Game Syncsタブで作成します。

#### 新規Switchを作成するには:

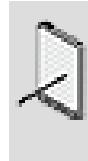
- Project Explorerで、Game Syncsタブに切り替える。

2. Switch Groupsリストで、以下のいずれかを行う。

- Switch Groupを選択して、Project Explorerのツールバーの、Switchアイコンをクリックする。
- Switch Groupを右クリックして、ショートカットメニューで、New Child > Switchを選択する。

新規Switchが、Switch Groupに追加される。

3. デフォルトのSwitch名を、内容が分かる名前に変更する。



### 注記

同じSwitch Groupの中にあるStateの名前は、必ず固有のものとし、文字、数字、アンダースコア以外は使えません。また、名前は必ず文字またはアンダースコアで始めます。

4. 必要に応じて、Switchを追加する。

## 関連トピック

- [「Switch Groupの作成」](#)
- [「Switch GroupやSwitchを削除する」](#)

## Switch GroupやSwitchを削除する

SwitchやSwitch Groupが不要となり、削除する必要が出てくることもあります。Switch Groupを削除すると、そのSwitch Groupの中のSwitchも、全て削除されるので、注意してください。



### 注記

Switchを削除すると、そのSwitchを使用していたプリセットやオブジェクトで、以後はそれを使えなくなります。

## Switchや、Switch Groupを削除するには

1. Project Explorerで、Game Syncsタブに切り替える。
2. Switch Groupリストで、削除するSwitch GroupまたはSwitchを右クリックして、Delete Selectionを選択する。

選択したSwitchまたはSwitch Groupが削除される。



### Tip

SwitchやSwitch Groupを誤って削除した場合は、Ctrl+Z を押すかEdit > Undoをクリックして削除を取り消します。

## 関連トピック

- [「Switchの作成」](#)
- [「Switch Groupの作成」](#)

## Switchに、Game Parameter値をマッピングする

Wwiseでは、Game Parameterを利用したSwitchの切り替えも可能です。Switch GroupやGame Parameterを作成した後に、Game Parameter値をSwitchにマッピングします。例えば、車両の衝突用に設定したSwitchを、RTPCを使って切り替えるのであれば、衝突に関連するサウンドやを、衝撃の強さによって変えられます。衝撃の強度を表す数値を使ってSwitchの切り替えをトリガーすることで、衝突が起きた時に、正しいオブジェクトが再生されることを、簡単に設定できます。



### Game Parameter値を、Switchにマッピングするには:

1. Project Explorerの、Game Syncsタブで、Game Parameter値で変化させるSwitch Groupを、ダブルクリックする。  
Switch Group Property Editorに、Switch Groupがロードされる。
2. チェックボックス**Use Game Parameter** を選択する。  
Graphビューが表示され、そのSwitch GroupのSwitchのリストが、Y軸に沿って表示される。
3. Game Parameterリストで、Switchを変化させるために使うGame Parameterを選択する。  
選択されたGame Parameter値の範囲が、X軸に沿って表示される。
4. Graphビューで、Game Parameterカーブに沿ってダブルクリックしてポイントを追加して、そのポイントを該当Switchまでドラッグする。  
Switchの切り替えが、指定したGame Parameter値にマッピングされる。
5. 必要に応じて、カーブ沿いをダブルクリックしてポイントを追加して、Switchにマッピングする。

### 関連トピック

- 「[Switch Groupの作成](#)」
- 「[Switchの作成](#)」
- 「[Switch GroupやSwitchを削除する](#)」
- [19章 RTPCの使い方](#)

## Switchに関する、コツとベストプラクティス

Switchを使う時に、以下のセクションに目を通して、オーディオ開発プロセス全体を通して効率的にStateを使って作業するための、コツやベストプラクティスを参照してください。

## Switchの名前変更

Switchの名前を変更する場合は、Switchがゲームにどのように実装されているかを、事前に確認してください。Switch名を含む文字列を使って実装していれば、名前を変更した後もSwitchを使用するには、追加のプログラミングが必要となります。

## SwitchやSwitch Groupの削除

SwitchやSwitch Groupを削除する場合は、Switchがゲームにどのように実装されているかを、事前に確認してください。Switchをゲームに実装するには、以下の2つの方式があります。

- Switchアクションを設定したイベントを、呼び出す方式: SwitchやSwitch Groupを削除すると、呼び出してもSwitchが存在しないので、Wwise上で問題が発生する。
- Switch GroupとSwitch自体を、呼び出す方式: サウンドデザイナーがSwitchやSwitch Groupを削除する場合は、変更内容をプログラマーに連絡する必要がある。

## オブジェクトを、複数のSwitch Groupにアサインするには

通常は、1つのオブジェクトを、1つのSwitch Groupに限りアサインします。しかし、ゲームの都合上、あるオブジェクトを複数のSwitch GroupのSwitchにアサインする必要がある場合は、2番目のSwitch Groupを、Master-Mixer Hierarchy、Actor-Mixer Hierarchy、またはInteractive Music Hierarchyの、より高いレベルでアサインします。

---

## 第19章 RTPCの使い方

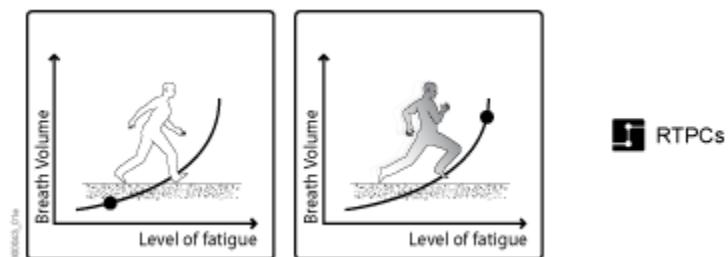
概要 .....	408
RTPCで使う、Game Parameterの管理 .....	409
Game Parameterを使って、プロパティ値をコントロールする .....	414
LFOの活用 .....	421
エンベロープの活用 .....	422
ゲームオブジェクトの観察 .....	424
RTPCに関する、コツとベストプラクティス .....	425

## 概要

ゲームをよりダイナミックに仕上げるために、一部のオブジェクトのプロパティをゲーム内の特定パラメータ値に結びつけることができます。Wwiseでは、RTPC (Real-time Parameter Control / RTPC) を使ってこれを実現できます。RTPCを作成するには、複数のポイントでカーブを設定します。ゲーム側のパラメータ (Game Parameter) に、Wwiseで設定したプロパティが連動する関係を、このカーブで設定します。ゲーム中にGame Parameterが変化すると、WwiseがRTPCカーブを使用して、対応する適切なプロパティ値を判断します。

### RTPCの活用例

例えば、ファーストパーソンシューティングゲームで、メインキャラクターの呼吸を、ゲーム中のキャラクターの疲労レベルに合わせる方法を考えます。キャラクターの疲労レベルが低い時には、呼吸サウンドを極めて静かにして、疲労レベルが高い時には、呼吸サウンドを大きくします。この場合、RTPCを利用して、Game Parameter (ここでは疲労レベル) を、Wwiseのプロパティ (ここではボリューム) にアサインします。次に、グラフビューを使って、呼吸サウンドのボリュームレベル (Breath volume) を、メインキャラクターのゲーム中の疲労レベル (Level of fatigue) に、マッピングします。



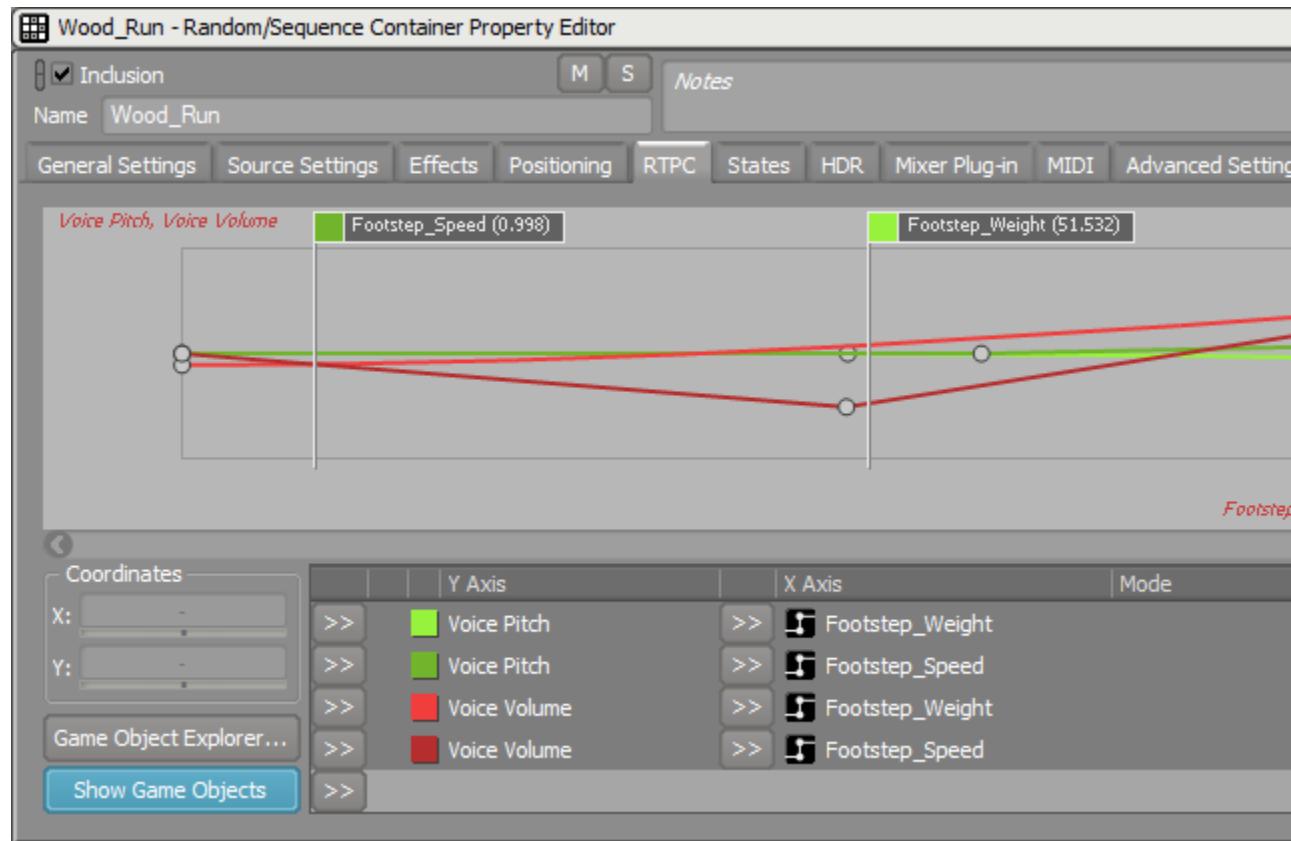
RTPCを、ゲーム中の他のエフェクトにも使用でき、例えば、水の深さに連動してローパスフィルタの数値を変えることや、爆発の強度に連動して低周波エフェクトの数値を変えることができます。

### Wwiseの、RTPCを理解する

WwiseのRTPCの作成には、以下が伴います:

- 「Game Parameterの作成」
- 「Game Parameterに対して、Wwiseプロパティを設定する」
- 「RTPCグラフ上で、数値をマッピングする」

あるプロパティ値と、Game Parameterとの関係は、Property Editor、Effect Editor、またはAttenuation Editorの、RTPCタブで設定します。



ゲームから送られるGame Parameter値（X軸）を、Wwiseオブジェクトのプロパティ値（Y軸）に対してマッピングすることで、そのオブジェクトのプロパティ変化を設定できます。コントロールポイントを設定すると、Wwiseがその間を補間して、RTPCカーブ（RTPCのグラフ）が作成されます。

1つのオブジェクト、バス、減衰インスタンス、またはエフェクトインスタンスに対して、複数のカーブを設定でき、1つのカーブで、そのオブジェクトのあるプロパティと、あるGame Parameterとの関係を表します。比較のために複数のカーブを同時に表示することも、1つだけを表示することもできます。

オブジェクトのプロパティと、ゲーム内のパラメータとの間に、より複雑な関係を設定するには、Blend Containerのブレンドトラック内でRTPCを利用します。Blend Container内の、RTPCの使用については「[Blend Containerの中身と、再生方法の設定](#)」を参照してください。



### 注意

RTPCは、プロジェクトにある全てのオブジェクト、バス、減衰インスタンス、エフェクトインスタンスに対して、設定することができますが、メモリやCPUに対する負荷が大きくなるため、対象を厳選して利用することが重要です。

## RTPCで使う、Game Parameterの管理

ゲーム中のパラメータを、Wwiseのプロパティにマッピングする前に、まずGame Parameterを作成する必要があります。Game Parameterリストの管理や、最小値や最大値の設定は、Project Explorerの、Game Syncsタブで行います。

インターフェース上で分かりやすいように、Game Parameterは、以下のアイコンで表されます。

アイコン	内容
	Game Parameter

Game Parameterの管理には、以下のタスクが伴うことがあります:

- 「Game Parameterの作成」
- 「Game Parameter値の範囲を設定する」
- 「Game Parameterの削除」
- 「ビルドインパラメータにGame Parameterをバインドする」

### Game Parameterの作成

オブジェクトのプロパティ値を、Game Parameterを使って変化させるには、まず、Project Explorerの、Game Syncsタブで、Game Parameterを作成する必要があります。Game Parameterを作成すれば、それを使ったRTPCを、無制限に作成できます。



#### 注意

Game Parameterの名前に、文字、数字、アンダースコアのみを使用し、またGame Parameterの名前は、必ず固有のものとします。

必要なGame Parameterは全て、Wwiseの以下のどちらの場所でも、作成できます。

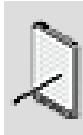
- Project Explorerの、Game Syncsタブ
- Property Editor、Attenuation Editor、またはEffect Editorの、RTPCタブ

### Project Explorerで、新規Game Parameterを作成するには:

- Project Explorerで、Game Syncsタブに切り替える。
- Game Parametersセクションで、以下のいずれかを行う。
  - ワークユニット（または仮想フォルダ）を選択して、Project Explorerのツールバーの、Game Parameterアイコンをクリックする。
  - ワークユニット（または仮想フォルダ）を右クリックして、ショートカットメニューで、New Child > Game Parameterを選択する。

Game Parameterのリストに、新規Game Parameterが追加される。

- デフォルトのGame Parameter名を、内容が分かる名前に変更する。



#### 注記

名前は必ず文字またはアンダースコアで始めます。

- 必要に応じて、Game Parameterをさらに追加する。

## RTPCタブで新規Game Parameterを作成するには:

1. Property Editor、Attenuation Editor、またはEffect EditorのRTPCタブで、RTPCリストの項目を1つ選択する。  
選択した項目のプロパティがまだ選択されていない場合は、ボタンを使って1つ選択する。
2. メニューでGame Parametersを選択してから、Newを選択する。  
New Game Parameterダイアログボックスが開く。
3. 作成する新規ゲームパラメータを入れる、ワークユニットを選択する。
4. Nameフィールドに表示されるデフォルトのGame Parameter名を、内容が分かる名前に変更する。
5. OKをクリックすると、新規Game Parameterが作成される。

### 関連トピック

- [「Game Parameter値の範囲を設定する」](#)
- [「Game Parameterの削除」](#)
- [「ビルトインパラメータにGame Parameterをバインドする」](#)

### Game Parameter値の範囲を設定する

Game Parameterを作成したあとは、その最小値と最大値を設定する必要があります。例えばレーシングカーの場合は、最低スピードと最高スピードを、「時速0km」と「時速300km」に設定できます。

また、デフォルトのGame Parameter値を指定することもできます。これは、特定の数値が明示されていないゲームオブジェクトにおいて、必ず使用するグローバル値です。なお、Wwiseで設定したデフォルトのGame Parameter値は、以下の状況では無視されます。

- あるゲームオブジェクトに関して、特定の数値が明示されている場合
- ゲームプログラマーが、SDKで、グローバルRTPC値を定義している場合

### Game Parameter値の範囲を設定するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. Game Parametersセクションで、範囲を設定するGame Parameterを、ダブルクリックする。  
Property Editorに、このGame Parameterがロードされる。
3. パラメータのMin（最小値）とMax（最大値）を指定して、パラメータ値の範囲を設定する。
4. ゲームオブジェクトに対して、特定の数値が明示されていない場合に使用するグローバル数値を、テキストボックスDefaultで指定する。

### 関連トピック

- [「Game Parameterの作成」](#)

- 「Game Parameterの削除」
- 「ビルトインパラメータにGame Parameterをバインドする」

### Game Parameterの削除

プロジェクトで不要となったGame Parameterは、削除できます。Game Parameterを削除すると、そのGame Parameterを使用していたオブジェクト、イベント、プリセットで、以後はそれを使えなくなります。



#### 注記

Game Parameterを削除する前に、必ずオーディオプログラマーに連絡して、ゲームコードに影響しないことを確認してください。

### Game Parameterを、削除するには:

- Project Explorerで、Game Syncsタブに切り替える。
- Game Parametersセクションで、削除するGame Parameterを右クリックして、Delete Selectionを選択する。

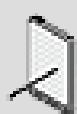
このGame ParameterがWwiseから削除される。

### 関連トピック

- 「Game Parameterの作成」
- 「Game Parameter値の範囲を設定する」

### ビルトインパラメータにGame Parameterをバインドする

このサウンドエンジンは、サウンドデザイナーがダイナミックなオーディオやモーションを作りあげるのに丁度良いと思うであろう値の標準的なセットを計算します。これらの「ビルトイン」パラメータには、パラメータメカニズムを組み込むためのバインドを使ってアクセスし、各フレーム毎にアップデートされます。ビルトインパラメーターに結びついたGame Parameterを使うには、追加のゲームプログラミングは必要ありません。



#### 注記

RTPC がビルトインパラメータに結びついている場合には、サウンドエンジンが各ゲームプロジェクトの値を更新します。バスもしくはバスエフェクトなどのゲームオブジェクトに関連性のないグローバルのWwise オブジェクトでRTPCが使われている場合には、デフォルト値が使用されます。

### 利用できるビルトインパラメーター:

- Distance (距離)

ゲームオブジェクトとリスナー間の距離。ゲームオブジェクトに複数のリスナーや複数のポジションが関連付けられている場合には、全てのリスナーおよびサウンドポジションの組み合わせの間で最短距離の値を取ります。

- Azimuth (方位)

水平面におけるリスナーとゲームオブジェクトの間の角度を度で示します。0°の値は、そのサウンドがリスナー直接正面にあることを示し、-90°はサウンドが左側、90°は右側、そして +/- 180°はサウンドがリスナーの直接後ろにあることを示します。

ゲームオブジェクトが複数のリスナーやサウンドポジションに割り当てられている場合、リスナーとサウンドポジションを最も近くに寄せる角度が取られます。

- **Elevation (高度)**

リスナーとゲームオブジェクト間の、水平方向に対して垂直の角度を度で示します。0°の値は、そのサウンドがリスナーと同じ水平面にあることを示し、90°はサウンドが直接上、-90°は直接下にあることを示します。

ゲームオブジェクトが複数のリスナーやサウンドポジションに割り当てられている場合、リスナーとサウンドポジションを最も近くに寄せる角度が取られます。

- **Emitter Cone**

Emitter Coneは、エミッターのorientation (向き) ベクトルと、エミッター・リスナー間の直線で形成されたベクトルの間の、3D角度を表します。値が0度の時はエミッターの正面にリスナーがあり、値が180度の時はエミッターがリスナーと反対の方向を向いていることを示します。

ゲームオブジェクトが複数のリスナーやサウンドポジションに割り当てられている場合、リスナーとサウンドポジションを最も近くに寄せる角度が取られます。

- **Obstruction**

Obstructionは、[SetObjectObstructionAndOcclusion API](#)を経由してゲームオブジェクトに設定された値へのアクセスを提供します。

ゲームオブジェクトに複数のリスナーが割り当てられている場合、Obstruction値は現在のサウンドポジションに一番近いリスナーに割り当てられた値を取ります。

- **Occlusion**

Occlusionは、[SetObjectObstructionAndOcclusion API](#)を経由してゲームオブジェクトに設定された値へのアクセスを提供します。

ゲームオブジェクトに複数のリスナーが割り当てられている場合、このOcclusion値は現在のサウンドポジションに一番近いリスナーに割り当てられた値を取ります。

- **Listener Cone**

Listener Coneは、リスナーの向きベクトル間の3Dの角度であり、リスナーとリスナー間の直線で形成されたベクトルです。0度の値はそのエミッターがエミッターの方向に直接向いており、値が180度はそのリスナーがリスナーの方向の反対向きになっていることを示します。

ゲームオブジェクトが複数のリスナーやサウンドポジションに割り当てられている場合、リスナーとサウンドポジションを最も近くに寄せる角度が取られます。

### • Diffraction

Diffraction（回折）は、ルームやポータルの音の伝播を使うときに、Wwise Spatial Audioが計算した回折角へのアクセスを提供します。

この組み込みパラメータを受信するには、ゲームオブジェクトがSpatial Audioに登録されていて、エミッタとリスナーが1つ以上のポータルで接続された別々のルームにある必要があります。

エミッタゲームオブジェクトは、ドライな回折に関連する値として、エミッタとリスナーを結ぶ直線パスからはずれた角度を示す発散角を、受信します。Spatial Audio内に登録されたルームゲームオブジェクトも回折値を受信しますが、これはウェット回折に関連する、ルーム内にある音の拡散音場の回折です。ウェット回折は、ポータル開口部の垂直線からの発散角度です。

複数のポータルがあり、音がリスナーに到達するまでのパスが複数ある場合は、パスの中でも回折角が最も狭いものが採用されます。

## 関連トピック

- 「Game Parameterの作成」
- 「Game Parameter値の範囲を設定する」
- 「Game Parameterの削除」

## Game Parameterを使って、プロパティ値をコントロールする

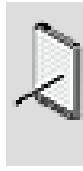
Game Parameterを作成したあとは、プロジェクトにある各種オブジェクト、バス、エフェクトインスタンス、減衰インスタンス、Switch Groupなどで使用する RTPCを作成できます。RTPCの設定は、Property Editor、Effect Editor、または Attenuation Editorの、RTPCタブで行うか、Blend Containerのブレンドトラック内で設定します。RTPCの作成には、以下の手順が伴います：

- 「Game Parameterに対して、Wwiseプロパティを設定する」
- 「RTPCグラフ上で、数値をマッピングする」

プロパティ値をGame Parameter値にマッピングするための、RTPCカーブ (RTPCのグラフ) を、グラフィビューで作成します。Wwiseのグラフィビューで同時に多数のカーブを表示することができるので、カーブは色分けされています。3種類の相対プロパティ（ボリューム、ピッチ、ローパスフィルタ）の色は既定されていますが、プロジェクト内のその他のプロパティは、任意に割り当てられた色で表示されます。この表示色は、ワークセッションごとに異なることがあります。

相対プロパティの表示色を、下表に示します。

Wwiseで設定するプロパティ	色
Volume	(Red) 
Pitch	(Green) 
LPF	(Blue) 



## 注記

全てのプロパティが、全てのオブジェクトにおいて、全てのプラットフォームで使用されているとは限らず、例えば、ピッチのプロパティは、ミュージックオブジェクトに影響しません。

カーブが色分けされていても、特に1つのカーブに専念したい時などは、いくつかのカーブを非表示にすると便利です。カーブの非表示については「[グラフビューで、カーブを表示する](#)」を参照してください。

プロジェクトで RTPC を使い始める前に、RTPC 値と既存のプロパティ値の関係を理解しておくことが重要です。RTPC を既存のプロパティ値に適用した場合は、最終的なプロパティ値が、以下のいずれかの方法で決まります。

- ・ **絶対値 (Absolute)** - RTPC で決定した数値が採用され、そのオブジェクトの既存プロパティ値は無視される。
- ・ **相対的 (Relative)** - RTPC で決定した数値が、そのオブジェクトの既存プロパティ値に、加算される。

最終的なプロパティ値を決定するための方法は、RTPC プロパティが絶対値か、相対的に設定される数値かによって、決まります。プロパティが絶対値であれば、元のプロパティコントロールが無効となります。相対的であれば、もとのプロパティコントロールが引き続き継続して有効です。



## 注記

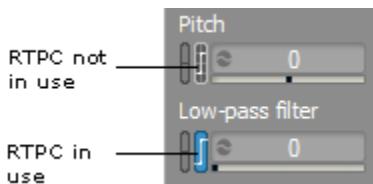
絶対設定か、相対設定かは、既定されているため変更できません。

## Game Parameterに対して、Wwiseプロパティを設定する

Game Parameterを作成して、そのパラメータ値の範囲を設定できたら、次にどの Game Parameter を、どのプロパティにアサインするのかを設定します。

Game Parameterにプロパティをアサインすると、以下となります。

- ・ Graph ビューに、XY 軸が表示される。
- ・ Graph ビューに、デフォルトの RTPC カーブが表示される。
- ・ このプロパティ値の横の RTPC アイコンが、青くなる。



## Game Parameterに、オブジェクトのプロパティをアサインするには:

1. オブジェクト、バス、エフェクトインスタンス、または減衰インスタンスを、該当する Editor にロードする。

2. RTPCタブに切り替える。
3. セレクタボタン (>>) をクリックして、リストにあるプロパティを、1つ選択する。

新規RTPCカーブが作成され、専用の色で表示されます。また、グラフビューのY軸に、Wwiseのプロパティが表示されます。

4. X Axisリストで、WwiseのそのプロパティにアサインするGame Parameterを選択する。

グラフビューのX軸に、Game Parameterが表示される。

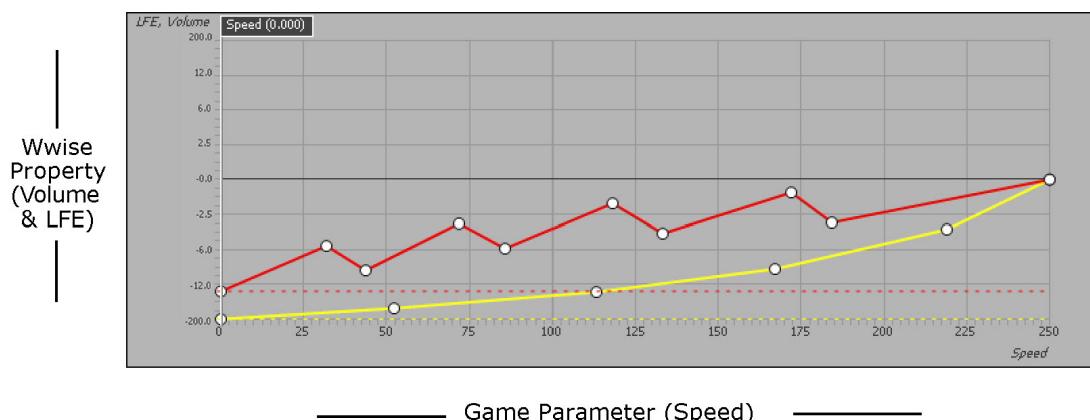
## 関連トピック

- 「Game Parameterを使って、プロパティ値をコントロールする」
- 「RTPCグラフ上で、数値をマッピングする」
- 「リストにあるRTPCの削除」

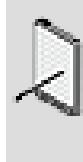
## RTPCグラフ上で、数値をマッピングする

WwiseのプロパティをGame Parameterにアサインできたら、両者の関係をグラフ(Graph)ビューを使って定義します。コントロールポイントを使い、プロパティ値をGame Parameter値にマッピングしていきます。例えば、車両が時速250kmの時に、サウンドのボリュームを0dBとする場合は、「250 km/h」と「0dB」の交点に、コントロールポイントを追加します。2つのコントロールポイント間が補間されて、RTPCカーブ(RTPCのグラフ)が作成されます。

例えば、Game Parameter値「Speed (スピード)」を、Wwiseのプロパティ値である「Volume (ボリューム)」にマッピングした様子を、以下のRTPCグラフに示します。「Speed」値(X軸)はゲーム側で決まり、「Volume」値(Y軸)は、自由に設定できます。



グラフのズームやパン、コントロールポイントの追加、移動、削除、ポイント間のカーブ形状の変更、「Linear(均等)目盛り」や「dB目盛り」の使用など、グラフビューの一般的な情報については[43章グラフビューについて](#)を参照してください。

**注記**

1つのオブジェクト内部で、またプロジェクトの複数のオブジェクト同士で、RTPCのカーブをコピーできます。RTPCカーブのコピーについては「[RTPCカーブのコピー](#)」を参照してください。

**グラフ上で、数値をマッピングするには:**

1. Graphビューで、カーブ上の任意の位置をダブルクリックして、コントロールポイントを設定する。
2. 以下のいずれかを行う:
  - コントロールポイントを、適切なXY座標の位置に、ドラッグする。
  - テキストボックスX Coordinateと、Y Coordinateに、XY座標の数値を直接入力して、コントロールポイントの位置を正確に指定する。
3. ポイントを追加し続けて、プロパティとGame Parameterの関係が全体的に表されたカーブを作成する。

**Tip**

コントロールポイントを作成したあと、マウスボタンを押したままにすると、そのままポイントを適切な場所にドラッグできます。

**関連トピック**

- 「[コントロールポイントの追加](#)」
- 「[コントロールポイントの選択](#)」
- 「[コントロールポイントの移動](#)」
- 「[コントロールポイント間の、カーブ形状の指定](#)」
- 「[Game Parameterを使って、プロパティ値をコントロールする](#)」
- 「[Game Parameterに対して、Wwiseプロパティを設定する](#)」
- 「[RTPCカーブのコピー](#)」
- 「[リストにあるRTPCの削除](#)」

**スマートピッチカーブの作成**

RTPCを使って、サウンドのピッチをGame Parameterに結びつける場合は、複雑なカーブを設定することがリアルなエフェクトへの鍵となります。スマートピッチカーブ (Smart pitch curve) を使えば、最小限の手間で自然なサウンドのRTPCピッチカーブを作成できます。

スマートピッチカーブは、以下の2種類の変数に基づいています。

- **Native value:**そのサウンドが、元のピッチで聞こえる時の、Game Parameter値。
- **Subdivision level:**1から10までの範囲で設定した、カーブの精度。

例えば、レーシングゲームのエンジンサウンドにスマートピッチカーブを適用できます。エンジンの回転サウンドを2000 RPMで録音した場合、ネーティブ値を

「2000 RPM」に設定したRTPCピッチカーブを作成します。Wwiseは、これを元に、自然なサウンドのカーブを作成するために必要な残りのポイントを、推定して設定します。

ピッチRTPCカーブの作成にスマートピッチカーブを使うと、カーブの直線部分のサブセグメントを作成するために、Wwiseでは以下の方程式が用いられます。

$$\text{Pitch (in cents)} = 1200 * \log_2 \left( \frac{x}{\text{reference } x} \right)$$

カーブのサブセグメントが多いほど、精度が高くなります。ただし、サブセグメントが増えると、ランタイムに必要な処理時間も増えます。CPUやメモリを節約するために、必要な効果を得られる最低限の分割レベル (Subdivision level) を選択してください。



#### Tip

正確なカーブの作成については「スマートピッチカーブの有効範囲」を参照してください。

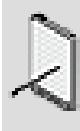
### スマートピッチカーブを作成するには:

1. RTPCグラフビューで、ピッチカーブを作成する。詳しくは「Game Parameter に対して、Wwiseプロパティを設定する」を参照。
2. ピッチカーブを右クリックして、メニューでBuild Smart Pitch Curve（スマートピッチカーブを作成）を選択する。

Build Smart Pitch Curveダイアログボックスが開く。

3. スマートピッチカーブの、ネーティブ値 (Native value) を入力する。ネーティブ値の定義は、サウンドが録音された時のプロパティ値である。
4. スマートピッチカーブの、分割レベル (Subdivision level) を入力する。カーブの分割レベルが上がるほど、セグメント数が増え、ランタイム時の算定にかかる時間が長くなる。
5. OKをクリックする。

スマートピッチカーブが作成される。



#### 注記

スマートピッチカーブで作成できるカーブのピッチ範囲は、「- 4800」セントから「4800」セントまでです。- 4800以下と、4800以上では、ピッチ変動の範囲外となり、y軸に垂直な直線となります。

### 関連トピック

- 「コントロールポイントの追加」

- 「コントロールポイントの選択」
- 「コントロールポイントの移動」
- 「コントロールポイント間の、カーブ形状の指定」
- 「Game Parameterを使って、プロパティ値をコントロールする」
- 「Game Parameterに対して、Wwiseプロパティを設定する」
- 「RTPCグラフ上で、数値をマッピングする」
- 「RTPCカーブのコピー」
- 「リストにあるRTPCの削除」

## RTPCカーブのコピー

プロジェクトによっては、同じRTPCカーブを、他のプロパティ、または場合によっては他のオブジェクトで使用したい場面も考えられます。カーブを別々に作成する代わりに、一度カーブの形状を作成してから、それをコピーして、同じオブジェクトの別のプロパティにペーストしたり、さらにはプロジェクト内の他のオブジェクトにペーストしたりすることもできます。

RTPCカーブのコピーは、Wwiseの以下の場所で行います。

- Master-Mixer Hierarchy、Actor-Mixer Hierarchy、Interactive Music Hierarchyにある、オブジェクトやバスのRTPCタブ
- エフェクトの、RTPCタブ
- ソースプラグインの、RTPCタブ
- Blend Track Editor

同一オブジェクトのRTPCカーブや、他のオブジェクトのRTPCカーブを、コピーするには:

1. カーブのリストで、コピーするカーブを選択する。



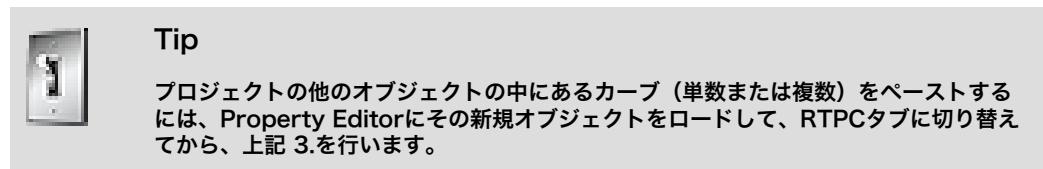
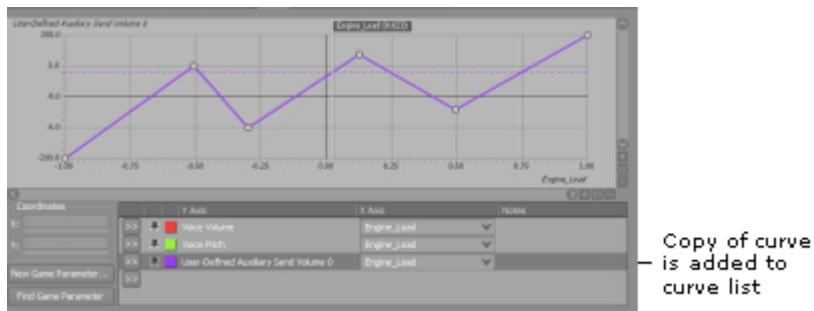
2. 以下のいずれかを行う:

- 選択したカーブを右クリックして、メニューでCopyを選択する。
- Ctrl+Cを押す。

3. 以下のいずれかを行う:

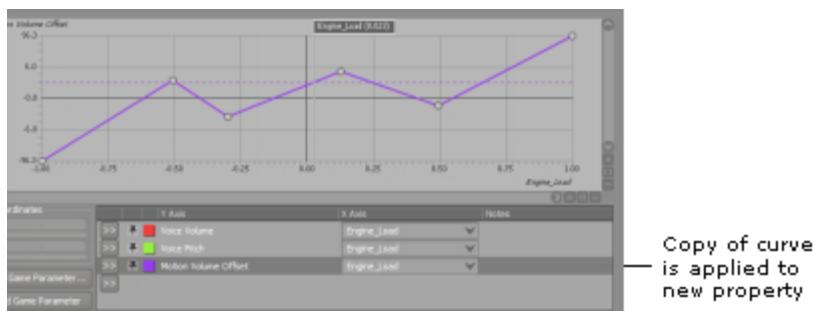
- 右クリックして、メニューでPasteを選択する。
- Ctrl+Vを押す。

そのRTPCカーブの正確なコピーが、リストに追加される。



4. セレクタボタン (>>) をクリックして、リストにあるプロパティを、1つ選択する。

選択したプロパティに、このカーブ形状が適用される。



## 関連トピック

- 「コントロールポイントの追加」
- 「コントロールポイントの選択」
- 「コントロールポイントの移動」
- 「コントロールポイント間の、カーブ形状の指定」
- 「Game Parameterを使って、プロパティ値をコントロールする」
- 「Game Parameterに対して、Wwiseプロパティを設定する」
- 「RTPCグラフ上で、数値をマッピングする」
- 「リストにあるRTPCの削除」

## リストにあるRTPCの削除

Wwiseのあるプロパティの数値を、Game Parameterを使って変化させるのをやめるには、RTPCリストから削除します。

### リストから、RTPCを削除するには:

1. RTPCリストで、削除するRTPCを選択する。
2. Deleteキーを押す。

リストから、このRTPCが削除される。

### 関連トピック

- ・「Game Parameterを使って、プロパティ値をコントロールする」
- ・「Game Parameterに対して、Wwiseプロパティを設定する」
- ・「RTPCグラフ上で、数値をマッピングする」

## LFOの活用

LFO (Low Frequency Oscillator) は時間の経過と共にプロパティ値のモジュレーションを発生させるために使います。LFOのプロパティを以下に示します。

Property	内容
Depth	オシレータの振幅変化率 (%)
Frequency	一秒間のサイクル数 (Hz)
Waveform	モジュレータの形状
Smoothing	波形のハードなエッジをならすためのLPF (%)
PWM (Pulse Width Modulation)	矩形波のみに適用されるパルス波の幅 (%)
Attack	オシレータがフル振幅に到達するのにかかる時間 (秒数)
Initial Phase	オシレータ波形の初期位相 (角度)
Scope	LFOインスタンスの作成方法を選択する。 <ul style="list-style-type: none"><li>• Voice: サウンドやオブジェクトの再生ごとに、LFOインスタンスを作成する。</li><li>• Note/Event: 全ての再生インスタンス、またはMIDIの場合は全てのノートで、LFOインスタンスを作成する。</li><li>• Game Object: ゲームオブジェクトごとにLFOインスタンスを作成する。</li><li>• Global: プロジェクト全体用に1つのLFOを作成する。</li></ul>

Wwiseには追加型のプロパティ (Voice Volume、Voice Pitchなど) と排他的なプロパティがあります。追加型のプロパティにLFOを追加すると、このプロパティの現在値にLFOモジュレーションが付加されます。排他的プロパティにLFOを追加すると、このプロパティの現在値がLFOモジュレーションによって置き換えられます。

### LFOでVoice Volumeのモジュレーションを行うには:

1. Project Explorerで、LFOを加えるオブジェクトを選択する。
2. Property Editorで、RTPCタブを開く。
3. RTPCリストで、[>>]ボタンをクリックする。
4. セレクタメニューで、Voice Volumeを選択する。

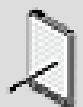
5. X軸のセレクタボタンをクリックする。
6. セレクタメニューで、LFO > Default (Custom)を選択する。
7. [...]ボタンをクリックしてLFOプロパティを編集する。
8. カーブを編集してモジュレーション範囲を設定する。

LFOオブジェクトはCustomまたはShareSetとして作成できる。Customオブジェクトはその場所、つまりそれを有するオブジェクト内に直接保存される。ShareSetは別のWork Unitに保存され、他のオブジェクトで再利用できる。



#### 注意

モジュレータの処理時間はRTPCの利用内容によって変わります。多くのプロパティにおいて、モジュレータはオーディオフレームごとに評価されます。ただしvoice volumeプロパティの場合は、フレームごとに関連モジュレータが評価されます。モジュレータはメモリやCPUを多く消費することがあるので、厳選して使ってください。



#### 注記

ModulatorのRTPCカーソル（LFOとエvelopeの両方）は、モジュレータの内部プロパティによって決定されるため、任意の時点で特定の値を表示しません。

## エンベロープの活用

エンベロープは、既に形状を決めたプロパティ値をコントロールするために使います。以下の通り、エンベロープの形状は [ADSR](#)型です。

Property	内容
Attack Time	ゼロからピークレベルまでの最初の上昇にかかる時間で、最初にキーを押した時からの時間を指す（秒数）。
Attack Curve	デフォルトの線形スロープ（50%）Attack Curveを変更して調整し、指数関数的なエンベロープ（0%）に設定するとゆっくり変化し始めてから増加し、対数関数的なエンベロープ（100%）に設定すると早く変化し始めて徐々におさまる。
Decay Time	Attackレベルから、指定したサステインレベルに落ち着くまでの時間。（秒数）
Sustain Level	キーを離すまで、サウンドの主要部分が続いている間のレベル。（範囲全体に対して%）
Release Time	キーを離したあとにレベルがサステインレベルからゼロまでディケイするのにかかる時間。（秒数）
Scope	エンベロープインスタンスがどのように作成されるかを定義: <ul style="list-style-type: none"> <li>• <b>Voice:</b> エンベロープインスタンスは各サウンド/オブジェクトの再生に作成される。</li> <li>• <b>Note/Event:</b> エンベロープインスタンスは各再生されているインスタンス、またはMIDIコンテキストで使われるときのノートに作成される。</li> </ul>
Trigger On	エンベロープをTriggerできるアクション/MIDIイベント（たとえば、アタックフェーズに入るなど）。 <ul style="list-style-type: none"> <li>• <b>Play:</b> play アクションまたは MIDI ノートイベントのいずれか。</li> <li>• <b>Note-Off:</b> MIDI ノートオフイベントのみ</li> </ul>
Auto Release	サステイン状態を終了してリリース状態に入るため、エンベロープがアクション/MIDIイベントを必要かどうかを判断します。設定す

Property	内容
	ると、エンベロープはSustain Time後、サステイン状態を終了します。設定しないと、エンベロープはサステイン状態を終了するためのイベントが必要となります。
Sustain Time	リリースを適用するまで、エンベロープのサステイン状態が続く時間。この数値は、Auto Releaseに設定した場合に限り有効である。
Stop playback after release	設定すると、リリース状態が完了すると関連したサウンドの再生が終了します。

エンベロープは、MIDIでも通常の再生でも使用できます。

MIDIのコンテキストで使用されると、このエンベロープはノートオンもしくはノートオフのいずれかで再生されるサウンドのために設定されます。ノートオンで再生されるサウンドの場合:

- ・エンベロープはノートオンでトリガーするように設定されます (Trigger On パラメータ)。
- ・以下が初めて発生するまで、エンベロープがサステインされます:
  - ・Release Envelopeイベントの受信、
  - ・MIDI note-offイベントの受信、
  - ・サステインフェーズのMax Duration (Auto Releaseが設定されている)。

一般的なサウンド再生で使用する場合は以下の通りです。

- ・エンベロープはノートオフでトリガーするように設定されます (Trigger On パラメータ)。
- ・以下が初めて発生するまで、エンベロープがサステインされます:
  - ・Release Envelopeイベントの受信、
  - ・サステインフェーズのMax Duration (Auto Releaseが設定されている)。

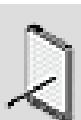
サウンド再生の一般的なコンテキストで使用される場合:

- ・エンベロープはplayアクションでトリガーされるように設定されます (Trigger On パラメータ)。
- ・以下が初めて発生するまで、エンベロープがサステインされます:
  - ・Release Envelopeイベントの受信、
  - ・サステインフェーズのMax Duration (Auto Releaseが設定されている)。



### 注意

モジュレータの処理時間はRTPCの利用内容によって変わります。多くのプロパティにおいて、モジュレータはオーディオフレームごとに評価されます。ただしvoice volumeプロパティの場合は、フレームごとに関連モジュレータが評価されます。モジュレータはプラットフォームのメモリやCPUを多く消費があるので、厳選して使ってください。



### 注記

ModulatorのRTPCカーソル (LFOとエEnvelopeの両方) は、モジュレータの内部プロパティによって決定されるため、任意の時点で特定の値を表示しません。

## MIDIオブジェクトにエンベロープを使う

エンベロープは、プロパティ値をコントロールする以外に、サウンドの寿命をコントロールするためにも使えます。エンベロープのオプションを使って、リリース完了後にオーナーを停止することができます。

**Actor-Mixer Hierarchyオブジェクト（楽器）にVoice Volumeエンベロープを追加するには：**

1. Project ExplorerでActor-Mixer Hierarchyのオブジェクトを選択する。
2. Property Editorで、 RTPCタブを開く。
3. RTPCリストの[>>]ボタンをクリックする。
4. セレクタメニューで、 **Voice Volume**を選択する。
5. Y軸の[>>]セレクタをクリックする。
6. セレクタメニューで、 **Envelope > Default (Custom)**を選択する。
7. [...]ボタンをクリックしてエンベロープを編集する。
8. カーブを編集してモジュレーション範囲を設定する。

## ゲームオブジェクトの観察

**ゲームオブジェクト**は、 RTPCによって複雑に影響を受けることがあります。両者の相互作用を追えるように、 RTPCグラフで、関心のあるゲームオブジェクトを表示するように設定できます。

**RTPCグラフでゲームオブジェクトを観察するには：**

1. Game Object Explorerを、 RTPCタブ画面の下パネルの左側にある**Game Object Explorer...**ボタンを使うなどして、開く。

オーディオプログラマーが作成したゲームオブジェクトのアクティブなものが全て表示される。

2. Game Object Explorerにゲームオブジェクトのウォッチを追加するには、「[ゲームオブジェクトウォッチを設定する](#)」の手順に従う。
3. RTPCタブ画面の下パネルに戻り、リストにあるRTPCを、それに関連するIn Game Parameterと共に1つ以上選択する。

Game Parameterを表すフラグが1つずつ、グラフに表示される。

4. 下パネルの**Show Game Objects**をクリックして、Wwiseツールバーの**Start to Capture**をクリックする。

ウォッチ中のゲームオブジェクトが全て、 RTPCグラフに表示される。



### 注記

このボタンを2つともクリックしなければ、 RTPCグラフにゲームオブジェクトが表示されません。

## 関連トピック

- 「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」
- 「ゲームオブジェクトのリストを理解する」

## RTPCに関する、コツとベストプラクティス

RTPC機能を使う前に、以下のセクションに目を通して、ゲーム中のサウンドやモーションを最大限に活用するための、コツやベストプラクティスを参照してください。



### RTPCの名前

Game Parameterの名前を変更する前に、プログラマーがゲームエンジンにどのように実装したかを、必ず確認してください。Game Parameterの名前を使って実装している場合は、Wwiseで変更するとプログラマーの追加作業が発生するので、できるだけ避けてください。

### パフォーマンス

RTPCを作成できる対象は、プロジェクトにある全てのオブジェクト、バス、エフェクトインスタンス、減衰インスタンス、Switch Group、ブレンドトラックですが、プラットフォームのメモリやCPUに対する負荷が大きいため、対象を厳選して利用することが重要です。

### スマートピッチカーブの有効範囲

多くの場合、スマートピッチカーブは、ネーティブ値の前後で良いエフェクトを出しますが、極端な数値では、必ずしも良いサウンドになりません。例えば、2000RPMで録音されたエンジンサウンドは2000RPMでは完璧に聞こえ、500～3500RPMの範囲でもおそらく良好でしょう。しかしそれを超えると、不自然に聞こえるかもしれません。1つの対処方法として、異なるネーティブ値の複数のサウンドを録音して、録音したサウンドを合わせたBlend Containerを成立させることです。Blend Containerについては「[Blend Containerの中身と、再生方法の設定](#)」を参照してください。

### RTPCを使って、ドップラー効果エフェクトを作成する

Wwiseのサウンドエンジンのパイプラインで使用するピッチのプロパティは、リサンプル用に高度に最適化されているため、リアルタイムで再生の加速や減速が可能です。Wwiseでドップラー風でのエフェクトをつくり出すための推奨案は、リスナーとサウンドソースの位置の差分をゲームエンジンでトラッキングする方法で、位置の差分が、基本的にスピード値と等しくなります。次にこのスピードを表すゲーム値でRTPCを作成して、対象サウンドのピッチ設定にマッピングします。これで、ゲーム中にリスナーとサウンドが近づく時、または遠ざかる時に、サウンド

のピッチが増加、または減少します。ドップラー効果をつくり出すには、この方法がCPUに最も負荷がかかりません。

Wwiseで複数のリスナー（分割スクリーンを含む）がある場合は、ピッチを利用するドップラー効果にデザイン上の特殊な配慮が必要です。現実世界では、速度や、オブジェクトとリスナーの距離に応じて、同じサウンドでもリスナーによってピッチ値が異なります。Wwiseで複数のゲームオブジェクトが同じ1つのサウンドを再生する時に、1つのインスタンスを使うため、サウンドに対して2つの異なるピッチ値を設定することはできません。このため、プログラミングで1つのピッチ値を決定する必要があります。

---

## 第20章 Triggerの使い方

概要 .....	428
Triggerの使い方 .....	428

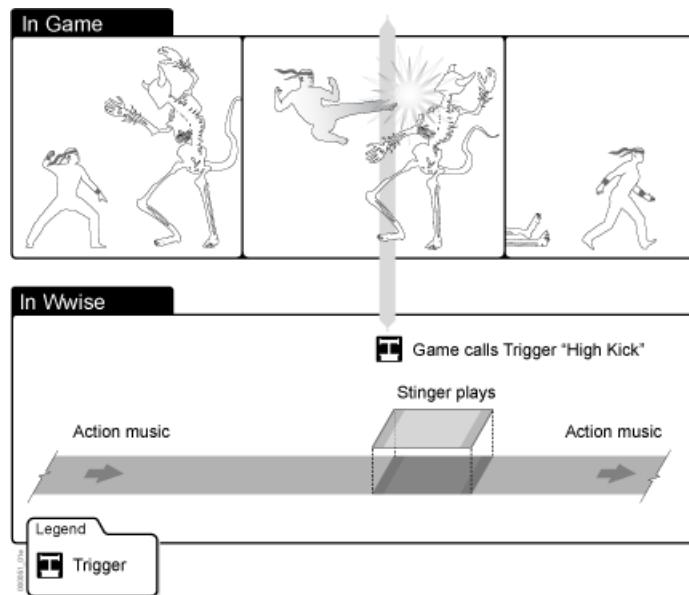
## 概要

他のゲームシンク機能と同様に、Triggerはゲームに呼び出され、ゲーム内の状況に応じてレスポンスを決定するWwiseのエレメントです。具体的には、インタラクティブミュージックにおいて、ゲームで発生した事象にTriggerが反応して、ステインガーを起動させます。ステインガーとは、再生中の音楽に重ねてミックスされる短いミュージックフレーズのことです。ゲームに対する音楽的な反応となります。例えば、忍者が武器を取り出した時に、シーンのインパクトを更に強めるため、既に再生中のアクションミュージックに重ねてスフォルツアンド風の音楽エフェクトを挿入できます。この場合、ゲームがTriggerを呼び出し、そのTriggerでステインガーが起動し、流れている音楽スコアに重ねてステインガーのミュージッククリップが再生されます。Triggerとステインガーと一緒に使う方法については[29章ステインガーの活用](#)を参照してください。

### Triggerの活用例

例えば、メインキャラクターの忍者ファイターが戦う格闘ゲームの場合を考えます。ゲーム中のいくつかのポイントで、キャラクターが敵と戦うアクションモードに入ります。ファイターが強力なキックをきめた時は、そのシーンの聴覚的な効果を強めるミュージッククリップを流すとします。このような場面の音楽を構築するには、ゲーム中の特定ポイントで呼び出すTriggerが必要で、今回の例では「High Kick（ハイキック）」というTriggerを設定します。同時に、興奮を高める瞬間的なプラス音楽の、短いMusic Segmentを定義します。

下図は、ゲームの重要なポイントでステインガーを再生させるTriggerのメカニズムを示しています。



## Triggerの使い方

ゲーム中の重要なポイントで、Triggerが1つのステインガーを呼び出し、そのステインガーが、短いMusic Segmentを、流れているミュージックに重ねて再生させま

す。ステインガーを作成するには、TriggerをMusic Segmentに割り当てて、再生方法を設定します。Triggerのリストは、Project Explorerの、Game Syncsタブで管理します。

インターフェース上で分かりやすいように、Triggerは、以下のアイコンで表されます。

アイコン	内容
	Trigger

Triggerの管理には、以下のタスクが伴います。

- ・「Triggerの作成」
- ・「Triggerの削除」

### Triggerの作成

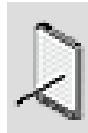
プロジェクトで必要なTriggerは全て、Project Explorerの、Game Syncsタブで作成できます。

#### Project Explorerで、新規Triggerを作成するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. Triggersセクションで、以下のいずれかを行う。
  - ・ワークユニット（または仮想フォルダ）を選択して、Project Explorerのツールバーの、Triggerアイコンをクリックする。
  - ・ワークユニット（または仮想フォルダ）を右クリックして、ショートカットメニューで、New Child > Triggerを選択する。

Triggerのリストに、新規Triggerが追加される。

3. デフォルトのTrigger名を、内容が分かる名前に変更する。



#### 注記

Triggerの名前は、必ず固有のものとし、文字、数字、アンダースコア以外は使えません。また、名前は必ず文字またはアンダースコアで始めます。

4. 必要に応じて、Triggerをさらに追加する。

### 関連トピック

- ・「Triggerの削除」
- ・29章スティンガーの活用

### Triggerの削除

Triggerが不要となり、削除する必要が出てくることもあります。スティンガーに設定されたTriggerを削除すると、スティンガーとMusic Segmentとの関係性も削除されるので、注意してください。

## Triggerを削除するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. Triggersセクションで、削除するTriggerを右クリックして、Delete Selectionを選択する。

選択したTriggerが、削除される。

## 関連トピック

- 「Triggerの作成」
- 29章スティンガーの活用

---

## 第21章 ダイナミックダイアログの、StateやState Groupの活用

概要 .....	432
StateやState Groupの活用 .....	433

## 概要

今日のゲーム、特にスポーツゲームでは、多くの場合、ダイナミックなオーディオコンポーネントを用いて、ゲームで起きるActionに対応させています。従来の方  
式で信ぴょう性のあるダイアログを成立させるには、何千ものアセットを作成し  
て、全ての可能なシナリオに対して、複雑なSwitch Container階層を1つ1つ構築  
する必要がありました。メモリ容量が限られる中で、プロジェクトのアセットを効  
率的に管理する必要があります。

このような限界に対応するために、Wwiseではダイナミックダイアログの新しい  
とらえ方を導入しています。ゲームの様々な条件や結果を、StateやState Group  
を用いて、Wwiseで事前に定義します。StateやState Groupはゲームに存在する  
様々なカテゴリーを表します。例えば、アメフトゲームのStateやState Groupとし  
て「Team」「Players」「Action」などが考えられます。また全てのStateやState  
Group、つまりカテゴリーに、対応するいくつかのState値を設定する必要があります。  
アメフトの例でいうと、Teams State GroupにDallas、Pittsburgh、New  
EnglandなどのState値を設定できます。

State GroupやStateを組み合わせたダイアログイベントで、あらゆるゲーム条件を  
全て再現します。この条件はパスと呼ばれ、それぞれ特定のボイスオブジェクトに  
アサインされます。ゲームのプレイ中に、現在のStateを、Wwiseのダイアログイ  
ベントの中の、作成したStateと比較することで、どのダイアログ（台詞）を再生  
するかが決まります。

### ダイナミックダイアログでの、State活用例

例えば、実況中継付きのゴルフゲームを考えましょう。ゲーム中のカテゴリーに  
合わせてStateやState Groupを作ります。次に、このカテゴリーに該当する全ての  
Stateを、StateやState Groupに設定します。ゴルフゲームの例では、「Players」  
「Clubs」「Courses」「Shots」「Locations」「Reactions」など様々なStateや  
State Groupが必要です。

下表は、ゴルフゲームにおける各種カテゴリーをStateやState Groupに分け、それ  
ぞれに該当するStateを整理した例です。

Arguments	Players	Clubs	Shots	Locations
Argument Values	Woods	Driver	Slice	Fairway
	Ames	Iron	Hook	Tee
	Cabrera	Wedge	Shank	Rough
	Garcia	Putter	Chip	Green

State GroupやStateが定義できたら、これらをゲームに必要なダイアログイベント  
に追加し始めます。Dialogue Eventの作成については「[ダイアログイベントの作  
成](#)」を参照してください。

## StateやState Groupの活用

StateやState Groupは、ゲームに存在する様々なカテゴリーを表し、例えばスポーツゲームのTeamsやPlayers、またはアクションアドベンチャーゲームのFriends、Enemies、Weaponsなどが、これに該当します。StateやState Groupのリストは、Project Explorerの、Game Syncsタブで管理します。

インターフェース上で分かりやすいように、State GroupやStateは、アイコンで表されます。

アイコン	内容
	State Group
	State Group値

StateやState Groupの管理には、以下のタスクが伴います。

- 「State Groupの作成」
- 「Stateの作成」
- 「State GroupやStateを削除するには:」

### State Groupの作成

プロジェクトで必要なStateやState Groupは全て、Project Explorerの、Game Syncsタブで作成できます。

### Project Explorerで、新規StateやState Groupを作成するには:

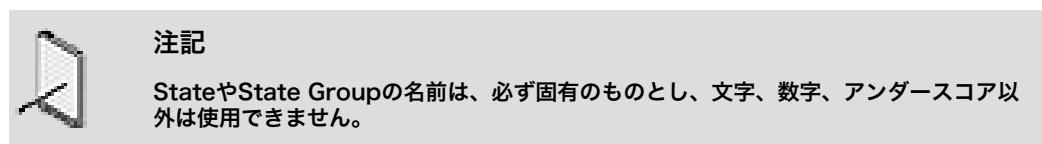
- Project Explorerで、Game Syncsタブに切り替える。
- State Groupsセクションで、以下のいずれかを行う。

Virtual Folder（またはWork Unit）を選択して、Project Explorerのツールバーの、State Groupsアイコンをクリックする。

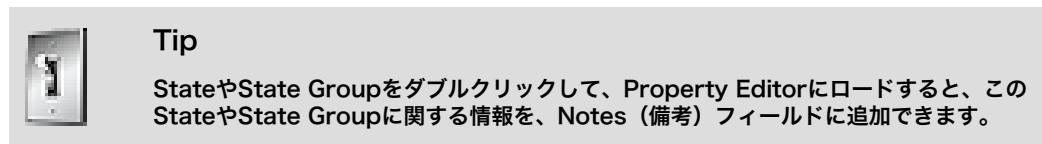
Virtual Folder（またはWork Unit）を右クリックして、ショートカットメニューでNew Child > State Groupを選択する。

新しいStateやState Groupが、StateやState Groupのリストに追加される。

- デフォルトのStateやState Group名を、内容が分かる名前に変更する。



- 必要に応じて、StateやState Groupを追加する。



## 関連トピック

- 「Stateの作成」
- 「State GroupやStateを削除するには:」

### Stateの作成

1つのStateやState Groupを作成すると、それに複数の設定値を設定することができます。設定値はそれぞれ、そのStateやState Groupの中にある、様々な選択肢を表します。例えば、StateやState Group Player NameにあるStateはそれぞれ、ゲーム内の選手の名前を表します。プロジェクトで必要なStateは全て、Project Explorerの、Game Syncsタブで作成できます。

### Project Explorerで、新規Stateを作成するには:

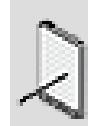
- Project Explorerで、Game Syncsタブに切り替える。
- State Groupsセクションで、以下のいずれかを行う。

StateやState Groupを1つ選択して、Project Explorerのツールバーの、Stateアイコンをクリックする。

StateやState Groupを1つ右クリックして、ショートカットメニューで、New Child > Stateを選択する。

StateやState Groupのリストに、新規Stateが追加される。

- デフォルトのStateの名前をStateやState Group値の内容が分かる名前に変更する。



#### 注記

Stateの名前は、そのStateやState Groupの中で必ず固有のものとします。

- 必要に応じて、Stateをさらに追加する。



#### Tip

Stateをダブルクリックして、Property Editorにロードすると、このStateに関する情報を、Notes (備考) フィールドに追加できます。

## 関連トピック

- 「State Groupの作成」
- 「State GroupやStateを削除するには:」

### State GroupやStateを削除するには:

State GroupやStateが不要となり、削除する必要が出てくることもあります。StateやState Groupを削除すると、そのStateやState Groupの中にあるStateも、全て削除される。StateやState Groupを削除すると、それを使用する全てのダ

## ダイナミックダイアログ の、StateやState Groupの活用

ダイアログイベントから、そのStateやState Groupが削除されるので、注意してください。

### StateやState Groupを削除するには:

1. Project Explorerで、Game Syncsタブに切り替える。
2. State Groupsセクションで、削除するStateやState GroupまたはStateを右クリックして、Delete Selectionを選択する。

選択したStateやState GroupまたはStateが削除される。



#### Tip

StateやState GroupまたはStateを選択して、Deleteキーを押すこともできます。

### 関連トピック

- ・「State Groupの作成」
- ・「Stateの作成」

**audio**kinetic

# パート V. インタラクティ ブミュージックを作成する



---

22. インタラクティブミュージックを理解する .....	439
概要 .....	440
インタラクティブミュージックを理解する .....	440
インタラクティブミュージックに関する、コツとベストプラクティス ....	442
23. Interactive Music Hierarchyの構築 .....	445
概要 .....	446
Music Segmentとは .....	446
コンテナの種類 .....	448
Interactive Music Hierarchyの、プロパティについて .....	449
Interactive Music Hierarchyに、オブジェクトを追加する .....	450
親オブジェクトの追加 .....	452
Interactive Music Hierarchyの、ミュージックオブジェクトの管理 .....	453
Interactive Music Hierarchyの構築に関する、コツとベストプラクティス .....	454
24. ミュージックオブジェクトの再生方法の設定 .....	455
概要 .....	456
ミュージックオブジェクトの時間設定 .....	456
Music Playlist Containerの、再生方法の設定 .....	457
Music Switch Containerの中身と、再生方法の設定 .....	460
インタラクティブミュージックの再生に関する、コツとベストプラクティス .....	464
25. Music Track、Music Segmentの活用 .....	466
概要 .....	467
セグメントに、Music Trackを追加する .....	468
Music Trackの再生方法の設定 .....	469
トラックに、サブトラックを追加する .....	472
サブトラックをSwitchやStateに結びつける .....	472
トラックの中身を入れる .....	473
セグメントから、トラックやサブトラックを削除する .....	474
Music Editorの、目盛に合わせたスナップ機能 .....	474
クリップの使い方 .....	475
セグメントの試聴 .....	477
キューの使い方 .....	479
26. MIDIの活用 .....	483
MIDIコンテンツを作成する .....	484
MIDIファイルのインポート .....	484
MIDIコンテンツとMIDIターゲットを理解する .....	484
MIDIコンテンツとオーディオコンテンツをミックスする .....	486
MIDIテンポを理解する .....	486
MIDIの再生スピードを変更する .....	487
27. MIDIインストゥルメントの作成 .....	488
Synth Oneインストゥルメントをデザインする .....	489
シンプルなMIDIサンプルインストゥルメントをデザインする .....	489
MIDIノートトラッキングを理解する .....	490
MIDIフィルタを理解する .....	490
MIDIイベントを理解する .....	491
MIDIイベントにフェードイン、フェードアウトを追加する .....	492
MIDIデータを使ってオブジェクトのプロパティ値をコントロールする ...	492
MIDI Keymap Editorを使う .....	492

---

MIDIキーボードでインストゥルメントをテストする .....	493
MIDIをDAWからWwiseにルーティングする .....	494
28. トランジションの活用 .....	495
概要 .....	496
トランジションを理解する .....	497
トランジションの追加 .....	498
トランジションのコピー、ペースト .....	499
トランジションの削除 .....	500
ソースやデスティネーションのプロパティを設定する .....	500
トランジションセグメントの使用 .....	504
インタラクティブミュージックのトランジションに関する、コツとベス トプラクティス .....	505
29. スティンガーの活用 .....	507
概要 .....	508
スティンガー (Stinger) の追加 .....	509
スティンガー (Stinger) の再生方法の設定 .....	511
スティンガーの削除 .....	512
スティンガー (Stinger) の試聴 .....	512

---

## 第22章 インタラクティブミュージックを理解する

概要 .....	440
インターラクティブミュージックを理解する .....	440
インターラクティブミュージックに関する、コツとベストプラクティス .....	442

## 概要

ビデオゲームにミュージックスコア（楽曲）を取り込むには、数々の特有な課題があります。ビデオゲームの音楽はインタラクティブであるだけでは足りず、簡単に実装でき、かつ利用可能なリソースが最大限に活かされなければなりません。Wwiseのインタラクティブミュージック機能は、このような課題の全てに対応できるツールを提供します。

Wwiseを使って、ゲームで再生するミュージックを、その時に起きているアクションに結びつけることで、ゲームのミュージックスコアにインタラクティブな側面を加えることができます。ビジュアル的に分かりやすいWwiseのインターフェースは、ミュージックピースを簡単にアレンジして編集できるように、またゲーム中にいつ、どのように再生されるのかを設定しやすいように、工夫されています。また、Wwiseのインタラクティブミュージック機能は、様々な方法で、リソースを節約します。少ない数のミュージックピースを、何時間にもおよぶゲームプレイ用に引き延ばして、プロジェクトの予算を節約できます。また、インタラクティブミュージック機能を使えば、今までコンポーザーなどの音楽プロが何時間もかけて行った作曲の単純作業を減らし、人的リソースも節約できます。この機能は、最終的なプロダクトに対しても、ミュージックの再生中に、再生時間、頻度、プロパティなどをコントロールできるので、ゲーム機のリソース管理にも役立ちます。

## インタラクティブミュージックを理解する

優れたインタラクティブミュージックは、ゲームプレイに適宜対応する、変化に富んだ意味のあるミュージックスコアを提供するので、ゲームエクスペリエンスが強化されます。例えば、格闘中はテンポを早くして盛り上げ、平和的な状況では落ち着いた軽いミュージックにできます。また、特別なイベントを、ステインガー、つまり既存の音楽に重ねて再生する短いミュージックを使い、強調できます。

また、巧みに設定されたインタラクティブミュージックのプロジェクトでは、貴重なリソースが最大限に活用されます。一般的に、ビデオゲームのミュージックスコアに対するソースミュージックの割合は大変小さく、作曲した1時間のミュージックを、ゲームプレイ30時間もたせることも、珍しくありません。Wwiseを利用すれば、効率性のニーズに対応しつつ多様性も提供できるように、再生方法を工夫できます。個別のミュージックピースを順番に、またはランダムな順で再生したり、ミュージックピースを構成するトラックを個別に再生したりすることもできます。このようにして、実際のミュージック素材が短くても、その何倍もの長さの、絶えず展開する数時間にもおよぶミュージックが、ゲームに提供されているかのような錯覚が起きます。

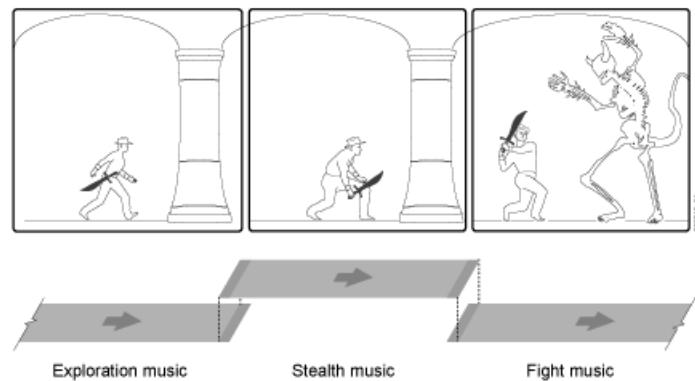
### インタラクティブミュージックの活用例

例えば、アドベンチャー系パズルゲームの作成を考えます。プレイヤーは、さつそうとした考古学者で、宝や冒険を求めて、古い寺院を探します。

ゲーム中にプレイヤーが行うアクションは主に3つあり、寺の中を探検し、罠を避け、寺の守護神達と戦います。まず、それぞれのアクションに対応するゲームStateとして、「Exploring (探検)」、「Sneaking (忍び足)」、「Fighting (ファイト)」をWwiseで定義します。これらのStateを利用して、ゲーム中の場面ごとに

Wwiseが再生するミュージックを決めることで、アクションに適したサウンドトラックを確実に再生します。

インタラクティブミュージック機能で達成できるエフェクトの様子を、下図に示します。



インタラクティブミュージックを使って、ゲームミュージックを長く、おもしろみのあるものにすることもできます。何しろ、もしゲームで探索できる場面が美しくエキゾチックであれば、何時間もかけて探りたいと思うかもしれません。それなのに考古学者の探検に伴うミュージックに繰り返しが感じられると、がっかりして、不満に思うでしょう。インタラクティブミュージックを使えば、限りあるソースの素材を、延長できるのです。

総合的にみて、インタラクティブミュージック機能は、時間潰し用のシンプルなゲームを、プレイヤーがとりこになるようなレベルまで引き上げるための、多彩な可能性を秘めたツールといえます。プレイヤーは、考古学者キャラクターの興奮と緊張感を体感するうちに、ゲームの熟考された罠やパズルを探るために、もう少し時間を費やすことになるかもしれません。

### インタラクティブミュージックのプロジェクト構造

Wwiseでは、非常に柔軟にインタラクティブミュージックを作成することができます。インタラクティブミュージック用のオブジェクトを組み合わせてゲーム用のスコアを作る方法は、ほぼ無数にあります。しかし、ある程度決まった構造に従つた方がワークフローとして効率的です。インタラクティブミュージックを使ったプロジェクトに採用できる、2つの基本的なプロジェクト構造を以下に示します。

- **垂直（縦型）のプロジェクト構造:** これは、1つのMusic Segmentに含まれる複数のトラックをシャッフルして、ゲームのミュージックスコアを変える構造です。音楽制作の現場で行われるトラックのミキシングと似ています。いくつかの長い、マルチトラックのセグメントを元に、変化あるミュージックスコアを作ることができます。
- **水平（横型）のプロジェクト構造:** これは、再生するセグメントを随時変えて、ゲームのミュージックスコアを変化させる構造です。複数の短い単独セグメントを、Interactive Music Hierarchyでアレンジする方法であり、Actor-Mixer Hierarchyでオブジェクトをアレンジする方法と似ています。これによって、ゲーム機に対する負荷を最小限に抑え、限定された数の短い音楽セグメントから、魅力的なミュージックスコアを作り出せます。

この2つの構造を組み合わせて、プロジェクトに与えられたリソースを効率的に使うのが、一般的な手法です。優れた構造であれば、際立つ音楽を生み出しながら、ゲーム機のリソースを上手に活用できます。

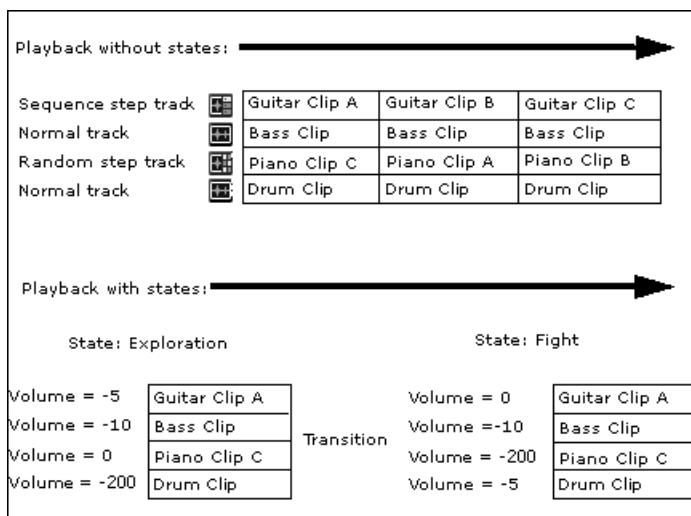
プロジェクト構造を計画するための他のアイディアについては「[インタラクティブミュージックに関する、コツとベストプラクティス](#)」を参照してください。

## インタラクティブミュージックに関する、コツとベストプラクティス

インタラクティブミュージックは複雑なツールで、選択肢も多様です。プロジェクトの初期段階で、インタラクティブミュージックに関する一貫した分かりやすい方針を立てることは、以後の時間節約や効率性につながります。インタラクティブミュージックプロジェクトの進め方には様々な手法があり、ゲームで最高の結果を達成するために、Wwiseの使い方を自由に工夫してください。インタラクティブミュージックの実装に関する提案を、以下に紹介します。

### 垂直（縦型）のプロジェクト構造の実装

垂直（縦型）のプロジェクト構造とは、複数のトラックからなるセグメントを、少數使うもので、それらのセグメントの繰り返しと切り替わりを利用します。基本的な垂直（縦型）プロジェクト構造の例を、以下に示します。



この例では、インタラクティブミュージックの基盤として、4トラックからなる Music Segmentを1つ、使っています。図の上半分では、このセグメントが、Stateを指定しない状態で再生されます。4トラックのうち、「Bass Clip」と「Drum Clip」の2トラックは「Normal track」設定なので、常に同じクリップが再生されます。一方、「Guitar」のトラックは「Sequence step track」設定なので、このトラックに入っている複数のクリップは、事前に指定した順番 (Sequence) に従い再生されます。また、「Piano」のトラックは「Random step track」設定なので、このトラックの複数のクリップはランダム (Random) な順番で再生されます。「Sequence step」や「Random step」の場合、変化のあるトラックをもたらす鍵となるのが、元のトラックに入っている1つのクリップの代わりとして使える複数のクリップが入った、複数のサブトラックを作成することです。

図の下半分の例では、Stateの変化に反応するように、4トラックがそれぞれ、調整されています。ゲームのStateが比較的静かな「Exploration（探検）」の時は、「Drum」が聞こえません。その代わりに、「Piano」のトラック（様々なサブトラックからなるもの）が再生されます。次に、動きのある「Fight」にStateが変わると、「Piano」トラックのボリュームが下げられ聞こえなくなり、「Drum」トラックのボリュームが上がります。同時に、「Guitar」ボリュームも上がります。全体的に、ミュージックがゲームのアクションに合わせて切り替わる結果となります。

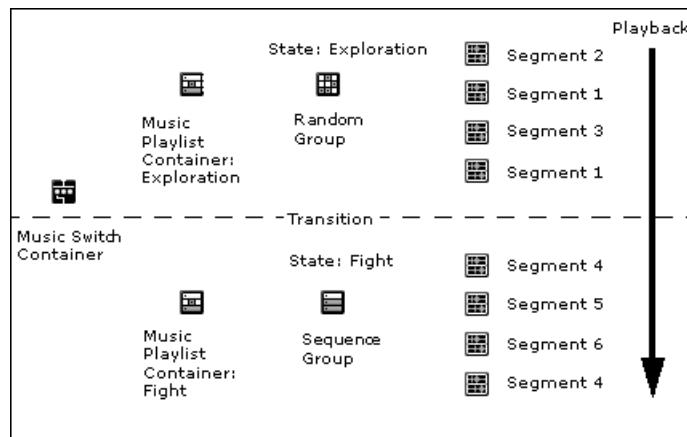
垂直（縦型）プロジェクトでは、再生するトラックを変化させる方法として、以下を利用できます。

- Stateの変化
- Random stepサブトラック
- Sequence stepサブトラック

垂直（縦型）アプローチは特に、少数の複雑なミュージックピースを使つた、State間の切り替わりが比較的少ないプロジェクトにむいています。

### 水平（横型）のプロジェクト構造の実装

水平（横型）プロジェクトは、複数の短いセグメントをアレンジした階層構造に基づいています。基本的な水平プロジェクト構造の例を、以下に示します。



この例では、「Exploration」と「Fight」の2種類のゲームStateを表すMusic Playlist Containerがあり、それぞれに、Stateに対応する複数のMusic Segmentが入っています。1つ目のコンテナの4つのセグメントは、「Random Group」としてアレンジされているので、ランダム(Random)な順番で再生されます。2つ目のコンテナの4つのセグメントは、「Sequence Group」としてアレンジされているので、事前に設定された順番(Sequence)に従い再生されます。Stateが「Exploration」から「Fight」に切り替わると、コンテナ「Exploration」の、再生中の最後のセグメントが、コンテナ「Fight」の最初のセグメントに、トランジションします。この結果、ミュージックの種類が完全に切り替わります。

水平（横型）プロジェクトでは、再生するセグメントを変化させる方法として、以下を利用します。

- Stateの変化
- Random Container
- Sequence Containers.

水平（横型）アプローチは、比較的シンプルで短いミュージックピースを多く用いるプロジェクトに推奨され、特にStateが頻繁に変化するプロジェクトに向いています。

## 2つのプロジェクト構造の併用

Wwiseを使ったプロジェクトの大半では、水平（横型）と垂直（縦型）の2種類の構造を組み合わせています。どちらの構造をどの程度採用するかは、プロジェクトに合わせて決めてください。例えば、ゲームのStateごとに異なるセグメントを1つずつ作成して（水平）、その中で再生するトラックを、ランダムなサブトラックを使って変化させることができます（垂直）。

## 複数のレベルにある、複数のMusic Switch Containerの使用

1つのMusic Switch Containerを、複数のStateやState GroupやSwitch Groupに結びつけることができます。ゲームから来る様々な入力にミュージックが依存するようなシナリオでは、Music Switch Containerのカスケード階層を作成することは避け、全ての入力を1つのMusic Switch Containerに入れると良いでしょう。ミュージックスイッチコンテナの中身は、仮想フォルダで整理でき、関連付けを設定したシステムなので、様々な関係の中で、複数のステート値やスイッチ値から、コンテナの中身を再利用できます。

トランジションが、異なる複数のグループで設定されている状態では、どのグループにもトランジションの漏れがないようなルールがあり、相互の優劣関係が決らないので、複数のSwitch Containerを様々な階層レベルで設定すると、各トランジションの優先順位を指定することが不可能とに注意してください。その結果、ミュージックトランジションがランダムに感じられ、全ての可能なトランジションを把握することが困難となります。

---

## 第23章 Interactive Music Hierarchyの構築

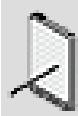
概要 .....	446
Music Segmentとは .....	446
コンテナの種類 .....	448
Interactive Music Hierarchyの、プロパティについて .....	449
Interactive Music Hierarchyに、オブジェクトを追加する .....	450
親オブジェクトの追加 .....	452
Interactive Music Hierarchyの、ミュージックオブジェクトの管理 .....	453
Interactive Music Hierarchyの構築に関する、コツとベストプラクティス ....	454

## 概要

ゲーム向けにミュージックスコアを準備する際に、多数の音楽系アセットを管理する必要がありますが、アセットをいくつかのグループにまとめて整理することで、作業がしやすくなり、大変便利です。各種オブジェクトを、アクターミキサー階層の中でグループにまとめて親子の関係を成立させたのと同様に、プロジェクトで使う全てのミュージックアセットを、インタラクティブミュージック階層の中で整理すると、効率的です。この構造を利用すれば、ミュージックアセットを整理できるだけでなく、設定したグループごとに、プロパティや動作を定義することもできます。

インタラクティブミュージックプロジェクトの様々なオーディオアセットをまとめ、その構造を作り上げるために、以下に示す数種類のミュージックオブジェクトを組み合わせることができます。

- ・ [「Music Segmentとは」](#)：ミュージッククリップが入った1つ以上のミュージックトラック。1つのセグメントの中のミュージッククリップの位置をそろえたり、アレンジしたりできる。
- ・ [「コンテナの種類」](#)：Music Segmentや他のミュージックコンテナなどを再生設定によってまとめたグループ。ミュージックコンテナには、以下の2種類がある。
  - ・ Music Switch Containerは、呼び出されたSwitchまたはStateに従い、ミュージックオブジェクトを再生する。
  - ・ Music Playlist Containerは、ランダムに、または指定した順に、再生する。



### 注記

ミュージックオブジェクトをまとめるために、仮想フォルダを使うことも可能ですが、他のミュージックオブジェクトと異なり、仮想フォルダにプロパティや動作を設定できません。

## Music Segmentとは

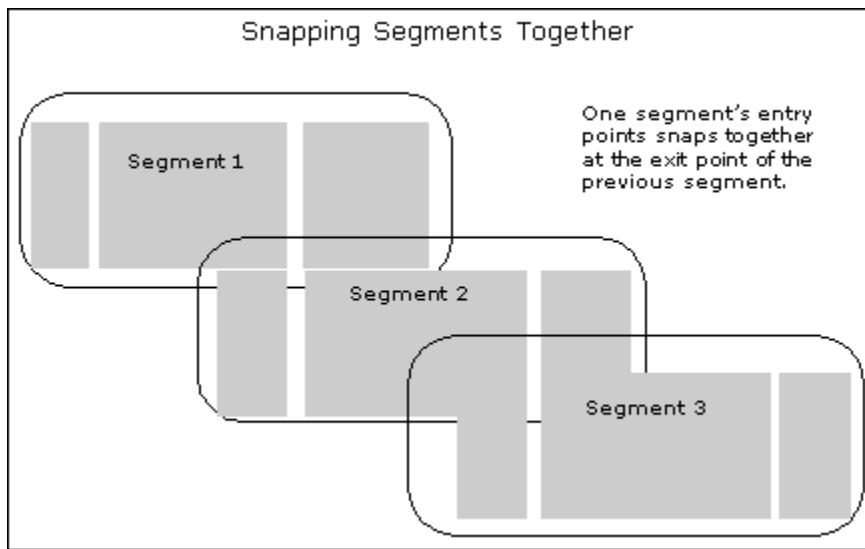
WwiseのMusic Segmentは、Interactive Music Hierarchyの基盤にある、核となるミュージックオブジェクトです。Actor-Mixer Hierarchyのサウンドオブジェクトと同様に、ミュージックプロジェクトで使う様々なオーディオアセットが入っています。ただし、サウンドオブジェクトとMusic Segmentでは、いくつかの重要な違いがあります。

- ・ あるMusic Segmentを、他のセグメントと位置をそろえることができる。
- ・ Music Segmentとミュージックソースの間には、ミュージックトラックという別のレイヤが入る。

### Music Segmentの位置をそろえる

スナップ (Snap) 機能を使って、2つのMusic Segmentを既定のポイントでつなぎ合わせて、ミュージックスコアをシームレスにアレンジできます。複数のセグメントをアレンジして、シンクポイント、つまりキュ (Cue) を使ってスナップ機能でつなげる様子を、下図に示します。キュとは、セグメントのエントリポイント (Entry point) やエグジットポイント (Exit point) などのキーポイントを表示するマーカーのこと、セグメントに付加されます。キューや、テンポ (Tempo)

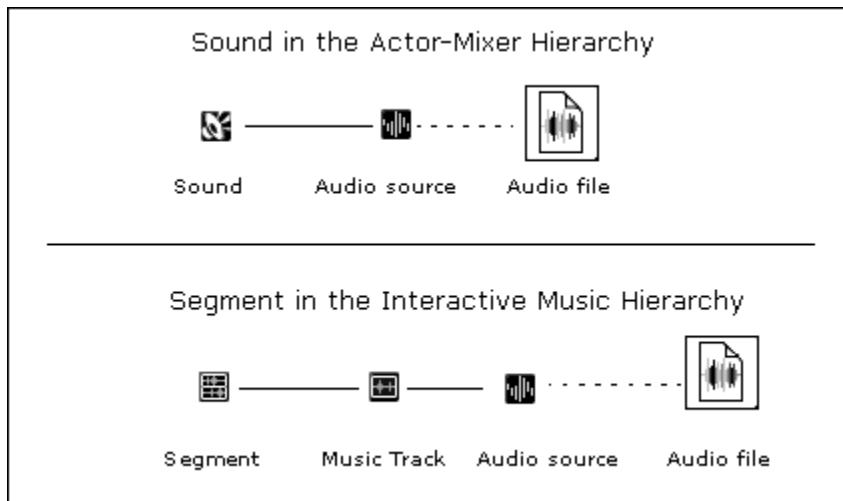
と拍子記号 (Time signature) の時間設定を使って、プレイリストにある複数のセグメントを、必要に応じて適切にそろえることができます。



### Music SegmentとMusic Track

一般的に、Music Segmentは、1つ以上のトラックで編成され、これがミュージッククリップの作業に使うレイヤとなります。1つのミュージッククリップは、1つのオーディオソースに対応し、そのオーディオソースに該当するオーディオファイルを、各種ゲームプラットフォーム向けにコンバージョンする時のコンバージョン設定が、オーディオソースに入っています。ファイルのコンバージョンを行うと、オリジナルのミュージックファイルは全てそのままの状態で残り、ソースの新しいバージョンが、プラットフォームごとに作成されます。この方法で、プラットフォームに最適化されたインタラクティブミュージックの編成が可能となります。[複数のプラットフォームに対応するオーサリング](#)についても、[複数のプラットフォームに対応するオーサリング](#)を参照してください。

オーディオファイルとサウンドオブジェクトの関係、そしてオーディオファイルとMusic Segmentの関係を、下図に示します。



サウンドオブジェクトに、他のファイルにリンクしている、様々なオーディオソースを追加して、どのオーディオソースを再生するかを選択できます。ただし、ミュージックトラックにオーディオソースを追加すると、Music Segment Editorでそのトラックにクリップが1つ、追加されます。

インターフェース上で分かりやすいように、Music Segmentやミュージックトラックは、以下のアイコンで表されます。

アイコン	内容
	Music Track
	Music Segment

## コンテナの種類

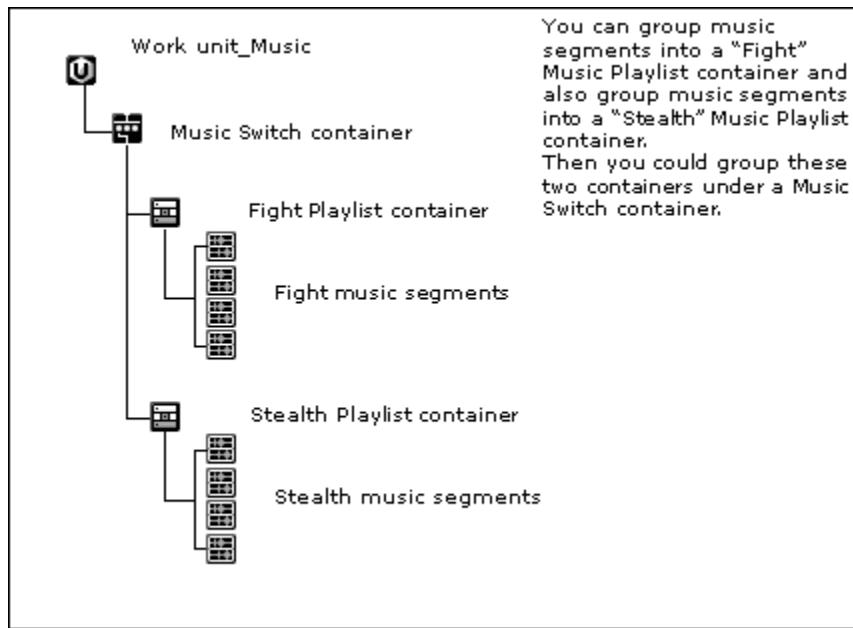
ゲームのミュージックスコアを最大限に活かすには、再生方法が異なるコンテナが要求されます。Interactive Music Hierarchyで設定を定義できるコンテナは、ミュージックSwitchと、ミュージックプレイリストの、2種類です。

Interactive Music Hierarchyで利用できるコンテナの種類を、下表に示します。

アイコン	内容
	Music Playlist Container
	Music Switch Container

## ミュージックアセットの整理方法の例

例えば、ファーストパーソンのアクション系アドベンチャーゲーム向けに、インタラクティブミュージックを作成する場合を考えます。ミュージックとして、「Stealth (ステルス)」用と「Fight (ファイト)」用が必要だと、既に分かっています。そこで、Switch Containerにミュージックオブジェクトをまとめて入れます。このようにすれば、子であるオブジェクトのオブジェクトプロパティを管理できます。Switch Containerの中に、2つのMusic Playlist Containerを追加して、Playlist Containerの中には、そのコンテナの様々なSwitchやStateにアサインされたミュージックオブジェクトを入れます。



ミュージックの階層は、プロジェクトの初期段階で、ゲームデザインを元に計画できます。その時に考慮すべきプロジェクトの他のエレメントとして、Work Unit、ルーティング、ゲームシンクなどがあります。プロジェクト全体を意識することで、様々なオブジェクトを効率的にグループ分けすることができます。

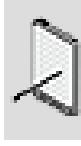
ルーティング構造の計画、Work Unitの構築、ゲームシンクの作成などについては、以下のセクションを参照してください。

- [6章Workgroupの活用](#)
- [「概要」](#)
- [17章Stateの活用](#)
- [20章Triggerの使い方](#)
- [18章Switchの使い方](#)
- [19章RPCの使い方](#)

## Interactive Music Hierarchyの、プロパティについて

Actor-Mixer Hierarchyと、Interactive Music Hierarchyでは、プロパティを同じように管理します。Wwiseでは、ミュージックオブジェクトのプロパティが、以下の2つのカテゴリーに分類されます。

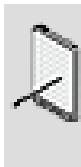
- **Relative properties:** 階層のレベルごとに設定する累積されるプロパティのこと、ボリュームやローパスフィルタに使う。相対プロパティの数値の合計が、最終的なプロパティ値となる。なお、全ての相対プロパティは、累計値が最小リミット、または最大リミットに達した時点で制限されることに注意すること。プロパティの最小値と最大値を、以下に示す。
  - Volume: (-200, +200) 単位dB
  - Low-pass filter: (0, 100) 単位%
  - High-pass filter: (0, 100) 単位%



### 注記

どのプラットフォームにおいても、Interactive Music Hierarchyでは、オブジェクトのピッチは考慮されません。

- **Absolute properties**：例えば、アウトプットルーティングなど、階層のあるレベル、一般的には最上位で定義されるプロパティで、その下の子オブジェクトに受け継がれる。ただし、絶対プロパティを階層中のどのレベルでもオーバーライドすることが可能。



### 注記

ミュージックファイルなどのサウンドオブジェクトから、モーションデータを生成できます。既存のサウンドからモーションを生成する方法については「[既存のサウンドから、モーションを生成](#)」を参照してください。

サウンドオブジェクトと同様に、ミュージックオブジェクトにもプロパティインジケータ（表示）があり、そのプロパティ値が他のプラットフォームにリンク（Link）しているのか、そのプロパティ値がRPCでGame Parameterに結び付けているのか、そのプロパティ値にRandomizerが適用されているのか、などを示します。

プロパティ値のリンク・アンリンク、RPCの使用、プロパティ値のランダム化などについては、以下のセクションを参照してください。

- 「[プラットフォームごとに、オブジェクトプロパティをカスタマイズする](#)」
- 「[Game Parameterを使って、プロパティ値をコントロールする](#)」
- 「[ランダム化したプロパティ値で、再生を強化](#)」

## Interactive Music Hierarchyに、オブジェクトを追加する

インタラクティブミュージック階層には、プロジェクトのミュージックオブジェクトを自由に整理して、完全なミュージックスコアをプロデュースできるような柔軟性があります。階層構築のスタート地点は、Work Unitです。1人で作業する場合は、Default Work Unitから直接、階層を作成し始めることができますし、それ以外の場合は、いくつかのWork Unitを作成してから、Work Unitにミュージックオブジェクトを追加します。ワークユニットやワークグループについて[6章Workgroupの活用](#)を参照してください。

実際にミュージック構造を構築するには、以下のいずれかを行います。

- プロジェクト構造を設定してから、構造にオーディオファイルをインポートする。
- オーディオファイルをインポートしてから、プロジェクト構造を作成してオーディオファイルを整理する。

オーディオファイルのインポート作業や、そのファイルからInteractive Music Hierarchyの新規オブジェクトが作成される方法については[7章メディアファイルの活用](#)を参照してください。

## Project Explorerのツールバーで、子オブジェクトを作成するには:

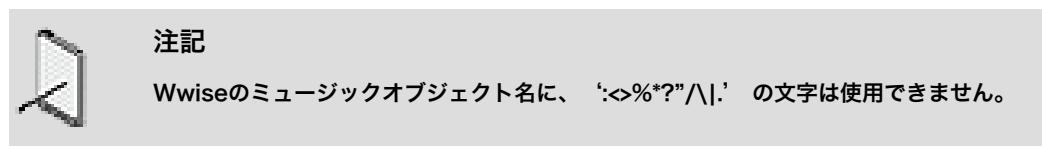
1. Project Explorerの、Audioタブで、作成するミュージックオブジェクトを入れるWork Unitを選択する。

Project Explorerのツールバーで、複数のアイコンが有効になる。

2. リストで、追加したいオブジェクトのアイコンをクリックする。

インタラクティブミュージック階層の選択したワークユニットの下に、オブジェクトが追加されます。

3. デフォルトのオブジェクト名を、内容が分かる名前に変更する。



続けて、この階層に他のミュージックオブジェクトも追加できます。作業前にオブジェクト同士の関係を充分に理解することで、適切に整理することができる。また、プロジェクトの後の段階において、時間の大幅な節約にもつながる。

## Interactive Music Hierarchyの中に、子オブジェクトを作成するには:

1. Project Explorerの、Audioタブの、Interactive Music Hierarchy のセクションで、作成するミュージックオブジェクトを入れるWork Unitを右クリックする。

2. ショートカットメニューで、New Childを選択する。

サブメニューに、追加できるオブジェクトのリストが表示される。

この階層レベルで追加できるオブジェクトの種類は、以下の通りである。

Virtual Folder

Music Switch Container

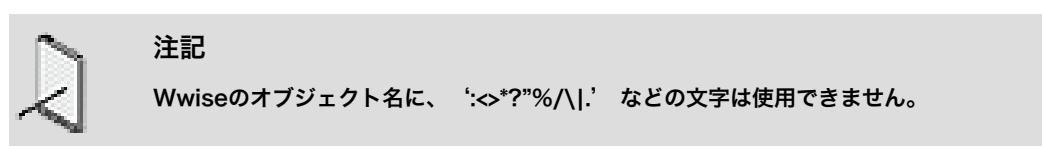
Music Playlist Container

Music Segment

3. リストで、追加したいオブジェクトを選択する。

Interactive Music Hierarchyに、このオブジェクトが追加される。

4. デフォルトのミュージックオブジェクト名を、内容が分かる名前に変更する。



これで、階層に他のオブジェクトを追加する準備ができる。作業前にオブジェクト同士の関係を充分に理解することで、適切に整理することができる。また、プロジェクトの後の段階において、時間の大規模な節約にもつながる。

## 関連トピック

- ・「親オブジェクトの追加」
- ・「Interactive Music Hierarchyの、ミュージックオブジェクトの管理」

## 親オブジェクトの追加

Work Unitの最初のオブジェクトを作成できたら、続いて他のオブジェクトをインタラクティブミュージック階層に追加して、相互に親子関係を持たせることができます。親オブジェクトは他のオブジェクトを含むので、親オブジェクトを作成してから、既存のオブジェクトを移動して中に入れることができます。親子関係を設定するメリットは、親のプロパティを変更して動作設定を変えると、下の子オブジェクトにも作用することです。Interactive Music Hierarchyのプロパティ設定については「Interactive Music Hierarchyの、プロパティについて」を参照してください。

**Project Explorerのツールバーで、親オブジェクトを作成するには:**

1. Project Explorerの、Audioタブの、インタラクティブミュージック階層セクションで、親オブジェクトを追加するミュージックオブジェクトを選択する。
2. Shiftキーを押して、選択したオブジェクトの親として追加できるミュージックオブジェクトのアイコンを表示させる。
3. リストで、追加したいオブジェクトアイコンをクリックする。

インタラクティブミュージック階層の選択したオブジェクトの親として、オブジェクトが追加されます。

4. デフォルトのミュージックオブジェクト名を、内容が分かる名前に変更する。



### 注記

Wwiseのオブジェクト名に、「:<>\*?%/\|.’などの文字は使用できません。

**親オブジェクトを作成するには:**

1. Project Explorerの、Audioタブの、インタラクティブミュージック階層セクションで、親を作成するオブジェクトを右クリックする。
2. ショートカットメニューでNew Parentを選択する。

サブメニューに追加できるオブジェクトのリストが表示される。

選択したオブジェクトの階層レベルによって、以下を新しい親オブジェクトとして追加できる。

Virtual Folder

Music Switch Container

Music Playlist Container

Music Segment

- リストで、追加したいオブジェクトをクリックする。

Interactive Music Hierarchyに新しい親オブジェクトが追加され、そのオブジェクトの種類に従いデフォルト名が付けられる。

- デフォルトのミュージックオブジェクト名を、内容が分かる名前に変更する。



#### 注記

Wwiseのオブジェクト名に、「:<>\*?%/\|.’などの文字は使用できません。

## 関連トピック

- 「Interactive Music Hierarchyの、ミュージックオブジェクトの管理」
- 「Interactive Music Hierarchyに、オブジェクトを追加する」

## Interactive Music Hierarchyの、ミュージックオブジェクトの管理

インタラクティブミュージック階層では、ショートカットメニューとWindowsの標準ショートカットを使って、オブジェクトの名前変更、カット、コピー、ペーストなどができます。なお、階層の構造を変える際は、以下に注意してください。

### ミュージックオブジェクトの移動

Action	Results
ミュージックオブジェクトの移動	階層内でオブジェクトの場所を変えると、移動先の親のプロパティや動作が、そのオブジェクトに適用される。
	イベントと結び付いたオブジェクトを移動すると、そのオブジェクトのイベントへの結び付きが維持される。

### ミュージックオブジェクトのコピー、ペースト

Action	Results
ミュージックオブジェクトのコピー	オブジェクトをコピーして新しい場所にペーストすると、ペースト先の親のプロパティや動作が、そのオブジェクトに適用される。また、その子オブジェクトも、一緒にコピーされる。
	イベントと結び付いたオブジェクトをコピーすると、新しいオブジェクトは、イベントへの結び付きがない。

### ミュージックオブジェクトのカット、削除

Action	Results
ミュージックオブジェクトのカット、削除	ミュージックオブジェクトをカットまたは削除すると、その子オブジェクトも削除される。

Action	Results
	<p>関連するコンバージョン済みオーディオファイルは、削除されない。関連するオブジェクトが存在しない、コンバージョン済みのオーディオファイルは、オーファン（Orphan / 孤立）ファイルと呼ばれる。オーファンファイルを削除するには、オーディオキャッシュのクリアが必要となる。オーファンオーディオファイルの削除については「<a href="#">キャッシュクリア</a>」を参照してください。</p> <p>イベントと結びついたオブジェクトを削除またはカットすると、そのイベントのオブジェクトが1つ不足する結果となる。</p>

## 関連トピック

- 「Interactive Music Hierarchyに、オブジェクトを追加する」
- 「ミュージックオブジェクトの時間設定」
- 「Music Playlist Containerの、再生方法の設定」
- 「Music Switch Containerの中身と、再生方法の設定」

## Interactive Music Hierarchyの構築に関する、コツとベストプラクティス

インタラクティブミュージックは複雑なツールで、選択肢も多様です。プロジェクトの初期段階で、インタラクティブミュージックの階層構築に関する一貫した分かりやすい方針を立てることは、以後の時間節約や効率性につながります。インタラクティブミュージックプロジェクトの進め方には様々な手法があり、ゲームで最高の結果を達成するために、Wwiseの使い方を自由に工夫してください。インタラクティブミュージックを最大限に活かすための、オブジェクトのグループへのまとめ方を、以下に紹介します。

### Interactive Music Hierarchyで、オブジェクトをグループにまとめる

ミュージック階層の構築を始める前に、オーサリング時間を短縮するだけでなく、プロジェクトのメモリ負荷を抑えるためにも、オブジェクトの最適な整理方法を充分に検討する必要があります。

メモリ消費を最適化するには、プロパティ設定を階層の上位で行い、そのグループ全体で共有できるように計画してください。共有できるプロパティの種類を、以下に示します。

- Positioning
- RTPCs
- States
- Randomizers

---

## 第24章 ミュージックオブジェクトの再生方法の設定

概要 .....	456
ミュージックオブジェクトの時間設定 .....	456
Music Playlist Containerの、再生方法の設定 .....	457
Music Switch Containerの中身と、再生方法の設定 .....	460
インターラクティブミュージックの再生に関する、コツとベストプラクティス .	
464	

## 概要

ミュージックの再生方法は、ゲームの状況に応じて変える必要があります。様々なシナリオに対応するために、WwiseではMusic Segmentやミュージックトラックの再生方法を、個別に設定することができます。また、ミュージックの再生をさらに自由に幅広くコントロールするために、オブジェクトをまとめるコンテナとして、ミュージックプレイリスト (Music playlist) とMusic Switch Containerの2種類のコンテナがあります。

ここで使うコンテナの種類によって、中に入れたオブジェクトの再生方法が決まります。例えば、Music Playlist Containerでは、コンテナ内のオブジェクトが具体的なプレイリストに従って再生されます。プレイリスト内にいくつかのグループを設定して、それらをランダムに再生させたり、順番に再生させたりすることができます。一方、Switch Containerは、コンテナ内のオブジェクトを、ゲーム側のSwitchまたはStateに基づいて再生します。

種類の異なるコンテナを組み合わせて、異なる再生方法を取り入れれば、多様な再生シナリオを創出できるので、同じものの繰り返しが減り、最終的には、ゲームのミュージックが強化されます。

## ミュージックオブジェクトの時間設定

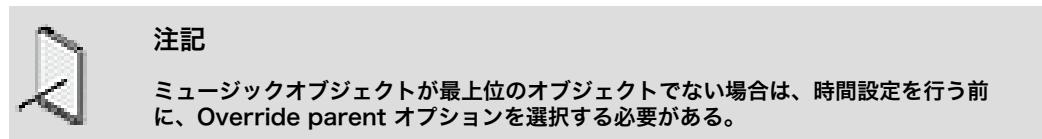
ゲームのインターラクティブミュージックを作成する時に、ミュージックの切り替わりのタイミングを、再生中のミュージックのテンポ (Tempo) や拍子 (Meter) に合わせると、非常に便利なこともあります。ミュージックオブジェクトの時間設定 (Time settings) を使って、Wwiseにインポートした元のミュージックの、テンポやメーターを指定できます。これらの設定は、Wwise上でミュージックオブジェクトをマーク付けするのに役立ち、ミュージックのシンクポイントやトランジションポイントが、定義しやすくなります。

テンポや拍子記号 (Time signature) 以外に、Music Segmentを自由な位置でバーチャルに分割する、グリッド設定も使えます。このように分割レベルをさらに細かくすることで、ミュージックのトランジション、State変更、ステインガーなどのタイミングを決めるシンクポイント (Sync points) を、かなり細かく指定できます。

時間設定を定義できる対象は、セグメント、Music Playlist Container、Music Switch Containerです。

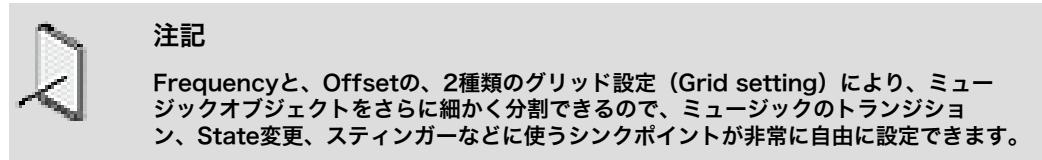
### ミュージックオブジェクトの時間設定を設定するには:

- Property Editorに、最上位のミュージックオブジェクトを1つロードする。



- テキストボックスTempoで、元となるミュージックのテンポを、BPM (Beats per Minute) 単位で指定する。

3. テキストボックスTime Signatureで、元となるミュージックの、1小節(Bar)のビート数と長さを指定する。
4. Frequencyリストで、このMusic Segmentを分割するもう1つの方式として、4小節ごと(Every 4 bars)、ビート(Beat)ごと、全音符(Whole note)ごとなどを選択する。
5. 周波数値(Frequency value)に対するオフセット(Offset)を作成する場合は、リストから既定のオプションを選択するか、Customを選択して、テキストボックスOffsetに、オフセット値をミリ秒単位で入力し、カスタムオフセットを設定する。



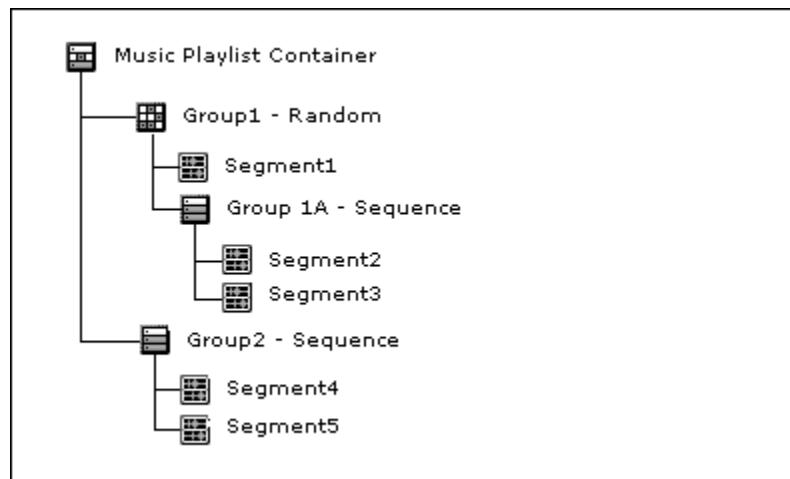
## 関連トピック

- ・「Music Trackの再生方法の設定」
- ・「Music Playlist Containerの、再生方法の設定」
- ・「Music Switch Containerの中身と、再生方法の設定」

## Music Playlist Containerの、再生方法の設定

制限された数のミュージックをプロジェクトで最大限に活かすには、Music Segmentを分類して、いくつかのMusic Playlist Containerに分けることができます。これらのミュージックオブジェクトは非常に自由に使い回せるので、様々なMusic Segmentを、多様な方法で分けられます。再生モード(Playback mode)や再生タイプ(Playback type)を使い分けて、グループごとに再生方法を変えられます。再生タイプで、そのグループをランダムに再生するのか、順番に再生するのかを決めます。再生モードで、そのグループが再生される時に、グループ内のいくつのセグメントを再生するのかを決めます。

あるプロジェクトで、いくつかのMusic Segmentを、Music Playlist Containerを使って分類した例を、下図に示します。



Music Playlist Containerの利用には、以下のタスクが伴います。

- ・「Music Playlist Container内の、グループの作成」
- ・「Music Playlist Container内のグループの、再生方法の設定」
- ・「ミュージックプレイリストの中身の設定」

### Music Playlist Container内の、グループの作成

Music Playlist Containerの中に、いくつかのグループ (Group) に分けたMusic Segmentを入れることで、複雑なミュージック構造をつくり出すことができます。必要な数のグループを追加してから、グループの配置を変えたり、グループの削除、カット、コピー、ペーストを行ったり、中にMusic Segmentを入れたりします。Music Playlist Containerの中身の入れ方については「[ミュージックプレイリストの中身の設定](#)」を参照してください。

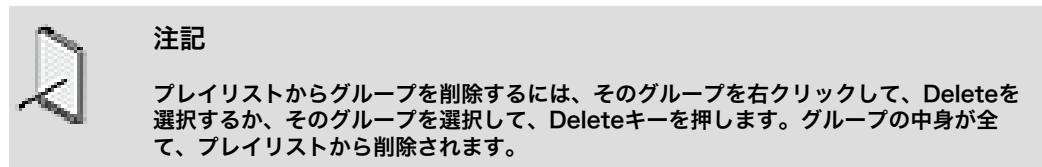
#### Music Playlist Container内に、グループ (Group) を作成するには:

1. Property Editorに、Music Playlist Containerを1つロードする。

Music Playlist Editorで、最上位グループが1つ、自動的に作成される。

2. プレイリスト階層に新しいグループを追加するには、New Groupをクリックする。

階層内で選択した親オブジェクトの下に、新規グループが追加される。



3. 必要に応じて、グループをさらにプレイリストに追加する。

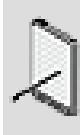
プレイリストにある様々なグループの再生方法は、どの時点でも設定できます。

#### 関連トピック

- ・「Music Playlist Container内のグループの、再生方法の設定」
- ・「ミュージックプレイリストの中身の設定」
- ・「Music Playlist Containerの、再生方法の設定」

### Music Playlist Container内のグループの、再生方法の設定

Music Playlist Containerに入っているグループごとに、再生モード2種類と、再生タイプ2種類から、選択します。再生タイプで、そのグループをランダムに再生するのか、順番に再生するのかを決めます。再生モードで、そのグループが再生される時に、グループ内のいくつのセグメントを再生するのかを決めます。



#### 注記

あるグループに対して選択できる再生オプションは、そのグループで設定した再生モードや再生タイプによって変わります。

### Music Playlist Container内にRandom groupを設定するには:

1. Property Editorに、Music Playlist Containerを1つロードする。

Music Playlist Editorで、最上位グループが1つ、自動的に作成される。

2. Group/Segmentリストから、最上位グループの再生方法 (Playback behavior) を設定するために、以下のオプションの1つを選択する。

- **Random Continuous** - グループが再生されると、グループ内の全てのミュージックオブジェクトを、ランダムな順番で連続再生する。
- **Random Step** - グループが再生されると、毎回、グループ内のミュージックオブジェクトを1つだけ、再生する。

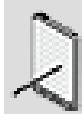
3. Random Typeリストから、以下のオプションを1つ選択する。

- **Standard**では、オブジェクト群が維持される。一度再生されたオブジェクトも、これから再生される可能性のあるオブジェクトのリストから外されないため、再び再生されることがある。

- **Shuffle**では、再生されたオブジェクトがオブジェクト群から外される。グループ内の全てのオブジェクトが再生されるまで、再び再生されない。

4. テキストボックスAvoid Repeatに、同じオブジェクトを再び再生するまでの間に、他のオブジェクトを何個、再生させるかを、入力する。

5. テキストボックスLoop Countで、グループ全体を何回再生するのか、またはステップを再生するのかを指定する。



#### 注記

ステップ再生のグループをループ再生する場合は、そのグループが再生された時に、毎回、再生するグループ内のオブジェクト数を定義します。

### Music Playlist Container内にSequence groupを設定するには:

1. Property Editorに、Music Playlist Containerを1つロードする。

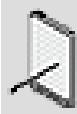
Music Playlist Editorで、最上位グループが1つ、自動的に作成される。

2. Group/Segmentリストから、最上位グループの再生方法 (Playback behavior) を設定するために、以下のオプションの1つを選択する。

**Sequence Continuous** - このグループが再生されると、毎回、グループ内の全てのミュージックオブジェクトを順番 (Sequence) に再生する。

**Sequence Step** - このグループが再生されると、毎回、グループ内のミュージックオブジェクトを1つだけ、再生する。再びグループが再生されると、グループ内にある次のミュージックオブジェクトを再生する。

3. テキストボックスLoop Countで、グループ全体を何回再生するのか、またはステップを再生するのかを指定する。



### 注記

ステップ再生のグループをループ再生する場合は、そのグループが再生された時に、毎回、再生するグループ内のオブジェクト数を定義します。例えば、あるシーケンスステップグループの中に6セグメントが入っていて、ループカウントが「2」であれば、そのグループが1回目に再生される時にセグメント1番と2番が再生され、2回目に再生される時にセグメント3番と4番が再生され、以後も続きます。

## 関連トピック

- ・[「Music Playlist Container内の、グループの作成」](#)
- ・[「ミュージックプレイリストの中身の設定」](#)
- ・[「Music Playlist Containerの、再生方法の設定」](#)

### ミュージックプレイリストの中身の設定

Actor-Mixer HierarchyのSequence Containerと同様に、Music Playlist Containerの中にある複数のセグメントから、そのプレイリストに追加するセグメントを選択して、再生の順番を決めます。1つのセグメントは必ず1つのグループに所属し、そのグループによって、所属するセグメントの再生方法（Playback behavior）が決まります。グループの再生方法の定義については[「Music Playlist Container内のグループの、再生方法の設定」](#)を参照してください。

### Music Playlist Containerの中身を設定するには

1. Property Editorに、Music Playlist Containerを1つロードする。
2. Contents Editorから、ミュージックプレイリストの中にあるグループに、セグメント（単数または複数）をドラッグして入れる。  
セグメントが、このグループに追加される。
3. グループの設定がランダムステップ（Random step）またはランダム連続（Random continuous）であれば、テキストボックスWeightを使い、グループ内の全てのミュージックオブジェクトに対して、1つずつ、ウェイト付けを行う。ウェイト付けは、ミュージックオブジェクトのプライオリティ（優先順位）を決める条件の1つである。
4. テキストボックスLoop Countで、セグメントを再生する回数を指定する。
5. 引き続き、Music Playlist Containerの中の様々なグループに、セグメントをドラッグして入れる。

## 関連トピック

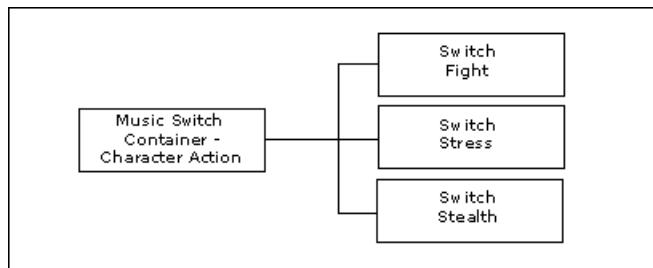
- ・[「Music Playlist Container内の、グループの作成」](#)
- ・[「Music Playlist Container内のグループの、再生方法の設定」](#)
- ・[「Music Playlist Containerの、再生方法の設定」](#)

## Music Switch Containerの中身と、再生方法の設定

Music Switch Containerを使って、ゲームのエレメントに対して発生する様々な結果に合わせて、ミュージックをグループにまとめることができます。選択

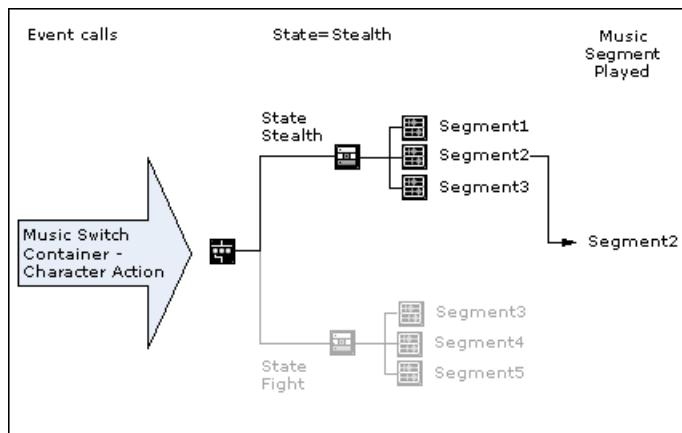
## ミュージックオブジェクトの再生方法の設定

肢1つに対して、SwitchまたはStateを1つ、Switch Containerに入れます。例えば、ゲームのメインキャラクター用に存在する、全てのアクションタイプを入れたMusic Switch Containerを作成できます。このコンテナに、ファイト場面(Fight)、ストレス状況(Stress)、キャラクターのステルスマード(Stealth)に対応するSwitch(Switch)を3つ、入れることができます。



1つのSwitchまたはStateの中に、その選択肢に関連した複数のミュージックオブジェクトを入れます。1つのSwitchまたはStateの中に、その選択肢に関連した複数のミュージックオブジェクトを入れます。ゲームがSwitch Containerを呼び出されると、Wwise側で、現在どのSwitchまたはStateが有効かを確認してから、どのコンテナまたはMusic Segmentを再生するかを判断します。

下図は、イベントによって、Music Switch Container「Character Action」が呼び出された時の流れを示します。このコンテナには、ゲーム中に発生する可能性のある、様々なキャラクターアクションごとに、ミュージックをまとめてあります。今回の例では、「Stealth」と「Fight」の2つのStateが入っています。下図では、イベントがMusic Switch Containerを呼び出した時に、キャラクターがステルスマード(つまりStateは「Stealth」)なので、ステルスマードに関連付けられているミュージックが再生されます。Music Playlist Containerに、「Stealth」Stateに該当する様々なミュージックを入れてあるので、キャラクターがステルスマードに入るたびに、ステルスマード用のミュージックの、異なるバリエーションが再生されます。



### Music Switch Containerを、ゲームシンクに関連付ける

Music Switch Containerは、State、Switch、またはGame Parameterを基に設定します。Music Switch ContainerをGame Parameterにリンクするには、Switch

Group Property Editorで、Switch GroupをGame Parameterにリンクする必要があります。RTPCをSwitchに結びつける方法については「[Switchに、Game Parameter値をマッピングする](#)」を参照してください。

Music Switch Containerを設定するには、まずSwitch GroupやStateやState Groupをアサインします。これで、ミュージックが反応するためのSwitch、State、またはRTPCが指定されます。

コンテナにStateやState GroupやSwitch Groupをアサインしたり、RTPCをSwitchとして使う前に、これらのStateやState Group、Switch Group、RTPCを作成する必要があります。作成方法については、以下を参照してください。

- [17章Stateの活用](#)
- [18章Switchの使い方](#)
- 「[Game Parameterの作成](#)」

## Music Switch Containerにゲームシンクを関連付けるには:

1. Music Editorに、Music Switch Containerを1つロードする。
2. 以下のいずれかを行う:
  - エディタの上部に表示された、[>>]ボタンをクリックし、ゲームシンクを1つ選択する。
  - Switch Groupオブジェクト、またはStateやState Groupオブジェクトを、Project Explorerから、エディタの上部にドラッグ＆ドロップする。

## 関連トピック

- 「[Music Switch Containerの中身と、再生方法の設定](#)」
- 「[ミュージックオブジェクトを、StateやSwitchに、関連付ける](#)」
- 「[ミュージックオブジェクトの時間設定](#)」

## Music Switch Containerのミュージックオブジェクトの再生方法の設定

あるミュージックを、部分的に何回も再利用しているのであれば、複数のSwitchまたはStateに、同じミュージックオブジェクトが入っていることもあります。Stateが切り替わった時に、もしその前後で同じミュージックオブジェクトが出てきたら、そのミュージックオブジェクトをどう再生するのかを、決めておく必要があります。ミュージックオブジェクトの再生を続けるのか、次のシンクポイントで再生を一旦停止して最初から再生し直すのかを、状況に応じて設定します。

下図では、「Stealth」Switchと「Stress」Switchの両方に、同じ子プレイリスト「S2\_Playlist」が設定されています。ゲーム中に、Switchが「Stealth」から「Stress」に切り替わった時や、その逆の時は、「S2\_Playlist」を再生し続けるのか、または再生中のMusic Segmentを次のシンクポイントで停止して、このプレイリストを最初から再生し直すのかを、事前に選択します。

## ミュージックオブジェクトの再生方法の設定

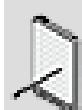
Switch/State	Child
<input checked="" type="checkbox"/> None	
<input checked="" type="checkbox"/> Fight	 F_Playlist
<input checked="" type="checkbox"/> Stealth	 S2_Playlist
<input checked="" type="checkbox"/> Stress	 S2_Playlist

Music Switch Containerのミュージックオブジェクトの再生方法を設定するには:

- Property Editorに、Music Switch Containerを1つロードする。
- グループボックスPlay Optionsで、以下のいずれかを行う。

**Continue to play on switch change** (Switch切り替え時に、再生を続ける) このオプションを選択すると、SwitchまたはStateが変わる時に、ソース (Source) とデスティネーション (Destination) の、両方のSwitchまたはStateにあるミュージックオブジェクトを、強制的に再生し続ける。

**Continue to play on switch change** (Switch切り替え時に、再生を続ける) このオプションの選択を外すと、SwitchまたはStateが変わる時に、ソース (Source) とデスティネーション (Destination) の、両方のSwitchまたはStateにあるミュージックオブジェクトの再生を、次のシンクポイント (Sync point) で一旦停止して、最初から再生し直す。



### 注記

トランジションをシームレスにするために、正確なシンクポイントを設定したり、フェードを適用したりできます。

## 関連トピック

- 「Music Switch Containerの中身と、再生方法の設定」
- 「Music Switch Containerを、ゲームシンクに関連付ける」
- 「ミュージックオブジェクトを、StateやSwitchに、関連付ける」
- 「ミュージックオブジェクトの時間設定」

## ミュージックオブジェクトを、StateやSwitchに、関連付ける

Music Switch Containerに、StateまたはSwitchのグループを1つ、アサインできたら、このグループの中にある複数のStateやSwitchに対して、様々なミュージックオブジェクトを設定できます。

ワイルドカードも含む可能性のある特定の組み合わせのStateもしくはSwitchは、パスとして参照されます。パスは、Music Switch Containerの子であるミュージックオブジェクトと関連付けられます。

### ミュージックオブジェクトをアサインするには:

1. Music Editorに、Music Switch Containerを1つロードする。

このコンテナのStateやState GroupやSwitch Groupが、Music Switch Containerの上部に表示される。

2. 上部に表示されたStateやState GroupやSwitch GroupのStateを、それぞれ1つずつを選択する。
3. **Add Path**ボタンをクリックし、エントリリストで、エントリを1つ、作成する。

作成したエントリが、オブジェクトをアサインされていない状態で、エントリリストに表示される。

4. このエントリにミュージックオブジェクトをアサインするには、このエントリの [...] ボタンをクリックする。

オブジェクトのブラウザが開き、このMusic Switch Containerの子が表示される。

5. オブジェクトを1つ選択して、OKをクリックする。

### ミュージックオブジェクトを、ドラッグ＆ドロップでアサインするには:

1. Music Editorに、Music Switch Containerを1つロードする。

このコンテナのStateやState GroupやSwitch Groupが、Music Switch Containerの上部に表示される。

2. Music Switch Containerの子を1つ、上部に表示されたのSwitchまたはStateの上にドラッグ＆ドロップして、この子との関連性を持たせる。

### 関連トピック

- ・「[Music Switch Containerの中身と、再生方法の設定](#)」
- ・「[Music Switch Containerを、ゲームシンクに関連付ける](#)」
- ・「[Music Switch Containerのミュージックオブジェクトの再生方法の設定](#)」
- ・「[ミュージックオブジェクトの時間設定](#)」

## インタラクティブミュージックの再生に関する、コツとベストプラクティス

インタラクティブミュージックは複雑なツールで、選択肢も多様です。プロジェクトの初期段階で、インタラクティブミュージックに関する一貫した分かりやすい方針を立てることは、以後の時間節約や効率性につながります。インタラクティブミュージックプロジェクトの進め方には様々な手法があり、ゲームで最高の結果を達成するために、Wwiseの使い方を自由に工夫してください。インタラクティブミュージックを最大限に活かすための提案と、再生中に発生するかもしれない問題に関するトラブルシューティング情報を、以下に紹介します。

## 再生に関するトラブルシューティング

ミュージックトラックの試聴中に、非同期やボイススターべーションなどの問題が発生した場合、ゲームに接続中であれば、Track Property Editorで、ミュージックトラックのルックアヘッドタイム (Look-ahead time) を新しく設定してみると、解決できるかもしれません。Wwise上で、ストリーミングしないトラックにおいてこのような問題が発生する場合は、Audio Preferencesで、ルックアヘッドタイムを調整してみてください。この設定値は、Wwise上でこれらのオブジェクトに適用する、デフォルトのルックアヘッドタイムを設定するものです。ルックアヘッドタイムのユーザープレファレンス設定については「[Music Trackのルックアヘッドタイムの設定](#)」を参照してください。

一般的に、同時に再生されるトラック数が多いほど、ボイススターべーションを防止してトラックの同期を維持するためには、長いルックアヘッドタイムが必要です。具体的に必要なルックアヘッドタイムの長さは、様々な要因によって変わり、圧縮フォーマットや、そのリクエスト時に使用される帯域幅の合計なども影響します。

## コンバージョンを、確実にサンプルアキュレートにするには

Wwiseで使用するサンプルコンバータは、以下の周波数のソースについて、サウンドの長さを毎分約12サンプル分、延長します。

- 44,100Hz
- 22,050Hz
- 11,025Hz

これはサウンドオブジェクトでは問題になりませんが、ミュージックオブジェクトの場合はエグジットマーカー (Exit marker) で再生が停止して、ファイルの最後まで再生されないこともあります。矛盾が生じるかもしれません。エラーを回避するために、ミュージックオブジェクトには、周波数が12,000、24,000、36,000、または48,000Hzのオーディオファイルの採用を推奨します。

---

## 第25章 Music Track、Music Segmentの活用

概要 .....	467
セグメントに、Music Trackを追加する .....	468
Music Trackの再生方法の設定 .....	469
トラックに、サブトラックを追加する .....	472
サブトラックをSwitchやStateに結びつける .....	472
トラックの中身を入れる .....	473
セグメントから、トラックやサブトラックを削除する .....	474
Music Editorの、目盛に合わせたスナップ機能 .....	474
クリップの使い方 .....	475
セグメントの試聴 .....	477
キューの使い方 .....	479

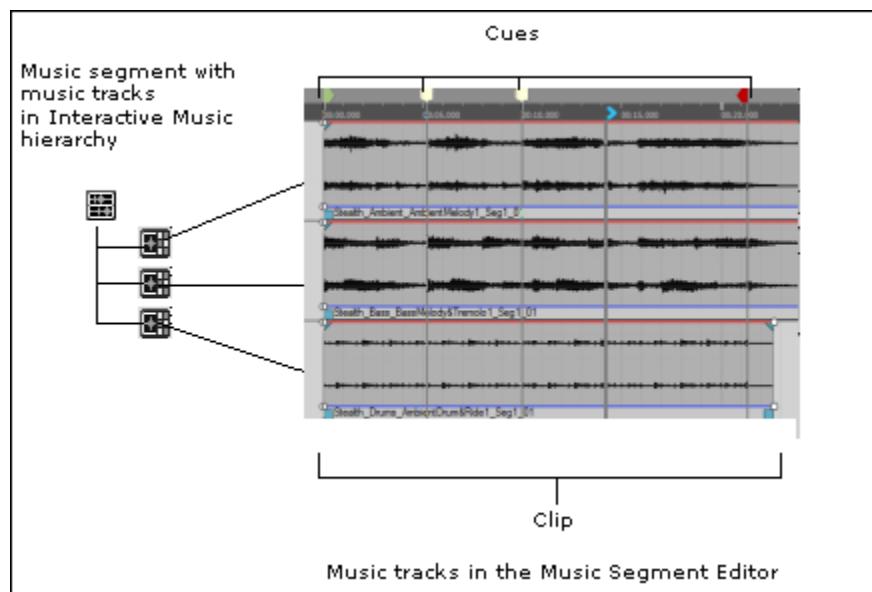
## 概要

Interactive Music Hierarchyの基本ユニットは、Music Segmentです。ゲームのミュージックトラックやサブトラックをアレンジして、レイヤを重ねたミュージックスコアを編成するために使う、マルチトラックシーケンサーに似ています。1つのミュージックトラックに入れるサブトラック数に制限はなく、それぞれのサブトラックに違うミュージッククリップが入ります。トラックごとに再生方法を変えることで、変化に富んだゲームミュージックが成立します。

インターフェース上で分かりやすいように、ミュージックトラックやMusic Segmentは、以下のアイコンで表されます。

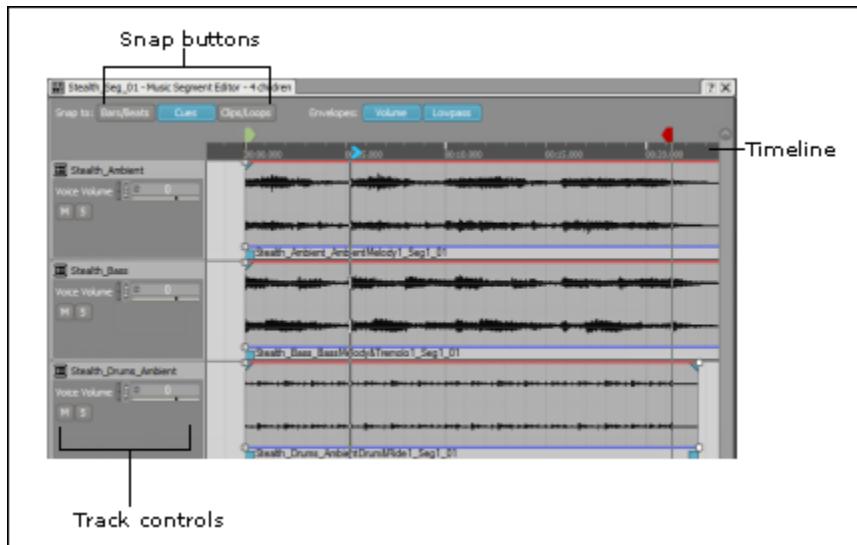
アイコン	内容
	Music Track: 親セグメントが再生されると、毎回、このトラックが再生される。
	Random Music Track: 親セグメントが再生されると、毎回、サブトラックがランダムな順番で再生される。
	Sequence Music Track: 親セグメントが再生されると、毎回、サブトラックが順番に再生される。
	Switch Music Track: 結びつけられたSwitch Groupの設定に従い、サブトラックが再生される。
	Music Segment: ミュージックトラックの親オブジェクト。

セグメントの中のミュージックトラックは全て、Music Segment Editorに表示されます。ミュージックトラックの中にはミュージッククリップがあり、画面で確認しながら編集できます。クリップ (Clip) は、それぞれのオーディオソースを表し、Music Segment Editorのタイムライン (Timeline / 時間軸) に沿って、波形で表示されます。クリップを操作したり、タイムラインの特定ポイントに合わせてキュー (Cue) をセグメントに追加したりできます。また、カスタムキューを設定して、いつプロパティ変更やトランジションを実行するのか、またはいつスティンガーを再生するのかを、指定できます。



Music Segment Editorは、以下の3エリアに分かれています。

- ・クリップやキーを、選択したエレメントに合わせるための、スナップボタン (Snap buttons)
- ・ミュージックトラックのプロパティ設定の変更や、トラックごとのミュート (Mute) や強制再生 (Force playback) の設定をする、トラックコントロール機能 (Track controls)
- ・クリップやキーを確認するための、セグメントのタイムライン (Timeline / 時間軸)



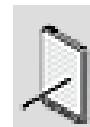
セグメントを使った作業には、以下のタスクが伴います:

- ・「セグメントに、Music Trackを追加する」
- ・「Music Trackの再生方法の設定」
- ・「トラックに、サブトラックを追加する」
- ・「トラックの中身を入れる」
- ・「セグメントから、トラックやサブトラックを削除する」
- ・「セグメントの試聴」
- ・「クリップの分割」
- ・「キーの使い方」

## セグメントに、Music Trackを追加する

セグメントにミュージックトラックを追加するのは、Wwiseの以下の場所です。

- ・Project Explorerの、Audioタブ
- ・Music Segment Editor



### 注記

新規ミュージックトラックの動作は、デフォルトで、Normalに設定されます。動作設定を変更するには [「Music Trackの再生方法の設定」](#) を参照してください。

### Audioタブで、新規ミュージックトラックを追加するには:

1. Interactive Music Hierarchyで、トラックを追加する対象セグメントを右クリックして、New Child > Music Trackを選択する。  
そのセグメントに新規トラックが追加され、Music Segment Editorに表示される。
2. デフォルトのミュージックトラック名を、内容が分かる名前に変更する。

### Music Segment Editorで、新規ミュージックトラックを追加するには:

1. Music Segment Editorに、セグメントを1つロードする。
2. Music Segment Editorで、右クリックしてショートカットメニューを開き、New Trackを選択する。  
New Music Trackダイアログボックスが開く。
3. デフォルトのミュージックトラック名を、内容が分かる名前に変更して、OKをクリックする。

Music Segment Editorに、新規トラックが追加される。

### 関連トピック

- 「トラックの中身を入れる」
- 「トラックの再生タイプの設定」
- 「セグメントから、トラックやサブトラックを削除する」

## Music Trackの再生方法の設定

プロジェクトにある全てのミュージックトラックに、一通りの動作設定がされています。この動作設定で、ミュージックトラックやそのサブトラックの再生方法が決まるほか、そのミュージックトラックのミュージックを、メモリに保存するのか、あるいはメディアやハードドライブから直接ストリーミングするのかも決まります。

ミュージックトラックに関して、以下の動作設定を行います。

- 「ミュージックのストリーミング」
- 「トラックの再生タイプの設定」

### ミュージックのストリーミング

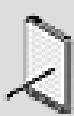
どのミュージックをメモリから再生し、どのミュージックをDVD、CD、ハードドライブなどからストリーミングするのかを、設定できます。ディスクやハードドライブからミュージックをストリーミングする場合は、再生遅延や同期問題を避けるために、以下のオプション設定も利用できます。

- Look-ahead time（ルックアヘッドタイム） - サウンドエンジンがストリーム用データをシークできるように、時間に余裕を持たせる場合は、その時間の長さを指定する。ルックアヘッドタイムが、トラックのレイテンシを決める。

- **Prefetch (プリフェッチ)** - ファイルの残り部分をフェッチする間に発生するレイテンシ時間を補う、小さなオーディオバッファを設定する機能。

ルックアヘッドタイムやプリフェッチサイズは、ハードドライブ、CD、DVDなど様々なメディアソースの要件に合わせて、指定します。プリフェッチサイズが、短すぎるとレイテンシが発生し、長すぎるとメモリ負荷が大きくなります。また、ルックアヘッドタイムが短すぎると、セグメント内でミュージックトラック同士の同期が失敗し、逆にルックアヘッドタイムが長すぎると、再生遅延が発生してしまいます。プリフェッチサイズとルックアヘッドタイムの条件を変えながら調整して、ミュージッククリップのレイテンシと、同期の、最適なバランスを見極める必要があります。

ルックアヘッドタイムとプリフェッチ設定を組み合わせて使えば、ゲーム中にストリーミングするミュージックの、非同期やボイススターべーションの可能性を減らすことができます。例えば、あるミュージックピースのシンクポイントが、セグメントの冒頭にない場合や、そのセグメントのプレエントリ部分を使用しない場合は、メモリにロードされたプリフェッチデータが使えないでの、代わりに、Wwiseはルックアヘッドタイムを使います。



#### 注記

オーサリングアプリケーションWwise内では、ストリーミングオプションを選択しているなくても、オーディオは常にストリーミングで再生されます。その結果、ゲーム中にストリーミングされないミュージックオブジェクトも、Wwiseで再生する際はストリーミングされます。このようなミュージックオブジェクトをWwiseで再生する時の、ミュージックトラック同士の非同期や、ボイススターべーションを避けるには、内部的な再生ルックアヘッドタイム (Internal playback look-ahead time) の設定が必要となることがあります。内部的な再生ルックアヘッドタイムの設定については「[Music Trackのルックアヘッドタイムの設定](#)」を参照してください。

### ミュージックトラックをストリーミングするには:

1. Property Editorに、ミュージックトラックを1つロードする。
2. Streamオプションを選択する。

Stream機能が設定可能となる。

3. テキストボックスLook-ahead timeに、サウンドエンジンがストリーミングデータをシークする時間として確保すべき長さを、ミリ秒単位で入力する。この設定時間がそのトラックのレイテンシとなるので、例えば設定値を「100 milliseconds」とすると、Playを押してから100ミリ秒だけ間をおいて、ミュージックが聞こえ始める。

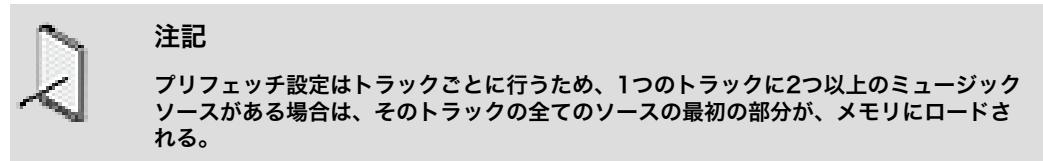


#### 注記

Wwiseが、様々なルックアヘッドタイムを設定した複数のトラックが入ったセグメントの、データをシークする時は、再生するソースのあるトラックのうち、最長のルックアヘッドタイムを、使用します。

4. ミュージックをトリガーした時に全く遅延なく再生させるには、Zero Latencyオプションを選択する。
5. ゼロレイテンシを達成するためには、サウンドの最初の部分をメモリに保存して、メディアドライブからファイルの残りをフェッチするまでのディレイを補

う必要がある。テキストボックスPrefetch lengthに、ミュージックのメモリに保存する部分を、ミリ秒単位で入力する。



## 関連トピック

- 「[トラックの再生タイプの設定](#)」
- 「[トラックの中身を入れる](#)」

## トラックの再生タイプの設定

ゲームのミュージックに変化を与えるには、セグメント内にあるトラックごとに、再生タイプ (Playback type) を設定します。通常のトラック (Normal track) であれば、単純に、そのセグメントが呼び出されるたびに同じトラックが再生されますが、ランダムトラック (Random track) やシーケンストラック (Sequence track) を作成することもできます。ランダムステップトラックや、シーケンスステップトラックは、中に入れるサブトラックの数に制限がないので、そのトラックが再生される度に、違うミュージックを再生させることができます。

3つのトラックタイプの再生動作の違いを、下図に示します。

Track Type	Segment A	Game calls Segment A (1st time)	Game calls Segment A (2nd time)	Game calls Segment A (3rd time)
		Wwise Plays	Wwise Plays	Wwise Plays
Normal	Track1	Track1	Track1	Track1
Random Step	Track1 Sub-track1 Sub-track2 Sub-track3	Sub-track1 Sub-track3	Sub-track1 Sub-track2	
Sequence Step	Track1 Sub-track1 Sub-track2 Sub-track3	Sub-track1 Sub-track3	Sub-track2 Sub-track3	

3種類のトラックタイプを組み合わせることで、そのセグメントがゲームに呼び出されるたびに、新しいミュージックが再生される印象を与えます。例えば、1つのMusic Segmentに4つのランダムステップトラックを入れ、そのランダムステップトラックに、それぞれ4つのサブトラックを入れたとします。ゲームがこのMusic Segmentを呼び出すと、そのミュージックのバリエーションとして256種類が提供されます。

## トラックの再生タイプを設定するには:

- Property Editorに、ミュージックトラックを1つロードする。

2. グループボックス**Track Type**で、以下のオプションを1つ選択する。
  - **Normal**は、親のセグメントが再生されると、毎回、現在のトラックが再生される。Normalモードでは、サブトラックを追加できない。
  - **Random Step** は、親のセグメントが再生されると、毎回、ランダムな順番で、サブトラックの1つが再生される。
  - **Sequence Step** は、親のセグメントが再生されると、毎回、順番通りに、サブトラックの1つが再生される。

**Switch** は、再生するサブトラック（1つ以上）を選択できる。

### 関連トピック

- 「[Music Trackの再生方法の設定](#)」
- 「[ミュージックのストリーミング](#)」
- 「[サブトラックをSwitchやStateに結びつける](#)」

## トラックに、サブトラックを追加する

ミュージックスコアに変化を与えるために、トラックのレイヤを増やしたりサブトラックを追加したりできます。トラックやサブトラックの動作としてランダム再生やシーケンス再生を設定できるので、再生の配列や組み合わせの可能性は、ほぼ無制限にあります。なおサブトラックを追加できるのは、ランダムステップ（Random Step）またはシーケンスステップ（Sequence Step）に設定されたトラックだけです。動作やトラックの設定方法については「[Music Trackの再生方法の設定](#)」を参照してください。各サブトラックの中身を入れた後に、1つのサブトラックを選んで試聴することもできます。Wwiseオーサリングアプリケーション内に限り、テスト目的のために、強制利用（Force Usage）機能を使って、トラックのランダム動作やシーケンス動作をオーバーライドすることができます。サブトラックを個別に試聴する方法については「[サブトラックの強制再生](#)」を参照してください。

### トラックに、サブトラックを追加するには:

1. サブトラックを追加する対象のトラックを、右クリックする。
2. ショートカットメニューで、Add Sub-Trackを選択する。

このトラックに、新規サブトラックが追加される。

### 関連トピック

- 「[トラックの中身を入れる](#)」
- 「[トラックの再生タイプの設定](#)」
- 「[セグメントに、Music Trackを追加する](#)」
- 「[サブトラックをSwitchやStateに結びつける](#)」

## サブトラックをSwitchやStateに結びつける

ミュージックスコアに変化を与えるために、Switch GroupやStateやState Groupを使って再生するサブトラックを決定できます。これを行うにはTrack Typeを

Switchに設定する必要があります。動作やトラックの設定方法については「[Music Trackの再生方法の設定](#)」を参照してください。

サブトラックをSwitchやStateに結びつけるには、トラックのSwitch GroupまたはStateやState Groupを選択する必要があります。トラックのSwitch GroupやStateやState Groupは、トラックのGeneral Settingsタブにある、Switch Typeボックスの設定項目で選択します。

SwitchやStateが切り替わるときのトラックの動作は、Transitionsタブで設定します。トランジションの設定方法については[28章トランジションの活用](#)を参照してください。

### サブトラックをSwitchまたはStateに結びつけるには:

1. サブトラックのヘッダ（黄色バー）を右クリックする。
2. ショートカットメニューで、Associationを選択する。
3. 既存SwitchまたはStateを選択するか、新たに作成する。

サブトラックがSwitchまたはStateに結びつけられる。

Switch名またはState名がサブトラックの上部に表示される。

### 関連トピック

- ・「[トラックの中身を入れる](#)」
- ・「[トラックの再生タイプの設定](#)」
- ・「[セグメントに、Music Trackを追加する](#)」

## トラックの中身を入れる

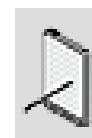
インタラクティブミュージックプロジェクトの作業の大部分は、トラック単位で行い、それぞれのトラックに対して、いくつかのミュージッククリップを追加してから、必要に応じて、これらのクリップをアレンジ、編集、ミックスします。

ミュージックトラックに中身を入れるには、Wwiseの他のオーディオファイルと同じ方法で、Audio File Importerを使ってトラックにミュージックファイルをインポートします。インポート操作については「[メディアファイルのインポート](#)」を参照してください。

### Quick Import

ミュージックトラックの中にオーディオファイルをインポートするためのショートカットを、下表に示します。

To	ショートカット
ミュージックファイルのインポート	Project Explorerで、1つのミュージックトラックに、ファイルをドロップして入れる。



#### 注記

置換(Replace)機能を使えば、必要に応じてオーディオファイルの置換もできます。ファイルの置換については「[メディアファイルの置換](#)」を参照してください。

### 関連トピック

- ・「セグメントから、トラックやサブトラックを削除する」
- ・「トラックの再生タイプの設定」
- ・「セグメントに、Music Trackを追加する」

## セグメントから、トラックやサブトラックを削除する

不要となったトラックやサブトラックは、セグメントから削除できます。トラックを削除すると、このトラックのサブトラックも全て、削除される。

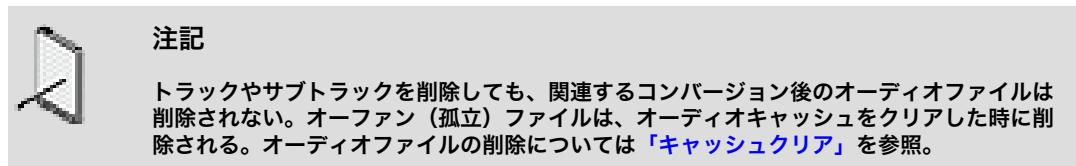
Music Segmentから、トラックやサブトラックを削除するには:

1. Music Segment Editorで、削除するトラックまたはサブトラックを1つ、右クリックする。



2. ショートカットメニューで、Deleteを選択する。

選択したトラックと、その中のサブトラックが、セグメントから削除される。



### 関連トピック

- ・「トラックの中身を入れる」
- ・「トラックの再生タイプの設定」
- ・「セグメントに、Music Trackを追加する」

## Music Editorの、目盛に合わせたスナップ機能

タイムラインに沿って、クリップ (Clip) 、キュー (Cue) 、再生カーソル (Play cursor) を正確に配置できるように、選択した目盛の位置に、スナップ (Snap) 機能で合わせることができます。

タイムラインに沿って、スナップさせるには:

1. 再生カーソル、クリップ、またはキューを、タイムラインにある以下のいずれかの位置に合わせてスナップさせるには、該当ボタンをクリックする:

Bars/Beats

Cues

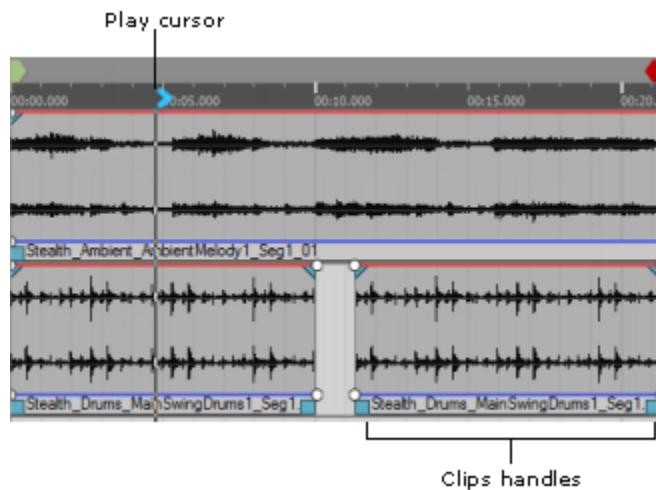
Clips/Loops

### 関連トピック

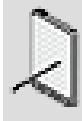
- ・「[キーの使い方](#)」
- ・「[クリップの使い方](#)」

## クリップの使い方

Music Segment Editorの様々なクリップ専用ツールを使えば、タイムラインに沿ってクリップを簡単に編集できます。あるトラックのクリップを上下にドラッグして他のトラックまで移動させたり、クリップを別のクリップに重ねたりすることができます。クリップのハンドル (Clip handle) を内側にドラッグするとクリップが短くなり、外側にドラッグすると長くなります。クリップを延長すると、クリップが繰り返されます。1回の繰り返しを、1回のループ (Loop) と呼びます。



Music Segment Editorでクリップを操作する時に利用できる、クリップ専用の様々なツールやインジケータを、下表に示します。

アイコン	名前	内容
	Clip handle クリップハンドル	ドラッグすると、クリップが短縮または延長される。クリップを延長すると、ループが作成される。
	Play cursor 再生カーソル	セグメントが再生中または一時停止中であれば、そのセグメントの現在の再生位置を表す。セグメントが停止中であれば、再生を再開する時の開始地点を示す。   <b>注記</b> Source Editor の Play Cursor の開始位置は、初期遅延があるときは順守されません。詳細は、Source Editor ドキュメント を参照してください。
	End cursor エンドカーソル	ドラッグすると、Music Segment Editorに表示された1つのセグメントの中の、全てのミュージックトラックが、短縮また

## Music Track、Music Segment の活用

アイコン	名前	内容
		は延長される。ミュージックトラックを短縮または延長しても、それらのトラックにあるクリップの長さや再生設定は、影響されない。
	Clip indicator クリップインジケータ	ループするクリップの、冒頭を表す。

### クリップのループ再生

Music Segment Editorにあるクリップハンドル (Clip handle) を使って、簡単にクリップをループできます。

#### Music Segment Editorでクリップをループ再生させるには:

1. クリップハンドルをクリックして、ハンドルをドラッグして、クリップを延長する。
2. クリップインジケータがタイムライン上に表示されるまで、クリップを延長する。

クリップのループが、作成されます。さらにクリップを延長して、必要な数のループを追加できる。

### クリップの移動

ミュージックのレイヤ操作やアレンジのために、トラックに入っているクリップを、移動することができます。ミュージックをミックスするために、Music Segment Editorで、クリップを好きなように並べたり重ねたりできます。スナップ (Snap) 機能を使えば、タイムラインに沿って、複数のクリップの位置を正確にそろえることができます。

#### クリップを移動するには:

1. トラックの中のクリップを1つ選択して、別の位置にドラッグする。



#### Tip

カット、コピー、ペーストなど、Windowsの標準コマンドを使って、クリップやトラック全体を移動することもできます。

### クリップの分割

Music Segment Editorで、クリップを分割 (Split) できます。この機能は、例えばバスドラムのヒットを単独で抜き取り、全体の長さに渡りリピートさせる時や、他のトラックにインサートする時などに、活用できます。また、抜き取ったバスドラムのクリップは、どのトラックにも追加できるので、ミュージックのアレンジを変えたり、作曲し直すことさえも可能です。

#### クリップを分割 (Split) するには:

1. クリップを分割する位置まで、再生カーソル (Play Cursor) をドラッグする。

2. このトラックの中で、分割するクリップを右クリックして、ショートカットメニューで、**Split on Play Cursor**（再生カーソルの位置で分割）を選択する。

再生カーソルの位置で、クリップが分割される。目的に応じて、これらのクリップを編集する。

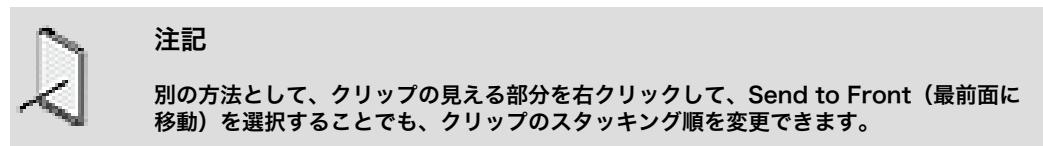
### クリップのスタッキング

複数のクリップがオーバーラップしていて、下にあるクリップのシンクポイントが見えない時などに、クリップの順番を入れ替えることができます。ショートカットメニューに表示される、**Bring to Front**（最前面に移動）コマンド、**Send to Back**（最背面に移動）コマンドを使います。

### クリップのスタッキング順を変更するには:

1. ミュージックトラックの中で、オーバーラップしているクリップを右クリックして、**Send to Back**（最背面に移動）を選択する。

クリップ同士の順番が逆になり、下にあったクリップが上にくる。

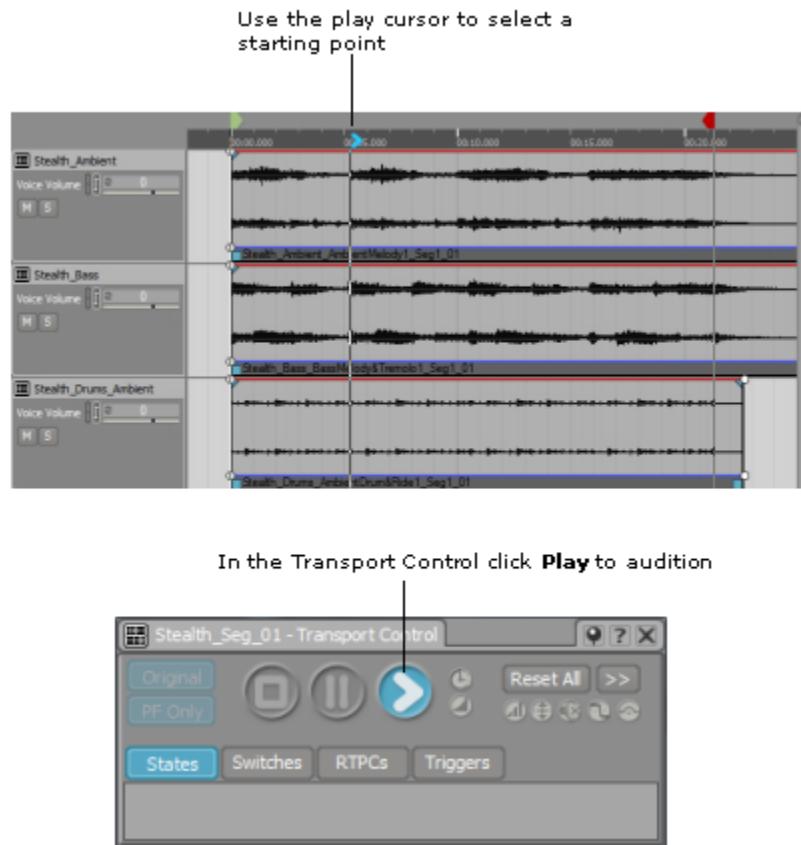


### 関連トピック

- 「クリップの使い方」
- 「クリップの分割」
- 「クリップの移動」
- 「クリップのループ再生」

## セグメントの試聴

あるセグメントの複数のトラックに、それぞれの中身を入れた後は、いつでも Transport Controlを使ってセグメントのトラックを試聴できます。あるクリップの特定ポイントを再生したい場合は、再生カーソル（Play cursor）で再生の開始位置（Starting point）を指定します。



セグメントを試聴する時は、目的に応じて、全てのトラックを再生して全体を聞くことも、特定のトラックをミュート（Mute）またはソロ（Solo）に設定して、トラックごとの内容を聞くこともできます。また、1つのサブトラックだけを強制的に再生（Force playback）することも可能です。

### Transport Controlでトラックを試聴するには:

1. Music Segment Editorに、Music Segmentを1つロードする。

Music Segment Editorに、そのMusic Segmentの全てのトラックとサブトラックがロードされ、そのMusic Segmentが、Transport Controlにロードされる。



2. セグメントを全て試聴するには、Transport Controlの、Playをクリックする。
3. あるミュージックトラックをミュートまたはソロに設定するには、Track Controlエリアにある、Muteボタン、またはSoloボタンをクリックする。
4. このトラックを、タイムラインの途中から再生するには、再生を開始させる位置まで、Play Cursorを移動する。

### 関連トピック

- 「クリップの使い方」

- ・「セグメントに、Music Trackを追加する」
- ・「トラックに、サブトラックを追加する」
- ・「サブトラックの強制再生」

### サブトラックの強制再生

ランダムトラックや、シーケンストラックを再生すると、その中のサブトラックが、ランダムまたは順番通りに再生されます。試行錯誤を繰り返しながら、あるサブトラックに入れるクリップを設定している時に、変更した箇所だけを試聴するには、そのトラックのランダム再生やシーケンス再生をオーバーライドして、聞きたいサブトラックを強制的に再生（Force playback）すると便利です。

サブトラックを、強制的に再生（Force playback）するには:

1. 

Music Segment Editorで、Force Usage（強制使用）アイコンをクリックして、そのサブトラックを、Transport Controlで強制的に再生させる。

アイコンが青色に変わり、選択したサブトラックが、Transport Controlで再生される。

2. このオプションを取り消すには、もう一度、Force Usageアイコンをクリックする。



#### Tip

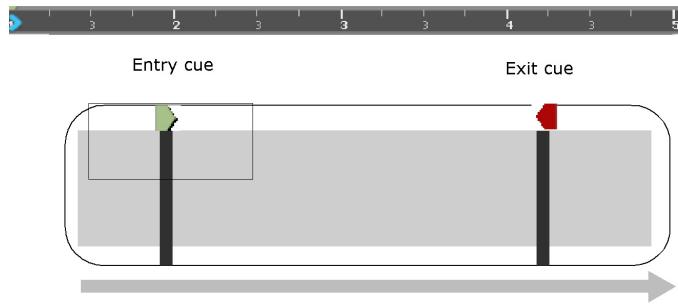
Transport Controlでこの設定を取り消すには、>>をクリックして、Resetメニューを開き、Reset All Music Tracks Force Usageを選択します。

### 関連トピック

- ・「クリップの使い方」
- ・「セグメントに、Music Trackを追加する」
- ・「トラックに、サブトラックを追加する」

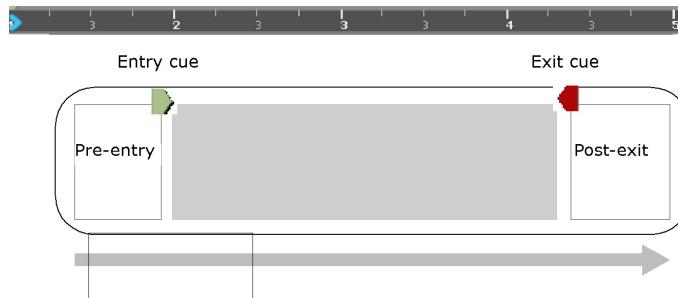
## キューの使い方

キュー（Cue）は、タイムライン上の特定ポイントを示すものです。基本的に、キューはシンクポイントとして利用され、インタラクティブミュージックで発生するState切り替え、ステインガー再生、トランジションなど、様々な切り替わりの際に、複数のMusic Segmentをそろえる役割があります。そろえる際は、テンポやサンプルアキュレートに関連する条件や、ゲームの背景状況も考慮する必要があります。セグメントには、必ずエントリキュー（Entry cue）とエグジットキュー（Exit cue）があり、カスタムキューが設定されていることもあります。



エントリキュー (Entry cue) は、最初の小節の最初のビートが始まる位置を表すと同時に、前のセグメントのエグジットキュー (Exit cue) とスナップする位置を表します。エントリキュー やエグジットキューはクリップから削除できませんが、編集できます。

セグメントのエントリキューより前の部分は、プレエントリ (Pre-entry) と呼ばれ、エグジットキューより後の部分は、ポストエグジット (Post-exit) と呼ばれます。セグメントのこれらの領域を再生するかどうかは、トランジション設定で決まります。トランジションについては「[トランジションを理解する](#)」を参照してください。



エントリキュー やエグジットキュー以外に、カスタムキューを必要なだけ追加して、ゲームの状況に合わせてミュージックをシンクさせることもできます。

キュー設定の作業には、以下のタスクが伴います:

- 「[カスタムキューの追加](#)」
- 「[キューの削除](#)」
- 「[キューの説明](#)」

### カスタムキューの追加

State切り替え、トランジション、ステインガー挿入などに使うカスタムキュー (Custom cue) は、タイムライン上のどの位置にも、追加できます。

### キューを、トラックに追加するには:

1. Music Segment Editorで、カスタムキュー (Custom cue) を追加する場所を、タイムラインの目盛り上で右クリックする。
2. ショートカットメニューで、Add Custom Cueを選択する。

- タイムライン上の選択した位置に、新規カスタムキューが追加される。
3. キューを、設定したい位置までドラッグする。
  4. 必要に応じて、さらにカスタムキューを追加する。

### 関連トピック

- 「エントリキュー、エグジットキューの使用」
- 「キューの説明」
- 「キューの削除」

### キューの削除

カスタムキューは、いつでも削除できます。エントリキュー (Entry cue) や、エグジットキュー (Exit cue) は、削除できません。

### カスタムキューを、削除するには:

1. トラックのタイムライン上で、削除するカスタムキュー (Custom cue) を、右クリックする。
2. ショートカットメニューで、Deleteを選択する。

トラックからこのキューが削除される。

### 関連トピック

- 「エントリキュー、エグジットキューの使用」
- 「キューの説明」
- 「カスタムキューの追加」

### キューの説明

キューの内容の説明を加えることができます。これで、キューの背景が分かります。例えば、ステインガー用に作成した特別なキューを見つける時などに便利です。

### カスタムキューの内容を記入するには:

1. トラックのタイムライン上で、説明を加えるカスタムキュー (Custom cue) を右クリックする。
2. ショートカットメニューで、Change Description (説明を変える) を選択する。

Change Descriptionダイアログボックスが開く。

3. デフォルトの説明文を、キューの内容が分かる説明に変更する。
4. OKをクリックする。

### 関連トピック

- 「エントリキュー、エグジットキューの使用」

- ・「[cueの削除](#)」
- ・「[カスタムクエの追加](#)」

### エントリクエ、エグジットクエの使用

タイムライン上で、エントリクエやエグジットクエの位置を移動して、そのセグメントのプレエントリ部分やポストエグジット部分を、延長または短縮することができます。

**選択したクリップで、セグメントの、エントリクエやエグジットクエを移動するには:**

1. クエを選択して、セグメントに沿って左または右にドラッグする。

**セグメントのエントリクエ、エグジットクエを、選択したクリップの両端に移動するには:**

1. エントリクエとエグジットクエの位置を設定するクリップを、右クリックする。
2. ショートカットメニューで、Move Entry/Exit Cues to Selectionを選択する。

選択したクリップの最初と最後に、エントリクエとエグジットクエが配置される。

### 関連トピック

- ・「[クエの削除](#)」
- ・「[クエの説明](#)」
- ・「[カスタムクエの追加](#)」

---

## 第26章 MIDIの活用

MIDIコンテンツを作成する .....	484
MIDIファイルのインポート .....	484
MIDIコンテンツとMIDIターゲットを理解する .....	484
MIDIコンテンツとオーディオコンテンツをミックスする .....	486
MIDIテンポを理解する .....	486
MIDIの再生スピードを変更する .....	487

## MIDIコンテンツを作成する

AMBおよびWAVファイル以外に、MIDIファイルもInteractive Music Hierarchyで使えます。これらのファイルタイプも、インポートしてからクリップを作成して、Music Segmentトラック（単数または複数）に入れます。WAVクリップの場合は1つのクリップが、それだけで成立する1つのサウンドソースです。一方、MIDIクリップには再生される音符が入っているだけで、音符を解釈するにはMIDIインストゥルメントを使う必要があります。インストゥルメントをターゲットする1つのMIDIクリップ、または多数のMIDIクリップから生成される、全てのMIDIイベントを、インストゥルメントが解釈します。

## MIDIファイルのインポート

MIDIファイルは、インタラクティブミュージック階層のミュージッククリップの中にインポートできます。MIDIファイルをActor-Mixer Hierarchyにインポートすることはできません。

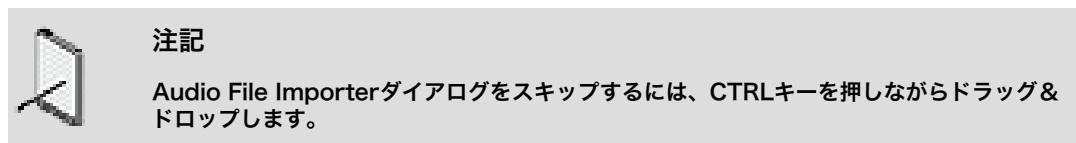
WAVクリップを作成するのと同じ方法で、MIDIクリップも作成できます。ファイルからクリップを作成する方法については「[トラックの中身を入れる](#)」を参照してください。

**MIDIファイルをドラッグ&ドロップでインポートするには:**

1. MIDIファイルを、Windows ExplorerまたはMac FinderからInteractive Music Hierarchyの任意の場所にドラッグする。

Audio File Importerが開く。

2. Importをクリックして、インポート操作を完了させる。



**Audio File Importerを使ってMIDIファイルをインポートするには:**

1. Project Explorerで、.midをインポートする場所を選択する。
2. キーボードショートカット(**Shift + I**)を使ってAudio File Importerを開く。
3. Add Files...ボタンをクリックする。
4. MIDIファイルまで移動して、OKをクリックする。
5. 再びOKをクリックして、インポート操作を完了させる。

## MIDIコンテンツとMIDIターゲットを理解する

MIDIファイルのレファレンスは、トラックやセグメントの中にあるミュージッククリップ内に格納されます。MIDIクリップには、ノート、ピッチベンド、CCなど

のMIDIソース情報が入っているだけです。再生させるインストゥルメントは定義されていません。再生に使うインストゥルメントは、Music Segmentや先祖のその他のミュージックにあるMIDIターゲットレファレンスで定義されます。

- MIDI再生用インストゥルメントは、Actor-Mixer Hierarchyで作成して保存する必要があります。
- MIDI再生用インストゥルメントは、一般的にサウンドに使われる標準オブジェクト構造を使います。

## 例

- Actor-Mixer Hierarchy
  - Instrument Work Unit
    - Drum Kit
    - Piano
  - Interactive Music Hierarchy
    - Default Work Unit
      - ソングセグメント
        - Piano Track (MIDIクリップ使用)
        - Drum Track (MIDIクリップ使用)
        - Guitar (オーディオクリップ使用)

前述の例の内容を、以下に示します。

- Piano Trackには、Actor-Mixer HierarchyにあるPianoオブジェクトへのMIDIターゲットレファレンスがあります。
- Drum Trackには、Actor-Mixer HierarchyにあるDrum KitオブジェクトへのMIDIターゲットレファレンスがあります。
- Guitar Trackには、MIDIターゲットがありません。

MIDIクリップの入ったMusic Segmentを再生すると、MIDIクリップが、指定されたMIDIターゲットにMIDIデータを送信します。実際のMIDIターゲット（インストゥルメント）はアクターミキサー階層にあり、MIDIデータを受信してからMIDI設定に従い再生します。

## ミュージックオブジェクトでMIDIターゲットを指定するには:

1. ミュージックオブジェクト (Music Track、Music Segment、親オブジェクト) を選択する。
2. Property Editorで、MIDIタブを開く。
3. MIDI Targetグループで、Override Parentをクリックする (該当する場合)。
4. [...]ボタンをクリックして、MIDIターゲットインストゥルメントまで移動する。

詳しくは[27章MIDIインストゥルメントの作成](#)を参照。

## MIDIコンテンツとオーディオコンテンツをミックスする

MIDIコンテンツをミュージック階層で再生すると、ミュージックオブジェクトがActor-Mixer HierarchyにあるMIDIターゲット（つまりインストゥルメント）にMIDIデータを送信します。MIDIターゲットに対して再生される実際のオーディオコンテンツは、MIDIソースの保存場所で再生されるのではなく、アクターミキサー階層上で再生されます。つまり、ミュージックオブジェクトに設定されたボイスプロパティは、再生されるインストゥルメントに影響しません。

例えば、Music SegmentやMusic Trackに設定されているボイスボリュームは、Actor-Mixer Hierarchyにあるピアノインストゥルメントに影響しません。ピアノのボイスボリュームを変更するには、MIDIソースではなく、ピアノオブジェクトのボイスボリュームを直接、変更する必要があります。

ミキシングをしやすくするために、インストゥルメントコンテンツとミュージックオーディオコンテンツを合わせるためのバス階層を作成できます。インストゥルメントとミュージックトラックを1つのバスにルーティングすることで、統一されたオーディオミキシングポイントを確保できます。

## MIDIテンポを理解する

MIDIクリップとWAVクリップには、ある大きな違いがあります。WAVクリップの長さは、ソースファイルのコンテンツによって個別に決まります。一方、MIDIファイルの長さは使用するテンポで決まります。使用するテンポは、クリップを作成するときにTempo Sourceプロパティで決定されます。

通常、MIDIファイルには正しい再生スピードのためのテンポ情報が入っています。このテンポ情報が、インタラクティブミュージックオブジェクト（セグメントや親）にあるテンポ情報と、コンフリクトを起こしてしまいます。ソースのMIDIファイルのテンポを使うのか、Interactive Music Hierarchyのテンポを使うのかを選択します。

テンポのソース設定を、下表に示します。

Tempo Source	Definition
File	MIDIファイルに入っているテンポを使用する。
Hierarchy	インタラクティブミュージックにおけるMIDIクリップの先祖に設定されたテンポを使用する。

### MIDIのテンポのソースを設定するには:

1. ミュージックオブジェクトを選択する。
2. Property Editorで、MIDIタブを開く。
3. MIDI Clip Tempoグループで、使用するSourceを選択する。

MIDIクリップのテンポは作成後に変更できますが、長さは変わりません。長さを変えるには、クリップハンドルを使います。詳しくは「[クリップの使い方](#)」を参照してください。

## MIDIの再生スピードを変更する

Voice Pitchプロパティは、Interactive Music Hierarchyでは設定できません。ただし、ミュージック階層内での再生スピードは変更できます。

ミュージックオブジェクトの再生スピードを変更すると、以下に影響します。

- ・オーディオクリップのピッチに影響。
- ・MIDIクリップの再生レートに影響。

再生スピードが1のときは、オリジナルのスピードで再生されます。再生スピードが2のときは、オリジナルの2倍速で再生されます。再生スピードが0.5のときは、オリジナルの半分の速さで再生されます。



### 注記

Playback Speedプロパティを、Property EditorのRTPCタブでGame Parameterに付随させることもできます。

### 再生スピードを変更するには

1. ミュージックオブジェクト（ミュージックトラック以外）を確認する。
2. Property Editorで、General Settingsタブを開く。
3. Playback Speedを設定する。

---

## 第27章 MIDIインストゥルメントの作成

Synth Oneインストゥルメントをデザインする .....	489
シンプルなMIDIサンプルインストゥルメントをデザインする .....	489
MIDIノートトラッキングを理解する .....	490
MIDIフィルタを理解する .....	490
MIDIイベントを理解する .....	491
MIDIイベントにフェードイン、フェードアウトを追加する .....	492
MIDIデータを使ってオブジェクトのプロパティ値をコントロールする .....	492
MIDI Keymap Editorを使う .....	492
MIDIキーボードでインストゥルメントをテストする .....	493
MIDIをDAWからWwiseにルーティングする .....	494

## Synth Oneインストゥルメントをデザインする

MIDIメッセージの分かるソースプラグインを使って、ソースプラグインからMIDIインストゥルメントを作成することが可能です。

ソースプラグインを使ったインストゥルメント（シンセサイザなど）を作成するには：

1. Actor-Mixer Hierarchyで空のサウンドオブジェクトを作成する。
2. Project Explorerで、このサウンドを選択する。
3. Contents Editorで、Add Source>>をクリックする。
4. セレクタメニューで、Synth Oneを選択する。
5. Viewsメニューで、Source Editorを選択する (Shift+X)。
6. Actor-Mixer HierarchyのSoundオブジェクトをクリックして、Source Editorに表示します。
7. Source Editorで、Frequency ModeをMIDI Noteに設定する。

これで、このインストゥルメントをミュージックオブジェクト（Music Segmentなど）の中でMIDIターゲットとして参照できます。

## シンプルなMIDIサンプルインストゥルメントをデザインする

サンプルインストゥルメントをデザインするときに、アクターミキサー階層の全てのコンテナ（ブレンドコンテナ、スイッチコンテナ、ランダムコンテナ、シーケンスコンテナ、サウンド）を利用できます。インストゥルメントデザインが複雑になるほど、実装も複雑になります。最もシンプルなサンプルインストゥルメントは、サンプル1つのインストゥルメントです。

1サンプルのインストゥルメントを作成するには：

1. Project Explorerで、インストゥルメントを作成する場所を選択する。
2. Viewsメニューで、Audio File Importerを選択する (Shift+I)。
3. Add Files...をクリックする。
4. .wavファイルまで移動して、OKをクリックする。
5. 再びOKをクリックして、インポート操作を完了させる。



### Tip

wavファイルをアクターミキサー階層に直接ドラッグ&ドロップすることもできます。

これで、このインストゥルメントをミュージックオブジェクト（Music Segmentなど）の中でMIDIターゲットとして参照できます。



### Tip

Source Editorを使って、インポートしたサンプルをトリムできます。ソースの冒頭にサイレンスがある場合、通常は取り除きます。

## MIDIノートトラッキングを理解する

MIDIノートトラッキングのパラメータは、Actor-Mixer HierarchyオブジェクトのProperty EditorのMIDIタブにあります。MIDIメッセージを受信したときに、サウンドオブジェクトの再生中にピッチシフトするのかどうかを、これらのパラメータを使って設定します。再生するサウンドをピッチシフトする場合は、MIDIメッセージのノート、そしてActor-Mixerオブジェクトのソースが示すノート、つまりルートノートに従いピッチをシフトします。

インストゥルメントのActor-Mixer Hierarchyのどのオブジェクトにおいても、ノートトラッキングのパラメータを指定したりオーバーライドしたりできます。Actor-Mixer Hierarchyのプロパティについては「[プロジェクト階層内のプロパティについて](#)」を参照してください。MIDIノートトラッキングのパラメータを、以下に示します。

- **Override parent:**これを設定すると、先祖にあたるオブジェクトのノートトラッキングパラメータが無視される。
- **Enable:**これを設定すると、オブジェクトを再生する際に、サウンドがピッチシフトされる。ピッチシフトは、受信したMIDIメッセージのノートと、Root noteパラメータを使って行われる。
- **Root note:**オブジェクトのソースが示すノート。

## MIDIフィルタを理解する

MIDIフィルタは、Actor-Mixer HierarchyオブジェクトのProperty EditorのMIDIタブにあります。複雑なオブジェクト構造でMIDIメッセージを受信したときに、MIDIフィルタを使って再生する子オブジェクトを選択できます。

MIDIフィルタを使うと、以下に基づいて、再生する子オブジェクトが決まります。

- MIDIノートキー
- MIDIノートベロシティ
- MIDIチャンネル

リサンプルを行うと、サウンドのピッチがルートノートから離れるにつれ不要なノイズが生じる傾向があるため、異なるルートノートの複数の録音サンプルを使ってインストゥルメントの範囲を完全にカバーするとよいでしょう。メモリが限られていても良い結果をもたらすことが多いテクニックとして、オクターブごとにいくつかの異なるサンプルルートノートを使い、ピッチを高くしたり低くしたりして、1オクターブの12半音を全てカバーする方法があります。

あるインストゥルメントの2オクターブで、ルートノートのピッチを2度下げて短2度上げた例を、以下に示します。

- インストゥルメント (Blend Container)
  - Root note: C3 - Range: Bb2 to C#3
  - Root note: E3 - Range: D3 to F3
  - Root note: G#3 - Range: F#3 to A3

- Root note: C4 - Range: Bb3 to C#4
- Root note: E4 - Range: D4 to F4
- Root note: G#4 - Range: F#4 to A4

Blend Containerは、コンテナが再生されると全ての子オブジェクトを同時に再生します。ただし、フィルタルールを満たさない子は、フィルタによってブロックされます。

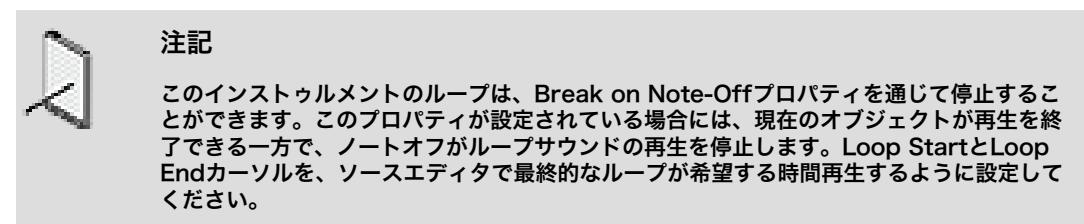
## MIDIイベントを理解する

MIDIイベントプロパティは、Actor-Mixer HierarchyオブジェクトのためのProperty EditorのMIDIタブにあります。MIDIメッセージを受信すると、そのMIDIイベントプロパティを使ってオブジェクトを再生するか判断します。このオブジェクトは、ノートオンもしくはノートオフイベントのいずれかで再生されます。これらのプロパティはオブジェクトの再生開始のみに使用されることに注意してください。オブジェクトの再生を停止するには、プロパティにエンベロープを割り当てる必要があります（「[エンベロープの活用](#)」を参照）。

典型的なシナリオは、Note-OnアクションでPlay（再生）となります。

### ループするインストゥルメントを作成するには：

- Project Explorerで、ループさせるサウンドを選択する。
- Property Editorで、General Settingsタブを開く。
- このサウンドのLoopを有効にする。
- Property Editorで、MIDIタブを開く。
- Note-OffアクションをStopに設定する。
- Viewsメニューで、Source Editorを開く（Shift+X）。
- 再びサウンドを確認する。
- Source Editorで、ループ開始とループ終了のカーソルを動かして、wavファイルのアタックとリリース部分を除外する。
- ループポイントが聞こえないようになるまでクロスフェードの長さを調整する。



インストゥルメントのリリースのときに、ノートオフに合わせて特定のサウンドを再生させるというシナリオも考えられます。例えば、ギターノートが終了するときのストリングミュートサウンドをトリガーさせるのに使えます。この場合、以下のオブジェクトを作成して設定します。

- Blend Container
  - Attack+Loop Sound: Note-On=Play, Note-Off=Stop

- Release Sound: Note-On=No Action, Note-Off:Play

## MIDIイベントにフェードイン、フェードアウトを追加する

インストゥルメントに強弱を加えたい場合には、インストゥルメントのVoice Volumeに付属するEnvelopeを使用できます。

Voice Volumeに Envelopeを追加する方法については、「[エンベロープの活用](#)」を参照してください。

## MIDIデータを使ってオブジェクトのプロパティ値をコントロールする

オブジェクトのプロパティ値をコントロールするために使えるMIDIメッセージを、以下に示します。

- MIDIノートベロシティ
- MIDIノートキー（ナンバー）
- MIDIノート周波数
- MIDIノートアフタータッチ
- MIDI CC値（0-127、Modulation Wheelも含む）
- MIDIピッチベンド

インストゥルメントのVoice VolumeをMIDIノートベロシティでコントロールするのが、典型的なシナリオです。

MIDIノートベロシティでボイスボリュームをコントロールするには:

1. Project Explorerで、Actor-Mixer Hierarchyのオブジェクトを選択する。
2. Property Editorで、RTPCタブを開く。
3. RTPCリストの[>>]ボタンをクリックして、新しい項目を追加する。
4. セレクタメニューで、Voice Volumeを選択する。
5. X軸の[>>]セレクタをクリックする。
6. セレクタメニューで、MIDI > MIDI Note Velocityを選択する。
7. RTPCグラフのVoice Volumeカーブを調整する。

## MIDI Keymap Editorを使う

MIDI Keymap Editorビューを使って、Actor-Mixer Hierarchyオブジェクトの全てのMIDIプロパティを編集できます。

ビューを開くには:

1. Project Explorerで、Actor-Mixer Hierarchyのオブジェクトを選択する。
2. Property Editorで、MIDIタブを開く。
3. Keymap Editorボタンを押す。

### 同じプロパティ値を複数のオブジェクトに設定するには:

1. 編集する複数のオブジェクトを、 MIDI Keymap Editorで選択する。
2. 選択したオブジェクトの1つのプロパティ値を設定する。

選択した複数のオブジェクトが同じ設定値になります。

### 複数のオブジェクトのプロパティ値をオフセットにするには:

1. 編集する複数のオブジェクトを、 MIDI Keymap Editorで選択する。
2. ALTキーを押しながら、選択したオブジェクトの1つのプロパティスライダを動かす。

選択した複数のオブジェクトのプロパティ値がオフセットされる。

### MIDI Keymap Editorにプロパティを追加するには:

1. MIDI Keymap Editorのビュー設定を開く (Ctrl+Alt+V)。
2. 追加するプロパティを選択する。
3. OKを押す。

新しい列が追加される。

## MIDIキーでインストゥルメントをテストする

MIDIインストゥルメントをデザインするときは、外部MIDIキーボードデバイスでインストゥルメントをテストできます。

### デバイスをWwiseに接続するには:

1. Projectメニューで、 Control Surface Devicesを選択する。
2. Addボタンをクリックする。
3. デバイスに名前をつける。
4. OKをクリックする。

デバイスがリストに追加される。

5. Receive From列で、 MIDI INに使うデバイスを選択する。

Connectedというメッセージが表示される。

6. Send To列で、 MIDI OUTに使うデバイスを選択する。

Connectedというメッセージが表示される。

7. Closeをクリックする。

デバイスを使う準備が完了する。

### キーボードキーを現在の選択にバインディングするには:

1. Viewsメニューで、 Control Surface Bindingsを選択する (Ctrl+Shift+Q)。

2. ビューの左上にある[>>]ボタンをクリックして、新規Control Surface Sessionを作成する。
3. Current Selectionグループ（フォルダ）をクリックする。
4. Add Bindingボタンをクリックする。
5. Property/Commandセレクタボタンをクリックしてメニューを開く。
6. Object Command > Pass MIDI Noteを選択する。
7. プロジェクトを保存する。



セッションとバインディングを作成したあとは、Project Explorerでオブジェクトを選択すると、このオブジェクトが自動的にMIDIインストゥルメントにロードされ、再生する準備が整います。

## MIDIをDAWからWwiseにルーティングする

外部アプリケーション（DAW、デジタルオーディオワークステーション）から送信されるMIDIメッセージを、バーチャルMIDIコネクタを使ってWwiseにルーティングすることができます。これは、Wwiseで作成したインストゥルメントを使いながらDAWでMIDIミュージックを作曲するときに便利です。

複数のインストゥルメントを同時に使ってオーサリングするには、全てのインストゥルメントを入れたBlend Containerを作成して、MIDIチャンネルフィルタを使ってそれぞれのインストゥルメントを区別します。

- **Mac:** 内蔵のIACドライバは、使用しようとしているものです。開始するには、Applications/Utilities.の下にあるAudio MIDI Setupを起動します。MIDIスタジオが表示されていない場合は、Windowに移動しShow MIDI Windowを選択します。MIDI Devicesをクリックし、IAC Driverアイコンをダブルクリックし、Device is onlineオプションをオンにして有効にします。任意の数のポートが追加されると、DAWに表示されます。ポートは、MIDI Note、MIDI Control Change、MIDI Syncメッセージの送受信に使用できます。
- **Windows:** Windowsは元々仮想MIDIドライバを持っていないので、仮想MIDIドライバをダウンロードする必要があります。MacのIAC Busに相当する最もよく知られているWindowsの同等機能は[MIDI OX Utility](#)のコンポーネントであるMIDI Yokeです。このドライバーアプリケーションに代わるものは、DAWからWwiseにMIDIメッセージをルーティングするために使用できるTobias Erichsenの[loopMIDI](#)です。アプリケーションによって作成された仮想MIDIポートをWwiseのControl Surface Devicesに追加する必要があります。

---

## 第28章 トランジションの活用

概要 .....	496
トランジションを理解する .....	497
トランジションの追加 .....	498
トランジションのコピー、ペースト .....	499
トランジションの削除 .....	500
ソースやデスティネーションのプロパティを設定する .....	500
トランジションセグメントの使用 .....	504
インタラクティブミュージックのトランジションに関する、コツとベストプラクティス .....	505

## 概要

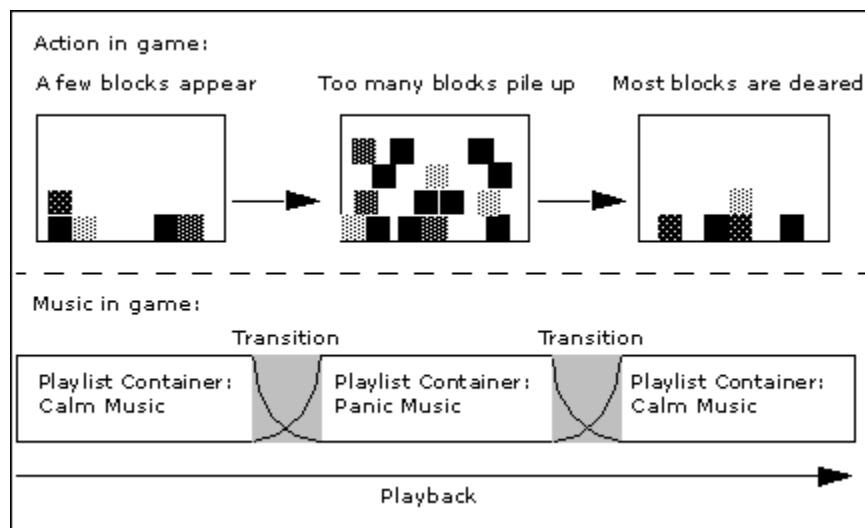
インタラクティブミュージックを聞きやすくするための鍵が、トランジション(Transition)です。スムーズなトランジション(移行)なしでは、ゲームミュージックが常に展開しつづける1つの楽曲であるという錯覚が、失われてしまします。下手につなぎ合わせた音楽セグメントは耳障りであり、ゲームのリズムを壊してしまいます。このような現象を回避するために、Wwiseでは、ミュージックオブジェクト間のトランジションができるだけシームレスに行えるように、カスタム設定できます。

トランジションは、最初にあるソース(Source)セグメントと、次のデスティネーション(Destination)セグメントの間のスムーズな橋渡しの役割を担いますが、Wwiseには、これを効果的に仕上げるツールが準備されています。トランジションは、個別のセグメント同士で設定することも、セグメントの入ったコンテナ同士で設定することもできます。このようにして、不調和で急な楽節の切り替わりが防げます。

### トランジションの活用例

例えば、ブロックパズルゲームを作る場合を考えます。プレイヤーは、ゲームを上手く進めている(ブロックを素早くクリアしている)か、手こずっている(ブロックが積み上がっている)か、常に2つの状態のどちらかにあります。これらの状態を反映するクールなミュージックが既に作曲してあれば、次は、それを最大限に活かす手段を考えます。1つの方法として、ミュージックが相互に切り替わる時に、そのトランジションが必ず、自然で音楽的になるように設定します。

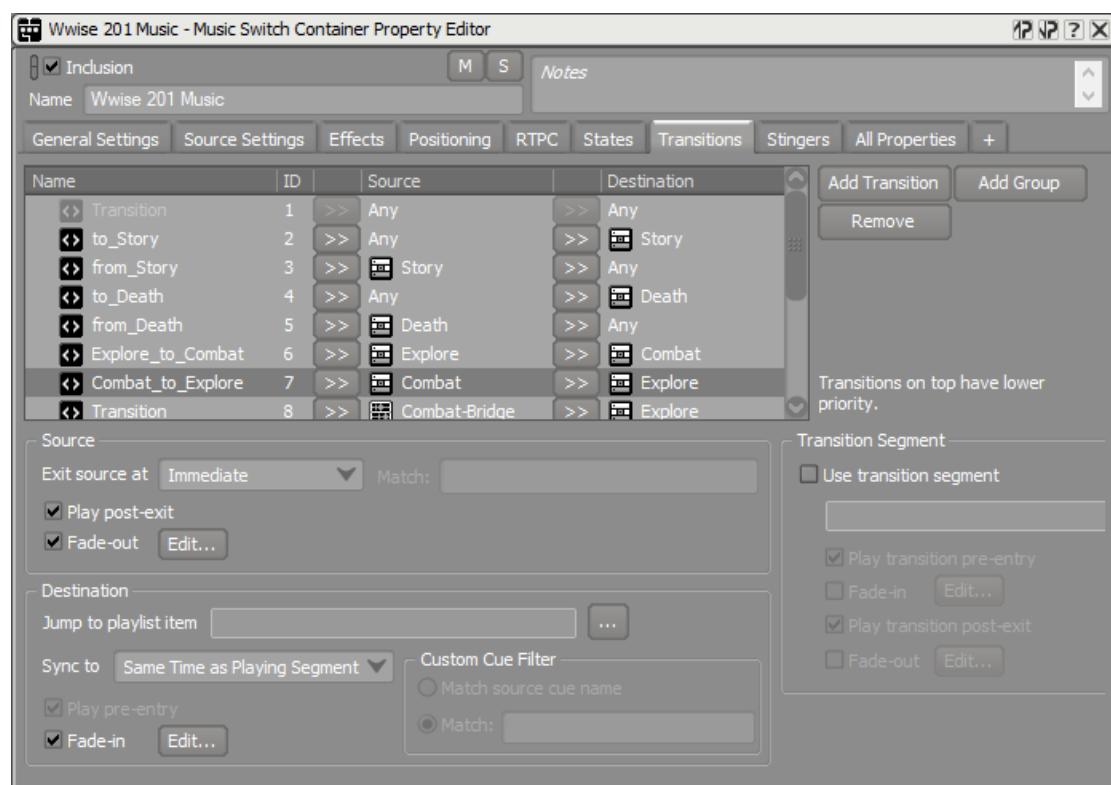
今回の例のゲームのニーズに応えるため、「Puzzle Sounds (パズルサウンド)」というMusic Switch Containerを作成して、中に入れるMusic Playlist Containerとして、「Calm (落ち着き)」と「Panic (パニック)」の2種類を作成します。次に、ゲーム中の状況を反映する2つのGameStateに、これらのコンテナをリンクさせます。2つのPlaylist Containerの切り替わりが上手く行われると、魅力的なミュージックスコアが生み出されます。トランジションの実装方法の例を、下図に示します。



今回の例では、2つのPlaylist Containerの間のトランジションを、フェードイン(Fade-in)カーブと、フェードアウト(Fade-out)カーブの設定で、スムーズにしています。コンテナからコンテナへの変更が、よりゆるやかで自然に感じられ、雑音も回避できます。

## トランジションを理解する

インタラクティブミュージックにおけるトランジション(Transition)は、ソース(Source)と呼ばれる1つのミュージックオブジェクトの再生が止まり、デスティネーション(Destination)と呼ばれる別のミュージックオブジェクトの再生が始まる時に、必ず発生する音楽的な現象です。全てのトランジションは、Property EditorのTransitionタブで設定したプロパティに従い、Wwiseによって自動的に実行されます。



ゲームのミュージックが、あるインタラクティブMusic Segmentから別のものに切り替わった時、またはセグメントの入ったあるコンテナから他のコンテナに切り替わった時に、Wwiseが必ずトランジションを実行します。トランジションを発生させる位置や、フェードの有無は、指定できます。また、トランジション中に再生させるトランジションセグメント(Transition segment)と呼ばれる楽節を設定することもできます。Wwiseのトランジションは全体的にカスタマイズできるので、インタラクティブミュージックにシームレスに導入できます。

## Transition Matrixを理解する

Transitionタブの中心となるのがTransition Matrixで、Music Switch ContainerまたはPlaylist Containerの中にある1つのオブジェクトが他のオブジェクトにトラ

ンジションする時のルールが、オブジェクトごとに表示されます。トランジション設定は、詳細なルールをコンテナ内のオブジェクトごとに1つ1つ設定することも、汎用性のあるルールを1つ以上のオブジェクトに設定することもできます。トランジションルールのソースやデスティネーションとして設定できるものを、以下に示します。

- ・ **ミュージックオブジェクト:** 具体的なセグメント、Music Playlist Container、またはMusic Switch Containerを指定する。
- ・ **仮想フォルダ:** ミュージックスイッチコンテナ内のミュージックオブジェクトを、仮想フォルダで整理できる。ミュージックトランジションのソースまたはデスティネーションとしてVirtual Folderを選択すると、このフォルダ内の、どのミュージックオブジェクトにも、トランジションルールが適用される。
- ・ **Any:** これを設定すると、コンテナの中にある、どのミュージックオブジェクトも、ソースまたはデスティネーションとして使えることを意味する。
- ・ **Nothing:** これを設定するとソースまたはデスティネーションが空であることを意味し、つまり、再生されるミュージックオブジェクトがない。

トランジションが必要な場合に、Wwiseは、現状に適した項目を求めて、このリストを下から上に探し始めます。もしマッチするトランジションがなければ、Wwiseはデフォルトの"Any to Any"トランジションを使用します。

## トランジションの追加

Music Switch ContainerやPlaylist ContainerのTransition Matrixは、必ず最低でもAny to Anyのトランジションは入っています。より具体的なトランジションルールを作成する場合は、Transition Matrixで直接、設定する必要があります。

### Music Switch Containerや、Playlist Containerに、トランジション設定を追加するには:

1. Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
2. Transitionsタブに切り替える。
3. Addをクリックする。

Transition Matrixに、AnyからAnyへの新規トランジションが追加される。

4. Source列で、セレクタボタン (>>) をクリックして、以下のオプションを1つ選択する。
  - ・ **Any:** ソース (Source) を特に指定せず、どのミュージックオブジェクトでも使えるトランジションを作成する場合。
  - ・ **Nothing:** ソース (Source) がない、つまりミュージックオブジェクトが再生されていない状態から始まるトランジションを作成する場合。
  - ・ **Browse:** 特定のミュージックオブジェクト、またはVirtual Folderに入った複数のオブジェクトを、ソース (Source) に指定して、トランジションを作成する場合。Project Explorer - Browserが開くので、ソースを選択する。

**Tip**

Transition Matrixに、Project Explorerのミュージックオブジェクトを直接ドラッグすることもできます。ただし、マトリックスに追加できるミュージックオブジェクトはProperty Editorにロードされたコンテナの子に限られるので、注意してください。

5. Destination列で、セレクタボタン (>>) をクリックして、以下のオプションを1つ選択する。
  - **Any:** デスティネーション (Destination) を特に指定せず、どのミュージックオブジェクトでも使えるトランジションを作成する場合。
  - **Nothing:** デスティネーション (Destination) がない、つまりミュージックオブジェクトが再生されていない状態へ移るトランジションを作成する場合。
  - **Browse:** 特定のミュージックオブジェクト、またはVirtual Folderに入った複数のオブジェクトを、デスティネーション (Destination) に指定して、トランジションを作成する場合。Project Explorer - Browserが開くので、デスティネーションを選択する。

カスタマイズされた新規トランジションが、Transition Matrixに追加される。

**Tip**

表中のトランジションを1つ選び、上下にドラッグすることで、順番を変えられます。このトランジションが挿入される場所が、赤線で表示されます。なお"Any to Any"トランジションを常にWwiseが最後に確認するので、Transition Matrixの一番上の位置から移動できません。

## 関連トピック

- 「トランジションの削除」
- 「ソースやデスティネーションのプロパティを設定する」
- 「トランジションのコピー、ペースト」
- 「トランジションセグメントの使用」

## トランジションのコピー、ペースト

トランジション作成の作業を早めるために、マトリックスMatrix内でトランジションルールを簡単にコピーしたりペーストしたりできます。

**Music Switch Containerや、Playlist Containerで、設定したトランジションをコピーするには:**

1. Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
2. Transitionsタブに切り替える。
3. Transition Matrixのコピーしたいトランジションを右クリックして、メニューでCopyを選択する。

- リスト上で、コピーしたトランジションを挿入する場所を右クリックして、メニューで、Pasteを選択する。

Transition Matrixの指定した場所に、トランジションのコピーが追加される。



#### Tip

マトリックス内でトランジションのコピーやペーストを行うには、ショートカットキー（**Ctrl+C**、**Ctrl+V**）も使えます。

### 関連トピック

- 「トランジションの追加」
- 「ソースやデスティネーションのプロパティを設定する」
- 「トランジションセグメントの使用」

## トランジションの削除

Transition Matrixで設定したトランジションルールが不要となった場合は、簡単に削除できます。トランジションルールを削除しても、そのソース（Source）やデスティネーション（Destination）に設定したミュージックオブジェクトは、削除されません。

TMusic Switch Containerや、Playlist Containerから、設定したトランジションを削除するには：

- Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
- Transitionsタブに切り替える。
- Transition Matrixで削除したいトランジションを選択する。
- Removeをクリックするか、Deleteキーを押す。

Transition Matrixから、選択したトランジションが削除される。

### 関連トピック

- 「トランジションの追加」
- 「ソースやデスティネーションのプロパティを設定する」
- 「トランジションのコピー、ペースト」
- 「トランジションセグメントの使用」

## ソースやデスティネーションのプロパティを設定する

デフォルトのトランジションは、1つのミュージックオブジェクトから別のものへの単純な切り替わりです。しかしトランジションの真の強みは、ソース（Source）やデスティネーション（Destination）のカスタマイズにより、ユニー

クな楽節が生み出された時に発揮されます。ソースやデスティネーションのプロパティ設定により、オブジェクト間のトランジションがスムーズな音楽として聞こえます。

オブジェクト間のトランジションに対するコントロール性や柔軟性を高めるために、ソースとデスティネーションの、エグジットポイント (Exit point) やエントリポイント (Entry point) を、いくつかの選択肢から選べます。また、デスティネーションのオブジェクトのエントリポイントを、ランダムに選択することも可能です。こうすることでトランジションが再生されるたびに、違うエントリポイントが使用されるので、同じトランジションの繰り返しが減ります。

ソースやデスティネーションのプロパティ設定は、1つのトランジションに限定された設定であることに注意してください。例えばTransition MatrixでHappy MusicからSad Musicへのトランジションと、Happy MusicからScary Musicへのトランジションが設定されているとします。ここで、ソースの「Happy Music」からのエグジットポイントを毎回、最初のカスタムキュ (Custom cue) とするには、両方のトランジションで、別々に設定する必要があります。



### 注記

ソースやデスティネーションに設定できるプロパティは多数ありますが、利用できるプロパティ設定は、使用するオブジェクトの種類や、それが入っているコンテナの種類によって、異なります。

### ソース (Source) のプロパティを設定するには:

1. Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
2. Transitionsタブに切り替える。
3. Transition Matrixのトランジションを1つ選択する。

これで、このトランジションのソースのプロパティを編集できる。

4. ソースが、Switch Containerに入っていれば、**Exit source at** (ソースをエグジットする方法) リストにある、以下のオプションを1つ選択する。

**Immediate:** ソースの再生をすぐに止める。

**Next Grid:** ソースの再生を、次のグリッドで止める。グリッドとは、ミュージックオブジェクトを任意の位置でバーチャルに区切る方法。

**Next Bar:** ソースの再生を、次の小節で止める。

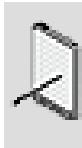
**Next Beat (次のビート):** ソースの再生を、次のビートで止める。

**Next Cue:** 次のキュがカスタムキュでもエグジットキュでも、ソースの再生を、次のキュで止める。

**Next Custom Cue:** ソースの再生を、次のカスタムキュで止める。再生中のMusic Segmentにカスタムキュがなければ、次のセグメントに入り、カスタムキュを見つけるまで再生する。

**Exit Cue:** ソースの再生を、エグジットキューで止める。

5. Next CueもしくはNext Custom Cueを選択した場合には、**Match** エディットボックスにキューネームを入力してトランジションが発生する可能性のある有効なキューナーの選択をより正確にすることができる。
6. トランジション中に、ソースのポストエグジットを再生させるには、**Play post-exit**を選択する。



#### 注記

ソースのポストエグジットが再生されるのは、そのソースがエグジットキューでまだ続く場合と、エグジットキュー以降でフェードアウトする場合だけです。これ以外の場合は、トランジション中にポストエグジットが再生されることはありません。

7. ソースの終わりをフェードアウトさせるには、**Fade-out**を選択する。

### デスティネーション (Destination) のプロパティを設定するには:

1. Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
2. Transitionsタブに切り替える。
3. Transition Matrixのトランジションを1つ選択する。

これで、このトランジションのデスティネーションのプロパティを編集できる。

4. デスティネーションが、Music Playlist Containerの場合、最初に再生するアイテム (Item) を、Playlist Containerから選択できる。Browseボタン (...) をクリックして、Jump to playlist item (プレイリストのアイテムにジャンプする) オプションの対象として設定するアイテムを選択する。
5. このトランジションがSwitch Containerに関するものであれば、Sync to (同期の対象) リストにある、以下のオプションを1つ選択する。

**Entry Cue:** デスティネーションの再生を、エントリキューから始める。

**Same Time as Playing Segment:** デスティネーションの再生を、ソースセグメントと同じタイムマークから始める。例えば、ソースセグメントが最初から10秒間、再生された場合は、デスティネーションの再生は、最初から10秒後の時点から始める。

**Random Cue** デスティネーションの再生を、ランダムに選択されたキューフrom から始める。このオプションを選択すると、全てのキューナーが選択肢となり、エントリキュー やカスタムキューナーも対象となる。

**Random Custom Cue** デスティネーションの再生を、ランダムに選択されたカスタムキューナーから始める。このセグメントにカスタムキューナーがなければ、エントリキューが使用される。

6. Random CueもしくはRandom Custom Cueを選択した場合には、Custom Cue Filterで有効な開始ポジションの選択をさらに正確にできる。

**Match:** この名前を持つキューナーだけが、開始ポジションとして選択できます。

**Match source cue name** (マッチするソースキューネーム) : このトランジションのためのソースセグメントに使われているキューと同じ名前を持つキューのみが、開始ポジションとして選択できます。

7. トランジション中に、デスティネーションのプレエントリを再生させるには、**Play pre-entry**を選択する。
8. デスティネーションの再生をフェードインで開始するには、**Fade-in**を選択する。

## 関連トピック

- 「トランジションの追加」
- 「トランジションの削除」
- 「トランジションのコピー、ペースト」
- 「トランジションセグメントの使用」

## フェード (Fade) の編集

フェードイン (Fade-in) やフェードアウト (Fade-out) は、それぞれデスティネーション (Destination) やソース (Source) のミュージックオブジェクトに設定できる、特別なプロパティです。また、トランジションセグメントの最初や最後にも、適用できます。フェードを使用することで、ミュージックオブジェクトの始まり方や終わり方が洗練され、トランジションをスムーズに行えます。フェードごとに、その長さやオフセットを設定できるほか、サウンドをさらにカスタマイズするため、フェードのカーブ形状も設定できます。

### フェードアウト (Fade-out) を編集するには:

1. Property EditorのTransitionsタブで、**Fade-out**が選択されていることを確認した上で、Editをクリックする。

Music Fade Editor が開く。

2. Timeフィールドに、フェードアウト時間を入力する。
3. Offsetフィールドに、エグジットキュー (Exit cue) から、フェードアウト終了までの時間を、入力する。
4. Curveリストから、フェードアウト用のカーブ形状を1つ選択する。

設定した通りにフェードアウトします。

### フェードイン (Fade-in) を編集するには:

1. Property EditorのTransitionsタブで、**Fade-in**が選択されていることを確認した上で、Editをクリックする。

Music Fade Editor が開く。

2. Timeフィールドに、フェードイン長さを入力する。
3. Offsetフィールドに、エントリキュー (Entry cue) と、フェードイン開始の間の時間を、入力する。

#### 4. Curveリストから、フェードイン用のカーブ形状を1つ選択する。

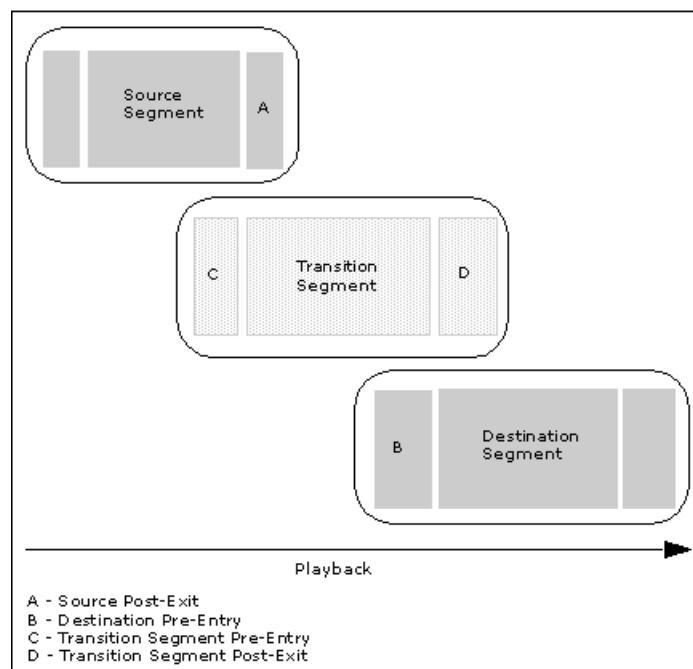
設定した通りにフェードインします。

#### 関連トピック

- 「トランジションの追加」
- 「トランジションの削除」
- 「ソースやデスティネーションのプロパティを設定する」
- 「トランジションのコピー、ペースト」
- 「トランジションセグメントの使用」

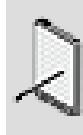
### トランジションセグメントの使用

場合によって、ソース (Source) の終わり部分とデスティネーション (Destination) の始まり部分に重ねて、別のミュージックを再生した方が、より良いトランジションに聞こえることがあります。この橋渡しとなるミュージックを、トランジションセグメントと呼び、Wwiseの全てのトランジションに設定できます。ミュージックトランジションにおいて、ソースとデスティネーションの間にトランジションセグメントを1つ再生する様子を、下図に示します。



また、ソース、デスティネーション、トランジションセグメントの、それぞれのプレエントリ (Pre-entry) やポストエグジット (Post-exit) の部分を、自由に組み合わせることで、さらにシームレスなトランジションを作成できます。

どのようなMusic Segmentも、トランジションセグメントとして使えます。セグメントの作成方法は[25章Music Track、Music Segmentの活用](#)を参照してください。



### 注記

あるトランジションセグメントを含むWork Unitが、プロジェクトからアンロードされてしまうと、そのトランジションセグメントは黄色でハイライト表示されます。

## トランジションセグメントを使うには:

1. Music Switch ContainerまたはPlaylist Containerを1つ、Property Editorにロードする。
2. Transitionsタブに切り替える。
3. Transition Matrixのトランジションを1つ選択する。
4. **Use transition segment**を選択する。
5. 以下のいずれかを行う:
  - セグメントを1つ、Project Explorerから、**Transition Segment**フィールドに、ドラッグする。



- **Browse**ボタン (...) をクリックして、Project Explorer - Browserでセグメントを1つ選択する。
6. トランジションセグメントの、プレエントリ (Pre-entry) 部分を再生させるには、**Play transition pre-entry**を選択する。
  7. トランジションの最初の部分をフェードイン (Fade-in) させるには、**Fade-in**を選択する。
  8. トランジションセグメントの、ポストエグジット (Post-exit) 部分を再生させるには、**Play transition post-exit**を選択する。
  9. トランジションの再生が終わる時にフェードアウト (Fade-out) させるには、**Fade-out**を選択する。



### 注記

トランジションセグメントの、フェードインやフェードアウトのプロパティを編集するには「[フェード \(Fade\)](#)」を参照してください。

## 関連トピック

- 「トランジションの追加」
- 「トランジションの削除」
- 「トランジションのコピー、ペースト」
- 「ソースやデスティネーションのプロパティを設定する」

## インタラクティブミュージックのトランジションに関する、コツとベストプラクティス

インタラクティブミュージックは複雑なツールで、選択肢も多様です。プロジェクトの初期段階で、インタラクティブミュージックに関する一貫した分かりやすい方

針を立てることは、以後の時間節約や効率性につながります。インタラクティブミュージックプロジェクトの進め方には様々な手法があり、ゲームで最高の結果を達成するために、Wwiseの使い方を自由に工夫してください。インタラクティブミュージックのトランジションを効率的に管理するための提案を、以下に示します。

### トランジションの順番

Wwiseは、トランジションが必要な時に現状に適した項目を求めてTransition Matrixを下から上へと順に探します。条件に合うトランジションが見つかれば、それが一番良いトランジションであるかどうかに関わらず、探すのをやめます。常に最適のトランジションが適用されるようにするには、下図のようにTransition Matrixに設定したルールを汎用性のあるルールから具体的なルールの順に上から下へ並べます。

Source	Destination
>> Any	>> Any
>> Any	>> Exciting Music
>> Exciting Music	>> Any
>> Casual Music	>> Creepy Music
>> Creepy Music	>> Casual Music

ソース (Source) やデスティネーション (Destination) としてAnyまたはNothingが設定されているトランジションは、具体的なミュージックオブジェクト名が入ったトランジションよりも、上に配置します。こうすれば、Wwiseが、汎用性のあるルールよりも先に、具体的なルールを見つけます。トランジションの作成や並べ方については「トランジションの追加」を参照してください。

---

## 第29章 スティンガーの活用

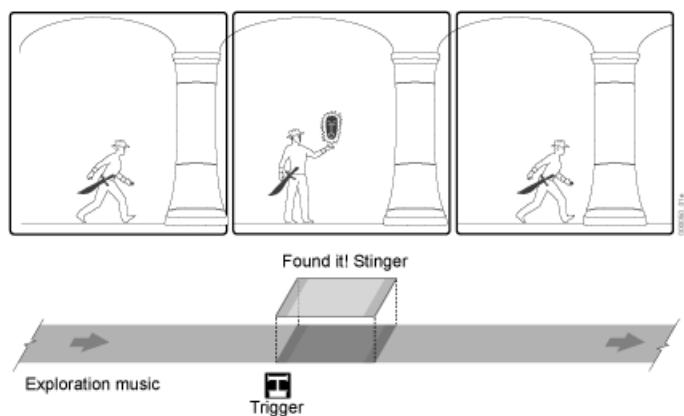
概要 .....	508
スティンガー (Stinger) の追加 .....	509
スティンガー (Stinger) の再生方法の設定 .....	511
スティンガーの削除 .....	512
スティンガー (Stinger) の試聴 .....	512

## 概要

インタラクティブミュージックを使ったフィードバックの範囲を広げるために、ゲームアクションの主要なポイントで、スティンガー (Stinger) を再生させることもできます。スティンガーとは、再生中の音楽に重ねてミックスされる、短いミュージックフレーズのことです。ゲームが、スティンガーミュージックセグメント (Stinger music segment) に関連したTriggerを呼び出すことで、スティンガーが再生されます。Triggerの使用については[20章Triggerの使い方](#)を参照してください。

### スティンガー (Stinger) の活用例

例えば、前例のさっそうとした考古学者が、古い宝や冒險を求めて寺院の中を熱心に探検しているとします。彼が周囲を探りながら歩き回る間は「Exploration music (探検ミュージック)」が再生されます。そのうちに宝を見つけると、素晴らしい発見を強調するかのように、別の短いミュージックが再生されます。ここでは、探検ミュージックが他のミュージックに切り替わるのではなく、ゲーム側が呼び出したTriggerによって「Found it (宝を見つけた)」というスティンガーが、再生中の探検ミュージックに重ねて再生され、この特別な瞬間が表現されます。スティンガーが終了すると、探検ミュージックの再生は続きます。



再生中のミュージックの上に、スティンガーを重ねるには、以下を行う必要があります。

- 1つのTriggerに、1つのミュージックオブジェクトを結び付ける。
- 1つのTriggerに、1つのMusic Segmentをマッピングすることで、スティンガーを作成する。
- スティンガーの再生方法を設定する。

### Interactive Music Hierarchyにおける、スティンガー (Stinger) の使用

スティンガーは、階層の様々なレベルで作成できるので、同じTriggerを、複数のセグメントにアサインできます。つまり、例えば、最上位のミュージックオブジェクトが「Headshot (頭部へのショット)」というTriggerを使い、その子である全てのオブジェクトも、同じTriggerの「Headshot」を使えます。このような場合、子オブジェクトのTrigger「Headshot」に対して、親レベルで設定したセグメント

とは異なるセグメントを、ステインガーとして設定します。これによって自動的に、親の「Trigger・ステインガー」の関係がオーバーライドされるので、ゲームの重要なポイントで再生できるステインガーの選択肢が増えます。ミュージックオブジェクトの階層において、1つのTriggerで再生できるステインガーは1つなので、再生されるのは、現在、再生中の子オブジェクトに設定されたステインガーだけとなります。

ステインガーを使った作業には、以下のタスクが伴います:

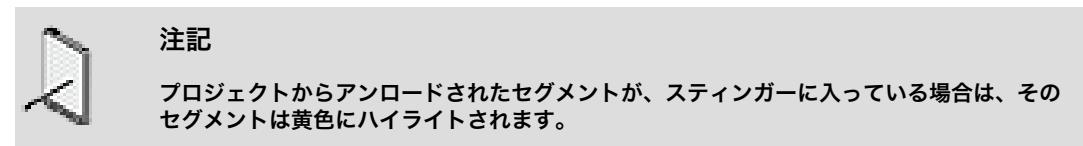
- ・「ステインガー (Stinger) の追加」
- ・「ステインガー (Stinger) の再生方法の設定」
- ・「ステインガー (Stinger) の試聴」
- ・「ステインガーの削除」

## ステインガー (Stinger) の追加

ミュージックオブジェクト用のステインガーは、Music Object Property EditorのStingersタブで、作成します。Wwiseでステインガーを追加するには、以下の2種類の方法があります:

- ・Project Explorerから、ドラッグ&ドロップする
- ・Property Editorのボタンを使う

この2種類の方法を組み合わせて使うことも、どちらか一方だけを使うことも、可能です。



### Project Explorerから、ステインガー (Stinger) をドラッグ&ドロップして、挿入するには

1. Property Editorに、ミュージックオブジェクトを1つロードして、Stingersタブに切り替える。
2. Project Explorerの、Game Syncsタブから、Triggerを1つドラッグして、ステインガーのリストに入れる。

Project ExplorerからドラッグしたTriggerを使って、自動的にステインガーが作成されます。このTriggerはデフォルトとして、どのMusic Segmentにも関連付けられていない（Nothing）。

3. このTriggerにMusic Segmentをアサインするには、Project Explorerの、Audioタブに切り替えて、Music Segmentを、Segment to Play（再生するセグメント）列までドラッグして入れる。

このMusic Segmentが、デフォルトで追加された"Nothing"オプションを置き換える、Triggerに関連付けられる。

4. Play At (再生するタイミング) 列で、以下のオプションを1つ選択して、このセグメントをいつ再生するのかを設定する。

- **Immediate:** 直ちに切り替え。ルックアヘッドタイム (Look-ahead time) 付きのトラックが設定されている場合は、ステインガーはその設定時間が過ぎてから、再生される。
- **Next Grid (次のグリッド)** - スティンガーの再生を、次のグリッドで始める。グリッドは、ミュージックオブジェクトをバーチャルに分割する、任意の周波数。
- **Next Bar (次のバー)** - スティンガーの再生を、次の小節で始める。
- **Next Beat (次のビート)** - スティンガーの再生を、次のビートで始める。
- **Next Cue (次のキュー)** - スティンガーの再生を、次のキューで始める。キューとは、エントリキュー、エグジットキュー、またはカスタムキューである。
- **Next Custom Cue (次のカスタムキュー)** - スティンガーの再生を、次のカスタムキューで始める。
- **Entry Cue (エントリキュー)** - スティンガーの再生を、エントリキューで始める。
- **Exit Cue (エグジットキュー)** - スティンガーの再生を、エグジットキューで始める。

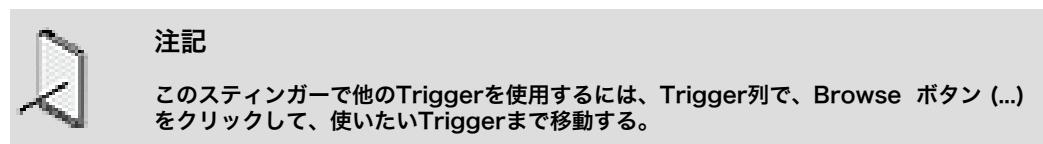
5. 必要に応じて上記2~4を繰り返し、ステインガーをさらに追加する。

Property Editorのボタンを使って、ステインガー (Stinger) を挿入するには:

1. Property EditorのStingersタブで、Addをクリックする。

Project Explorer - Browserダイアログボックスが開く。

2. ミュージックオブジェクトにアサインするTriggerを選択して、OKをクリックする。



3. Triggerを選択して、OKをクリックする。

このTriggerが、ステインガーに追加される。

4. Segment to Play列でBrowseボタン(...)をクリックする。

Project Explorer-Browserが開きます。

5. 使いたいセグメント (Segment) まで移動して、OKをクリックする。

このセグメントが、ステインガーに追加される。

6. • **Immediate:** 直ちに切り替え。ルックアヘッドタイム (Look-ahead time) 付きのトラックが設定されている場合は、ステインガーはその設定時間が過ぎてから、再生される。

- **Next Grid** (次のグリッド) - スティンガーの再生を、次のグリッドで始める。グリッドは、ミュージックオブジェクトをバーチャルに分割する、任意の周波数。
- **Next Bar** (次のバー) - スティンガーの再生を、次の小節で始める。
- **Next Beat** (次のビート) - スティンガーの再生を、次のビートで始める。
- **Next Cue** (次のキュー) - スティンガーの再生を、次のキューで始める。キューとは、エントリキュー、エグジットキュー、またはカスタムキューである。
- **Next Custom Cue** (次のカスタムキュー) - スティンガーの再生を、次のカスタムキューで始める。
- **Entry Cue** (エントリキュー) - スティンガーの再生を、エントリキューで始める。
- **Exit Cue** (エグジットキュー) - スティンガーの再生を、エグジットキューで始める。

7. 必要に応じて、スティンガーをさらに追加する。

## 関連トピック

- 「[スティンガー \(Stinger\) の再生方法の設定](#)」
- 「[スティンガーの削除](#)」
- 「[スティンガー \(Stinger\) の試聴](#)」

## スティンガー (Stinger) の再生方法の設定

1つのミュージックオブジェクト用に、複数のスティンガーを作成できたら、ゲーム中におけるこれらのスティンガーの再生を管理するための設定を、決めます。スティンガーを最大限に活かして使うために、再生に関して考慮すべき点を2つ、以下に示します。

- 同じスティンガーを、再び再生するまでにあける時間。スティンガーは、ミュージックスコアを盛り上げるために使いますが、同じスティンガーを、間をあけずに何度も再生すると、インパクトが薄れてしまうので、避けてください。
- スティンガーは、ミュージックスコアを盛り上げるために使いますが、同じスティンガーを、間をあけずに何度も再生すると、インパクトが薄れてしまうので、避けてください。ただし、プレイリストにある次のセグメントの、事前に設定した最初の機会に、このセグメントを再生することができます。なお、このオプションを選択しないと、このスティンガーは再生されません。

## スティンガー (Stinger) の再生設定 (Playback settings) を設定するには:

1. Property Editorの、Stingersタブで、**Don't play this Stinger again for x seconds** (x秒の間、このスティンガーを再び再生しない) フィールドに、そのStingerを再び再生するまでの間隔を、秒単位で入力する。指定された秒数が経過する前に、Triggerが同じスティンガーを呼び出した場合は、このTriggerは無視される。



#### 注記

「Don't play stinger again for (このスティンガーを再生しない)」に設定する時間は、スティンガーの同期ポイントから経過した時間、つまりスティンガーのエントリーキュー (Entry cue) が発生した瞬間から経過した時間のことです。

2. 現在のセグメントで、そのスティンガーを再生する時間が充分がない場合に、次のセグメント中に再生させるには、Allow playing the stinger in next segment (次のセグメントでスティンガーを再生する) オプションを選択する。



#### 注記

本オプションを選択しないと、このスティンガーは再生されません。

### 関連トピック

- ・「スティンガー (Stinger) の追加」
- ・「スティンガーの削除」
- ・「スティンガー (Stinger) の試聴」

## スティンガーの削除

不要となったスティンガーは、削除できます。

スティンガー (Stinger) リストから、スティンガーを削除するには:

1. Stingersタブで、削除するスティンガーを選択する。
2. Removeをクリックする。

選択したスティンガーが、リストから削除されます。



#### 注記

スティンガーを削除しても、それに関連したTriggerや、Music Segmentには、影響しません。

### 関連トピック

- ・「スティンガー (Stinger) の再生方法の設定」
- ・「スティンガー (Stinger) の追加」
- ・「スティンガー (Stinger) の試聴」

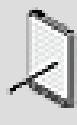
## スティンガー (Stinger) の試聴

Stingerセグメントを作成してから、Transport Controlでそれを単独に視聴したり、別のセグメントと共に視聴して他のミュージックの上に重ねて再生した時のインパクトを確認したりできます。スティンガーは、階層の様々なレベル

で作成できるので、複数のセグメント設定で、同じTriggerが使われている場合があります。ただし、ミュージックオブジェクトの階層において、1つの特定のTriggerによって再生できるのは、1つのスティンガーに限られ、どのスティンガーを再生するかは、子オブジェクトによって決まります。例えば、Music Switch Containerレベルで、「Headshot（頭部へのショット）」というTriggerを使うスティンガーを作成した場合、同じ「Headshot」Triggerを、子オブジェクトのPlaylist Containerにあるスティンガーにも使っている場合は、このMusic Switch ContainerをTransport Controlにロードして、Triggerを呼び出すと、Playlist Containerのスティンガーだけが再生されます。

さらに、Soundcasterでシミュレーションを作成して、そのシミュレーションをProfiler機能でプロファイリングして、パフォーマンス問題をモニターできます。シミュレーションやプロファイリングについては、以下を参照してください。

- ・「[シミュレーションを構築する](#)」
- ・「[Performance Monitorを使った、モニタリングとトラブルシューティング](#)」



### 注記

プロジェクトからアンロードされたセグメントが、スティンガーに入っている場合は、セグメントは黄色にハイライトされます。このスティンガーセグメントが入ったワークユニットを再びプロジェクトにロードするまで、このスティンガーの試聴ができません。

Transport Controlで、スティンガー（Stinger）を試聴するには：

1. Transport Controlに、スティンガーセグメント（Stinger segment）をロードする。



2. Play アイコンをクリックする。

このスティンガーセグメントが再生される。

再生中のミュージックに、スティンガー（Stinger）を重ねて試聴するには：

1. Transport Controlにミュージックオブジェクトを1つロードする。
2. Playアイコンをクリックする。

Transport Controlにロードされたミュージックオブジェクトが再生される。

3. ゲームシンクのエリアで、Triggersボタンをクリックして、Triggerリストを表示させる。



4. Call Triggerアイコンをクリックする。

該当するスティンガーが、再生中のミュージックオブジェクトに重ねて再生される。さらに他のTriggerを自由に選択して、それぞれのスティンガーを再生

させることで、ゲーム中に様々なスティンガーが再生される様子をシミュレーションできる。

### 関連トピック

- ・「スティンガー (Stinger) の再生方法の設定」
- ・「スティンガー (Stinger) の追加」
- ・「スティンガーの削除」

**audio**kinetic

## パート VI. プロジェクト を仕上げる



---

30. アウトプットの管理 .....	518
概要 .....	519
オブジェクトのアウトプットルーティングの指定 .....	519
User-Defined Auxiliary Send (ユーザーが定義するAUXセンド) を使う .....	521
Game-Defined Auxiliary Sendを使う .....	522
ラウドネスノーマライゼーション、またはメイクアップゲインを使い、ボリュームを調整する .....	523
ボイスパイプラインを理解する .....	524
組み込みオーディオデバイス .....	526
セカンダリーアウトプットを理解する .....	528
HDRを理解する .....	533
HDRの使用 .....	537
HDRの追加情報 .....	546
ファイナルミックスの作成 .....	550
31. プラットフォームとランゲージのバージョン管理 .....	567
概要 .....	568
複数のプラットフォームに対応するオーサリング .....	568
プロジェクトのローカリゼーション作業 .....	589
バージョンに関する、コツとベストプラクティス .....	594
32. シミュレーションの作成 .....	602
概要 .....	603
シミュレーションを構築する .....	604
シミュレーション中の再生の管理 .....	607
ゲームシンクを使ったシミュレーション .....	610
シミュレーションを使った、プロパティの最終調整 .....	613
シミュレーション作成に関する、コツとベストプラクティス .....	616
33. Wwiseにおけるメモリ管理 .....	618
概要 .....	619
Memory Managerの各コンポーネントについて .....	619
各種メモリプールのサイズを設定する .....	620
メモリ関連のトラブルシューティング .....	622
メモリプールの最適化 .....	623
メモリ管理に関する、コツとベストプラクティス .....	628
34. Profiling .....	630
概要 .....	631
Wwiseにおける、プロファイリングの種類を理解する .....	632
ローカルまたはリモートで、ゲーム機に接続する .....	637
サウンドエンジンからくるデータをキャプチャーする .....	640
Performance Monitorを使った、モニタリングとトラブルシューティング .....	649
Game Object Explorerで、オブジェクトやリスナーをトラッキングする .....	652
Game Object 3D Viewerで、オブジェクトを観察する .....	656
Game Sync Monitorで、ゲームシンクを評価する .....	660
プロファイリングに関する、コツとベストプラクティス .....	661
35. SoundBankの管理 .....	664
概要 .....	665
ゲームにSoundBankをロードする方法を理解する .....	667

---

SoundBankの構築 .....	672
SoundBankの管理 .....	689
SoundBankのカスタム属性を設定する .....	692
プロジェクトの各種SoundBankの生成 .....	700
CopyStreamedFilesツールの利用 .....	704
SoundBank管理の戦略 .....	705
SoundBankに関する、コツとベストプラクティス .....	718
36. ファイルパッケージの管理 .....	722
概要 .....	723
File Packagerプロジェクトの使い方 .....	723
プロジェクト内のファイルパッケージの管理 .....	726
DLC（ダウンローダブルコンテンツ）の概要 .....	730
ファイルパッケージを生成する .....	731
コマンドラインで、File Packager引数を使用 .....	732
ファイルパッケージャに関する、コツとベストプラクティス .....	735

---

## 第30章 アウトプットの管理

概要 .....	519
オブジェクトのアウトプットルーティングの指定 .....	519
User-Defined Auxiliary Send (ユーザーが定義するAUXセンド) を使う .....	521
Game-Defined Auxiliary Sendを使う .....	522
ラウドネスノーマライゼーション、またはメイクアップゲインを使い、ボリュームを調整する .....	523
ボイスパイプラインを理解する .....	524
組み込みオーディオデバイス .....	526
セカンダリーアウトプットを理解する .....	528
HDRを理解する .....	533
HDRの使用 .....	537
HDRの追加情報 .....	546
ファイナルミックスの作成 .....	550

## 概要

全てのオブジェクトが入ったプロジェクト階層の上にあるのがMaster-Mixer Hierarchyです。この別個のバス階層は、[9章 アウトプットバス構造の構築](#)で前述の通りプロジェクト階層にある様々な構造をまとめるために使えます。マスター Audio Bus（マスターオーディオバス）を、メインのサウンドミックスに使いまします。ゲームのセカンダリアウトプットであるチャットヘッドフォンや、ゲームコントローラのモーションなどには、別のバス構造を作成するべきです。

これらのマスターバスはプロジェクトのをコントロールする最終的なレベルを提供します。バスはプロジェクト階層の上にあるので、ゲームの最終ミキシングに利用できます。

一部のプラットフォームでは、いくつかのエフェクトをマスターバスに適用できるので、ゲームに求められる没入感を仕上げるのに利用できます。

## 関連トピック

- ・「[概要](#)」
- ・「[バスのプロパティ設定](#)」
- ・「[バス階層の構築例](#)」

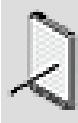
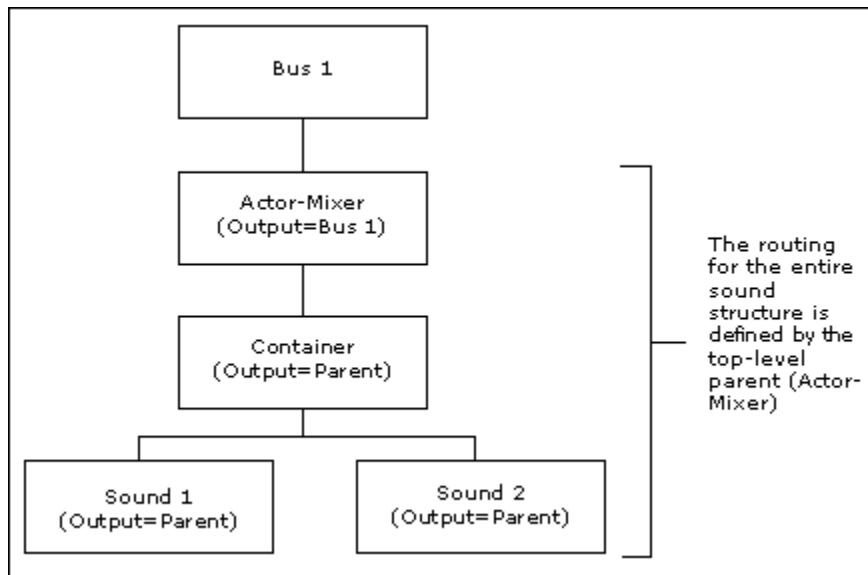
## オーディオアウトプットのフォーマット

Wwiseの最終ミックスは、既定のLPCMチャンネルとしてアウトプットされます。なお、Wwiseでは、マルチチャンネルエンコーディングを行わず、LPCMデータをゲーム機やシステムに送るだけです。ゲーム機またはシステムがLPCMデータを受信すると、Dolby、DTS、DPL2など、そのゲーム機やシステムが対応するほぼ全てのフォーマットで、アウトプットできます。

## オブジェクトのアウトプットルーティングの指定

階層中のオブジェクトは全て、必ずバスにルーティングされます。構造全体のルーティングを一括して設定するには、最上位の親オブジェクトのルーティングを設定するだけです。アウトプットルーティングは絶対プロパティなので、設定は自動的に下にある全ての子オブジェクトに継承されます。ただし、必要に応じて親の設定をオーバーライドできます。

親オブジェクトのアウトプットルーティング設定が自動的に全ての子オブジェクトに継承される様子を、下図に示します。

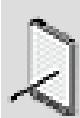


### 注記

デフォルトとして、Default User Settingsで指定したバスに自動的にオブジェクトがルーティングされます。

オーディオオブジェクトのアウトプットルーティングを指定するには:

1. Property Editorに、最上位オブジェクトを1つロードする。



### 注記

このオブジェクトが最上位オブジェクトでない場合、Outputオプションを設定する前にOverride parentオプションを選択する必要があります。

2. グループボックスOutput Busで、Browseボタン(...)をクリックする。

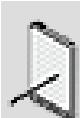
Project Explorer - Browserが開く。

3. このオブジェクトが通るAudio Busを選択する。
4. OKをクリックする。

現在のオブジェクトとその下にある全ての子オブジェクトに、選択したAudio Busを適用する。

現在のプラットフォームのアウトプットルーティングをアンリンクするには:

1. Property Editorに、最上位オブジェクトを1つロードする。



### 注記

このオブジェクトが最上位のオブジェクトでない場合、Outputオプションを設定する前にOverride parentオプションを選択する必要があります。

2. グループボックスOutput Busで現在のバス名を右クリックして、コンテキストメニューを開く。
3. コンテキストメニューでUnlinkedを選択する。

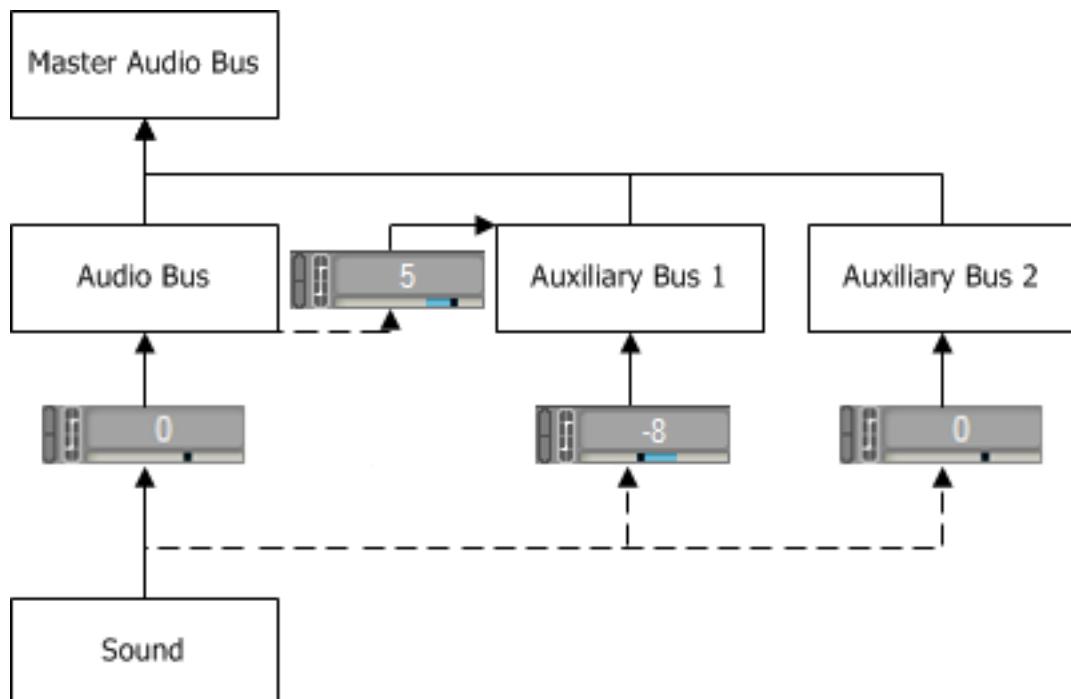
バスルーティングがアンリンクされるので、このプラットフォームに別のバスを指定できる。

## User-Defined Auxiliary Send（ユーザーが定義するAUXセンド）を使う

オーディオのルーティングに指定したAudio Bus以外にもUser-Defined Auxiliary Send（ユーザーが定義したAUXセンド）を、1つのActor-Mixerオブジェクトやインタラクティブミュージックオブジェクトに対して最大4つ、指定することができます。同様に、Audio Bus自体は最大4つのUser-Defined Auxiliary Sendを持つことができます。AUXセンドとは、オーディオ信号の一部を追加のバス（補助バス）に送信して並列処理させる方法です。

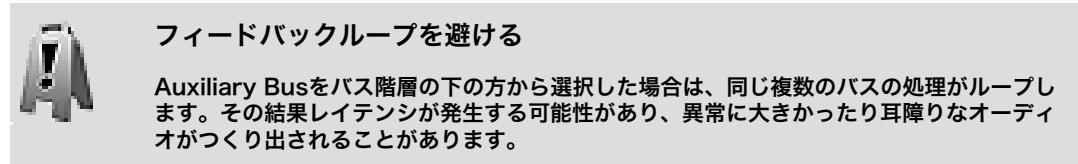
User-defined auxiliary sendsは静的なAUXセンドをオーサリング作業で直接設定するために使い、ゲーム側がダイナミックに設定してコントロールすることの多い、Game-defined auxiliary sends（ゲーム定義のAUXセンド）と異なります。

次のイラストは、AUXセンドの減衰量が5 dBで、AUXセンドが2つあるAudio Busにルーティングされたサウンドを示しています。どちらのAUXセンドもボリューム減衰があり、Auxiliary Busにルーティングされます。



## User-Defined Auxiliary Sendを追加するには:

1. Project Explorerでオブジェクトをダブルクリックして、そのオブジェクトのPropertiesを確認する。
2. Project ExplorerにあるAUXバスをUser-Defined Auxiliary Sendsリストにドラッグ&ドロップする、または  
[...]ボタンをクリックして、Project Explorerセレクタを使いAuxiliary Busを選択する。
3. 選択したAuxiliary BusへのSend volumeを設定する。



## User-Defined Auxiliary Sendを使い、ゲーム環境をコントロールする

AUXセンドをシンプルなシナリオのゲーム環境をコントロールするために利用できます。多くの場合、環境は閉鎖された空間のリバーブやアーリーリフレクションによって表現されます。サウンドエミッタごとに、以下のエレメントをゲーム側がコントロールできます。

- **Send volume:** ウェット部分、つまり反射音に該当するように設定できる。
- **Output bus volume:** ドライ部分、つまり直接音に該当するように設定できる。
- **Output bus low pass filter:** サウンドに対するオブストラクションとオクルージョンのレベルに該当するように設定して、直接音つまりドライ部分の周波数反応に影響を与えることができる。

上記の最初の2つの値にGame Parameterを設定することで、ゲームオブジェクトごとにドライシグナルとウェットシグナルの量をコントロールできます。ゲーム側は、リスナーとサウンドエミッタの間の距離を計算して、Game Parameterにこの数値をアサインします。Game Parameter設定でRTCPカーブを設定して、信号のウェット部分とドライ部分が距離の変化と共にどのように変わるべきを定義できます。

アウトプットバスLPFに対してGame Parameterを設定することで、サウンドのオクルージョンやオブストラクションの程度をゲーム側の数値を使ってコントロールできます。

## Game-Defined Auxiliary Sendを使う

User-defined auxiliary sends（「ユーザー定義のAUXセンド」）4つ以外に、最大4つのGame-defined auxiliary sends（「ゲーム定義のAUXセンド」）を追加できます。ゲーム側で定義するAUXセンドは、主に以下のWwise SDK機能を使って、ゲームによってコントロールされます。

- [AK::SoundEngine::SetGameObjectAuxSendValues\(\)](#)

ゲームオブジェクトのAuxiliary BusへのSend volumeを設定する場合に、この関数を使う。よくウェットボリュームと呼ばれる。

- `AK::SoundEngine::SetGameObjectOutputBusVolume()`

OutputBusVolumeを設定する場合に、この関数を使う。よくドライボリュームと呼ばれる。

なお、オブジェクトのGame-Defined Auxiliary Send機能を利用するには、**Use game-defined auxiliary sends**オプションを有効にする必要があります。

### Game-Defined Auxiliary Sendを有効にするには:

1. Project Explorerでオブジェクトをダブルクリックして、オブジェクトの詳細を確認する。
2. Property Editorで、General Settingsタブを開く。
3. グループボックス**Game-defined Auxiliary Sends**で以下を行う。
  - **Override parent**をクリックできれば、クリックする。
  - **Use game-defined auxiliary sends**をクリックする。

オブジェクトを特定してからGame-Defined Auxiliary Sendを有効にすることで、Game-Defined Auxiliary Sendの影響を受けるのがどのオブジェクトかをコントロールできます。Game-Defined Auxiliary Sendで環境をコントロールするシナリオなどでは、環境によって影響を受けるオブジェクトと受けないオブジェクトを区別できます。

Game-Defined Auxiliary Sendが有効な場合は、ゲーム値に追加されるGame-defined volume（ゲーム定義のボリューム）をWwiseで直接変更してゲーム SENDのボリューム値を調整できます。

### Game-Defined Auxiliary Sendを減衰と組み合わせて使う

Game-Defined Auxiliary Sendを、オブジェクトのポジショニングプロパティで設定する減衰と組み合わせて使えます。減衰を設定することで、ゲームが設定する以下のプロパティをリスナーとゲームオブジェクトの間の距離を基にコントロールできます。

- **Game-defined send volume** ウェットシグナルが距離によって減衰する量をコントロールする。
- **Output bus volume** ドライシグナルが距離によって減衰する量をコントロールする。

## ラウドネスノーマライゼーション、またはメイクアップゲインを使い、ボリュームを調整する

個別のオーディオソースのボリュームを調整するには、メイクアップゲインや自動ソースノーマライゼーションを利用することができます。これらのボリューム

コントロールは、Wwiseの他のボリュームコントロール機能と異なり、ボリュームに関連する全てのロジカル操作の観点から、トランスペアレントです。例えば、ボイスをスレッシュルド以下（バーチャルボイス）とするかどうかの判断に影響せず、HDR減衰の評価にも考慮されません。また、Voice Monitorにも表示されません。

ソースごとのゲインを合計した値が、WwiseのProfilerの、Voicesタブの、Normalization / Make-Up Gain 列に表示されます。

### ソースのノーマライゼーション

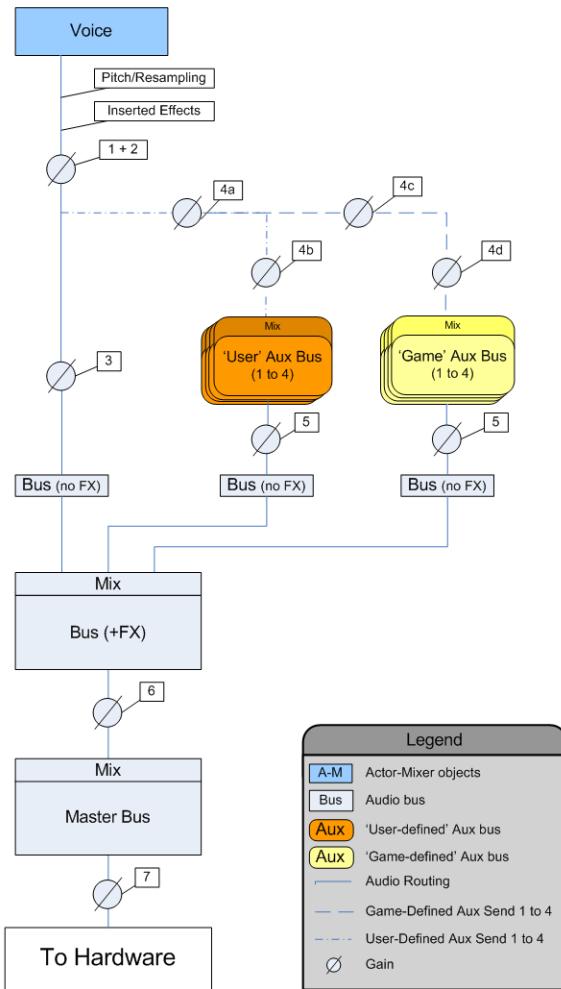
使用するオリジナルファイルの解析フェーズで収集したラウドネスデータを、Wwiseで利用して、予想されるラウドネスに基づきランタイムにアセットを自動的にノーマライズすることができます。ラウドネス値の算出は、ITU-R BS. 1770の勧告である、K-weightingフィルタ、-70dBの絶対ゲーティング、-10dBの相対ゲーティング、400msのウィンドウ、オーバーラップ75%を適用するラウドネス測定に、ほぼ準拠しています。

ソースのノーマライゼーション処理は非破壊的であり、サウンド解析で得たラウドネス値はWwiseで別途、保存され、ランタイムで適切なノーマライゼーションゲインが適用されます。このゲインは、ラウドネス値を元に算出され、「ゲイン = ラウドネス - 23 [dB]」となります。例えば、サウンド解析によるラウドネス値が-37dBであれば、ソースノーマライゼーションを有効にすると、ランタイムにWwiseによって「+14dB (+37 - 23)」のノーマライゼーションゲインが適用されます。また、サウンド解析によるラウドネス値が-12dBであれば、ノーマライゼーションゲインは「-11dB (+12 - 23)」です。つまり、小さいサウンドは増幅され、大きいサウンドは減衰されます。なお、ラウドネス値を元に算出したノーマライゼーションゲインに満足できない場合は、メイクアップゲインを使って微調整できます。

## ボイスパイプラインを理解する

ボイスに関する、処理手順、ルーティング方法、そして様々なボリュームやエフェクトを適用する場所などを、下図に示します。

## アウトプットの管理



番号	内容
1	全チャンネルに、ボリュームを適用 <ul style="list-style-type: none"> <li>• Voice Volume* (バスやオートダッキングのVoice Volumeパラメータは、このレベルで適用)</li> <li>• ノーマライゼーション、マイクアップゲイン</li> <li>• HDR減衰</li> <li>• Occlusion                             <ul style="list-style-type: none"> <li>• プロジェクト設定「Occlusion Volume curve」</li> <li>• API: <code>SetObjectObstructionAndOcclusion()</code></li> <li>• API: <code>SetScalingFactor()</code></li> </ul> </li> </ul> LPFプロパティと、Occlusion LPFカーブ (Project Settings)
2	チャンネルごとに、独立したボリューム <ul style="list-style-type: none"> <li>• Positioning                             <ul style="list-style-type: none"> <li>• 2Dパンニング、3Dポジショニング</li> <li>• Center%</li> <li>• Distance Attenuation Curve (距離減衰カーブ)</li> <li>• API: <code>SetGameObjectOutputBusVolume()</code></li> <li>• API: <code>SetPosition()</code></li> <li>• API: <code>SetMultiplePositions()</code></li> <li>• API: <code>SetListenerSpatialization()</code></li> </ul> </li> </ul>

番号	内容
3	いわゆる「ドライパス」 <ul style="list-style-type: none"> <li>アウトプットバスボリューム (+ RTPC)</li> <li>Distance Attenuation Curve (距離減衰カーブ) 「Output Bus Volume」</li> <li>API: <a href="#">SetGameObjectOutputBusVolume()</a></li> <li>API: <a href="#">AkSpeakerVolumeMatrixCallbackInfo()</a></li> <li>Obstruction               <ul style="list-style-type: none"> <li>Project Setting 「Obstruction Volume curve」</li> <li>API: <a href="#">SetObjectObstructionAndOcclusion()</a></li> </ul> </li> <li>エフェクトなしのバスのインサート: ボリューム*</li> </ul> Obstruction LPFカーブ (Project Settings)
4	いわゆる「ウェットパス」 <ul style="list-style-type: none"> <li>4a - Distance Attenuation Curve (距離減衰カーブ) 「Aux Send Volumes」               <ul style="list-style-type: none"> <li>API: <a href="#">AkSpeakerVolumeMatrixCallbackInfo()</a></li> </ul> </li> <li>4b - 「ユーザー定義のAUXセンド」の、ボリューム (+ RTPC)</li> <li>4c - 「ゲーム定義のAUXセンド」の、ボリューム (+ RTPC)</li> <li>4d - API: <a href="#">SetGameObjectAuxSendValues()</a></li> </ul>
5	<ul style="list-style-type: none"> <li>ボリューム*</li> <li>Positioning               <ul style="list-style-type: none"> <li>2Dパンニング、3Dポジショニング</li> <li>Center %</li> </ul> </li> <li>Distance Attenuation Curve (距離減衰カーブ)</li> <li>API: <a href="#">AkSpeakerVolumeMatrixCallbackInfo()</a></li> <li>エフェクトなしの親バスのインサート: ボリューム*</li> </ul>
6	<ul style="list-style-type: none"> <li>ボリューム*</li> <li>Positioning               <ul style="list-style-type: none"> <li>2Dパンニング、3Dポジショニング</li> <li>Center %</li> </ul> </li> <li>Distance Attenuation Curve (距離減衰カーブ)</li> <li>エフェクトなしの親バスのインサート: ボリューム*</li> </ul>
7	• ボリューム*

\* ボリューム = 「ボリュームスライダ」 + 「RTPC」 + 「State」 + 「Set Volumeアクション」

\*\* Bus Volume = Slider + RTPC + State + Set Bus Volume action

## 組み込みオーディオデバイス

オーディオデバイスは、対応プラットフォームがそれぞれ提供する物理的アウトプットやバーチャルアウトプット全てを表しています。Wwiseはデフォルトでいくつかのオーディオデバイスに対応しています。一部は、全てのプラットフォームに対応していません。複数のプラットフォーム用にゲームをデザインする時に、プラットフォームの性能が異なる場合は、オーディオが正しいアウトプットに向けられるように[Link/Unlink](#)機能を使う必要があるかもしれません。方法は、あなたのマスター・ミキサー階層にあるマスターバスの1つで、プロパティページを開いてAudio Deviceプロパティを変更します。

デフォルトオーディオデバイスのプラットフォーム別の詳細を見るには、SDKドキュメンテーションの[Default Wwise Audio Devices](#)を確認してください。

サードパーティプラグインをインストールして、バーチャルアウトプットを追加することもできます。???

対応するアウトプットについては、下記一覧を参照してください。

### System

システムが提供するデフォルトのオーディオアウトプット。全てのオーディオが、全てのプラットフォームで、ここに送られると考えます。

### Communication

チャットや遠隔通信用に設計されたアウトプット。ヘッドセットや、バーチャルデバイスなどが考えられます。以下のプラットフォームが対応しています:

- Windows
- Xbox One
- PS4

### DVR Bypass

プラットフォームの中にはDVR機能があり、ゲーマーがそのゲームプレイを録画して、公開できるものがあります。そこで、ゲームオーディオの一部でありユーザーが置き換えることのできるゲーム音楽の著作権について、法的な問題が発生します。ゲームスタジオ側は音楽をゲームに使う権利を有しているものの、いかなる形式であれエンドユーザーが配信する権利はありません。このため、このプラットフォームでは通常ユーザーのBGMの録音を禁止しています。このバーチャルオーディオデバイスを使えば、一部を別のミックスにいれて、それをDVRに送らずに、あとでメインアウトプットにミキシングすることができます。以下のプラットフォームが対応しています:

- Xbox One
- PS4

### Game Controller Speaker

プラットフォームによって、ゲームコントローラにスピーカーがあります。1人のプレイヤーに対して、1つのアウトプットを設定できます。以下のプラットフォームが対応しています:

- PS4

### Game Controller Headphones

プラットフォームによって、ゲームコントローラにヘッドフォンを接続できます。1人のプレイヤーに対して、1つのアウトプットを設定できます。以下のプラットフォームが対応しています:

- Xbox One
- PS4

## Auxiliary

対応しているのはPlayStation 4だけです。詳細は、メーカーのドキュメンテーションや[SDK ドキュメンテーション](#)を参照してください。

## No Output

これは、どこにもアウトプットしないバーチャルデバイスです。テストの時に便利で、全てのプラットフォームが対応しています。

## 関連トピック

- [???](#)
- 9章アウトプットバス構造の構築
- 「Secondary Bus Hierarchy」
- セカンダリアウトプットをインテグレートする

## セカンダリアウトプットを理解する

セカンダリアウトプットとは、メインテレビやメインスピーカーアウトプット以外のあらゆるオーディオアウトプットを指します。状況によって、このアウトプット用に別のオーディオミックスを行う必要があります。ゲームで使うと思われる各アウトプット用に、それぞれ、マスター・セカンダリバスを必要な数だけ作成できます。最も一般的なセカンダリアウトプットは、ゲームコントローラのスピーカーやヘッドフォンです。これ以外の独立したアウトプットも考えられます（チャット、BGM、ヘッドフォンなど）。このセクションでは主にゲームコントローラのスピーカーについて説明しますが、ほかの全ての種類のアウトプットにも該当する内容です。

セカンダリアウトプットからアウトプットするには、サウンドをマスター・セカンダリバス階層に送る必要があります、その2種類の方式を以下に示します：

- 対象サウンドのアウトプットバスのプロパティを、セカンダリバス階層にあるバスに直接設定する方式。これは、ほかのサウンドルーティングと同じように使います。通常は1つのセカンダリアウトプットインスタンスだけに結び付いているサウンドなどに向いています。例えば、プレイヤーが発射するガンショット、テニスラケットの打音、PDAの音、ゲームプレイのフィードバックなどがあります。
- 対象サウンドをマスター・オーディオバス階層のバスにルーティングしてからセカンダリバス階層のAUXバスへ送る、ユーザーセンドまたはゲームセンドの設定を追加する方式。複数のアウトプットやテレビから同時に聞こえてくるサウンドなどに向いています。例えば、スパイカメラやアナウンスなどです。

バス階層は、ミキシングの構造を定義するだけです。そのミックスを具体的なアウトプットと関連付けるには、対応するマスター・セカンダリバスで、適切な[Audio Deviceシェアセット](#)を選択します。

ゲーム中に複数のインスタンスが存在するかもしれないプレイヤー関連のアウトプット（ゲームコントローラやヘッドフォンなど）の場合は、それに関連するミ

ミキシング階層が、プレイヤーごとにコピーされることを、知つておく必要があります。プロジェクトで定義するのは、特定のアウトプットタイプ用のミキシングレシピのようなものだけです。実際にどのコピー構造にルーティングされるかは、プログラマーの設定するリスナーとゲームオブジェクトの関連付けで決まります。これを以下の例を用いて説明します。

- ・「セカンダリアウトプットの例」
- ・「BGMのミキシングと、DVRの例」
- ・???

### セカンダリアウトプットの例

全ての例で、Secondary Bus階層として下記のバス構造を用いています。メイン Audio Bus階層は省略します。この構造はアウトプットごとに複製されるので、最終的なミックスは互いに独立しています。なお、インスタンス化されるのは必要なバスだけです。

図30.1 マスター/セカンダリバス構造

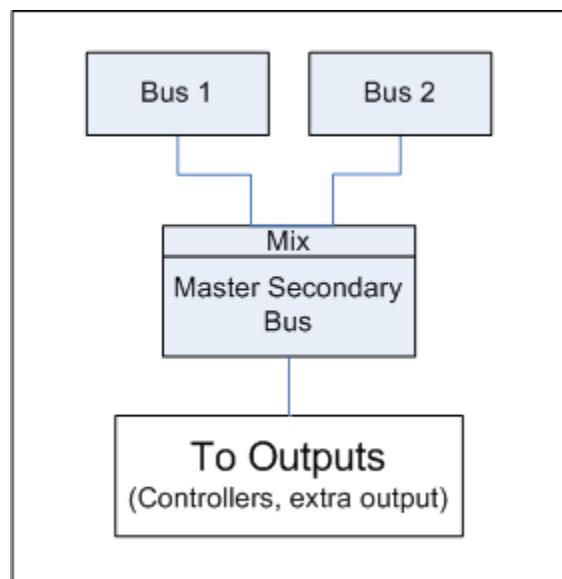
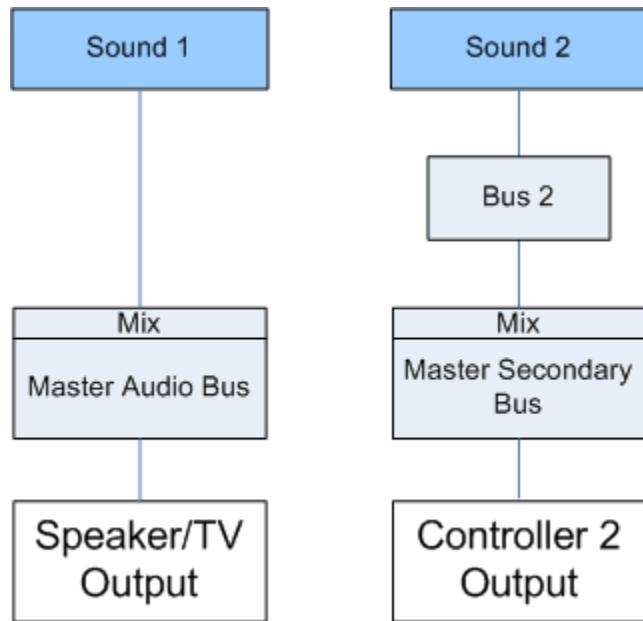


図30.2 サウンドがメインアウトプットに送られる単純な例



これは、1つのサウンドがメインアウトプットに送られる単純な例です。通常のゲームでは、複数のサウンドが複数のサブバスにルーティングされます。ここでは簡略化して、テレビアウトプットで再生される音が1つと仮定します。

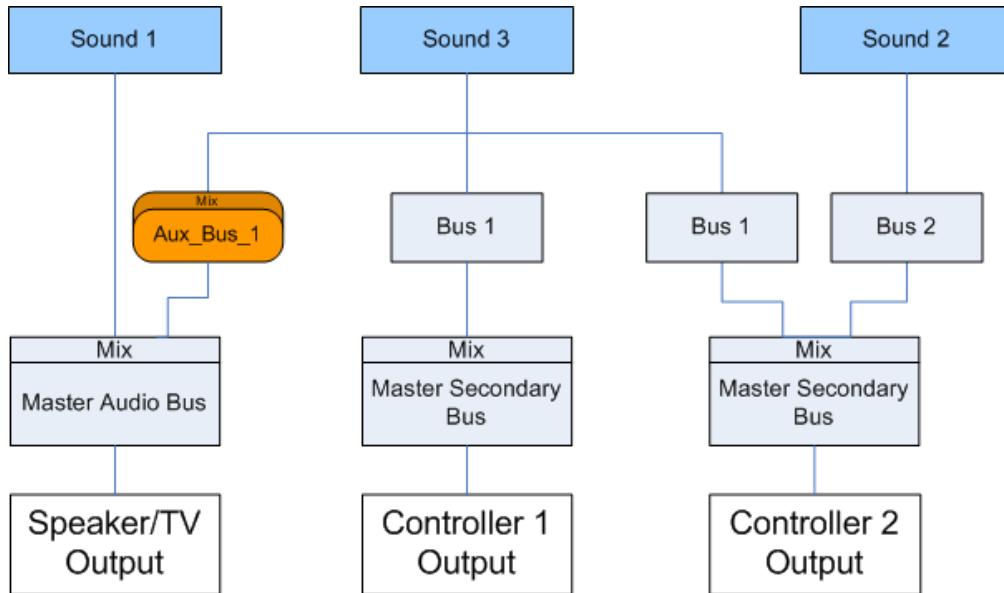
図30.3 サウンドが2人目のプレイヤーのコントローラアウトプットに送られる単純な例



次に、2人目のプレイヤーのゲームコントローラで別のサウンドを再生するとします。例えば、メニュー画面にこのプレイヤーだけが入り他のプレイヤーが入らない場合の、メニュー音を考えてみます。この場合、メニュー音が通常通り設定されたアウトプットバスにミキシングされ、TVミックスとは別となります。このサウンドがプレイヤー2のコントローラだけで再生されるのは、ゲーム内で、リスナーとゲームオブジェクトの関連付けが設定されたからです。この場合プログラ

マーは、コントローラ2のリスナーだけが聞くゲームオブジェクトで、このサウンドを再生させます。

図30.4 サウンドがプレイヤー1、プレイヤー2、テレビに送られる



上記の例で明らかなことが2点あり、つまりあるサウンドを複数のセカンダリアウトプットと同時にTVミックスへもルーティングできるということです。この例では、事前に録音された助けを求めるプレイヤー1の無線通信というサウンドが、同じチームのプレイヤー2に聞こえ、またテレビの視点がこのサウンドを発するプレイヤー2の近くにあるため、テレビからも聞こえます。なお、このサウンドのアウトプットバスはBus 1に設定されています。このバスと、マスター・セカンダリバスは、プレイヤー1とプレイヤー2の両方で複製されます。当然、プレイヤー2のミックスにはSound 1以外に、プレイヤー2にしか聞こえないSound 2も入らなければなりません。エフェクトがある場合は、両者のマスター・セカンダリバスのコピーに同じエフェクトが適用されますが、その対象となるオーディオシグナルが異なります。

またSound 3からTVミックスに向かうSendがあります。他のアウトプットに対して送る場合は、全種類のSendを使えます。この例では、距離関係や減衰によって無線通信がカメラにも聞こえることがあるので、メインTVミックスへ入れる必要があり、設定されたAuxiliary Busから送ります。

### BGMのミキシングと、DVRの例

著作権のある音楽は、一般的にPS4やXbox OneのDVRでの録音は禁止されています。どちらのコンソールでも、音楽だけを別に管理して録音するのを防止する、専用のバーチャルデバイスがあります。Wwiseでも、音楽だけを違う階層で別にミキシングすることになります。インテグレーションのデモプロジェクトで、これを実装してみました。NonRecordable\_Bus、RecordableMusic、NonRecordableMusicの設定を確認できます。コードの観点からは、DemoBGM.cppを見てください。

以下の通り、NonRecordableMusicは、Non Recordable Busへとルーティングされ、ここから信号がDVR\_Bypassアウトプットにアウトプットされます。

図30.5 音をNonRecordable\_Busにルーティング

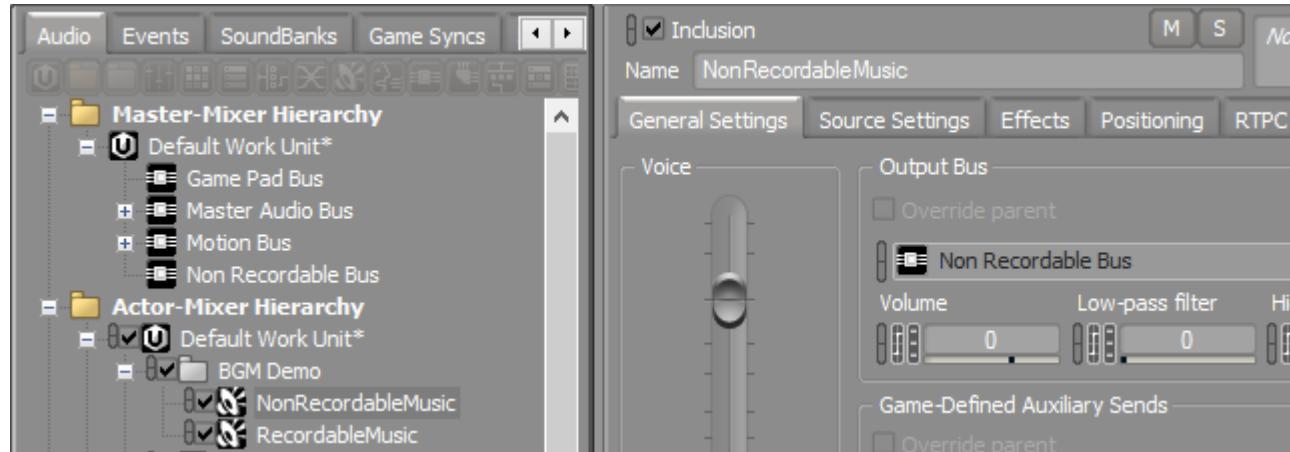
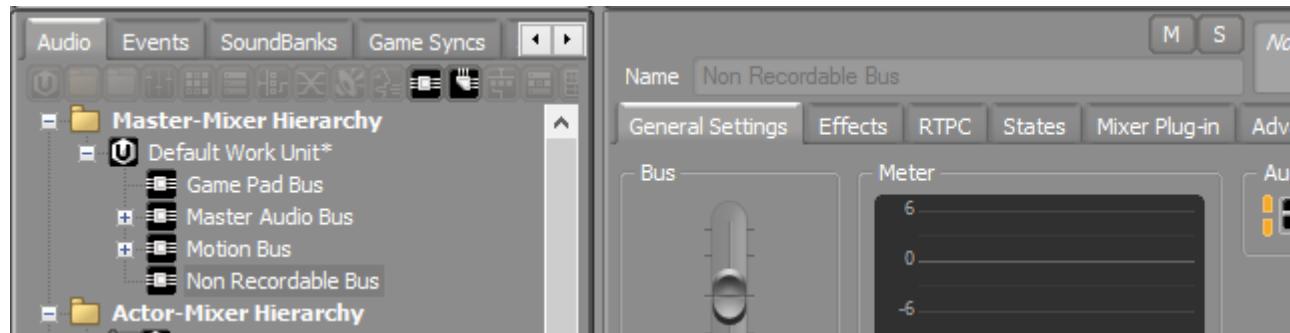


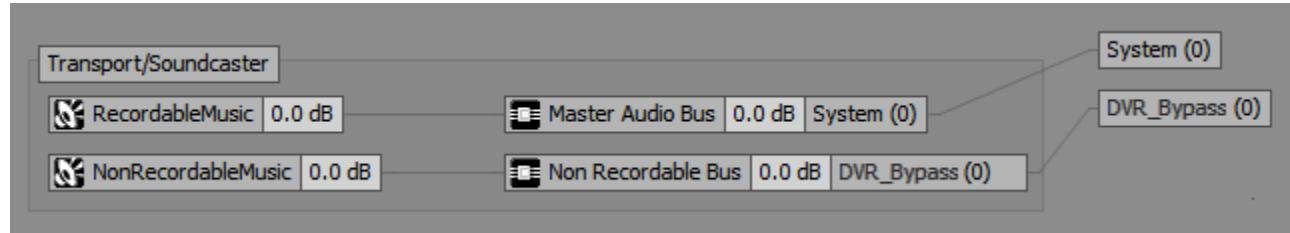
図30.6 Bus Audio Deviceに、DVR\_Bypassを設定



コード的には、これを実装するための作業はそれほどありません。アウトプットを初期化するために、ゲームがAudio Deviceシェアセットの名前を使い、`AK::SoundEngine::AddOutput()` をコールする必要があります。

Wwise Advanced Profilerを使うと、各サウンドが別々のバスや別々のアウトプットに行くのが分かります。これが、Xbox OneやPS4の通常の流れです。（中間のバスミュージックは、省略しています。）

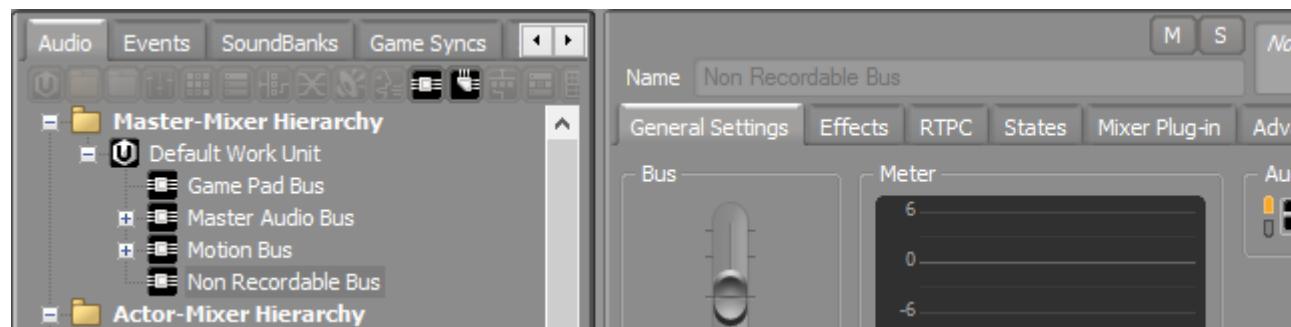
図30.7



### ハードウェアアウトプット不足の対処

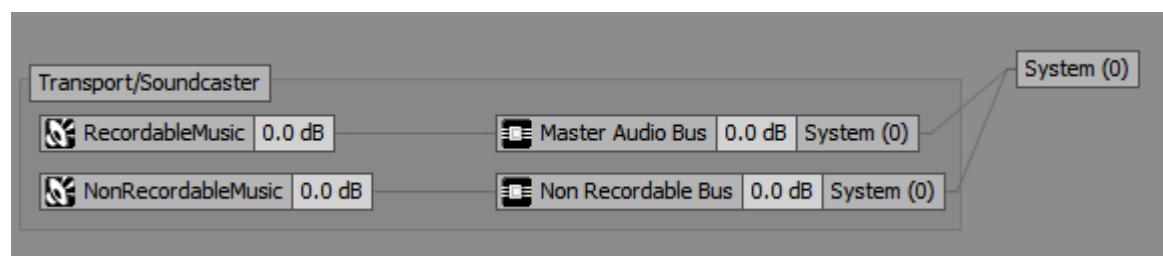
DVR機能は、ほかの対応プラットフォームには存在しないので、何か違う処理が必要です。その場合、当然ゲーム中に音楽を失いたくないので、正しくルーティングするようにします。対処するには、Audio Deviceプロパティをアンリンクするだけです。マスター・バスのプロパティエディタで、Audio Deviceプロパティをアンリンクすることもできます。以下の例では、Non Recordable Busのアウトプットを通常のシステムアウトプットにしました。ただしXbox OneやPS4ではDVR\_Bypassにアウトプットされるので、リンクアイコンは半分オレンジ色です。以上を、インテグレーションデモプロジェクトで設定しました。この場合、コードでアウトプットを初期化するためのAK::SoundEngine::AddOutputをコールする必要はなく、それはシステムのアウトプットが起動時に必ず初期化されるからです。

図30.8 Audio DeviceのアウトプットをSystemに設定



両方のサウンドを再生すると、2つのソースのシグナルが別々のバスを通り、最終的に同じシステムアウトプットでミキシングされるのが分かります。

図30.9 デバイスに2つのマスター・バスがミキシング



## HDRを理解する

HDR（ハイダイナミックレンジ）オーディオは、自然界で見られるような非常に幅広いダイナミックレンジに渡る様々なレベル値を使って、ミックスする技術です。また、HDRは、この幅広いレンジのレベルを、使用するサウンドシステムのデジタルアウトプットに適したレンジにダイナミックにマッピングする、ランタイムシステムでもあります。

人間の聴力閾値から、大気中で可能な最大サウンドまでの、実際の可聴ダイナミックレンジは、ゲームプレイ用スピーカーが提供するダイナミックレンジの、何倍

もの広さです。HDRシステムの役割は、現実世界の約190dBのダイナミックレンジ全体を、96dB（デジタル機器で利用できるダイナミックレンジ）まで、畳み込む、つまり圧縮することであり、実際には周囲のノイズレベルでさらに狭くなります。

HDR撮影は、1枚の写真で部分ごとにトーンマッピングを適用して、部分ごとにコントラストを向上させます。HDRオーディオも、同じ仕組みで、トーンマッピングの代わりにサウンドレベルのマッピングを、時間ごとに実行します。つまり、オーディオシーンを構成する様々なサウンドレベルに合わせて、HDRシステムが隨時、自動的にマッピングを調整します。

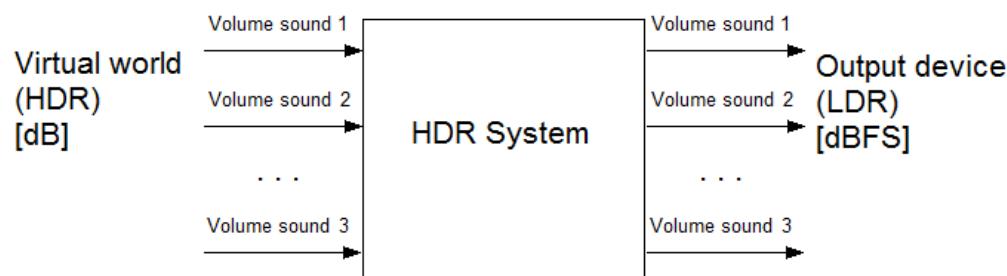
### HDR関連の用語

Term	Definition
Decibel (dB)	サウンドのレベルを、他のサウンドレベルまたは任意の基準値と比較して、対数で表した測定単位。振幅の場合、1dBは $20 \cdot \log_{10}(A/AR)$ である。 $+20\text{dB}$ の差は、サウンドの振幅が基準値の10倍であることを示す。
Decibel full scale (dBFS)	シグナルの振幅を、デバイスでクリップが発生する直前の最大振幅値と比較して、対数で表した測定単位。つまり、0dBFSは、デジタルオーディオアウトプットで生成できる最大のサウンドである。16ビットのデジタルオーディオアウトプット機器は、0dBFSから、最小で-96dBFSまでのレンジ（範囲）がある。このため、WwiseのMaster Audio Busから出すオーディオシグナルのレベルは、この範囲内とする。

### HDRの例

HDRオーディオでは、16ビットのアウトプット機器の標準ダイナミックレンジである96dBよりもはるかに広いダイナミックレンジのボリューム値を、現実世界と同じようにゲームのバーチャルワールドのサウンドにアサインできます。下図に示すように、このボリューム値をdBFS単位に変換するのが、HDRシステムの役目です。

図30.10 HDRシステムのインプット、アウトプット



インプットされるのは、バーチャルワールドのサウンドレベルで、任意の基準値に対する相対値として、デシベル (dB) で表されます。インプット値の大小は任意で選択できるため、ハイ（広い）ダイナミックレンジとなります。アウトプットは、それぞれのサウンドのレベルをdBFSで表したもので、アウトプット値のレンジは、アウトプット機器によって決まり、一般的にインプットよりもダイナミックレンジが狭くなります。

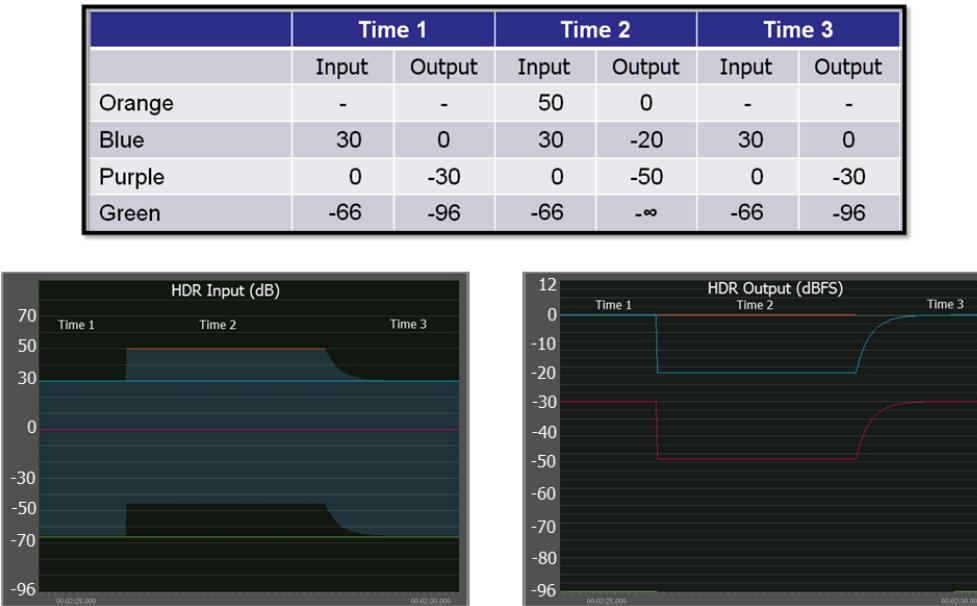
HDRシステムの処理を簡単に説明すると、まず時間ごとに、バーチャルワールドのサウンドのうち最大ボリュームがアサインされているサウンドを、システムが選択し、それを自動的にアウトプット値0dBFSとしてマッピングしたあと、他の全てのサウンドを、相互関係を維持したままマッピングします。

それでは、具体例で説明します。図30.11「HDRウィンドウ」の左のグラフの通り、「Time 1」という時間において、バーチャルワールドでサウンド「Blue」が+30dBで再生されるといします。基準値0dBは任意値です。「Blue」が「Time 1」における最大サウンドであるため、HDRシステムのアウトプットでは0dBFSで再生されます。一方、別のサウンド「Purple」はバーチャルワールドで0dBで再生され、つまりこれはサウンド「Blue」よりも30dB小さいことになります。したがってHDRシステムのアウトプットで-30dBFSとなります。3つ目のサウンド「Green」はバーチャルワールドで-66dBで再生されるので、HDRシステムのアウトプットでは-96dBFSとなります。なお、システムのアウトプットのダイナミックレンジが96dBに制限されているので、「Green」のレベルは、全ての可聴サウンドの下限と一致します。「Time 1」では、「Green」より小さいサウンドは一切聞こえません。

この例のバーチャルワールド（インプット側）のレンジ [-66, +30]dB を、HDR ウィンドウ（HDR Window）と呼びます。これは図30.11「HDRウィンドウ」の左側の青い部分で示されます。HDRウィンドウの縦幅は一定で、アウトプットのダイナミックレンジによって決まります。16ビット機器の場合は最大96dBに等しくなりますが、実際にはさらに狭いのが一般的です。「Time 1」時点のバーチャルワールドの最大サウンドは「+30dB」なので、-66dBより小さいサウンドは、HDRウィンドウより下方にあり聞こえません。

次に、このあとの「Time 2」という時間に、別のサウンド「Orange」がバーチャルワールドにおいて+50dBで再生し始めたとします。ボリュームの大きい新サウンドに対応するために、HDRシステムはウィンドウを20dBだけ上に移動させ、インプット側の範囲を[-44, +50]dBに変えます。この時点で、全サウンドが新しい数値にマッピングされます。+50dBのサウンドを0dBFSで再生し、+30dBのサウンドを-20dBFSで再生し、-66dBのサウンドはウィンドウより下となるため、全く聞こえなくなります。次に、サウンド「Orange」の再生が「Time 3」で終わると、ゆっくりとウィンドウが元の位置に戻り、他のサウンドは元のボリュームを回復します。

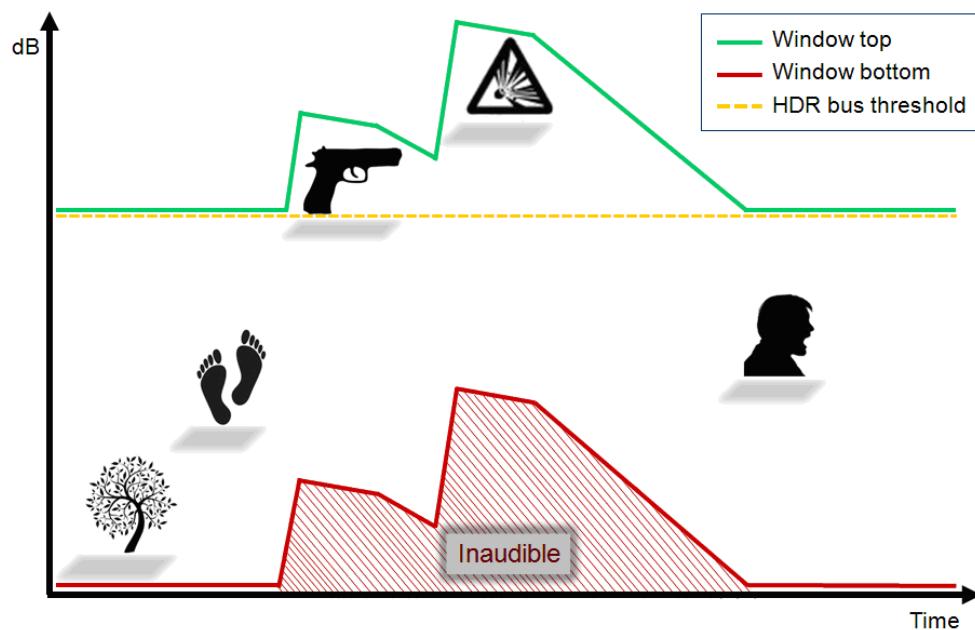
図30.11 HDRウィンドウ



左のグラフはシステムのインプットを示すもので、任意の基準値に対するサウンドレベルをデシベル (dB) で表示しています。HDRウィンドウは青色で塗られた範囲です。「Time 1」では、ウィンドウの最上部が、再生中の最大サウンド「Blue」と一致しています。「Time 2」で別のサウンド「Orange」が+50dBで再生し始めると、すぐにウィンドウが対応して上に移動します。この時、-66dBのサウンド「Green」は明らかにウィンドウより下なので、聞こえなくなります。この時のシステムアウトプットのサウンドレベルを、右のグラフにdBFSで表示しています。次に、サウンド「Orange」の影響でHDRウィンドウが+20dBだけ上に移動すると、他のサウンドのボリュームが-20dBだけ小さくなります。この時、サウンド「Green」はアウトプットから完全に除外されます。ここで、サウンド「Orange」が、「Time 1」期間中のサウンド「Blue」のアウトプットと同じレベルで再生されていることに注目してください。サウンド「Orange」の再生が終わると、ウィンドウが元の位置にゆっくりと下がり、これに応じて他のサウンドのボリュームは上がります。

HDRシステムは、ダイナミックレンジリミッター・コンプレッサーのように作用します。ボリュームの大きいサウンドを再生する時は、小さいサウンドを聞こえなくして、小さいサウンドだけを再生する時は、再び聞こえるようするというように、ミックスに作用します。HDRの世界におけるサウンド同士の相対レベルは維持されるので、より広いダイナミックレンジの印象を与えながら、実際にはアウトプット機器の狭いダイナミックレンジ内にサウンドが圧縮されています。さらに、大きいサウンドを再生する時にシステムがボリュームを自動的にダッキングするので、よりクリーンでフォーカスの絞られたミックスとなります。この考え方を、下図に示します。

図30.12 システムのインプット側からみた、HDRの概要



ウィンドウは、より大きいサウンドが再生される時だけ、上に移動します。インプット側でウィンドウが上に移動すると、アウトプット側のサウンドボリュームが下がります。それまで聞こえていた木の葉の音などが、ガンショットを再生すると全く聞こえなくなります。システムアウトプットのサウンドの実際のボリュームは、各サウンドの位置から、その時点でのウィンドウの上部までの距離によって変わります。ガンショットも爆発も、個別に再生した時は同じレベルでシステムからアウトプットされますが、爆発はガンショットよりも音が大きいとされるので、爆発がガンショットのボリュームをダッキングする結果となり、プレイヤーは実際に爆発音の方が大きい印象を受けます。

## HDRの使用

ユーザーが、HDRレベルと、フルスケール（機器のスケール）の間の変換機の役割を担うための、バスを1つ、Wwiseで選択しておく必要があります。このHDRバスが取り込むインプットレベルが、Wwiseでユーザーが設定するロジカルレベルです。このため、HDRバスにルーティングするサウンドのボリュームを、0dBよりもはるか上に設定することができます。重要なのはHDRウィンドウにおけるサウンドの位置だけで、その位置は再生中のサウンドに合わせてHDRバスがダイナミックに移動させます。つまり、HDRバスは、オーディオリミッター・コンプレッサーのコントロール機能と似たものを提供することで、ロジカルリミッター・コンプレッサーの役割を果たします。HDRウィンドウが時間の経過とともにどう移動するのかをコントロールするための、非常に短いアタックや、ユーザーが設定できるリリースがあります。また、スレッシュホールド（Threshold）もあり、HDRウィンドウが移動できる範囲の最低ラインとみなせます。



### dBSPLの活用

これまでの文献で、多くのHDRオーディオシステムのインプット側のボリュームレベルが、音圧 (dB SPL) で表現されています。dBSPLは基準値 (0dB SPL) が人間の聞き取れるスレッショルドに相当する、デシベルを使った測定方法です。このSPLという概念はWwiseに存在せず、その理由は、導入すると不要に複雑になり、インターフェース上で邪魔となる上、システムが使いやすくなるわけでもないためです。その代わり、インプット側の基準値を任意値としてあるので、ユーザーが定義できます。dBSPLを単位として使いたい場合は、サウンド構造のボリュームを、直接、正のdBSPL値に設定します。ただし、Wwiseのボリューム用スライダの上限が+12dBなので、基準値を別途、選択して、希望するSPL値から引き算をして対応する相対dBレベルを求める方法が現実的かもしれません。例えば、100dB SPLを基準値、つまり0dBとして定義するとします。すると80dB SPLのサウンドのボリュームスライダは-20dBに、130dB SPLのサウンドのボリュームスライダは+30dBとなります。また、HDRバスのスレッショルドも合わせて設定する必要があります。

- 「プロジェクトのHDRを有効にする」
- 「HDRバスに、サウンドをルーティングする」
- 「ウィンドウをモニターする」
- 「HDRのダイナミック設定」

### プロジェクトのHDRを有効にする

HDRサブシステムとして利用するバスを、単数または複数、指定するだけで、プロジェクトでHDRを有効にできます。

#### バスのHDRを有効にするには:

- 1つのバスオブジェクトを、確認する。
- Property Editorの HDRタブをクリックする。
- Enable HDRをクリックする。

HDRバスやその子孫にルーティングされる全てのサウンド構造のボリュームが、HDRバスの影響を受けます。HDRバスにルーティングされないサウンドは影響を受けませんが、サウンドデザイン内にある、HDRバスにルーティングされた他のサウンドと共に存させることができます。また、1つのプロジェクト内でも、互いに完全に独立していれば、複数のHDRバスを設定することも可能ですが、HDRバスの上位や下位に、他のHDRバスを設定することはできません。

前述の通り、HDRウィンドウの上端にあるサウンドが、0dBFSでアウトプットされます。同じプロジェクトの、他の非HDR部分とミックスする前に、HDRバスのバスボリューム用スライダを調整して、HDRシステムのアウトプットをスケールダウンしてください。

#### HDRバスに、サウンドをルーティングする

Actor-Mixer HierarchyやInteractive Music Hierarchyにあるサウンドオブジェクトに対して、HDRシステムを適用するには、オブジェクトをHDRバスにルーティングする必要があります。

#### サウンドを、HDRバスにルーティングするには:

1. サウンドオブジェクトの内容を表示させる。

2. Property Editorで、General Settingsタブを開く。
3. **Output bus**として、HDR busを選択する。

### ウィンドウをモニターする

Voice Monitorビューに、各種ボイスのボリュームと、それぞれのエンベロープ（あれば）が表示されます。

### HDRシステムを理解して、デバッグするには:

1. Voice Monitorビューを開く。
2. ビュー内に、HDRバスをドラッグして入れる。
3. Modeを、Bus inputまたはBus outputに設定する。Bus inputモードは、HDR圧縮が行われる前の、HDRバスが受け取るボイスレベルを、dB値として表示し、Bus outputモードは、HDR圧縮とバスアウトプットゲインのあとでの、HDRバスからのアウトプットの状態を表示する。本ドキュメントの図の多くは、Voice Monitorビューの「両サイド」のスクリーンショットである。

### HDRのダイナミック設定

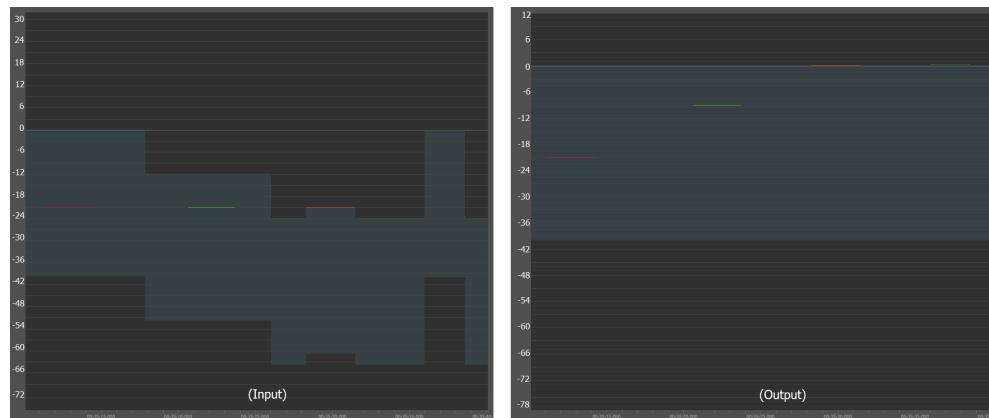
各HDRバスに、それぞれのHDRウィンドウがあり、ウィンドウの位置はその時点で最も大きいサウンドのレベルによって決まり、またウィンドウの縦幅はプロジェクトのボリュームスレッショルドによって、決まります。HDRウィンドウの動作は、HDRバスのHDRタブで編集できます。HDRバスがオーディオリミッター・コンプレッサーのような働きをするので、コントロール機能も似ています。

- 「[HDRスレッショルドの使い方](#)」
- 「[HDRのアタック・リリース](#)」
- 「[HDRレシオの使い方](#)」

### HDRスレッショルドの使い方

典型的なオーディオリミッター・コンプレッサーのスレッショルドと同様に、HDRスレッショルドによって、HDRウィンドウが移動する範囲の最小インプットレベル（[図30.12「システムのインプット側からみた、HDRの概要](#)』の黄色の線を参照）が決まり、この最低レベルより上では、コンプレッサーが作動し始めます。再生中のサウンドが小さいサウンドばかりの時は、アウトプットにおいて、これらのサウンドレベルは、HDRスレッショルドの影響を直接受け、つまり、ウィンドウの上端から遠ければ遠いほど、アウトプットレベルが小さくなります。このスレッショルドより上でサウンドが再生されると、小さいサウンドが自動的にダッキングされます。なお、圧縮率（圧縮率については後述）が無限の場合、[図30.13「HDRスレッショルドの効果](#)』に示す通り、HDRスレッショルドより上のインプットレベルにある2つのサウンドは、それぞれのインプットレベルに関わらず、それが再生中の唯一のサウンドであれば、同じアウトプットレベルで再生されることを、充分に理解してください。

図30.13 HDRスレッショルドの効果

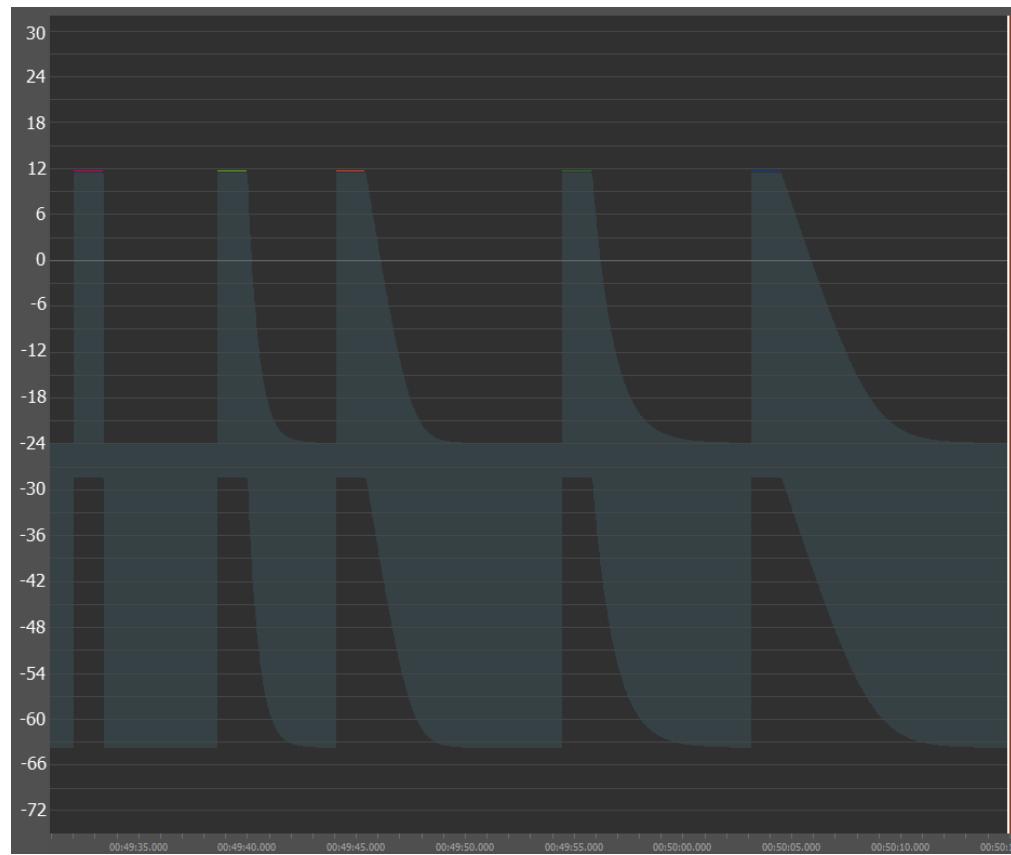


上図のグラフは、左がインプット、右がアウトプットを示しています。最初は-21dB、スレッショルドが0dBに設定されているので、-21d BFSとしてアウトプットされます。2回目は、スレッショルドが-12dBなので、サウンドは-9d BFSとしてアウトプットされます。3回目は、サウンドがスレッショルドの-24dBよりも上にあるため、ウィンドウが移動し、0d BFSとしてアウトプットされます。次にサウンドが0dBのインプットレベルで再生されると、スレッショルドよりもさらに高くなりますが、それでも0d BFSとしてアウトプットされます。

### HDRのアタック・リリース

標準的なオーディオコンプレッサーには、圧縮の仕方を時間で決めるアタックとリリースが実装されています。同じように、ユーザーはHDRバスのリリースタイムとリリース形状をコントロールして、HDRウィンドウが低い数値へとリリースされる方法を設定できます。様々なリリースタイムやリリースモードを、[図30.14 「HDR release: 0, 0.5 exponential, 0.5 linear, 1 exponential, 1 linear」](#)に示します。パンピング現象などを抑えるために、調整してください。なお、アタックタイムのコントロール機能はなく、これは、HDRシステムのアタックが、オーディオリミッターと同様に瞬時でなければならないためです。

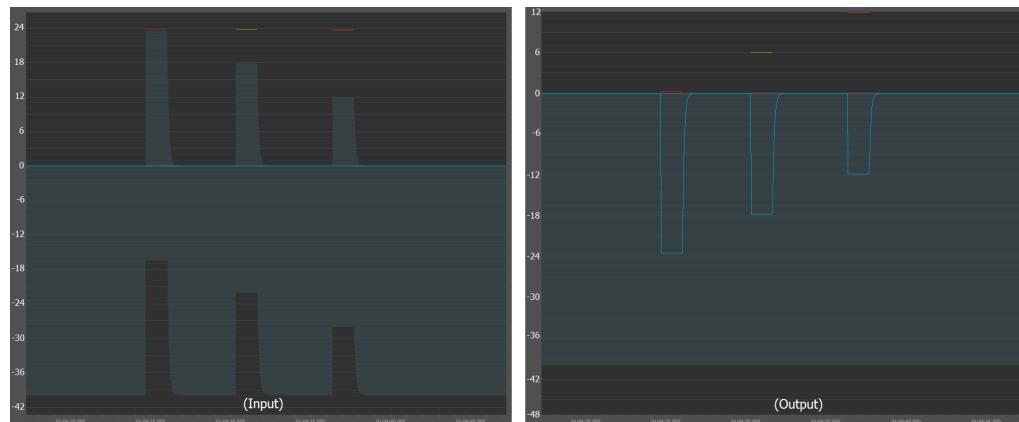
図30.14 HDR release: 0, 0.5 exponential, 0.5 linear, 1 exponential, 1 linear



### HDRレシオの使い方

ここまで、HDRシステムをオーディオリミッターのように表現してきました。HDRウィンドウの上端が、最大サウンドのレベルを厳密に追うので、[図30.13「HDRスレッショルドの効果」](#)に示す通り、HDRスレッショルドより上の全てのサウンドは個別に再生すると、同じアウトプットレベルとなります。前述の通り、HDRは、サウンド同士の相対的なダイナミックレンジを増加させる印象を与えるものの、絶対的なダイナミックレンジは狭くなります。この絶対的なダイナミックレンジをある程度、取り返すには、HDRリミッターをHDRコンプレッサーに転換させます。これには、有限の、より小さい圧縮率（レシオ）を設定します（デフォルトの圧縮率は無限）。圧縮率によって、HDRウィンドウが最大サウンドをどれだけ厳密に追うのかが決まり、サウンドのレベルとHDRスレッショルドの差を用いて算出されます。[図30.15「HDRレシオの影響」](#)は、圧縮率の影響を示しています。なお、圧縮率が無限でない場合に、大きいサウンドがHDRウィンドウを超過してしまい、HDRシステムから0dBFSより大きくアウトプットされる可能性があります。HDRバスや、シグナルバスの下流にある他のバスのボリュームを下げて、ヘッドルームを確保してください。

図30.15 HDRレシオの影響



このサウンドのインプットレベル（左のグラフ）は+24dBで、100:1、4:1、2:1のレシオで、HDRバスで再生されます。対応するアウトプットレベルとして、0dBFS、+6dBFS、+12dBFSが、右のグラフに表示されます。別のサウンドは、インプットレベルが0dBで、-24dB、-18dB、-12dBのダッキングが行われます。2:1の時、スレッシュルドより上のエネルギーの半分だけがHDRウィンドウを上に移動するのに使われ、その結果、他のサウンドが-12dBだけダッキングされます。なお、レベル差分の24dBは、常に維持されます。また、小さいレシオを用いると、ウィンドウ上端より上にサウンドが出る可能性があり、0dBFSを超えてしまうので、バスやバスの親のボリュームを下げて、充分なヘッドルームを確保するべきです。

### 振幅エンベロープの使い方

HDRシステムは、Wwiseで設定したロジカルボリューム値を使うため、インプットサウンドの実際の振幅が把握できていない状態にあり、インプットサウンドのことを、持続する限りボリュームが一定である「ブラックボックス」とみなします。多くの場合、サウンドの振幅は変化します。例えば、アタックとリリースの部分から成るインパクトサウンドがあるとします。HDRウィンドウの位置を決めるような大きいサウンドであれば、このサウンドが終わるまでウィンドウの位置が常に一定に保たれてしまい、不自然です。ウィンドウへの影響を、図30.16「エンベロープなしの、ディケイするインパクトサウンドの、HDRウィンドウ」に示します。

Wwiseでエンベロープトラッキングを有効にすれば、HDRシステムが「ブラックボックス」の中をのぞくことができます。対象サウンドのプロパティの、HDRタブで、チェックボックスEnable Envelopeを選択すると、オーディオファイルの振幅エンベロープの解析結果がサウンドのメタデータに添付されます。HDRシステムがランタイムに、このデータを利用してウィンドウを適宜、移動させる様子を、図30.17「エンベロープ付きの、ディケイするインパクトサウンドが、小さく安定したバックグラウンドと重ねて再生される時の、HDRウィンドウ」に示します。

下図では、インプットを左側、アウトプットを右側、そして対応するアウトプット波形をその下に示します。

図30.16 エンベロープなしの、ディケイするインパクトサウンドの、HDRウィンドウ

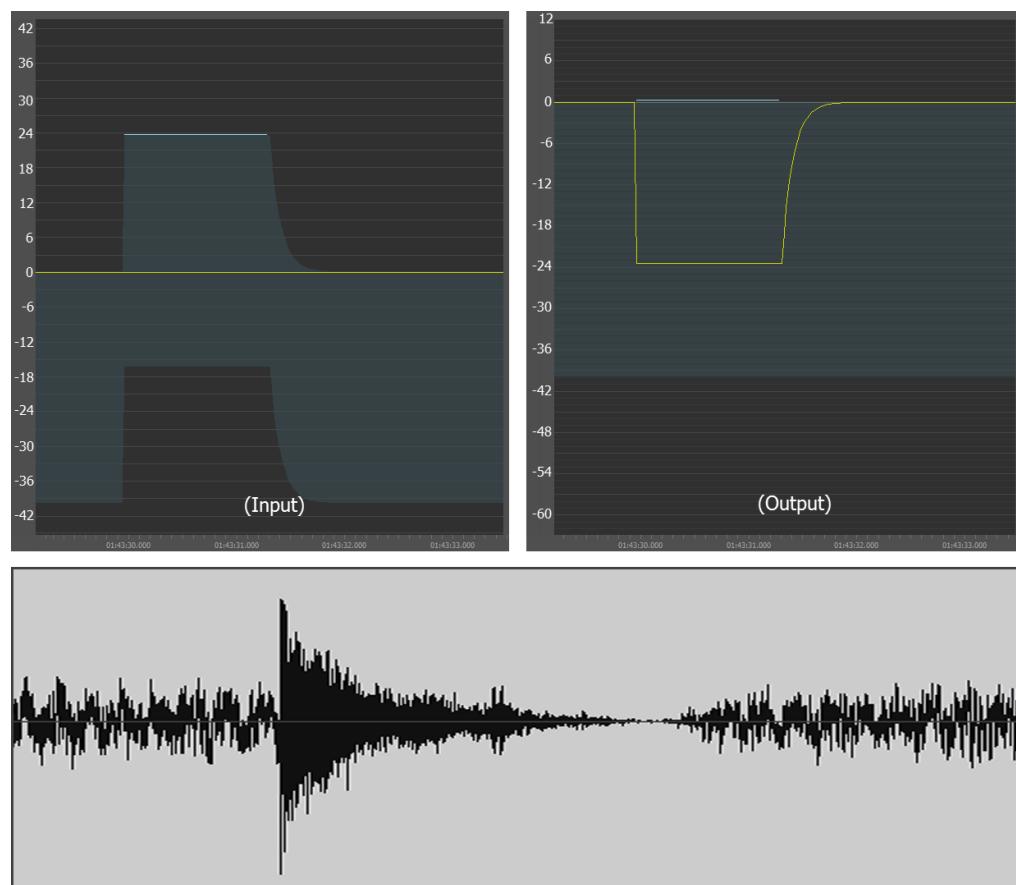
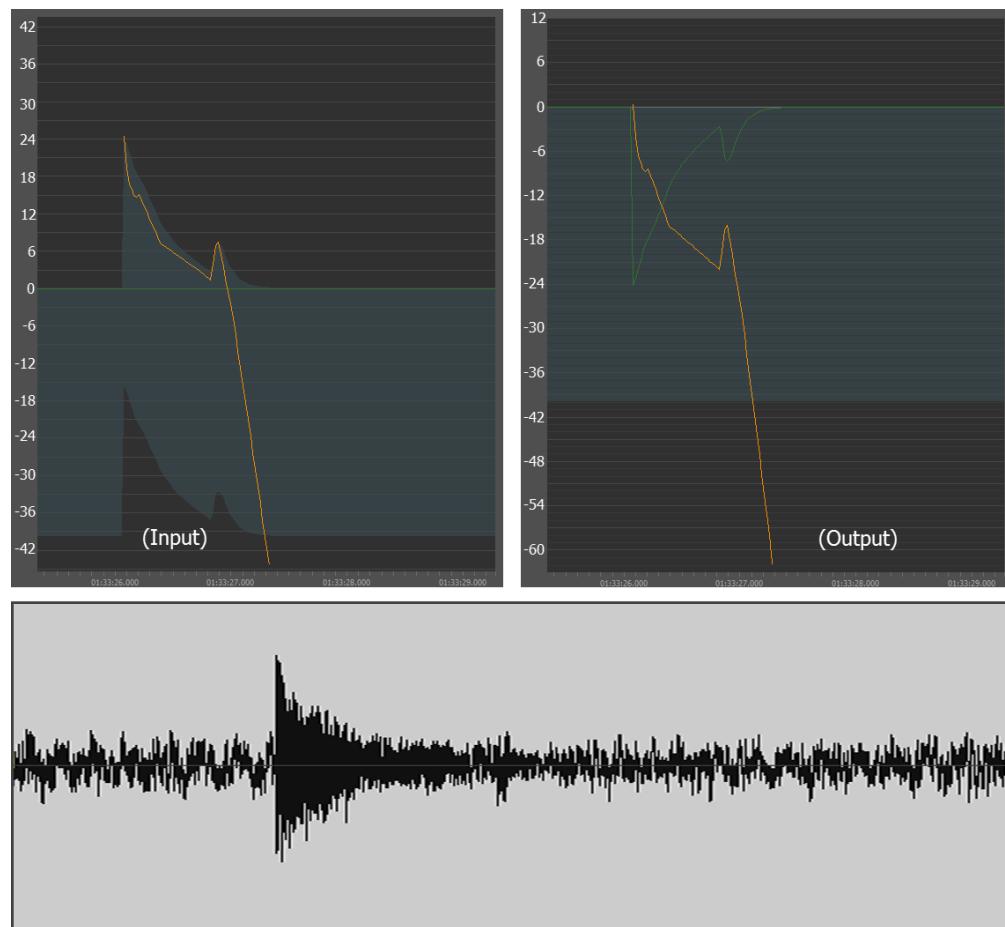


図30.17 エンベロープ付きの、ディケイするインパクトサウンドが、小さく安定したバックグラウンドと重ねて再生される時の、HDRウィンドウ



上の図のうち、最初の図では、インパクトサウンドのあとに水平となる部分がありますが、HDRシステムがサウンドを一定であると理解するためです。

#### エンベロープ感度と、マニュアル編集

エンベロープを有効にしたあと、ソースエディタを開いて、その効果をプレビューできます（下図）。精度やグラフの頂点の数に不満があれば、グループボックステンvelope Trackingにある、Sensitivity（感度）スライダの数値を変えます（[図30.19「Property EditorのHDRタブ」](#)参照）。いくつかのサウンドをグループにまとめて作業するには、上位にあるサウンド構造、例えばアクターミキサーなどに対して、エンベロープを有効にして感度を設定します。なお、感度設定の効果は、元のオーディオデータに依存するため、似たサウンドの複数のバリエーションがある場合は、効果がそれぞれ異なるかもしれません。また、各サウンドのエンベロープを個別にマニュアル編集することも可能で、求める結果に近いグラフを得るために、感度コントロールの調整や、点の移動、削除、追加をします。

図30.18 Source Editor画面のRMSモードの、エンベロープ表示

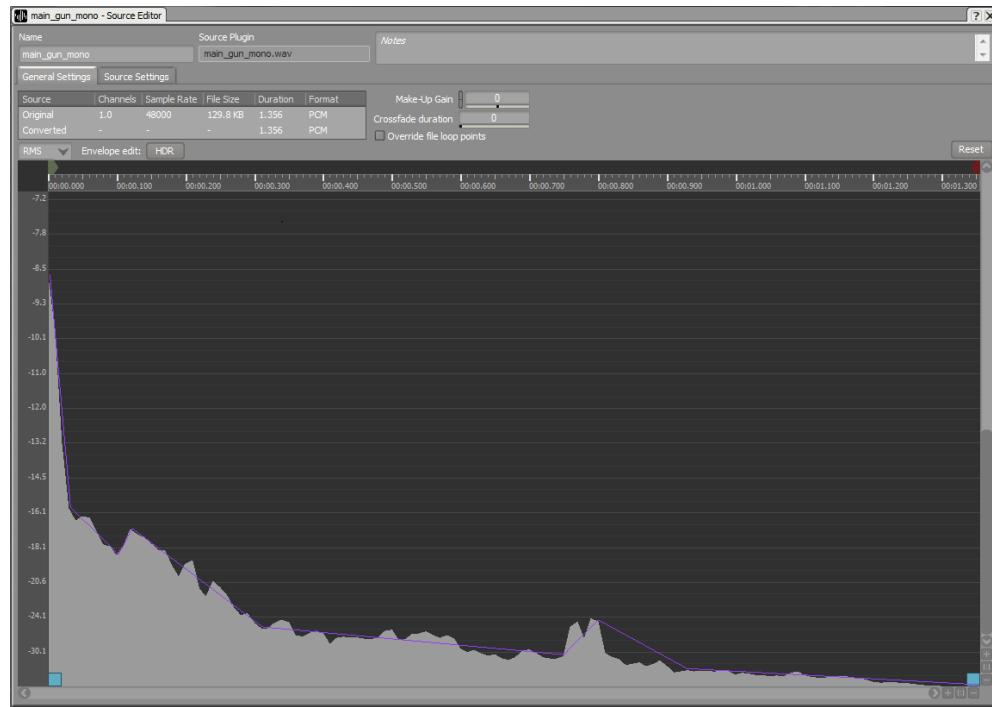
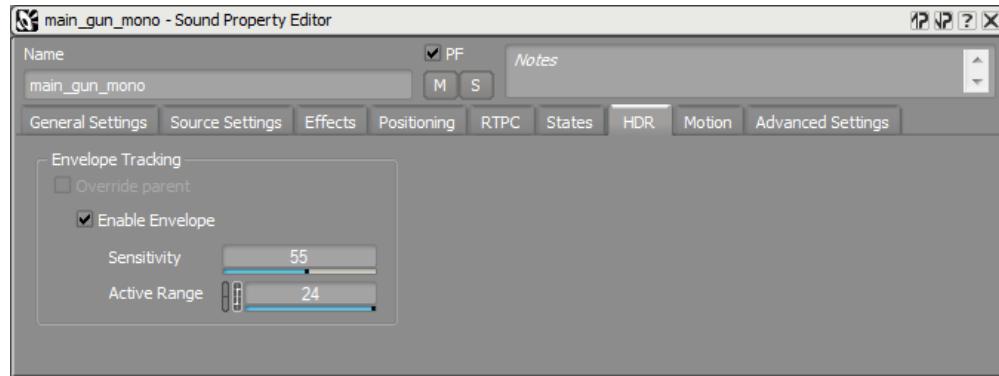


図30.19 Property EditorのHDRタブ



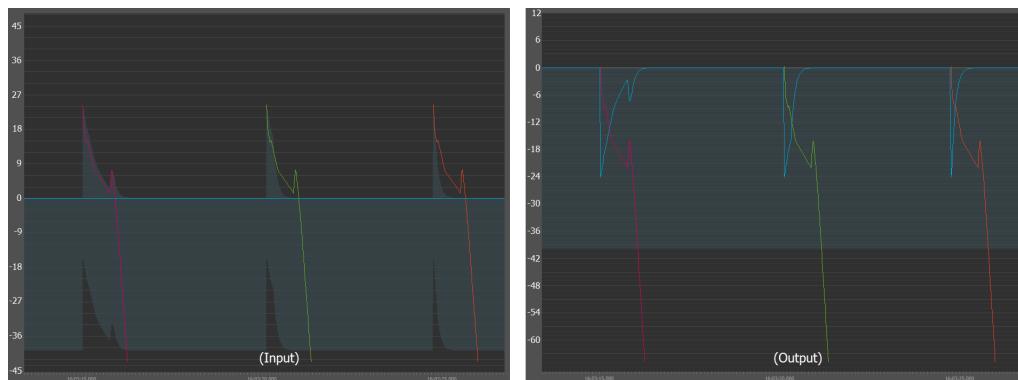
#### アクティブレンジとしての、ROI

一般的に、0dBから-96dBまで及ぶようなインパクトサウンドなど、レベル差の大きいサウンドがHDRウィンドウを左右するのは、サウンドが持続する間中ではなく、一定の間だけとする方が良いことが分かっています。その根拠は、HDRがラウドネスに基づく優先システムとして作用することは言え、大きいサウンドの音の小音量の部分まで同等の重要性を持たせるべきではない、という矛盾にあります。例えば、グレネードサウンドのリリース部分は、アタックと比較して重要性が非常に低く、たとえショットガンよりも絶対値が実際に大きいとしても、ショットガンのインパクトを覆い隠すべきではありません。この場合、グレネードサウンドのROI (Region of Interest、関心領域) を、最初のインパクト部分に限定します。最初のインパクトによってHDRウィンドウが上に移動して、他を全てダッ

キングします。一方、リリースに入り、再生がピーク値よりも12dBほど下がった時点では、たとえグレネードのボリュームから12dBを差し引いたボリュームが、オーディオシーンの他の全てのサウンドよりもレベルが高いとしても、他のサウンドをダッキングし続けるのは望ましくないことが多いでしょう。HDRウィンドウの制御をROIに限定することは、HDR画像に例えるならば、様々な色調の領域を美しくブレンドさせる手法であると言えます。

WwiseでROIを指定するには、サウンドのピーク値から下の範囲をアクティブレンジとして、dB単位で設定します。サウンドのエンベロープがアクティブレンジよりも低くなると、サウンドのROIから出たとみなされ、HDRシステムにおいて、HDRウィンドウを左右する要因として扱われません。アクティブレンジの異なる3つの似たサウンドを、[図30.20 「アクティブレンジ」](#)に示します。

**図30.20 アクティブレンジ**



一定のバックグラウンドサウンドの上で、ディケイする同じサウンドが3回、再生され、そのアクティブレンジは、それぞれ96dB、12dB、6dBに設定されています。サウンドが、ピーク値からこのdB数だけ下がると、ウィンドウがサウンドを追うことを止め、リリースしてアイドル状態に戻ります。ウィンドウの移動は、バックグラウンドサウンドのレベルに明らかに影響します。一方、ウィンドウもアクティブレンジも、リリースサウンド自体には全く影響を与えません。

## HDRの追加情報

あるサウンドのエンベロープは、他のサウンドに影響するだけで、そのサウンド自身に影響を与えないことを充分に理解してください。この特徴こそ、HDRシステムとオーディオコンプレッサーの一番の違いです。最も大きいサウンドは随时、そのサウンドのピーク値によって調整されますが、自分自身のエンベロープの影響を受けないので、HDRシステムサウンドがトランスペアレントに聞こえます。その結果、エンベロープを使用すると、インプットされる複数のサウンド間のレベル差は厳密には守られません。つまり、前述の通り、オーディオシーンにおいて様々な「それぞれのラウドネス」を有する複数のサウンドをブレンドできるというメリットがあります。ただし、HDRウィンドウの移動が激しいと、オーディオシーンのリアリズムが多少失われたように感じられることがあります。HDR画像処理で細かいトーン領域を大量に使うと、同じ現象が発生します。インターネット上には、現実的でないHDR写真の例が沢山あります。HDRスレッショルドより上のサウンドレベルを設定する時は、慎重に進め、つまり適当に数値を入れ込むのではなく、

注意してレベル設定を行い、大きいサウンドのスペースをミックスの中で確保できるように、エンベロープを設計してください。

エンベロープはメモリを消費するので、使わないエンベロープは有効にしないでください。HDRスレッショルドより上にいくことのない小さいサウンドのエンベロープは無意味なので、そのようなエンベロープは無効にします。

アクティブレンジ周辺を上下するエンベロープは、望ましくない状況を引き起こすことがあるので、注意してください。下図の例では、長い爆発音がアクティブルレンジを出入りするため、HDRウィンドウが何回も急激に移動します。この場合、エンベロープの精度を下げたり、マニュアル編集したりしてください。さらに、現実と正確に一致しないようなエンベロープを巧みにデザインして、サウンドの初期インパクト中だけオーディオシーンのダッキングを行い、地響き部分では早めにリカバリさせても良いでしょう。

グラフの頂点が少ないシンプルなエンベロープの方が、オーディオシーンの不規則な振幅変動が少なく、一般的に有利です。

図30.21 エンベロープ編集 (a)

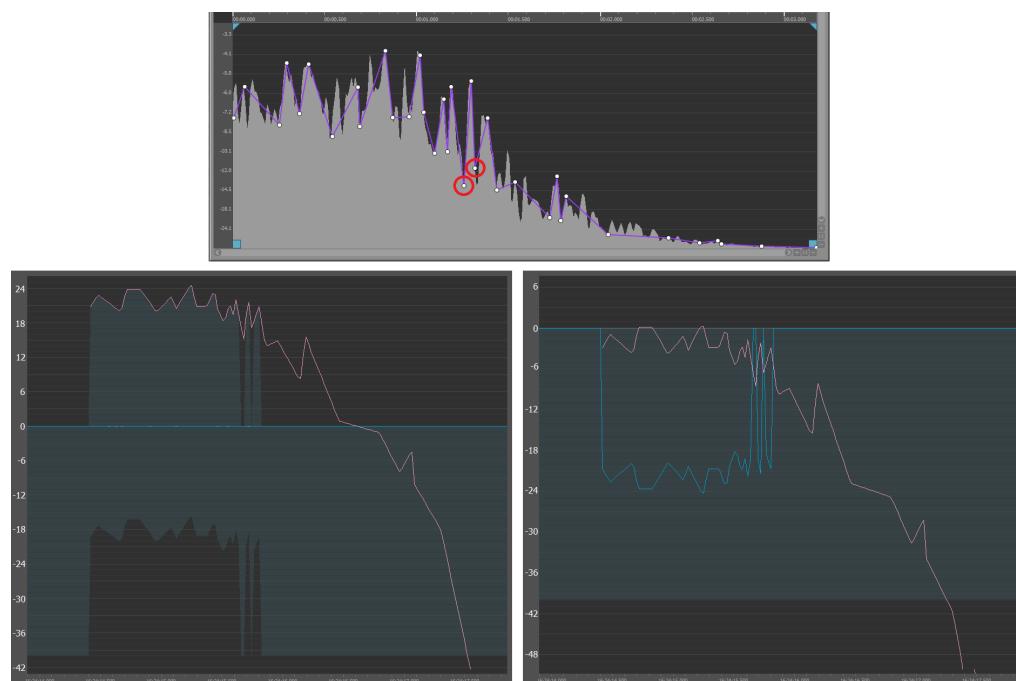


図30.22 エンベロープ編集 (b)

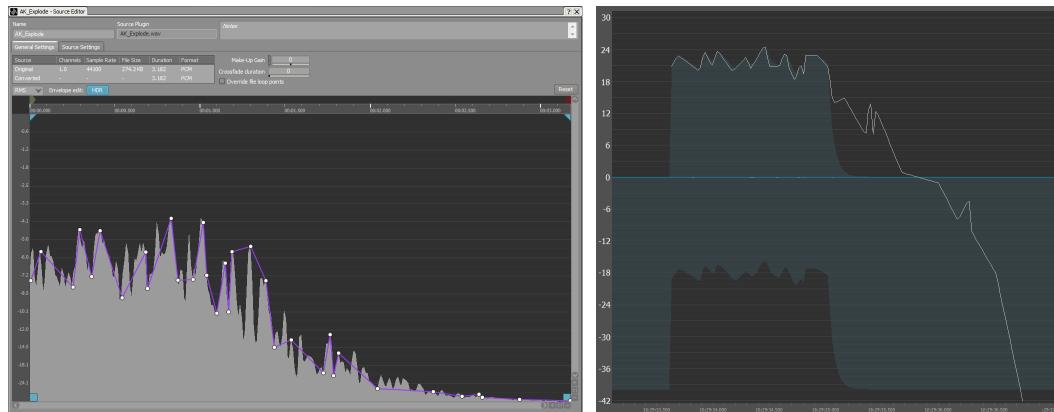
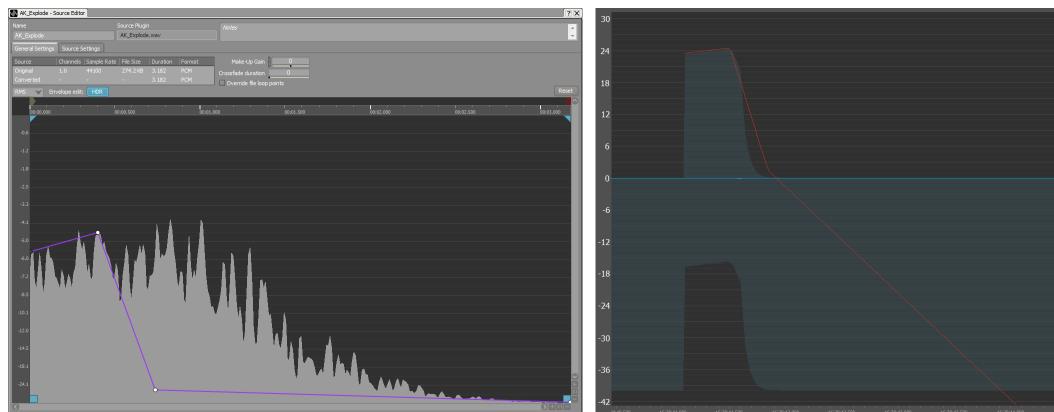


図30.23 エンベロープ編集 (c)



(a) では、爆発サウンドのエンベロープの、ソースエディタ画面が表示されています。赤丸は、アクティブレンジより下のポイントを示します。再生中にエンベロープがROIを出たり入ったりするため、HDRウィンドウが激しく動きます。その影響を受けたバックグラウンドサウンドのボリュームの極端な結果が、アウトプットに明確に出ます。この問題を解決するために (b) では、エンベロープをマニュアルで編集し、問題の2点を排除しています。前述の通り、爆発サウンドのエンベロープを編集しても、HDRウィンドウのみ、つまり他のサウンドのみが影響を受け、爆発サウンド自体は影響を受けません。また、デザイン上の観点から、ダッキングを爆発サウンドの初期インパクト期間だけにして、地響きの部分で早めに他のサウンドをフルボリュームで発揮させたいと考えれば、(c) のようにエンベロープを編集できます。興味深いことに、爆発音は3例どれでも全く同じように再生されます。

### HDRとWwiseのボイスパイプライン

前述の通り、HDRシステムが使うのはオーサリングツールWwiseで設定したロジカルボリュームであり、オーディオデータの実際の振幅は無視されます。HDRシステムのロジックが検討するレベルは、HDRバスへのインプットレベルに相当し、HDRバスをInput busモードで確認すると、Voice Monitorに表示されます。

これらのレベルは、ボイスボリューム、つまりActor-Mixer Hierarchy、Master-Mixer Hierarchyの各種コントロールバス、アクション、RTPC減衰、距離減衰などの効果を合算したものに依存し、さらに、シグナルフローにおいてHDRバスより上流に位置する各種ミキシングバスのゲインにも依存します。複数のシグナルバスがHDRバスに進む時、例えば複数のAUXセンドを使う時は、HDRロジックは最大ゲインのパスを使用します。

オーディオフレームごとに、サウンドエンジンはまず、全てのボイスのボリュームを、HDRバスが見る状態に計算します。次にサウンドエンジンが、HDRシステムのロジックを実行すると、ロジックがHDRウィンドウの位置に従ったグローバル（全体的）HDRゲイン・減衰を返すので、このゲインを各ボイスに適用します。これが終わると、サウンドエンジンは通常通りにボイスを処理、つまり、ボイスをボリュームスレッショルドと比較してバーチャルにするかどうかを判断し、データを出し、ボリュームを適用します。

### メイクアップゲインと、ソースノーマライゼーション

HDRシステムのロジックは、ソースノーマライゼーションと、メイクアップゲインという、Wwiseが提供する2種類のボリュームプロパティを無視します。これらのボリュームコントロール機能は、主にオーディオアセットをロジカルボリュームから独立してノーマライズする際に使い、Wwiseにインポートする前に波形エディタで行える作業と同じです。例えば、ローカリゼーションメイクアップゲインを使って2つのランゲージのボリュームの違いをならすことができますが、HDRシステムの動作は、全てのランゲージで同じでなければなりません。このため、HDRロジックに無視されるボリュームコントロール機能の存在は重要であり、ソースノーマライゼーションやメイクアップゲインが、この役割を担います。

なお、バーチャルボイスシステムや、Voice Monitorビューは、メイクアップゲインやソースノーマライゼーションを考慮していません。しかし、バーチャルボイスはHDR減衰を考慮します。このため、HDRウィンドウより下にあるボイスがバーチャルになることがあります。

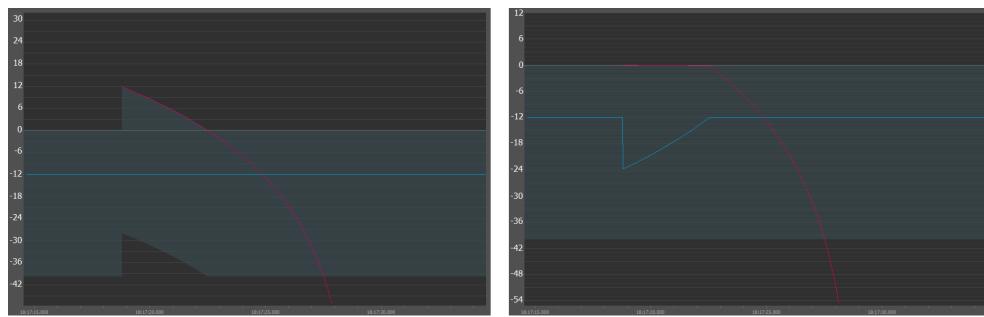
Property EditorのSource Settingsタブにある、Make-Up Gain（メイクアップゲイン）スライダを、デザイン的な観点でHDRと共に利用することもできます。HDRシステムの視点からトランスペアレントなので、（シグナルフローの観点で）「HDR後のボリューム」と解釈することができ、HDRウィンドウとは独立してサウンドのボリュームを変更できます。HDRスレッショルドより上で、単独に再生成されるサウンドは通常、同じレベルでシステムから出てきます。メイクアップゲインを使えば、結果としてボリュームが大きくなります。例えば、FPSゲームで、HDRシステムのダイナミックミキシング機能のメリットを活かしながらも、プレイヤーのガンのボリュームを他よりもかなり大きくしたいとします。その場合は、メイクアップゲインを使って増加させます。

### HDRと距離減衰の関係

距離減衰カーブは、通常通りに設定できます。ただし、HDRシステムで発生する予想外の動作があるので、注意してください。前述の通り、再生するサウンドが、HDRスレッショルドよりも上にあり、オーディオシーン内で最大であれば、システムから0dBFSで出されます。このサウンドが $\pm 50$ m離れたところで再生されても、まだスレッショルドよりも上にあり、シーン内で最大サウンドであれば、やは

りシステムから0dBFSで出されます。結果として、近い距離では減衰カーブが機能しないかのような印象を受けます。しかしそうではありません。実際には、距離が近くなる時のインプットボリュームの増加が、アウトプットボリュームを増加させず、他のサウンドをダッキングするために使われているのです。他に何も再生されていない状態では気付きませんが、大きいサウンドの減衰カーブをデザインする時に、この点を考慮する必要があります。

図30.24 大きいサウンドに対する距離の効果



ここでは、リスナーが大きいサウンド（赤）から、徐々に離れていく様子を示しています。インプット側（左）のボリュームは正しいようですが、アウトプット側（右）では、大きいサウンドがHDRスレッショルドよりも上にある限り、距離が徐々に離れているにもかかわらずボリュームが変化しないので、距離減衰が機能していないかのような誤った印象を与えます。実際には、近距離にいる時の、スレッショルドを超えた分のボリュームオフセットが、大きいサウンドではなく、その他のサウンド（青）に影響を与えるのです。

### HDRウィンドウをインプット変数として使う

HDRウィンドウの位置を、バスのHDRタブの、Windowメーターでモニターできます。また、この数値にGame Parameterをアサインすることも可能です。RPCを使えば、このGame Parameterを、どのオブジェクトのどのプロパティにもマッピングできます。例えば、ウィンドウ位置を利用してサウンドインスタンスの最大数を調整することもできます。

### HDRウィンドウ位置に、Game Parameterを付けるには:

1. HDRバスを確認して、Property Editor>に内容を表示させる。
2. HDRタブをクリックする。
3. グループボックス**Window Top Output Game Parameter**で、 [...]をクリックして、ゲームパラメータを選択する。
4. このGame Parameterが設定される範囲を、HDRバスのインプットとして、デシベル単位で設定する。

## ファイナルミックスの作成

プロジェクトのバス構造が完成したら、ゲームにある様々なサウンド、ミュージック、モーションのミックスの微調整やトラブルシューティングを行います。

Wwiseでは、ゲームオーディオのミキシングのための主なツールとして、以下の2種類があります。

- **Master-Mixer Console** - 様々なバスのプロパティを、1つの画面にまとめて表示する、ミキシングコンソール。
- **Mixing Desk (ミキシングデスク)** - バスやオブジェクトの様々なプロパティを、1つの画面にまとめて表示する、柔軟でパワフルなミキシングコンソール。

どちらのビュー（画面）でも、ゲームの最終的なオーディオミックスを、リアルタイムで作成できます。ただし、Mixing Deskの方が、編集できるプロパティの種類が多く、1つのミキシングセッションでオブジェクトとバスの両方をミックスできるので、コントロールできる範囲が広くなります。



### ダウンミキシング

ミキシングセッションの最終的なアウトプットを適切に調整するには、ダウンミキシングのWwiseのアプローチを理解することが重要です。詳しくは[付録A #####](#)を参照してください。

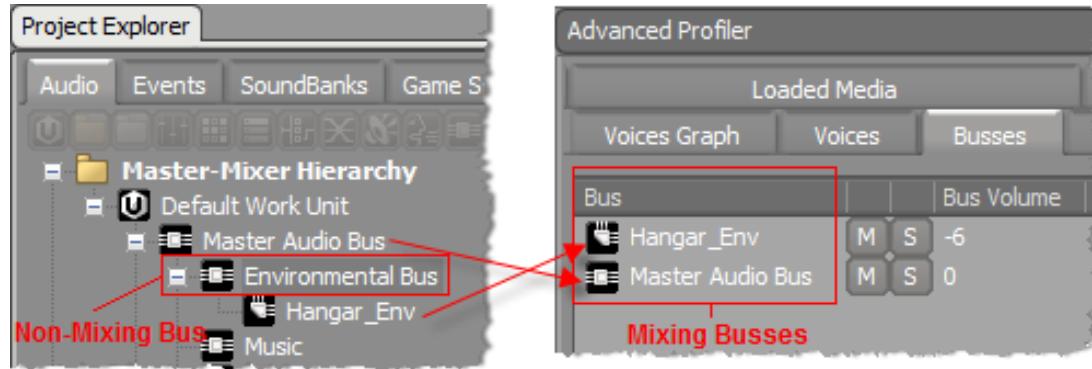
### ミキシングバスと非ミキシングバスの違い

WwiseのMaster-Mixer Hierarchyのオブジェクト（バス）はミキシングバスに設定しない限り、裏で非ミキシングバスとして扱われています。非ミキシングバスはバスのプロパティをインプット側でプロパゲーションしますが、これはCPUやメモリを効率的に使うためです。非ミキシングバスがミキシングバスに昇進するのは、このようなプロパゲーションが以下の理由で行えない時です：

- Master Audio Busであるため、
- Auxiliary Busであるため、
- HDRバスであるため、
- インサートエフェクトが1つ以上あるため、
- ミキサー・プラグインをマウントしているため、
- チャンネル・コンフィギュレーションがParentに設定されていないため、
- ポジショニングが有効になっているため、
- バスのボリュームにRTCPを持っています。

バスのステータスを確認する1つの方法として、キャプチャー中にAdvanced ProfilerのVoices GraphタブやBussesタブを開いてみます。ミキシングバスしか表示されず、非ミキシングバスは表示されません。

Wwise Integration プロジェクトの以下のスクリーンショットでは、Environmental BusはAdvanced ProfilerのBussesタブに表示されないので非ミキシングバスであると考えられる一方、その子のHangar\_Env Auxiliary Busは表示されています。なお非ミキシングバスであるEnvironmental Busのボリュームは、Hangar\_Env Auxiliary BusのBus Volume列に含まれます。



### シグナルレベルをモニターする

Wwiseでシグナルを測定する方法はいくつかあります。その1つがラウドネスレベルです。

#### ラウドネスの概要

ITU-R BS.1770とEBU R128の規格は、オーディオプログラムのラウドネス値やラウドネスレンジを、ラウドネスユニット（LUまたはLUFS）で測定するための勧告や仕様を提示しています。また、トゥルーピークレベルに関する仕様や勧告も提示しています。これらの仕様や勧告は、Wwiseの、Loudness MeterビューとMeterビューに採用されています。

#### 測定値

Wwiseは、PCMシグナルの様々な測定値を提供します。

測定値	内容
Loudness level	デジタルシグナルの、ラウドネス感覚を表す測定値を提供。ラウドネス値は、様々なサイズのタイムウィンドウを使って計算できる。 ラウドネス値はLoudness Meterビューに表示される。 単位: LUFSまたはLU
Loudness range	一定期間に記録されたラウドネス値の統計分析と、ダイナミックレンジの測定値を提供。 ラウドネスレンジはLoudness Meterビューに表示される。 単位: LU
True-Peak level	アナログ世界の最大連続シグナルレベルの推定値を提供。デジタル・アナログコンバータや外部コーデックが飽和状態でないことを確認するために便利な数値。 トゥルーピークレベルはMeterビューに表示される。 単位: dBTP
Peak level	非常に短い時間の、PCMシグナルの最大・最小測定値を提供。 ピークレベルは、Wwiseの以下の場所に表示される。 <ul style="list-style-type: none"> <li>• Wwise Toolbar</li> <li>• Master-Mixer Console</li> <li>• Mixing Desk</li> </ul>

測定値	内容
	<ul style="list-style-type: none"> <li>Meterビュー</li> </ul> <p>単位: dBFS</p>
Root mean square (RMS)	<p>RMSレベルは、ピークメーター値で測定され、非常に短い時間単位で計算された平均値が適用される。</p> <p>RMSレベルはMeterビューに表示される。</p> <p>単位: dBFS</p>

### Loudness Units (ラウドネス単位)

EBU R128とITU-R BS.1770に、ラウドネス測定の単位（ユニット）が定義されています。ラウドネスの測定単位はdBFSではありません。

単位	内容
Loudness Unit (LU)	<p>0 LUは、基準値（デフォルトは、-23LUFS）に等しい。</p> <p>注: デフォルトの23LUFSは、Loudness Meter設定で変更できる。</p>
Loudness Unit Full Scale (LUFS)	ITU-R BS.1770-3とEBU R128によると、0 LUFS (0LKFS) は、0 dBFS @ 1KHzに等しい。
LKFS	<p>LKFSは、LUFSと同じである。</p> <p>ITU-R BS.1770-3は、LKFSを使用。</p> <p>EBU R128は、LUFSを使用。</p>

### ラウドネス測定のスコープ

ラウドネス測定には、3種類のタイムスコープ（時間的範囲）があります。統合モードでは、ゲームのラウドネス値を、より広いスコープで測定できます。

Term	内容
Momentary	横に移動（スライド）する長さ0.4秒の長方形のタイムウインドウを使って計算した、ラウドネス値。測定にゲーティングを使わない。
Short-term	横に移動（スライド）する長さ3秒の長方形のタイムウインドウを使って計算した、ラウドネス値。測定にゲーティングを使わない。
Integrated	ユーザーキャプチャーを使って計算した、ラウドネス値。統合ラウドネスはゲーティングを使用 (-70LUFSより低い値は全て無視) し、瞬間的ラウドネスに基づいた、長期的測定（キャプチャータイム全体の統合）である。Captureボタンをクリックすると測定が開始し、再度クリックすると終了する。Resetボタンをクリックするとキャプチャーが再開する。

### 統合ラウドネスの追加情報

EBU R128による統合ラウドネスの推奨ターゲットレベルは、-23LUFSです。ゲームのターゲットラウドネス値として、-23LUFSを指定することで、ゲームをプレイするアウトプット機器（テレビ、携帯機器など）でアウトプットされる他のソースと同等のアウトプットレベルを、ゲームでも得やすくなります。また、ゲーム全体のラウドネスのターゲットレベルを指定することで、ゲームの異なるレベルや領域において同じラウドネスを確保できます。ゲーム内容はダイナミックなため、ターゲットレベルの-23.0LUFSに向けて厳密なノーマライズを達成することが現実的に難しい部分では、±1.0LUの誤差が許容されます。多くの場合、ゲームのある部分のラウドネス値を予測することは困難です。様々なバックグラウンドエレメン

ト（BGMなど）だけで成立する部分や、意図的に低くミックスされた部分などでは、ラウドネス値が許容範囲から外れることがあります。

ASWG (Sony Worldwide Studios Audio Standards Working Group) の、ASWG-R001は、以下を推奨しています。

- ・家庭用プラットフォーム向けに開発するタイトルのオーディオコンテンツの、平均ラウドネス値は、許容範囲は±2LUでターゲットレベル-23LUFSにノーマライズするが、この許容範囲はインターラクティブ型エンターテイメントのオーディオコンテンツの特性であるノンリニア性を考慮すると容認できる範囲である。
- ・モバイルプラットフォーム向けに開発するタイトルのオーディオコンテンツの、平均ラウドネス値は、許容範囲は、±2LUでターゲットレベル-18LUFSにノーマライズする。
- ・測定対象はオーディオコンテンツ全体とし、ダイアログ、サウンドエフェクト、ミュージックなどに限定してはならない。
- ・オーディオコンテンツの測定は、現実的な範囲でできるだけ長く、最低でも30分は測定し、1つのタイトルで測定する領域は、そのタイトルのゲームプレイの様々な面を全て網羅するように選択された、複数の代表的な部分とする。

### ラウドネスレンジ (LRA) の追加情報

推奨される最大LRA値は、ジャンル、ターゲット層、想定されるゲームプレイ環境などによって異なります。また、モバイルゲームなどの配信プラットフォームによって、最大LRA値が変わることもあります。試聴する環境別のラウドネスレンジの例を、以下に示します。

環境	LRA
ホームシアター	20
居間	18
台所	15
居間（深夜）	9
交通機関（携帯型）	6

以下は、一般的なルールです。

- ・うるさい環境で、リスナーが全てのコンテンツを正しく聞き取るには、低いLRAが要求される。
- ・音量が大きいラウドスピーカーは、より大きいLRAに対応できる。

ラウドネスレンジは、ダイナミックな圧縮が必要かどうかを判断する基準にもなる測定値です。

EBUは、ラウドネスレンジを測定することで、必要に応じて、あるプログラムを視聴者または配信プラットフォームの許容ウインドウ内に収めるダイナミック圧縮の方法を、適切に判断できると提唱しています。実際には、全体的な低レベル圧縮によって納得のゆく効果が得られ、シグナルレンジ全体の一貫した圧縮が確保されると考えられます：

- ・低いスレッショルド (-40dBFS未満)

- 適度な圧縮レシオ (1:1.2 ~ 1:1.5)
- 長いリリースタイム (1秒より長い)

前述の通りEBUは、元のラウドネス値によっては、コンプレッサーのメイクアップゲインの適切な調整で、ターゲットレベル-23LUFSへの移行を並行して実行できると提案しています。

### トゥルーピークの追加情報

トゥルーピーク測定で、アナログクリッピングを検知できます。ピークがサンプル間にある場合でも、あとに続くデジタル・アナログコンバーター、サンプルレートコンバーター、よく使われるコーデックなどで発生する可能性のあるディストーションを、予測して回避できます。

EBU R 128の推奨する最大許容トゥルーピーク値は、-1dBTPです。

### Master Audio Busのトゥルーピークをモニターするには:

1. メニューから、Views > Meter > Meter - Sync Group 1...4 を選択して、4つあるMeterビューの1つを開く。
2. [...]をクリックして、Master Audio Busを選択する。
3. リストで、True Peakを選択して、トゥルーピークをモニターする。

### Wwiseでラウドネスを測定

### Master Audio Busのラウドネスをモニターするには:

1. メニューから、Views > Loudness Meter > Loudness Meter - Sync Group 1...4 を選択して、4つあるLoudness Meterビューの1つを開く。
2. [...]をクリックして、Master Audio Busを選択する。
3. Captureをクリックして、Integrated loudness (統合ラウドネス) 計算を開始する。

### ラウドネス値をキャプチャーするには

1. Loudness Meterビューを開く。
2. ビューのタイトルバーで、シンクグループ選定アイコンから、シンクグループを1つ選択する。
3. セレクタ [...]をクリックして、ラウドネスを測定するバスを1つ選択する。
4. Wwiseツールバーで、Captureをクリックする。  
注意: Loudness MeterビューのCaptureをクリックしないこと。
5. Performance Monitorを開く（または、F6を押してProfilerレイアウトを表示するか、Shift-Nを押す）。
6. Performance Monitorツールバーで、ビュー設定アイコンをクリックする。
7. 最初に選択したシンクグループに対応するラウドネス値を、リストから単数または複数選択して、有効にする（例えば、Momentary Loudness Level - Sync Group 1）。

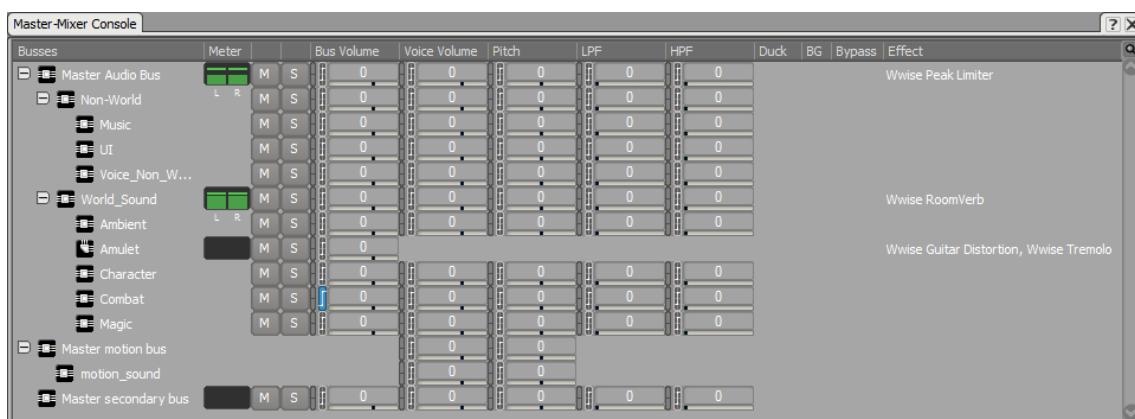
8. OKをクリックする。

### Master-Mixer Consoleについて

Master-Mixer Consoleは、ゲームオーディオの最終的なミックスを作成するためのミキシングコンソールのことです。1つのビューに、バスに関する様々なプロパティがまとめて表示されます。ゲームに接続した状態であれば、このコンソールを使って、ゲームで再生中のオーディオやモーションの各種プロパティを、リアルタイムで試聴しながら調整できます。

また、プロジェクトに設定した様々なバスにルーティングされる各種オブジェクトの詳細情報をモニターすることができます。モニターできる情報の一例を以下のようになります。

- **Meter** - 各チャンネルのピークレベルを表示。
- **Duck** - バスがダッキングされているかどうか。
- **BG** - バスがBGMオプションに結び付いているかどうか。
- **Bypass** - バスの、Bypass All（全てをバイパスする）のオプションが選択されているかどうか。
- **Effect** - バスに適用されている、特定のエフェクト、または複数のエフェクト。



### Master-Mixer Consoleを使って、ミックスを最終調整する

プロジェクト全体のルーティングが完成したら、ゲームに接続してゲームをプレイ中にリアルタイムで様々な相対プロパティを調整して、最終的なミキシングを作成できます。Master-Mixer Consoleを使えば、プロジェクトにある全てのバスの相対プロパティを簡単に調整できます。相対プロパティは累積されるので、設定した相対プロパティ値はオブジェクトに設定してあるプロパティ値に加算されます。

ミキシングの最終調整を行いながら、バス（単数または複数）をミュートまたはソロすることで、ゲームの特定サウンドやモーションのある側面に注目したり、具体的な問題のトラブルシューティングをしたりすることができます。



#### 注記

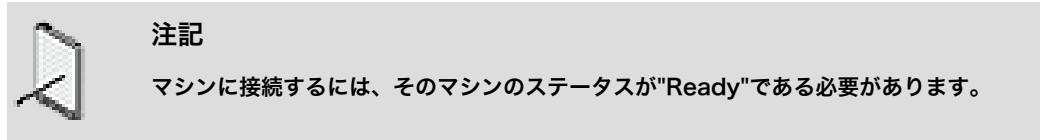
バスのピッチを変更しても、そのバスを通る様々なミュージックオブジェクトに対する影響は一切ありません。

## Master-Mixer Consoleを使って、ファイナルミックスを作成するには:

1. ツールバーのRemoteボタンをクリックして、ゲームに接続する。

Remote Connectionダイアログボックスが開き、ネットワーク上で稼働中のマシンのリストが表示される。

2. リストで接続するマシンを選択する。



3. Connectをクリックする。

Wwiseがリモートゲームマシンに接続する。ゲームマシンの名前がツールバーに表示される。

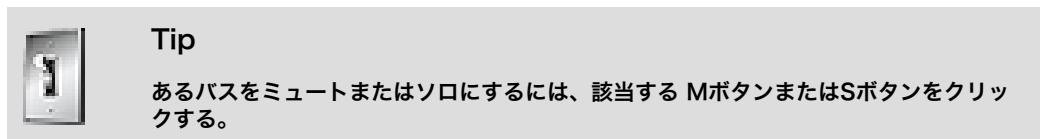
4. メニューバーで、View > Master-Mixer Consoleをクリックする。

Master-Mixer Console が表示される。

5. ゲームの稼働中に、オーディオまたはモーションを試聴する。
6. 以下のプロパティ値を調整するには、数値を入力するか該当スライダをドラッグする。

**Volume** - アウトプットのレベル、つまり振幅。

**Pitch** - アウトプットの、再生スピード

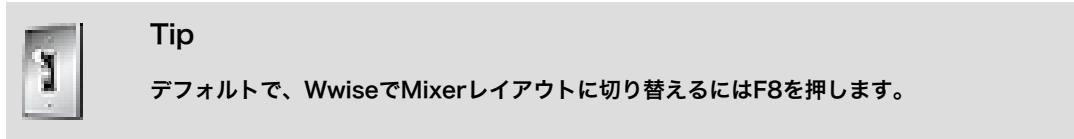


## 関連トピック

- ・「Master-Mixer Consoleについて」
- ・「オブジェクトにエフェクトを適用する」
- ・「バスの相対プロパティを設定する」
- ・「信号をダッキングする」

## Mixing Deskについて

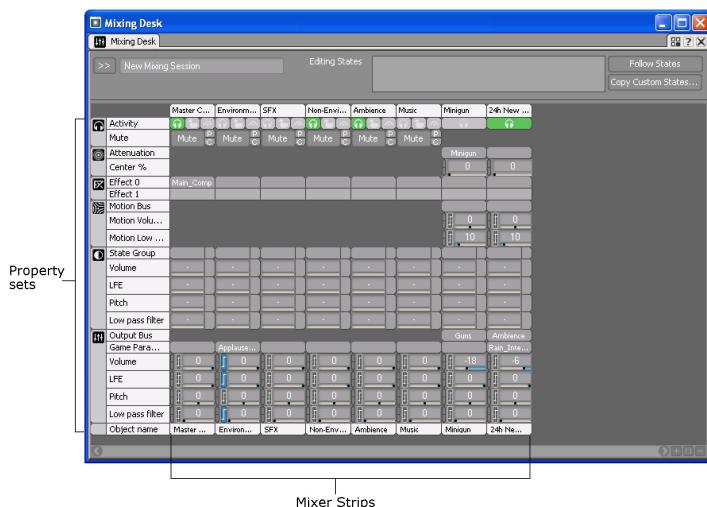
Mixing Deskは、ゲームのオーディオミックスをリアルタイムで調整できるように、様々なプロパティを1つの画面にまとめて表示した、柔軟でパワフルなミキシングコンソールです。どのオブジェクトやバスも、Mixing Deskに追加でき、オブジェクトのルーティング、エフェクトShareSetとAttenuation ShareSetの適用、Stateのプロパティ変更などを行えるほか、オブジェクトやバスのそれぞれのプロパティ調整も可能です。



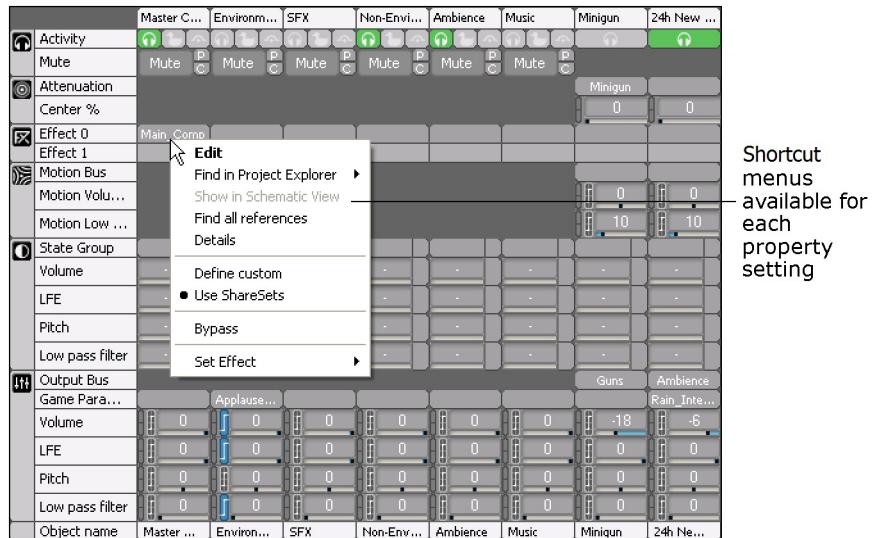
#

また、キャプチャーセッション (Capture session) を開始すると、Mixing Desk 上で各オブジェクトのアクティビティを観察でき、例えば、再生中のボイス、ダッキングされたバス、バイパスされたエフェクトなどがあれば、確認できます。さらに、どのオーディオオブジェクトも、ミュート (Mute) やソロ (Solo) の設定ができるので、オーディオミックスを構成するオブジェクトの個別調整が可能です。

Mixing Deskは、縦にミキサーのミキサーストリップ (Mixer strips) を表示して、角行にWwiseの一般的なプロパティ (Property sets) が示された、グリッド (表) です。1つのミキサーストリップが、1つのオブジェクトを表し、そのオブジェクト名がストリップの上下に表示されます。



また、ミキサーストリップ上に表示されたプロパティに関するコンテキストメニューを、表示できます。コンテキストメニューには、そのプロパティ設定に関連したコマンドが、いくつか表示されます。例えば、エフェクト (Effect 0 ~ 1) の行を1つ右クリックすると、挿入済みのエフェクトの編集 (Edit) 、新規エフェクトの設定 (Set Effect) 、エフェクトのバイパス (Bypass) などを選択できる。コンテキストメニューを表示させるには、ミキサーストリップ上にあるプロパティ設定領域の1つを右クリックします。



Mixing Desk上のエレメントは全て、そのミキシングセッション (Mixing session) の中に保存されます。つまり、ゲームの様々なコンポーネントに合わせて、様々なミキシングセッションを作成できます。また、一旦ミキシングセッションをセットアップしてから、後日、ゲームのオーディオミックスを調整し続けることもできます。

### ミキシングセッションの構築

Mixing Deskを使ってゲームオーディオのミキシングを最終調整する前に、まずミキシングセッション (Mixing session) を作成して内容を設定する必要があります。ミキシングセッションを作成すると、Mixing Deskの内容がプロジェクト内で保存され、あとから再び確認できるので、プリセットしたセッション、つまり Soundcaster (サウンドキャスター) セッションと似ています。作成したミキシングセッションは、Project Explorerの、Sessionsタブにある、該当するWork Unitの下に保存されます。

インターフェース上で分かりやすいように、ミキシングセッションは、以下のアイコンで表されます。

アイコン	内容
	Mixing Session

### ミキシングセッションの作成

Mixing Deskを使い始める前に、ミキシングセッション (Mixing session) を作成する必要があります。ミキシングセッションには、一種のミキシングコンソール上に並べられた、複数のバスやオブジェクトが入っています。ミキシングセッションで使われる全てのバスやオブジェクト用に、スライダやプロパティ値のコントロール機能が表示され、ゲームのオーディオミックスを調整できるようになっています。

ミキシングセッションの作成は、Wwiseの以下の場所で行えます。

- Project ExplorerのSessionsタブ
- Mixing Deskビュー

## Project Explorerで、ミキシングセッションを作成するには:

1. Project Explorerで、Sessionsタブに切り替える。
2. Mixing Sessionsタイトルの下で、以下のいずれかを行う。

ワークユニット（または仮想フォルダ）を選択して、Project Explorerのツールバーの、**Mixing Session**アイコンをクリックする。

Work Unit（または仮想フォルダ）を右クリックして、ショートカットメニューで、**New Child > Mixing Session**をクリックする。

新規ミキシングセッションが、Work Unit（または仮想フォルダ）に追加されます。

3. デフォルトのミキシングセッション名を、内容が分かる名前に変更する。
4. 新規ミキシングセッションをダブルクリックして、これをMixing Deskにロードする。

これで、ミキシングセッションに、バスやオブジェクトを入れる準備が完了する。

## Mixing Deskで、新規ミキシングセッションを作成するには:

1. Mixing Deskで、ミキシングセッション用セレクタボタン(>>)をクリックして、リストで、Newを選択する。  
New Mixing Sessionダイアログボックスが開く。
2. 新規ミキシングセッションを保存する場所のWork Unitを、選択する。
3. Nameフィールドに表示されたデフォルトのミキシングセッション名を、内容が分かる名前に変更する。
4. OKをクリックする。

ミキシングセッションが作成され、次にバスやオブジェクトを入れることができる。

## 関連トピック

- 「Mixing Deskについて」
- 「Mixing Deskに表示させる情報をカスタマイズする」
- 「ミキシングセッションに、オブジェクトやバスを追加するには」
- 「ミキシングセッションの、ミキサーストリップを並べ替える」
- 「ミキシングセッションのミキサーストリップの、サイズ変更」
- 「ミキシングセッションの、ミキサーストリップの削除」
- 「Mixing Deskを使って、オーディオをミキシングする」

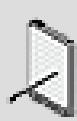
### ミキシングセッションに、オブジェクトやバスを追加するには

ミキシングセッションを作成できたら、プロジェクトにある様々なバスやオブジェクトを入れ始めます。Mixing Deskにバスやオブジェクトを1つ追加すると、ミキサーストリップ (Mixing strip) が1つ、作成されます。ミキサーストリップには、そのオブジェクトまたはバスの、プロパティ設定の概要が表示されます。また、複数のコントロール機能があり、様々なアクティビティをモニターできるほか、プロパティの変更、エフェクトやStateの設定、ポジショニングのプロパティ設定などができます。



Mixing Deskに追加できるオブジェクトは、以下の通りです。

- Busses
- Actor-mixers
- Containers
- Sounds
- Music Containers
- Music Segments
- Music Tracks



#### 注記

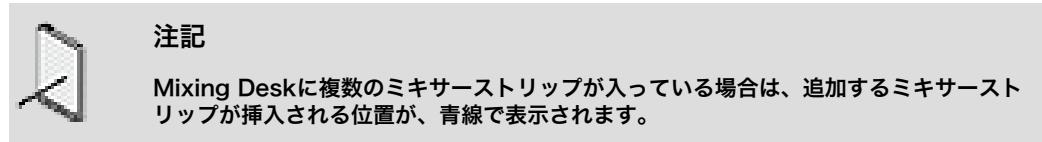
ミキシングセッションに入っているミキサーストリップが、既にアンドロードされているオブジェクトまたはバスに関するものである場合は、そのミキサーストリップは空の状態になります。

## ミキシングセッションに、オブジェクトやバスを、追加するには:

1. Mixing Deskに、ミキシングセッションをロードするには、以下のいずれかを行う:
  - Mixing Deskのセレクタ (>>) をクリックして、リストのミキシングセッションを1つ選択する。
  - Project Explorerの、Sessionsタブで、ミキシングセッションを1つ、ダブルクリックする。
2. バスまたはオブジェクトを1つ、Project Explorerから、Mixing Deskに、ドラッグする。

このオブジェクトまたはバスを表すミキサーストリップ (Mixing strip) が1つ、Mixing Deskに作成される。

3. 引き続き、Mixing Deskにオブジェクトやバスを追加する。



## 関連トピック

- 「Mixing Deskについて」
- 「Mixing Deskに表示させる情報をカスタマイズする」
- 「ミキシングセッションの作成」
- 「ミキシングセッションの、ミキサーストリップを並べ替える」
- 「ミキシングセッションのミキサーストリップの、サイズ変更」
- 「ミキシングセッションの、ミキサーストリップの削除」
- 「Mixing Deskを使って、オーディオをミキシングする」

### ミキシングセッションの、ミキサーストリップを並べ替える

ミキシングセッションにバスやオブジェクトを追加できたら、ミキサーストリップをドラッグして位置を変えることで、自由に並べ替えられます。

### ミキシングセッションのミキサーストリップを、並べ替えるには:

1. Mixing Deskに、ミキシングセッションを1つロードする。
2. ミキサーストリップのタイトルバーをクリックして、Mixing Desk上の別の位置までドラッグする。

ミキサーストリップの移動先の位置が、青線で表示されます。

3. 引き続き、必要に応じてミキサーストリップの並べ替えを行う。

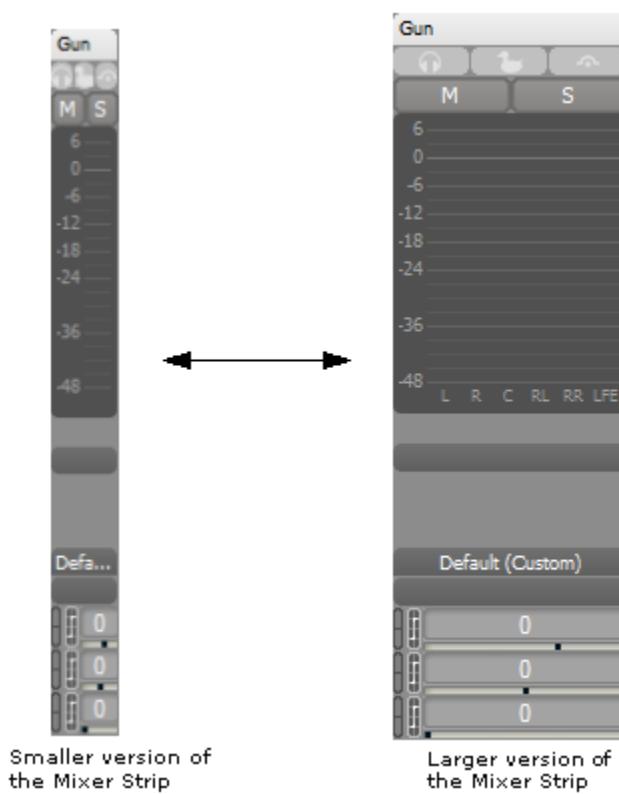
## 関連トピック

- 「Mixing Deskについて」

- 「Mixing Deskに表示させる情報をカスタマイズする」
- 「ミキシングセッションの作成」
- 「ミキシングセッションに、オブジェクトやバスを追加するには」
- 「ミキシングセッションのミキサーストリップの、サイズ変更」
- 「ミキシングセッションの、ミキサーストリップの削除」
- 「Mixing Deskを使って、オーディオをミキシングする」

### ミキシングセッションのミキサーストリップの、サイズ変更

ミキサーストリップ (Mixer strip) が小さすぎる場合や、大きすぎる場合は、サイズを変更できます。



### ミキサーストリップの大きさを変更するには:

1. Mixing Deskに、ミキシングセッションを1つロードする。
2. Mixing Deskの、グリッド部分をクリックする。
3. 以下のいずれかを行う:

ミキサーストリップを全体的に大きくするには、Ctrlキーを押しながらマウスのスクロールホイールを上にスクロールするか、スクロールバーの横にある「+」クリックする。

ミキサーストリップを全体的に小さくするには、Ctrlキーを押しながらマウスのスクロールホイールを下にスクロールするか、スクロールバーの横にある「-」クリックする。

## 関連トピック

- ・「Mixing Deskについて」
- ・「Mixing Deskに表示させる情報をカスタマイズする」
- ・「ミキシングセッションの作成」
- ・「ミキシングセッションに、オブジェクトやバスを追加するには」
- ・「ミキシングセッションの、ミキサーストリップを並べ替える」
- ・「ミキシングセッションの、ミキサーストリップの削除」
- ・「Mixing Deskを使って、オーディオをミキシングする」

### ミキシングセッションの、ミキサーストリップの削除

ミキシングセッション中に、いつでも、Mixing Deskにあるミキサーストリップ（単数または複数）を削除できます。

### ミキシングセッションのミキサーストリップを、削除するには

1. 以下のいずれかを行う：

- ・ミキサーストリップのタイトルバーを右クリックして、メニューで、**Remove**を選択する。
- ・削除するミキサーストリップのタイトルバーをクリックして、Deleteキーを押す。

このミキサーストリップが、ミキシングデスクから削除されます。

## 関連トピック

- ・「Mixing Deskについて」
- ・「Mixing Deskに表示させる情報をカスタマイズする」
- ・「ミキシングセッションの作成」
- ・「ミキシングセッションに、オブジェクトやバスを追加するには」
- ・「ミキシングセッションの、ミキサーストリップを並べ替える」
- ・「ミキシングセッションのミキサーストリップの、サイズ変更」
- ・「Mixing Deskを使って、オーディオをミキシングする」

### Mixing Deskに表示させる情報をカスタマイズする

Mixing Deskは、カスタマイズが可能で、表示される情報の種類を変えることができます。Mixing Desk Settingsダイアログボックスを使えば、表示する情報の種類の追加や削除、プロパティ用スライダまたはフェーダの選択、フェーダの高さ設定などができます。

### Mixing Deskに表示させる情報をカスタマイズするには:

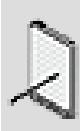
1. Mixing Deskを開くために、以下のいずれかを行う。
  - ・メニューバーで、Views > Mixing Deskをクリックする。
  - ・Ctrl+Shift+Mを押す。

2. Mixing Deskのタイトルバーで、View Settingsアイコンをクリックする。



Mixing Desk Settingsダイアログボックスが開く。

3. Mixing Deskで表示させたい情報の種類の、チェックボックスを全て選択する。



#### 注記

各オプションの概要を見るには、Mixing Desk Settingsダイアログボックスの、Helpアイコンをクリックする。

4. 縦型フェーダーの表示を選択した場合は、テキストボックスVertical fader heightで、フェーダーの高さを指定する。

5. OKをクリックして、設定内容を適用する。

## 関連トピック

- 「Master-Mixer Consoleについて」
- 「Mixing Deskについて」
- 「ミキシングセッションの作成」
- 「ミキシングセッションに、オブジェクトやバスを追加するには」
- 「ミキシングセッションの、ミキサーストリップを並べ替える」
- 「ミキシングセッションのミキサーストリップの、サイズ変更」
- 「ミキシングセッションの、ミキサーストリップの削除」
- 「Mixing Deskを使って、オーディオをミキシングする」

## Mixing Deskを使って、オーディオをミキシングする

Mixing Deskは、ゲームにある様々なオーディオコンポーネントをミックスするために、最適のツールです。ゲームに接続してからキャプチャーセッションを開始すれば、オーディオのアクティビティをモニターしながら、ミキシングセッションの中でリアルタイムに、様々なオブジェクトやバスの、個々のプロパティを調整できます。



#### Tip

デフォルトで、WwiseでMixerレイアウトに切り替えるにはF8を押します。

#

## Mixing Deskを使って、オーディオをミキシングするには:

1. Mixing Deskに、ミキシングセッションを1つロードする。
2. ツールバーのRemoteボタンをクリックして、ゲームに接続する。

Remote Connectionダイアログボックスが開き、ネットワーク上で稼働中のゲーム機（Console）のリストが、表示されます。

3. リストで、接続するゲーム機を1つ選択する。

4. Connectをクリックする。

Wwiseが、ゲーム機にリモート (Remote) 接続する。接続したゲーム機の名前が、ツールバーに表示される。

5. Start Captureボタンが未選択であれば、クリックして、キャプチャーセッションを開始する。

これで、ゲームのオーディオに関連した、以下のアクティビティをモニターできます。

アイコン	Activity	内容
	Voice playback	オブジェクトが再生中であることを示す。バスの場合は、再生中のボイスをこのバスにルーティングすることを示す。
	Ducking	バスが、ダッキングされていることを示す。
	Effect bypass	インサートされたエフェクトが、バイパスされていることを示す。

6. ゲームの稼働中に、オーディオを試聴する。

7. ミキシングを調整するには、以下のツールを利用する:

- プロパティ設定用のスライダ、フェーダ: ミキシングセッションに入っているオブジェクトやバスの、それぞれのボリューム、ピッチ、ローパスフィルタなど、各種プロパティの調整に使う。
- コンテキストメニュー: 選択中のプロパティに関する、具体的なコマンドをアクセスするために使う。例えば、エフェクト (Effect 0 ~ 1) の行を1つ右クリックすると、挿入済みのエフェクトの編集 (Edit)、新規エフェクトの設定 (Set Effect)、エフェクトのバイパス (Bypass)などを選択できる。
- Follow states (Stateを追う) : ゲーム中で発生するStateの変移を追うための設定。



#### Tip

いつでも、ミキサーストリップにあるオブジェクト名、バス名、State名、またはGame Parameter名を、ダブルクリックすることで、該当するエレメントを、Property Editorにロードできます。

## 関連トピック

- 「Master-Mixer Consoleについて」
- 「Mixing Deskについて」
- 「Mixing Deskに表示させる情報をカスタマイズする」
- 「ミキシングセッションの作成」
- 「ミキシングセッションに、オブジェクトやバスを追加するには」
- 「ミキシングセッションの、ミキサーストリップを並べ替える」
- 「ミキシングセッションのミキサーストリップの、サイズ変更」
- 「ミキシングセッションの、ミキサーストリップの削除」

---

## 第31章 プラットフォームとランゲージのバージョン管理

概要 .....	568
複数のプラットフォームに対応するオーサリング .....	568
プロジェクトのローカリゼーション作業 .....	589
バージョンに関する、コツとベストプラクティス .....	594

## 概要

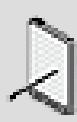
ゲーム制作の現場では、常に品質の最適化と、メモリ消費のバランスを見極める必要があり、様々なプロパティ値やアセットを調整して試しながら、それぞれのプラットフォームの制約内で、できる限り質の高いゲーミングエクスペリエンスを提供する手段が求められます。Wwiseでは、1つのプロジェクトの中で、プラットフォーム別に、アセット、プロパティ、コンバージョン設定などをカスタマイズできます。さらに、プロジェクトのローカリゼーション作業のために、ランゲージ用の一部のプロパティやコンバージョン設定も、カスタマイズできます。シミュレーションやプロファイリングのツールを使い成果を検証して、全てのランゲージバージョンやプラットフォームバージョンが、正しく構築されているかを確認できます。

プロジェクトが対応する全てのランゲージやプラットフォーム用に、オーディオが完成したら、それぞれのSoundBankを、同時に生成できます。SoundBankに関する作業については[35章 SoundBankの管理](#)を参照してください。

## 複数のプラットフォームに対応するオーサリング

複数のプラットフォームに対応するには、プロジェクトを複数に分ける、というのがオーディオデザイナーの通常の考え方です。Wwiseでは、1つのプロジェクトで全てに対応できます。新しいプロジェクトを作成した時点で、自動的に、複数のプラットフォームに対応できる体制が整備され、同時開発も可能です。デフォルトとして、対応可能な全てのプラットフォームが、プロジェクト上で有効になっていますが、いくつかのプラットフォームをプロジェクトから削除すれば、ワークフローの効率性を上げることができます。プロジェクトからプラットフォームを削除する方法については「[プラットフォームを削除するには](#)」を参照してください。

プロジェクトを作成して、アセットやミュージックをインポートした後に、プラットフォーム別にカスタマイズできる、コンバージョン設定ShareSet (ShareSet) を設定し、また、各プラットフォームの長所や制約に従って、アセットの使い方を、プラットフォームごとに定義します。



### 注記

Wwiseで様々なプラットフォーム向けにオーサリングするのは、それらのプラットフォーム向けの開発の権利があることが前提です。新たに別のプラットフォームの権利を取得した場合は、その追加プラットフォームを含めるために、インストール用パッケージをアップデートする必要があります。

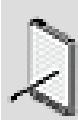
Wwiseでは、対象プラットフォームごとにプロジェクトをカスタマイズするために、以下を行うことができます。

- 「オーディオファイルをコンバージョンする」
- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「他のプラットフォームに、切り替える」
- 「あるプラットフォームの設定を、他のプラットフォームにコピーする」

## オーディオファイルをコンバージョンする

メディアファイルをWwiseにインポートできたら、各種プラットフォーム向けに、ファイルのコンバージョンを行います。メディアファイルの多くに、同じコンバージョン設定を使えるため、最初にいくつかのコンバージョン設定ShareSet (Conversion setting ShareSet) を作成して、これらのShareSetを、プロジェクトの様々なオブジェクトに適用できます。プロジェクト階層にある他のプロパティと同様に、コンバージョン設定のShareSetも、親から子に継承されます。もちろん、階層のどのレベルにおいても、設定をオーバーライドできるので、階層の特定のオブジェクトに対して、別のコンバージョン設定ShareSetを適用することもできます。

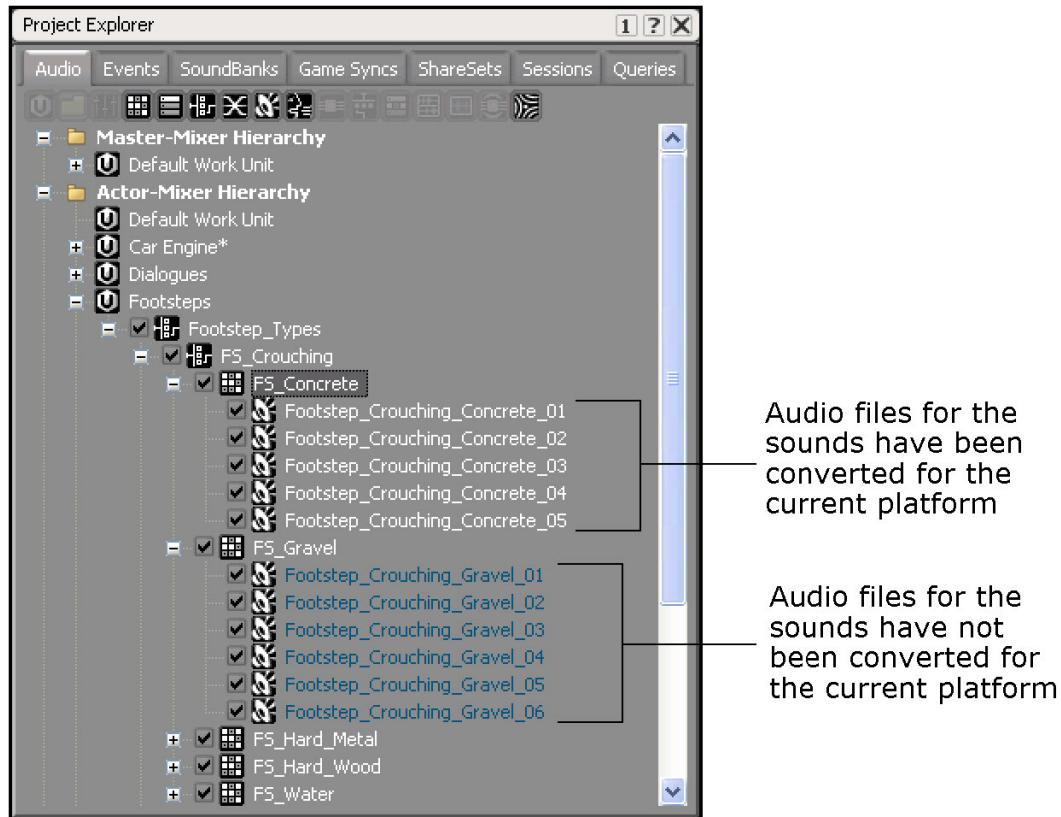
メディアファイルのプロパティや動作を設定する前にコンバージョンを行うことは、必須ではありませんが、SoundBank (SoundBank) 生成前に、各オブジェクトにコンバージョン設定ShareSetを1つずつアサインして、カスタマイズ設定のメリットを活かすことが推奨されます。ShareSetを設定していないオブジェクトには、Wwiseがデフォルトのコンバージョン設定ShareSetを適用します。デフォルトのコンバージョン設定ShareSetについては「[コンバージョン設定のデフォルト指定](#)」を参照してください。



### 注記

Wwiseは、コンバージョン処理を迅速に行うために、マシン上のCPUコアを全て使います。

現在のプラットフォーム (Current platform) 向けにコンバージョンされていないソースに関するサウンドオブジェクトは全て、Project Explorerの、Audioタブで、青字表示されます。

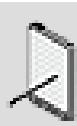


コンバージョンには、以下の3つの処理が伴います。

- ・「オーディオ用の、コンバージョン設定ShareSetを作成する」
- ・「コンバージョン設定ShareSetを、オブジェクトにアサインする」
- ・「オーディオファイルをコンバージョンする」

#### オーディオ用の、コンバージョン設定ShareSetを作成する

コンバージョン設定を管理するために作成するShareSet (ShareSets) は、プロジェクトのニーズや、各アクティブラットフォームの要件に基づいています。ここで決めた設定の多くが、オーディオプロジェクト全体のパフォーマンスや品質に、多大な影響を与えます。プロジェクトのオブジェクトにShareSetを適用した後にコンバージョン設定ShareSetを調整できるので、対象プラットフォームやゲーム自体の制約の範囲内で最大限の効果を達成できます。また、オーディオファイルをインポートする時に、ランゲージバージョンやソースバージョン別に既に設定してあるShareSetを再利用することで、より早く作業できます。



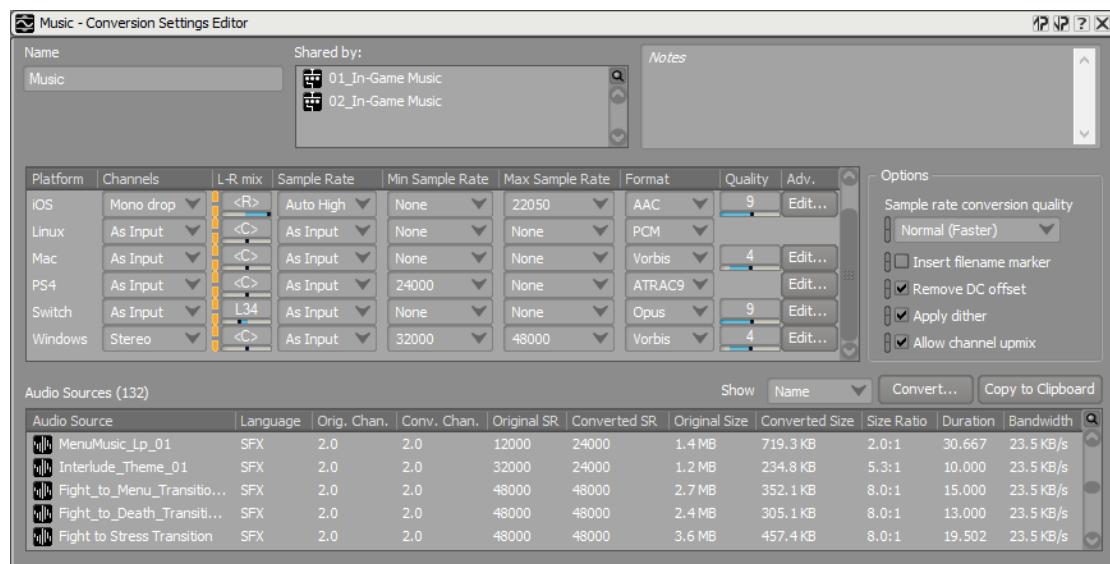
#### 注記

ShareSetの1つを、プロジェクトのデフォルトコンバージョン設定として採用することができます。プロジェクトのデフォルトコンバージョン設定の設定方法については「[コンバージョン設定のデフォルト指定](#)」を参照してください。

Conversion Settings Editorは、以下の2つの主要セクションに分かれています。

## プラットフォームとランゲージのバージョン管理

- **Settings:** Audio Sourcesの上のエリア。サンプルレート (Sample Rate)、オーディオフォーマット (Format)、チャンネル数 (Channels)などのコンバージョンの詳細をプラットフォーム別に設定する。
- **Results:** 全てのオーディオソースのリストを表示したエリア。チャンネル数 (Chan.)、サンプルレート (SR)、ファイルサイズ (Size)などの、ソースのオリジナル (Original) とコンバージョン後 (Converted) を比較できる。

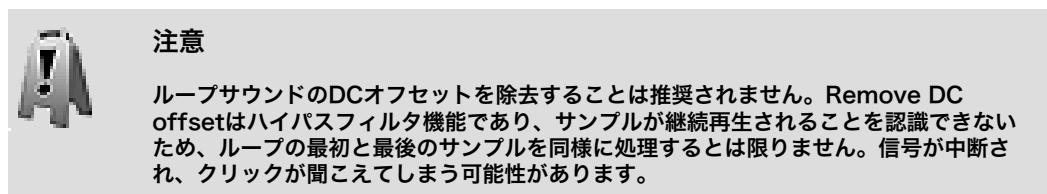


オーディオのコンバージョン処理後も、ピッチ (Pitch) と長さ (Duration) は、オリジナルファイルと同じままですが、コンバージョン設定で以下のプロパティを変更できます。

- Number of channels
- Left-Right mix
- Sample rate
- Audio format
- Audio format quality (オーディオフォーマット品質)
- Sample rate conversion quality (サンプルレートのコンバージョン品質)

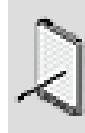
また、以下の設定を選択できます。

- 吹き替え用や字幕用に、ファイルネームマーカー挿入 (Insert filename marker)
- Remove DC offset



- Apply Dither

- Allow channel upmixは、チャンネルがStereoまたはStereo dropと指定されている場合にモノソースファイルをステレオに変換する。



#### 注記

本オプションを選択しないと、Channels設定に関わらずモノソースファイルはモノのままとなる。

### オーディオチャンネルについて

マルチチャンネルのオーディオソースを変換する場合は、どのチャンネルを維持するのかを判断する必要があります。詳しくは「[チャンネルコンフィギュレーション](#)」を参照してください。

Wwiseで選択できるオプションを、以下に示します。

チャンネルのオプション	内容	Comments
As Input	オリジナルのメディアファイルのオーディオチャンネル数を、維持する。	プラットフォームによってサポートされていないオーディオフォーマットもあり、その場合はマルチチャンネルファイルがステレオにダウンミックスされる。
Mono	全チャンネルを、1チャンネルにミックスする。	L-R Mixは、Stereo to Monoコンバージョンだけに使われる。他のチャンネルコンフィギュレーションは <a href="#">付録A ダウンミキシングの動作</a> に従いダウンミックスする。 LFEチャンネルは、常にドロップされる。
Mono drop	最初のチャンネル以外を、全てドロップする。	オリジナルファイルのチャンネルコンフィギュレーションに従い、第1チャンネルをL、またはCとする。
Stereo	全チャンネルを、FLとFRにミックスする。	L-R Mixは、Mono to Stereoコンバージョンだけに使われる。他のチャンネルコンフィギュレーションは <a href="#">付録A ダウンミキシングの動作</a> に従いダウンミックスする。 LFEチャンネルは、常にドロップされる。
Stereo drop	LとRチャンネルに定義したチャンネル以外の、全てのチャンネルをドロップする。	LやRチャンネルとして定義されるチャンネルが全くなく、Cチャンネルが定義されている(Mono)場合は、以下のコンバージョンを行う。 Left = 0.707C、Right = 0.707C コンバージョン後のファイルサイズは、オリジナルファイルの2倍になる。

なお、Wwiseでは、マルチチャンネルエンコーディングを一切行わず、LPCMデータをステレオまたは5.1サラウンドで、ゲーム機やシステムにフィードするだけです。ゲーム機またはシステムがLPCMデータを受信すると、Dolby、DTS、DPL2など、そのゲーム機やシステムが対応するほぼ全てのフォーマットで、アウトプットできます。ただし、一定の制限があり、以下にその一部を示します。

- Android、iOSプラットフォーム上でのステレオ出力のみをサポートします。
- Switch™ プラットフォームでのステレオと5.1サラウンド出力のみをサポートします。

- MacとtvOSプラットフォームでのステレオと5.1および7.1サラウンド出力のみをサポートします。
- その他のWindows、Playstation 4、Xbox Oneなどのプラットフォームは最大7.1チャンネルアウトプットにネイティブで対応している。Wwiseは全ての標準チャンネルコンフィギュレーション（最大13.1チャンネル）に対応でき、さらにアノニマスコンフィギュレーション（最大256チャンネル）にも対応している。なお、これらのコンフィギュレーションの場合は、それを解釈したり専用ハードウェアに送信したりできる特別なシンク（sink）プラグインを使う必要がある。

### サンプルレートについて

サンプルレートとは、デジタルオーディオ信号を1秒間にサンプルする回数です。サンプルレートの決定には、多くの要因が関係し、他の様々な「品質」対「パフォーマンス」の選択と同様に、サンプルレートの設定もまた、バランスの問題となります。コントロール範囲をできるだけ広げるために、Wwiseでは、以下の通り、サンプルレートコンバージョン設定で、多様な選択肢を提供しています。

- **As Input**（インプットのまま） - オリジナルファイルと同じサンプルレートを使い、ファイルをコンバージョンする設定。このサンプルレートが、特定のプラットフォームやオーディオフォーマットで採用できない場合は、採用できる最も近いサンプルレートが使われる。
- **Auto (Low/Medium/High)** - WwiseでファイルのFFT分析を行い結果、選択されたサンプルレートを使い、ファイルをコンバージョンする設定。品質設定をLow/Medium/High（低・中・高）から選ぶことで、アルゴリズムに使用するカットオフ閾値を変えられる。Project Settingsダイアログボックスで、これらの品質設定の閾値を設定することで、品質のレベルを調整できる。Wwiseが行うサンプルレートの自動検知については「[サンプルレートの自動検知設定](#)」を参照。
- **300 to 48000** - 具体的なサンプルレートを使い、ファイルをコンバージョンする設定。サンプルレートの範囲はプラットフォームによって異なり、最大48,000Hzである。



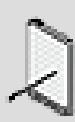
#### 注記

As InputやAutoの場合は、Min Sample RateとMax Sample Rateのペアのボックスを使ってサンプルレートをさらに制限できます。

### オーディオフォーマットについて

オーディオファイルのコンバージョンを行う前に、どのフォーマットにコンバージョンするのかを決めます。Wwiseは複数の異なるオーディオフォーマットに対応することで、プラットフォーム固有の制約に対処するための柔軟性と操作性を高めています。Wwiseが対応する様々なオーディオフォーマットを以下に示します：

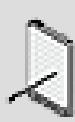
- **AAC** - Mac および iOS プラットフォーム上でのWwiseで使用できる知覚コーディング音声圧縮方式。AACは、同様のビットレートにおいてMP3より優れた音質を実現すると言われている。圧縮は変数・コンテンツ依存で、品質設定は "quality (品質)" スライダでコントロール可能。iOSでは、ハードウェア支援のコーデックが使用可能な場合、これによりAACがデコードされる。iOSハードウェアは、一度に一つのAAC音声のみデコードできることに注意。



### 注記

iOSは、AACのハードウェアサポートを提供するので、このプラットフォームで再生するバックグラウンドミュージックには、AACフォーマットが推奨されます。ただし、AACサウンドが同時に2つ以上再生されることがないように、必ず確認してください。別のAACサウンドが追加されると、ソフトでデコードされるので、多大なCPU負荷がかかります。このため、Music Segment同士がオーバーラップすることの多いインタラクティブミュージックでは、このフォーマットを使わないでください。なお、ハードウェアアシストデコーディングの場合、Wwiseのプロファイラ機能で確認できるCPU負荷（CPU usage）が、非常に高く見えます。これは、デコーディングが、Wwiseのオーディオ処理スレッド内で行われるためです。しかし、実際にはこの時間の大部分は、ハードウェアを待っている時間であり、その間はゲームの他のスレッド用にCPUを使うことができます。また、新しいAACボイスをスタートさせるのに、かなりの時間がかかることがあります。iOSでボイススタートベーションが発生する可能性もあります。AACは、ループに関する制限がありませんが、サンプルアキュレートな再生には向いていません。許容されるチャンネルコンフィギュレーションは、モノ、ステレオ、0.1、1.1、5.0、5.1だけです（その他のコンフィギュレーションは、自動的にステレオにダウンミックスされます）。なお、iOSは、マルチチャンネルのファイルや、LFEチャンネルに対応していません。

- **ADPCM** - サウンド信号とそのサウンド信号から作られた予測の差分を量子化する、オーディオファイル変換・符号化（エンコード）方式。ADPCM量子化ステップは適応的なので、信号を直接量子化するPCM（パルス符号変調）符号化とは異なる。基本的にADPCMは音質を犠牲にしてメモリやCPUを非常に効率的に使う。このため、主にモバイルプラットフォームで採用される。
- **ATRAC9** - PS4向けの知覚符号化方式のことで、非常に優れた知覚音質を維持しながら、さまざまなビットレートでのオーディオファイル符号化を可能にする。
- **Opus** - 音声と汎用オーディオの両方に最適化された、低レイテンシのオーディオコーデック。音質を損なうことなく他のコーデックより優れた圧縮率を発揮します。
- **PCM** - 別個の2進数表現すなわちパルス符号を選択する、オーディオファイル変換・符号化（エンコード）方式。暗号化された2つのポイント間を測定して、最も近いポイントに関連する数値を選択することにより量子化する。
- **Vorbis** - 非常に優れた知覚音質を保ちながら固定または変動ビットレートでのオーディオファイルエンコードを可能にする知覚符号化（エンコード）方式。データ圧縮効率と知覚音質のバランスは、Quality Factor（品質係数）設定の使用またはチャンネルあたりの最大、最小および平均ビットレート指定によりコントロールされる。Vorbisエンコーダは、シークテーブルが必要な場合がある。詳しくは「[Vorbisエンコーダに、シークテーブルを使う](#)」を参照してください。

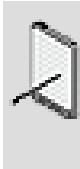


### 注記

Audiokineticの特別な実装により、Vorbisが全プラットフォーム向けに、高度に最適化されています。

- **XMA** - ハードウェアサポートのXbox One向け知覚符号化オーディオ圧縮方式。XMAは、Windows Media Audio Proのコンソール最適化バージョン。圧縮は変数・コンテンツ依存、品質設定は「圧縮品質（Compression Quality）」スライダより制御可能。最新版のXMA 2.0は、XMAデータのシークを容易にするシークテーブル作成のためのブロックサイズパラメータを備える。XMAオーディオ

ディオフォーマットが対応しているのは、モノファイルとステレオファイルだけです。Xbox Oneでは最大7.1までの設定がサポートされています。



#### 注記

VorbisとXMAは変換コーデックなので、シームレスなループが可能ですが、サンプルアキュレートのループができません。この2種類のオーディオフォーマットでは、素材や、元のループポイントの位置によって、結果がかなり異なります。XMAファイルには、最初と最後にサイレンスのパディングがあります。ファイルが非常に小さい場合、特に連続して再生する場合は、このパディングによるレイテンシが目立つ可能性があります。VorbisはXMAと比較して、同じビットレートにおいては品質が明らかに高いので、可能な限りVorbisを使用する。サンプルアキュレートに関する問題が発生した場合は、ADPCMや、PCMも、試してください。

どのフォーマットも、長所や短所があるので、ゲームのCPUやメモリの制約によって、選択するフォーマットを決めてください。どのオーディオフォーマットを、どこで使うべきかについては「[オーディオフォーマット](#)」を参照してください。

プラットフォームにより仕様が異なるため、どのプラットフォームでも全てのフォーマットを使えるわけではありません。各プラットフォームで使用できるフォーマットを、下表に示します。

Platform	AAC	ADPCM	ATRAC9	OPUS	PCM		Vorbis	XMA
Android	#	✓	#	#	✓		✓	#
iOS	✓	✓	#	#	✓		✓	#
Linux	#	✓	#	#	✓		✓	#
Mac	✓	✓	#	#	✓		✓	#
Switch	#	✓	#	✓	✓		✓	#
PS4	#	✓	✓	#	✓		✓	#
Windows	#	✓	#	#	✓		✓	#
Windows Phone	#	✓	#	#	✓		✓	#
Xbox One	#	✓	#	#	✓		✓	✓



#### 注記

マルチチャンネルファイルの全てのコンフィギュレーションに対応しているオーディオフォーマットは、ADPCM、PCM、Vorbisのみです。オーディオフォーマットが対応しないチャンネルコンフィギュレーションの場合は、対応している次のコンフィギュレーションにWwiseがダウンミックスします。



#### 注記

Opusは現在、Mac版のWwiseで開発する場合、使用できません。

### DCオフセットについて

DCオフセットは、Wwiseのボリュームに影響したりノイズを発生させたりすることがあるので、DCオフセットフィルタを使って除去すると良いでしょう。ただ

し、サンプルアキュレートなコンテナの場合は、DCオフセットを除去してはいけない状況もあります。また、サウンドが0dBにノーマライズされている場合など、状況によっては、DCオフセットの除去が必要となることがあります。なお、コンバージョンプロセスで、DCオフセットがデフォルトで除去されます。しかし必要に応じて、Conversion Settingダイアログボックスで、この設定を無効にできます。



#### 注意

オーディオソースから直接、モーションを生成している場合は、DCオフセットを除去すると、ゲーム機用のモーションアウトプットに影響するがあるので、注意してください。

### コンバージョン設定ShareSetを作成するには:

1. Project Explorerで、ShareSetsタブに切り替える。
2. Conversion Settingsのセクションで、新規シェアセットを作成する場所となる Work Unitを1つ選択する。
3. Project Explorerのツールバーの、Conversion Settingsアイコン をクリックする。

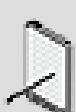
選択したワークユニットに、新規コンバージョン設定シェアセットが作成されます。

4. 適切なShareSet名を入力して、Enterを押す。
5. 新規ShareSetをダブルクリックして、Conversion Settings Editorにロードする。
6. プラットフォームごとに、以下から1つ選択して、それぞれのチャンネル設定を指定する。
  - **As Input** - オリジナルのメディアファイルのオーディオチャンネル数を、維持する。
  - **Mono** - 全チャンネルを、1つのモノチャンネルにミックスする。
  - **Mono drop** - 最初のチャンネル以外を、全てドロップする。
  - **Stereo** - 全チャンネルを、FLとFRにミックスする。
  - **Stereo drop** - LとRに定義されたチャンネル以外の、全てのチャンネルをドロップする。



#### 注記

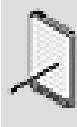
デフォルトで、チャンネル設定は、全てのプラットフォームでリンクしています。あるプラットフォームのチャンネル設定を別途、指定するには、そのプロパティをアンリンクしてから、設定を定義します。



#### 注記

モノファイルのチャンネル数を増やしたくない場合は、必ずAllow channel upmixオプションを無効にします。

7. ステレオのソースをモノに変換する場合、またはその逆の場合は、L-R Mix設定を使って、左右のチャンネルにアサインする信号の、パワーレベルを指定できます。



### 注記

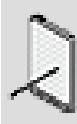
デフォルトで、L-R Mix設定は全てのプラットフォームでアンリンクしています。いくつかのプラットフォームに共通のL-R Mix設定を指定するには、そのプロパティをリンクさせてから設定を定義します。

8. Sample Rateリストで、コンバージョン中にオーディオファイルを1秒間にサンプルする回数を選択する。ゲームの具体的な状況に応じて、以下のオプションを1つ選択する。

- **As Input** - オリジナルファイルと同じサンプルレートを使い、ファイルをコンバージョンする設定。このサンプルレートが、特定のプラットフォームやオーディオフォーマットで採用できない場合は、採用できる最も近いサンプルレートが使われる。
- **Auto (Low/Medium/High)** - WwiseでファイルのFFT分析を行い、選択されたサンプルレートを使い、ファイルをコンバージョンする設定。品質設定をLow/Medium/High（低・中・高）から選ぶことで、アルゴリズムに使用するカットオフ閾値を変えられる。Project Settingsダイアログボックスで、これらの品質設定の閾値を設定することで、品質のレベルを調整できる。Wwiseが行うサンプルレートの自動検知については「[サンプルレートの自動検知設定](#)」を参照。
- **300 to 48000** - 具体的なサンプルレートを使い、ファイルをコンバージョンする設定。サンプルレートの範囲はプラットフォームによって異なり、最大48,000Hzである。

9. Sample Rateの設定がAs InputまたはAutoの場合は、Min Sample RateとMax Sample Rateの設定を使ってコンバージョンサンプリングレートを制限する。

10. コンバージョンで使うオーディオフォーマットを次から1つ選択する:  
[AAC](#)、[ADPCM](#)、[ATRAC9](#)、[OPUS](#)、[PCM](#)、[Vorbis](#)、[>WMA](#)。

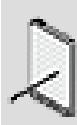


### 注記

AAC、ADPCM、Opus、Vorbis、XMAの各種オーディオフォーマットのエンコーディングパラメータを変更するには、Editボタンをクリックします。これらのエンコーディングパラメータの詳細説明を見るには、そのダイアログボックスの、Helpボタンをクリックします。オーディオフォーマットVorbisのパラメータ設定に関するベストプラクティスについては、レファレンスドキュメンテーションのVorbis Encoder Parametersのページを参照してください。

11. Sample rate conversion qualityリストで、ファイルのサンプルレートをコンバージョンする時に使う方式を選択する。以下のオプションのどちらかを選択する。

- **Normal (速い)** - 最良のオプションと比較して、3倍から6倍ほど早い、良品質のコンバージョンを提供。
- **High (遅い)** - 最良の品質のコンバージョンを提供する。



### 注記

コンテンツに高周波が含まれる予定で、24kHz以下のサンプルレートにコンバージョンする場合は、Highオプションを選択することを推奨します。

12.コンバージョンした全てのファイルの冒頭に、マーカーを設定するには、Insert Filename Markerリストで、Yesを選択する。

マーカーにはファイル名だけが含まれ、ファイルのパスや拡張子は、含まれません。ファイル名表示は、サウンドエンジンで再生中のサウンドにアクションを結びつけたい場合、例えば吹き替えや字幕作業の時などに、便利です。

13.コンバージョン処理中に、DCオフセットを除去しない場合は、チェックボックスRemove DC Offsetを外す。

デフォルトで、このオプションは選択されています。通常の状況では、DCオフセットの除去が推奨されます。ただし、DCオフセットを除去する必要がない場合もあり、その一例を、以下に示します。

- ・サンプルアキュレートであるコンテナに追加されるサウンド。
- ・0 dBにノーマライズされているサウンド。

DCオフセットが、Wwiseのオーディオ信号に与える影響については「[「DCオフセット除去」](#)」を参照してください。



#### 注意

オーディオソースから直接、モーションを生成している場合は、DCオフセットを除去すると、モーションアウトプットに影響するので、注意してください。

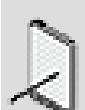
14.ビットレートのコンバージョンにディザリングを適用しない場合は、チェックボックスApply Ditherを外す。

ディザリングとは、量子化の前に信号に追加されるノイズのことで、量子化プロセスによって発生するディストーションやノイズのモジュレーションを削減するためのものである。ディザリングは、ビット深度が変わる時、例えば24ビットから16ビットに変わる時に限り、適用されます。

15. Conversion Settings Editorを閉じる。

指定した設定が、自動的に保存されるので、プロジェクト階層の1つ以上のオブジェクトに、このShareSetをアサインすることができる。

16.上記1~14を繰り返して、プロジェクトに必要なコンバージョン設定ShareSetを作成する。



#### 注記

Audio Sources表に中身を入れる前に、コンバージョン設定ShareSetを1つのオブジェクトにアサインして、この設定を使ってオーディオファイルをコンバージョンする必要があります。コンバージョン設定ShareSetをオブジェクトにアサインする方法については「[「コンバージョン設定ShareSetを、オブジェクトにアサインする」](#)」を参照してください。

## 関連トピック

- ・「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- ・「コンバージョン設定ShareSetを、オブジェクトにアサインする」
- ・「オーディオファイルをコンバージョンする」

- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「プラットフォームごとに、ソースを選択する」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「他のプラットフォームに、切り替える」

## DCオフセット除去

DCオフセットは、振幅0から信号までの距離を%で示したものです。通常のケースでは、サウンドに問題が起きないようにDCオフセットを除去すべきです。オフセットを維持したい場合は、Wwiseへの好ましくない影響があるので、注意してください。

- ノーマライズしたサウンドにDCオフセットがあると、オフセットがヘッドルームを消費するので、オーディオ処理中の最高ボリュームに到達できません。この問題はミックス全体に影響する可能性があり、DCオフセットのあるサウンドとないサウンドの2つをミックスすると、両方にDCオフセットが適用されてしまいます。
- リカーシブDSPアルゴリズムを適用した信号は全て、DCオフセットのオーディオファイルに非常に敏感に反応する可能性があり、ノイズが発生してしまいます。これは、特にリバーブプラグインや、Environmental FXプラグインで起きる可能性があります。



### 注意

ループサウンドのDCオフセットを除去することは推奨されません。Remove DC offsetはハイパスフィルタ機能であり、サンプルが継続再生されることを認識できないため、ループの最初と最後のサンプルを同様に処理するとは限りません。信号が中断され、クリックが聞こえてしまう可能性があります。

最適でないパフォーマンスによる問題を避けるために、DCオフセットは、ランタイムに除去しません。このため、Wwiseにファイルをインポートする前にDCオフセットを除去することが推奨され、DCオフセット除去フィルタ（多くのオーディオオーサリングツールで提供）を利用したり、Wwiseのコンバージョン処理中に行ったり（Conversion Settingsウィンドウにて）してください。

## コンバージョン設定ShareSetを、オブジェクトにアサインする

プロジェクトの様々なコンバージョン設定ShareSet（Conversion setting ShareSet）が作成できたら、プロジェクト階層にあるオブジェクトに、アサインしていきます。プロジェクト階層にある他のオブジェクトプロパティと同様に、コンバージョン設定ShareSetも、親から子に継承されるので、あるアクターミキサーにコンバージョン設定ShareSetをアサインすると、その下にある全てのコンテナやオブジェクトが、自動的に同じShareSetを使用します。もちろん、親オブジェクトの設定ShareSetをオーバーライドできるので、階層の特定のオブジェクトや、特定のソースに対して、別のコンバージョン設定を適用することもできます。

デフォルトで、コンバージョン設定インスタンスを作成すると、必ずShareSetとなります。ただし、特定のオブジェクト用に、ShareSetを調整することができます。他のオブジェクトと、そのコンバージョン設定をシェア（共有）したくない場合は、ShareSetのカスタムインスタンスを作成します。ShareSetのカスタムイン

スタンスを使用すると、設定が現在のオブジェクトだけに適用されます。このコンバージョン設定に変更を加えても、影響を受けるのは、この1つのオブジェクトだけです。

コンバージョン設定ShareSetを、オブジェクトにアサインするには：

1. Property Editorに、最上位オブジェクトを1つロードする。



2. Source Settingsタブに切り替える。
3. グループボックスConversion Settingsで、セレクタボタン(>>)をクリックする。

プロジェクトで現在、使用可能なコンバージョン設定ShareSetのリストが、表示される。

4. ロードしたオブジェクトに適用するShareSetを、リストから選択する。
5. このオブジェクトに、ShareSetのカスタムインスタンスを適用したい場合は、Modeリストで、Define customを選択する。

これで、コンバージョン設定インスタンスの名前に「Custom」という言葉が挿入されます。これで、このインスタンスを変更しても、影響を受けるのは、このインスタンスを使用する1つのオブジェクトだけである。

6. Editボタンをクリックして、このShareSetまたはカスタムインスタンスをConversion Settings Editorにロードして、コンバージョン設定の様々な詳細を、調整する。
7. Audio Sources表で、このコンバージョン設定が適用されるオーディオソースのリストを見て、間違いがないか確認する。

この表中の情報を完成させるには、オーディオソースをコンバージョンする必要があります。オーディオソースのコンバージョンについては「[オーディオファイルをコンバージョンする](#)」を参照。

## 関連トピック

- 「[オーディオ用の、コンバージョン設定ShareSetを作成する](#)」
- 「[オーディオファイルをコンバージョンする](#)」
- 「[プラットフォームごとに、オブジェクトプロパティをカスタマイズする](#)」
- 「[プラットフォームごとに、ソースを選択する](#)」
- 「[1つのプラットフォームで、プロジェクトのエレメントを除外する](#)」
- 「[他のプラットフォームに、切り替える](#)」

## オーディオファイルをコンバージョンする

コンバージョン設定ShareSetを作成して、これらをプロジェクトの様々なオブジェクトに適用できたら、次に、オーディオファイルのコンバージョンを行います。コンバージョン処理は、階層中の1つのオブジェクトの各種ファイルだけを対

象とすることも、コンバージョンをまだ行っていない全てのファイルを対象とすることも、できます。さらに、コンバージョン対象を、以下の範囲で指定することもできます。

- **Platforms** - 現在のプラットフォーム、またはプロジェクトの全てのプラットフォーム。
- **Languages** - 現在のランゲージ、またはプロジェクト用に作成した全てのランゲージ。コンバージョンされるのは、サウンドボイスファイルのみ。
- **Sources** - サウンドの、Contents Editorで、In Use (使用中) となっているバージョンのみ、または全てのバージョン (つまりソース) のサウンド。

オーディオファイルのコンバージョン処理には時間がかかり、プロジェクトのアセットが何千にもものぼると、特に長くなります。処理を早めるために、Wwiseは自動的に、マシンの全てのCPUコアを使用します。



#### 注記

オーディオソースのコンバージョンで問題が発生した場合は、ファイル名が長すぎることも考えられます。ファイル名の長さ制限については [「Wwiseのファイルの長さ制限」](#) を参照してください。

### オブジェクトのオーディオファイルをコンバージョンするには:

1. Audio File Conversionダイアログボックスを開くために、以下のいずれかを行う。
  - Project Explorer、またはContents Editorの、Audioタブで、コンバージョンするオブジェクトを1つ、右クリックして、メニューで、Convertを選択する。
  - オブジェクトまたはオーディオソースを1つ選択して、メニューバーで、Edit > Convertをクリックする。
  - SoundBank Editorの、Editタブで、1つ以上のオブジェクトを選択して、右クリックして、メニューで、Convertを選択する。
  - Conversion Settings Editorで、Audio Sources表のオーディオソースを1つ以上選択して、Convertをクリックする。

Audio File Conversionダイアログボックスが開く。

2. グループボックスPlatformsで、リストのプラットフォームを1つ以上選択する。
3. グループボックスLanguages で、以下のオプションを1つ選択する。
  - Current language
  - All languages
4. グループボックスSources で、以下のオプションを1つ選択する。
  - In Use versions (使用中バージョン)
  - All versions (全バージョン)
5. OKをクリックする。

Conversion - In Progressダイアログボックスが開く。最初のプログレスバーに、オーディオソースのコンバージョン処理の全体的な進行状況が表示され

る。さらに、1つ以上の補助プログレスバーに、ファイルごとのコンバージョン処理の進行状況が表示される。補助プログレスバーの数は、マシンのCPUコア数によって決まる。

ファイルコンバージョンが完了すると、ダイアログボックスが閉じる。



#### 注記

Wwiseで、オーディオソースのコンバージョン中に問題が発生すると、ダイアログボックスの最後にあるコンバージョンログに、その情報が表示されます。コンバージョン処理中であっても、Stopをクリックして、エラーメッセージの内容を確認できます。

- Conversion Settings Editorで、Audio Sources表の情報を見て、コンバージョン後の全てのソースに関して、チャンネル数が正しいか、フォーマットが正しいか、適格なサンプルレートが使われているか、妥当なサイズか、などを確認できる。



#### 注記

他のプラットフォームにおけるコンバージョン処理結果を確認するには、Wwiseのツールバーの、Platform Selectorリストで、そのプラットフォームを選択します。

Audioタブのオーディオファイルを、全てコンバージョンするには：

- Projectメニューで、<Convert All Audio Filesを選択する。

Audio File Conversionダイアログボックスが開く。

- グループボックスPlatformsで、定義されたプラットフォームを1つ以上選択する。
- グループボックスLanguagesで、以下のオプションを1つ選択する。
  - Current language
  - All languages
- グループボックスSourcesで、以下のオプションを1つ選択する。
  - In Use versions (使用中バージョン)
  - All versions (全バージョン)
- OKをクリックする。

Conversion - In Progressダイアログボックスが開く。最初のプログレスバーに、オーディオソースのコンバージョン処理の全体的な進行状況が表示される。さらに、1つ以上の補助プログレスバーに、ファイルごとのコンバージョン処理の進行状況が表示される。補助プログレスバーの数は、マシンのCPUコア数によって決まる。

ファイルコンバージョンが完了すると、ダイアログボックスが閉じる。



#### 注記

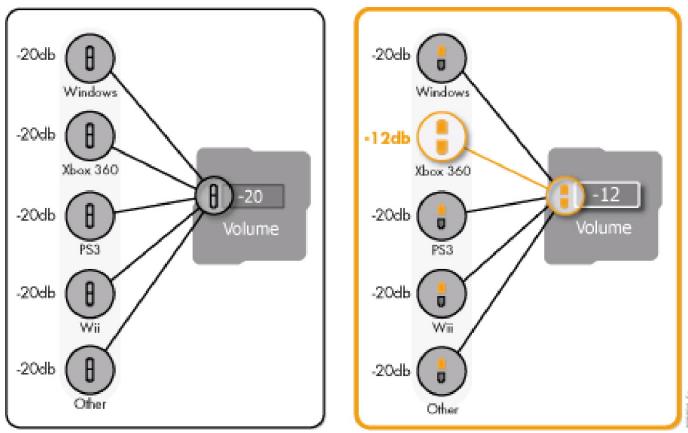
Wwiseで、オーディオソースのコンバージョン中に問題が発生すると、ダイアログボックスの最後にあるコンバージョンログに、その情報が表示されます。コンバージョン処理中であっても、Stopをクリックして、エラーメッセージの内容を確認できます。

## 関連トピック

- 「オーディオ用の、コンバージョン設定ShareSetを作成する」
- 「コンバージョン設定ShareSetを、オブジェクトにアサインする」
- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「プラットフォームごとに、ソースを選択する」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「他のプラットフォームに、切り替える」

### プラットフォームごとに、オブジェクトプロパティをカスタマイズする

Wwiseで、あるプラットフォーム用にサウンドのプロパティを設定すると、これらのプロパティが、デフォルトで、他のアクティブプラットフォームで同じく設定されます。この時、プロパティが全てのプラットフォームで「リンクしている（Linked）」と言います。複数のプラットフォームを対象としたプロジェクトの作成の効率化につながります。しかし、特定のプラットフォームでカスタマイズしたいプロパティがある場合は、そのプロパティをアンリンクして（Unlinked）、新たに設定することができます。プロパティをアンリンクすると、そのリンクインジケータがオレンジ色に変わります。



Properties are linked across all active platforms.

Volume is unlinked and customized for "XBox 360" platform, and the remaining platforms are partially unlinked.

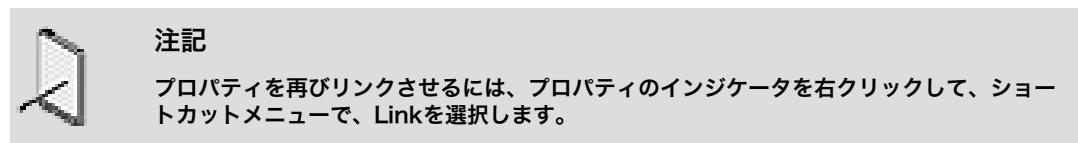
あるプラットフォームでプロパティを設定する時に、他のプラットフォームでプロパティがアンリンクされているかどうかを、簡単に確認できます。リンクインジケータが半分オレンジ色であれば、他でアンリンクされていることを示します。

### 1つのプラットフォームで、プロパティをアンリンクするには:

- アンリンクするプロパティの横の、リンクインジケータを右クリックする。  
ショートカットメニューが表示される。
- Unlinkをクリックする。

インジケータがオレンジ色に変わり、設定した値は現在のプラットフォームにだけ適用される。

同じプロパティがリンクされているプラットフォームに切り替えると、別のプラットフォームでこのプロパティがアンリンクされていることが、インジケータで表示される。

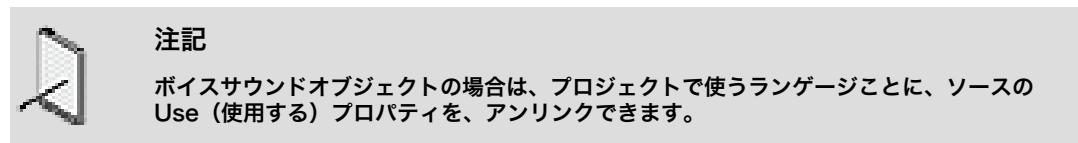
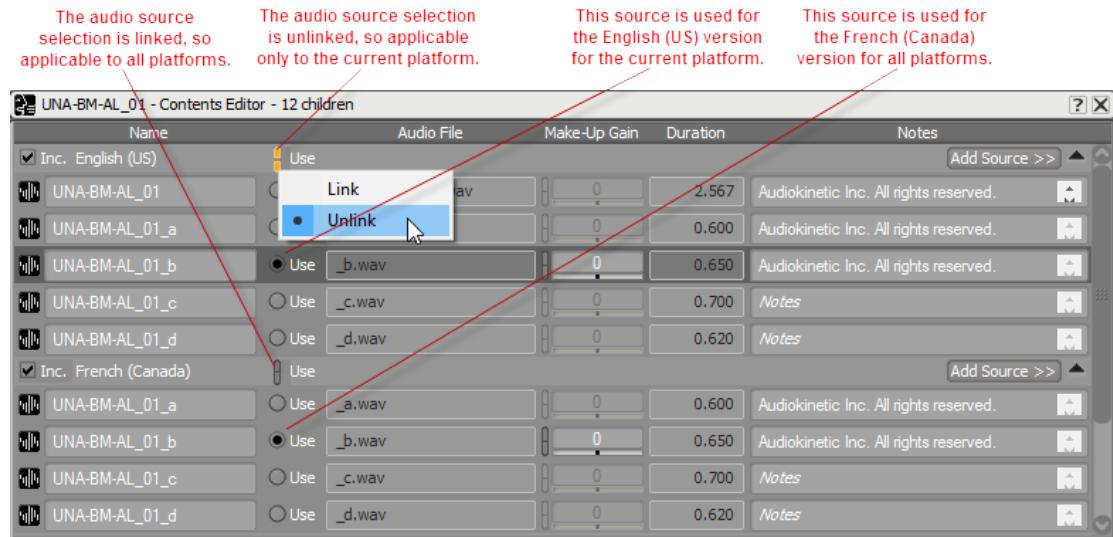


## 関連トピック

- 「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- 「オーディオ用の、コンバージョン設定ShareSetを作成する」
- 「オーディオファイルをコンバージョンする」
- 「プラットフォームごとに、ソースを選択する」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「他のプラットフォームに、切り替える」

## プラットフォームごとに、ソースを選択する

Wwiseのオブジェクトには、複数のソースが設定されていることがあります。デフォルトで、どのプラットフォームでも、オブジェクトを再生すると、同じソースが使われます。しかし、Contents Editorでソースをアンリンク (Unlink) すると、プラットフォームごとに、異なるソースを指定できます。設定したソースは、そのプラットフォームのSoundBankを生成する時に使用されます。



例えば、ファイティングゲームのキャラクター名を、プレイするプラットフォームによって変えるとします。ゲームの本来のヒーロー名は「Max」です

が、Windowsでは、「Arthur」というキャラクター名に変えます。つまり、一部のボイスサウンドオブジェクトにオーディオソースが2種類含まれ、両者はほとんど同じ台詞となります。台詞内の名前だけが、変わります。この場合、Windowsのバージョンだけで、サウンドオブジェクトのUseプロパティをアンリンクして、このバージョンでは、「Max」の台詞ではなく「Arthur」の台詞を使うように設定します。

プラットフォームごとに、どのソースを使うのかを設定する時に、ソースのステータスを、リンクインジケータで確認できます。現在のプラットフォームでアンリンクされているソースは、リンクインジケータが全体的にオレンジ色です。一方、他のプラットフォームでこのソースがアンリンクされている状態であれば、インジケータが半分オレンジ色です。

### あるプラットフォームで使うソースを選択するには:

1. Contents Editorで、使用するソースを1つ選び、そのソースの、Useオプションをクリックする。

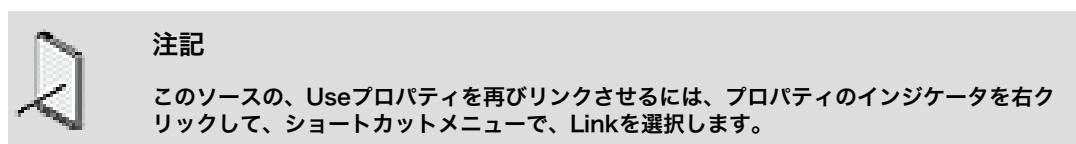
選択したソースが、全プラットフォームで使用される。

2. このソースを、現在のプラットフォームだけで使用する場合は、タイトルバーにある、Use linkインジケータを右クリックする。

ショートカットメニューが表示される。

3. Unlinkをクリックする。

インジケータがオレンジ色に変わり、設定したオーディオソースが、選定されたランゲージのプラットフォームの時だけに、使用される。



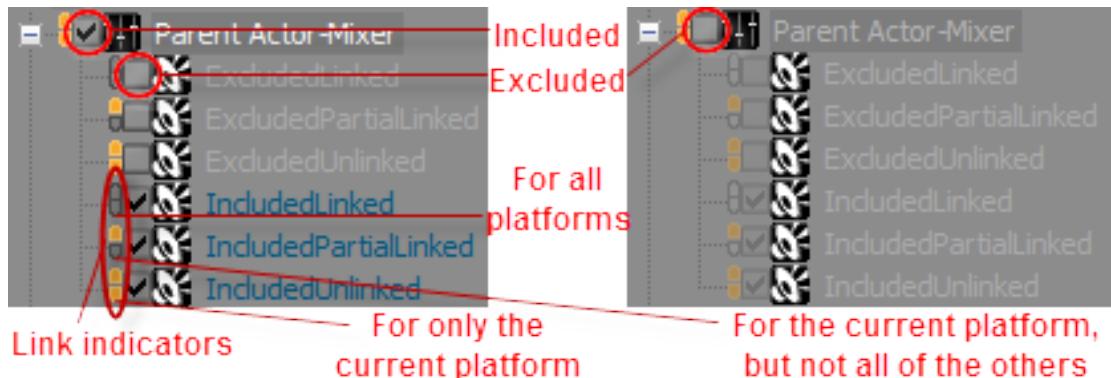
### 関連トピック

- 「あるプラットフォームの設定を、他のプラットフォームにコピーする」
- 「オーディオ用の、コンバージョン設定ShareSetを作成する」
- 「オーディオファイルをコンバージョンする」
- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「他のプラットフォームに、切り替える」

### 1つのプラットフォームで、プロジェクトのエレメントを除外する

オーディオの開発中は、ゲームやゲームプラットフォームのメモリ容量を気にしなければなりません。また、どのプラットフォームであれば、どのアセットを適格に活用できるかを、検討する必要があります。全てのプラットフォームを最適化するには、特定のオブジェクトを、プラットフォームによって含める（Included）・

除外する (Excluded) 状況もでてきます。あるオブジェクトを除外すると、このオブジェクトは、除外したプラットフォーム用に生成するSoundBankに、含まれません。



あるプラットフォームで、特定のオブジェクト、フォルダ、Event Action、ランゲージなどを含める (Include) ・除外する (Exclude) 設定は、以下の様な、Wwiseの様々なビューで表示されるプラットフォームチェックボックスで行います。

- Project Explorer (Effects, Audio Devices, the Virtual Folders, Work Unitsを含むオブジェクトの左にあるチェックボックス)
- Property Editor (アイテムアイコンの左にあるチェックボックス)
- Contents Editor (**Inc. Language** チェックボックス、ただしこの"Language"にプロジェクトのランゲージの1つが表示される)
- Event Editor (**Inclusion**)
- List View (**Inclusion**)
- Multi Editor (**Inclusion**)
- Query Editor (**Inclusion**)

どのオブジェクトを含める (Include) ・除外する (Exclude) のかを決めたら、そのプラットフォーム用に設定したオーディオを、Transport Controlで試聴できます。また、Soundcasterや、Game Profilerを使えば、あるプラットフォームでどのオブジェクトを含める (Include) ・除外する (Exclude) のかを決める参考になります。プロファイリングやシミュレーションの詳細は、以下のセクションを参照してください。

- [34章Profiling](#)
- [32章シミュレーションの作成](#)

Wwiseのビューで、現在のプラットフォームから、あるプロジェクトエレメントを除外するには:

1. チェックボックスが表示されるWwiseのビューのいずれかで、プラットフォームチェックボックスのチェックを外す、または、ショートカットメニュー **Inclusion > Exclude Selection** オプションを使う。

このプロジェクトエレメントが、現在のプラットフォームから除外 (Exclude) される。



**Tip**

複数のエレメントを一度の操作でエレメントを含める (Include) か除外する (Exclude) かを同時に選ぶことができます。

全プラットフォームで、プロジェクトエレメントを除外するには:

1. リンクインジケータがグレーであることを確認することで、除外するオブジェクトエレメントがリンクされていることを確認してください。
  - もし、リンクインジケータがオレンジの場合、**Inclusion > Link**を選んでください。
  - もし、上部でリンクインジケータがオレンジの場合、リンクをされていないプラットフォーム (全オレンジ) を探すために選択されたプラットフォームを**変更し**、すべてのリンクを変更してください。



**Tip**

のリンクインジケータについてをご覧ください。 「[プロパティ値のリンク・アンリンク](#)」。

2. Wwiseのビューで、現在のプラットフォームから、あるプロジェクトエレメントを除外するには: [list](#)と同様のステップに従ってください。

## 関連トピック

- 「[あるプラットフォームの設定を、他のプラットフォームにコピーする](#)」
- 「[オーディオ用の、コンバージョン設定ShareSetを作成する](#)」
- 「[オーディオファイルをコンバージョンする](#)」
- 「[プラットフォームごとに、オブジェクトプロパティをカスタマイズする](#)」
- 「[プラットフォームごとに、ソースを選択する](#)」
- 「[他のプラットフォームに、切り替える](#)」

## 他のプラットフォームに、切り替える

Wwiseプロジェクトには、アクティブなプラットフォームバージョンが全て入っているので、開発サイクル中にいつでも、簡単に他のプラットフォームに切り替えることができます。例えば、同時に複数のプラットフォームで開発している場合に、あるオブジェクトのプロパティを、プラットフォーム別に順番に設定したい場合などに、切り替えて設定できます。プラットフォームを切り替えると、現在のプラットフォームのバージョンが表示されます。

## 他のアクティブプラットフォームに切り替えるには:

1. ツールバーの矢印をクリックして、Platform Selectorリストを表示する。

2. Platform Selectorリストで、別のプラットフォームを選択する。

選択したプラットフォームバージョンが、表示される。

#### 注記

プロジェクトに追加したプラットフォームだけが、Platform Selectorリストに追加されます。プラットフォーム追加については「[プラットフォームの追加、削除、コピー](#)」を参照してください。

## 関連トピック

- 「オーディオ用の、コンバージョン設定ShareSetを作成する」
- 「オーディオファイルをコンバージョンする」
- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「プラットフォームごとに、ソースを選択する」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」

## あるプラットフォームの設定を、他のプラットフォームにコピーする

Wwiseのプロジェクトでは、プラットフォーム別の編集作業が可能なので、あるプラットフォーム向けの設定を、他のプラットフォームにコピーしたい状況があるかもしれません。例えば、2種類のプラットフォーム向けの作業をしばらく続けて、第3のプラットフォームを追加する準備ができた時点で、既にカスタマイズしたプラットフォームの1つをコピーして、そこからスタートできます。

ダイアログボックスCopy Platform Settingsは Platform Managerでアクセスでき、プラットフォームをコピーできます。ソースとターゲットのプラットフォームを指定すれば、プラットフォーム用の全ての設定がソースプラットフォームからターゲットプラットフォームにコピーされます。

#### 注記

プラットフォーム設定はベースプラットフォームと異なります。一旦後者をプラットフォームに選択すると、それは変わりません。新しいベースプラットフォームが必要になった場合は、新しいプラットフォームを作成する必要があります。

## ソースからターゲットにコピーされるプラットフォーム用の設定：

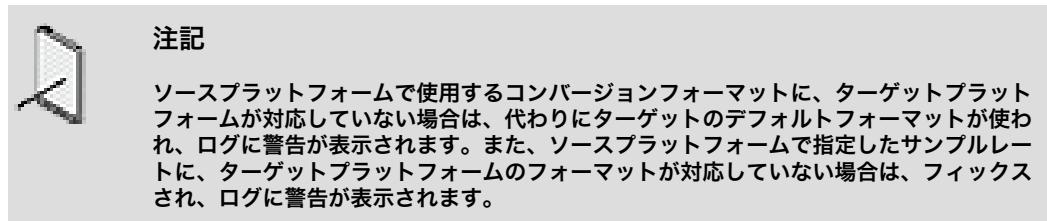
- アンリンクされたオブジェクトパラメータ

#### 注記

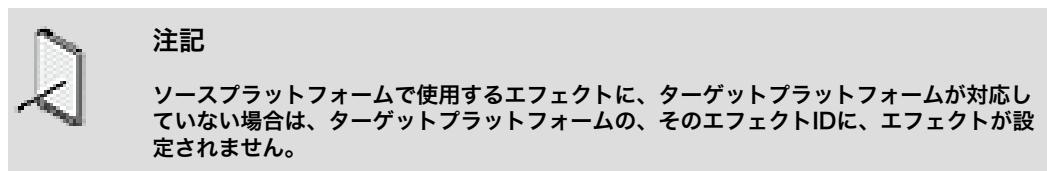
例外として、ダイアログProject Settingsの、SoundBanksタブと、External Sourcesタブにある、そのプラットフォーム専用の設定は、ターゲットプラットフォームにコピーされません。

- Inclusion/exclusion
- アンリンクされた、アクティブソース（サウンドの、Contents Editorで指定されたもの）

- コンバージョン設定



- アンリンクされた、3D減衰 (Property EditorのPositioningタブで指定されたもの)
- アンリンクされた、3D減衰カーブ (Attenuation Editorで設定されたもの)
- アンリンクされた、エフェクト (Property EditorのEffectsタブで指定されたもの)



- アンリンクされた、ダイアログイベントの引数パスの、ターゲットオブジェクト

ソースプラットフォームでアンリンクされている設定は、ターゲットプラットフォームに、アンリンクされた状態でコピーされます。ソースプラットフォームでリンクされた設定が、ターゲットプラットフォーム側でアンリンクされている場合は、ターゲットプラットフォームで再度、リンクされます。

## 関連トピック

- 「概要」
- 「プロジェクトのボリューム閾値の設定」
- 「プラットフォームごとに、オブジェクトプロパティをカスタマイズする」
- 「1つのプラットフォームで、プロジェクトのエレメントを除外する」
- 「プラットフォームごとに、ソースを選択する」
- 「オーディオ用の、コンバージョン設定ShareSetを作成する」
- 「Attenuation ShareSetを、オブジェクトに適用」
- 「オブジェクトプロパティの減衰カーブ」
- 「エフェクトの使用」
- 「パスにオブジェクトをアサインする」

## プロジェクトのローカリゼーション作業

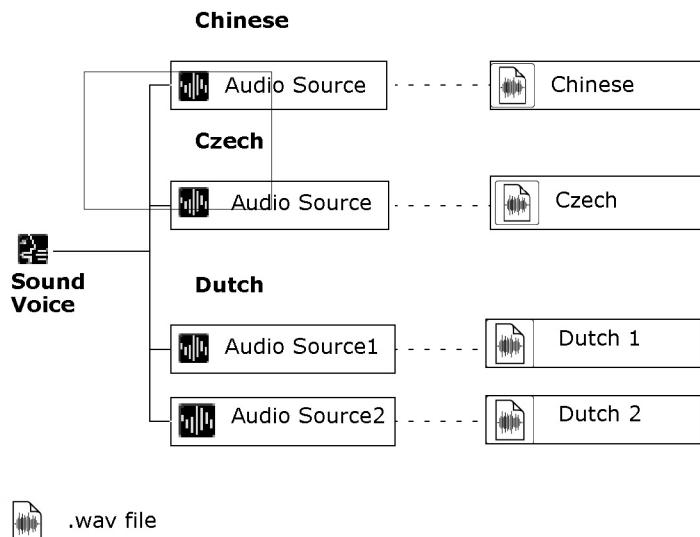
優れたローカリゼーションプロジェクトは、サウンドボイス (Sound Voice) オブジェクトを、翻訳されたバージョンに単純に置き換えるだけではありません。ローカライズされたバージョンは、元の言語を使って開発した時と同じオーディオエクスペリエンスを再現しなければなりません。具体的には、各ランゲージバージョン

のプロパティを調整してシンクすることで、ローカライズされたプロジェクトを、オリジナルバージョンと同じ品質レベルにします。ゲームのリリーススケジュールによって、ローカリゼーション作業を開発と併行して行う場合や、ゲームが完成した後に行う場合があります。Wwiseを使えば、開発サイクルのどの時点でも、ローカリゼーションを行えます。Wwiseでは、1つのプロジェクト内で、複数のプラットフォーム用に同時にオーサリングできるのと同様に、1つのプロジェクトで複数のランゲージバージョンを作成して、それぞれのランゲージバージョンの最適化や編集が行えます。

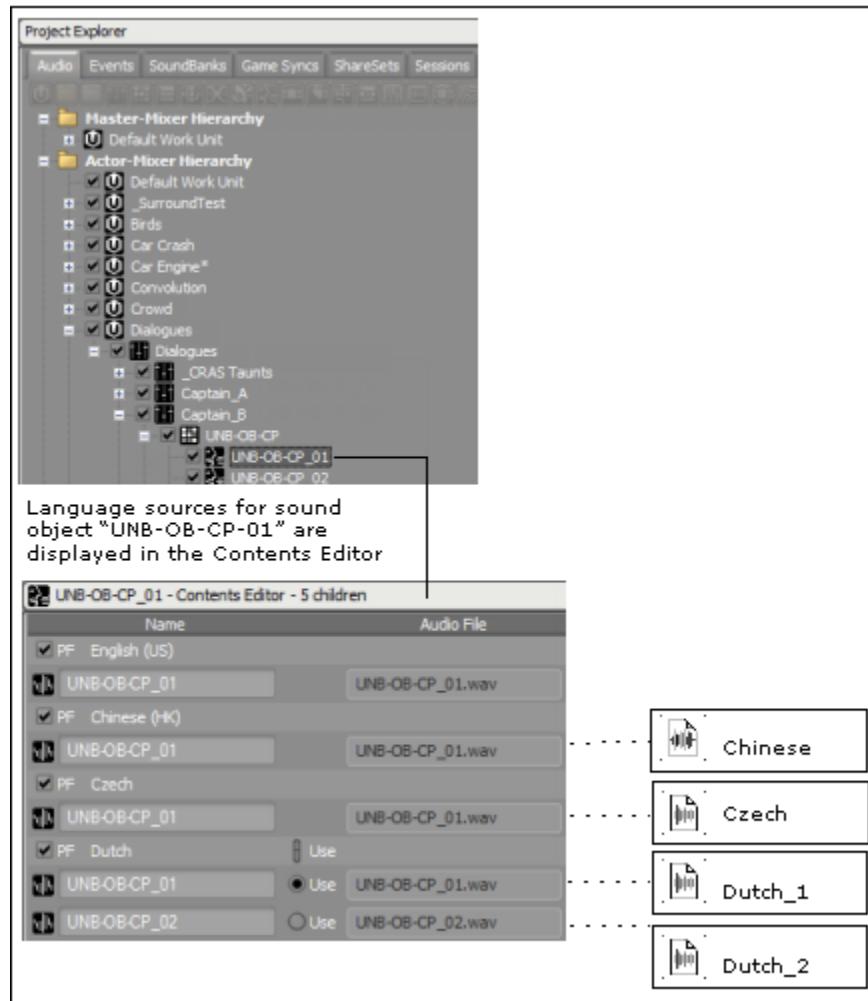
Wwise上で行うプロジェクトのローカリゼーション作業には、以下のタスクが伴います。

- ・プロジェクトで使うランゲージを全て、作成する。
- ・レファレンスランゲージとして、1言語を指定する。
- ・ランゲージファイルをインポートする。
- ・それぞれのランゲージバージョンを、試聴してプロファイリングする。

Wwiseでは、ランゲージごとのバージョンが、サウンドボイス (Sound Voice) オブジェクトのソースとして保存され、それぞれのソースが、異なるオーディオファイルと結び付いています。サウンドオブジェクトと、そのランゲージソースや、ランゲージファイルとの関係を、下図に示します。



サウンドボイス (Sound Voice) オブジェクトの中に、ランゲージごとのソースが含まれますが、表示されるのは、Contents Editor画面においてだけです。

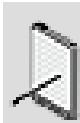
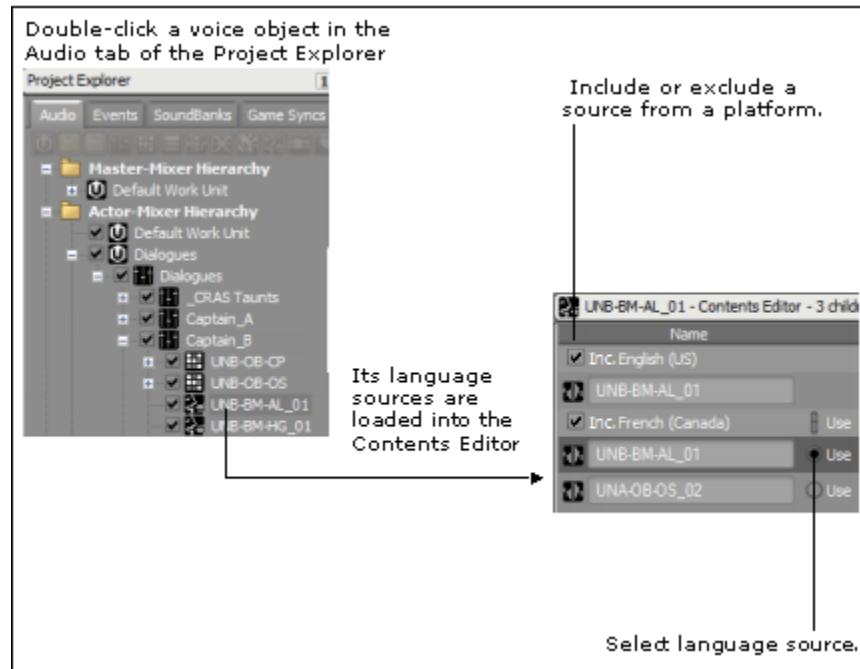


## ランゲージファイルをインポートする

ランゲージファイルの準備ができたら、Audio File Importerを使ってプロジェクトにインポートします。ファイルをインポートすると、Project Explorer の、Audioタブの、選択したボイスオブジェクトの中に、ランゲージソースが作成され、Contents Editorに表示されます。プロジェクトのローカリゼーションをまとめて行うには、階層の最上位でローカライズされたファイルをインポートします。

Contents Editorを使って、プラットフォームごとにランゲージソースの含める (Include)・含めない (Exclude) を設定したり、プラットフォームごとにランゲージソースを選択したりできます。ランゲージごと、またはソースバージョンごとにカスタマイズする方法や、プラットフォームごとで使うランゲージソースを決める方法については、以下を参照してください。

- 「[プラットフォームごとに、ソースを選択する](#)」
- 「[1つのプラットフォームで、プロジェクトのエレメントを除外する](#)」。



### 注記

プロジェクトのローカリゼーション用にオーディオファイルをインポートする時に、インポートするファイル名が、レファレンスランゲージのオーディオファイルと全く同じであることを、必ず確認してください。

プロジェクトに、ランゲージオーディオファイルをインポートするには:

- これからインポートするランゲージファイルの保存先となるオブジェクトを1つ、アクターミキサー階層で選択する。



### Tip

簡単にプロジェクト全体のローカリゼーションを行うには、アクターミキサー階層の最上位の親を選択して、このレベルでランゲージファイルをインポートします。

- Wwiseのメニューで、Project > Import Audio Files...をクリックする（やShift + Iをクリック）。

Audio File Importerが開く。

- Import Modeとして、Localize languagesを選択する。

Import as Sound Voiceオプションが自動的に選択され、Destination languageメニューが使用可能になります。インポートする言語は、既存のプロジェクト言語でなければなりません。

- 次のいずれかのオプションをクリックします：

- ファイルの追加...

インポートするメディアファイル (AMB、MID、WAV) を選択できるFile Openダイアログボックスが開きます。

- フォルダを追加...

Folder Open ダイアログボックスでは、インポートするメディアファイルを含むフォルダを選択できます。

- Import Tab Delimited...

Open ダイアログボックスでは、インポートするオーディオファイルと作成する構造を定義するタブ区切りのテキストファイル (TXTまたはTSV) を選択できます。



5. インポートするファイルまたはフォルダの場所を参照します。

6. 選択が完了したら、**Open**をクリックします。

選択したファイルが、Audio File Importerにロードされる。

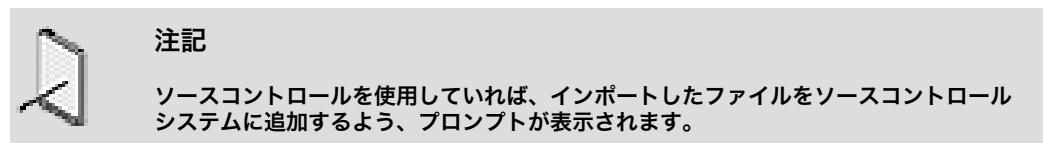
7. **Import**をクリックする。

Importingダイアログボックスが開き、ファイルインポートプロセスの進行を確認できる。



インポートが完了すると、Importingダイアログボックスが閉じ、Audio File Importerの画面に戻る。

8. オーディオファイルのインポート作業が終了したら、**Close**をクリックして、Audio File Importerを閉じる。



### 他のランゲージバージョンに、切り替える

Wwiseプロジェクトには採用したランゲージが全て入っているので、開発サイクル中にいつでも、簡単に他のランゲージに切り替えることができます。[Language Manager](#)で設定した全てのランゲージが、Language Selectorリストに表示されます。

## 他のランゲージに切り替えるには

1. ツールバーの矢印をクリックして、Language Selectorリストを表示する。
2. Language Selectorリストで、別のランゲージを選択する。

プロジェクトのローカリゼーション後のバージョンが、表示されます。

## ランゲージバージョンの試聴と、プロファイリング

Wwiseでは、ランゲージバージョンごとに、試聴、シミュレーション、プロファイリングなどを、通常の方法で行えます。その作業結果に基づいて、プラットフォームごとに、ランゲージソースのどのバージョンを採用するのか、またどのランゲージソースを採用するのかを選択して、最も効果的で効率的な成果を得ることができます。

各ランゲージバージョンの試聴、シミュレーション、プロファイリングについては、以下のセクションを参照してください。

- [41章Transport Control画面について](#)
- [「Performance Monitorを使った、モニタリングとトラブルシューティング」](#)
- [32章シミュレーションの作成](#)

## バージョンに関する、コツとベストプラクティス

以下のセクションに目を通して、オーディオ開発プロセス全体を通してプラットフォームバージョンやランゲージバージョンを効率的に管理するためのコツやベストプラクティスを参照してください。

### 複数ファイルのコンバージョン

階層の高位オブジェクト、例えばコンテナなどを、右クリックして、ショートカットメニューで、Convertを選択すると、一度に複数のオーディオファイルのコンバージョンができます。オブジェクトグループ全体のコンバージョン設定を行うには、最上位オブジェクトを右クリックして、ショートカットメニューで、Conversion Settingsを選択します。また、Project Explorerで複数のオブジェクトを選択して、それらのコンバージョン設定の一括定義や、一括コンバージョンを行うことも可能です。

### コンバージョン設定の戦略

多くの場合、ゲーム開発スタジオでは、各プラットフォームを充分に分析して、メモリフットプリント、CPU、帯域幅などのアイテムごとに、「バジェット（配分）」を割り当てます。オーディオチームにも、このバジェットの一部が与えられ、プラットフォームごとにバジェットを効率的に管理するのは、オーディオチームの責任となります。プラットフォームによって、バジェットの割り当てが違うので、オーディオデザイナーは、コンバージョンの様々な設定項目を使いこなして、プラットフォームによって差があるバジェットを効率的に補いながらも、最高のサウンドを提供しなければなりません。Game Profilerを使ってゲーム中のパフォーマンスをモニターして、現状を確認しながら、コンバージョン設定を調節して、バジェット内に納めてください。サウンドデザイナーのバジェットへの影響を、設定ごとに、次のセクション以降で説明します。

## オーディオフォーマット

バジェットの検討事項として、オーディオフォーマットを考慮する必要があります。プロジェクトに最良のフォーマットを選択するには、Wwiseが対応するフォーマットの、それぞれの特徴を確認してください。一般的に、ファイルの圧縮率が小さいほど、CPU負荷が減りますが、通常圧縮されていないファイルは、より多くのメモリースペースやディスクスペースを使います。

オーディオファイルのエンコーディング方法を決めるための、基本的な提案と情報を、下表に示します。

ファイル フォーマット	圧縮比率 (約)	CPU負荷	Memory	一般的な使用例
PCM	1:1	小	大	高い忠実度が要求されるサウンド
ADPCM	4:1	中	中	アンビエントサウンド、SFX
Vorbis	3-40:1	大	中から極小	ダイアログ、ミュージック、アンビエントサウンド、SFX

プラットフォームとランゲージ  
のバージョン管理

ファイル フォーマッ ト	圧縮比率 (約)	CPU負荷	Memory	一般的な使用例	制約事項
					小さいサウンド(10ミリ秒未満では使用しないシクにはシクテブルが必要
XMA	2-20:1	小	中から極小	正確なループ再生が不要な、アンビエントサウンドやSFX	メタデータのオーバヘッドが大ストリミングサウンドのループはフイ

## プラットフォームとランゲージのバージョン管理

ファイルフォーマット	圧縮比率(約)	CPU負荷	Memory	一般的な使用例	制約事項
					ループ全体のループに限定されループポイントで無音の空き時間が発生することがあるメモリ上のサウンドはサンプルバリ12でのループに

プラットフォームとランゲージ  
のバージョン管理

ファイル フォーマッ ト	圧縮比率 (約)	CPU負荷	Memory	一般的な使用例	制約事項 対応サンプル アキレト再生には不向き メタデータのオバヘドが非常に大きいセトアブ時間が長いサンブルアキレト再生には不向き メタデ
AAC	3-23:1	大 (iOSでハードウェア がアシストするデコーダ を使用する場合は、小)	中から小	バックグラウンド (インタラクティブで ない) ミュージック	
Opus	10-60:1	極小	中から極小	ダイアログ、ミュージック、アンビエン トサウンド、SFX	

## プラットフォームとランゲージ のバージョン管理

ファイル フォーマッ ト	圧縮比率 (約)	CPU負荷	Memory	一般的な使用例	制約事項
					タ の オ バ ヘ ド が 他 の フ マ ト よ り や や 大 き い の で 非 常 に 小 さ い サ ウ ンド ( 10 ミ リ 秒 未 満 で は 使 用 し な い

ファイル フォーマッ ト	圧縮比率 (約)	CPU負荷	Memory	一般的な使用例	制約 事項

### Vorbisエンコーダに、シークテーブルを使う

Vorbisエンコーダは、ディスクスペースを節約するために、デフォルトとしてシークテーブルを使用しません。ただし、以下の状況では、シークテーブルを使う必要があります。

- Vorbisエンコーダを、From Elapsed Timeに設定したバーチャルボイスと組み合わせて使う場合。
- Vorbisエンコーダを、インタラクティブミュージックファイル用に使い、特に、サウンドエンジンが、ファイルの冒頭以外の位置から読み始める必要がある場合。

- Vorbisエンコーダで、シークアクションを使う場合。

Vorbisでエンコードされたファイル用に、シークテーブルを有効にするには、以下を行います。

- **Conversion Settings**ダイアログで、Vorbisエンコーダを使用するプラットフォームのEditをクリックする。
- **Vorbis Encoder Parameters**ダイアログで、**Seek table granularity**リストの数字を選択してシークテーブルの精度を指定する。

シークテーブルの精度が細かいほど、メモリが必要となります。サンプルアキュレートなシーク（インタラクティブミュージックや、シークアクション）は、精度が低すぎるとCPU負荷が急増したり、ディスクのストリーミング精度よりも大きいとI/O使用が不足したりすることがあります。目安としてディスクのストリーミング精度以下のシークテーブルサイズ、つまり初期化で[AkDeviceSettings::uGranularity](#)で使う値を選択します。

---

## 第32章 シミュレーションの作成

概要 .....	603
シミュレーションを構築する .....	604
シミュレーション中の再生の管理 .....	607
ゲームシンクを使ったシミュレーション .....	610
シミュレーションを使った、プロパティの最終調整 .....	613
シミュレーション作成に関する、コツとベストプラクティス .....	616

## 概要

開発のプロセス中にいつでも、作成中のオブジェクトやイベントを使ってWwiseでシミュレーションを行えます。Wwiseのシミュレーション環境であるSoundcasterを使って、サウンド構造、ミュージック構造、モーション構造を、非同期で再生できます。つまり、何をいつ再生するのかをコントロールできます。イベントのテストやリアルタイムミキシングなどに非常に便利です。強力なツールであるSoundcasterは、以下の場面で活用できます。

- ・プロトタイプ制作、実験
- ・プルーフオブコンセプト（概念実証）の開発
- ・サウンド、音楽、モーションの同時試聴

Soundcasterを使ったシミュレーションは、Wwiseだけを使って行うことも、ゲームにリモート接続した状態で行うこともできるので、様々な使い方が可能です。シミュレーション機能として、以下を実行できます。

- ・特定プラットフォーム向けのオーディオやモーションの試聴
- ・コンバージョン前のオーディオファイルの試聴
- ・オーディオやモーションの再生中の、プロファイリング
- ・マニュアル操作でゲーム中のアクションをシミュレーションしながら、オーディオやモーションのWwise上でのミキシングやテスト
- ・ゲーム中やWwiseで、オーディオやモーションのプロファイリング
- ・1つのゲームオブジェクトに関連する各種のサウンド、ミュージック、モーションオブジェクトを使った実験
- ・ゲーム中に、サウンド、ミュージック、モーションの、ミキシングやテスト

Soundcasterでシミュレーションを作成する前に、Soundcasterセッションという、シミュレーションでこれから使うWwiseのオブジェクトやイベントが入ったプリセットを作成する必要があります。Soundcasterレイアウトは、1つのセッションとして、Project Explorerの、Sessionsタブの、該当するWork Unitの下に保存されます。インターフェース上で分かりやすいように、Soundcasterセッションは、以下のアイコンで表されます。

アイコン	内容
	Soundcaster Session

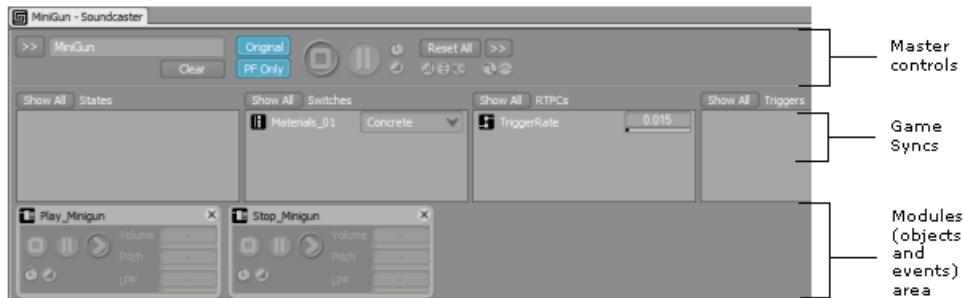
Soundcasterをアクセスするには、Project Explorerで、セッションをダブルクリックするか、メニューバーで、Views > Soundcasterをクリックします。メニューバーからSoundcasterを開くと、最後に開いたSoundcasterセッションが表示されます。



Soundcaster画面は、以下の3つに分かれています。

- ・マスターコントロール（Master controls）

- ・ゲームシンク (Game syncs)
- ・オブジェクト、イベント (Objects and events)



Soundcasterの3つの部分を使うことで、シミュレーションを構築する時に、ミキシング機能や再生機能を活用できます。

## シミュレーションを構築する

Soundcasterは非常に柔軟性が高く、どのような目的にも合わせて、シミュレーションを作成できます。シミュレーションに使うサウンド、ミュージック、モーションオブジェクトを自由に追加して、あとから何度も変更できます。シミュレーションの構築には、以下の作業が伴います:

- ・「Soundcasterセッションを作成する」
- ・「Soundcasterに、オブジェクトやイベントを追加する」



### Soundcasterセッションを作成する

Soundcasterを使い始める前に、Soundcasterセッション (Session) を作成する必要があります。Soundcasterセッションの作成は、Wwiseの以下の2つの場所で行います。

- ・Project Explorerの、Sessionsタブ
- ・Soundcaster画面の、マスターコントロールの部分

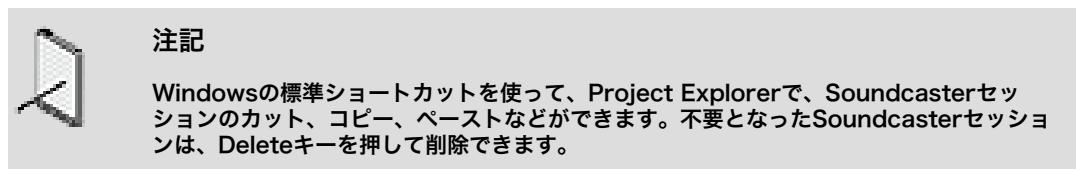
### Project Explorerで、Soundcasterセッションを作成するには:

1. Project Explorerで、Sessionsタブに切り替える。
2. Soundcaster Sessions表示の下で、以下のいずれかを行う。
  - ・ワークユニット（または仮想フォルダ）を1つ選択してから、Project Explorerのツールバーで、Soundcaster Sessionアイコンをクリックする。

- Work Unit（または仮想フォルダ）を右クリックして、ショートカットメニューで、New Child > Soundcaster Sessionをクリックする。

このWork Unit（または仮想フォルダ）に、新規Soundcasterセッションが1つ追加されます。

- デフォルトのSoundcasterセッション名を、内容が分かる名前に変更する。
- 新規Soundcasterセッションを、Soundcasterで開くには、このセッションをダブルクリックする。



### Soundcasterで、新規Soundcasterセッションを作成するには:

- Soundcasterで、Soundcaster Sessionのセレクタボタン(>>)を押してから、リストで、Newを選択する。  
New Soundcaster Sessionダイアログボックスが開く。
- これから作成する新規Soundcasterセッションを保存するWork Unitを、1つ選択する。
- Nameフィールドで、デフォルトのSoundcasterセッション名を、内容が分かる名前に変更する。
- OKをクリックする。

Soundcasterセッションが作成される。

### Soundcasterに、オブジェクトやイベントを追加する

Soundcasterにオブジェクトやイベントを1つ追加すると、モジュールが1つ、作成されます。モジュールとは、そのオブジェクトやイベントを表すもので、プロパティや動作が表示されます。また、いくつかのコントロール機能があり、プロパティの変更や、サウンド、ミュージック、モーションオブジェクトなどの再生ができます。



サウンド、ミュージック、モーションなどの組み立てが進み、それらを使ってミキシングやテストを行う準備ができたら、それぞれのモジュールを、Soundcasterに追加します。モジュールとして、Soundcasterに追加できるものを、以下に示します。

- Busses
- Actor-mixers
- Containers
- Sounds
- Music Containers
- Music Segments
- Events



#### 注記

Soundcasterセッションに、プロジェクトからアンロードされたオブジェクトやイベントに関するモジュールがある場合は、そのモジュールのコントロール機能は使えなくなり、モジュールのタイトルが黄色で表示されます。

### Soundcasterに、オブジェクトやイベントを追加するには:

#### 1. 以下のいずれかを行う:

- オブジェクトを1つ、Project Explorerの、Audioタブから、Soundcasterにドラッグする。
- イベントを1つ、Project Explorerの、Eventsタブから、Soundcasterにドラッグする。

オブジェクトまたはイベントをドラッグした位置に、青いモジュール挿入インジケータが表示される。マウスボタンを離すと、選択したオブジェクトまたはイベントが、Soundcasterのモジュールとして追加される。

### 関連トピック

- [「Soundcasterセッションを作成する」](#)
- [「Soundcasterのモジュールを削除する」](#)
- [「Soundcasterのモジュールを並べ替える」](#)
- [「Soundcasterモジュールの再生」](#)

### Soundcasterのモジュールを並べ替える

Soundcasterにオブジェクトやイベントをドラッグする時の順番は、あとから変えられるので、気にする必要はありません。モジュールは好きな順番で再生できますが、具体的なシーケンスを表すために、またはテストをしやすくするために、特定の順番に並べ替えることもできます。

### Soundcasterのモジュールを、並べ替えるには:

#### 1. 移動したいモジュールを、移動先の位置までドラッグする。

このモジュールが移動され、移動先に元々あったモジュールは、Soundcaster上で下に移動する。

### 関連トピック

- [「Soundcasterセッションを作成する」](#)

- ・「Soundcasterのモジュールを削除する」
- ・「Soundcasterに、オブジェクトやイベントを追加する」
- ・「Soundcasterモジュールの再生」

### Soundcasterのモジュールを削除する

モジュールが不要となった場合や、次のシミュレーションを作成したい時は、Soundcasterのモジュールを削除できます。

### Soundcasterのモジュールを、削除するには:

#### 1. 以下のいずれかを行う:

- ・モジュールを1つ削除するには、そのモジュールのタイトルバーの、Closeボタンをクリックする。
- ・全てのモジュールを削除するには、画面のマスターコントロール部分の、Clearをクリックする。

Soundcasterから、モジュールが削除される。

### 関連トピック

- ・「Soundcasterセッションを作成する」
- ・「Soundcasterに、オブジェクトやイベントを追加する」
- ・「Soundcasterのモジュールを並べ替える」
- ・「Soundcasterモジュールの再生」

## シミュレーション中の再生の管理

Soundcasterにモジュールを追加できたら、ミキシング機能や再生機能が使用できます。

シミュレーションを再生する時に、以下のタスクを実行できます。

- ・特定プラットフォーム専用の、オブジェクトやイベントの試聴
- ・コンバージョン前のオーディオファイルの試聴
- ・ゲームに接続中に、特定のゲームオブジェクトに関する各種オブジェクトやイベントの試聴
- ・好きな順番で、様々なモジュールの再生
- ・1つ、または全てのモジュールの、再生の管理
- ・ゲームシンクの実行
- ・モジュールの各種プロパティの調整
- ・再生中に、オブジェクトのEvent Actionの調整
- ・インタラクティブミュージックの試聴

### 再生するWwiseオブジェクトを指定する

シミュレーションを作成する時に、どのサウンドを再生させるのかを、以下について、具体的に指定できます。

- ・ **プラットフォーム (Platforms)** : シミュレーションするプラットフォームで、試聴するWwiseオブジェクトの指定。サウンドとプラットフォームの作業については「[1つのプラットフォームで、プロジェクトのエレメントを除外する](#)」を参照。シミュレーションを作成する時に、現在のプラットフォームに入っているサウンドのみを再生するのか、そのモジュールに入っている全てのサウンドを再生するのかを、選択できる。
- ・ **コンバージョン後のサウンド (Converted Sounds)** : コンバージョンしたソースと、オーディオファイルのオリジナルバージョンを比較する。インポートしたオーディオファイルのコンバージョンを行うと、そのオリジナルバージョンもWwise上で保存され、いつでも試聴できる。これらのファイルは、インポート用のコンバージョンを行っているが、プラットフォーム別のコンバージョンは行っていない。デフォルトで、Soundcasterはコンバージョン後のサウンドを再生するものの、元のインポートされたバージョンを再生することも可能。

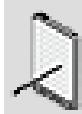
### 特定プラットフォーム用のサウンドやモーションオブジェクトを再生するには

1. シミュレーションするプラットフォームが、正しく選択されていることを、ツールバーで確認する。
2. Project Explorerの、Sessionsタブで、設定するSoundcasterセッションを、ダブルクリックする。

選択したセッションが、Soundcasterにロードされる。

3. マスターコントロール部分で、**Inc. Only**.

Inc. Onlyボタンが青くなり、現在のプラットフォームにあるオブジェクトとイベントだけが、試聴できるようになる。



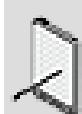
#### 注記

全てのオブジェクトやイベントを再生するには、Inc. Only をもう一度クリックして、このオプションを解除します。

### コンバージョン前のサウンドを、再生するには:

1. 画面のマスターコントロール部分で、**Original**をクリックする。

ボタンが青くなり、インポートしたサウンドをコンバージョンする前のバージョンが、Soundcasterの全てのモジュールにおいて再生される。



#### 注記

コンバージョン後のサウンドを再生するには、Originalボタンをもう一度クリックして、選択を解除してください。

### 関連トピック

- ・ [「Soundcasterを使って試聴する」](#)
- ・ [「ゲームシンクを使ったシミュレーション」](#)

- ・「シミュレーションを使った、プロパティの最終調整」
- ・「ローカルまたはリモートで、ゲーム機に接続する」

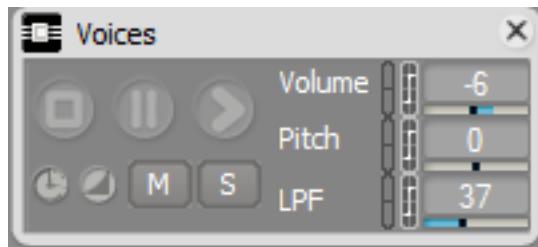
### Soundcasterを使って試聴する

再生させるサウンドやモーションオブジェクトなどを選択できたら、これらのモジュールを、Soundcasterでどのように試聴するのかを決めます。非同期再生（Asynchronous playback）機能を使えば、複数のモジュールの、試聴の順序、連続させるのか、オーバーラップさせるのか、などを自由に決められます。再生順序を変えてみることで、テストや実験ができるだけでなく、最も良い順番を決める手がかりにもなります。シミュレーションの目的に合わせて、モジュールを再生してください。再生に伴う基本的なタスクを、以下に示します。

- ・モジュールの再生
- ・モジュール再生の一時停止
- ・モジュール再生の停止

### Soundcasterモジュールの再生

モジュールを試聴するには、モジュールのコントロール機能を使って直接再生します。再生させたい全てのモジュールを、希望する順番で再生し始めた後に、自由にモジュールの一時停止や停止をできます。ただし、アクターミキサーとバスを再生することはできません。



マスターコントロールを使って、全てのモジュールを一斉に停止または一時停止することもできます。



### Soundcasterモジュールを、個別に再生するには:

1. モジュールの、Playをクリックする。

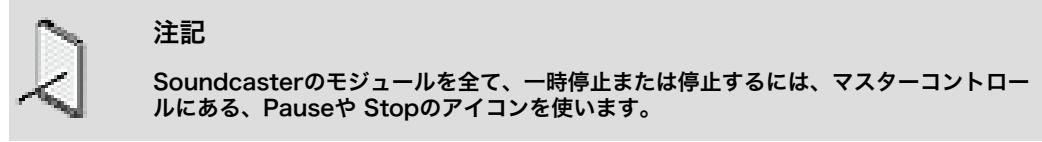
設定されているサウンドが、最後まで再生されます。

2. 再生を一時停止するには、Pauseをクリックする。

モジュール上と、マスターコントロール部分で、Pauseアイコンが黄色に変わることで、再生を再開するには、Pauseをもう一度クリックする。



3. 再生を停止するには、Stopをクリックする。



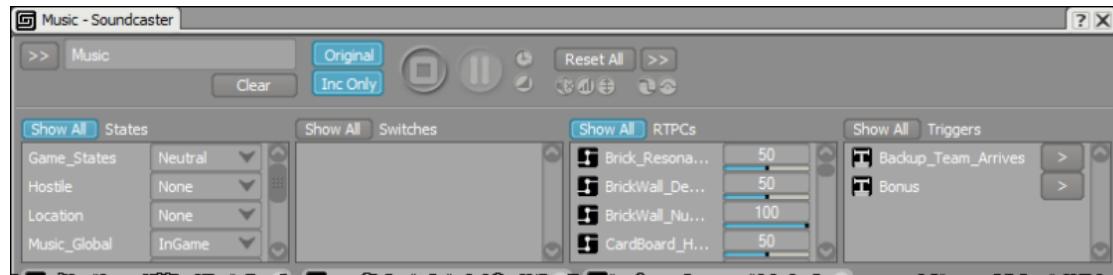
## 関連トピック

- 「再生するWwiseオブジェクトを指定する」
- 「再生中に、Stateを有効化する」
- 「再生中に、Switchをアサインする」
- 「Game Parameterの変化を、シミュレーションする」
- 「リアルタイムミキシングと、ポジショニング」
- 「リセット機能の利用」

## ゲームシンクを使ったシミュレーション

Soundcasterでは、プロジェクトで作成したゲームシンクもアクセスできます。サウンド、ミュージック、モーションオブジェクトに対して適用する、ステート、スイッチ、RTPC、トリガーなどを、設定した通りに、ゲーム上で、またはWwise上で、試聴できます。ゲームに接続して、Soundcasterでモジュールを適宜、追加すれば、ゲームアクションに合わせて、リアルタイムでゲームシンクを使いながら、試聴、テスト、ミキシングなどをできます。

Show All（全て表示）ボタンを選択すると、作成してあるゲームシンクが全て、Soundcasterのゲームシンク部分に表示されます。このボタンを選択していない場合は、Soundcasterにロードされている各種Wwiseオブジェクトに関連するゲームシンクだけが、表示されます。



### 再生中に、Stateを有効化する

様々なモジュールをSoundcasterに追加して、どのWwiseオブジェクトを再生するのかが決まれば、各種Stateを再生中に有効にできます。あるオブジェクト

を、Soundcasterにドラッグして入れると、モジュールの中のWwiseオブジェクトに適用されるState GroupやStateが、Statesリストに追加されます。Stateが表示されていなければ、Show Allをクリックしてください。全てのState GroupやStateが、States部分に表示されます。Stateの作成や、オブジェクトをStateにアサインする方法については、以下のセクションを参照してください。

- ・「Stateの活用」
- ・「オブジェクトやバスにStateをアサインする」

### 再生中にStateを有効にするには:

1. Stateリストで、適用するStateを1つ選択する。

Soundcasterの中のWwiseオブジェクトのうち、選択したStateに登録してある全てのオブジェクトに、このStateが適用される。



### 関連トピック

- ・「再生中に、Switchをアサインする」
- ・「Game Parameterの変化を、シミュレーションする」
- ・「シミュレーション中に、Triggerを呼び出す」

### 再生中に、Switchをアサインする

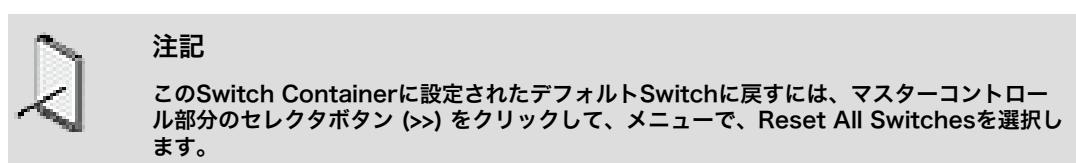
様々なモジュールをSoundcasterに追加して、どのWwiseオブジェクトを試聴するのかが決まれば、各種Switchを再生中に適用することができます。あるオブジェクトを、Soundcasterにドラッグして入れると、モジュールの中のWwiseオブジェクトに適用されるSwitch GroupやSwitchが、Switches部分に追加されます。Switchが表示されていなければ、Show Allをクリックしてください。全てのSwitch GroupやSwitchが、Switches部分に表示されます。Switchの作成と使い方については、以下のセクションを参照してください。

- ・「Switchの使い方」
- ・「Switch Containerの種類の設定」

### モジュールに、Switchをアサインするには:

1. Switchリストで、適用するSwitchを1つ選択する。

選択したSwitch Groupに登録された様々なSwitch Containerが、選択したSwitchに該当するWwiseオブジェクトを再生する。



## 関連トピック

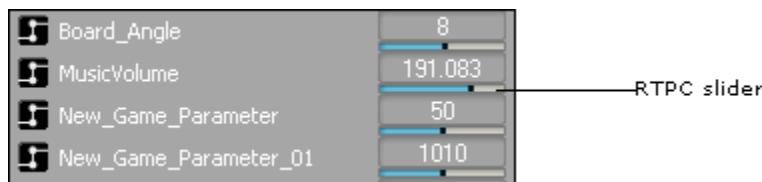
- 「再生中に、Stateを有効化する」
- 「Game Parameterの変化を、シミュレーションする」
- 「シミュレーション中に、Triggerを呼び出す」

## Game Parameterの変化を、シミュレーションする

様々なモジュールをSoundcasterに追加して、どのWwiseオブジェクトを試聴するのかが決まれば、Game Parameterにマッピングしたプロパティ値を、テストできます。Game Parameterを作成してプロパティ値をマッピングする方法については、以下のセクションを参照してください。

- 「RTPCで使う、Game Parameterの管理」
- 「Game Parameterに対して、Wwiseプロパティを設定する」

あるオブジェクトを、Soundcasterにドロップして入れると、それに関連するGame Parameterと、それらに伴うプロパティ値のスライダが、RTPC部分に追加されます。Game Parameterが表示されていなければ、Show Allをクリックしてください。作成した全てのGame Parameterが、RTPC部分に表示されます。スライダは、Game Parameter値の範囲を表します。Game Parameter値を既にWwiseのプロパティ値にマッピングしてあるので、パラメータ値を変えると、自動的にプロパティ値が変わります。

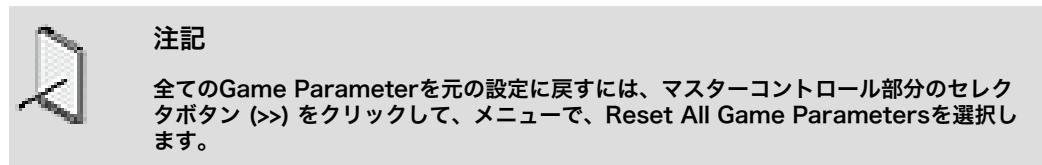


プロパティが変化する様子を、シミュレーションの再生中に試聴できます。

## 再生中にGame Parameter値を変更するには:

- 再生中に、RTPCスライダを使って、Game Parameter値を変更する。

Game Parameterと、それに関連するオブジェクトの各種プロパティの間のマッピング設定に基づいて、そのオブジェクトのプロパティ値が変化します。



## 関連トピック

- 「再生中に、Switchをアサインする」
- 「再生中に、Stateを有効化する」
- 「シミュレーション中に、Triggerを呼び出す」

## シミュレーション中に、Triggerを呼び出す

様々なモジュールをSoundcasterに追加して、どのWwiseオブジェクトを試聴するのかが決まれば、スティンガー (Stinger) と呼ばれるミュージックフレーズをミュージックに重ねて再生させるTriggerを、テストすることができます。ゲーム中の重要ポイントでTriggerがスティンガーを呼び出して、再生中のミュージックに重ねて再生する様子を、シミュレーションできます。Soundcasterで試聴するTriggerやTrigger用のスティンガーの作成については、以下のセクションを参照してください。

- ・[「Triggerの使い方」](#)
- ・[29章スティンガーの活用](#)

あるオブジェクトを、Soundcasterにドラッグして入れると、それに関連するTriggerが、Triggers部分に追加されます。Triggerの表示がなければ、Show Allをクリックしてください。作成してある全てのTriggerが、Triggers部分に表示されます。

### 再生中にTriggerを呼び出すには:

1. モジュールのミュージックオブジェクトの再生中に、Triggersリストで、試聴するTriggerを1つを選択する。



2. Call Triggerアイコンをクリックする。

該当するスティンガーが、再生中のミュージックオブジェクトに重ねて再生される。引き続きTriggerを選択して、該当するスティンガーを再生させることで、ゲーム中のミュージックをシミュレーションできます。

### 関連トピック

- ・[「Soundcasterを使って試聴する」](#)
- ・[「Triggerの使い方」](#)
- ・[29章スティンガーの活用](#)

## シミュレーションを使った、プロパティの最終調整

Soundcasterで様々なイベントやオブジェクトを試聴している時に、他のプロパティ値やミキシング設定を試すこともできます。様々なサウンドオブジェクト、ミュージックオブジェクト、モーションオブジェクトのそれぞれのプロパティ値を、Soundcasterで直接、変更できます。

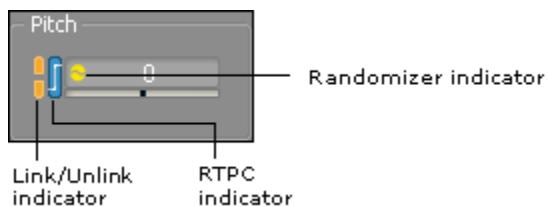


**注記**

イベントを使った作業の場合は、プロパティのアクションを変更するには、Event Managerを開く必要があります。プロパティやイベントを使った作業については「[イベントに、アクションを追加する](#)」を参照してください。

## プロパティのインジケータ

モジュールに表示されるプロパティ値に、1つ以上のインジケータ（表示）が付くことがあります。これらのインジケータは、そのプロパティ値が他のプラットフォームにリンクしているのか、そのプロパティ値がRTPCでGame Parameterに結び付けているのか、そのプロパティ値にRandomizerが適用されているのか、などを示します。Wwiseのインジケータの使い方については「[プロパティのインジケータ](#)」を参照してください。



プロパティ値のリンク・アンリンク、RTPCの使用、プロパティ値のランダム化などについては、以下のセクションを参照してください。

- 「[プラットフォームごとに、オブジェクトプロパティをカスタマイズする](#)」
- 「[Game Parameterを使って、プロパティ値をコントロールする](#)」
- 「[ランダム化したプロパティ値で、再生を強化](#)」

上記のプロパティインジケータ以外にも、Soundcasterには再生関連のインジケータがいくつかあり、再生中に、特定の動作やアクションが発生すると、青くなります。これらのインジケータは、マスター コントロール部分と、個別のモジュールに表示されます。Soundcasterにある、プロパティやアクションパラメータのその他のインジケータを、下表に示します。

アイコン	名前	Indicates
⌚	Delay	イベント、Random Container、またはSequence Containerに、ディレイが適用された。
fade	Fade	イベント、Random Container、またはSequence Containerに、フェードが適用された。
音量	Set Voice Volume	イベントの中の、サウンドオブジェクト、ミュージックオブジェクト、またはモーションオブジェクトに、ボリューム設定アクションが適用された。
音符	Set Pitch	Eventの中の、Sound SFXオブジェクト、ミュージックオブジェクト、またはモーションオブジェクトに、ピッチ設定アクションが適用された。
音量	Mute	Eventの中の、サウンドオブジェクト、ミュージックオブジェクト、またはモーションオブジェクトに、Mute Action(ミュートアクション)が適用された。
E	Set Low-Pass Filter	イベントの中の、サウンドオブジェクト、またはミュージックオブジェクトに、ローパスフィルタ設定アクションが適用された。
音量	Enable Bypass	イベントの中のオブジェクトに、バイパス許可アクションが適用された。

これらのオブジェクトプロパティの調整については、以下のセクションを参照してください。

- 「コンテナ内のオブジェクトを、全て再生」
- 「Event Actionのプロパティの設定」

### リアルタイムミキシングと、ポジショニング

シミュレーションの中で、各種プロパティ設定を使って、再生前や再生中に、以下を行えます。

- リアルタイムでミックスして、ボリューム、ピッチ、ローパスフィルタの、プロパティ値を変更できる。これらのプロパティのミキシングについては「[相対プロパティのボリューム、ピッチ、LPF、HPFの設定](#)」を参照。
- ポジショニングを調整して、2Dや3Dの、サウンド伝搬やモーション伝搬のプロパティを変更できる。ポジショニングについては[11章Positioningの定義](#)を参照。



#### Tip

プロパティの変更を、Property Editorや、Event Editorで直接行うには、モジュールのタイトルバーをダブルクリックします。

### リセット機能の利用

あるイベントのモジュールを、Soundcasterで試聴してから、このイベントに関連するオブジェクトのプロパティを変更したいと考えることもあるでしょう。オブジェクトのプロパティは、イベントのアクションによって一時的に変わることがあります。そこで、プロパティを変更する前に、プロパティをリセットして、元の数値に戻してください。イベントやEvent Actionについては「[イベントの使い方](#)」を参照してください。各モジュールのEvent Actionをリセットすることも、Soundcasterのマスターコントロール部分で全てのモジュールのEvent Actionをリセットすることもできます。

#### 注記

マスターコントロール部分で、Soundcaster上の全てのミュージックトラックの強制的な再生を、解除することもできます。

### マスターコントロール部分で、Event Actionをリセットするには:

- 画面のマスターコントロール部分で、Reset Allの隣のセレクタボタン (>) をクリックする。

- Resetメニューが開く。
2. Resetメニューで、以下のいずれかを選択する。
    - **Reset All**: 全オブジェクトを元の設定に戻す。
    - **Reset All Random and Sequence Containers**: オブジェクト用にトリガーした全てのランダムアクションやシーケンスアクションを解除する。
    - **Reset All Game Parameters**: オブジェクト用にトリガーした全てのGame Parameterを解除する。
    - **Reset All Set Mute**: オブジェクト用にトリガーした全てのミュートアクションを解除する。
    - **Reset All Set Pitch**: オブジェクト用にトリガーした全てのピッチアクションを解除する。
    - **Reset All Set Volume**: オブジェクト用にトリガーしたボリュームアクションを解除する。
    - **Reset All Set Bus Volume**: オブジェクト用にトリガーしたボリュームアクションを解除する。
    - **Reset All Set Low-pass Filter**: オブジェクト用にトリガーした全てのLPFアクションを解除する。
    - **Reset All Bypass Effect**: オブジェクト用にトリガーした全てのエフェクトバイパスアクションを解除する。
    - **Reset All States**: オブジェクトの全てのSet Stateアクションを解除する。
    - **Reset All Switches**: オブジェクトに設定した全てのSet Switchアクションを解除する。
    - **Reset All Music Tracks Force Usage**: Soundcasterを使った特定トラックの強制再生を解除する。
    - **Reset Position**: Attenuation Preview内のリスナーのポジションをデフォルトポジションに戻す。



#### 注記

マスター・コントロール部分にあるリセット機能には、他のコマンドもあります。Random ContainerやSequence Containerのリセットについては「[Soundcasterモジュールの再生](#)」を参照してください。State、Switch、Game Parameterのリセットについては「[ゲームシンクを使ったシミュレーション](#)」を参照してください。

## 関連トピック

- 「[ゲームシンクを使ったシミュレーション](#)」
- 「[シミュレーション中の再生の管理](#)」

## シミュレーション作成に関する、コツとベストプラクティス

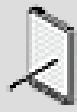
Wwiseのシミュレーション機能は、柔軟性に優れているので、Wwise上やゲーム上で効率的にプロファイリング、ブルーフォブコンセプト作成、テストなどが行えます。ゲームで最高の結果を達成するために参考となるコンセプトを、以下に紹介します。

## リアルタイムミキシングと、オブジェクトのプロパティ

ゲームやGame Simulatorに接続していれば、Wwise上でリアルタイムに、以下の相対プロパティ値を変更できます。

- Volume
- Pitch
- LPF

これらを変更するには、まず変更するプロパティの対象であるオブジェクトを、Transport Control、またはSoundcasterに、ロードする必要があります。オブジェクトがロードされていないと、そのオブジェクトがサウンドエンジンに登録されないため、変更を加えても有効になりません。アクターミキサーなど、Transport Controlでロードできないオブジェクトもありますが、アクターミキサーの子オブジェクトをロードすれば、その親オブジェクトもサウンドエンジンに登録されます。Soundcasterにはアクターミキサーをロードすることができるのでも、問題ありません。一旦オブジェクトを登録すると、ゲームとの接続が切れるまで、登録された状態が続きます。



### 注記

Transport Controlで、あるオブジェクトにピンを付けると、そのオブジェクトのピンを外すまで、他のオブジェクトをロードできないので、注意してください。ただし、Soundcasterにオブジェクトをロードした場合は、サウンドエンジンにそのオブジェクトが登録されます。

Randomizerや、エフェクト、減衰、ソースプラグインなど、一部のプロパティは、変更を加えても、そのオブジェクトが次にゲーム中に再生されるまで、加えた変更が反映されません。

---

## 第33章 Wwiseにおけるメモリ管理

概要 .....	619
Memory Managerの各コンポーネントについて .....	619
各種メモリプールのサイズを設定する .....	620
メモリ関連のトラブルシューティング .....	622
メモリプールの最適化 .....	623
メモリ管理に関する、コツとベストプラクティス .....	628

## 概要

Wwiseで作成する豊かで洗練されたゲームのサウンドスケープには、無限の可能性があります。ただしプラットフォームごとに使用できるメモリのリソースに限りがあることを、常に念頭におく必要があります。ゲームのオーディオをデザインする時に、このような制約を忘れないでください。プロジェクトのオーディオエメントの構築を進める際に、Wwiseが提供する様々なツール、例えば再生制限、バーチャルボイス、ボリューム閾値などを使えば、ゲームのメモリ使用を効率的に管理できます。

Profiler機能は、ゲームのメモリ問題のトラブルシューティングに役立つ数々の情報を提供します。Profilerが提供する情報を完全に理解して、Wwiseのメモリ節約ツールを充分に活用するには、Wwiseがどのようにして最終的なメディアファイル、サウンド構造、イベント、ゲームシンク、エフェクトなど、プロジェクトのあらゆる面を管理しているのかを理解する必要があります。

メモリマネージャの様々なコンポーネントの概要と、ゲームのメモリを効率的に管理するためのコツやベストプラクティスを本章で紹介します。Memory Managerのコンポーネントの詳細については、[Wwise SDK ドキュメンテーション](#)を参照してください。

## Memory Managerの各コンポーネントについて

Wwiseサウンドエンジンでは、Memory Managerがゲームのオーディオ専用メモリを管理します。Memory Managerは、ゲームのセットアップ時に、オーディオプログラマーによって初期化されます。プログラマーは、Memory Managerの初期化を行う時に、ゲームの様々なオーディオコンポーネントを管理するために必要な個数のメモリープールを、作成する必要があります。

Wwiseには、以下の3種類のメモリープールがあります:

- ・ **デフォルトメモリプール (Default memory pool)** - 上位のサウンド構造用の、一般使用のためのプール。このプールに、主に以下のものが含まれる。
  - ・ オーディオ構造のコンテンツ。
  - ・ 送出したイベントの、コマンドキュー。
  - ・ 登録されたゲームオブジェクト、ゲームオブジェクトのポジション、リスナー、RTPC、Switch、Stateなどのためのメモリアロケーション。
- ・ **ロワーエンジンメモリプール (Lower engine memory pool)** - 主要オーディオプロセスパイプラインのプール。このプールは一般的に、オーディオ再生、オーディオ処理、エフェクト用メモリなどに使う。
- ・ **メディアメモリプール (Media memory pool)** - ディスクにあるメディアファイルのコピーを格納する、汎用プール。

オーディオプログラマーは、ゲームの複数のメモリープールを作成する時に、それらのサイズも決定する必要があります。一旦プールを作成すると、サイズを変更できません。メモリープールは、Memory Managerの寿命中に作成され、破棄できます。プールは、均等なブロック (Block) に分けられ、このブロックが最小のアロケーション単位となります。

Memory Managerは、独自のテクノロジーやUnrealなど、他のテクノロジーでオーバーライドできます。カスタムメモリマネージャの実装については、Wwise SDKドキュメンテーションの、「[Memory Managerのオーバーライド](#)」を参照してください。

### ゲームで、メモリプールがどのように使われるのか

SoundBankをメモリにロードすると、非常に基礎的なレベルで、サウンドエンジンが以下の2つの機能を実行します。

- SoundBankのメタデータをパースして、サウンド階層とイベント情報を、デフォルトメモリプールに保存する。
- メディアファイルを、ユーザーが設定したメモリプールにコピーする。

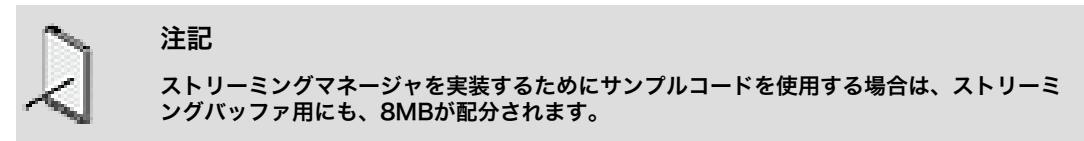
ゲーム中に、イベントによってオーディオがトリガーされた時に、ロワーエンジンメモリプールが、必要なオーディオ処理を行ってから、最終的なオーディオを再生します。

## 各種メモリプールのサイズを設定する

サウンドエンジンの内部メモリプールのデフォルトサイズは非常に大きく、これは、初めてサウンドエンジンを使う時に、ワーストケースシナリオでも稼働できるだけの充分なリソースを確保しておくためです。

実際には、各プールのデフォルトサイズは以下の通りです:

- 16デフォルトメモリプールは、16 MB
- 16ロワーエンジンメモリプールは、16 MB



デフォルトメモリプールのサイズは、最終的に作成するゲームの種類によって変わります。例えば、ゲームのオーディオアセットが合計数ギガバイトあっても、一度に4、5個以上を再生することが稀で、エフェクトも皆無または少数であれば、デフォルトメモリプールの必要メモリがより大きくても、ロワーエンジンメモリプールの必要メモリは小さいかもしれません。一方、ゲームのサウンド数は少なくとも、全て同時に再生して、リバーブやエコーなど多くの環境エフェクトを使う場合、例えばファーストパーソンシューティングゲームなどでは、デフォルトメモリプールの必要メモリがより小さくなり、ロワーエンジンメモリプールの必要メモリはより大きくなるかもしれません。

ロワーエンジンメモリプールのサイズは、最終的にサウンドエンジンで何をするのかによります。ゲームで多数のボイスを同時に再生して大量のエフェクトを処理する計画であれば、このプールの必要メモリが増えます。



PCで、バンクがロードされずサウンドが再生されていない状態では、メモリ消費量が約150kBになると予想されます。正確なメモリ消費を知るには、Wwiseの、Profilerレイアウトで、Performance Monitorの、Total Used Memory（メモリ消費合計）の数値を確認してください。当然、ロードされるアセット数、再生されるボイス数、ゲームオブジェクトのポジション数などが増えると、この値も増えます。

それぞれのメモリプールに必要なメモリを左右する条件として、オーディオ品質、ロードするサウンド数、同時に再生するボイス数、サウンド構造の複雑性、使用するエンコード方式、処理するエフェクトのタイプと数、3Dポジションが設定されているサウンド数、Switch数など、多数あります。

必要なメモリを確実に知る唯一の方法は、以下の通りです。

- 各プールに、それぞれ任意のメモリをアサインする。可能であれば、まずデフォルトのメモリサイズからスタートする。
- Wwiseをゲームに接続する。
- サウンドエンジンの、Profileバージョンを使い、ビジーな（込み入った）ゲームシナリオを稼働させる。
- Advanced Profilerの、Memoryタブを確認する。各プールに必要なサイズの目安が、Peak Used列で分かる。一般的に、それぞれのプールの最適サイズは、ピークのメモリ使用値よりも約15～20%ほど大きい値となる。
- Communications、Monitor、Monitor Queueの3種類のメモリプールのサイズを、合計から引く。これらのプールはデバッグ専用のメモリであり、リリースバージョンでは作成されない。これらのプールのサイズを除くと、リリースバージョンのメモリ使用量の目安がつく。



#### 注記

ゲームに接続せずに、サウンドエンジンのメモリ使用量を簡単にプロファイリングすることができます。キャプチャーセッションを開始して、Transport Control、またはSoundcasterで、イベントやオブジェクトを1つ以上、再生するだけです。これらのイベントやオブジェクトを再生する時のメモリ使用量が、Advanced Profilerの、Memoryタブに表示されます。

### テスト用のワーストケースシナリオを決める

メモリプールの種類によって、ワーストケースシナリオが異なることがあります。例えば、同時に64個のサウンドを再生すると、ロワーエンジンプール（Lower engine pool）のメモリ消費が急激に増加しますが、デフォルトメモリプールのメモリ消費はわずかに増えるだけです。一方、同時に20個のSoundBankをロードすると、デフォルトエンジンのメモリ消費が増加しますが、ロワーメモリプールのメモリ消費は影響を受けません。しがたって、ゲームには様々な状況があり、それぞれメモリの使い方が異なる可能性が高いので、全てのテストで同じワーストケースシナリオを使うのは推奨されません。

テストすべき条件の1つが、ゲームレベルが変わってSoundBankをロードしたりアンロードしたりする時で、これはデフォルトメモリプールのメモリ消費がピークに達する状況です。例えば、ゲームで、レベル1からレベル2に切り替わる時に、両方のSoundBankが瞬間的にメモリにロードされた状態となることがあれば、メモリ消費が急激に上がる可能性があります。



### 注意

Windowsプラットフォームでピーク時のメモリ消費をチェックするテストの前に、ゲームが稼働するPCのスピーカーのセットアップコンフィギュレーションが、5.1以上となっていることを確認してください。これは、スピーカー設定がステレオの場合に、サウンドエンジンが一定の最適化処理を行うため、5.1の時よりもメモリ消費が少ない可能性があるからです。

### どのサイズからスタートすべきか

多くの場合、デフォルトのメモリプールサイズ（1プールあたり16MB）からスタートするのが良いでしょう。これだけ大きいプールサイズであれば、実際のメモリ消費が簡単に分かり、プールごとに適宜、小さくできます。ただし、各種プールに割り当てることができるメモリ量が、これよりも制約されている場合（16MBが多すぎる場合）は、以下の数値からスタートしてみてください。

- ・デフォルトメモリプール: 2MB
- ・ローエンジンメモリプール: 2MB
- ・ストリーミングマネジメント: 2MB

ストリーミングのバッファサイズは、ストリーミングするサウンドの数、サウンド品質、ストリーミングサウンドの時間の余裕などによって変わります。8MBというデフォルト値は、ほとんどのゲームではメモリの無駄遣いと受け取られますが、前述の通り、大きくスタートして、メモリ使用量が明確になった時点で削減するのが、最適の方法です。

### メモリプールのサイズと、プラットフォーム

メモリプールのサイズは、プラットフォームとは独立しています。プラットフォームごとにプールサイズを設定する時に大事なのは、メモリプールのサイズは、プラットフォームの種類よりも、オーディオの内容と使用方法によって決まるということです。そこで、各種プールのサイズは、どのプラットフォームでも同じサイズからスタートして、Profilerのメモリ消費指標を参考にしながら、それぞれのプラットフォームに合わせます。

## メモリ関連のトラブルシューティング

各種メモリプールの初期サイズを設定してから、ゲームを稼働させて、問題がないかを確認します。ここでは、発生した問題を検知するために、Profilerが重要なツールとなります。

WwiseのProfilerで、ゲーム中にメモリアロケーションが失敗すると、必ず警告通知（Warning notification）が、キャプチャーログ（Capture log）に送られます。ゲームプレイ中のどのポイントで、どのメモリプールのメモリが不足していたかを、Warning Notificationリストで確認できます。

Profilerの使用については[34章Profiling](#)を参照してください。

### メモリ不足

メモリ不足の状況がいつ発生したかによって、サウンドエンジンの反応が変わります。様々なシナリオでメモリ不足が発生した時に何が起きるかを、下表に示します。

シナリオ	メモリが不足した場合
サウンドエンジンを初期化する時。	初期化が失敗する。
SoundBankをロードする時。	SoundBankのロードが失敗する
ボリュームなどのプロパティの、トランジションを開始する時。	トランジション処理をスキップして、パラメータは、トランジションなしで新しい値に直接移る。
サウンドを再生する時。	サウンドエンジンの初期化で、メモリ閾値 (Memory threshold) を有効にしてあるかによる。デフォルトでメモリ閾値は無効となっている。この場合は、単に再生が失敗する。もしメモリ閾値が有効であれば、プライオリティールールが適用される。メモリ閾値を有効にすることが、強く推奨される。

## メモリプールの最適化

各種メモリプールを最適化するためには、どこでメモリを節約できるのかを、理解する必要があります。メモリ消費が大きい処理を、以下に示します。

- ・バンクをロードすると、デフォルトメモリプールのメモリ消費量が増えます。なお、メモリの消費量は、バンクによって異なります。バンクが、デフォルトメモリプールで使うメモリの大きさは、バンクの物理的なサイズによるのではなく、その中に入っているサウンドやイベントの合計数によります。
- ・特定のエフェクト、例えばリバーブやディレイなどは、再生中にロワーエンジンプールのメモリをある程度、消費します。
- ・複数のサウンドを同時再生すると、ロワーエンジンプールで使われるメモリの大きさが、急激に増えます。
- ・複数のアクションを、短期間に送信すると、デフォルトメモリプールのメモリ消費量が増えます。
- ・ゲームオブジェクトの登録、オブジェクト別のパラメータ設定、オブジェクトのポジション設定などはいずれも、デフォルトメモリプールのメモリを少量、使います。このメモリを開放するには、使用していないゲームオブジェクトの登録を解除しなくてはならないので、注意してください。解除しないと、使用中のメモリ量が増加し続けます。

メモリを最適化するには、以下のメモリプール種類について検討する必要があります:

- ・「ロワーエンジンメモリプール」
- ・「デフォルトメモリプール」
- ・「メディアのメモリプール (SoundBank) 」

### ロワーエンジンメモリプール

ロワーエンジンメモリプール (Lower engine memory pool) のメモリは、サウンドの再生に使われます。一度に再生中のサウンド数によって直接、影響されます。また、同時に使うエフェクトの合計数や種類によっても影響を受けます。ここで削減するには、同時にいくつのサウンドを聞かせたいのかを決める必要があります。ゲームによって、サウンドが10個以上聞こえるシナリオがほとんどないものもあれば、100個のゲームもあります。メモリプールのサイズを設定する時は、常にワーストケースシナリオを採用してください。

ほぼ比例関係にありますが、実際には、使用するコーデックや適用するエフェクトの数など、様々な条件によって変わります。

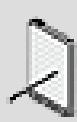
また、170個ものサウンドを同時再生することで、ゲームの価値が実際に上がるのかどうかを、改めて考える必要もあります。ある点を超えると、人間の耳には個々のサウンドが聞き分けられなくなり、単なるノイズとなってしまいます。ゲームにとって最も価値のあるボイス数を見つけ出す必要があります。これにはもちろん、試行錯誤が伴います。ゲームのシナリオを複数、プロファイリングして、Advanced Profilerの、Memoryタブを使って、リソースがどれだけ使われているのかを、確認してください。

ロワーエンジンプールのメモリ消費量を削減するには、一度に再生するボイス数を減らす必要があります。以下の戦略のいずれかを使えば、減らせます。

- **再生制限:** Advanced Settingsの設定。例えば、跳弾50弾分のサウンドが聞こえる必要が、本当にあるのか。おそらく、不要であり、15弾程度の無理のない数に制限することができます。また、バスの数も制限できます。
- **再生プライオリティ:** Advanced Settingsの設定。例えば、弾丸は、ダイアログほど重要でないとします。そこで、サウンド数が多すぎる場合は、まず弾丸の再生を落します。Playback Limitsと組み合わせて使います。
- **Distance-based Priority Offset:** Advanced Settingsの設定。一般的に、遠くにあるオブジェクトは、近くのものほど重要ではありません。例えば、10m先にある弾丸の音は、もっと近くに弾丸サウンドが15個ある場合は、聞こえる必要がありません。
- **Memory Threshold:** SDKの設定。サウンドエンジンの様々なメモリプールを初期化する時に、プールの使用率に上限を設けて、超えるとプライオリティ設定に基づいてボイスをキルする（消す）ように設定できます。これで、メモリに固定の使用制限がかかり、メモリ不足（Out of Memory）の状況をほぼ完全に回避できます。なお、メモリにフラグメンテーションが起きるので、全てを使い切ることが不可能なため、スタート地点として「0.9」つまり90%程度が適切です。
- **Volume Threshold:** Project Settingsの設定。これは、小さすぎて聞こえないようなサウンドをキルする（消す）ために使います。この設定と共に使われるものは、閾値以下の動作設定（Below Threshold Behavior）や、遠くに行けば一般的に音が小さくなる減衰（Attenuation）設定です。デフォルト値は「-80」です。
- **ボリューム閾値以下の動作（Below Volume Threshold Behavior）:** Advanced Settingsの設定。負荷が最も低い動作（CPU、メモリ）は、「Kill Voice（ボイスを消す）」です。このオプションは、短いサウンドやループするアンビエンスサウンドに使うと便利で、ボイスは消されますが、ループは消されません。次に望ましいオプションは、「Send To Virtual（バーチャルボイスリストに送る）」+「Play from beginning（最初から再生）」であり、その次は「Send To Virtual（バーチャルボイスリストに送る）」+「Resume（再開）」、その次は「Send To Virtual」+「Play from elapsed time（経過した時間から再生）」と続きます。最も負荷が高いのは「Continue to play（再生を続ける）」と、「Play from elapsed time（経過した時間から再生）」のオプションです。なお、Wwiseのデフォルト値は「Continue To Play（再生を続ける）」なので、注意してください。

## デフォルトメモリプール

デフォルトメモリプール (Default memory pool) には、メモリにロードされた各種サウンドやイベントに関する構造的なメタデータが入っています。Wwiseで定義した動作を実行するために必要なオブジェクトのプロパティが、全て入っています。また、登録されている全てのゲームオブジェクトと、それらに関連したゲームシンク値、ポジション、向き (Orientation)などの情報も含まれます。メモリにロードされるバンク数が増えると、メタデータも追加されるので、必要なデフォルトメモリプールのサイズが大きくなります。必要なメモリサイズを最終的に決めるのは、1つのシナリオ、レベル、マップ、またはゲーム領域などで再生される可能性のあるサウンドの、最大数です。



### 注記

デフォルトメモリプールに、メディアは一切入りません。

サウンド構造やイベントの組み合わせは、それぞれ、デフォルトメモリプールのメモリを平均で約300バイト使います。例えば、プロジェクトにあるサウンドが（ダイアログも含め）50,000個以上であれば、構造データだけでも、約15MBのメモリが必要です。明らかに、全てを同時にメモリにロードすることはできないので、構造データもメディアと同様に、ロードしたりアンロードしたりする必要があります。

これまでの数々のWwiseプロジェクトを参考にすると、デフォルトメモリプールに占める割合が最も高いのが、サウンド構造（25～50%程度）で、メモリにロードする個数があまりにも多いためです。もちろん、ロードするサウンド数が少なければ、デフォルトメモリプールも小さくできます。次にメモリ消費が高いのがイベントのメタデータで、プール全体の約10%を占めます。イベントは非常に小さいので、数が多くてもあまり場所をとりません。単体でメモリ消費が最も高いのが、おそらくRandom ContainerやSequence Containerで、単純なサウンドと比べ、メモリ消費が3倍にになります。しかし一般的に、サウンドやイベントよりもRandom ContainerやSequence Containerの数がかなり少ないので、合算しても、メモリプールに占めるコンテナの割合は低いです。

基本的な考え方として、オープンワールドゲームは、すぐに再生できるように待機させるサウンドが多いため、必要メモリが大きくなる傾向があります。このようなゲームでは、デフォルトメモリプールが5～8MBほど必要です。これに比べて、他の種類のゲームのメモリ消費はかなり低く、例えば2～3MBほどです。なお、これらの数値は、Wwiseを使った数々のゲームが今まで必要としたメモリの情報を、参考として提供するものであって、Wwiseでデザインするゲームが、このメモリ容量に収まることを保証するものではありません。

デフォルトメモリプールのメモリ消費を削減するためのベストプラクティスを、以下に示します。

- 多数のサウンド構造やイベントを入れた大きいバンクは、いくつかの小さいバンクに分割すること。必要に応じて、バンクをロードしたりアンロードしたりします。バンクはキャラクター別に分けるだけでなく、必ず状況別の分割も行います。
- APIの、ExecuteActionOnEventを使って、イベント数を減らすこと。  
「Play/Stop」イベントの代わりに、Playイベント1つと、Stopするための

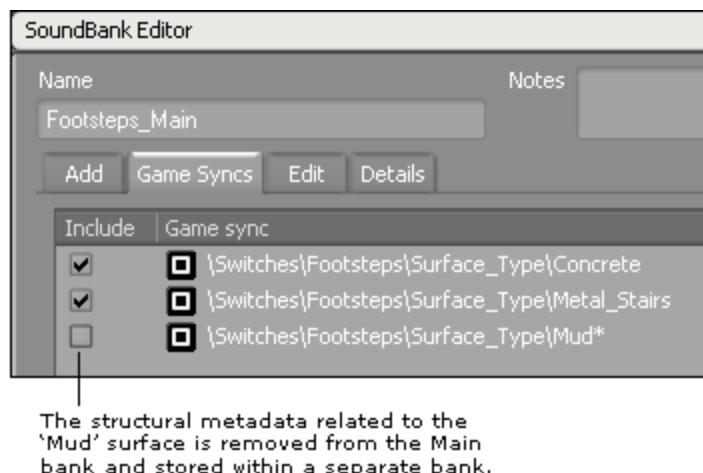
ExecuteActionOnEventの呼び出し1つに、置き換えます。同じPlayイベントの「Pause/Resume」も同様です。

- ゲームオブジェクトを細かく管理すること。役割が終わったものは、すぐに登録を解除します。全くメリットがない上に、メモリを消費してしまうので、使用しないゲームオブジェクトのプールを生かしたままにしないでください。例えば、NPCが死んだ場合は、そのゲームオブジェクトの登録を解除すること。他の目的で再利用しないでください。必要となった時点で、新しいゲームオブジェクトを登録します。基本的な考え方として、数千個のゲームオブジェクトが生きている状態は、多すぎると言えます。
- サウンドの整理には、仮想フォルダを使い、アクターミキサーを使わないこと。仮想フォルダはメモリを消費しませんが、アクターミキサーは消費します。アクターミキサーは、デフォルトプロパティ以外の、他のプロパティも似た複数オブジェクトに限り、使います。アクターミキサーを使うと、プロパティが似ていれば、1回出てくるだけなので、メモリを節約できます。もちろん、アクターミキサーが、Set Voice VolumeやSet Voice Pitchなどのイベントにレファレンスされるかどうかにも、ります。
- 大きい階層は、サイズや複雑性を減らすように努力すること。よくある大きい階層として、「Impact（衝撃）」階層や、「Footstep（足音）」階層などがあります。これらは変数が多くいため、大きくなりがちで、残念ながら分割できません（メディアは分割できますが、構造データは分割できません）。このような階層を小さくする方法を、以下にいくつか示します。
  - 変化するのが、単純なプロパティであれば、RTPCを使います（サンプルは同じで、ボリューム、ピッチ、Randomizerなどが変わること）。
  - Switch階層の複数のレベルの一部（または全て）を、同等のStateやState Group（State Group）が設定されたダイアログイベントに置き換えられます。頻繁に変わる変数を使うと、便利です。例えば、「Impact」階層では、衝撃を受ける素材を事前に既定できず、イベントが発生した時に、分かるだけです。Switch値を保持しなければならないSwitch Containerの代わりに、同じ値のStateやState Groupを設定したダイアログイベントを使えます。ダイアログイベントは、同じサンプルやサブ構造を簡単に再利用できるのが、メリットの1つです。また、ダイアログイベントでは、ゲームオブジェクトごとにメモリを使いません。StateやState GroupパスをSwitch Containerにリンクして、変数値のレベルをさらに増やすこともできます。
  - 例えば、「Footsteps（足音）」にこの方法を活用できます。StateやState GroupパスをSwitch Containerにリンクして、変数値のレベルをさらに増やすこともできます。この仕組みの基にあるのは、ゲームオブジェクトの寿命中に「Footwear」はあまり変わらず、他の2つの変数値はよく変わるという論理です。
  - この方法のもう1つのメリットは、ダイアログイベントを、ダイアログイベントで再生される関連サウンド構造が入っていない、イベントだけのSoundBankに入れられることです。つまり、そのイベントに関連した、複雑な各種サウンド構造を、ほかの複数のSoundBankに分割できるのです。例えば、「Footsteps（足音）」の例では、「BankEvent」の中に、足音のDialogue Eventを入れ、「Concrete（コンクリート）」関連の構造やメディアを「BankConcrete」に入れ、「Dirt（泥）」関連の構造やメディアを「BankDirt」に入れます。使用方法としては、新たな地面の種類に接近する

時に、SoundBankをダイナミックにロードしたりアンロードしたりするだけです。

- Switch Container階層を、複数のSoundBankに分けることができます。SoundBankにSwitch Containerを追加する時に、コンテナのサブブランチも全て、自動的に含まれます。ただし、SoundBank Editorの、Game Syncsタブや、Editタブで、マニュアル操作で一部のサブブランチを除外することもできます。例えば、「Footstep」階層の最上位にあるSwitch変数が、地面の種類だとします。ゲーム中に、常に全種類の地面を待機させる必要がなければ、Switch Containerの階層を複数のSoundBankに分割して、ゲームの状況に応じて、必要な地面の種類だけをロードすることができます。そこで、例えば「Concrete（コンクリート）」や「Metal Stairs（鉄骨階段）」など、ゲーム中にどこにでも出現するような地面の種類をメインバンク「Footstep」に入れて、「Mud（泥）」など、出現するシーンや場所が限定される地面に関しては、状況別に分けたほかのSoundBankに入れることができます。

ある衝撃サウンドのバリエーションをつくり出すために、.wavファイルをいくつも準備するのではなく、SoundSeed Impactの同等サウンド1つで対応できます。例えば「ガチャン」という音を10種類も用意せずに1つのサウンドで置き換えれば、経済的なだけでなく、バリエーションが10種類よりも多くなります。この手法を使うと大きな差が出るので、見落とさないでください。



## メディアのメモリプール (SoundBank)

SoundBank (SoundBank) のメモリ消費は主に、中にあるサウンドデータに起因します。サウンドデータ（メディア）のメモリ消費は、以下のテクニックで抑制できます。

- 多数のサウンドが入った大きなバンクは、小さいバンクに分ける。必要に応じて、バンクをロードしたりアンロードしたりする。
- ディスクからストリーミングするサウンドの数を増やす（サウンドオブジェクトのプロパティ）。
- APIの、PrepareEvent/UnprepareEventを使う。
- オーディオの圧縮率を上げる（コンバージョン設定、コーデックなど）。
- サンプルレートを下げる。サンプルレート自動検知（Automatic Sample Rate Detection）機能を検討してみる（コンバージョン設定）。

- ・衝撃 (Impact) 関連のサウンドを、プラグインSoundSeed Impactの同等サウンドに置き換える。サウンドの複数のバリエーションを入れたRandom Containerがある場合は、このようなコンテナを置き換えるだけで、メモリを大幅に節約できます。なお、SoundSeed Impactを使って作成できる衝撃音以外のバリエーションも、いくつかあります。
- ・風 (Wind) 関連のサウンドを、プラグインSoundSeed Wind/Wooshの同等サウンドに置き換える。風のアンビエントサウンドは、ループサウンドであることが多い、メディアスペースをかなりとります。プラグインSoundSeed Airを使って作成できるサウンドは多岐におよび、空を切る刃音、プロペラ、車窓から吹き込む風、空調音などのほかに、波の音や離れた高速道路の音などのノイズも、作成できます。

## メモリ管理に関する、コツとベストプラクティス

ゲームのオーディオメモリを効率的に管理するための、コツやベストプラクティスについて、以下のセクションを参照してください。

### メモリ閾値の活用

本書の別のセクションでも述べた通り、サウンドエンジンのメモリプールの1つ、または複数に対して、メモリ閾値 (Memory threshold) を設定することができます。オーディオプログラマーが、サウンドエンジンの初期化パラメータにある以下の値を変更することで、メモリプールの閾値 (Threshold) を設定できます。

- AkInitSettings::fDefaultPoolRatioThreshold
- AkPlatformInitSettings::fLEngineDefaultPoolRatioThreshold

デフォルトで、メモリ閾値にデフォルト値「1」（つまり100%）を設定して、解除してあります。有効にするには、「0」から「1」までの間の数値を設定します。

メモリ閾値が解除されている時は、メモリアロケーション機能 (Memory allocator) が正常に作動します。閾値を有効にすると、メモリの使用率を、指定した閾値よりも低く維持するために、エンジンが定期的にチェックします。閾値を超えると、最低プライオリティのサウンドを、システムが順にドロップし始めます。

メモリ閾値が設定されていない状況では、エンジンは各サウンドのプライオリティを遵守しますが、メモリ量が少なくなると、高プライオリティのサウンドを再生するメモリが不足してしまえば、それをドロップします。メモリ閾値を設定してある場合は、プライオリティの低いサウンドからドロップして、プライオリティが高いサウンドのためにスペースを空けます。

### サウンドエンジンの、メモリフラグメンテーションの対処

まとめた量のデータをメモリに追加したり削除したりすると、多少のフラグメンテーションが必ず発生します。Wwise自体がメモリフラグメンテーションの対処を行うことはありませんが、発生するフラグメンテーションを減らすためにメモリプールの設計で考慮してあります。

デフォルトメモリプールは、主にサウンド構造のメタデータ用に使われ、主に小さいオブジェクトが入っているので、フラグメンテーションの問題はそれほどありません。ロワーエンジンメモリプールは、オーディオのパイプラインと処理のために確保しており、大きいブロックで構成されますが、オーディオが消えると、これらのブロックは開放されます。この2種類のメモリプールに関しては、それぞれ約10%のオーバーヘッドを割り当てるだけで、フラグメンテーションをほぼ完全に回避できます。

一方、様々なバンクに保存されるオーディオデータは、LoadBank()コールごとに、ユーザーが設定する別のプールにコピーされるので、オーディオデータのメモリ管理に関しては、オーディオプログラマーが細かくコントロールできます。メモリフラグメンテーションの発生を最小限に抑えるために、オーディオプログラマーは、CreatePoolのために AkFixedSizeBlocksMode フラグを送ることで、サイズが固定されたブロック (Fixed size blocks) のプールを使い、ブロックサイズを、各プールの最大のSoundBankのサイズにします。これで、全てのSoundBankを、最大バンクのサイズと同じであるように配分されます。もちろん、プールが複数あり、プールごとに、特定のSoundBankサイズに合わせて、スロット（ブロック）を一定数提供するという方法も可能です。AkFixedSizeBlocksMode プールのメモリブロックは、GetBlock()で取得され、常に同じサイズとなります。このテクニックで、フラグメンテーションを完全に回避できます。

---

## 第34章 Profiling

概要 .....	631
Wwiseにおける、プロファイリングの種類を理解する .....	632
ローカルまたはリモートで、ゲーム機に接続する .....	637
サウンドエンジンからくるデータをキャプチャーする .....	640
Performance Monitorを使った、モニタリングとトラブルシューティング ....	649
Game Object Explorerで、オブジェクトやリストナーをトラッキングする .....	652
Game Object 3D Viewerで、オブジェクトを観察する .....	656
Game Sync Monitorで、ゲームシンクを評価する .....	660
プロファイリングに関する、コツとベストプラクティス .....	661

## 概要

ゲームの開発者にとって最も高いハードルの1つは、各プラットフォームの制限や制約に配慮しながらも、リッチで没入感のあるゲームを開発することです。ゲームのオーディオやモーションを様々なプラットフォームに対応させる手段が、Wwiseには多数あります。さらに、Wwiseの**Profiler**レイアウトを使い、各プラットフォームにおけるオーディオやモーションのパフォーマンスをテストできます。

**Profiler**レイアウトを使って、開発プロセスの最中にいつでも、どのプラットフォームでも、ゲームの詳細をプロファイリングできます。プロファイリングは、Wwiseでローカルに行うことも、ゲーム機にリモート接続して行うこともできます。どちらの方式も、サウンドエンジンから直接、プロファイリング情報をキャプチャします。サウンドエンジンのアクティビティをモニターすることで、メモリ、ボイス、ストリーミング、エフェクトなどに関連する具体的な問題を検知して、トラブルシューティングできます。また、Game Simulatorや、Soundcasterを使い、ゲームに実装する前から、プロトタイプのプロファイリングが行えます。

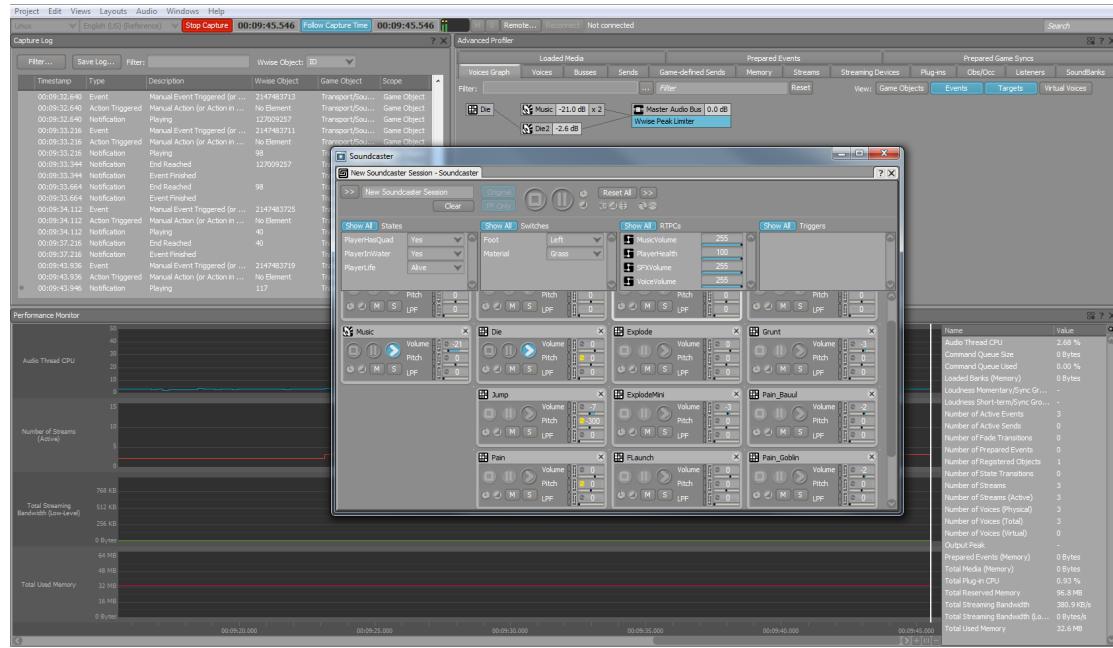
### ローカルで行うプロファイリング

オーディオやモーションをゲームに実装する前から、Soundcasterで、イベント、サウンドオブジェクト、モーションオブジェクト、ゲームシンクの切り替えなどをトリガーして、Wwiseプロジェクトを様々な角度からプロファイリングできます。

#### ローカルで行うプロファイリング:

1. Profilerレイアウトに切り替える。
2. Soundcasterに複数のEvents、オブジェクト、バスを入れる。
3. Start Captureをクリックする。
4. SoundcasterでEventsやオブジェクト、Game Sync切り替えなどをトリガーさせる。

サウンドエンジンの情報が、Profilerの各種ビューでキャプチャされ、表示されます。



### ゲームプレイ中に使うプロファイリング

オーディオやモーションをゲームに実装できたら、Wwiseをゲームに接続して、ゲームプレイ中にリアルタイムで、プロファイリングできます。ゲームに接続できれば、ゲーム中にリアルタイムでオブジェクトの相対プロパティの調整やミックスもできます。

**注意**

デバッグコンフィギュレーションのパフォーマンスが最適化されていないため、プロファイリングする時は、ゲームのデバッグビルドからであっても、Wwiseサウンドエンジンのプロファイルビルドコンフィギュレーション (Profile build configuration) に接続することが推奨されます。したがって、あなた自身のゲームのデバッグビルドを実行している場合であっても、WwiseのDebugビルドを使用しないことをお勧めします。プロファイリングのためWwiseサウンドエンジンのProfileビルド構成にのみ接続することをお勧めします。

### ゲームをプロファイルする：

1. ゲームを開始する。
2. Profilerレイアウトに切り替える。
3. Remote Connection機能を使ってゲームに接続する。
4. Start Captureをクリックする。
5. ゲームをプレイし始める。

サウンドエンジンの情報が、Profilerの各種ビューでキャプチャされ、表示されます。

## Wwiseにおける、プロファイリングの種類を理解する

Wwiseで行えるプロファイリングは、以下の2種類です。

- ・ゲームの、プロファイリング
- ・ゲームオブジェクトのプロファイリング

ゲームのプロファイリングを行うと、サウンドエンジンやプロジェクトの様々なコンポーネントの観点から、パフォーマンス要件や要求に関する情報を表示します。サウンドのエフェクトがプラットフォームのパフォーマンスに与える影響を累積値としてリアルタイムで表示でき、個別のボイスの影響を調査できます。

ゲームオブジェクトのプロファイリングにおいても、サウンドエンジンのアットプットを分析しますが、個別のゲームオブジェクトの観点から結果を表示します。ゲームオブジェクトをトラッキングするので、リアルタイムでゲームオブジェクトの行動や動作傾向を観察することができます。これにより、問題を起こしているゲームオブジェクトなどを検出できます。

ゲームオブジェクトは、ゲーム内に存在する個別の物体です。それらはエミッタまたはリスナになることができるゲーム内のすべての要素について、オーディオプログラマによって登録または作成されます。これには、プレイヤーキャラクター、ノンプレイヤーキャラクター (NPC)、武器、車両、モンスター、トーチのような周辺物などが含まれます。独立して移動するようにプログラマされたゲームオブジェクトは、サウンド、音楽、モーション、またはそれらに関連付けられたバスオブジェクトを配置することができます。ゲームオブジェクト用のプロファイラ機能 (Game Object Explorer、3D Game Object Viewer、Game Sync Monitor) を組み合わせて、ゲーム中やシミュレーション中のゲームオブジェクトを観察します。

### ゲームサウンドのプロファイリング活用例

例えば、大都市の中心で戦う巨大なモンスターを、プレイヤーが操作するファイティングゲームを作成する場合を考えます。ゲームプロファイリングツールと、ゲームオブジェクトプロファイリングツールの両方を使い、ゲームサウンドのパフォーマンスを分析します。

ゲームプロファイリングツールを使い、以下を分析します。

- ・様々なモンスター、警察官、一般市民などに関連する多数のサウンドが、そのプラットフォームのストリーミング機能を使う様子。
- ・建物が崩壊する音など、バックグラウンドのノイズが、いつ、どのようにバーチャルボイスに移行するか。
- ・モンスターの様々な唸り声に、どのエフェクトプラグインが適用されていて、これらのエフェクトで、CPUへの負荷がどう変化するか。
- ・どのAPIファンクションがタイムラインで特定の瞬間にコールされたか。

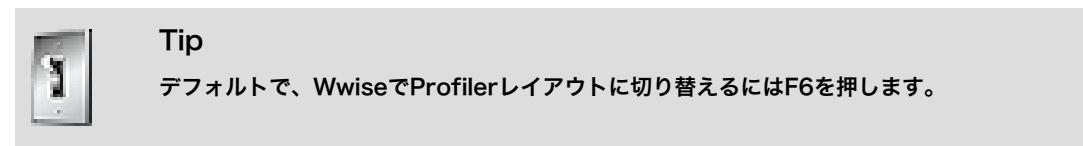
ゲームオブジェクト用プロファイリングツールを使い、以下を分析します。

- ・各モンスターの様々なサウンドの減衰半径が、他のモンスターの減衰半径と、どう影響し合うか。
- ・警察ヘリなどのゲームオブジェクトが、お互いの位置関係や、モンスターとの位置関係に応じて、動く様子。
- ・RPCの設定、例えば「Panic (パニック)」などの設定が、一般市民ゲームオブジェクトのサウンド再生に、どのような影響を与えるか。

このように、ゲームプロファイリングツールや、ゲームオブジェクト用プロファイリングツールを使い、ゲーム稼働時のサウンドスケープを、完全に把握できます。

### ゲームプロファイラーの、各種ビューを使いこなす

Wwiseのゲームプロファイリングツールは3つのビューに分かれています。合わせて使うことで、プロジェクトのパフォーマンスを細かく分析できます。これらのビューは、Profilerレイアウトに組み込まれています。



#### Capture Log

Capture Log（キャプチャーログ）は、サウンドエンジンからくる情報を全て、キャプチャーして記録できます。Capture Logのフィルタ（Filter）機能を使って、キャプチャーログに表示させる情報を設定します。どの種類のデータをキャプチャするかを制限するためには、Advanced Profiler's Profiler Settingを調整してください。

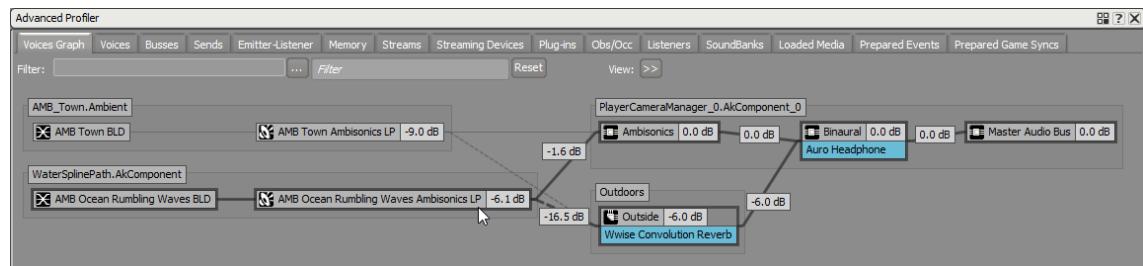
Timestamp	Type	Description	Wwise Object	Game Object	Scope
00:00:51.968	Error	Unload bank failed, request...			
00:00:51.968	Bank	Bank Unload Request Recei...	2185786256		
00:00:51.968	Bank	Bank Unloaded (from Gravel)	2185786256		
00:00:54.336	Bank	Bank Load Request Receive...	983016381		
00:00:54.357	Bank	Bank Loaded (from Car)	983016381		
00:01:04.917	Bank	Bank Unload Request Recei...	983016381		
00:01:04.917	Bank	Bank Unloaded (from Car)	983016381		
00:01:06.666	Bank	Bank Load Request Receive...	2309453583		
00:01:06.688	Bank	Bank Loaded (from MarkerT...)	2309453583		
00:01:09.760	Bank	Bank Unload Request Recei...	2309453583		
00:01:09.760	API Call	StopPlayingID: Fade Curve:...			
00:01:09.760	Message	StopPlayingID API called			
00:01:09.760	Bank	Bank Unloaded (from Marker...)	2309453583		
00:01:11.210	Bank	Bank Load Request Receive...	2279279248		
00:01:11.253	Bank	Bank Loaded (from Interact...)	2279279248		
00:01:11.253	State Changed	From state "None" to state "..."	3991942870		
00:01:22.816	Music Switch C...	Fight_Dying -> FightLoopLe...	523495112		
00:01:22.816	State Changed	From state "Fight_Damaged..."	3991942870		
00:01:25.034	Music Switch C...	Winning_TheFight -> WinTh...	523495112		
00:01:25.034	State Changed	From state "Fight_Dying" to ...	3991942870		
00:01:26.826	Music Switch C...	Fight_Dying -> FightLoopLe...	523495112		
00:01:26.826	State Changed	From state "Winning_TheFig..."	3991942870		
00:01:27.872	Music Switch C...	GameOver -> GameOver	523495112		
00:01:27.872	State Changed	From state "Fight_Dying" to ...	3991942870		
00:01:30.773	Bank	Bank Unload Request Recei...	2279279248		
00:01:30.773	API Call	StopPlayingID: Fade Curve:...			
00:01:30.773	Message	StopPlayingID API called			
00:01:30.773	Bank	Bank Unloaded (from Interac...)	2279279248		
00:00:40.960	API Call	SetRPCValue: Target Value...	CC #26	Transport/Soundcaster	Game Object
00:00:40.960	API Call	SetRPCValue: Target Value...	CC #32: C...	Transport/Soundcaster	Game Object
00:00:40.960	API Call	SetRPCValue: Target Value...	CC #50: C...	Transport/Soundcaster	Game Object

## 関連トピック

- 「サウンドエンジンからくるデータをキャプチャーする」

### Advanced Profiler

Advanced Profiler（アドバンスドプロファイラ、詳細プロファイラ）に、サウンドエンジンが実行する各アクティビティのパフォーマンス情報として、CPU、メモリ、帯域幅などが表示されます。サウンドエンジンからキャプチャーした情報が、リアルタイムで表示されます。Advanced Profiler画面には、複数のタブがありますが、タブの内容をProfiler Settingsダイアログボックスで設定できます。



Wwise Audio Labのキャプチャ内容を示したAdvanced Profilerであり、プレイヤーが村で外に立っている

### Performance Monitor

Performance Monitorは、パフォーマンスをモニターして、問題のトラブルシューティングを行うために、サウンドエンジンの様々な指標をまとめた画面です。必要な情報だけをPerformance Monitorに表示するように設定できます。



## 関連トピック

- 付録C ショートカット

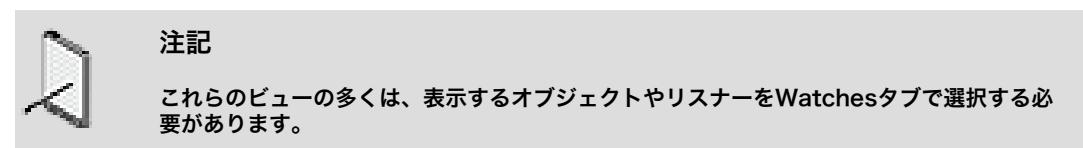
## ゲームオブジェクト用プロファイラーの、各種ビューを使いこなす

Wwiseのゲームオブジェクト用プロファイリングツールは3つのビューに分かれていて、合わせて使うことで、様々なゲームオブジェクトをモニターして、ゲームのオーディオやモーションに与える影響を確認できます。これらのビューは、Game Object Profilerレイアウトで同時に見れます。



### Game Object Explorer

Game Object Explorer（ゲームオブジェクトエクスプローラ）は、Wwiseのゲームオブジェクト用プロファイリングツールの、コントロールセンターです。オーディオプログラマーがゲームオブジェクトをゲームで登録すると、リアルタイムでGame Object 3D Viewer、Game Sync Monitor、そしてRTPCタブやAttenuation Editorのグラフビューで観察できます。ゲームに接続すると、すぐにGame Objectsリストに、現在ゲームに登録されている全てのゲームオブジェクトが表示されます。

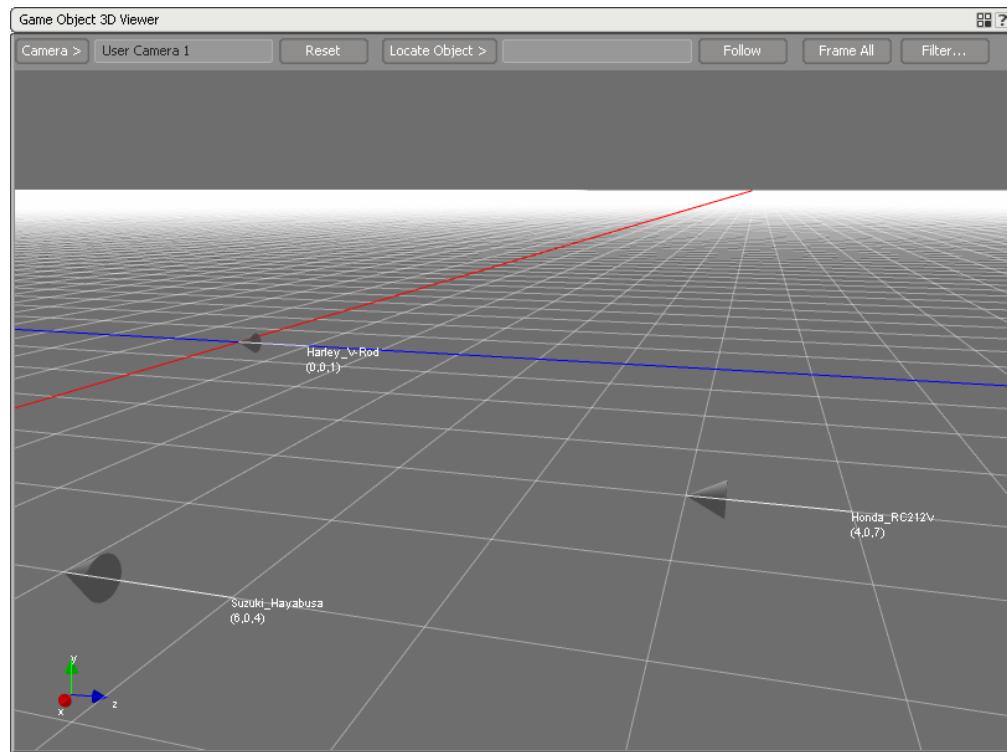


Game Object Explorer

Name	ID	Last Registered	Last Unregistered	Is Alive
AKPhysicsTriggerWaterImpact (phy_water...)	19136937	00:13:35.296	-	✓
AMBIENCE	4294967...	00:13:35.296	-	✓
Boy	196665	00:13:35.296	-	✓
CollisionRect2D	44761606	00:13:35.296	-	✓
CollisionRect2D	786709	00:13:35.296	-	✓
door	26411081	00:13:35.296	-	✓
pelvis	196646	00:13:35.296	-	✓
Sound	2622011	00:13:35.296	-	✓

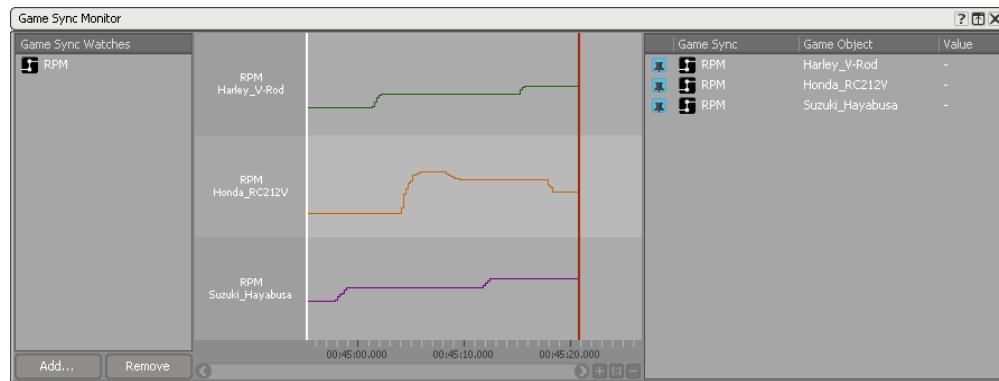
### Game Object 3D Viewer

Game Object 3D Viewer は、ゲームオブジェクトや他のゲームオブジェクトとの関係、そして、リスナーとしての潜在的な役割などを3次元で視覚表現するものです。さまざまな種類の情報を表示し、異なるゲームオブジェクトの表示方法を変更するように、 Game Object 3D Viewerを設定することができます。



### Game Sync Monitor

Game Sync Monitor（ゲームシンクモニター）は、RTPC値をリアルタイムに分析するためのツールです。ウォッチ（観察）中のゲームオブジェクトに設定された RTPC 値の変化が、ゲームプレイ中に、グラフとして表示されます。



## ローカルまたはリモートで、ゲーム機に接続する

ゲーム内で様々なサウンドのシミュレーションを行ったり、ある特定のプラットフォームでゲームの様々な面をプロファイリングしてトラブルシューティングしたりするには、まず、ゲームを稼働させるマシンに接続する必要があります。LAN でアクセスできる稼働中のWwiseサウンドエンジンがあれば、接続できます。

**注意**

デバッグコンフィギュレーションのパフォーマンスが最適化されていないため、プロファイリングする時は、ゲームのデバッググビルドからであっても、Wwiseサウンドエンジンのプロファイルビルドコンフィギュレーション (Profile build configuration) に接続することが推奨されます。

プロファイル機能で接続するマシンを見つけやすくするために、Wwiseはネットワーク上の同じサブネットで、Wwiseのいずれかのバージョンのサウンドエンジンが稼働する全てのPCやゲーム機を自動的に検出します。サブネット外にあるゲーム機やPCに接続する時は、プラットフォームのIPアドレスを直接入力します。

**注記**

WwiseのRemote Connectionsダイアログボックスに、接続したいゲームが見つからない場合は、「[リモート接続の問題のトラブルシューティング](#)」を参照して、この問題のトラブルシューティングを行ってください。

**ローカルまたはリモートにあるゲーム機に接続するには:**

- ツールバーで、Remoteボタンをクリックする。

Remote Connectionダイアログボックスが開き、Wwiseのいずれかのバージョンのサウンドエンジンが稼働する、ネットワーク上にあるゲームシステムのリストが表示される。

- リストで、接続するプラットフォームを1つ選択する。

**注記**

マシンに接続するには、そのゲーム機のステータスが"Ready"である必要があります。

- 接続した瞬間に情報のキャプチャーを開始するには、Start capture on connect (接続してキャプチャー開始) オプションを選択する。
- Connectをクリックする。

Wwiseが、ローカルまたはリモートプラットフォームに接続します。接続したマシンの名前がツールバーに表示される。

**注記**

プラットフォームにマニュアル操作で接続するには、Connect To IPをクリックしてリモートマシンのIPアドレスを入力します。

**ローカルまたはリモートにあるマシンとの接続を切断するには:**

- ツールバーで、Disconnectボタンをクリックする。

Wwiseが、そのマシンとの接続を切断する。

**関連トピック**

- [「Remote Connection History Listを活用する」](#)

- ・「サウンドエンジンからくるデータをキャプチャーする」
- ・「ネットワークポートの指定」
- ・「リモート接続の問題のトラブルシューティング」

## Remote Connection History Listを活用する

過去に接続したことのある全てのマシンやシステムは、WwiseのHistory（履歴）に記録されます。このため、接続したことのあるシステムは簡単に見つけられます。マシンに接続する前にそのステータスを検証する必要があります。マシンのステータスは、以下のいずれかです。

- ・Ready: 接続を受ける準備ができている。
- ・Busy: 既に1つのマシンに接続されているため、他の接続を受け入れられない。
- ・Different Version: Wwiseのバージョンとリモートシステムで稼働中のゲームのバージョンの互換性がない。
- ・Not Available: 既にネットワークに接続されていない。

履歴を整理すると、リストに表示されるマシンを非表示にできます。

## Historyリストを使ってシステムに接続するには:

1. ツールバーで、Remoteボタンをクリックする。

Remote Connectionダイアログボックスが開き、ネットワーク上で稼働中のマシンのリストが表示される。

2. Historyタブに切り替える。
3. Historyリストで、接続するマシンを選択する。
4. 接続した瞬間に情報のキャプチャーを開始するには、Start capture on connect（接続してキャプチャー開始）オプションを選択する。
5. Connectをクリックする。

Wwiseが、ローカルまたはリモートマシンに接続する。接続したマシンの名前がツールバーに表示される。

## Historyリストを消去するには

1. ツールバーで、Remoteボタンをクリックする。

Remote Connectionダイアログボックスが開き、ネットワーク上で稼働中のマシンのリストが表示される。

2. Historyタブに切り替える。
3. Historyリストで、削除するマシンを選択する。
4. 以下のいずれかを行う:
  - ・Removeをクリックする。
  - ・Deleteキーを押す。

このマシンがHistoryリストから削除される。

## 関連トピック

- 「ローカルまたはリモートで、ゲーム機に接続する」
- 「サウンドエンジンからくるデータをキャプチャーする」
- 「リモート接続の問題のトラブルシューティング」

## サウンドエンジンからくるデータをキャプチャーする

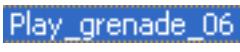
マシンに接続できたら、サウンドエンジンから来るデータを直接キャプチャーしてゲームのオーディオやモーションのプロファイリングを開始します。サウンドエンジンから来る情報は、全てCapture Log（キャプチャーログ）に表示されます。Capture Logで記録される情報の種類を、以下に示します。

- Notifications
- Markers
- Events
- Actions
- Properties
- States
- Switches
- SoundBank
- Events Preparation
- Errors
- Messages
- MIDIイベント
- API Calls
  - Events
  - RTPCs
  - Game Syncs
  - Game Objects
  - Listener Spatialization
  - Positions
  - Auxiliary Bus
  - Obstruction/Occlusion
  - Motion
  - MIDI
  - Dynamic Sequence
  - Spatial Audio
  - Others

これらの項目は全て、Performance MonitorやAdvanced Profilerで、モニターできます。これらのビューに、メモリ、ボイス、エフェクト、ストリーミン

グ、SoundBank、プラグインなどの詳細情報が入っています。パフォーマンスのモニターについては「[Performance Monitorを使った、モニタリングとトラブルシューティング](#)」を参照してください。また、Wwiseでキャプチャーしたデータを解析して、ゲーム、つまりプロジェクトの具体的なオーディオエレメントの統計データを表示させることもできます。統計情報の収集については「[キャプチャセッションから統計情報を収集する](#)」を参照してください。

Capture Logで多数のエントリ（行）が表示されることがあるので、素早く見分けられるようにWwiseでは以下のインジケータ（しるし）や色分けが使われています。

項目	内容
<b>インジケータ</b>	
白・灰色の円	白と灰色の円は、キャプチャーログにおいて、Performance Monitorのタイムカーソルの位置から100ミリ秒以内でキャプチャーされたエントリを示します。インジケータが白ければ白いほど、このエントリがタイムカーソルの位置に近いことを意味します。カーソルをログの特定の行のタイムスタンプに強制的に移動させるには、Shiftを押しながらその行をクリックします。
青の円	青い円は、キャプチャーログのアイテム（項目）のうち、互いに関係するものを示します。ログでアイテムを1つ選択すると、他のアイテムに青い円が表示されます。
<b>色による表示</b>	
選択したエントリ	キャプチャーログで選択されたエントリが、青色にハイライト表示されます。 
関連エントリ	選択したエントリに関連するエントリは全て、ティールブルー色にハイライト表示されます。 
エラー	キャプチャープロセス中に発生したエラーは全て、黄色にハイライト表示されます。 
Messages	メッセージは全て、緑色にハイライト表示されます。 

Capture Logで使われる様々なインジケータや色分けを、下図に示します。

Any errors are highlighted in yellow.

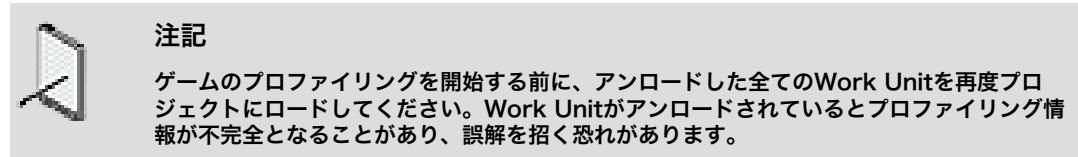
White/gray indicators show entries captured within 100ms of the Performance Monitor time cursor.

Blue indicators show which entries are related to one another. Related entries are also highlighted in teal blue.

All messages are highlighted in green.

Timestamp	Type	Description	Wwise Object	Game Object
00:00:07.081	Bank	Bank Load Request Received (from Init.bnk)	Init	
00:00:07.081	Bank	Bank Loaded (from Init.bnk)	Init	
00:00:07.081	Bank	Bank Load Request Received (from main.bnk)	main	
00:00:07.097	Bank	Bank Loaded (from main.bnk)	main	
00:00:07.113	Message	Main Character walks.		
00:00:07.113	Message	Unknown Event name : S_FOOTOO		
00:00:07.113	Error	ID not found		
00:00:07.144	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.144	Action Triggered	Play	footsteps	External Auth...
00:00:07.160	Notification	Play	F5_Concrete_04	External Auth...
00:00:07.237	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.237	Action Triggered	Play	footsteps	External Auth...
00:00:07.253	Notification	Play	F5_Concrete_02	External Auth...
00:00:07.347	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.347	Action Triggered	Play	footsteps	External Auth...
00:00:07.375	Notification	Play	F5_Concrete_01	External Auth...
00:00:07.440	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.440	Action Triggered	Play	footsteps	External Auth...
00:00:07.452	Notification	Play	F5_Concrete_03	External Auth...
00:00:07.542	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.542	Action Triggered	Play	footsteps	External Auth...
00:00:07.558	Notification	Play	F5_Concrete_04	External Auth...
00:00:07.643	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.643	Action Triggered	Play	footsteps	External Auth...
00:00:07.659	Notification	Play	F5_Concrete_02	External Auth...
00:00:07.729	Notification	End Reached	F5_Concrete_04	External Auth...
00:00:07.729	Notification	Event Finished		
00:00:07.737	Message	rtpclspos=FRONT, Sound plays in the BACK		
00:00:07.747	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.747	Action Triggered	Play	footsteps	External Auth...
00:00:07.764	Notification	Play	F5_Concrete_01	External Auth...
00:00:07.851	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.851	Action Triggered	Play	footsteps	External Auth...
00:00:07.879	Notification	Play	F5_Concrete_03	External Auth...
00:00:07.934	Notification	End Reached	F5_Concrete_01	External Auth...
00:00:07.934	Notification	Event Finished		
00:00:07.952	Event	Event Triggered	S_FOOTOO	External Auth...
00:00:07.952	Action Triggered	Play	footsteps	External Auth...
00:00:07.952	Notification	End Reached	F5_Concrete_02	External Auth...
00:00:07.952	Notification	Event Finished		
00:00:07.973	Notification	Play	F5_Concrete_01	External Auth...
00:00:08.061	Event	Event Triggered	S_FOOTOO	External Auth...

アイディアやプロトタイプなどをゲームに実装する前に、WwiseのSoundcasterで直接キャプチャーしてプロファイリングすることも可能です。Soundcasterの使い方については[32章シミュレーションの作成](#)を参照。



### キャプチャーする情報タイプを指定する

転送データが膨大な量にのぼれば、サウンドエンジンからのキャプチャーが、Wwiseのパフォーマンスに影響することもあります。そこで、Advanced Profilerの、Profiler Settingsダイアログボックスを使い、生成する情報タイプを制限すると良いでしょう。情報の種類 (Information type) を1つ以上、選択から外すことで、以下が実現します。

- ネットワークの転送帯域幅を節約できる。
- Wwiseのメモリスペースを節約できる。
- データを演算しない分、ゲームのCPU時間を短縮できる。
- Wwiseでデータを処理せずグラフ化しない分、WwiseのCPU時間を短縮できる。

また、1回のキャプチャーセッションの、プロファイラ (Profiler) とゲームオブジェクトプロファイラ (Game Object Profiler) の、2つのレイアウトにある全

てのビューの内容をそのまま、プロファイリングセッションのファイルにダンプするのかどうかを、Wwiseで指定できます。このファイルのデータを、あとからWwiseにロードし直して、ゲームのオーディオを詳しく分析できます。



#### 注記

Wwiseは制限を超えた場合、古いキャプチャセッションを消去します。このため、さらに古いキャプチャセッションを保存したい場合は、プロファイリングセッションが終わる度に一番古いファイルの名前を変えるか、Number of sessions keptを増やす必要があります。

### キャプチャーする情報タイプを、指定するには:

#### 1. 以下のいずれかを行う:

- Advanced Profilerの、タイトルバーで、View Settingsアイコンをクリックする。
- メニューバーで、Project > Profiler Settingsをクリックする。
- Alt+Gを押す。

Profiler Settingsダイアログボックスが開く。

#### 2. 情報タイプを、以下から1つ以上選択する。

- **Plug-in Data:** 様々なプラグインの情報
- **Memory Data:** サウンドエンジンの、Memory Managerに登録されているメモリプールの情報
- **Stream Data:** サウンドエンジンが管理する、様々なストリームの情報
- **Voices Data:** サウンドエンジンが管理する様々なボイスの、それぞれの情報
- **Listener Data:** サウンドエンジンが管理する様々なリスナーの、それぞれの情報
- **Obstruction/Occlusion Data:** 様々なゲームオブジェクトに影響する、オブストラクション (Obstruction) やオクルージョン (Occlusion) の情報
- **Markers Notification Data:** 各種オーディオファイルの、マーカーノティフィケーション (Marker notifications) や、各種Music Segmentの、カスタムキューノティフィケーション (Custom cue notifications) の情報
- **Output Data:** アウトプットピークや、アウトプットDCオフセットの情報
- **SoundBanks:** メモリにロードしたSoundBankの情報
- **Loaded Media:** メモリにロードしたメディアの情報
- **Prepared Events:** PrepareEvent機能でプリペア (Prepare) した各種イベントの情報
- **Prepared Game Syncs:** PrepareGameSyncs機能でプリペア (Prepare) した各種ゲームシンク (Game syncs) の情報
- **Music Transitions:** ミュージックオブジェクトが切り替わる時に起きる、トランジションの情報
- **Streaming Device Data :** すべてのストリーミングデバイスからの情報
- **Meter :** バスとメータービューにメーターを駆動するための信号測定データの使用に関する情報

- **Auxiliary Sends Data** : サウンドエンジンによるAuxiliary Sendsそれぞれに関連する情報
- **API Calls** : WwiseのAPIへの呼び出しについての情報



3. Capture Logの下に、使用できるメモリの再大容量を入力する。メモリサイズは、最小10MB、最大1000MBである。

Game Profilerが使用するメモリは、Capture Log（キャプチャーログ）の指定メモリ容量を超えることはありません。

4. Number of sessions kept（保存セッション数）に数値を入力して、ディスクに保存するプロファイルセッション数を指定する。これ以上のファイルが存在する場合は、最古のファイルから削除される。

5. OKをクリックする。

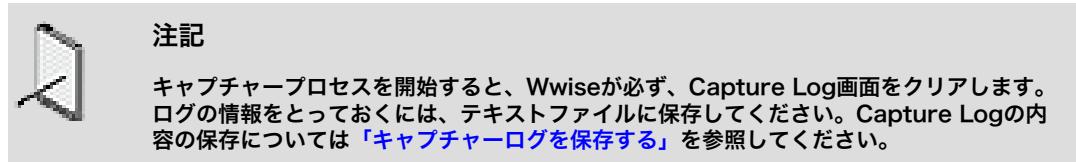
Game Profilerが、選択した情報タイプのデータだけを、キャプチャーする。

## 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログのフィルタを設定する」
- 「キャプチャーログをソートする」
- 「キャプチャーセッションから統計情報を収集する」
- 「キャプチャーログを保存する」
- 「以前のリモートキャプチャーセッションのデータをロードする」

## キャプチャープロセスの開始と停止

サウンドエンジンからの情報をキャプチャーし始めると、データが自動的に、Capture Log、Performance Monitor、そしてAdvanced Profilerに記録されます。キャプチャーされる情報をすぐに確認することも、問題が発生した時にキャプチャーを停止して調査することもできます。情報がキャプチャーされると同時にモニターしたい場合は、Follow Capture Time（キャプチャータイムを追う）オプションを選択して、Capture Logや、Performance Monitorを、常にキャプチャータイムと同期させます。



## キャプチャープロセスを開始するには:

1. 以下のいずれかを行う:

- ・メニューバーで、Layouts > Profilerをクリックする。
- ・F6を押す。

Profilerレイアウトが表示される。

2. データがキャプチャーされると同時にパフォーマンスをモニターするには、ツールバーで、Follow Capture Timeボタンをクリックする。

このオプションを選択すると、以下のように表示される。

- ・Capture Log画面が、キャプチャーされたデータに合わせて、自動的にスクロールしていく。
- ・Performance Monitor画面のタイムカーソルが、ゲームタイムカーソル(Game time cursor)と一緒に動く。

3. Capture Logでデータのキャプチャーを開始するには、ツールバーの、Start Captureボタンをクリックする。

4. キャプチャーを停止するには、Stop Captureをクリックする。

## 関連トピック

- ・「ローカルまたはリモートで、ゲーム機に接続する」
- ・「キャプチャーログのフィルタを設定する」
- ・「キャプチャーログをソートする」
- ・「キャプチャーセッションから統計情報を収集する」
- ・「キャプチャーログを保存する」
- ・「以前のリモートキャプチャーセッションのデータをロードする」
- ・「Performance Monitorを使った、モニタリングとトラブルシューティング」

## キャプチャーログのフィルタを設定する

Capture Logに表示される情報が、多すぎる場合もあります。このような時は、見たい情報だけを表示するように、Capture Logにフィルタ(Filter)を設定します。例えば、ある特定のゲームオブジェクトに関連する情報だけを表示させたり、Event関連の情報や、State関連の情報だけを表示させたりすることができます。

## Capture Logのフィルタを設定するには:

1. Capture Logで、Filterをクリックする。

Capture Log Filterダイアログボックスが開く。

2. Capture Logに表示させるオプションだけを選択して、キャプチャーログにフィルタを設定する。



### 注記

個々のオプションの内容を知るには、Capture Log Filterダイアログボックスの、Helpアイコンをクリックします。

3. OKをクリックする。

選択した基準に基づいて、Capture Logに選別（フィルタ）された内容が表示される。

## 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログをソートする」
- 「キャプチャーセッションから統計情報を収集する」
- 「キャプチャーログを保存する」
- 「以前のリモートキャプチャーセッションのデータをロードする」

### キャプチャーログをソートする

Capture Logにある全ての列ヘッダ（項目名）を、情報のソートに利用できます。例えば、表示内容を「Type（タイプ）」別、「Wwise Object（Wwiseオブジェクト）」別、「Game Object（ゲームオブジェクト）」別などにソートできます。



### Capture Log（キャプチャーログ）をソートするには:

1. Capture Logで、列のヘッダを1つクリックする。

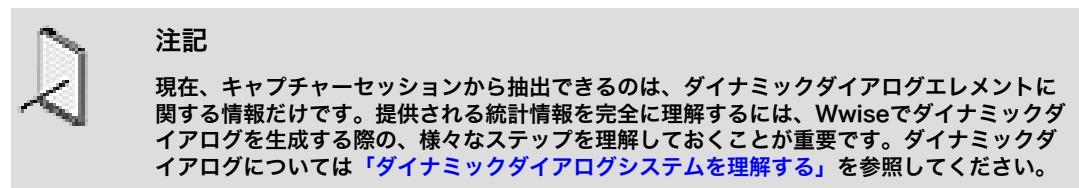
表示内容が、選択した列ヘッダに従ってソートされる。

## 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログのフィルタを設定する」
- 「キャプチャーセッションから統計情報を収集する」
- 「キャプチャーログを保存する」
- 「以前のリモートキャプチャーセッションのデータをロードする」

### キャプチャーセッションから統計情報を収集する

キャプチャーセッションの最中や終了後にいつでも、Wwiseプロジェクト、つまりゲームの特定オーディオエレメントに関する統計情報を確認するために、キャプチャーされた情報をWwiseで分析することができます。



Profiler Statisticsビューに、キャプチャーセッションで抽出した統計情報を表示されます。既定のクエリがいくつかあり、キャプチャーログ内にある、様々な情報サ

プセットの統計情報を提供します。例えば、キャプチャーセッション中に完結した (Resolved) パスはどれか、これらパスが完結したのは何回か、その結果オーディオが再生されたのは何回か、などを判定できます。また、フィルタを設定して情報の的を絞ったり、問題を見つけたり、対策が必要な場合にとるべきアクションを判断したりできます。

全てのアクティブなキャプチャーセッションから統計情報を抽出できるので、リモート接続したゲームからキャプチャーしたデータや、Wwise内でローカルにキャプチャーしたデータを使い、統計情報を収集できます。アクティブなキャプチャーセッションとは、Capture Logに情報が表示されているセッションのことです。

### キャプチャーセッションから、統計情報を収集するには:

1. ローカルまたはリモートで、キャプチャーセッションを開始する。
2. メニューバーで、Views > Profiler Statisticsをクリックする。

Profiler Statisticsビューが開く。

3. Queriesリストで、以下のクエリを1つ選択する。

- **Paths Used** - キャプチャーセッション中に完結して、Dynamic Sequence (ダイナミックシーケンス) リストに追加されたパスはどれか、またこれらパスが完結したのは何回か、そしてその結果オーディオが再生されたのは何回か、を表示する。
- **Paths Added** - キャプチャーセッション中にトリガーされたダイアログイベントを表示する。ダイアログイベントがトリガーされるた結果、Dynamic Sequence (ダイナミックシーケンス) リストに追加されたオーディオオブジェクトと、それが追加された回数を表示する。
- **Dialogue Events Distribution** - キャプチャーセッション中にトリガーされた様々なダイアログイベントの統計情報を表示する。具体的には、完結されたダイアログイベントの回数と、それが実際にオーディオオブジェクトの再生に結び付いた回数を比較表示する。

4. Runをクリックする。

クエリ結果が、Resultsペインに表示される。

5. 不要な情報があれば、フィルタオプションを1つ以上選択して非表示にする。検索の対象を絞ることで、何らかの対応が必要な時に、素早く判断できる。

### 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログのフィルタを設定する」
- 「キャプチャーログをソートする」
- 「キャプチャーログを保存する」
- 「以前のリモートキャプチャーセッションのデータをロードする」

### キャプチャーログを保存する

Wwiseでは、キャプチャープロセスを開始する前にCapture Logを毎回クリアするので、Capture Logの情報を維持するにはファイルに保存する必要がありま

す。ファイルはPROFファイル（デフォルト）またはTXTファイルとして保存します。PROFファイルをWwise Profilerレイアウトにロードするには、[Wwiseツールバー](#)のRemote...ボタンをクリックしてから、Connect To IPをクリックします。ゲームをプロファイルした直後のように、情報が表示されます。テキストファイルとして保存する場合は、Wwiseがキャプチャーログの情報だけを保存します。現在の列やソートはそのまま残ります。さらに情報のソートやフィルタを行いたい場合は、外部の表計算プログラムにテキストファイルをインポートできます。

### キャプチャーログを保存するには:

1. Capture Logで、Save Logをクリックする。  
Save Asウィンドウが開く。
2. ファイルを保存する場所まで移動する。
3. テキストボックス File nameに、ファイル名を入力する。拡張子PROFまたはTXTを選択する。
4. Saveをクリックする。

Capture Logの内容が、テキストファイルとして保存される。

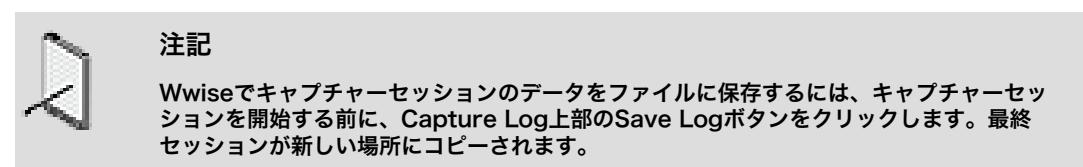
### 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログのフィルタを設定する」
- 「キャプチャーログをソートする」
- 「キャプチャーセッションから統計情報を収集する」
- 「以前のリモートキャプチャーセッションのデータをロードする」
- 「Performance Monitorを使った、モニタリングとトラブルシューティング」

### 以前のリモートキャプチャーセッションのデータをロードする

Wwiseは、ゲームの情報をキャプチャーしながら、その情報を、ProfilingSession.profというプロファイリングセッションファイルにダンプします。この情報をあとからWwiseに再びロードすれば、ゲームオーディオの細かい分析に活用できます。

新規プロファイリングセッションを開始すると、ファイルが新たに作成され、ファイル名のうしろに連番が付きます。保存するファイル数は、Profiler Settingsで指定します。



### 以前のキャプチャーセッションのデータをロードするには:

1. 以下のいずれかを行い、Profilerレイアウトに切り替える。
  - メニューバーで、Layouts > Profilerをクリックする。

- F6を押す。
2. ツールバーで、Remoteをクリックする。  
Remote Connectionsダイアログボックスが開く。
  3. Connect To Fileをクリックする。
  4. .profファイルを1つ選択して、Openをクリックする。
- このキャプチャーセッションで収集した全てのデータが、Profilerレイアウトや、Game Object Profilerレイアウトにロードされる。

**複数のプロファイリングセッションから、統計情報を計算するには:**

1. Viewメニューで、Profiler Statistics Viewを開く。
2. ツールバーで、Remoteボタンをクリックする。  
Remote Connectionsダイアログボックスが開く。
3. Connect To Fileボタンをクリックする。
4. 必要なPROFファイルを全て選択して、Openをクリックする。

選択した全てのキャプチャーセッションで収集したデータが処理され、処理結果が統計情報として、Profiling Statisticsビューに表示される。

## 関連トピック

- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログのフィルタを設定する」
- 「キャプチャーログをソートする」
- 「キャプチャーセッションから統計情報を収集する」
- 「キャプチャーログを保存する」
- 「Performance Monitorを使った、モニタリングとトラブルシューティング」

## Performance Monitorを使った、モニタリングとトラブルシューティング

Wwiseがサウンドエンジンと緊密にインテグレートされているため、ゲームプレイ中に、数多くの主要パフォーマンス指標 (Key performance indicators) をリアルタイムでモニターできます。また、Game Profilerを、Soundcaster、Game Simulator、その他のSoundFrameアプリケーションなどで利用できるので、ゲームに実装する前の、アイディア段階やプロトタイプ段階でも、パフォーマンスをモニターできます。

サウンドエンジンのアクティビティ情報をWwiseがキャプチャーすると、Performance Monitor画面に、各種パフォーマンスのグラフがリアルタイムで作成されます。また、グラフ内の実際の数値や率が、グラフの右側にある、Performance Data (パフォーマンスデータ) リストに表示されます。

それぞれのグラフを見れば、ゲームのオーディオが、プラットフォーム特有の制限など具体的な制約条件を守っていない箇所が、すぐに分かります。 Performance

Monitor、Capture Log、Advanced Profilerの3つの機能を組み合わせれば、問題が発生した際にトラブルシューティングできます。

Performance Monitorはカスタマイズできるので、Performance Monitor画面に表示させるパフォーマンス指標（Performance indicators）やカウンタ（Counters）を変更できます。Performance Monitorのカスタマイズについては「[Performance Monitorをカスタマイズする](#)」を参照してください。

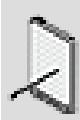


### Tip

データをキャプチャーしながらパフォーマンスをモニターすることも、キャプチャーセッションを実行して、あとから、発生した問題を確認するためにグラフをスクロールして調べることもできます。データをキャプチャーすると同時にパフォーマンスをモニターしたい場合は、Follow Capture Time（キャプチャータイムを追う）オプションを選択して、Capture Logや、Performance Monitorを、常にキャプチャータイムと同期させます。

## パフォーマンスをモニターするには:

- 外部のゲーム機、PC、SoundFrameアプリケーション、または稼働バージョンのGame Simulatorに接続する。



### 注記

Soundcasterを使って、モデルやプロトタイプをプロファイリングする場合は、様々なモジュールを再生する時に情報がキャプチャーされます。

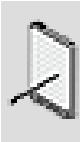
- 以下のいずれかを行う:

- メニューバーで、Layouts > Profilerをクリックする。
- F6を押す。

Profilerレイアウトが表示される。

- ツールバーで、Start Captureボタンをクリックする。

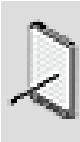
Wwiseが、対象プラットフォームのパフォーマンスを、サウンドエンジンからキャプチャーした情報に基づいて、グラフで表示します。



### 注記

塗りつぶされたブロック部分は、キャプチャーした情報が最低値または最大値を超えた箇所を示しています。グラフの上部にあるブロックは、最大値よりも高い数値を、下部にあるブロックは、最小値よりも低い数値を表します。

- Performance Monitorのタイムカーソル（白）を、パフォーマンス問題が発生したグラフ上の箇所までドラッグする。



### 注記

グラフィューをスクロールすると、Capture Logの表示箇所や、Performance DataペインやAdvanced Profilerに表示される情報が、自動的に更新されます。

- Capture Logに表示されるタイムカーソルのインジケータ（白円）を使って、サウンドエンジンが実行するイベントやアクションなどのアクティビティの

うち、どれが問題の原因かを調べる。または、Capture Logのエントリ（行）を、Shift-クリックして、このタイムスタンプ（Timestamp）に、全てのプロファイラービューを同期させる。

6. オブジェクトまたはイベントの内容やプロパティを変更するには、Capture Logのエントリをダブルクリックする。そのEventまたはオブジェクトを表示した、Event EditorまたはProperty Editorが開くので、必要に応じて調整する。
7. このアクティビティのパフォーマンスを、さらに詳しく見るには、Advanced Profilerにある、様々なタブに表示される情報を確認する。
8. Performance Monitorのグラフをテキストファイルとして保存するには、グラフ上を右クリックして、**Save All Counters to File**を選択する。キャプチャーの開始時点から現時点まで、200ミリ秒単位で記録した全てのカウンター（Counters）が、タブ区切りのテキストファイルに保存される。

## 関連トピック

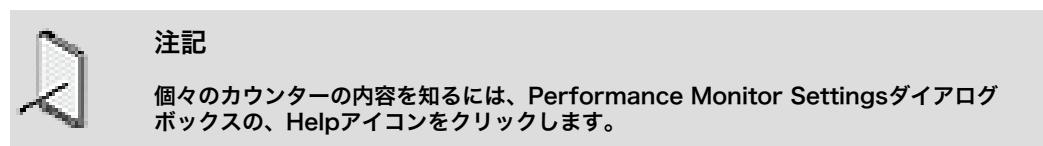
- 「Performance Monitorをカスタマイズする」
- 「キャプチャープロセスの開始と停止」
- 「キャプチャーログを保存する」

## Performance Monitorをカスタマイズする

複数の主要指標（Key indicators）を同時にモニターしたい場合もあれば、特定のエリアにフォーカスしたい場合もあるかもしれません。モニターセッションの目的に合わせて、Performance Monitorに表示するカウンターの種類を、その都度、追加したり削除したりできます。

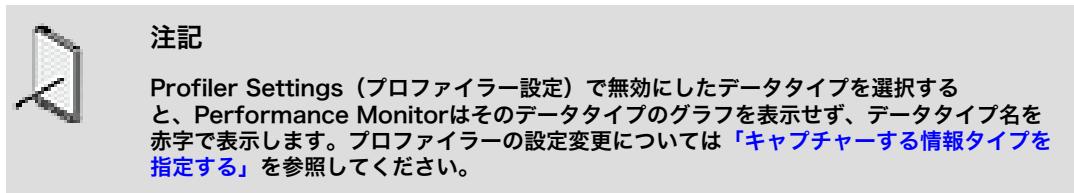
## Performance Monitorの表示内容を変更するには:

1. Performance Monitorのタイトルバーで、View Settingsアイコンをクリックする。  
Performance Monitor Settingsダイアログボックスが開く。
2. Show In Graph（グラフ表示）列で、Performance Monitorのグラフビューに表示させたいカウンターを、全て選択する。



3. Show In List（リスト表示）列で、Performance Dataリストに表示させたいカウンターを、全て選択する。
4. 選択したカウンターの、Min.（最低値）とMax.（最大値）をそれぞれ設定する。最小値と最大値で、そのカウンターのグラフの高さ（上下の幅）が定義される。
5. OKをクリックする。

Performance Monitorが更新され、選択したカウンターだけが表示される。



## Game Object Explorerで、オブジェクトやリスナーをトラッキングする

Game Object Explorer は、エミッター、リスナー、またはその両方としての役割を含めて、ゲームオブジェクトを学習するための出発点です。このビューでは、ゲーム内に登録されているすべてのゲームオブジェクトを見ることが可能で、Wwise がどのゲームオブジェクトを監視するかを制御できます。ウォッチ対象のゲームオブジェクトを選択すると、Game Sync Monitorに表示され、Game Object 3D Viewer、RTPCグラフ、Attenuation Editorにも表示されます。

Game Object Explorerでゲームオブジェクトを追跡するには、次の操作が必要です：

- 「ゲームオブジェクトのリストを理解する」
- 「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」

### ゲームオブジェクトのリストを理解する

Game Object Explorerの、Game Objectsタブに、現在のセッションでアクティブである、またはアクティブであったゲームオブジェクトが、全て表示されます。ゲームオブジェクトの作成と削除はオーディオプログラマーの担当で、作成された時にゲームオブジェクトの名前やIDを提供するのもプログラマーです。

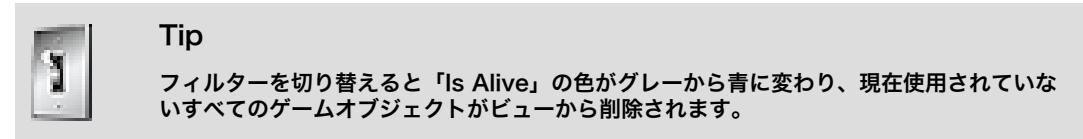
ゲームに接続すると、このリストが自動的に更新されます。

The screenshot shows the Game Object Explorer window with the 'Game Objects' tab selected. The table lists various game objects with columns for Name, ID, Last Registered, Last Unregistered, and Is Alive.

Name	ID	Last Registered	Last Unregistered	Is Alive
AKPhysicsTriggerWaterImpact (phy_water...)	19136937	00:13:35.296	-	✓
AMBIENCE	4294967...	00:13:35.296	-	✓
Boy	196665	00:13:35.296	-	✓
CollisionRect2D	44761606	00:13:35.296	-	✓
CollisionRect2D	786709	00:13:35.296	-	✓
door	26411081	00:13:35.296	-	✓
pelvis	196646	00:13:35.296	-	✓
Sound	2622011	00:13:35.296	-	✓

現在アクティブなゲームオブジェクトは「Is Alive (生きている)」と表示されます。ゲームオブジェクトは、登録 (Registered) されると「Alive」となり、登録が解除 (Unregistered) されると「Alive」でなくなります。例えば、サー

ストパー・ソン・シューターでは、破壊されたゲームオブジェクトや、ゲームのジオメトリにおいてプレイヤーが検知できる範囲より遠くなつたゲームオブジェクトは、「Alive」状態が終了するかもしれません。一方、リアルタイムストラテジー・ゲームでは、マップ上に存在する全てのゲームオブジェクトが、破壊されるまで「Alive」状態であるかもしれません。



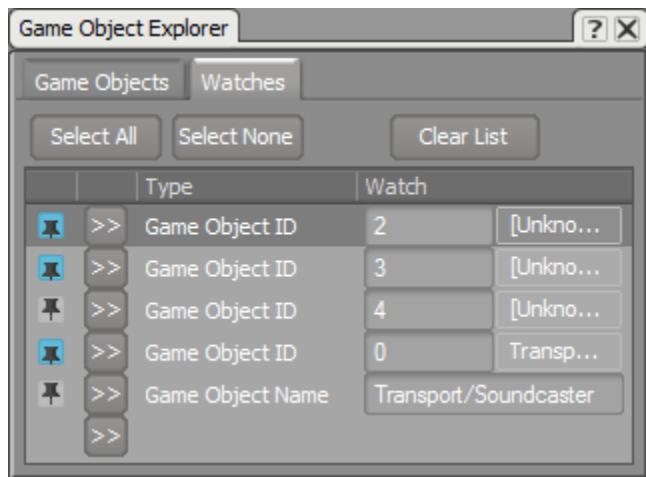
Game Objectsタブのほとんどの列は(他の[tables](#)のように)情報のみを表示します。しかし、**Mute**と**Solo**列は、選択されたゲームオブジェクトとそれぞれのWwiseオブジェクトの音をデザイナーが分離できる該当ボタンを提供します。

## 関連トピック

- 「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」
- 「ゲームオブジェクトウォッチを設定する」
- 「ウォッチの編集と削除」

### ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする

ウォッチ(Watch / 観察)機能は、ゲームのゲームオブジェクトをモニターする個別のプロセスです。1つのゲームオブジェクトにWatchを1つ割り当てると、そのアイテムのステータス詳細が、定期的に表示されます。設定したウォッチによって取得した情報は、Game Object 3D Viewer、RTPCタブのグラフ、Attenuation Editor、そしてGame Sync Monitorで使われます。ウォッチするゲームオブジェクトがなければ、Game Object Profilerのビューにデータが一切表示されません。



ゲームオブジェクトウォッチの対象は、名前(Name)またはIDで指定します。1つのゲームで同じ名前の付いたゲームオブジェクトが複数あることがあります、オブジェクトIDは固有です。このため、1つのウォッチで複数の同じオブジェクト(例えば全てのドローン)をモニターしたい時は、名前で指定したウォッチを作成します。一方、ある特定のオブジェクトだけをモニターしたい時は、IDで指定したウォッチを作成します。

ウォッチを使った作業には、以下のタスクが伴います。

- ・「ゲームオブジェクトウォッチを設定する」
- ・「ウォッチの編集と削除」
- ・「表示するウォッチを選択する」

### ゲームオブジェクトウォッチを設定する

ゲームオブジェクトを観察するウォッチ (Watch) を設定するには、以下の2つの方法があります。

- ・Game Objects (ゲームオブジェクト) リストで設定。
- ・Watches (ウォッチ) リストで設定。

### Game Objectsリストで、ゲームオブジェクトウォッチを設定するには:

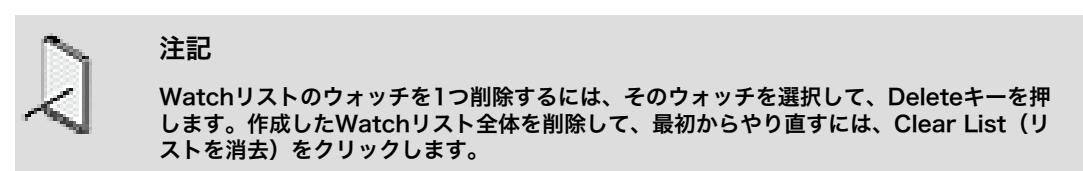
1. Game Object Explorerの、Game Objectsタブで、ゲームオブジェクトを右クリックして、Add Game Object to Game Object Watch List (By IDまたはBy Name)を選択する。

このゲームオブジェクトの固有IDまたはオブジェクト名を観察するウォッチが、Watchesリストに追加される。

### Watchesリストで、ゲームオブジェクトウォッチを設定するには:

1. Game Object Explorerで、Watchesタブに切り替える。
2. セレクタボタン (>>) をクリックして、以下のオプションを1つ選択する。
  - ・Game Object Name (ゲームオブジェクト名) は、オブジェクト名を指定して、ゲームオブジェクトのウォッチを作成する場合。
  - ・Game Object ID (ゲームオブジェクトID) は、オブジェクトIDを指定して、ゲームオブジェクトのウォッチを作成する場合。
  - ・Global Game Object : グローバルゲームオブジェクトのウォッチを設定。(主に、グローバルに使用する値やバスに適用する値をウォッチする際に便利。)
3. Watch列に、ウォッチの観察対象の名前 (Name) 、またはIDを入力して、Enterキーを押す。

ウォッチが設定され、Watchリストに追加される。



### 関連トピック

- ・「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」
- ・「ウォッチの編集と削除」

- 「表示するウォッチを選択する」

#### ウォッチの編集と削除

作成したウォッチ (Watch) は、簡単に編集や削除ができます。なお、編集や削除をしても、元のゲームオブジェクトには影響せず、そのデータを収集するWatchだけが影響を受けます。

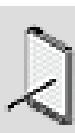
#### ウォッチを編集するには:

- Game Object Explorerで、Watchesタブに切り替える。
- Watch (ウォッチ) のType (タイプ) を切り替えるには、セレクタボタン (>>) をクリックして、別のタイプを選択する。
- ウォッチの内容表記を変更するには、Watches列をクリックして、新しい名前またはIDを入力する。

#### ウォッチを削除するには:

- Game Object Explorerで、Watchesタブに切り替える。
- ウォッチを1つ選択して、Deleteキーを押す。

このウォッチが、Watchリストから削除される。



#### 注記

作成したWatchリスト全体を削除して、最初からやり直すには、Clear List (リストを消去) をクリックします。

#### 関連トピック

- 「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」
- 「ゲームオブジェクトウォッチを設定する」
- 「表示するウォッチを選択する」

#### 表示するウォッチを選択する

デフォルトで全てのWatchにピンが付くので、これらのウォッチに関連するゲームオブジェクトやリスナーが、Game Object 3D Viewer、RTPCのグラフとAttenuation Editorのグラフに自動的に表示されます。もしこれらのビューが混み合っているようであれば、いくつかのWatchの選択を外すこともできます。表示するオブジェクト数を減らすとCPU負荷も下がるので、Game Object 3D Viewerのパフォーマンスも改善されます。



#### 注記

これは、Game Sync Monitorに表示されるゲームオブジェクトに影響しません。Watchリストにある全てのゲームオブジェクトがGame Sync Monitorに表示されます。

#### Watchesリストのウォッチをピン付け・ピン付け解除するには:

- Game Object ExplorerのWatchesタブで、ウォッチ名の左のピンアイコンをクリックする。

**Watchesリストのウォッチを全てピン付けするには:**

1. Game Object Explorerの、Watchesタブで、Select All（全て選択）をクリックする。

**Watchesリストのウォッチを全てピン付け解除するには:**

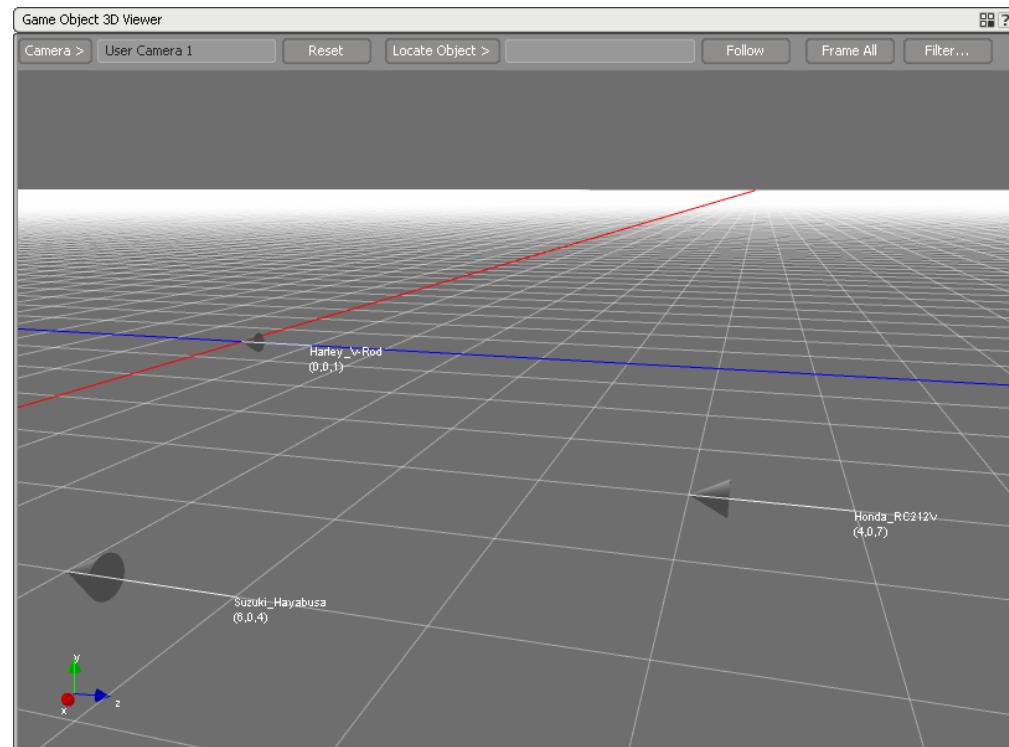
1. Game Object Explorerの、Watchesタブで、Select None（全て非選択）をクリックする。

### 関連トピック

- ・「ウォッチ機能で、ゲームオブジェクトやリスナーをモニターする」
- ・「ウォッチの編集と削除」

## Game Object 3D Viewerで、オブジェクトを観察する

Game Object 3D Viewerには、ゲームオブジェクトに設定した様々なWatchのデータが、グラフィカルに表示されます。このビューアを使用すると、オブジェクトがエミッタやリスナとしての役割を果たす際に、オブジェクトがどのように相互にやりとりし、相対的に移動するかをリアルタイムで確認できます。



Game Object 3D Viewerでは、ウォッチ (Watch) 中のゲームオブジェクトやリスナーがそれぞれ、矢印で表示されます。Viewer Settings (ビューアー設定) で選択したオプションに基づいて、ゲームオブジェクトに適用中のコーン (Cone) や半径 (Radius) も、観察できます。

**Tip**

Game Object 3D Viewerの操作機能の多くは、キーボードショートカットを使って実行できます。全てのショートカットを見るには[付録C #####](#)を参照してください。

## Game Object 3D Viewerの表示データを指定する

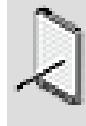
Game Object 3D Viewerの設定を変えて、表示するデータ数を減らすことも増やすこともできます。当然、選択したデータのタイプ (Type) が多いほど、ビューの見え方が複雑になります。

### Game Object 3D Viewerで表示するデータを指定するには:

1. Game Object 3D Viewerで、Settings (設定) アイコンをクリックする。

Game Object 3D Viewer Settingsダイアログボックスが開く。

2. Game Object 3D Viewerに表示させたいオプションを全てを選択する。

**注記**

個々のオプションの内容を知るには、Game Object 3D Viewer Settingsダイアログボックスの、Helpアイコンをクリックします。

3. Closeをクリックする。

選択した条件に基づいて、Game Object 3D Viewerのデータが表示される。

## 関連トピック

- ・「[カメラを使う](#)」
- ・「[ウォッチのフィルタを設定する](#)」

### カメラを使う

3Dビューアーのカメラ (Camera) をマウスで操作して動かすことで、3D空間の中を自由に移動できます。このようにして、カメラを動かしながら、ウォッチ (Watch) 対象のゲームオブジェクトやリストナーが、互いに作用し合う様子を観察できます。

### カメラのフォーカスをあてるゲームオブジェクトを選択するには:

1. Game Object 3D Viewerで、Locate Object (オブジェクトを見つける) をクリックして、リストから、カメラのフォーカスを向けるオブジェクトを1つ選択する。

選択したオブジェクトがカメラ視野の中央に表示される。

### カメラをゲームオブジェクトにアサインするには:

1. Game Object 3D Viewerの、Locate Object (オブジェクトを見つける) リストで、ゲームオブジェクトを1つ選択する。

## 2. Follow (追う) をクリックする。

カメラが、キャプチャー中に、動くオブジェクトを追う。

Game Object 3D Viewerに、全てのゲームオブジェクトを表示させるには：

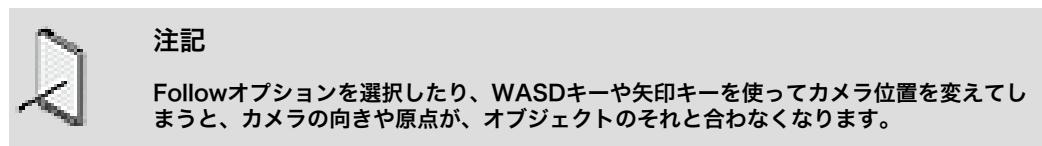
1. Game Object 3D Viewerで、Frame All (全てフレームに入れる) をクリックする。

選択してある全てのゲームオブジェクトがビューアーに入るように、ビューアーが調整される。

ゲームオブジェクトの視点から、ゲームを見るには：

1. Game Object 3D ViewerのCameraリストから、カメラの原点となるウォッチ (Watch) 中のオブジェクトを選択します。

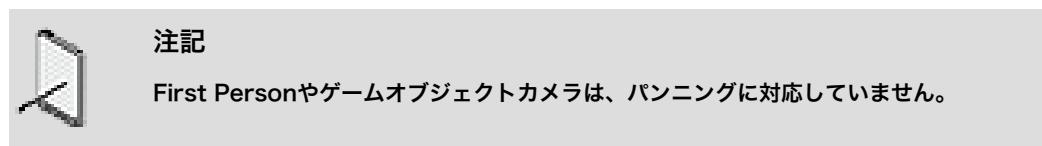
選択したオブジェクトが、カメラ視野の原点となり、カメラがオブジェクトの角度と同じ角度になる。



カメラをパンニングさせるには:

1. Game Object 3D Viewer内で、Ctrlを押しながら、クリックしてドラッグする。

カメラが上下左右にパンする。



カメラをズームさせるには:

1. Game Object 3D Viewer内で、右クリックしてドラッグする。

カメラがズームイン・ズームアウトする。

カメラを回転させるには:

1. Game Object 3D Viewerの、Camera (カメラ) リストで、User Camera 1 (ユーザーカメラ1) 、またはUser Camera 2 (ユーザーカメラ2) を選択する。
2. Game Object 3D Viewer内で、クリックしてドラッグする。

カメラが、センターポイントを中心に回転する。



### Tip

カメラを切り替えても、どちらのカメラ設定も維持されます。

**ファーストパーソンシューターのように、カメラを操作するには:**

1. Game Object 3D Viewerの、Cameraリストで、First Personカメラ、またはゲームオブジェクトカメラの1つを選択する。

これで、以下が可能になる。

- WASDキーや、矢印キーを押して、カメラを動かせる。
- クリック＆ドラッグで、カメラの向きを変えられる。
- Shiftを押し続けて、速く動かせる。
- 右クリックしてドラッグするか、マウスのセンターホイールを使って、ズームイン・ズームアウトできる。

**カメラを、デフォルト位置にリセットするには:**

1. Game Object 3D Viewerで、Reset（リセット）をクリックする。

カメラが、デフォルトの位置に戻る。

### 関連トピック

- 「Game Object 3D Viewerの表示データを指定する」
- 「ウォッチのフィルタを設定する」

### ウォッチのフィルタを設定する

デフォルトで、ウォッチ（Watch）している全てのゲームオブジェクトが、Game Object 3D Viewerに表示されます。ウォッチしている一部のゲームオブジェクトだけを表示させるには、表示内容のフィルタを設定します。

**Game Object 3D Viewerで、表示内容のフィルタを設定するには:**

1. Game Object 3D Viewerで、Filter（フィルタ）をクリックする。

Game Object 3D Viewer Filterダイアログボックスが開く。

2. ウォッチを表示または非表示にするには、ピンアイコンを選択するか、選択を外す。



### 注記

ウォッチ中のオブジェクトのうち、ピン付き（Pinned）のものだけが、Game Object 3D Viewerに表示されます。

3. 全てのウォッチを表示させるには、Pin All（全てにピン）をクリックする。

4. 全てのウォッチを非表示にするには、Unpin All（全てのピン解除）をクリックする。
5. Closeをクリックして、Game Object 3D Viewerに戻る。

#### 関連トピック

- 「[Game Object 3D Viewerの表示データを指定する](#)」
- 「[カメラを使う](#)」

## Game Sync Monitorで、ゲームシンクを評価する

Game Sync Monitorは、ウォッチ (Watch) している様々なゲームオブジェクトに適用されるRTPCの数値をトラッキングする、プロファイリングツールです。ウォッチ機能がRTPC値を受信すると同時に、リアルタイムでグラフ表示されます。

Game Sync Monitorでゲームシンク (Game syncs) を評価する作業には、以下の手順が伴います。

- 「[Game Sync Monitorに、ゲームシンクを追加する](#)」
- 「[モニター結果のフィルタを設定する](#)」

#### Game Sync Monitorに、ゲームシンクを追加する

Game Sync Monitorは、ゲームでウォッチ (Watch) 中の各種オブジェクトのために使われるRTPCであればどれでも、その関連するデータを表示します。ただし、データのグラフを見るには、このビューで表示させるゲームシンクを追加する必要があります。

#### Game Sync Monitorに、RTPCを追加するには:

1. Game Sync Monitorの、Game Syncs Watchesペインで、Addをクリックする。

Project Explorer - Browserダイアログボックスが開く。

2. 追加するRTPCを1つ選択して、OKをクリックする。

このRTPCが、Game Sync Monitorに追加される。

#### 関連トピック

- 「[Game Sync Monitorで、ゲームシンクを評価する](#)」
- 「[モニター結果のフィルタを設定する](#)」

#### モニター結果のフィルタを設定する

Game Sync Monitorに、RTPCを追加すると、ウォッチ (Watch) しているゲームオブジェクトのうち、このゲームシンク (Game sync) の影響を受けるものは全て、デフォルトでグラフビューに表示されます。当然、あるゲームシンクが、多数のオブジェクトに影響する場合は、見え方が複雑になります。見え方を単純

化するには、一部のゲームオブジェクトとゲームシンクのグラフの、ピンを外して（Unpin）、非表示にします。

## Game Sync Monitorで、グラフを表示・非表示するには

1. Game Sync Monitorで、ゲームシンクの横にあるピン（Pin）アイコンをクリックして、グラフを表示または非表示にする。

### 関連トピック

- ・「[Game Sync Monitorで、ゲームシンクを評価する](#)」
- ・「[Game Sync Monitorに、ゲームシンクを追加する](#)」

## プロファイリングに関する、コツとベストプラクティス

Game Profiler機能を使う前に、以下のセクションに目を通して、ゲームの様々な面をプロファイリングしたり、トラブルシューティングしたりするための、コツやベストプラクティスを参照してください。

### ゲームプロファイラのパフォーマンス

転送データが膨大な量にのぼれば、サウンドエンジンからのキャプチャが、Wwiseのパフォーマンスに影響することもあります。そこで、Advanced Profilerの、Profiler Settingsダイアログボックスを使い、生成する情報タイプを制限すると良いでしょう。情報の種類（Information type）を1つ以上、選択から外すことで、以下が実現します。

- ・ネットワークの転送帯域幅を節約できる。
- ・Wwiseのメモリースペースを節約できる。
- ・データを演算しない分、ゲームのCPU時間を短縮できる。
- ・Wwiseでデータを処理せずグラフ化しない分、WwiseのCPU時間を短縮できる。

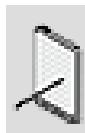
### キャプチャ中にパフォーマンスをモニターする

サウンドエンジンからデータをキャプチャーしている最中にパフォーマンスをモニターするには、Follow Capture Time（キャプチャタイムを追う）オプションを選択しておく必要があります。このオプションを選択しないと、しばらく経つと、Profilerの全てのビューが更新されず情報を全くキャプチャーしていないかのように、見えます。もし、Follow Capture Timeオプションを選択するのを忘れた場合は、Performance Monitorのタイムカーソル（白）をドラッグして、グラフ上の別の位置まで動かせば、キャプチャーされたデータをスクロールして確認できます。タイムカーソルをドラッグしていくと、Profilerの他のビューも自動的に更新されます。

### リモート接続の問題のトラブルシューティング

- ・Wwiseをゲームに接続したい時に、Remote Connectionsダイアログボックスに、そのコンピュータまたはゲーム機が表示されない場合は、以下を確認してください。

- Communicationsモジュールが、ゲームで正しく初期化されているか。詳しくはWwise SDKドキュメンテーションの"Initializing Communications"を参照。
- プロジェクト設定の、Game Discovery Broadcast Portに指定されているポート番号が、ゲームでコミュニケーションの初期化に使うAkCommSettings構造で指定されているポート番号と同じか。詳しくは「[ネットワークポートの指定](#)」やWwise SDKドキュメンテーションの"Initializing Communications"を参照。
- コンピュータまたはゲーム機が、ネットワークに正しく接続されているか。一部のゲーム機には複数のネットワークアダプタがあり、デバッグ用や通常のネットワークトラブル用のものもあるので、注意すること。Wwise側から再度、接続を試みる前に、全てを接続しておくこと。
- コンピュータまたはゲーム機が、Wwiseの稼働するコンピュータと同じサブネットにあるか。
- アクティブなファイアウォールが接続をブロックしている。Wwiseの稼働するコンピュータにファイアウォールが設定されていれば、それを無効にして、ファイアウォールが接続問題の原因かどうかを確認する。もし接続できるようであれば、ファイアウォールの例外リストに、Wwiseアプリケーションを登録してみる（登録方法は、ファイアウォールのドキュメンテーションを参照）。Wwiseとのコミュニケーションに使うポートを変更する必要があれば（ダイナミックポートを固定ポートに変更する、など）、「[ネットワークポートの指定](#)」や、Wwise SDKドキュメンテーションの「通信の初期化」を参照。
- リモート接続するコンピュータまたはゲーム機のIPアドレスが分かれれば、Remote Connectionsダイアログボックスで、Connect to IPをクリックして、IPアドレスを入力する。これでゲームに接続できれば、ブロックされているのがリモートコンピュータまたはゲーム機からのブロードキャストだけということになる。一度、コンピュータやゲーム機を接続すると、以後はRemote Connectionsダイアログボックスの、Historyタブに、それが自動的に表示されるので、次に接続する時は、IPアドレスを入力しなくてよい。
- サウンドエンジンが、デバグまたはプロファイルの状態となっていないか。リースビルドのサウンドエンジンには、接続できない。
- 非常に大きいプロジェクトからゲームに接続しようとする時に、接続画面でハングアップするようであれば、以下を試してみてください。
- プロジェクトを開いて現在のSoundcasterセッションとMixing Deskセッションを空にした上でTransport Controlにロードされたものを削除してから、Synchronize modified objectsの選択を外してゲームに接続してみる。



### 注記

なぜこれで接続できる可能性があるかというと、次の様な背景がある。Wwiseをゲームに接続すると、Wwiseはゲームに情報をプッシュして、Wwiseに表示される内容が、ゲームの内容とシンクしていることを確認する。しかしSynchronize modified objectsを選択していないと、Wwiseは何でもプッシュしないようになる。かわりにTransport Controlで選択されているものと、SoundcasterやMixing Deskのアクティブセッションにあるものだけをプッシュする。ゲームに接続しようする時に、WwiseプロジェクトのSoundcasterやMixing Deskのアクティブセッションに大量のデータが入っていると、問題が発生する可能性がある。

またリモート接続でSynchronize modified objectsオプションを選択すること 자체も、Wwiseセッションの全ての変更をリモートインスタンスで再生するものと同期するので、この問題に寄与している可能性がある。これに時間がかかりすぎるかもしれない。このような問題が発生する場合は、このオプションの選択を外してから再接続してみる。

---

## 第35章 SoundBankの管理

概要 .....	665
ゲームにSoundBankをロードする方法を理解する .....	667
SoundBankの構築 .....	672
SoundBankの管理 .....	689
SoundBankのカスタム属性を設定する .....	692
プロジェクトの各種SoundBankの生成 .....	700
CopyStreamedFilesツールの利用 .....	704
SoundBank管理の戦略 .....	705
SoundBankに関する、コツとベストプラクティス .....	718

## 概要

ゲームのオーディオコンポーネントやモーションコンポーネントを効率的に管理するために、Wwiseは全てのオーディオやモーションをSoundBankに入れます。SoundBankは基本的に、データ（オーディオデータやモーションデータ）、またはメディア、またはその両方が格納されたファイルです。SoundBankは、ゲーム中の特定ポイントでプラットフォームのメモリにロードされます。必要なものだけをロードするため、メディアに使用するメモリ量を各プラットフォームで最適化できます。SoundBankの中身はそれまでの作業の成果物であり、ゲームに入る最終的なコンテンツが入っています。

Wwiseには、以下の2種類のバンクがあります。

- **Initialization/Init bank** - プロジェクトの一般的な情報を全て入れた特別なバンクで、バス階層、State、Switch、RTPC、環境エフェクトなどの情報が含まれる。該当する場合はAudio Device ShareSetsも含まれる。

WwiseがSoundBankを生成すると、初期化バンクが自動的に作成される。通常、初期化バンクはゲームの最初に1回ロードされ、ゲームプレイ中にプロジェクトの全般情報に簡単にアクセスできるようにする。ゲームを開始する時に最初にロードする必要があり、これがロードされないと、他のバンク（コンテンツ）をロードできない場合もある。初期化バンクのデフォルト名は、Init.bnk。

- **SoundBank (SoundBank)** - イベントデータや、サウンド、ミュージック、モーションの各種構造データ、オーディオファイルなどが共に入ったファイル。一般的に、SoundBankは初期化バンクと異なり、ゲームの最中に適宜ロードやアンロードされるため、プラットフォームのメモリが効率的に使われる。

プラットフォームは全て違いますが、Wwiseで簡単にSoundBankを各種プラットフォームに対応させることができます。プラットフォームごとのSoundBankを全て、同時に生成できます。さらにWwiseは、各プラットフォームの制限に準拠していることを確認するために、SoundBank用のトラブルシューティングのツールも提供しています。

1つのゲームに対して、Wwiseで使うプロジェクトは1件だけです。非常に大きいプロジェクトに複数の人が関わっている場合は、プロジェクトを複数のWork Unitに分割することができます。Work Unitについては「[プロジェクトをWork Unitで分割](#)」を参照してください。

## WwiseのSoundBank用のビュー

効率的に作業するために、WwiseにはSoundBank (SoundBank) 用のレイアウトが1つあります。プロジェクトの様々なSoundBankを作成、管理、生成するのに必要な全てのビューがこのレイアウトにあり、SoundBank Manager、SoundBank Editor、Project Explorer、Event Viewerなどの各種ビューが表示されます。



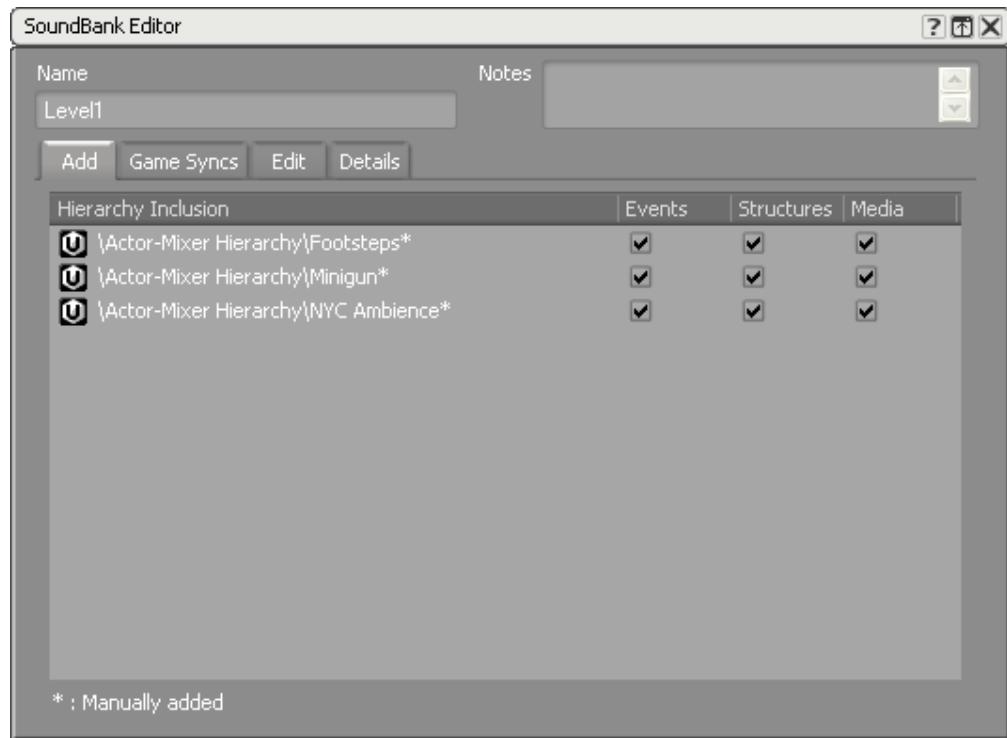
### Tip

デフォルトで、WwiseでSoundBankレイアウトに切り替えるにはF7を押します。

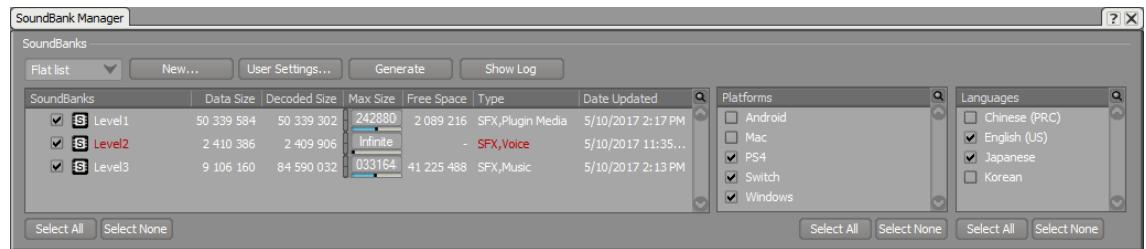
SoundBank Editorは、SoundBankを構築する場所で、ここで、各SoundBankへ、イベント、サウンド構造、ミュージック構造、モーション構造、オーディオファイルなどを追加していきます。SoundBank Editorを開くには、Project Explorer、またはSoundBank Managerで、SoundBankを1つ、ダブルクリックします。

SoundBank Editorにあるタブは、以下の4つです。

- **Add** - SoundBankに追加した実際のイベント、オブジェクト階層、Work Unit、フォルダなどだけが、表示される。一緒に自動的にSoundBankに追加される様々な子オブジェクトは、Editタブだけに表示される。Addタブで、SoundBankに入れる情報やメディアのタイプを、階層エレメントごとに決める。
- **Game Syncs** - Addタブに表示される各種イベントやサウンド構造に関連する、様々なState、Switch、Triggerのリストが表示される。このタブで、そのSoundBankから除外するゲームシンクを選択できる。あるゲームシンクを除外すると、これに関連する全てのサウンド構造やメディアファイルも除外される。
- **Edit** - Addタブに表示される様々なプロジェクトエレメントに関連する、全てのイベント、オブジェクト構造、メディアファイル、そして全ての子オブジェクトの詳しいリストが表示される。また、SoundBankに入っているメディアファイルの詳細情報として、サンプルレート、オーディオフォーマット、ファイルサイズなどが表示される。このタブで提供される情報を編集して、プラットフォームごとの制約や条件に合わせて、個別のファイルのコンバージョン設定(Conversion settings)を簡単に調節できる。リストをランゲージ別やオブジェクトタイプ別にフィルタして、そのSoundBankから除外するプロジェクトエレメントの選択を外すこともできる。
- **Details** - 表示中のSoundBankに入っている様々なエレメントのサイズ(Size)の詳細情報や、不足しているファイルや置換されたファイルが、表示される。



プロジェクトの開発中にいつでも、SoundBank Managerを使ってSoundBankを生成できます。SoundBank Managerには、作成してあるSoundBankのリストと、それぞれのタイプ (Type) やサイズ (Data Size) などの基本情報が表示されます。また、プラットフォーム (Platforms) やランゲージ (Languages) のリストもあり、ここで選択したプラットフォームやランゲージ用に、SoundBankが生成されます。



## ゲームにSoundBankをロードする方法を理解する

WwiseでどのようにSoundBankを構築して生成するのかを学ぶ前に、SoundBankの情報をゲームがロードして管理する方法を理解することが大事です。どの手法を選ぶかは、開発するゲームの種類、対応させるプラットフォームの種類、そしてプロジェクトチーム側で設定する制限など、様々な条件に基づいて決めます。

できるだけ柔軟に対応して、ほとんどの種類のゲームの要件を満たせるように、Wwiseには、ゲームにオーディオやモーションをロードする方式がいくつかあり、以下がその一部です:

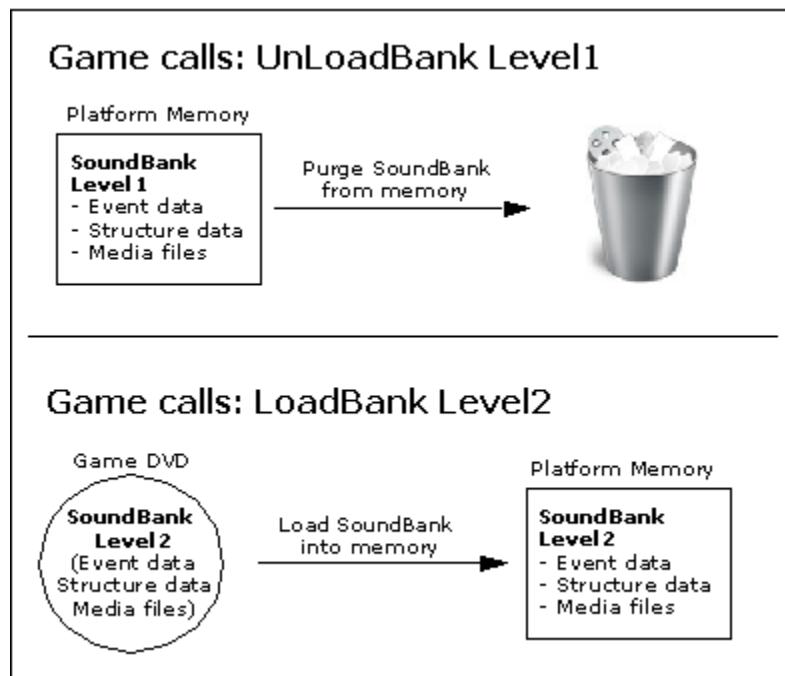
- ・「フルSoundBankをロードする」

- 「プリペアSoundBank（全コンテンツ）（Prepare Bank (All content)方式）」
- 「アクションイベントをプリペアする（Prepare Event方式）」

### フルSoundBankをロードする

従来のSoundBankロード方式はEventデータ、構造データ、メディアが混在するSoundBankを使いました。これらSoundBankのコンテンツ（中身）をそのまま、特定のタイミングでゲームにロードしたりアンロードしたりして、イベントデータや関連するメディアが、トリガーされた時に再生できるように準備しておきます。

従来からあるこの方式で、2つのSoundBankを作成して、プレイヤーがゲーム中にレベル1からレベル2に移る時に、それぞれのSoundBankをプラットフォームメモリに対してロード（Load）したりアンロード（Unload）したりする様子を、下図に示します。



ここでは、SoundBankにある全てのデータやメディアを、一度にメモリにロードするので、必要な時に備えてどのデータやメディアも再生できる状態になるだけでなく、ゲームプレイ中に実行するディスクシークが非常に少ないため、他のディスク負荷の高いタスクを行うディスクスペースを確保できます。

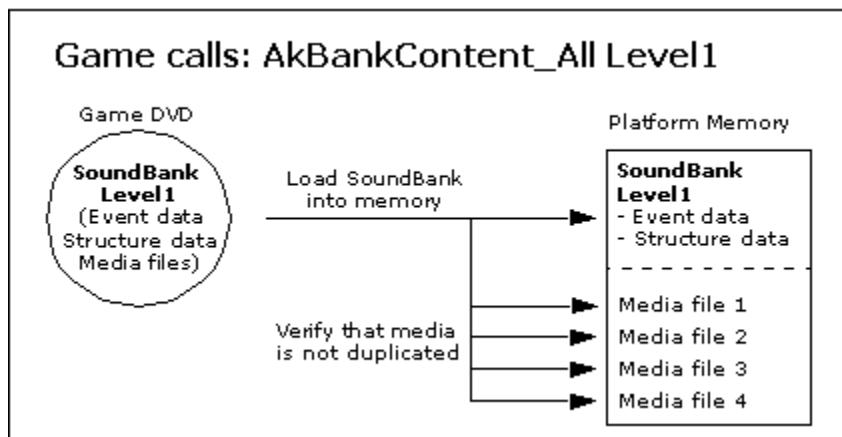
この方式の主な欠点は、SoundBankがロードされている状態では、これらのSoundBankが終始、かなりのメモリスペースをとるため、規模が大きく洗練されたゲームの場合は、柔軟に対応しづらくなることです。また、この方式では、SoundBankの中のメディアファイルのうち、既にメモリにロードされているものがあるかを確認しないまま、コンテンツ（中身）を全てロードします。つまり、1つのメディアファイルが、メモリに何回もロードされてしまう可能性があります。しかし、欠点があるものの、この従来の方式はピンボールゲームのように、常に全てのデータやメディアを使用可能な状態にしなくてはならない場合には、適しています。

## プリペアSoundBank (全コンテンツ) (Prepare Bank (All content)方式)

LoadBank()メカニズムの欠点を一部、解消するために、バンクをロードせずに、AkBankContent\_All()を使って、SoundBankをプリペア (Prepare / 用意) できます。この方式では、各SoundBankに様々な種類のコンテンツ (イベント、構造データ、メディアファイル) が入っていても、メディアファイルをそのままロードせず、「プリペアイベント (Prepare Event / イベントの用意)」メカニズムを使い全てのメディアをメモリにロードします。このメカニズムでメディアをロードする時、Wwiseはまず、各メディアファイルが既にメモリに存在するかどうかを調べてから、ロードします。メモリに同じメディアファイルがロードされることを防ぎ、メモリ使用率を最小限に抑えます。

この方式を使うと、メモリを節約できるほか、ディスクアクセスが、必ず順番に行われます。PrepareEvent()を使ってイベントを1つずつプリペア (用意) すると、ランダムなディスクシークが発生する可能性が高くなりますが、これを回避できます。

「プリペアバンク (全コンテンツ)」のメカニズムを使って、プラットフォームメモリにメタデータやコンテンツをロードする様子を、下図に示します。



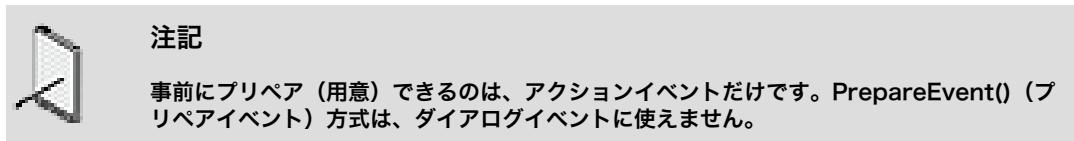
### Vorbisオフライン解凍

SoundBankにVorbis暗号化データをロードする時にPreparation\_LoadAndDecodeのプリペアタイプを使えばこれらのメディアファイルを圧縮されていないPCMファイルに解凍できます。その結果SoundBankのサイズが大きくなりますが、イベントがメディアをコールした時に解凍せずにすぐに利用できます。

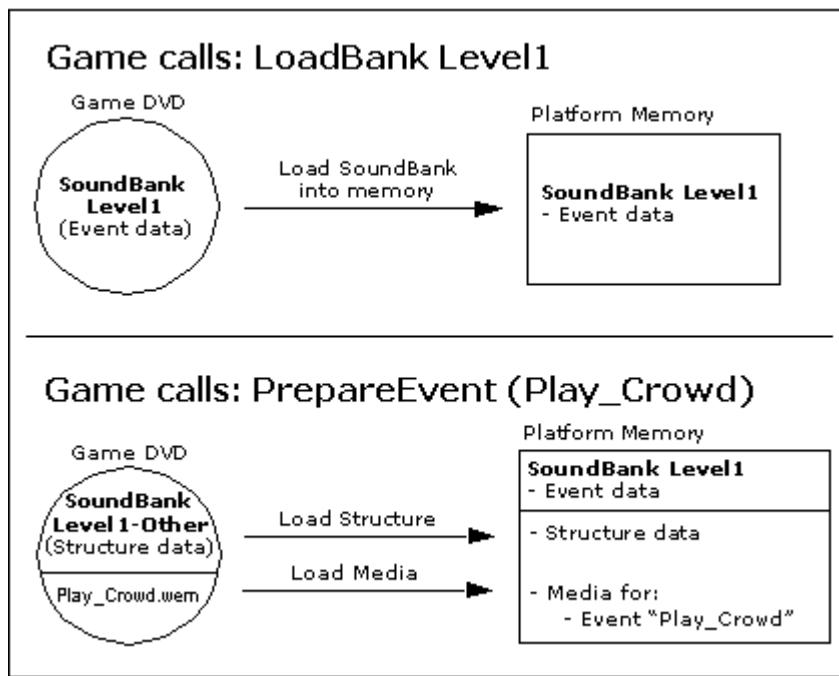
### アクションイベントをプリペアする (Prepare Event方式)

PrepareEvent() 方式は、必要な時だけメディアをロードするダイナミックな方式です。この方式を使うにはアクションイベントのメタデータをSoundBankに入れ、関連メディアファイルをファイルシステムからアクセスできる状態にする必要があります。対応する構造メタデータはイベントと同じSoundBankに入れるか、別のSoundBankに入れます。この方式では、アクションイベントのメタデータが入ったSoundBankは、LoadBank() (バンクをロード) を使って、メモリにロード

され維持されます。アクションイベントはゲームに呼び出される前にサウンドエンジンでプリペアします。イベントをプリペアすることで、ファイルシステムにある全ての関連メディアファイルがロードされ、関連する構造メタデータが既にロードされていない場合はこれもバンクからロードされます。アクションイベントが不要になると、プリペアを解除すれば対応メディアファイルがメモリからページされます。メモリ使用率を最小限に抑え、メモリ内でメディアファイルが重複するのを防ぐために、Wwiseはメディアファイルをロードする前に、必ずチェックを行い、そのメディアファイルがまだメモリに入っていないことを確認します。



イベントを事前にプリペア（用意）することで、必要なメディアファイルだけがメモリにロードされる様子を、下図に示します。

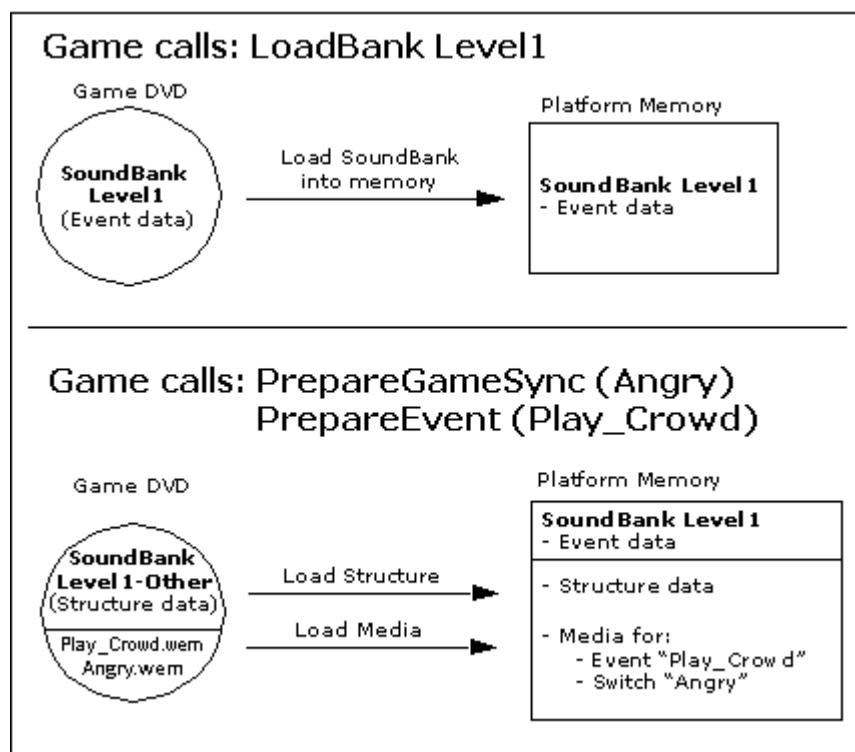


メタデータ（構造と呼ぶこともある）がイベントと同じバンクに格納されていない場合に、他のSoundBankに保存された対応データを見つける手段が必要となります。探せるように、Wwiseは、他のSoundBankに保存されている関連コンテンツのレファレンス（参照情報）を入れます。他のSoundBankを参照するには、名前またはIDを使います。なお、サウンドエンジンでSoundBank名を使う場合は、Project Settingsダイアログボックスの、SoundBanksタブで、**Use SoundBank Names**（SoundBank名を使用）を選択します。IDを使う場合は、この選択を外します。

SoundBankが参照するメディアは、ディスク上にファイルとして保存するか、その他の方法でローレベルIOが扱える状態（File Package内など）でなければなりません。

この方式は全般的にメモリ使用が非常に効率的ですが、ディスクシークが増えるので、既にディスクからストリーミングしているファイル数が多い状況には向かない可能性があります。また、ゲームで複数のStateやSwitchを使う場合は、不要なメディアファイルもメモリにロードされる可能性もあります。例えば、ゲーム中の観衆 (Crowd) の声として、観衆の満足度やエネルギーレベルに合わせて複数のサウンドがあるとしても、ゲームの特定ゾーンで、「Angry (憤慨)」 Switchに該当するサウンドだけが使われるのであれば、他のサウンドをメモリにロードしても意味がありません。このような問題を避けるため、StateやSwitchも、プリペアする段階を設ければ、プリペア (用意) されたStateやSwitchに関連するメディアファイルだけをロードできます。

Switchだけを事前にプリペア (用意) して、メモリにロードされるメディアの数をさらに抑えた例を、下図に示します。



ゲームシンクを事前にプリペア (用意) することでメモリ使用を最適化できますが、同時に、メモリにメディアをロードする速さが遅くなるので、注意が必要です。プリペアされたゲームシンクに該当するサウンドを探すため、サウンドエンジンがディスクの中をシークする必要があるので、読み込み時間が長くなります。

この第2の方式では、必要なオーディオファイルだけをダイナミックにロードするので、多数のサウンドが存在する巨大なゾーンやレベルなどの状況や、メモリにイベントデータ、構造データ、メディアを保存するスペースが非常に限られた状況でも、柔軟に対処できます。

ご覧の通り、どの方式にも長所と短所があります。ゲームの様々なシナリオ、要件、制限などに基づいて、どの方式を採用するか決めてください。1つの方式を選んだにしろ、いくつかを組み合せたにしろ、戦略が決まれば、SoundBankに中身を入れて調整を開始できます。



### 注記

ここで紹介した、[LLoadBank](#)、[AkBankContent\\_All](#)、[PrepareEvent](#)、[PrepareGameSyncs](#)、[AkBankContent\\_Struct](#) の各種機能は、Wwise APIを通して使用できます。バンクのロードや、イベントやゲームシンクのプリペア（用意）については、Wwise SDKの、[Integrating Banks](#) のセクションを参照してください。

SoundBank管理に利用できる様々な方式の詳細については「[SoundBank管理の戦略](#)」を参照してください。

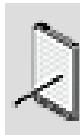
## SoundBankの構築

作成するSoundBank（SoundBank）の数を検討する時は、デザイナーやプログラマーがゲーム全体を分析して、ゲームに実装された全てのWwiseイベントのリストを作成するのが一般的です。リストを確認しながら、ゲームに必要な各種SoundBankのサイズと数を判断します。次に、選択したSoundBank方式や、ゲームのキャラクター、オブジェクト、ゾーン、レベルなどに基づいて、様々なイベント、オブジェクト構造、メディアなどを、複数のSoundBankに分けて整理します。

Wwiseでは、メモリにロードできるSoundBank数に制限はありません。また、SoundBank自体のサイズにも、制限ありません。プロジェクトごとに条件が違うので、何が一番合うかを決めてください。

SoundBankの構築には、以下のタスクが伴います：

- ・「[SoundBankを作成する](#)」
- ・「[SoundBankに中身を入れる](#)」
- ・「[SoundBankのコンテンツを管理する](#)」
- ・「[プロジェクトの各種SoundBankの生成](#)」



### 注記

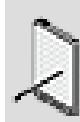
SoundBankのサイズを制限する唯一の条件は、システムまたはゲーム機の使用可能なメモリです。

### SoundBankを作成する

SoundBankは、Wwiseでマニュアル操作で1つずつ作成するか、イベントをゲームに実装する時に使用する3Dアプリケーションやレベルエディタから生成される定義ファイル（Definition file）を1つインポートして、同時に複数のSoundBankを作成します。定義ファイルのインポートについては「[定義ファイルをインポートして、SoundBankの中身を入れる](#)」を参照してください。

SoundBankをマニュアルで作成できるのは、Wwiseの以下の2つのエリアです。

- ・Project Explorerで、[SoundBankを作成する](#)には:
- ・SoundBank Managerで、[SoundBankを作成する](#)には:



## 注記

SoundBankを作成すると、Wwiseが自動的に「New\_SoundBank\_X」という名前を付けます。SoundBank名は、内容が分かる名前に変更できます。SoundBank名を変更する予定がある場合には、後の段階で変更すると追加のプログラミングが必要となる可能性があるので、開発プロセスの早い段階で行ってください。

## Project Explorerで、SoundBankを作成するには:

1. 以下のいずれかを行い、SoundBankレイアウトに切り替える。
  - メニューバーで、Layouts > SoundBankをクリックする。
  - F7を押す。
2. Project Explorerで、SoundBanksタブに切り替える。
3. 以下のいずれかを行う:
  - 仮想フォルダまたはWork Unitを選択して、Project Explorerツールバーの、SoundBankアイコンをクリックする。
  - 仮想フォルダまたはWork Unitを右クリックして、ショートカットメニューで、New Child > SoundBankを選択する。

Project Explorerで、新規SoundBankがハイライト表示される。

4. SoundBankの新しい名前を入力して、Enterを押す。

SoundBankが1つ作成され、SoundBankリストに追加される。



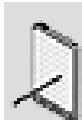
## 注記

SoundBank名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

5. テキストボックスMax Size（最大サイズ）に、このSoundBankに割り当てる、ゲーム中メモリの最大サイズを入力する。

## SoundBank Managerで、SoundBankを作成するには:

1. 以下のいずれかを行い、SoundBankレイアウトに切り替える。
    - メニューバーで、Layouts > SoundBankをクリックする。
    - F7を押す。
  2. SoundBank Managerで、Newボタンをクリックする。
- New SoundBankダイアログボックスが開く。
3. SoundBankの階層で、作成するSoundBankの保存先となるWork Unitを選択する。
  4. Nameフィールドで、デフォルトのSoundBank名を、新しいSoundBankに適した名前に置き換える。



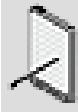
## 注記

SoundBank名に使えるのは、英数字（アクセント記号なし）とアンダースコアだけです。また、名前は必ず文字またはアンダースコアで始めます。

## 5. OKをクリックする。

SoundBankが1つ作成され、SoundBankリストに追加される。

## 6. テキストボックスMax Size（最大サイズ）に、このSoundBankに割り当てる、ゲーム中メモリの最大サイズを入力する。



### 注記

Workgroupプラグインの使用中に、チェックアウトしていないWork Unitに新しいSoundBankを作成した場合は、プロジェクトを保存しようとした時に初めて、Work Unitをチェックアウトするかどうか、Wwiseが確認します。

## 関連トピック

- 「[SoundBankに中身を入れる](#)」
- 「[SoundBankのコンテンツを管理する](#)」
- 「[SoundBankの詳細をモニターする](#)」
- 「[プロジェクトの各種SoundBankの生成](#)」

## SoundBankに中身を入れる

作成するゲームの種類や、選択したSoundBank戦略によって、SoundBankに入れる情報やメディアの種類が異なります。どの戦略を選択したかによって、SoundBankのサイズや内容も決まります。SoundBankには以下に挙げる種類のプロジェクトエレメントを組み合わせて入れることができます:

- Event data
- Sound, music, or motion structure data
- Media files

SoundBankに中身を入れるには、以下の2つの方式があります:

- 「[定義ファイルをインポートして、SoundBankの中身を入れる](#)」 - 定義ファイル (Definition file) をインポートする方式。
- 「[マニュアル操作で、SoundBankの中身を入れる](#)」 - Project Explorerから、プロジェクトエレメントであるイベント、オブジェクト構造、Work Unitなどをドラッグして、選択したSoundBankに入れ的方式。

## 定義ファイルをインポートして、SoundBankの中身を入れる

イベントをゲームに実装する時に使用する3Dアプリケーションやレベルエディタから生成される定義ファイル (Definition file) をWwiseにインポートすることで、SoundBankを作成して中身を入れることができます。定義ファイルとは、ゲームにある全てのイベントを、SoundBank別に並べた、タブ区切りのテキストファイルのことです。定義ファイルには、必ずSoundBank名と、該当するイベント名が、タブ区切りであります。Wwiseが読み込み可能な形式の定義ファイルの例を、以下に示します。

```
SB1#"Event_01"
SB2#"Event_02"
SB2#"Event_03"
SB3#"Event_04"
```

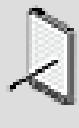
## SoundBankに含める・除外するエレメントを、キーワードで指定する

定義ファイルに、特別なキーワード（Keywords）を入れて、そのSoundBankに含める（Included）、または除外する（Excluded）プロジェクトエレメントの種類を定義することができます。使用できるキーワードを、以下に示します。

- **Event** - SoundBankに、イベント（Event）情報を含めることを指定するキーワード。
- **Structure** - SoundBankに、サウンド、ミュージック、モーションの、構造（Structure）情報を含めることを指定するキーワード。
- **Media** - SoundBankに、メディアファイル（Media files）を含めることを指定するキーワード。
- **-GameSyncExclusion**（ゲームシンクの除外） - SoundBankから除外するゲームシンク（Game Sync）を1つ指定するキーワード。ゲームシンクを1つ除外すると、これに関連する全てのサウンド構造やメディアファイルも、除外される。キーワード「-GameSyncExclusion」は、必ず以下のいずれかのキーワードと組み合わせて使う。
  - **State** - SoundBankから除外するState（State）を1つ指定するキーワードで、指定したStateに関連する全てのオブジェクトやメディアファイルも、除外される。キーワード「State」の後に必ず、除外するStateやState Group名（State group name）とState名（State name）を、タブ区切りで入れる。
  - **Switch** - SoundBankから除外するSwitch（Switch）を1つ指定するキーワードで、指定したSwitchに関連する全てのオブジェクトやメディアファイルも、除外される。キーワード「Switch」の後に必ず、除外するSwitch Group名（Switch group name）とSwitch名（Switch name）を、タブ区切りで入れる。
  - **Trigger** - SoundBankから除外するTriggerを1つ指定するキーワードで、指定したTriggerに関連する全てのオブジェクトやメディアファイルも、除外される。キーワード「Trigger」の後に必ず、除外するTrigger名（Trigger name）を入れる。
- **-DialogueEvent** - SoundBankに含めるダイアログイベントを1つ指定するキーワード。キーワード「-DialogueEvent」の後に必ず、含めるダイアログイベントの名前、GUID、またはショートIDを入れる。また、キーワードの後に、他のキーワード「Event」や「Structure」や「Media」を1つまたは複数入れて、SoundBankに含めることを指定できる。
- **EffectShareset** - SoundBankに含めるエフェクトシェアセットを1つ指定するキーワード。キーワードの後に必ずEffectシェアセットの名前、GUID、またはショートIDを入れる。また、キーワードの後に、他のキーワード「Structure」や「Media」を1つまたは複数入れて、SoundBankに含めることを指定できる。
- **-AuxBus** - SoundBankに含めるAuxiliary Bus（Aux Bus）を1つ指定するキーワード。キーワード「-DialogueEvent」の後に必ず、含めるバスの名前、GUID、またはショートIDを入れる。また、キーワードの後に、他のキーワード「Structure」や「Media」を1つまたは複数入れて、SoundBankに含めることを指定できる。

```
SB1#"Event_01"
SB1#"Event_02"#Structure
SB1#"Event_03"#Media#Structure
SB1#"Event_04"#Event#Media#Structure
```

上図では、1番目のSoundBank「My\_SoundBank\_Normal」に、Event\_01に関連する全てのイベントデータ、構造データ、メディアファイルを入れます。2番目のSoundBank「My\_SoundBank\_EventandStructure」には、Event\_02に関連するイベントデータと構造データだけを入れます。3番目のSoundBank「My\_SoundBank\_Media」には、Event\_03に関連するメディアファイルだけを入れます。

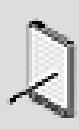


### 注記

定義ファイル (Definition file) に含まれるイベントは全て、既にWwiseで作成してある必要があります。作成されていないイベントがあれば、Import Definition Log (定義インポートログ) に「Event Missing (イベントがない)」として表示されます。

キーワード「-GameSyncExclusion (ゲームシンクの除外)」を使う場合は、除外するゲームシンクごとに改行して、新しいエントリを作成する必要があります。1つのゲームシンクを除外すると、これに関連する全てのオブジェクト構造やメディアファイルも除外されます。定義ファイルに、「-GameSyncExclusionキーワード」を記入した例を、下図に示します。（→が、タブを表す。）

```
SB1#-GameSyncExclusion#State#StateGroupName#StateName
SB1#-GameSyncExclusion#Switch#SwitchGroupName#StateName
SB1#-GameSyncExclusion#Trigger#Trigger
```



### 注記

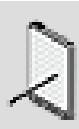
ゲームシンク (StateやState Group、State、Switch Group、Switch、Triggerなど) の名前に、クオテーションマーク ("") を付ける必要はありません。

```
SB1#-EffectShareSet#"effect1"#Media
SB1#-EffectShareSet#"effect2"#Structure
SB1#-EffectShareSet#"effect3"#Media#Structure
```

## イベントやエフェクトを、文字列でなく、IDで指定する

イベントの名前に文字列を使っていないゲームの場合は、定義ファイル (Definition file) の中でイベントを定義するために、以下のいずれかのシステムを使います。

- 16進数 (Hexadecimal)
- 10進数 (Decimal)



### 注記

16進数や10進数は、定義ファイルの中でイベントを指定するために使えますが、ゲームシンク除外 (Game sync exclusions) のState、Switch、Triggerの指定には使えません。

定義ファイルでイベントを指定するために使える3つのシステムの例を、下図に示します。（→が、タブを表す。）

```
SB1#"Event_01"
SB1#26507443
SB1#0x19478B3
```



### Tip

SoundBank定義ファイル (SoundBank definition file) を使って、既にゲームに実装されたイベント、未実装のイベント、そしてWwiseでまだ未作成のイベントを、確認できます。

す。プログラマーにゲームからイベントのリストを生成してもらい、Wwiseにその定義ファイルをインポートします。ログファイル (Log file) の情報を使い、ゲームの各種イベントと、Wwiseで作成したものを、照らし合わせます。

## 定義ファイルをインポートして、SoundBankを作成するには:

1. Project Explorerで、SoundBanksタブに切り替える。
2. 作成するSoundBankの保存先となるWork Unitを、右クリックする。
3. ショートカットメニューで、Import SoundBank Definitionを選択する。

Openダイアログボックスが開く。

4. 定義ファイルが保存されている場所まで、移動する。
5. Openをクリックする。

Import Definition Logダイアログボックスが開く。

6. ログのインポートアクティビティを確認する。インポートアクティビティとして、以下のいずれかが表示される。
  - Inclusion Added (新たに追加) - 新しいイベントまたはエフェクトが、既存SoundBankに追加された。
  - SoundBank Created (SoundBank作成) - 新規SoundBankが作成された。
  - Inclusion Removed (削除) - イベントまたはエフェクトが、既存SoundBankから削除された。
  - Event Missing (イベントなし) - イベントが1つ、プロジェクトにもう存在しないか、まだ作成されていない。
  - Effect Missing (エフェクトなし) - エフェクトが1つ、プロジェクトにもう存在しないか、まだ作成されていない。
  - Exclusion Added (除外設定) - ゲームシンクが1つ、既存SoundBankから除外された。
  - Exclusion Deleted (除外無効) - 除外されたゲームシンクが1つ、既存SoundBankに戻された。
  - Exclusion Missing (除外対象なし) - ゲームシンクが1つ、プロジェクトにもう存在しないか、まだ作成されていないため、除外 (Exclusion) リストでそのゲームシンクの除外を設定できなかった、またはその除外設定を削除できなかった。
  - No Change Detected (変更検知できず) - インポートしたSoundBankが、Wwiseに既にあるものと、同一である。

7. Closeをクリックする。

め定義ファイルで定義された全てのSoundBankが、Wwiseで作成され、中に指定したイベント、オブジェクト構造、メディアが入る。

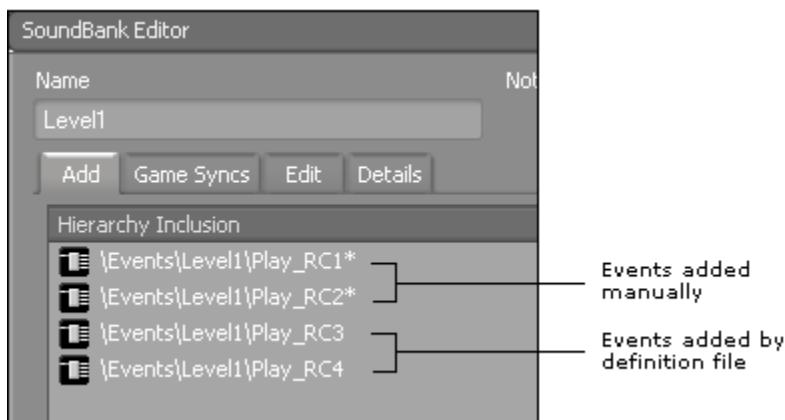
## 関連トピック

- 「SoundBankを作成する」
- 「SoundBankに、プロジェクトエレメントを含める・除外する」
- 「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」

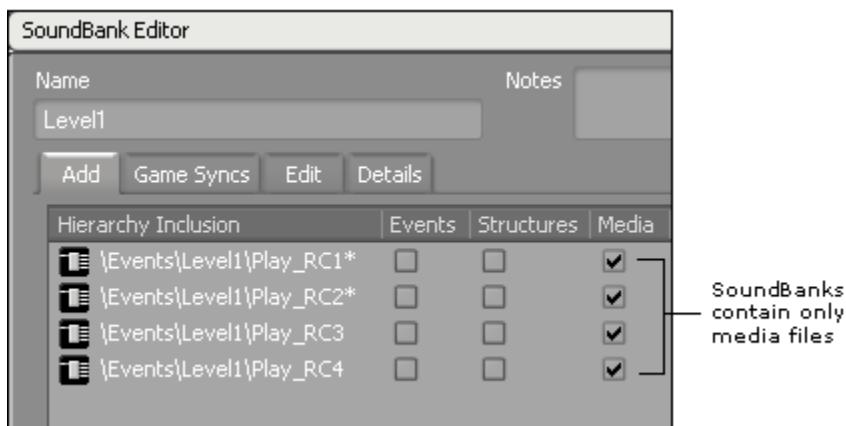
- 「SoundBank内のエレメントを検索する」
- 「SoundBankのエレメントのリストに、フィルタを設定する」
- 「SoundBankから、プロジェクトエレメントを削除する」
- 「SoundBankの詳細をモニターする」
- 「プロジェクトの各種SoundBankの生成」

### マニュアル操作で、SoundBankの中身を入れる

定義ファイル (Definition file) を1つインポートして、全てのSoundBankが作成できたら、イベント、ゲームシンク、オブジェクトなどを、マニュアル操作で追加したり除外したりすることで、SoundBankごとに中身を調整します。マニュアル操作でSoundBankに追加または除外したプロジェクトエレメントが分かるように、SoundBank Editorの、Addタブと、Game Syncsタブで、その項目の横に\* (アステリスカ) が表示されます。



SoundBankに1つのプロジェクトエレメントを追加すると、それに関連する全てのイベント (Events)、オブジェクト構造 (Structures)、メディアファイル (Media) なども、自動的にSoundBankに追加されます。しかし、SoundBankに入れるプロジェクトエレメントのタイプを限定して、SoundBankの中身を調節することができます。例えば、あるSoundBankにメディアだけを入れたい場合などです。



作業効率を上げるために、Wwiseでは、Project Explorerから、SoundBank Editorに、構造、Work Unit、またはフォルダをそのままドラッグして入れることができます。これらのプロジェクトエレメントに子オブジェクトが入っている場合は、子オブジェクトも自動的にSoundBankに追加されますが、これらが表示されるのは、Editタブだけです。

SoundBankに親オブジェクト、Work Unit、またはフォルダが追加されると、元のプロジェクトエレメントとのアクティブルリンクが維持されます。例えば、あるサウンドバンクにイベントのワークユニットが1つ入っているとします。次に、このワークユニットにイベントを2つ追加すると、2つのイベントも自動的にサウンドバンクに追加されます。このSoundBankを、SoundBank Editorにロードすると、Addタブでは最初のイベントWork Unitだけが表示されますが、Editタブでは、Work Unit内の全てのイベントが表示され、追加した2つのイベントも入っています。Wwiseでは、このリンクを維持することで、プロジェクトに加えた最新の変更点が、必ずSoundBankに含まれるようにします。



#### 注記

プロジェクトからアンロードされたプロジェクトエレメントが、SoundBankに含まれている場合は、SoundBank Editorの、Addタブで、このプロジェクトエレメントが黄色で表示されます。

### マニュアル操作で、SoundBankの中身を入れるには:

1. 以下のいずれかを行い、SoundBanksレイアウトに切り替える。
  - メニューバーで、Layouts > SoundBankをクリックする。
  - F7を押す。
2. SoundBank ManagerにあるSoundBankを1つダブルクリックして、SoundBank Editorにロードする。
3. Project Explorerから、SoundBank Editorの、Addタブに、以下のエレメントをドラッグする。

Actor-Mixers、Containers、Sounds、モーションオブジェクト、Musicオブジェクト。

Folders

Events

Work Unit

プロジェクトエレメントの階層全体が、関連する全てのイベント、オブジェクト構造、メディアファイルと共に、SoundBankに自動的に追加される。次の作業として、SoundBankに含めるプロジェクトエレメントのタイプを決めて、SoundBankの中身をさらに絞り込む。

### 関連トピック

- 「[SoundBankを作成する](#)」
- 「[SoundBankに、プロジェクトエレメントを含める・除外する](#)」

- ・「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- ・「SoundBank内のエレメントを検索する」
- ・「SoundBankのエレメントのリストに、フィルタを設定する」
- ・「SoundBankから、プロジェクトエレメントを削除する」
- ・「SoundBankの詳細をモニターする」
- ・「プロジェクトの各種SoundBankの生成」

## SoundBankのコンテンツを管理する

様々なSoundBankを作成して、コンテンツ（中身）を入れることができたら、次にSoundBankにいくつかのエレメントを含めたり（Include）、除外したり（Exclude）、削除したりして、内容を調整する必要が、おそらく出てくるでしょう。SoundBankに様々なプロジェクトエレメントを入れることができるので、Wwiseでは、プロジェクトエレメントを含めたり除外したりする様々な手段を提供するほか、オブジェクトを簡単に見つけるための検索ツールと、2種類のフィルタ設定を備えています。

SoundBankのコンテンツ管理には、以下のタスクが伴います:

- ・「SoundBankに、プロジェクトエレメントを含める・除外する」
- ・「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- ・「SoundBank内のエレメントを検索する」
- ・「SoundBankのエレメントのリストに、フィルタを設定する」
- ・「プロジェクトエレメントを、別のSoundBankに移動する」
- ・「SoundBankから、プロジェクトエレメントを削除する」

## SoundBankに、プロジェクトエレメントを含める・除外する

SoundBankに、イベントやサウンド構造を追加すると、Wwiseが自動的に、それらに関連する全てのイベント、サウンド構造、オーディオファイルも、SoundBankに含めます（Include）。ゲームのSoundBank戦略によっては、SoundBankから全てのイベント、または全てのサウンド構造、または全てのメディアファイルを、除外（Exclude）するとも考えられます。

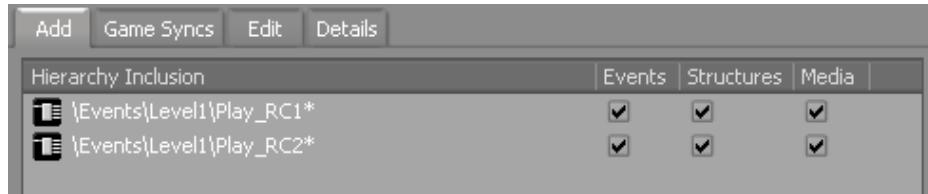
## SoundBankのエレメントを含める（Include）・除外する（Exclude）の活用例

例えば、小さいプロジェクトに、以下のプロジェクトエレメントが入っているとします。

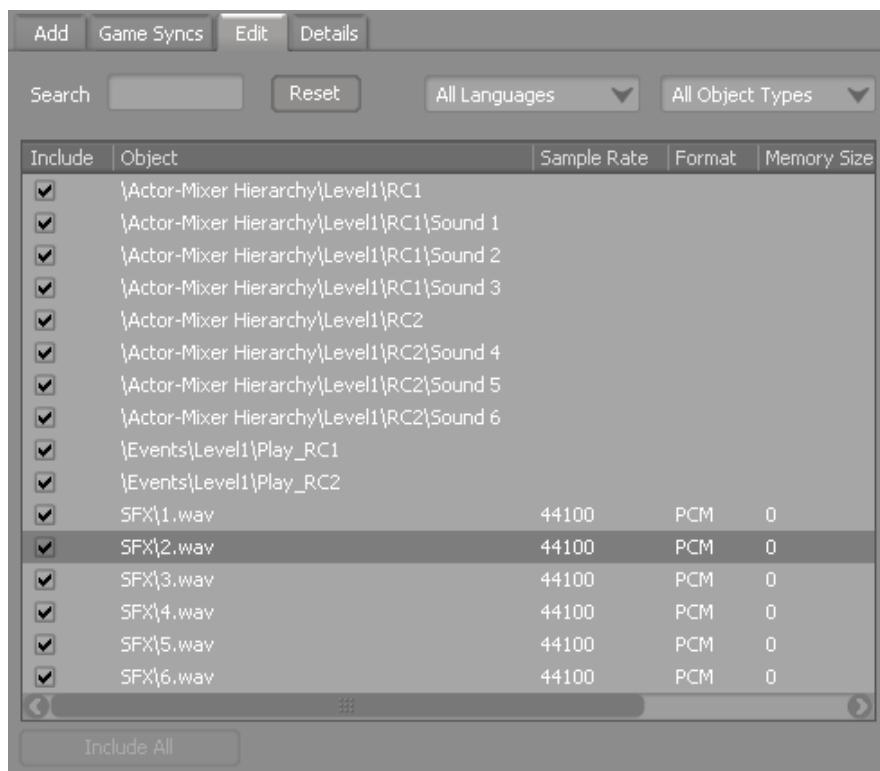
- ・Random Container 2個
- ・Six sound objects 6個
- ・Six audio files 6個
- ・Two play Events 2個

ここでは、プロジェクトの全イベント（Events）を1つのSoundBankに入れて、メディアファイル（Media files）に関しては、呼ばれる（Call）前にイベントをプリペア（用意）することで、メディアファイルをダイナミックにロードする計画で

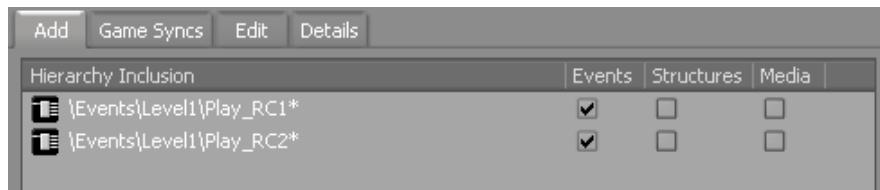
す。この場合、SoundBankを1つ作成して、SoundBank Editorの、Addタブに、2個のイベント（Play\_RC1、Play\_RC2）をドロップして入れます。



デフォルトで、2個のRandom Container（RC1、RC2）、6個のサウンドオブジェクト（Sound 1～Sound 6）、6個のオーディオファイル（.wav）が全て、一緒にSoundBankに含まれます。このSoundBankの全てのコンテンツが、SoundBank Editorの、Editタブに表示されます。

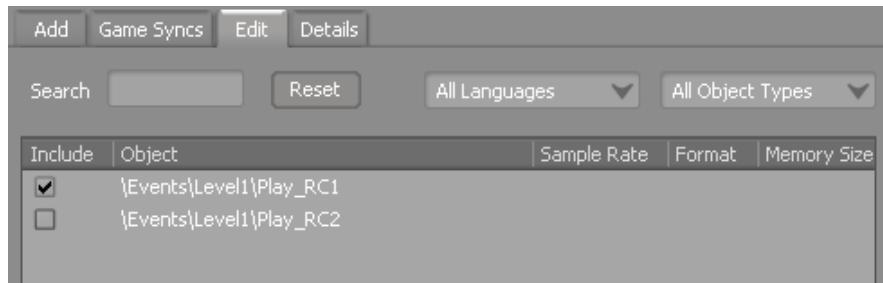


このSoundBankに含めるものをイベント（Events）だけにするには、Addタブに戻り、Structures（構造）列と、Media（メディア）列のチェックボックスを全て、クリアします。



これで、このSoundBankに入っている（Inclusion）のが、2個のイベントのデータだけになります。このSoundBankをさらに細かく調整するには、SoundBank

Editorの、Editタブで、除外するエレメントを選択します。例えば、このSoundBankに、イベントを片方だけ入れたい場合は、入れたくない方を除外します。

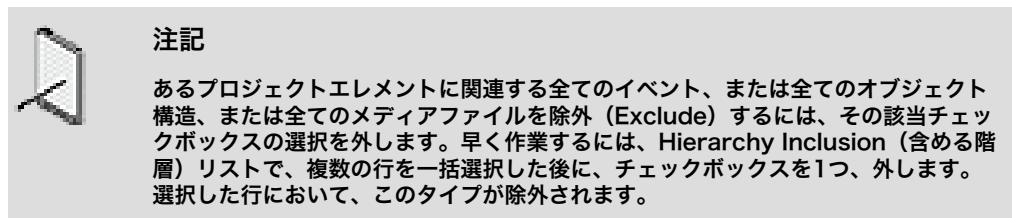


SoundBankにメディアを含む (Include) 場合は、個々のメディアファイルの詳細情報として、サンプルレート (Sample rate) 、オーディオフォーマット (Format) 、ファイルサイズ (Memory size) などが表示されます。この追加情報を確認して、各種プラットフォームの制限内に収まるように、個々のファイルのコンバージョン設定を調整するのも、簡単にできます。あるメディアファイルのコンバージョン設定を変えるには、リストでそのファイル名を右クリックして、Conversion Settings (コンバージョン設定) を選択するだけです。

ゲームに適した戦略を立てた上で、SoundBankに含めるエレメントや除外するエレメントを柔軟に設定したり、個々のメディアファイルのコンバージョン設定を変えたりして、ゲームのメモリ制限に効率的に対応できます。

### SoundBankに含める (Include) プロジェクトエレメントのタイプを設定するには:

1. SoundBank Editorに、SoundBankを1つロードする。
2. SoundBankに含める (Include) プロジェクトエレメントとして、リストの行ごとに、以下のタイプ (種類) から1つまたは複数を選択する。
  - Events - 関連する全てのイベントデータを、SoundBankに含める場合。
  - Structures - 関連する全てのオブジェクト構造データを、SoundBankに含める場合。
  - Media - 関連する全てのメディアファイルを、SoundBankに含める場合。



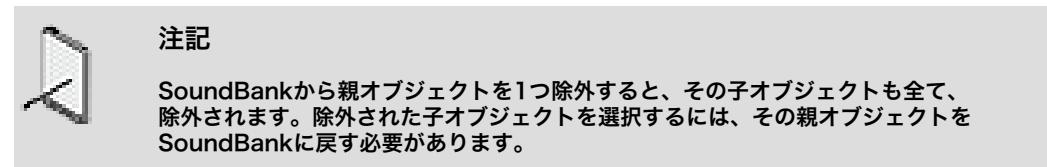
### SoundBankから、個々のプロジェクトエレメントを除外 (Exclude) するには:

1. SoundBank Editorに、SoundBankを1つロードする。
2. Editタブに切り替える。

このSoundBankに含まれる (Include) 、全てのイベント、オブジェクト構造、メディアファイルのリストが、表示される。

3. Search (検索) ツールやフィルタツールを使い、このSoundBankから除外 (Exclude) するプロジェクトエレメントを見つける。
4. 見つけたイベント、オブジェクト構造オブジェクト、またはメディアファイルの、チェックボックスの選択を外す。

このプロジェクトエレメントと、これに関連する全てのプロジェクトエレメントが、SoundBankから除外される。



5. 必要に応じて、上記3~4を繰り返して、このSoundBankから除外したいプロジェクトエレメントを全て、除外する。

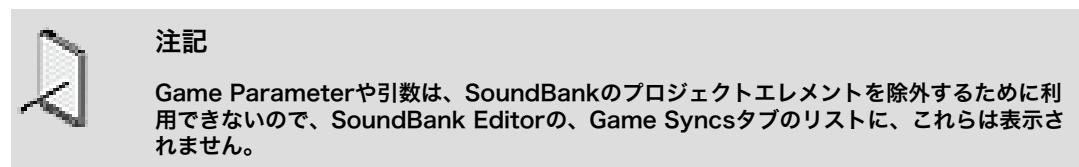
## 関連トピック

- 「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- 「SoundBank内のエレメントを検索する」
- 「SoundBankのエレメントのリストに、フィルタを設定する」
- 「プロジェクトエレメントを、別のSoundBankに移動する」
- 「SoundBankから、プロジェクトエレメントを削除する」

### プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する

SoundBankに、サウンドを含める (Include) ・除外する (Exclude) ための、もう1つの早く簡単な方法は、ゲームシンク (Game syncs) を利用することです。あるSoundBankに、イベントやサウンド構造を追加すると、これに関連するゲームシンクのリストが、SoundBank Editorの、Game Syncsタブに自動的に表示されます。ゲームシンクを除外することで、SoundBankの特定エレメントを除外できます。SoundBankからゲームシンクを1つ除外すると、そのゲームシンクをレファレンス (Reference / 参照) する全てのオブジェクト構造 (Object structures) やメディアファイル (Media files) も、同時に除外されます。

ゲームシンクへの関連性に基づいて、サウンドを含めたり除外したりすることで、ゲームの各ポイントで、どのサウンドをロードするのかを、より細かく設定できます。つまり、ゲームのプラットフォーム側のメモリ制限に、効率的に対応できます。

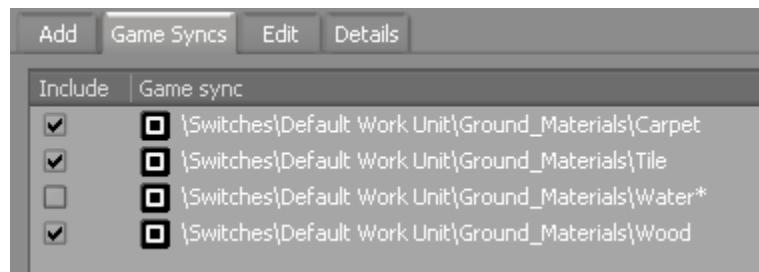


### エレメントを含める・除外するための、ゲームシンクの活用例

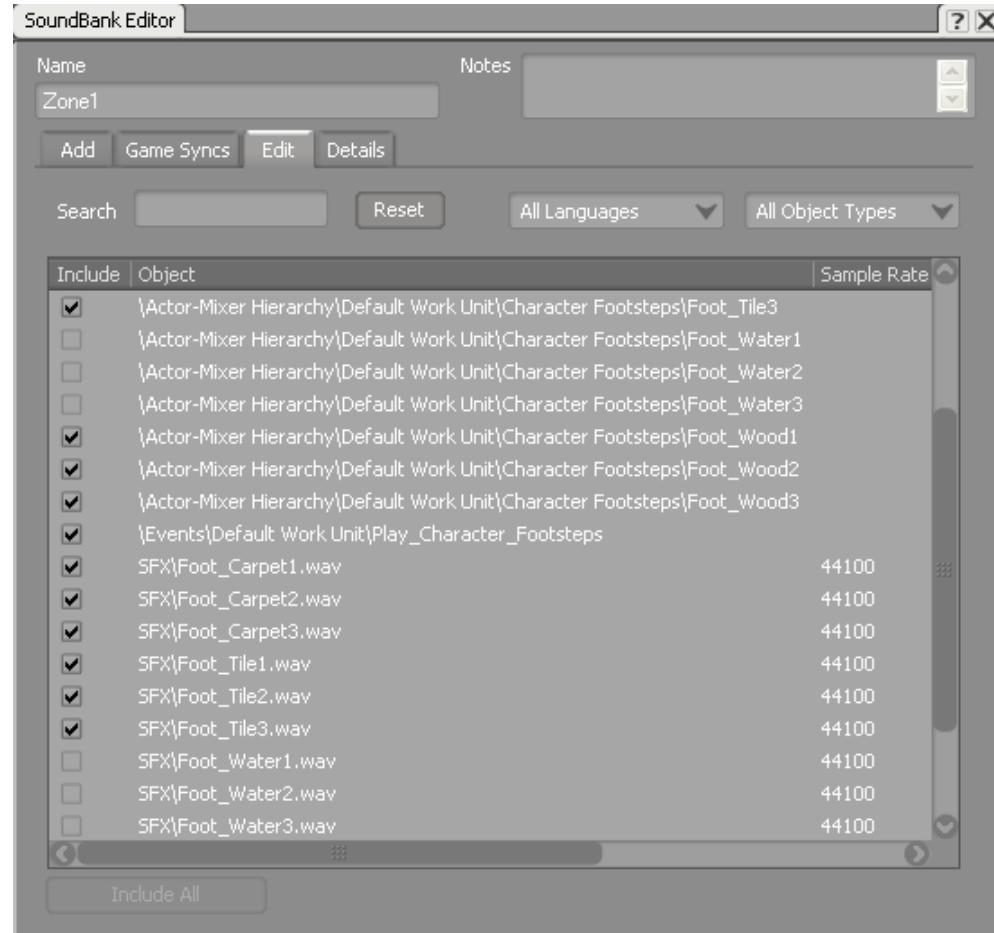
例えば、3つのゾーンから成り立つ、小規模なゲームの場合を考えます。ゲームが完了するまでに、メインキャラクターは第1～3のゾーン（Zones）を歩いて通る必要があります。Wwiseで、メインキャラクターの足音（Footsteps）サウンドをSwitch Containerで切り替えます。キャラクターが歩いて通る地面や床の素材（Ground materials）として、以下の4種類のSwitch（Switches）を作成します。

- Carpet
- Tile
- Wood
- Water

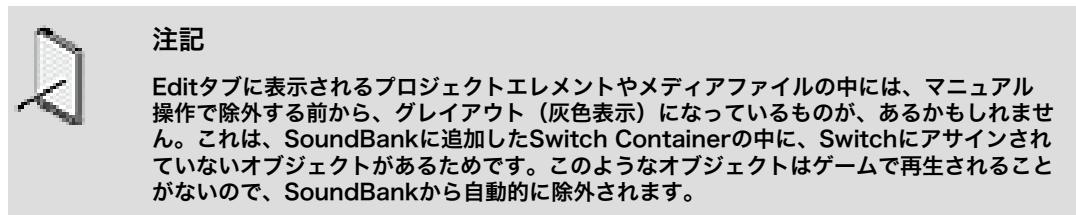
ゲーム中にSwitch「Water」が使われるのは、Zone 2とZone 3だけで、他の3種類のSwitchは、3つの全てのゾーンで使われるとします。SoundBankは、ゾーンごとに1つずつ作成したので、3つあります。Switch ContainerはどのSoundBankにも含める（Include）必要がありますが、Zone 1のSoundBankからは、水（Water）関連のイベント、サウンド構造、そしてメディアファイルを全て、除外します。手順としては、Zone 1用のSoundBankをロードしてから、Game Syncsタブに切り替えて、Switch「Water」の選択を外します。



これで、SoundBank「Zone1（ゾーン1）」から、水（Water）関連のサウンド構造やメディアファイルが除外されます。Editタブに切り替えると、このSwitch「Water」をレファレンスする全てのサウンド構造、イベント、メディアファイルが、このSoundBankに含まれないようになっています。



また、ゲームシンクを除外設定した定義ファイル（Definition file）をインポートすると、SoundBankからゲームシンクが自動的に除外されます。定義ファイルの使い方については「[定義ファイルをインポートして、SoundBankの中身を入れる](#)」を参照してください。



ゲームシンクを利用して、SoundBankからプロジェクトエレメントを除外するには：

1. SoundBank Editorに、SoundBankを1つロードする。
2. Game Syncsタブに切り替える。

Addタブにある各種イベントやオブジェクトに関連する、全てのゲームシンクのリストが、表示される。

3. SoundBankから除外するゲームシンクを見つけ、該当するチェックボックスの選択を外す。

このゲームシンクと、関連する全てのオブジェクトやメディアファイルが、SoundBankから除外される。

4. 上記3を繰り返して、SoundBankから除外したいゲームシンクを全て、除外する。

## 関連トピック

- 「SoundBankに、プロジェクトエレメントを含める・除外する」
- 「SoundBank内のエレメントを検索する」
- 「SoundBankのエレメントのリストに、フィルタを設定する」
- 「プロジェクトエレメントを、別のSoundBankに移動する」
- 「SoundBankから、プロジェクトエレメントを削除する」

## SoundBankに、プラグインメディアを含める

エフェクトの中には、コンボリューションリバーブ (Convolution Reverb) のように、メディアファイルが必要なものもあります。例えば、コンボリューションリバーブには、そのプラグインメディアとして、インパルスレスポンスファイルが必要です。ゲーム中にこのエフェクトが問題なく機能するためには、これらプラグインメディアファイルが、SoundBankに含まれている必要があります。また、これらのメディアファイルを含むSoundBankが、エフェクトのインスタンス化が必要な時に、ロードされている必要があります。

SoundBankに、プラグインメディアファイルを含める方法は、以下の通りいくつかあります。

- ShareSetエフェクト (Shareset effect) オブジェクトを、SoundBankの中に含めるリスト (Inclusion list) に挿入する。
- オーナーオブジェクト (つまり、バス階層オブジェクトまたはアクターミキサー階層オブジェクト) を、SoundBankの中に含めるリスト (Inclusion list) に挿入する。
- エフェクトのオーナーがアクターミキサー階層オブジェクトであれば、イベントを、SoundBankの中に含めるリスト (Inclusion list) に挿入する。

SoundBankにプラグインのメディアファイルが含まれたかどうかを確認するには、SoundBank Editorの、Editタブを見ます。

## SoundBank内のエレメントを検索する

ゲームが複雑になると、1つのSoundBankに、プロジェクトエレメントが何百個、あるいは何千個と入ることがあります。そこで、探しているプロジェクトエレメントを簡単に見つけるツールが必要です。Editタブでプロジェクトエレメントを見つけるには、検索 (Search) 機能を使います。

## SoundBankの中のエレメントを、検索するには:

1. SoundBank Editorに、SoundBankを1つロードする。
2. Editタブに切り替える。
3. Search (検索) フィールドで、探しているプロジェクトエレメントの名前を入力する。

Wwiseがリストにフィルタをかけ、入力した名前にマッチするオブジェクトだけが表示される。

4. Resetをクリックすると、Searchフィールドが空白になり、SoundBankの全てのコンテンツが再び表示される。

## 関連トピック

- 「SoundBankに、プロジェクトエレメントを含める・除外する」
- 「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- 「SoundBankのエレメントのリストに、フィルタを設定する」
- 「プロジェクトエレメントを、別のSoundBankに移動する」
- 「SoundBankから、プロジェクトエレメントを削除する」

### SoundBankのエレメントのリストに、フィルタを設定する

SoundBankに入っているプロジェクトエレメントを探す時に、リストにフィルタをかけて、ランゲージ (Languages) やオブジェクトタイプ (Object type) を限定して、表示させることができます。

### SoundBankのエレメントのリストに、フィルタを設定するには:

1. SoundBank Editorに、SoundBankを1つロードする。
2. Editタブに切り替える。
3. リストの項目に対して、ランゲージのフィルタを設定するには、Language Filter (ランゲージフィルタ) で、以下のオプションを1つ選択する。

**All languages (全ランゲージ)** - プロジェクトにある、全ランゲージのプロジェクトエレメントを、表示する。

**“Language” (ランゲージ名)** - 選択したランゲージ用のプロジェクトエレメントだけを、表示する。

4. リストの項目に対して、オブジェクトタイプのフィルタを設定するには、Object Type (オブジェクトタイプ) で、以下のオプションを1つ選択する。

**All object types (全オブジェクトタイプ)** - SoundBankにある、全てのオブジェクトを表示する。

**Events (イベント)** - SoundBankにある、イベントだけを表示する。

**Structures (構造)** - SoundBankにある、オブジェクト構造だけを表示する。

**Media (メディア)** - SoundBankにある、メディアファイルだけを表示する。

## 関連トピック

- 「SoundBankに、プロジェクトエレメントを含める・除外する」

- ・「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- ・「SoundBank内のエレメントを検索する」
- ・「プロジェクトエレメントを、別のSoundBankに移動する」
- ・「SoundBankから、プロジェクトエレメントを削除する」

### プロジェクトエレメントを、別のSoundBankに移動する

典型的なゲーム開発プロジェクトでは、SoundBankの数やコンテンツ（中身）が、徐々に変化します。そこで、複数のSoundBankを効率的に管理することが重要となります。SoundBankのコンテンツを効率的に扱えるように、Wwiseでは、あるSoundBankの中のプロジェクトエレメントを、別のSoundBankに簡単に移動できます。

### プロジェクトエレメントを、別のSoundBankに移動するには:

1. SoundBank Editorに、SoundBankを1つロードする。
2. SoundBank Editorの、Addタブで、別のSoundBankに移動するプロジェクトエレメントを、1つまたは複数を選択する。
3. 選択項目を、SoundBank Editorの、Addタブから、SoundBank Managerの、別のSoundBankまで、ドラッグして入れる。

ドラッグしたプロジェクトエレメントが、元のSoundBankから削除され、新しいSoundBankに追加される。

### 関連トピック

- ・「SoundBankに、プロジェクトエレメントを含める・除外する」
- ・「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- ・「SoundBank内のエレメントを検索する」
- ・「SoundBankのエレメントのリストに、フィルタを設定する」
- ・「SoundBankから、プロジェクトエレメントを削除する」

### SoundBankから、プロジェクトエレメントを削除する

SoundBankから、エレメントを1つ削除すると、これに関連する全てのイベント、オブジェクト構造、メディアファイルも、削除されます。

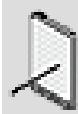
無効となったプロジェクトエレメントは、WwiseがSoundBankの生成段階で無視するため、エラーや無駄なメモリ消費につながりませんが、プロジェクトの一貫性を保つために、SoundBankから削除した方が良いでしょう。無効となったイベントやオブジェクト構造は、SoundBank Editorの、Addタブで、名前の横に「Missing（なし）」と表示されます。

### SoundBankから、プロジェクトエレメントを削除するには

1. SoundBank Editorに、SoundBankを1つロードする。

2. Addタブで、削除するオブジェクトを1つまたは複数を選択する。
3. Deleteキーを押す。

プロジェクトエレメントと共に、関連するイベント、オブジェクト構造、メディアファイルが全て、SoundBankから削除されます。



#### 注記

SoundBankの複数のプロジェクトエレメントが、あるオブジェクトまたはメディアファイルを参照している場合は、プロジェクトエレメントの1つを削除しても、このオブジェクトまたはメディアファイルを、別のプロジェクトエレメントがまだレファレンス（参照）しているので、自動的に削除されることはありません。

## 関連トピック

- 「SoundBank内のエレメントを検索する」
- 「SoundBankに、プロジェクトエレメントを含める・除外する」
- 「プロジェクトエレメントを含める・除外するために、ゲームシンクを利用する」
- 「SoundBankのエレメントのリストに、フィルタを設定する」
- 「プロジェクトエレメントを、別のSoundBankに移動する」

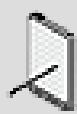
## SoundBankの管理

一旦作成した様々なSoundBankは、開発プロセス中にいつでも、変更できます。SoundBankの管理には、以下のタスクが伴います。

- 「SoundBankの名前を変更する」
- 「SoundBankを削除する」
- 「SoundBankの詳細をモニターする」

### SoundBankの名前を変更する

作成したSoundBankの名前は、必要に応じて、あとから変更できます。



#### 注記

SoundBank名を変更する予定がある場合には、後の段階で変更すると追加のプログラミングが必要となる可能性があるので、開発プロセスの早い段階で行ってください。

### SoundBank名を変更するには:

1. Project Explorerで、SoundBanksタブに切り替える。
2. 名前を変更するSoundBankを選択して、以下のいずれかを行う。

F2を押す。

SoundBankの、Nameフィールドを、クリックする。

SoundBankの名前が、ハイライト表示される。

3. SoundBank名を新しい名前に変更して、Enterを押す。

## 関連トピック

- 「SoundBankに中身を入れる」
- 「SoundBankを削除する」
- 「SoundBankの詳細をモニターする」

## SoundBankを削除する

不要となつたSoundBankは、削除できます。

### SoundBankを削除するには:

- Project Explorerで、SoundBanksタブに切り替える。
- 以下のいずれかを行う:
  - 削除するSoundBankを1つ右クリックして、Delete Selectionを選択する。
  - 削除するSoundBankを1つクリックして、Deleteキーを押す。

選択したSoundBankが削除される。



## 関連トピック

- 「SoundBankに中身を入れる」
- 「SoundBankの名前を変更する」
- 「SoundBankの詳細をモニターする」

## SoundBankの詳細をモニターする

プラットフォームごとのメモリ制限を守ることは必須なので、プロジェクトのSoundBankを1つ1つ、管理してトラブルシューティングすることは重要です。SoundBankを効率的に管理するためには、最大サイズを超えているSoundBankや、不足しているランゲージファイルや、生成後のSoundBankの合計サイズなどの、重要な情報を把握する必要があります。

Wwiseは、SoundBank Editorの、Detailsタブに、これら詳細情報を表示します。

The screenshot shows the SoundBank Editor window with the 'Details' tab selected. It displays a table of file statistics:

Types	Languages	SFX	Music	Voice	Motion	Data Size	Free Space	Missing Files	Files Replaced	Memory Size
SFX, Voice, Motion	English (US)	30 671 556	0	717 128	14 476	31 403 180	-1 403 180	0	0	31 403 160
SFX, Voice, Motion	French (France)	30 671 556	0	717 128	14 476	31 403 180	-1 403 180	0	1	31 403 160
SFX, Voice, Motion	Spanish (Mexico)	30 671 556	0	717 128	14 476	31 403 180	-1 403 180	0	1	31 403 160

SoundBankに、サウンドボイス (Sound Voice) オブジェクトが含まれる場合は、Detailsタブの情報がランゲージごとに表示されます。また、問題がありそうな項目は、すぐに分かるように赤字で表示されます。



#### 注記

SoundBank Managerや、SoundBank Editorのメモリ情報は、全てバイト (Bytes) 表示です。

### SoundBankの詳細をモニターするには:

1. SoundBank Managerで、詳細を確認したいサウンドバンクを1つダブルクリックする。

関連情報が、SoundBank Editorの、Detailsタブに表示されます。

2. Detailsタブで、以下の情報が確認できる。

- **Types** - このSoundBankに含まれるオブジェクトのタイプで、SFX、Voice、Music、Motionなど。
- **Languages** - このSoundBankのランゲージ。
- **SFX** - Sound SFXオブジェクトの合計メモリ使用量。
- **Music** - ミュージックオブジェクトの合計メモリ使用量。
- **Voice** - あるランゲージ用のサウンドボイスオブジェクトの合計メモリ使用量。
- **Data Size** - あるランゲージ用の、SFX、ボイス、モーションの全オブジェクトの合計メモリ使用量。
- **Free Space** - SoundBankの未使用スペースの大きさ。この数値が表示されるのは、最大サイズ (Maximum size) が指定されている場合のみ。
- **Missing Files** - あるランゲージで不足しているメディアファイルの数。
- **Files Replaced (代替ファイル)** - サウンドボイス (Sound Voice) オーディオファイルのうち、まだ用意できず、現在はレファレンスランゲージ (Reference language) のオーディオファイルで置き換えられているファイルの数。
- **Memory Size** - メモリにロードされるSoundBankデータの、メモリ使用量。
- **Prefetch Size** - ストリーミングする様々なメディアファイルのための、プリフェッチデータのメモリ使用量。プリフェッチデータは、ストリーミングするメディアファイルが、遅延なく再生されるように、SoundBankと共にメモリにロードされる。
- **File Size** - このSoundBankを生成した時の、合計ファイルサイズ。

### 関連トピック

- 「[SoundBankに中身を入れる](#)」
- 「[SoundBankのコンテンツを管理する](#)」
- 「[SoundBankを削除する](#)」

## SoundBankのカスタム属性を設定する

それぞれのプラットフォームやランゲージ用に様々なSoundBankを生成する前に、SoundBankの詳細設定を決める必要があります。基本的にSoundBankの設定は、プロジェクト全体を対象に、Project Settingsダイアログボックスで設定しますが、それをオーバーライドしてカスタムユーザー設定を作成することもできます。定義できるSoundBankのユーザー設定は、以下の通りです。

- ・「[SoundBankのカスタム設定を定義する](#)」 - 生成されたSoundBankに、どの情報を持たせるかを設定。
- ・「[保存したSoundBankの場所をカスタム設定する](#)」 - ハードドライブまたはネットワークの、どの場所にSoundBankを保存するかを設定。
- ・「[SoundBankのプリジェネレーション・ポストジェネレーションの処理を、カスタマイズする](#)」 - SoundBank生成の直前に実施するタスクの設定。
- ・「[SoundBankのプリジェネレーション・ポストジェネレーションの処理を、カスタマイズする](#)」 - SoundBank生成の直後に実施するタスクの設定。
- ・「[外部ソースファイルのインプット・アウトプット場所を指定する](#)」 - テンプレートソース (Template Source) プラグインに使う様々な外部オーディオソースの、インプットパスとアウトプットパスを指定する。

これらの属性を、プロジェクト全体を対象に設定する方法については「[プロジェクトのSoundBankの設定](#)」を参照してください。

### SoundBankのカスタム設定を定義する

SoundBankの設定内容によって、生成プロセスにどのような情報を入れるのか、どのように入れるのか、そしてどのフォーマットで生成するのか、などが決まります。SoundBank設定はプロジェクト全体を対象に設定しますが、場合によっては、これをオーバーライドして個別のカスタム設定を作成することもあります。設定内容は、SoundBank内のデータやメディアを、ゲームがどのようにアクセスするのかによって、決まります。



#### 注記

カスタム設定は、それを設定した人（ユーザー）だけが使用でき、その人が生成するSoundBankだけに適用されます。

### SoundBankのカスタム設定を定義するには

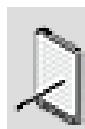
1. SoundBank Managerで、User Settingsをクリックする。  
SoundBank User Settingsダイアログボックスが開く。
2. Override Project SoundBank Settingsオプションを選択する。
3. SoundBankのカスタム設定を定義するために、以下のオプションを選択する。
  - **Allow SoundBanks to exceed maximum size:** SoundBankのサイズが、指定した最大サイズを超えて、生成する場合。
  - **Generate SoundBank content files:** 各SoundBankの、コンテンツリストのファイルを作成する場合。コンテンツファイルには、各種イベント、バ

ス、State、Switchの情報や、ストリーミングするメディアファイルやメモリ内のメディアファイルのリストが、含まれます。

- **Generate header file:** イベント名、State名、Switch名、Game Parameter名を、それぞれのIDにマッピングするヘッダファイルを作成する場合。
- **Generate max attenuation info for events:** SoundBanksInfo.xmlファイルに、各イベントの最大減衰に関する情報を含める場合。
- **Estimated duration** 各イベントの SoundBanksInfo.xml ファイルに、サウンドを永久にループさせる、もしくは1回のみのサウンドかを設定するだけでなく、各イベントの推定最長ならびに最短持続時間を含める場合。
- **Use SoundBank Names :** 生成する「.bnk SoundBank」ファイルのファイル名や、バンクの中で他のSoundBankを参照する時に、SoundBank名を使うのか（チェック入り）、またはIDを使うのか（チェックなし）の設定。

4. ヘッダファイルを生成する場合は、その保存先を決める必要がある。保存先を設定するには、以下を行う：

- テキストボックスに、パスを直接入力する。
- **Browse (...)** ボタンをクリックして、ブラウザで保存場所まで移動する。



#### 注記

ヘッダファイルの保存場所は、フルパスまたは相対パスを使って指定します。相対パスを使う場合、パスの起点としてプロジェクトフォルダを使います。

5. SoundBankのコンテンツファイルを生成する場合は、**SoundBank content file format**オプションで、生成するテキストファイルのフォーマットを選択する。



#### Tip

ファイルパス、オブジェクト名、オブジェクトノーツに非ANSI文字が含まれる場合は、Unicodeフォーマットを使用するべきです。

6. OKをクリックして、設定を適用する。



#### 注記

オーバーライド設定を解除するまで、SoundBankを生成する時に、ここで設定したカスタムユーザー設定が使われます。

## 関連トピック

- 「[保存したSoundBankの場所をカスタム設定する](#)」
- 「[SoundBankのプリジェネレーション・ポストジェネレーションの処理を、カスタマイズする](#)」

### 保存したSoundBankの場所をカスタム設定する

プロジェクト用に生成された全てのSoundBankが、Project Settingsダイアログボックスの、SoundBanksタブで指定した、デフォルトのプロジェクト保存場所に自動的に保存されます。これをオーバーライドする必要があれば、カスタムユーザー設定を作成します。

SoundBankの保存場所は、フルパスまたは相対パスを使って指定します。相対パスを使う場合、パスの起点としてプロジェクトフォルダを使います。例えば、以下のフルパスと相対パスは、同じ場所を指定しています。

- C:\Wwise Projects\My Project\GeneratedSoundBanks\Windows
- GeneratedSoundBanks\Windows\

## 保存するSoundBankの、新しい保存場所を指定するには:

1. SoundBank Managerで、User Settingsをクリックする。  
SoundBank User Settingsダイアログボックスが開く。
2. Override Project SoundBank Paths（プロジェクトのSoundBankのパスをオーバーライド）オプションを選択する。
3. パスを指定するために、以下のいずれかを行う。
  - テキストボックスに、パスを直接入力する。
  - Browse (...) ボタンをクリックして、ブラウザで保存場所まで移動する。
4. OKをクリックして、変更内容を適用する。



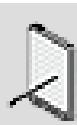
## 関連トピック

- 「[SoundBankのカスタム設定を定義する](#)」
- 「[SoundBankのプリジェネレーション・ポストジェネレーションの処理を、カスタマイズする](#)」

## SoundBankのプリジェネレーション・ポストジェネレーションの処理を、カスタマイズする

ワークフローによっては、SoundBank生成の直前、または直後に実行すべき処理やタスクがあるかもしれません。例えば、特定のSoundBank生成の前に、そのSoundBankファイルをソースコントロールシステムからチェックアウトする処理や、SoundBank生成直後に、ストリーム用ファイルをSoundBankのディレクトリにコピーする処理などです。このような情報は、通常、プロジェクト全体を対象に設定しますが、場合によっては、設定をオーバーライドして、自分のカスタム設定を作成することもできます。

Wwiseでこのような種類のタスクを、SoundBankのプリジェネレーション (Pre-generation / 生成前) タスク、ポストジェネレーション (Post-generation / 生成後) タスクと呼び、コマンドラインを作成して定義します。Wwiseには、必要なコマンドラインを必要な数だけ簡単にビルドするための、専用のコマンドラインエディタがあります。処理をさらにシンプルにするために、コマンドラインで使える全てのWwise固有の変数やWindows環境変数のリストが、エディタに入っています。



## 注記

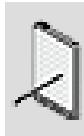
ビルドプロセスは、ある外部ツールからエラーが返された場合や、見つからない場合は、中断されることがあります。該当するログの重大度レベル (Severity level) を、ログの設定で、Fatal Errorとしてすることで、これが実現されます。詳しくは「[ログのメッセージの、重大度を変更する](#)」を参照。

カスタム設定したGlobal opening step (グローバルなオープニングステップ) が失敗すると、ビルドプロセス全体が中断され、バンクが一切作成されません。一方、1つのプラットフォーム専用の、1つのカスタムステップで失敗が発生すると、そのプラットフォームだけをスキップされます。ただし、Global closing step (グローバルなクロージングステップ) は、プロセスの一部が成功しなかったため、スキップします。失敗と判断されるのは、外部プロセスが「0」以外の数値を返した時です。

カスタマイズしたコマンドラインを書くために使えるWwise変数を、下表に具体的に示します。

Command Line Variable	内容
\$(AllowExceedMaximum)	SoundBankが、設定された最大サイズを超過した場合、生成するか否かを示す。 Allow SoundBanks to exceed maximumオプションを選択した場合、この変数は「true」に設定される。
\$(ContentFileFormat)	生成されるSoundBankのコンテンツファイルの、ファイルタイプを示す。以下が、可能な値。 <ul style="list-style-type: none"> <li>• ANSI</li> <li>• Unicode</li> </ul>
\$(GenerateContentFile)	各SoundBankのコンテンツリストのファイルを作成するか否かを示す。 コンテンツファイルには、イベント、バス、State、Switchの情報や、ストリーム用やメモリ用の全オーディオファイルのリストが含まれる。 Generate SoundBank content filesオプションを選択した場合、この変数は「true」に設定される。
\$(GenerateHeaderFile)	イベント、State、Switch、Game Parameterの名前をIDにマッピングしたヘッダファイルを、生成するか否かを示す。 Generate header fileオプションを選択した場合、この変数は「true」に設定される。
\$(GenerateMaxAttenuationInfo)	イベントの最大減衰の情報を生成するか否かを示す。 <b>Metadata Options: Max attenuation</b> オプションが選択されている時、この変数は「True」に設定される。
\$(GenerateEstimatedDuration)	イベントに推定最長ならびに最短持続時間、および持続時間タイプ情報が生成するか否かを示す。 <b>Metadata Options: Estimated Duration</b> オプションが選択されている時、この変数は「True」に設定される。
\$(HeaderFileFullPath)	ヘッダファイルのフルパスで、具体的には: \$(HeaderFilePath)\Wwise_IDs.h
\$(HeaderFilePath)	ヘッダファイルを保存する、パスまたは場所。 このパスは、テキストボックスHeader file pathから取得される。
\$(InfoFilePath)	現在のプラットフォームの、Infoファイルの完全なファイル名。
\$(IsRunningFromCmdLine)	Wwiseが、-generatesoundbanksフラグ付きのコマンドラインから起動されたか否かを示す。
\$(LanguageList)	コマンドラインにパスされた言語のリスト、またはSoundBank Managerで選択された言語のリスト。

Command Line Variable	内容
	 <p><b>注記</b> このリストは、半角スペース区切りのリストです。</p>
\$(Platform)	現在のプラットフォーム名。
\$(SoundBankList)	コマンドラインにパスされたSoundBankのリスト、またはSoundBank Managerで選択されたSoundBankのリスト。
	 <p><b>注記</b> このリストは、半角スペース区切りのリストです。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。</p>
\$(SoundBankPath)	現在のプラットフォームのSoundBankが保存されている、パスまたは場所。
\$(UseSoundBankNames)	生成されたSoundBankファイル (.bnk) の名前として、SoundBank名 (true) とID (false) のどちらを使用するかを指定し、これは SoundBank内から他のSoundBankのメディアをレファレンスする時にも、適用される。  Use SoundBank namesオプションを選択した場合、この変数は「true」に設定される。
\$(WwiseExeDriveLetter)	Wwise実行ファイル (Wwise.exe) がある、ワークステーションのドライブレター。
\$(WwiseExePath)	Wwise実行ファイル (Wwise.exe) のパスまたは場所。
\$(WwiseExeProcessID)	Wwise実行ファイル (Wwise.exe) の処理ID (Process ID) の数値。
\$(WwiseProjectDriveLetter)	Wwiseプロジェクトがある、ワークステーションのドライブレター。
\$(WwiseProjectName)	現在のプロジェクト名。
\$(WwiseProjectPath)	Wwiseプロジェクトのパスまたは場所。
\$(WaapiWampPort)	WA APIがWAMPプロトコル用に使うポート。
\$(WaapiHttpPort)	WA APIがHTTPプロトコル用に使うポート。

**注記**

環境変数は自動的に、\$(WWISESDK) のようにマッピングされます。

柔軟性を高めるために、Wwiseでは以下の種類のステップに対して、異なるコマンドラインを設定できます。

- **Global opening step** - 全プラットフォームにグローバル（全体的）に適用し、他のどのステップよりも前に実行するコマンドライン。
- **Platform-specific pre-generation step** - 特定プラットフォームに適用し、SoundBank生成前に実行するコマンドライン。
- **Platform-specific post-generation step** - 特定プラットフォームに適用し、SoundBank生成後に実行するコマンドライン。
- **Global closing step** - 全プラットフォームにグローバル（全体的）に適用し、全てのステップの後に実行するコマンドライン。

ポストジェネレーション（生成後）のデフォルトのステップとして、プラットフォームごとに、ストリーミングファイルをSoundBankのディレクトリにコ

ピーするステップが、プロジェクトに必ず入っています。なお、他にもコマンドラインを設定して、どのような種類のタスクでも自動化できます。またWwiseには、SoundBankの中の全てのデータやメディアが入ったパッケージをFile Packager（ファイルパッケージャ）機能で生成する、別のファクトリコマンドラインも同梱されています。File Packagerについては[36章ファイルパッケージの管理](#)を参照してください。ファクトリコマンドラインのロードについては「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」を参照してください。

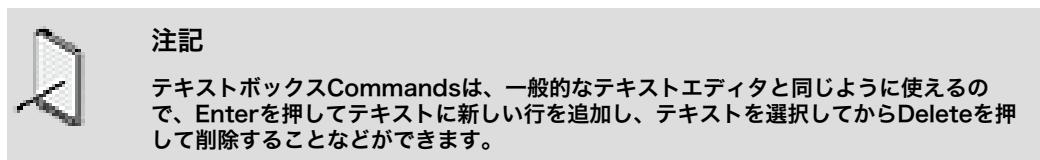
作成したコマンドラインをファイル (.wcmdline) に保存して、後で同じプロジェクトや別のプロジェクトで使用したり、他のユーザーとシェアしたりすることもできます。コマンド保存については「[カスタムコマンドラインを、ファイルに保存](#)」を参照してください。

## ユーザー専用の、SoundBankのプリジェネレーション（生成前）タスクを、設定するには:

1. SoundBank Managerで、User Settingsをクリックする。  
SoundBank User Settingsダイアログボックスが開く。
2. Override Project Pre-Generation Stepオプションを選択する。
3. Project Settingsダイアログボックスで設定した、プリジェネレーション（生成前）に適用する、Global opening step（グローバルなオープニングステップ）に追加や変更を行うには、該当するEditボタン (...)をクリックする。

Pre-Generation Step Editorが開く。

4. テキストボックスDescriptionに、実行するステップやタスクの内容がよく分かる名前を入力する。
5. テキストボックスCommandsで必要に応じて、新規コマンドラインを書くか、既存コマンドラインを編集する。



6. コマンドに、組み込みマクロや環境変数をインサートしたい場合は、以下を行う：

グループボックスMacrosで、以下のオプションを1つ選択する。

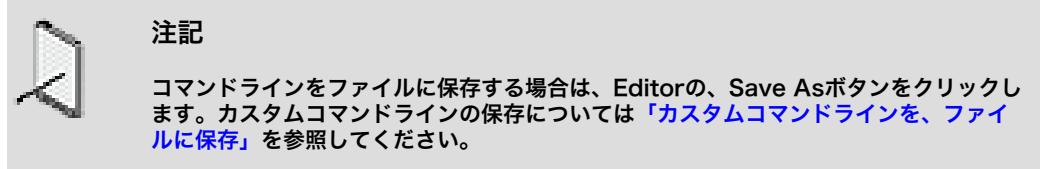
- Built-in Macros - 組み込みマクロのことであり、Wwiseコマンドライン内で使える、Wwise固有の変数リストを表示。
- Environment Variables - 環境変数のことであり、Wwiseコマンドライン内で使える、Windows固有の環境変数リストを表示。

コマンドラインに変数を追加するには、以下のいずれかを行う：

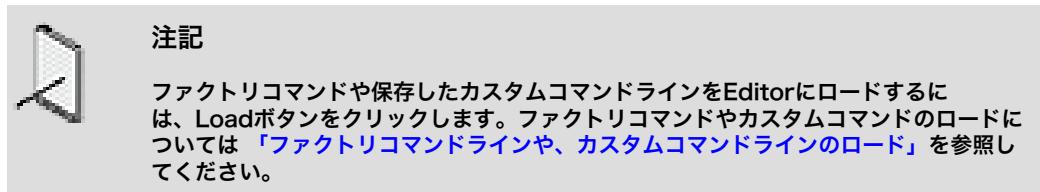
- リストの変数をダブルクリックする。
- リストの変数を選択して、Insertをクリックする。

必要に応じて、コマンドラインに変数を追加する。

7. プリジェネレーションのステップを追加する場合は、最初の行の終わりで、Enterを押し、新しいコマンドラインを入力する。
8. OKをクリックして、コマンドラインを保存して、Pre-Generation Step Editorを閉じる。

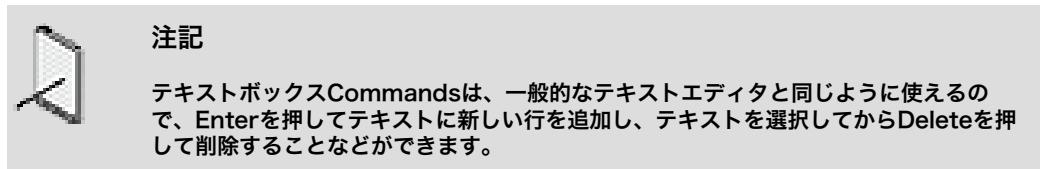


9. 既存のプラットフォーム別プリジェネレーションステップに、追加または変更を行うには、プラットフォームごとに上記3~8を繰り返す。



## ユーザー専用の、SoundBankのポストジェネレーション（生成後）タスクを、設定するには

1. SoundBank Managerで、User Settingsをクリックする。  
SoundBank User Settingsダイアログボックスが開く。
2. Override Project Post-Generation Step（ポストジェネレーションステップをオーバーライド）オプションを選択する。
3. Project Settingsダイアログボックスで設定した、既存のポストジェネレーションステップに追加や変更を行うには、該当するEditボタン (...)をクリックする。  
Post-Generation Step Editorが開く。
4. テキストボックスDescriptionに、実行するステップやタスクの内容がよく分かる名前を入力する。
5. テキストボックスCommandsで必要に応じて、新規コマンドラインを書くか、既存コマンドラインを編集する。



6. コマンドに、組み込みマクロや環境変数をインサートしたい場合は、以下を行う：

グループボックスMacrosで、以下のオプションを1つ選択する。

- Built-in Macros - 組み込みマクロのことであり、Wwiseコマンドライン内で使える、Wwise固有の変数リストを表示。

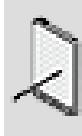
- **Environment Variables** - 環境変数のことであり、Wwiseコマンドライン内で使える、Windows固有の環境変数リストを表示。

コマンドラインに変数を追加するには、以下のいずれかを行う：

- リストの変数をダブルクリックする。
- リストの変数を選択して、**Insert**をクリックする。

必要に応じて、コマンドラインに変数を追加する。

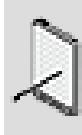
7. プリジェネレーションのステップを追加する場合は、最初の行の終わりで、**Enter**を押し、新しいコマンドラインを入力する。
8. **OK**をクリックして、コマンドラインを保存して、Post-Generation Step Editorを閉じる。



#### 注記

コマンドラインをファイルに保存する場合は、Editorの、**Save As**ボタンをクリックします。カスタムコマンドラインの保存については「[カスタムコマンドラインを、ファイルに保存](#)」を参照してください。

9. グローバルなクロージングステップや、プラットフォームごとのステップを追加するには、上記3~8を繰り返す。



#### 注記

ファクトリコマンドや保存したカスタムコマンドラインをEditorにロードするには、**Load**ボタンをクリックします。ファクトリコマンドやカスタムコマンドのロードについては「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」を参照してください。

## 関連トピック

- 「[SoundBankのカスタム設定を定義する](#)」
- 「[保存したSoundBankの場所をカスタム設定する](#)」
- 「[ファクトリコマンドラインや、カスタムコマンドラインのロード](#)」
- 「[カスタムコマンドラインを、ファイルに保存](#)」

## 外部ソースファイルのインプット・アウトプット場所を指定する

プラグインTemplate Sourceを使う場合は、ランタイムにテンプレートソースと関連付ける外部オーディオファイルの場所を、指定する必要があります。また、コンバージョン後のソースを保存するフォルダも指定して、Wwiseがゲーム中に使えるようにします。

## 外部オーディオソースのインプットパス・アウトプットパスを指定するには：

1. SoundBank Managerで、**User Settings**をクリックする。

SoundBank Settingsダイアログボックスが開く。

2. External Sourcesタブに切り替える。

3. SoundBankのジェネレーション（生成）プロセスの一貫として、外部ソースをコンバージョンするには、**Convert external sources when generating banks**（バンク生成時に外部ソースをコンバージョン）オプションを選択する。
4. Project Settingsダイアログボックスで指定したインプットパスを変更するには、**Override Input Path**オプションを選択する。
5. 外部ソースファイルが保存されているフォルダを指定するために、以下のいずれかを行う。
  - External Sourcesリストをクリックして、オーディオファイルの場所を示すパスを入力する。
  - 1つのゲームプラットフォームの、**Browse**ボタン (...) をクリックして、そのプラットフォーム用のオーディオファイルが入っているフォルダを見つけて、**Open**をクリックする。
6. 上記3を、全てのプラットフォームで繰り返す。
7. Project Settingsダイアログボックスで設定したアウトプットパスを変更するには、**Override Output Path**オプションを選択する。
8. 外部ソースを保存するフォルダを指定するために、以下のいずれかを行う。
  - External Sources Output Folderリストをクリックして、オーディオファイルを保存する場所のパスを入力する。
  - 1つのゲームプラットフォームの、**Browse**ボタン(...)をクリックして、そのプラットフォームのオーディオファイルを保存するフォルダを見つけて、**OK**をクリックする。
9. 上記5を、全てのプラットフォームで繰り返す。
- 10 OKをクリックして、SoundBank Settingsダイアログボックスを閉じる。

## プロジェクトの各種SoundBankの生成

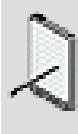
SoundBankは、対応するゲームプラットフォームごと、そしてランゲージごとに生成できます。 SoundBankは開発プロセス中にいつでも生成できるので、各種 SoundBankが様々なゲームプラットフォームにどのように実装されるかをテストできます。 Wwiseでは、効率的に作業できるように、全てのプラットフォームと全てのランゲージのSoundBankを、同時に生成できます。

生成したSoundBankに含めることができる情報の種類を、以下に示します。

- イベントの情報
- サウンド、ミュージック、モーション、アクターミキサー、コンテナの情報
- メモリに保存されるメディアの、サウンドデータ、ミュージックデータ、モーションデータ
- ストリーム用メディアファイルのうち、ゼロレイテンシに設定したファイルの、プリフェッチデータ
- ストリーム用メディアファイルへの、レファレンス（参照情報）

SoundBankに含まれる情報はそのプロジェクト専用なので、あるSoundBankと一緒に使えるのは、同じプロジェクトで生成した他のSoundBankだけです。

プロジェクトから既に削除したイベントやオブジェクト構造がSoundBankに含まれていても、SoundBankを生成できます。無効となったプロジェクトエレメントは、WwiseがSoundBankの生成段階で無視するため、エラーや無駄なメモリ消費につながりません。ただし、プロジェクトの一貫性を保つために、SoundBankから削除した方が良いでしょう。

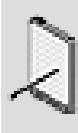


#### 注記

SoundBankの中のイベントやオブジェクト構造が無効になると、そのアイコンが消え、SoundBank Editorの、Addタブで、名前の横に「Missing (なし)」と表示されます。

SoundBankを生成する前に設定すべきオプションがいくつかあり、例えばコンテンツファイルやヘッダファイルを生成するのか、SoundBank名が使えるのか、SoundBankをどこに保存するのか、ストリーム用ファイルをSoundBankのディレクトリにコピーするのか、などです。このようなSoundBankの設定は、以下の2つの異なるレベルで設定されます：

- **プロジェクトレベル (Project level) で設定** - プロジェクト全体のデフォルト設定を作成すること。これらの設定は、Project Settingsダイアログボックスで設定する。プロジェクト全体を対象に、つまりプロジェクトレベルで、SoundBankの設定を決めるには「[プロジェクトのSoundBankの設定](#)」を参照。
- **ユーザーレベル (User level) で設定** - プロジェクト設定をオーバーライドする、カスタムユーザー設定を作成すること。ユーザー設定は、SoundBank User Settingsダイアログボックスで設定する。カスタムユーザー設定については「[SoundBankのカスタム設定を定義する](#)」を参照。



#### 注記

Xbox OneやPlayStation 4などの特定ゲーム機向けのSoundBankを生成する前に、そのプラットフォームの認定コンテンツクリエイター・デベロッパーになる必要があり、プラットフォームに必要な.cabファイルを入手しなければなりません。インストールパッケージ (Installation package) に、.cabファイルが入っていないプラットフォームは、SoundBank Platformsリストの選択可能なプラットフォームとして表示されません。Wwiseや関連.cabファイルのインストール方法については「[Wwise Installation and Migration Guide](#)」の「[Installation Issues](#)」を参照してください。

## プロジェクトのSoundBankを生成するには:

1. 以下のいずれかを行い、SoundBankレイアウトに切り替える。
  - メニューバーで、Layouts > SoundBankをクリックする。
  - F7を押す。



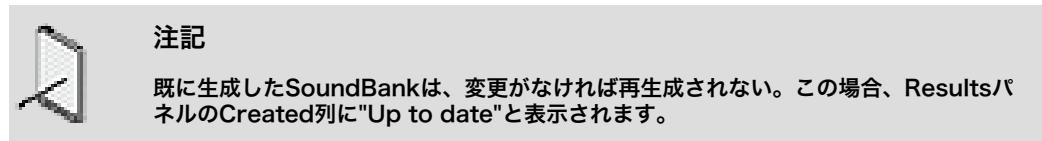
#### Tip

他のレイアウトから、SoundBank Managerのフローティングビューを開くには、Views > SoundBank Managerをクリックするか、ショートカットキー (Shift +B) を使います。

2. SoundBank Managerで、今回生成する全てのSoundBankを選択する。
3. Platformsリストで、SoundBankを生成する対象のプラットフォームを、全て選択する。

4. Languagesリストで、SoundBankを生成する対象のランゲージを、全て選択する。
5. Generateをクリックして、SoundBank生成を開始する。

ダイアログGenerating SoundBanksが開き、SoundBank生成の進行状況が表示される。全てのSoundBankの生成が完了すると、Generating SoundBanks - Completedダイアログボックスが開く。



6. ダイアログボックスのメッセージを確認して、全てのSoundBankの生成が成功したことを確認する。
7. 最終的な検証のために、ログに表示された項目を全て確認する。SoundBank logには、生成プロセス中に発生したエラーや警告 (Warnings) などのメッセージが全て、含まれる。SoundBank Logについては「[SoundBank生成中に発生した問題のトラブルシューティング](#)」を参照。
8. SoundBank Logの検証が終われば、Generating SoundBanks - Completedダイアログボックスと共に閉じる。

SoundBankファイルが指定したフォルダに保存され、ゲームに実装できる状態となる。

## 関連トピック

- 「[プロジェクトのSoundBankの設定](#)」
- 「[SoundBankのカスタム属性を設定する](#)」
- 「[SoundBankの名前を変更する](#)」
- 「[SoundBankの詳細をモニターする](#)」
- 「[SoundBankを削除する](#)」
- 「[スクリプトを使用してSoundBankを生成する](#)」
- 「[SoundBank内のエレメントを検索する](#)」
- 「[SoundBank生成中に発生した問題のトラブルシューティング](#)」

## SoundBank生成中に発生した問題のトラブルシューティング

### Log Ignoreリストに、メッセージを追加する

SoundBank生成中のログに表示される警告やメッセージを、以後は表示させたくない、または表示させる必要がない場合も考えられます。不要な警告やメッセージを、SoundBank Logに表示させないためには、Log Ignore（ログで無視）リストに追加します。また、Log Ignoreリストからそのメッセージタイプを削除すると、再びログに表示されます。

Log Ignore Listに表示させるメッセージは、Project Settingsダイアログボックスでも設定できます。詳しくは「[ログに表示されるメッセージの管理](#)」を参照してください。

## Log Ignoreリストにメッセージを追加するには:

1. SoundBank Logで、ログに表示させたくないメッセージタイプを、全て選択する。
2. 選択項目を右クリックして、Add Message(s) to Ignore Listを選択する。

プロジェクトで、次回SoundBankを生成する時に、選択したメッセージタイプのメッセージは、SoundBank Logに表示されません。

## 関連トピック

- [「SoundBank生成中に発生した問題のトラブルシューティング」](#)
- [「プロジェクトの各種SoundBankの生成」](#)
- [「スクリプトを使用してSoundBankを生成する」](#)

SoundBank生成中に、Wwiseが問題を検知すると、問題ごとの情報を、SoundBank Logに表示します。この情報は、SoundBankを生成しながら、Generating SoundBanksダイアログボックスの下の方に表示されるログで見れます。また、SoundBank Logビューを使って、あとからこの情報を確認することもできます。

状況に応じて、特定のメッセージを、SoundBank Logに表示させないような設定もできます。表示させたくないメッセージは、Log Ignore（ログで無視）リストに追加します。Log Ignoreリストについては[「Log Ignoreリストに、メッセージを追加する」](#)を参照してください。また、メッセージごとに、Logに表示される回数を制限できます。この制限については[「ログに表示されるメッセージの管理」](#)を参照してください。

## 関連トピック

- [「Log Ignoreリストに、メッセージを追加する」](#)
- [「プロジェクトの各種SoundBankの生成」](#)
- [「スクリプトを使用してSoundBankを生成する」](#)

## ログのメッセージの、重大度を変更する

SoundBank Logに表示されるメッセージ、警告（Warning）、エラーについて、設定されている重大度（Severity）を変更したい状況も考えられます。メッセージの重大度は、Project Settingsダイアログボックスの、Logタブで変更できます。

詳しくは[「ログに表示されるメッセージの管理」](#)を参照してください。

## 関連トピック

- [「SoundBank生成中に発生した問題のトラブルシューティング」](#)
- [「プロジェクトの各種SoundBankの生成」](#)
- [「スクリプトを使用してSoundBankを生成する」](#)

## スクリプトを使用してSoundBankを生成する

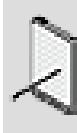
定期的にSoundBankを生成する計画であれば、スクリプトを作成して生成プロセスを自動化することもできます。これを行うには、Wwise SDKを使い、コマンドラインからSoundBankを生成します。この機能の詳しい説明は、Wwise SDKドキュメンテーションの「[Wwise SDK > 応用 > コマンドラインからの SoundBank生成](#)」のセクションを参照してください。

## CopyStreamedFilesツールの利用

CopyStreamedFilesツールは、プロジェクトのストリーム用ファイル（Streamed files）を、SoundBankが保存されている場所にコピーするための、スタンドアロンツールです（デフォルトで、SoundBankは、GeneratedSoundBanksディレクトリの、プラットフォーム別のフォルダに保存されます）。このツールがコマンドラインで呼ばれると、SoundBanksInfo.xmlファイルをスキャンして、ストリーム用ファイルを検出し、次にプロジェクトのキャッシュフォルダから、SoundBankディレクトリの適切なフォルダに、コピーします。デフォルトで全てのプロジェクトに、このタスクを実行するためにCopyStreamedFilesツールを呼び出す、ポストジェネレーション（生成後）ステップのコマンドラインが入っていますが、プロジェクトの具体的なニーズが満たされるように、このステップを変更したり、新しいステップを作成したりできます。

CopyStreamedFilesツールは、Wwiseの実行可能ディレクトリにあります。このツールをコマンドラインから呼び出す時に、SoundBanksInfo.xmlの場所やストリーミングファイルをコピーする場所などいくつかのパラメータを定義する必要があります。コマンドライン内で使用できるパラメータのリストを、下表に示します。

Parameter	内容
-info <file path>	SoundBanksInfo.xmlファイルの場所を指定する。このオプションは、Wwiseでポストジェネレーションステップとして使う時は、"\${InfoFilePath}"（クオーテーションマーク付き）と設定する。
-outputpath <folder path>	ストリーム用ファイルをコピーするベースフォルダを指定するが、これはSoundBankが生成されるベースフォルダと同じにするべきである。Wwiseでポストジェネレーションステップとして使う時は、"\${SoundBankPath}"（クオーテーションマーク付き）と設定する。
-languages "language1 languageN"	コピーするランゲージのリストを指定する。 <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <b>注記</b>            リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオーテーションマークを使うこと。SFXもパッケージするには、ランゲージリストでキーワードとしてSFXを使う。         </div> <p>この引数は、オプショナル設定である。デフォルトとして、全てのランゲージがコピーされる。</p>
-banks "soundbank1 soundbankN"	ストリーミングファイルをコピーする必要のあるSoundBankのリストを指定する。名前に、bnk拡張子は含まれない。または、SoundBank名のリストが入ったテキストファイルを指定するには、拡張子TXTを含むテキストファイルのフルパスを使用することもできる。

Parameter	内容
	 <p><b>注記</b> リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。</p>
-verbose	この引数は、オプショナル設定である。デフォルトとして全てのストリーミングファイルがコピーされる。
-verbose	追加コンソールのテキストアウトプットを可能にする。この引数は、オプショナル設定である。

## 関連トピック

- 「SoundBankのカスタム属性を設定する」
- 「プロジェクトの各種SoundBankの生成」
- 「36章ファイルパッケージの管理」

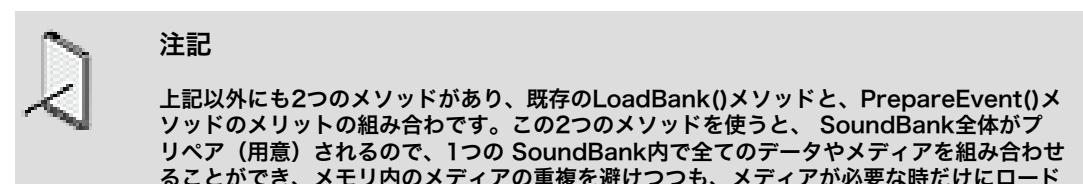
## SoundBank管理の戦略

ゲームのSoundBankを生成して実装するための様々なメソッドを、以下のセクションで説明します。1つのゲームで、メソッドを1つだけ選ぶことも、組み合わせて使うこともできます。全てのゲームが違うので、ゲームの具体的な要件に従い、SoundBank生成のメソッドを選んでください。

SoundBankの作成に関する様々な選択肢がある中、何を選ぶかによって、ゲームのオーディオアセットやモーションアセットの管理に必要な作業量が大幅に増減し、ゲームのパフォーマンスにも直接影響します。サウンドデザイナーとサウンドプログラマーの両者が、以下の各セクションを注意深く読み、どのような選択肢があるのかを充分に理解することが、非常に大事です。お互いに協力することで、ゲームの具体的なニーズを満足させる戦略を立てることができます。なお、どのソリューションを選ぶこともできますが、戦略を決めるにあたり、ゲームのメモリ使用量、I/Oアクセス、そしてゲームへの実装の複雑さ、などを考慮してください。どのメソッドにもメリットやデメリットがあり、多くの場合、メモリ使用量と実装のしやすさのバランスをみながら、決断することになるでしょう。

以下のセクションで、以下のメソッドを説明します。

- 「メソッドNo.1: "All-in-one"バンク」
- 「メソッドNo.2: 複数のコンプリートなSoundBank」
- 「メソッドNo.3: メディアのマイクロマネジメント」
- 「メソッドNo.4: アクションイベントをプリペアする」
- 「メソッドNo.5: イベントやゲームシンク (Switch, State) をプリペアする」



されます。バンクのプリペア (Preparing banks) については、Wwise SDKドキュメンテーションの「[サウンドエンジン統合のウォークスルー > Wwise エレメントのゲームへの統合 > バンクの統合 > 統合の詳細 - バンク > バンクのロード > バンクの準備](#)」を参照してください。



### オフライン解凍

Vorbisオーディオフォーマットではメディアファイルを事前に解凍しておけます。その結果SoundBankのサイズが大きくなりますが、メディアをコールした時に解凍する待ち時間を省けるのでアクセス時間が短縮されます。

この戦略を他のアプローチと組み合わせながら、アセット管理を最適化できます。

## メソッドNo.1: "All-in-one"バンク

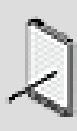
このメソッドは、以下の場合に適しています。

- ・ゲームのオーディオアセットやモーションアセットの数が、限定されている。
- ・ゲームのフリーメモリが、大量にある。

もちろん、無駄使いできるメモリのあるゲームは滅多にありませんが、このテクニックの大きなメリットは、使い方と管理が非常にシンプルだということです。このテクニックを使う主な理由は、1つのWwiseプロジェクトを全て、最短時間でゲームに実装できることです。

### Wwiseで"All-in-one"バンクを作成するには

1. 1つのSoundBankを作成して、適切な名前を付ける。
2. • このSoundBankを、SoundBank Editorにロードして、以下のアイテムを、Addタブにドラッグする。
  - “\Actor-Mixer Hierarchy\Default Work Unit”
  - “\Events\Default Work Unit”
  - “\Interactive Music Hierarchy\Default Work Unit” (ただし、インタラクティブミュージックを使う場合のみ)
  - “\Dynamic Dialogue\Default Work Unit” (ただし、ダイナミックダイアログを使う場合のみ)
  - デフォルトで、Event、Structure（構造）、Mediaの各列のチェックボックスが、全て選択されます。このメソッドは全てを1つのSoundBankに入れるので、チェックボックスの選択をそのままにします。



### 注記

新しいプロジェクトにあるのは、Default Work Unitだけです。もしこれ以外のWork Unitを作成した場合は、必要に応じてこれらのWork UnitもSoundBankに入れます。

3. この1つのSoundBankに全てのWork Unitを追加できたら、SoundBankを生成して、生成されたバンクフォルダをゲームアプリケーションにコピーする。

## "All-in-one" SoundBankを、ゲームに実装するには:

1. 今回の例のゲームのSoundBankは1つだけなので、ゲームを初期化する時にこれをロードするだけである。もちろん、最初にサウンドエンジンを正しく初期化する必要がある。

"All-in-one" SoundBankをゲームにロードする方法として、以下のサンプルコードを参照してください。

```
// Initialize the sound engine here.
...
// Load the Init bank and the "All in one" SoundBank.
AKBankID bankID; // Not used in this sample.
AKRESULT eResult = AK::SoundEngine::LoadBank( L"Init.bnk", AK_DEFAULT_POOL_ID, bankID );
if( eResult == AK_Success )
{
    eResult = AK::SoundEngine::LoadBank( L"MyAllInOneBank.bnk", AK_DEFAULT_POOL_ID, bankID );
}
...
...
```

### その他

"All-in-one" SoundBankを利用するメリットとデメリットを下表に示します。

メリット	デメリット
<p>サウンドデザイナーにとって、バンクのコンテンツ（中身）が最も管理しやすい。</p> <p>ゲームが常に同じSoundBankをロードするので、あるSoundBankのコンテンツを変更しても、ゲームのリコンパイルが不要。</p> <p>複雑なSoundBankのロードやアンロードが、ゲーム中に不要。</p> <p>どのサウンドが使って、どれが使えないかを知るために、ゲーム中にトラッキングする必要がない。</p>	<p>ゲームの全てのイベント、構造、そしてメモリに保存するメディアが、全て常にロードされた状態のため、メモリ使用効率が悪い。</p>

このメソッドは、ゲームのオーディオやモーションを素早く簡単に実装するには良い方法ですが、プロジェクトの終盤になってから、ゲームのメモリをより効率的に活用できるメソッドに変更することは、避けてください。

## メソッドNo.2: 複数のコンプリートなSoundBank

このメソッドは、以下の場合に適しています。

•

このメソッドは、シングルプレイヤーゲームなど、可能性のある全てのサウンドやモーションが、ゲーム中のプレイヤーの現在地のみで決まるゲームに適しています。

まず最初に、SoundBankの分割方法を決めます。例えば、以下のように SoundBankを分割することができます。

- 1つの汎用SoundBankに、ゲーム中に発生する可能性がある、全てのイベントを入れる。このSoundBankは、常にメモリにロードされている。

- レベルごとにSoundBankを1つ、または環境ごとにバンクを1つ、設定する。メインキャラクターの現在地によって変わるサウンドやモーションを入れる。
- さらに、ゲームの独自要件に基づいて、他のSoundBankをいくつか用意しても良い。

## Wwiseで、コンプリートな（全てそろった）SoundBankを複数、作成するには:

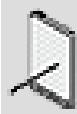
1. ゲームに必要なSoundBankを作成して、例えば、「CommonEvents（共通イベント）」「Level\_1」「Level\_2」「Level\_3」などそれに適切な名前を付ける。
2. Wwiseで、イベントを、複数の仮想フォルダに分ける。1つのSoundBankに対して、1つの仮想フォルダを作成して、各仮想フォルダを該当するSoundBankにドラッグする。サウンドバンクに仮想フォルダを追加することで、プロジェクトに新しいイベントが追加されても、その都度サウンドバンクのコンテンツを編集する必要がない。仮想フォルダの中身に変更があれば、そのSoundBankが自動的に更新される。
3. 全てのイベントを、適切なSoundBankに追加する。この手順は、元のフォルダにないイベントがあれば、それを追加するための手順である。あるイベントを複数のSoundBankに入れる必要があれば、必要な全てのSoundBankにそのイベントを追加すれば良い。
4. これらのSoundBankを生成して、生成されたSoundBankフォルダを、ゲームアプリケーションにコピーする。

## 複数のコンプリートなSoundBankを、ゲームに実装するには:

1. ゲーム中に、適切なタイミングで適切なSoundBankをロードすれば良い。例えば、ゲームの最初に汎用SoundBank（General SoundBank）をロードして、他のSoundBankは、プレイヤーのゲーム中の実際の位置に基づいて、ロードできる。なお、ゲームによっては、レベル間のトランジションのために、複数のレベルを同時にロードするので、多くのメモリが必要なものもある。

このようなSoundBankをゲームにロードする方法として、以下のサンプルコードを参照。

```
// Initialize the sound engine here.  
...  
  
// Load the Init bank and the Common SoundBank.  
AkBankID bankID; // Not used in this sample.  
AKRESULT eResult = AK::SoundEngine::LoadBank( L"Init.bnk", AK_DEFAULT_POOL_ID, bankID );  
if ( eResult == AK_Success )  
{  
    eResult = AK::SoundEngine::LoadBank( L"MyAllInOneBank.bnk", AK_DEFAULT_POOL_ID, bankID );  
}  
  
...  
// And at various places in the code, based on the actual needs:  
eResult = AK::SoundEngine::LoadBank( L"Level_1.bnk", AK_DEFAULT_POOL_ID, bankID );  
...  
eResult = AK::SoundEngine::LoadBank( L"Level_2.bnk", AK_DEFAULT_POOL_ID, bankID );  
...  
eResult = AK::SoundEngine::LoadBank( L"Level_3.bnk", AK_DEFAULT_POOL_ID, bankID );  
...  
eResult = AK::SoundEngine::LoadBank( L"Level_1.bnk" );  
...  
eResult = AK::SoundEngine::LoadBank( L"Level_2.bnk" );  
...  
eResult = AK::SoundEngine::LoadBank( L"Level_3.bnk" );
```



## 注記

LoadBank()コマンドの代わりに、AkBankContent\_Allコマンドを使って SoundBankをプリペア(用意)することもできます。SoundBankをプリペアするこのメリットは、メモリに入っているメディアの重複を避けられることです。Wwise SDKドキュメンテーションの「Preparing Banks」セクションを参照してください。

## その他

Multiple Complete SoundBankを利用するメリットとデメリットを下表に示します。

メリット	デメリット
"All-in-one" SoundBankのテクニックより、メモリ使用量が大幅に削減される可能性がある。 ゲームに簡単に実装できる。	オンラインゲームや、イベントベースのゲームなど、オーディオ要件やモーション要件が、メインキャラクターの現在地などの単純な事実以外で変わるゲームには、適していない。 重複したデータがSoundBankに入っていれば、メモリにロードされたメディアファイルが重複する可能性がある。 異なるSoundBankに似たようなコンテンツが入っていることがあるので、ディスク上のSoundBankの合計スペースが、増える可能性がある。

## メソッドNo.3: メディアのマイクロマネジメント

このメソッドは、以下の場合に適しています。

- ・ゲームのメモリに保存するアセットが、大量にある。
- ・サウンドデザイナーが、ゲームのメディア要件を事前に予測するのが困難である。
- ・あるイベント用に再生するサウンドやを決めるのに、SwitchやStateを活用するプロジェクトである。
- ・サウンドを、定義されたセクションに簡単に分割することができない。

中には非常に複雑なゲームもあり、サウンドやモーションをトリガーする時に様々な要因が関与し、例えば、ゲームに出てくる素材、ゲーム内の時間、ゲームオブジェクトの動き、さらにオンラインマルチプレイヤーゲームの場合は、他のプレイヤー達のアクションまでもが、関係することもあります。イベントベース、またはオブジェクトベースの環境では、サウンドをロードするタイミングを、他のゲームオブジェクトとの距離関係に基づいて決めることができます。例えば、ロードする複数のSoundBankのリストを、ゲームオブジェクトごとに設定して、そのゲームオブジェクトが一定の範囲内にある時、または単にゲーム中で存在する時に、これらのSoundBankをロードするのも、1つの方法です。

また、SwitchやStateによって、再生するオブジェクトが変わります。オブジェクトを再生させるイベントを、SoundBankに追加する時に、再生される可能性のある全てのメディアも自動的に追加されます。例えば、「Play\_Footstep (足音を再生)」という1つのイベントがあるとします。Switchの切り替えによって現在の地面素材が指定され、これに基づいてイベントが適切なサウンドを再生します。こ

の方法は確実ですが、例えば、ゲームプレイがロンドンのビルの中で進行しているのに、「footstep\_sand.wav（砂の足音）」や「footstep\_winter.wav（冬の足音）」などのサウンドをメモリに入れておくことは、無駄になりかねません。

この場合、メモリを無駄使いしないために、1つのSoundBankにイベント構造やオブジェクト構造を追加して、次に、各SoundBankにどのオブジェクトをいれるのかを指定します。例えば、様々な地面の上の足音の例を考えて見ましょう。この場合、以下のSoundBankを作成します。

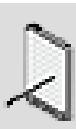
- ・「EventBank（イベントバンク）」：「Play\_Footstep」のイベントと構造を入れたバンク。
- ・「Winter\_Footstep\_bank（冬の足音バンク）」：ゲームの一部（冬）だけで発生する足音用のメディアを入れたバンク。
- ・「Desert\_Footstep\_bank（砂漠の足音バンク）」：ゲームの一部（砂漠）だけで発生する足音用のメディアを入れたバンク。
- ・「Common\_Footstep\_bank（一般足音バンク）」：どこでも発生する足音（床板、コンクリートなど）用のメディアを入れたバンク。

#### メディアをマイクロマネージ（細かく分別）するSoundBankを、Wwiseで作成する

例えば、ゲームに3種類の地面があるとします（雪、砂、コンクリート）。Wwiseでは、Switch Containerを1つ設定して、Switch「ground\_texture（地面素材）」に従い、3つのRandom Containerの1つを、再生します。3つのRandom Containerには、指定された地面の足音サウンドとして、それぞれに4種類のサウンドが入っています。

#### メディアをマイクロマネージ（細かく分別）するSoundBankを、Wwiseで作成するには：

1. “EventBank”というSoundBankを1つ作成して、SoundBank Editorにロードする。
2. イベント“Play\_Footstep”を、SoundBank Editorの、Addタブにドラッグする。
3. 本イベントのMedia列のチェックボックスを外し、Events列とStructures列のチェックボックスは、選択したままにする。
4. 3種類の地面素材に対して、SoundBankを1つずつ作成する。
5. 3つのSoundBankに、それぞれの地面素材に関連するRandom Containerを、1つずつドラッグして入れる。



##### 注記

3つのRandom Containerをドラッグ&ドロップする代わりに、個々のサウンドをドラッグ&ドロップして入れることもできます。ただし、コンテナを使うと、コンテナの中身が変更されても、コンテナ内のサウンドが全て自動的にSoundBankに追加され、マニュアル操作で変更点を修正しなくて済む、というメリットがあります。

6. 地面素材の3つのSoundBankは、それぞれEvents列とStructures列のチェックボックスを外し、Media列のチェックボックスだけを、選択したままにする。
7. これらのSoundBankを生成して、生成されたSoundBankフォルダを、ゲームアプリケーションにコピーする。

この時点では、バンクは合計4つあり、最初のSoundBankの中に、再生するオーディオに関するイベント（Event）データと構造（Structure）データがあり、それ以外の3つのSoundBankには、それぞれ1つの地面素材に関するメディア（Media）だけが入っている。

マイクロマネージ戦略に基づき、ゲームにSoundBankを実装するには：

1. ゲーム中に、共通SoundBankをゲームの冒頭でロードすれば、あとは、他のSoundBankを必要な時にロードするだけとなる。例えば、ゲームの始めに、イベントのSoundBankと、一般足音のSoundBankをロードすれば、その他のSoundBankは、ゲーム中のプレイヤーの実際の位置に合わせて、ロードできる。

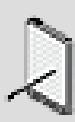
このようなSoundBankをゲームにロードする方法として、以下のサンプルコードを参照。

```
// Load Init and the Eventbank.
AkBankID bankID; // Not used in this sample.
AKRESULT eResult = AK::SoundEngine::LoadBank( L"Init.bnk", AK_DEFAULT_POOL_ID, bankID );
if( eResult == AK_Success )
{
    eResult = AK::SoundEngine::LoadBank( L"Eventbank.bnk", AK_DEFAULT_POOL_ID, bankID );
}
if( eResult == AK_Success )
{
    eResult = AK::SoundEngine::LoadBank( L"Common_Footstep_bank.bnk", AK_DEFAULT_POOL_ID, bankID );
}

...
// And at various places in the code, possibly based on the location:
eResult = AK::SoundEngine::LoadBank( L"Winter_Footstep_bank.bnk", AK_DEFAULT_POOL_ID, bankID );
...
eResult = AK::SoundEngine::LoadBank( L"Desert_Footstep_bank.bnk", AK_DEFAULT_POOL_ID, bankID );
...
eResult = AK::SoundEngine::LoadBank( L"Level_3.bnk", AK_DEFAULT_POOL_ID, bankID );
...
eResult = AK::SoundEngine::UnloadBank( L"Winter_Footstep_bank.bnk" );
...
eResult = AK::SoundEngine::UnloadBank( L"Desert_Footstep_bank.bnk" );
```

## その他

ここでは具体的な例を1つあげましたが、このテクニックは他にも多様な使い方があります。SoundBankごとに、入れるオブジェクトやイベントを1つずつ決めるので、全てのSoundBankの中身を完全にコントロールできます。なお、ゲームの全てのオブジェクトに対して、SoundBankを1つずつ作成することができますが、新しいサウンドやができるたびに、そのSoundBankをゲームの適切な場所でロードするために、新たなコードが必要となり、常に維持するのが非常に難しくなります。どのゲームでも、設定の細かさと、ゲームへの実装のしやすさの間で、ちょうど良いバランスを見つけることが最終目的となります。



### 注記

サウンドを1つ1つロードする方法を検討しているのであれば、SoundBank全体をロードするのではなく、イベントのプリペア（用意）機能の方が、適しているかもしれません。イベントのプリペアについては「メソッドNo.4: アクションイベントをプリペアする」を参照してください。

メディアをマイクロマネージするメリットとデメリットを、下表に示します。

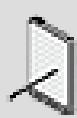
メリット	デメリット
メモリ使用の最適化には、最も適している。 ゲーム中のどの時点に、どのメディアをロードするのかを、完全にコントロールできる。	どのSoundBankをいつロードするのかを判断するには、サウンドデザイナーとゲーム開発者の間で緊密な連絡が必要となる。

#### メソッドNo.4: アクションイベントをプリペアする

このメソッドは、以下の場合に適しています。

- メモリの使用量を抑えるには、メディアを非常に細かく分ける必要がある。
- どのメディアアセットをどのSoundBankにアサインするのかを管理する手間を省きたい場合。

プリペア (Prepare / 用意) されたアクションイベントとは、いったい何でしょうか。 [PrepareEvent\(\)](#) 機能を呼び出す時に、システムがそのアクションイベントを分析して、このイベントに関連する全ての構造 (Structure) やメディア (Media) がメモリにロードされているかを、確認します。ロードされていないければ、システムが自動的に不足情報をディスクからストリーミングします。イベントのプリペア (用意) 状態は、プリペア状態を明確に解除するまで、続きます。



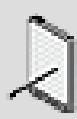
##### 注記

事前にプリペア (用意) できるのは、アクションイベントだけです。 [PrepareEvent](#) (プリペアイベント) 方式は、ダイアログイベントに使えません。

この方式で最低1つのSoundBankを作成する必要がありますが、構造部分はイベントと同じバンクに入れることも全く別のSoundBankに入れることもできます。

[PrepareEvent](#) メカニズムを使用するSoundBankを構築するときの条件は、必要なイベントや構造が必ず1つ以上のSoundBankに存在することと、個別のメディアアセットがファイルシステムでアクセスできることです。なお、構造データを複数の SoundBank に分割して入れると、メモリ効率が改善されます。

アクションイベントをプリペアする前に、そのイベント自体が、SoundBankからメモリにロードされた状態でなければなりません ([LoadBank\(\)](#)を使う)。これは、イベントの中に、そのイベントをプリペアするための前提条件が入っているからです。



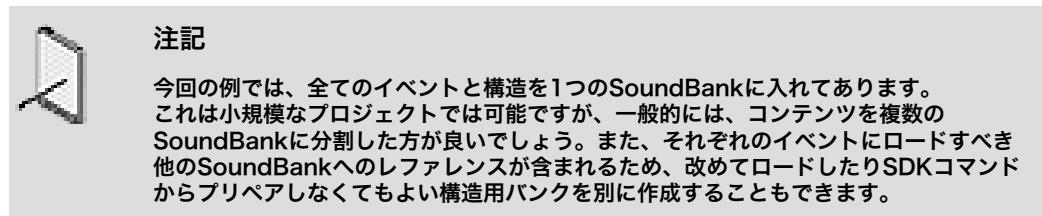
##### 注記

[AK::SoundEngine::PrepareBank](#)と組み合わせて、イベントをプリペアすることも可能です。 [PrepareBank](#) メカニズムを使う最大の利点は、バンクをイベント用バンクと、メディア用バンクに分ける必要がないことです。このメソッドでは、全てのコンテンツを同じ SoundBankに入れますが、[AK::SoundEngine::PrepareBank](#)が呼び出された時に、メモリにロードされるのはバンクのメタデータコンテンツだけです。ゲーム側でメディアが必要になれば、[PrepareEvent](#)を使ってロードできます。

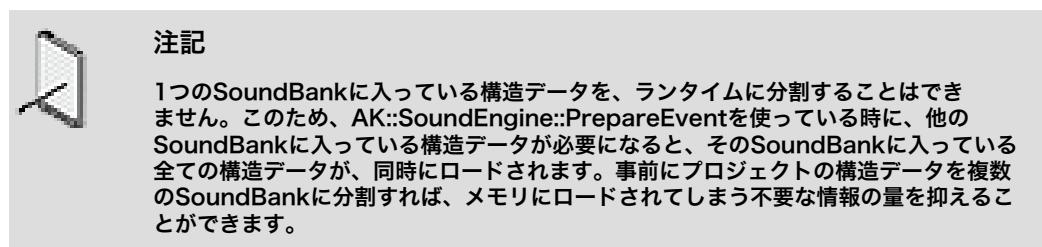
イベントをプリペアする場合に、SoundBankを設定するには:

- “Event”というSoundBankを1つ作成して、SoundBank Editorにロードする。
- 「Event」 SoundBankに、プロジェクトのアクションイベントをいくつか追加するか、単純にイベントのWork Unitをいくつか、追加する。

3. **Media** 列のチェックボックスを外し、**Events**列と**Structures**列のチェックボックスだけを選択したままにする。 **PrepareEvent()**を使用する場合は、メディアをバンクでなくディスクから直接ロードする必要がある。



4. これらのSoundBankを生成して、生成されたSoundBankフォルダを、ゲームアプリケーションにコピーする。



### ゲーム中にイベントをプリペアするには:

1. ゲームの最初にイベントのSoundBankをロードして、イベントのプリペア（用意）を、そのイベントがゲームで必要となった時に行う。関連する構造やメディアは、自動的にロードされる。

ゲーム中にイベントをプリペアする方法として、以下のサンプルコードを参照。

```
// Initializing the sound engine.

AkInitSettings initSettings;
AkPlatformInitSettings platformInitSettings;
AK::SoundEngine::GetDefaultInitSettings( initSettings );
AK::SoundEngine::GetDefaultPlatformInitSettings( platformInitSettings );

// Set the required settings.

...
// Set PrepareEvent related settings.
initSettings.bEnableGameSyncPreparation = false; // Not used in the current sample.

// (Optional) Allocate a memory pool into which prepared media will be loaded.
// If this is not done, memory will be allocated directly in the default memory pool.
{
    initSettings.uPrepareEventMemoryPoolID = AK::MemoryMgr.CreatePool( NULL, 4*1024*1024, 1024, AkMalloc );
}

// (Optional) Give the memory pool a name. This can be very useful for profiling.
AK::MemoryMgr::SetPoolName( initSettings.uPrepareEventMemoryPoolID, "PrepareEventPool" );

}

AKRESULT eResult = AK::SoundEngine::Init( initSettings, platformInitSettings );
if( eResult != AK_Success )
{
    // Handle error.
}

// Load Init bank and the event/structure bank.
AkBankID bankID; // Not used in the sample.
AKRESULT eResult = AK::SoundEngine::LoadBank( L"Init.bnk", AK_DEFAULT_POOL_ID, bankID );
if( eResult == AK_Success )
{
    eResult = AK::SoundEngine::LoadBank( L"Events.bnk", AK_DEFAULT_POOL_ID, bankID );
```

```

}

...
// And then, at various points in the code:
AkLpCtstr pEventsNameArray[1] = { L"My_Event_Name" };

// Preparing an event:
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, pEventsNameArray, 1 ); // 1 is the array size.

// Unpreparing an event:
eResult = AK::SoundEngine::PrepareEvent( Preparation_Unload, pEventsNameArray, 1 ); // 1 is the array size.

```

## その他

`AK::SoundEngine::PrepareEvent()` の呼び出し (Call) は、I/Oファンクションの呼び出しとして扱われます。前例では、ブロッキングファンクション (Blocking function) を使いました。`AK::SoundEngine::PrepareEvent()` ファンクションの他のオーバーロードを使い、ノンブロッキングコール (Non-blocking calls) にして、別のコールバックで、完了ノティフィケーション (Completion notification) を回復 (Revive) できます。

Action Eventをプリペアする、メリットとデメリットを、下表に示す。

メリット	デメリット
SoundBankの生成プロセスがシンプルである。 メディアの分割レベルが非常に細かい。 全体的なメモリの節約を維持できる。 プロセスを自動化しやすい。	メディアアセットを1つ1つロードするため、ディスクの読み込みとシークの回数が増える可能性がある。 使用されるメモリの合計量の制御が減少します。 Interactive Music の操作は複雑になります。

## メソッドNo.5: イベントやゲームシンク (Switch、State) をプリペアする

このメソッドは、以下の場合に適しています。

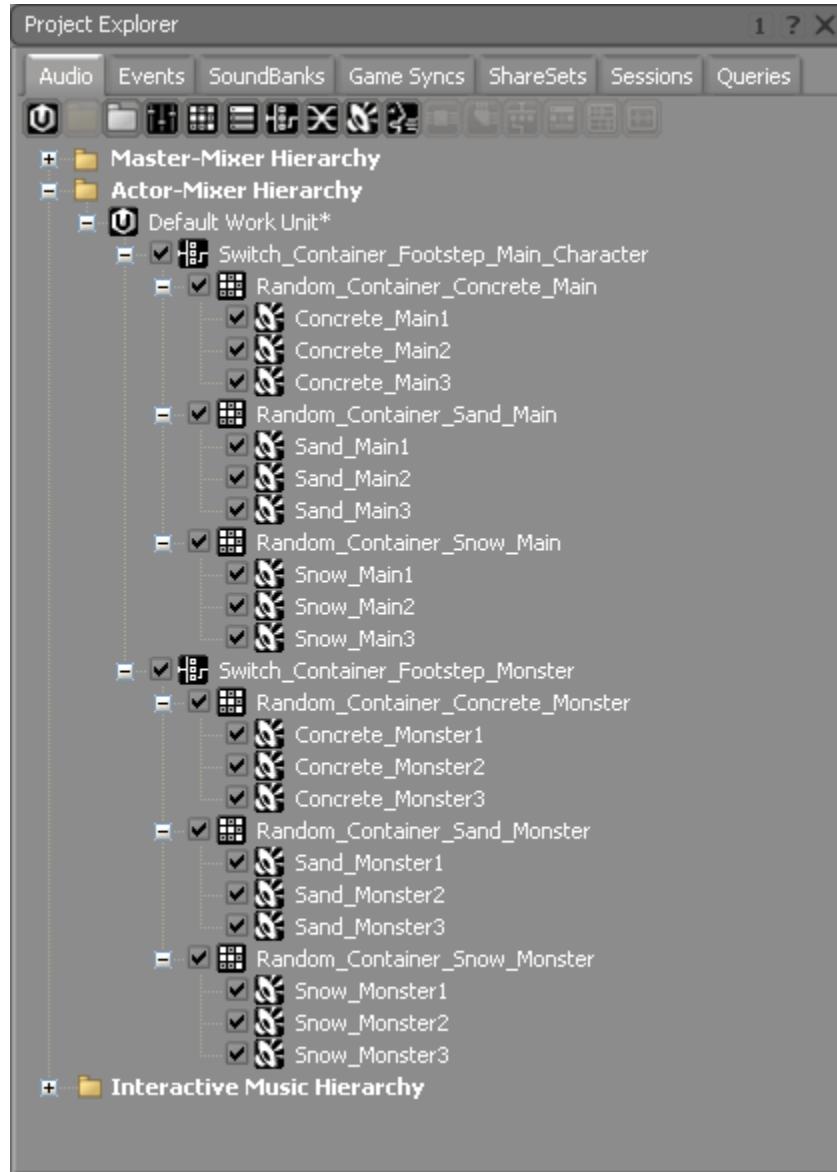
- メモリの使用量を抑えるには、メディアを細かく分ける必要がある。
- メディアアセットを複数のSoundBankに分割する手間を省きたい場合。
- スイッチやステートに基づいて、異なるサウンドを再生するイベントがプロジェクトにある。
- SwitchやStateに基づいてミュージックを再生する、インタラクティブミュージックがプロジェクトにある。

基本的にメソッドNo.4と同じですが、イベントをプリペアする時にロードされるメディアを、より細かくコントロールします。このメソッドでは、プリペアされたイベントと、現在アクティブなゲームシンクの、両方に関連するメディアだけが、メモリにロードされます。

例えば、シンプルなプロジェクトに、2つのイベント 「Play\_Maincharacter\_FootSteps (メインキャラクターの足音を再生)」 と 「Play\_Monster\_Footsteps (モンスターの足音を再生)」 があるとします。それぞれのイベントは、キャラクターが移動中の地面素材に合わせて、異なるSwitch Containerで、異なるサウンドをランダムに再生します。Switch Groupの名前は

「GroundTexture」であり、可能なStateは、「Snow」「Concrete」「Sand」の3つです。

Wwiseに表示されるSwitch Containerの階層は、以下の通りです。



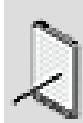
この例では、メモリにロードされる可能性のあるサウンドが、18個（6つのグループに、それぞれ3つのサウンド）あります。そこで[「メソッドNo.4: アクションイベントをプリペアする」](#)を採用できますが、イベントごとにメモリにロードされるサウンドの数、つまり分割の細かさは、6個までとなります。

そこで[「メソッドNo.3: メディアのマイクロマネジメント」](#)を採用して、より細かく分割できますが、このシンプルなプロジェクトに対して、SoundBankが6個も必要になります（現実のプロジェクトでは、SoundBank数がすぐに増えてしまします）。モンスターが出現した時に、どのような地面素材が可能かを確認してから、適切なSoundBankをロードする必要があります。

メソッドNo.5では、発生する可能性のあるイベントとゲームシンクを指定するだけで、あとは適切なメディアだけがロードされます。さらにシンプルにするために、全てのメディアを、1つのSoundBankにまとめることもできます。

### イベントやゲームシンクをプリペアする場合に、バンクを設定するには:

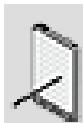
1. “Events”というSoundBankを1つ作成して、SoundBank Editorにロードする。
2. SoundBank Editorの、Addタブに、2つのイベントをドラッグする。
3. **Media**列のチェックボックスを外し、Events列と**Structures**列のチェックボックスだけを選択したままにする。PrepareEvent()を使用する場合は、メディアをバンクでなくディスクから直接ロードする必要がある。



#### 注記

今回の例では、全てのイベントと構造を1つのSoundBankに入れてあります。これは小規模なプロジェクトでは可能ですが、一般的には、コンテンツを複数のSoundBankに分割した方が良いでしょう。また、それぞれのイベントにロードすべき他のSoundBankへのレファレンスが含まれるため、改めてロードしたりSDKコマンドからプリペアしなくてもよい構造用バンクを別に作成することもできます。

4. これらのSoundBankを生成して、生成されたSoundBankフォルダを、ゲームアプリケーションにコピーする。



#### 注記

1つのSoundBankに入っている構造データを、ランタイムに分割することはできません。このため、AK::SoundEngine::PrepareEvent()を使っている時に、他のSoundBankに入っている構造データが必要になると、そのSoundBankに入っている全ての構造データが、同時にロードされます。事前にプロジェクトの構造データを複数のSoundBankに分割すれば、メモリにロードされてしまう不要な情報の量を抑えることができます。

### ゲーム中に、イベントとゲームシンクをプリペアするには:

1. ゲーム側が2つのイベントを必要とする前に、SoundBank「Event」をロードする。
2. 素材「Concrete」を、常にアクティブな状態にしておく。
3. 常に、メインキャラクターの足音イベント（Main character footstep event）を、プリペアする。
4. 地面素材が近くにある時は、ゲームシンクをアクティブにする。

ゲーム中にゲームシンクをプリペアする方法として、以下のサンプルコードを参照してください。まず最初に、サウンドエンジンを初期化して、bEnableGameSyncPreparationフラグなどの必要設定を、設定する。

```
// Initializing the sound engine.

AkInitSettings initSettings;
AkPlatformInitSettings platformInitSettings;
AK::SoundEngine::GetDefaultInitSettings( initSettings );
AK::SoundEngine::GetDefaultPlatformInitSettings( platformInitSettings );

// Set the required settings.
```

## SoundBankの管理

```
...
// Set PrepareEvent related settings.

// The flag bEnableGameSyncPreparation is set to true to activate the prepare
// Game Sync mechanism. When set to false, the media associated with all Game
// Syncs is loaded and there is no need to call AK::SoundEngine::PrepareGameSyncs.
//
// When set to true, no media that is Game Sync dependent will be loaded unless
// the Game Sync is activated by calling AK::SoundEngine::PrepareGameSyncs.
initSettings.bEnableGameSyncPreparation = true;

// (Optional) Allocate a memory pool into which prepared media will be loaded.
// If this is not done, memory will be allocated directly in the default memory pool.
{
    initSettings.uPrepareEventMemoryPoolID = AK::MemoryMgr::CreatePool( NULL, 4*1024*1024, 1024, AkMalloc );
}

// (Optional) Give the memory pool a name. This can be very useful for profiling.
AK::MemoryMgr::SetPoolName( initSettings.uPrepareEventMemoryPoolID, L"PrepareEventPool" );
}

AKRESULT eResult = AK::SoundEngine::Init( initSettings, platformInitSettings );
if( eResult != AK_Success )
{
    // Handle error.
}
```

次に、Initバンクと、イベントのSoundBankをロードする。

```
// Load Init bank and the Event/structure bank.
AkBankID bankID; // Not used in the sample.
AKRESULT eResult = AK::SoundEngine::LoadBank( L"Init.bnk", AK_DEFAULT_POOL_ID, bankID );
if( eResult == AK_Success )
{
    eResult = AK::SoundEngine::LoadBank( L"Events.bnk", AK_DEFAULT_POOL_ID, bankID );
}

// ... At this point, the two Events are loaded, but not prepared.
// No media is currently loaded.
```

次に、必要なイベントやゲームシンクを、プリペアする。

```
AkLpCtstr pNameArray[1];

// Prepare the main character footstep event.
pNameArray[0] = L"Play_Maincharacter_FootSteps";
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, pEventsNameArray, 1 ); // 1 is the array size.

// ... At this point, one Event has been prepared, but no media has been loaded.

// Now, since concrete is always available in the game.
pNameArray[0] = L"Concrete";
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, in_eType, "GroundTexture", pNameArray, 1 );

// ... At this point, the 3 sounds (Sound_Concrete_main_1, Sound_Concrete_main_2, and Sound_Concrete_main_3)
// are loaded.

// Now, let's say that the main character enters a land where there is snow.
pNameArray[0] = L"Snow";
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, in_eType, "GroundTexture", pNameArray, 1 );

// ... At this point, the 3 sounds (Sound_Snow_main_1, Sound_Snow_main_2, and Sound_Snow_main_3)
// just got loaded.

// Then let's say that a monster suddenly appears.
pNameArray[0] = L"Play_Monster_Footsteps";
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, pEventsNameArray, 1 ); // 1 is the array size.

// ... At this point, 6 more sounds ( Sound_Concrete_Monster_1.2.3 and Sound_Snow_Monster_1.2.3 ).

// And now our player decides to run away from the monster, which goes after him.
// They run so far that they arrive at a place where there is no snow.

pNameArray[0] = L"Snow";
eResult = AK::SoundEngine::PrepareGameSyncs( Preparation_Unload, in_eType, "GroundTexture", pNameArray, 1 );
// ... At this point, the 6 snow-related sounds (Sound_Snow_Monster_1.2.3 and Sound_Snow_main_1.2.3 )
// are unloaded from memory.

...
```

## その他

ゲームシンクをプリペアする、メリットとデメリットを、下表に示す。

メリット	デメリット
バンクの生成プロセスがシンプルである。 メディアの分割レベルが非常に細かい。 全体的なメモリ使用量を低く抑えられる。 プロセスを自動化しやすい。 使う可能性のあるメディアだけが、ロードされる。	メディアアセットを1つ1つロードするため、ディスクの読み込みとシークの回数が増える可能性がある。 合計メモリ使用量をコントロールしにくい。 1つのゲームシンクをアクティブにした時に、データを新たにロードしなくてはならないイベントを大量にプリペアする場合は、ストリーミング帯域が大幅に増える。

## SoundBankに関する、コツとベストプラクティス

SoundBankのコンテンツを決める前に、以下のセクションに目を通して、ゲームのSoundBankを効率的に管理するための、コツやベストプラクティスを参照してください。

### 初期化バンク (Initialization Bank) の更新

デフォルトで、SoundBankが1つ以上、生成されると、初期化バンク (Init.bnk) が更新されます。これは、この2種類のSoundBankを確実に同期させるためです。初期化バンクは、プロジェクトの一般的な情報を全て入れたバンクで、バス階層、State、Switch、RTPC、環境エフェクトなどの情報が含まれます。プロジェクトのこれらのエレメントに変更があった場合は、Wwiseが新しく生成された SoundBankと共にこのSoundBankを更新します。一方、これらのプロジェクトエレメントに変更がない場合は、WwiseはInit.bnkの新バージョンを生成しません。

**初期化バンク (Initialization bank)** - プロジェクトの一般的な情報を全て入れた特別なバンクで、バス階層、State、Switch、RTPC、環境エフェクトなどの情報が含まれる。WwiseがSoundBankを生成すると、初期化バンクが自動的に作成される。通常、初期化バンクはゲームの最初に1回ロードされ、ゲームプレイ中にプロジェクトの全般情報に簡単にアクセスできるようにする。ゲームを開始する時に最初にロードする必要があり、これがロードされないと、他のSoundBankをロードできない場合もある。初期化バンクのデフォルト名は、Init.bnk。

### SoundBankとメモリ

SoundBankに複数の長いサウンドを入れると、プラットフォームのメモリ消費量が増えてしまいます。メモリを消費しすぎないように、長いサウンドやミュージックファイルは、ストリーミングしてください。サウンドやミュージックオブジェクトのストリーミングについては「[サウンドのストリーミング](#)」や、「[ミュージックのストリーミング](#)」を参照してください。

### 似たエレメントは、SoundBankでまとめる

同じようなエレメントを、1つのSoundBankにまとめるようにします。例えば、ゲーム中に常にロードされたままとなる、メニューやメインキャラクターなどのエレメントは、1つのSoundBankにまとめます。一方、ゲームのプレイ中にロードされたりアンロードされたりするエレメントに関しては、それぞれのグループを論理

的なユニット、つまり基本ブロックにまとめて入れ、ゲーム中に交換したり置き換えたりします。

### Work Unitやフォルダを使う

プロジェクトに変更を加える度にSoundBankを編集する手間を省くには、各種 SoundBankのセットアップと同じ醉うに、フォルダやWork Unitをセットアップします。Wwiseでは、SoundBank内のエレメントと、プロジェクト内のエレメントの間にアクティブなリンクを維持するので、これらのフォルダを一旦 SoundBankに追加すると、SoundBankが自動的に更新され、編集作業は一切、不要となります。

### SoundBankで、イベントIDを使う

プロジェクトの最終段階では、イベントのIDは、処理する前にサウンドエンジンが名前をハッシュする必要がなくより速く検証できるので、イベントの文字列ではなくIDを使うようにします。

### インテグレティレポートの生成

SoundBankを生成する前に、インテグレティレポートを生成します。インテグレティレポートに、プロジェクトのエラーリストと、解決のための提案が表示されます。 SoundBankを生成する前に全てのプロジェクトエラーを解決することで、ゲーム中のオーディオやモーションの問題発生を、抑えられます。

### SoundBank Editorで、複数項目を一括選択する

SoundBank Editorの、Addタブで作業をしている時に、複数の項目で、同じタイプのプロジェクトエレメントを除外する必要がある時は、Hierarchy Inclusion（階層を含める）リストで、複数の項目を選択してから、1つのチェックボックスを外します。これで、選択した項目の全てにおいて、このタイプが除外されます。また、複数選択をしたアイテムに対して、一括で1つのタイプを含めることもできます。

### SoundBank名の制約

プラットフォーム側に、ファイルシステムによるファイル名の制限がある場合は、SoundBankを、ファイルパッケージにパッケージします。ファイルパッケージは完結したユニットなので、プラットフォームのファイルシステムを完全に抽象化して、システムのファイル名に関する制限を回避できます。

### DLCの注意点と制約

DLCで期待通りのサウンド構造を実現するには、変更や追加が既存のリリース構造にどう影響するかを理解する必要があります。特に注意を払うべき一般的な状況を、以下に詳しく説明します。

#### 子の追加

DLCの中のオブジェクトによっては、それに子を追加するためには既存 SoundBankを完全に再パッケージしてDLCに含める必要があります。ただし、オブジェクトによっては子を追加するのに、DLCの追加SoundBankを使って追加できます。SoundBankを改めて再パッケージする必要のあるオブジェクトと、別の

SoundBankをDLCで追加して子を追加できるオブジェクトを、それぞれ下表に示します。

DLCで子を追加するために、SoundBankを再度パッケージする必要のあるオブジェクト	DLCで子を追加するために、別のSoundBankを追加するオブジェクト
<ul style="list-style-type: none"> <li>• Random Container、Sequence Container</li> <li>• Switch Container</li> <li>• Blend Container</li> <li>• Music Switch Container</li> <li>• Music Playlist Container</li> <li>• Music Segment</li> </ul>	<ul style="list-style-type: none"> <li>• Actor-Mixer</li> <li>• Folder</li> <li>• Work Unit</li> </ul>

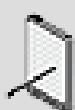
例えば、以下のような階層の場合を考えます。

- WorkUnit
  - ActorMixer
    - ContainerA
      - Sound1
      - Sound2

ContainerAが、メインリリースのSoundBank1に含まれていたとします。DLCで、Actor-Mixer Hierarchyの下に新しいコンテナを問題なく追加できます。

- WorkUnit
  - ActorMixer
    - ContainerA
      - Sound1
      - Sound2
    - ContainerB
      - Sound1
      - Sound2

次に新しくSoundBank2を作成して、これにContainerBを追加するとします。SoundBank1とSoundBank2をロードすると、2つの構造が自動的にマージされます。ただし、DLCでContainerAにサウンドを追加するには、新SoundBankの中にパッケージすることはできません。これを追加するには、SoundBank1を改めて生成して、DLCに含める必要があります。メインリリースのバージョンが、これに置換されます。



#### 注記

DLCリリースが予定されるゲームの場合、Eventや構造と、メディアコンテンツを分けると、DLCサイズに関してサウンドバンクの再パッケージングの影響が小さくなるので、良い対策方法です。

#### Initバンクの変更

DLCでバス、ゲームシンク（Game Parameter、Switch、State）、環境エフェクトなどを変更した場合は、必ずInit.bnkファイルの新バージョンをDLCの中で再パッケージします。

### 複数のDLCをリリースする

複数のDLCを、同時または続けてリリースする場合は、特に注意が必要です。コンテンツが徐々に発展していく場合は、DLCのビルドを以前のDLCプロジェクトの上にビルドする必要があります。

例えば、メインリリース (Init.bnk + RTM.bnkを含む) と、DLC-A (Init.bnk + DLC-A-specific.bnkを含む) 、DLC-B (Init.bnk + DLC-B-specific.bnk) をリリースするとします。バス構造が、RTM、DLC-A、DLC-Bでそれぞれ異なる場合は、DLC-Aに含まれる追加バスは、全てDLC-Bにも入れなければなりません。

3つのビルドに、バスを含むInit.bnkファイルを入れる必要があります。ユーザー側の組み合わせとして、以下が考えられます：

- DLCなし
- DLC-Aのみ
- DLC-Bのみ
- DLC-AとDLC-B

DLC-AとDLC-Bの両方を持っているユーザーの場合、DLC-Bで提供されたInit.bnkがロードされます。DLC-BのInit.bnkは、DLC-Aの上にビルドされているため、DLC-Aに必要なバスも入っています。

---

## 第36章 ファイルパッケージの管理

概要 .....	723
File Packagerプロジェクトの使い方 .....	723
プロジェクト内のファイルパッケージの管理 .....	726
DLC（ダウンローダブルコンテンツ）の概要 .....	730
ファイルパッケージを生成する .....	731
コマンドラインで、File Packager引数を使用 .....	732
ファイルパッケージャに関する、コツとベストプラクティス .....	735

## 概要

File Packagerは、Wwiseプロジェクトで作成した様々なSoundBank、ルースメディア、ストリーム用メディアファイルをファイルパッケージ（単数または複数）にまとめるinandアロンユーティリティです。ファイルパッケージはファイルシステムを抽象化したもので、完結したユニットです。ファイルパッケージをロードすると、メモリにルックアップテーブルが作成されて、PCKファイル内のどこにメディアがあるのかをWwiseに伝えます。つまり、ファイルパッケージを使うことで、プラットフォーム側のファイルシステムにみられる、ファイル名の長さやファイル数などに関する制限をある程度、回避できます。また、ファイルパッケージを使うと、複数のランゲージバージョンや、リリース後に出現するダウンロードコンテンツなどが、管理しやすくなります。

プロジェクトのSoundBankやストリーム用メディアファイルの情報は、SoundBanksInfo.xmlファイルを、File Packagerにインポートする時に、取得されます。SoundBanksInfo.xmlファイルは、SoundBankが生成される度に、Wwiseが自動的に作成するものです。

デフォルトで、全てのファイルが、ファイルパッケージ「default.pck」に追加され、このファイルパッケージは、全てのプラットフォームのSoundBankパスディレクトリ（SoundBank path directory）のルートで作成されます。また、複数の新規パッケージを作成して、それぞれにファイルをマニュアル操作で追加したり、作成したパッケージの保存場所を変えたりできます。

ファイルパッケージは、File Packagerでマニュアル作成できるほか、SoundBank生成の処理の一貫として、自動的にコマンドラインで、File Packagerを起動することもできます。このコマンドラインは、プロジェクト全体で設定することも、SoundBankのユーザー設定としてカスタム設定することもできます。SoundBankのポストジェネレーション（生成後）のステップとして、File Packagerをコマンドラインで起動する方法については、以下のセクションを参照してください。

- ・「プロジェクトのSoundBankの設定」
- ・「SoundBankのカスタム属性を設定する」
- ・「コマンドラインで、File Packager引数を使用」

## File Packagerプロジェクトの使い方

File Packagerで1つのセッションを作成すると、1つのプロジェクトとして保存できるので、作業内容を簡単に保存して、あとから見直せます。File Packagerプロジェクトには、1つのセッションで作成した全てのパッケージの情報が保存され、例えばパッケージの個数、ファイルアサインの設定、パッケージごとの中身（コンテンツ）やファイルの順番などの情報が入っています。

File Packagerを開くと、毎回、新しいセッションが自動的に作成されます。このセッションを新規プロジェクトとして保存するか、以前に作業したプロジェクトを1つ、開きます。

## File Packagerを開くには:

1. Startメニューで、All Programs > Audiokinetic > Wwise \_version\_ build \_build\_ > Tools > File Packagerをクリックする。

File Packagerが開き、新規セッションが表示される。

## 関連トピック

- ・「SoundBankの情報を、プロジェクトにインポートする」
- ・「既存のFile Packagerプロジェクトを開く」
- ・「File Packagerプロジェクトを保存する」

### SoundBankの情報を、プロジェクトにインポートする

パッケージを作成して中身を入れ始める前に、Wwiseで各種SoundBankを生成した時に作成される、SoundBankデータをインポートする必要があります。このデータは全て、SoundBanksInfo.xmlファイルに保存され、GeneratedSoundBanksディレクトリの、プラットフォームフォルダにあります。Wwiseでは、プラットフォームごとに、それぞれのSoundBanksInfo.xmlを生成します。

プロジェクトを開くと、毎回、File Packagerが、SoundBanksInfo.xmlファイルを読み込み、Files to package (パッケージするファイル) ビューに表示します。Wwiseで、このSoundBankやストリーム用メディアファイルを変更した場合は、両者のプロジェクトで矛盾が生じる可能性があります。ファイルが不足している場合は、印が付き、赤でハイライト表示されます。

### プロジェクトに、SoundBank情報をインポートするには:

1. File Packagerを開く。
2. SoundBanks Info fileフィールドの横の、Browseボタンをクリックする。

Windowsの、Openダイアログボックスが開く。

3. インポートする、SoundBanksInfo.xmlファイルまで移動して選択して、Openをクリックする。

全てのSoundBankやストリームファイルが、Files to packageビューにロードされる。



## 関連トピック

- ・「File Packagerプロジェクトの使い方」
- ・「既存のFile Packagerプロジェクトを開く」

- ・「File Packagerプロジェクトを保存する」

### 既存のFile Packagerプロジェクトを開く

File Packagerのプロジェクトを保存すると、あとからプロジェクトをまた開いて、作業を続けられます。File Packagerインスタンスを同時に複数、開くことができるので、複数のプロジェクトで同時に作業できます。

### 既存のFile Packagerプロジェクトを開くには:

1. File Packagerを開く。
2. メニューバーで、File > Openをクリックする。

Windowsの、Openダイアログボックスが開く。  
3. File Packagerプロジェクトが保存されているフォルダまで移動する。  
4. .wfpprojファイルを選択して、Openをクリックする。

プロジェクト情報が、File Packagerにロードされる。

### 関連トピック

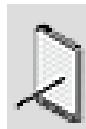
- ・「File Packagerプロジェクトの使い方」
- ・「SoundBankの情報を、プロジェクトにインポートする」
- ・「File Packagerプロジェクトを保存する」

### File Packagerプロジェクトを保存する

File Packager上で作成した1つのセッションは、1つのプロジェクトとして保存できるので、作業を中断して保存して、あとから再開できます。File Packagerのプロジェクトは、File Packagerファイル「.wfpproj」として保存されます。

### File Packagerプロジェクトを保存するには:

1. 以下のいずれかを行う:
  - ・メニューバーで、File > Saveをクリックする。
  - ・Ctrl+Sを押す。
  - ・全てのプロジェクト情報が、.wfpprojファイルとして保存される。



#### 注記

あるプロジェクトを初めて保存する時は、Save Asダイアログボックスが開き、プロジェクト名と保存場所を指定するよう、プロンプトが表示されます。

### 関連トピック

- ・「File Packagerプロジェクトの使い方」
- ・「SoundBankの情報を、プロジェクトにインポートする」
- ・「既存のFile Packagerプロジェクトを開く」

## プロジェクト内のファイルパッケージの管理

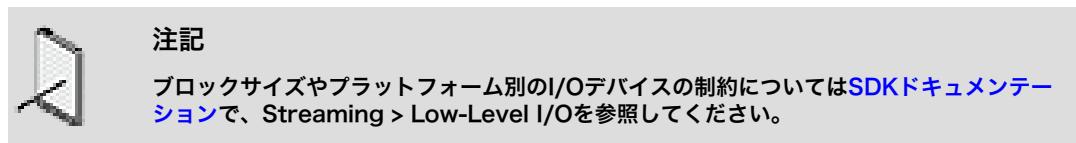
File Packagerプロジェクトを1つ作成して、SoundBank情報やストリームファイル情報をこのプロジェクトにインポートできたら、必要な各種パッケージを作成して、中身を入れます。これが完成したら、これらのパッケージを生成して、ゲームに入れられるようにします。

プロジェクトのファイルパッケージ管理には、以下のタスクが伴います。

- 「プロジェクトにファイルパッケージを追加する」
- 「プロジェクトのファイルパッケージを削除する」
- 「ファイルパッケージの中身を入れる」
- 「パッケージ内のファイルを並べ替える」

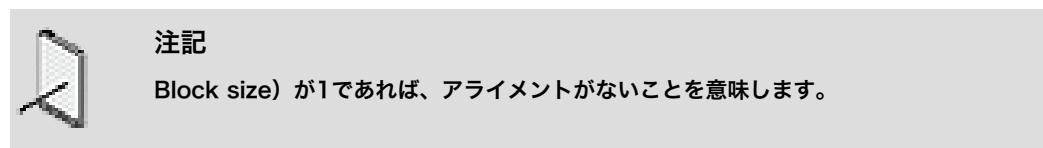
### プロジェクトにファイルパッケージを追加する

1つのプロジェクトに、パッケージを無制限に追加できます。各種パッケージを作成できたら、**Block Size**（ブロックサイズ）プロパティを設定して、パッケージのバイトのアライメント（Byte alignment）を定義します。指定するバイト数は、各プラットフォームのI/Oデバイスの要件によって決まる。



### プロジェクトにファイルパッケージを追加するには:

1. Packagesリストで、Addをクリックする。  
Packagesリストに、新規パッケージが1つ追加される。
2. デフォルトのパッケージ名を、新規パッケージの内容が分かる名前に変更する。
3. Enterをクリックする。
4. 全てのパッケージが作成できるまで、上記1～3を繰り返す。
5. パッケージ内のデータをアラインするバイト数を指定するには、**Block Size**フィールドに数値を入力する。指定するバイト数は、各プラットフォームのI/Oデバイスの要件によって決まる。各プラットフォームの一般的な数値は1です。



### 関連トピック

- 「プロジェクトのファイルパッケージを削除する」
- 「ファイルパッケージの中身を入れる」
- 「パッケージ内のファイルを並べ替える」

- 「ファイルパッケージを生成する」

### プロジェクトのファイルパッケージを削除する

不要となったパッケージは、いつでも簡単にプロジェクトから削除できます。

### プロジェクトから、ファイルパッケージを削除するには:

1. Packagesリストで、プロジェクトから削除するパッケージを1つを選択する。
2. Removeをクリックする。

このパッケージが、プロジェクトから削除される。

### 関連トピック

- 「プロジェクトにファイルパッケージを追加する」
- 「ファイルパッケージの中身を入れる」
- 「パッケージ内のファイルを並べ替える」
- 「ファイルパッケージを生成する」

### ファイルパッケージの中身を入れる

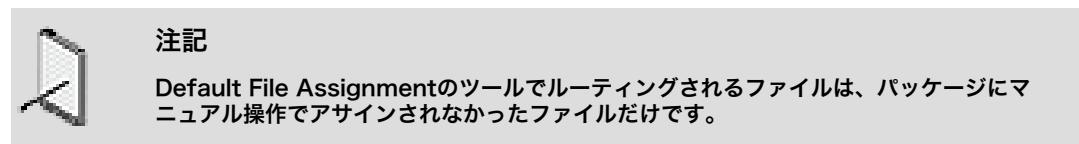
パッケージを作成できたら、それぞれに、様々なSoundBankやストリーム用メディアファイルを入れ始めます。

プロジェクトに入っている各種パッケージに、中身を入れる方法は、以下の2種類があります。

- 「自動的に、ファイルをパッケージにアサインする」
- 「マニュアル操作で、ファイルをパッケージにアサインする」

### 自動的に、ファイルをパッケージにアサインする

あるパッケージに複数のファイルをマニュアル操作で追加していくのは時間がかかるので、File Packagerでは、ランゲージタイプやファイルタイプに基づいて、ファイルのグループを、特定のパッケージに自動的にアサインするいくつかのルールを作成できます。これらのルールを設定することで、パッケージの中身をそろえる時間を大幅に短縮できます。



### アサインされていないファイルを、自動的にパッケージにルーティングするには:

1. File Packagerの、Default file assignmentビューで、リストに表示されたランゲージやSFXごとに、そのSoundBankやストリームファイルをどのパッケージにアサインするのかを、設定する。
2. 残りの全てのSoundBankを、どのパッケージにアサインするのかを、該当するリストからパッケージを1つ選択して設定する。

3. 残りの全てのストリームファイルを、どのパッケージにアサインするのかを、該当するリストからパッケージを1つ選択して設定する。

マニュアル操作で明確に1つのパッケージにアサインしていないファイルは全て、自動的に、既定のパッケージに、アサインされる。

## 関連トピック

- 「マニュアル操作で、ファイルをパッケージにアサインする」
- 「プロジェクトにファイルパッケージを追加する」
- 「プロジェクトのファイルパッケージを削除する」
- 「パッケージ内のファイルを並べ替える」
- 「ファイルパッケージを生成する」

### マニュアル操作で、ファイルをパッケージにアサインする

例えば、リリース後に提供する予定の新規ゲームコンテンツを作成する時など、あるパッケージに、具体的なファイルをいくつか、マニュアル操作で追加したい状況も考えられます。ファイルをマニュアル操作でパッケージに追加すると、ファイルの自動アサイン設定は、全てオーバーライドされます。

マニュアル操作でSoundBankをパッケージに追加する時は、追加するのが SoundBankなのか、SoundBankに関連しているストリームファイルなのか、または両方なのかを、選択します。Package Contentsビューの、Resulting Contents タブに切り替えると、いつでも、そのパッケージに実際に含まれているコンテンツを確認できます。

### マニュアル操作で、ファイルをパッケージに追加するには:

1. Packagesビューで、パッケージを1つ選択する。

Package Contentsビューに、このパッケージのコンテンツ（内容）が表示されます。

2. Files to packageビューから、Package Contentsビューの、Added Itemsタブに、ファイル（単数または複数）を、ドラッグする。

Added Itemsタブのタイトルバーに、このパッケージにマニュアル操作で追加したファイル数が、角括弧で表示される。

また、Resulting contentsタブでも、タイトルバーに数字が表示される。これは、このパッケージに入っているファイルの合計数である。この2つの数字は同じとは限らず、例えばパッケージにSoundBankを1つ、追加して、その中に、複数のストリーム用メディアファイルへのレファレンスが入っている時などは、同じではない。

3. パッケージにSoundBankを1つ追加する時に、パッケージに実際に追加するファイルタイプを決めるために、Inclusion modeリストの以下の選択肢を1つ選択する。

- All files（全ファイル） - このSoundBankファイルと、関連する全てのストリーム用メディアファイルを入れる場合。

- **SoundBank only (SoundBankだけ)** - このSoundBankのファイルだけを入れる場合。
  - **Streamed files (ストリームファイル)** - 関連するストリーム用メディアファイルだけを入れ、追加した実際のSoundBankファイルは、入れない場合。
4. Resulting Contentsタブに切り替えて、このパッケージに実際に入れたコンテンツを確認する。

## 関連トピック

- 「自動的に、ファイルをパッケージにアサインする」
- 「プロジェクトにファイルパッケージを追加する」
- 「プロジェクトのファイルパッケージを削除する」
- 「パッケージ内のファイルを並べ替える」
- 「ファイルパッケージを生成する」

### パッケージ内のファイルを並べ替える

デフォルトとして、あるパッケージ内の全てのファイルに関して、パッケージに追加する時の順番は、特にありません。しかし、ゲーム中のディスクシークを最小限にするために、特定のファイル同士を近づけたい場合も考えられます。パッケージ内のファイルの順番は、File Order Editorを使って並べ替えることができます。

パッケージ内の全てのファイルを並べ替えるのは、時間もかかり不要なので、File Packagerでは、順番を決める必要のあるファイルだけを、並べ替えることができます。残のファイルは全て、「Remaining files inserted here (残りのファイルはここに挿入)」というプレースホルダを使って、順番リストに自動的に追加されます。順番リストに、マニュアル操作でファイルを1つ追加すると、このファイルは、自動的にプレースホルダから外されます。

### パッケージ内のファイルを並べ替えるには:

1. 以下のいずれかを行う:

- Packagesリストで、パッケージを1つ、ダブルクリックする。
- Packagesリストで、パッケージを1つ選択して、Edit file order (ファイルの順番変更) をクリックする。

File Order Editorが開く。

2. Package filesビューで、マニュアル操作で順番に並べるファイル（単数または複数）を選択する。

3. Add to ordered list (順番リストに追加) ボタンをクリックする。

選択したファイルが、順番リストで次に空いている位置に追加される。



#### Tip

Package filesビューから、順番リストの中の特定の位置に、ファイルを直接ドラッグすることもできます。

4. リストの中にある1つのファイルの順番を変えるには、このファイルを順番リストの新しい場所までドラッグする。  
挿入ポイントが、横線で表示される。
5. パッケージ内の各種ファイルの最終的な順番が完成するまで、ファイルの順番の調整を続ける。



#### Tip

File orderリストからファイルを削除するには、削除するファイル（単数または複数）を選択して、Remove from ordered list（順番リストから外す）をクリックします。

### 関連トピック

- 「プロジェクトにファイルパッケージを追加する」
- 「プロジェクトのファイルパッケージを削除する」
- 「ファイルパッケージの中身を入れる」
- 「ファイルパッケージを生成する」

## DLC（ダウンローダブルコンテンツ）の概要

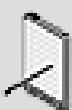
DLC（Downloadable content）は一般的に、ゲームのメインビルド（DLCのない最初のゲームリリース）に追加されるオーディオコンテンツのパッケージを指します。

ゲームのプロダクションフェーズに、WwiseでSoundBankやストリームファイルが作成されます。File PackagerとDefault file assignmentを使い、全てのSoundBankやストリームファイルを1つのパッケージにまとめられますが、ランゲージ別に分けたり、ストリームファイル、ルースメディア、SoundBankに分けたりできます。

しかし一旦ゲームをリリースすると、メインリリースで使われたコンテンツは、Default file assignmentではパッケージできなくなります。全てのコンテンツを、パッケージにマニュアル操作で挿入することになります：

- メインパッケージを、マニュアル作成する
- 全てのコンテンツを、メインパッケージにドロップする

基本的に、ここでマニュアル作成したパッケージに、メインリリースに同梱された全てのSoundBankやストリームファイルが含まれます。これを参照基準として、新規作成したコンテンツを識別します。次に、File Packagerプロジェクトファイルを保存します。



#### 注記

DLCコンテンツは、必ずメインリリースで使用したWwiseプロジェクトを使って作成してください。また、両方のリリースのSoundBankやパッケージフォーマットの互換性を確保するために、必ずWwiseの同じバージョンを使ってください。

### File Packagerの操作

- 1つのDLCに対して、1つのFile Packagerパッケージをマニュアル作成する。
- ユーザーは、File Packagerの中の新規コンテンツを見つける。
- 新規コンテンツを、マニュアル作成したパッケージ (DLC用) にドロップする。

これで、メインリリースに入っていたコンテンツを変更できます。既存 SoundBankや既存ストリームファイルを変更する場合は、新しいパッケージに入れる必要があります。ゲーム側では、メインリリースパッケージと新規 (DLC) パッケージを、古いものから順にロードします。同じオブジェクトの複数の異なるバージョンをロードすると、Wwiseは最新版を最もプライオリティが高いものとします。

### 例

- メインリリースに含まれたSoundBankAは、Package #1にパッケージされている。
- SoundBankAのコンテンツを、メインリリース後に変更する。改めて新規の Package #2にパッケージされ、DLCの一部となる。
- DLC付きのゲームは、まずPackage #1をロードして、次にPackage #2をロードする。
- ゲームがSoundBankAをロードする時に、WwiseはPackage #2の中のものをロードする (Package #1の中のものよりもプライオリティが高いため)。

### 関連項目

- [「DLCの注意点と制約」](#)

## ファイルパッケージを生成する

プロジェクトのパッケージ（単数または複数）を作成して中身を入れ終え、ファイルの順番も設定できたところで、パッケージを1つ、または全て、生成 (Generate) します。ファイルパッケージは、Generated SoundBanksのプラットフォームフォルダに、.pckファイルとして保存されます。ただし、必要であれば別の場所を指定できます。

生成中にエラーが発生した場合は、Generationログに表示されます。エラーが発生しなければ、これらのパッケージファイルをゲームディスクに追加できます。

### ファイルパッケージを生成するには:

1. ファイルパッケージを別の場所に保存するには、Output directoryフィールドの横の、Browseボタンをクリックする。

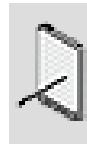
Browse for Folderダイアログボックスが開く。

2. ファイルパッケージを保存する場所まで移動して、フォルダを選択してから、OKをクリックする。

Output directoryフィールドに、フルパスが追加される。

3. メニューバーで、Generate > All packagesをクリックする。

Generating Packagesダイアログボックスが開き、生成プロセスのプログレスが表示される。エラーがあった場合は、ログで黄色の円でハイライト表示されます。



### 注記

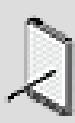
全てのパッケージを毎回、生成しない場合は、Packagesビューでパッケージを1つ以上選択してから、Generate > Selected Packagesをクリックします。

## 関連トピック

- 「プロジェクトにファイルパッケージを追加する」
- 「プロジェクトのファイルパッケージを削除する」
- 「ファイルパッケージの中身を入れる」
- 「パッケージ内のファイルを並べ替える」
- 「コマンドラインで、File Packager引数を使用」

## コマンドラインで、File Packager引数を使用

コマンドラインからパッケージファイルを生成する場合は、そのパッケージの詳細を定義するために使う引数が決まっています。コマンドラインで使用できる引数のリストを、下表に示します。

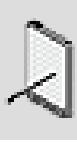
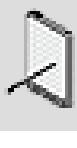


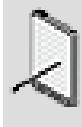
### 注記

File Packagerの、Ver. 2009.2や、それ以前のバージョンで使用された引数は、現バージョンと互換性があります。

引数 (Argument)	内容
-generate	<p>ファイルパッケージ生成プロセスを開始する。この引数は、ファイルパッケージャのプロジェクトパス (Project path) や、引数-infoと組み合わせて使える。具体例を以下に示す。</p> <ul style="list-style-type: none"> <li>• File Packager Projectを使って生成: -generate &lt;project_path&gt;</li> <li>• SoundbanksInfo.xmlファイルを使い、1つのパッケージを生成: -generate -info &lt;soundbank_info_filename&gt; -output &lt;package_name&gt;</li> <li>• SoundbanksInfo.xmlファイルを使い、ストリームのパッケージを1つ、バンクのパッケージを1つ生成する。 -generate -info &lt;soundbank_info_filename&gt; -output_stm &lt;package_name&gt; -output_bnk &lt;package_name&gt;</li> </ul>
-info <filename>	<p>SoundBanksInfo.xmlファイルの場所を指定する。</p> <p>以下を単数または複数使う。</p> <p>-output -put_</p>
-output <filename>	SoundBankのファイルとストリームメディアのファイルを含む、1つのファイルパッケージを生成する。<filename>で、ファイル名

引数 (Argument)	内容
	<p>(File name) と、生成されたファイルパッケージを保存する場所を指定する。</p> <p>&lt;projectpath&gt;を指定しない時に、この引数を使える。フルファイルパス、または単純にパッケージ名を使う。どちらの場合も、ファイルパッケージの拡張子を入れる必要がある。</p> <p>この引数を、-output_stm、または-output_bnkと組み合わせて、ファイルの中身が異なる複数のパッケージファイルを生成できる。</p> <p>この値はプロジェクト内で定義されるが、引数-outputが明確に定義された場合は、プロジェクト内で指定した値よりも、優先される。</p>
-output_stm <filename>	<p>ストリーム (Stream) 用メディアファイルだけを含むファイルパッケージを1つ生成する。&lt;filename&gt;で、ファイル名 (File name) と、生成されたファイルパッケージを保存する場所を指定する。</p> <p>&lt;projectpath&gt;を指定しない時に、この引数を使える。フルファイルパス、または単純にパッケージ名を使う。どちらの場合も、ファイルパッケージの拡張子を入れる必要がある。</p> <p>この引数を、-output_stm、または-output_bnkと組み合わせて、ファイルの中身が異なる複数のパッケージファイルを生成できる。</p>
-output_bnk <filename>	<p>SoundBank (SoundBank) ファイルだけを含むファイルパッケージを1つ生成する。&lt;filename&gt;で、ファイル名 (File name) と、生成されたファイルパッケージを保存する場所を指定する。</p> <p>&lt;projectpath&gt;を指定しない時に、この引数を使える。フルファイルパス、または単純にパッケージ名を使う。どちらの場合も、ファイルパッケージの拡張子を入れる必要がある。</p> <p>この引数を、-output_stm、または-output_bnkと組み合わせて、ファイルの中身が異なる複数のパッケージファイルを生成できる。</p>
-output_loose <directory> -output_loose	<p>ルースメディアファイルだけを含むファイルパッケージを、単数または複数生成する。ルースメディアファイルとは、1つ以上のSoundBankのイベントやファイル構造から参照され、このプロジェクトのどのSoundBankにも含まれないメディア。&lt;directory&gt;の指定は任意で、生成したファイルパッケージを保存する場所を指定する。&lt;directory&gt;を指定しない場合は、デフォルトのディレクトリである、SoundBanksInfo.xmlディレクトリが使われる。</p> <p>&lt;projectpath&gt;を指定しない時に、この引数を使える。相対ディレクトリまたは絶対ディレクトリを使う。</p> <p>この引数を、-output、-output_stm、または-output_bnkと組み合わせて、ファイルの中身が異なる複数のパッケージファイルを生成できる。</p>
-blocksize <number>	<p>パッケージ内のデータをアラインする、バイト数。指定するバイト数は、各プラットフォームのI/Oデバイスの要件によって決まる。</p> <p>各プラットフォームの一般的な数値は1です。</p> <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;">  <p><b>注記</b> デフォルト値の「1」は、アライメントがないことを意味する。</p> </div> <p>ロックサイズやプラットフォーム別のI/Oデバイスの制約については<a href="#">SDKドキュメンテーション</a>で、Streaming &gt; Low-Level I/Oを参照してください。</p>
-hideprogressui <boolean>	Wwise内からパッケージを生成している間、ダイアログGenerating Packagesを表示させるのかを、指定する。

引数 (Argument)	内容
	「true」に設定すると、プログレスダイアログを表示しない。デフォルトで、この引数は「false」に設定されている。
<projectpath>	1つの.wfpprojファイルを直接指す、File Packagerプロジェクトパスを1つ、指定する。引数generateと共に使うと、他の引数を無視して、File Packagerプロジェクトを使ってパッケージを生成する。 この引数は、オプショナル設定である。
-soundbanks_dir <directory>	生成したSoundBankを、Wwiseが保存したディレクトリを指定している、SoundBanksInfo.xmlファイル (SoundBanksInfo/RootPaths/SourceFilesRootにある) の情報を、オーバーライドする。 通常の書き方: 「projectpath\GeneratedSoundBanks\platform\」。 この引数は、オプショナル設定である。
-cache_dir <directory>	コンバージョンしたファイルを、Wwiseが保存したディレクトリを指定している、SoundBanksInfo.xmlファイル (SoundBanksInfo/RootPaths/SourceFilesRootにある) の情報をオーバーライドする。 通常の書き方: 「projectpath\.cache\platform\」。 この引数は、オプショナル設定である。
-languages "<language1><languageN>"	パッケージするランゲージのリストを指定する。 <div style="text-align: center;">  <b>注記</b>            リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。            SFXもパッケージするには、ランゲージリストでキーワードとしてSFXを使う。         </div> <p>この引数は、オプショナル設定である。デフォルトとして、全てのランゲージがパッケージされる。</p> <p>この引数は、File Packagerプロジェクトが指定されていない時に限り、使用する。</p>
-banks "<soundbank1><soundbankN>"	パッケージするSoundBankのリストを指定する。名前に、bnk拡張子は含まれない。 <div style="text-align: center;">  <b>注記</b>            リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。         </div> <p>この引数は、オプショナル設定である。デフォルトとして、全てのSoundBankがパッケージされる。</p> <p>この引数は、File Packagerプロジェクトが指定されている時も、使用できる。指定したSoundBankへのレファレンスがあるパッケージを、自動的にビルドする。</p>
-includedFilesInBanks "<soundbank1><soundbankN>"	中に含まれるメディアファイルをルースファイルとしてパッケージに追加するSoundBankのリストを指定する。この設定は、ルースメディアがPrepareEvent APIに必要であり、かつメモリにロードしたバンクにも必要なメディアである場合に限り使用する。名前に、bnk拡張子は含まれない。 <div style="text-align: center;">  <b>注記</b>            リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。         </div>

引数 (Argument)	内容
	この引数は、オプショナル設定である。デフォルトとして、SoundBankに含まれるメディアファイルは、ルースファイルとしてパッケージに追加しない。
-excludedFilesInBanks “<soundbank1> <soundbankN>”	<p>参照しているメディアファイルがバンクに含まれず、ルースファイルとしてパッケージに追加すべきSoundBankのリストを指定する。この設定は、ルースメディアがPrepareEvent APIに必要であり、かつメモリにロードした別のSoundBankにも必要なメディアである場合に使用できる。名前に、bnk拡張子は含まれない。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <b>注記</b>            リストは、スペース区切りである。リストを1つの引数に含めるには、ダブルクオテーションマークを使うこと。         </div> <p>この引数は、オプショナル設定である。デフォルトとして、SoundBankに含まれないがこのSoundBankが参照するメディアファイルは、他のSoundBankから参照されていない場合に限り、ファイルパッケージに追加する。</p>

## ファイルパッケージに関する、コツとベストプラクティス

ファイルパッケージの個数やコンテンツを決める前に、以下のセクションに目を通して、ファイルパッケージを最大限に活用するための、コツやベストプラクティスを参照してください。

### リリース後のダウンロード用コンテンツ (DLC) を、パッケージする

ゲームのリリース後も、プレイヤーにDLC（ダウンローダブルコンテンツ）を提供することで、ゲームの寿命を延長することが、一般的になっています。このような新規コンテンツの配信を管理するための簡単な手順を、以下に示します。

#### ファイナルリリース向けコンテンツをパッケージするには:

1. Wwiseプロジェクトで、ゲームの最終ビルド用のSoundBankを、生成する。
2. File Packagerを開く。
3. Wwiseが生成した、SoundBanksInfo.xmlファイルを、インポートする。
4. デフォルトのファイルアサインメント (File assignments) 設定を、Noneに設定する。
5. 全てのファイルを、マニュアル操作で1つのパッケージに追加する。

これで、あとから新規コンテンツを追加した場合に、それを見つけやすくなる。

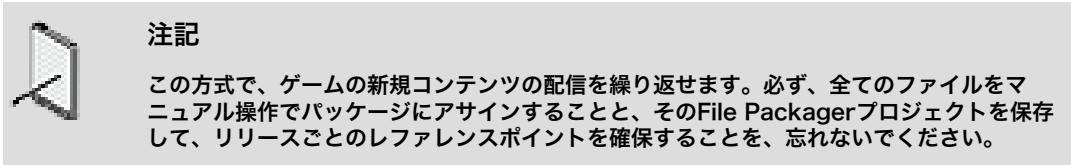
6. パッケージ（単数または複数）を生成して、ゲームディスクに追加する。
7. このFile Packagerプロジェクトを保存して、ファイナルリリースのレファレンスポイントとする。

#### リリース後にDLCをパッケージするには:

1. Wwiseで新規コンテンツを作成してから、DLC用のSoundBankを生成する。
2. ファイナルリリースの、レファレンスポイントとしたFile Packagerプロジェクトを、再度開く。

デフォルトとして、File Packagerは、ファイルSoundBanksInfo.xmlをスキヤンして、Files to packageビューに中身を入れます。追加した新規コンテンツは、どのパッケージにもアサインされません。

3. DLC用のパッケージを1つ以上、作成して、新しく作成したSoundBankやストリーム用メディアファイルを、これらのパッケージにマニュアル操作で追加する。
4. 新規パッケージ（単数または複数）を生成して、ユーザーベースに配信する。
5. このFile Packagerプロジェクトを再び保存して、今回のDLCリリースのレファレンスポイントとする。



audio**kinetic**

## パート VII. Wwiseを使いこなす



---

37.	Project Explorer画面について	740
	概要	741
	Project Explorerの記号や色文字	743
	Project Explorerで作業する	744
38.	Event Viewer画面について	747
	概要	748
	Event Viewerで作業する	748
39.	Property Editor画面を使う	753
	概要	754
	Property Editorを使って作業する	770
40.	Contents Editor画面について	774
	概要	775
	Contents Editorで作業する	782
41.	Transport Control画面について	791
	概要	792
	再生方法の設定	794
	Transport Controlでオブジェクトをピン付けする	797
	コンテンツの再生、一時停止、停止	798
	再生中にゲームシンクを使う	798
42.	Schematic View画面について	803
	概要	804
	Schematic Viewのカスタマイズ	804
	Schematic Viewを活用する	806
43.	グラフビューについて	809
	概要	810
	グラフビュー表示設定を変更する	811
	グラフビューのコントロールポイントの使い方	816
	グラフビューのカーブの使い方	818
44.	タイムラインについて	822
	概要	823
	ポジショニングにおけるタイムラインの使い方	826
	Music Segment Editorのタイムラインを使う	827
45.	検索、クエリ、レファレンスの活用	829
	概要	830
	プロジェクトエレメントを検索する	830
	あるオブジェクトをレファレンスする、プロジェクトエレメントを検索する	832
	クエリの使い方	837
	クエリに関する、コツとペストプラクティス	845
46.	プリセットを使用する	846
	概要	847
	プリセットを使用する	847
47.	コントロールサーフェスの使用	851
	概要	852
	コントロールサーフェスデバイスをWwiseに接続する	852
	Control Surface Sessionを作成する	853
	コントロールサーフェスのバインディングを理解する	853
	コントロールサーフェスのバインディングを作成する	854
	コントロールサーフェスのView Groupsを理解する	857

---

Control Surface Sessionのコンフリクト対応 .....	858
コントロールサーフェスのツールバーの使用 .....	859

---

## 第37章 Project Explorer画面について

概要 .....	741
Project Explorerの記号や色文字 .....	743
Project Explorerで作業する .....	744

## 概要

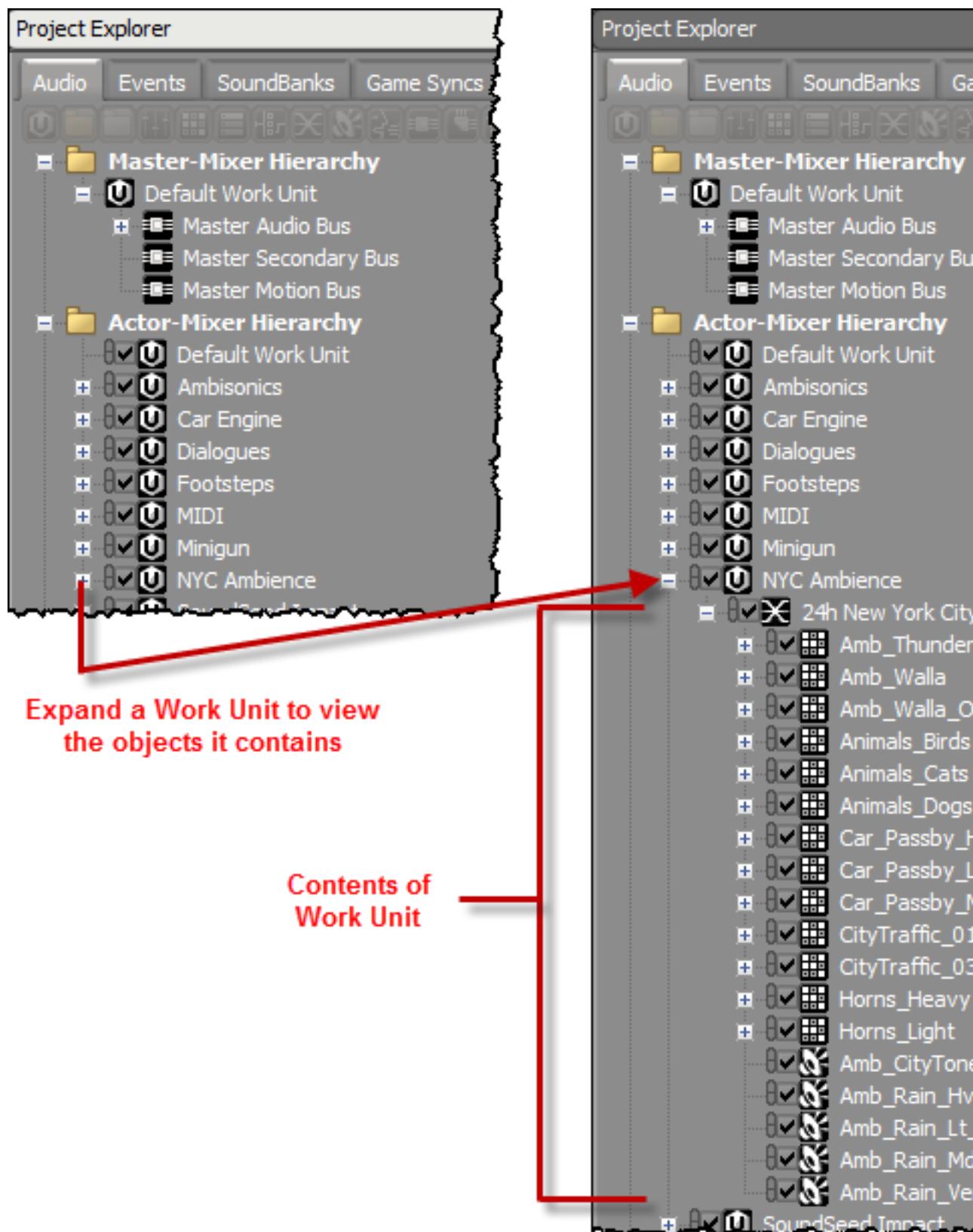
Project Explorerは、Wwiseプロジェクトの全てのエレメント（エレメント）を管理するための画面です。Project Explorerには様々なタブがあり、それぞれを開いて、そのタブの階層構造に表示される個々のエレメントを作成、名前変更、カット、コピー、ペースト、削除できます。また各タブのツールバーから簡単に、プロジェクト階層に親オブジェクトや子オブジェクトを追加できます。Wwiseプロジェクトの様々なエレメント、例えばオーディオアセット、ミュージックアセット、モーションアセット、バス、イベント、SoundBank、ゲームシンクなどを、管理して整理する機能が、この1つの場所に集約されています。

また、複数のチームメンバーが同じプロジェクトの異なる部分を同時に作業できるように、各種プロジェクトエレメントを複数のWork Unitに分割できるのも、Project Explorerからです。Work Unitの作成については「[プロジェクトをWork Unitで分割](#)」を参照してください。

Project Explorerには、以下のタブがあります。

- **Audio** - プロジェクトのサウンド、ミュージック、モーション、バスの構造を表示。
- **Events** - プロジェクトの全てのイベント、つまりアクションとダイアログの両方を表示。
- **SoundBanks** - プロジェクトの全てのSoundBankを表示。
- **Game Syncs** - プロジェクトの全てのState、Switch、Trigger、Game Parameter、引数を表示。
- **ShareSets** - プロジェクトの全てのEffect ShareSetやAttenuation ShareSetを表示。
- **Sessions** - プロジェクトの全てのSoundcasterセッションを表示。
- **Queries** - プロジェクトの全てのクエリを表示。

Project Explorerに表示される様々なレベルを検索する時に、グループを展開または縮小するには、オブジェクトの横のプラス (+) やマイナス (-) 記号をクリックします。



## Project Explorerの記号や色文字

Wwiseでは、プロジェクトのオブジェクトやエレメントを1つずつ、アイコンで表すので、Project Explorerなどインターフェースの各所で、オブジェクトやエレメントのタイプを簡単に識別できるようになっています。全てのアイコン名と内容のリストを見るには「[Wwiseのアイコン](#)」を参照してください。

この他にも、色分けなどをを利用して、Project Explorerの一部のオブジェクトのステータスが分かるようになっています。例えば、オブジェクトの色によって、そのオブジェクトに関連するソースがあるのか、また、特定プラットフォーム用にコンバージョンされているのか、などが分かります。Project Explorerで採用している記号や色分けを、下表に示します。

記号、色	例	Usage	タブ
アステリック		このWork Unit内のプロジェクトエレメントに、変更が加えられたことを示す。プロジェクトを保存すると、消える。	全て
Link icon		<p>グレイのリンクアイコンは、該当するオブジェクトの設定がプラットフォームにおいて可能であることを示しています。Wwiseでは、1つのオブジェクトに関連する様々なプロパティ値の大部分を、リンク・アンリンクできるので、同時に複数のプラットフォーム向けにオーサリングできます。デフォルトでは、複数のアクティブなプラットフォームにおいて、すべてのプロパティ値はリンクされていますので、同じ値に設定できます。あるプロパティ値をアンリンクすると、そのプラットフォームでカスタマイズできるようになります。</p> <p>リンク・アンリンクについては「<a href="#">プラットフォームごとに、オブジェクトプロパティをカスタマイズする</a>」を参照してください。</p>	Audio and ShareSets ( Effects and Audio Devices用)
チェックマーク		チェックマークが付いたオブジェクトは、現在のプラットフォームに含まれていることを示す。チェックマークがないオブジェクトは、現在のプラットフォームに含まれていない。	Audio and ShareSets ( Effects and Audio Devices用)
赤字のオブジェクト名		Sound SFX、Music Track、またはミュージックトラックが作成されたが、ソースに関連付いていないことを示す。	Audio

記号、色	例	Usage	タブ
青字のオブジェクト名	F5_Concrete_01	このSound SFX、またはMusic Trackが、現在のプラットフォーム用にコンバージョンされていないソースに関連付いていることを示す。	Audio
白字のオブジェクト名	F5_Concrete_01	音またはミュージックトラックが、現在のプラットフォーム用にコンバージョンされたソースに関連付けていることを示す。	Audio

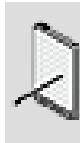
Workgroupプラグインを使用している場合は、プロジェクトファイルのステータスが分かるように、Project ExplorerのWork Unitに、特別なオーバーレイアイコンが表示されます。例えば、Marked for add（マーク済み）、Checked out（チェックアウト中）、未チェックアウトなど、ファイルのステータスが、アイコンで表示されます。オーバーレイアイコンについては「[Workgroupプラグインを使ったプロジェクトファイルの管理](#)」を参照してください。

## Project Explorerで作業する

Project Explorerで、標準的なWindows Explorer（またはMac Finder）コマンド、例えば名前変更、カット、コピー、ペーストなどを、ショートカットメニューから実行できます。さらにタブ内で、またはWwiseインターフェースの他のビューに、プロジェクトエレメントをドラッグ＆ドロップできます。なお、オブジェクトを移動すると、親子関係に影響するので、注意してください。これらのエレメントや、エレメント同士の関係の構築と管理については、以下のセクションを参照してください。

- ・「Actor-Mixer Hierarchyの構築」
- ・23章Interactive Music Hierarchyの構築
- ・15章イベントの管理
- ・16章ダイナミックダイアログの管理
- ・35章SoundBankの管理
- ・「Switchの使い方」
- ・17章Stateの活用
- ・19章RPCの使い方
- ・20章Triggerの使い方
- ・21章ダイナミックダイアログの、StateやState Groupの活用
- ・「エフェクトの使用」
- ・「減衰インスタンスの管理」
- ・「シミュレーションを構築する」
- ・「クエリの作成」

Project Explorerのショートカットメニューは標準的なコマンド以外に各タブ専用のコマンドもあり、例えばプラットフォームを含めたり除外したり、Audioタブでオーディオファイルをインポートしてコンバージョンしたり、SoundBanksタブでSoundBankの定義ファイルをインポートしたりするコマンドがあります。



### 注記

1つのレイアウトで、同時に複数のProject Explorerインスタンスを開くことができます。

## 表示オプションの設定

プロジェクトの複雑な階層を管理しやすくして、プロジェクトエレメントをすぐに見つけられるように、Project Explorerの表示設定を変更できます。例えば、Actor-Mixer Hierarchyの中身を検索しやすくするために、全てのグループを自動的に縮小表示させることができます。

### Project Explorerの表示設定を変えるには:

1. Project Explorerの、どのタブからでも、オブジェクトを1つ右クリックする。

ショートカットメニューが表示される。

2. ショートカットメニューで、**Expand Options**を選択する。

サブメニューExpand Optionsが表示される。

3. 以下のオプションを1つ選択する。

- **Auto Expand Child Object** (子オブジェクトの自動展開) : 親オブジェクトを展開または縮小した時に、全ての子オブジェクトを展開させる。このオプションは、デフォルトで無効になっている。
- **Expand All** (全て展開) : 階層の全てのオブジェクトを展開させる。
- **Collapse All** (全て縮小) : 階層の全てのオブジェクトを縮小させる。

Project Explorerの様々なレベルの間を素早く移動するには、以下のキーボードショートカットを使用できる。

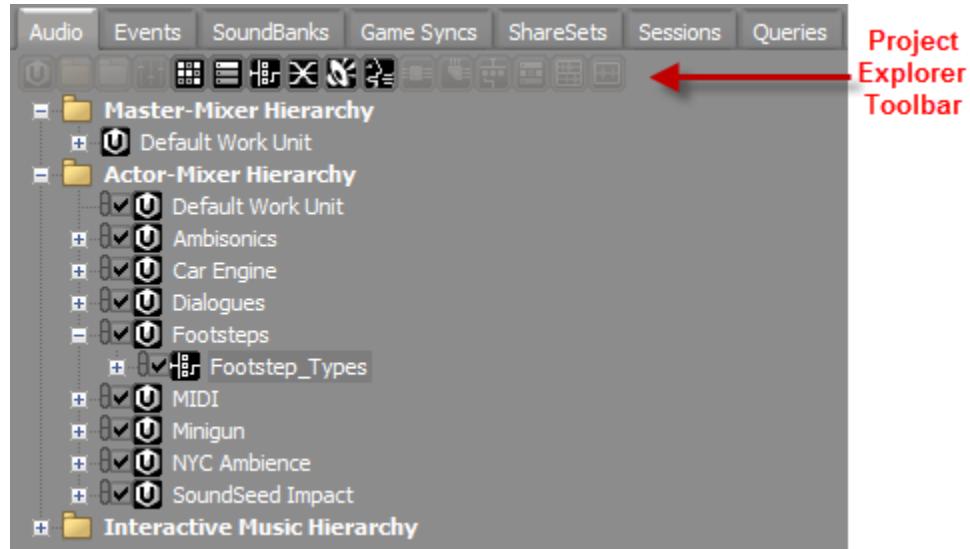
Keyboard Shortcut	To
上向き矢印	オブジェクトリストで、上に移動する。
下向き矢印	オブジェクトリストで、下に移動する。
右向き矢印	オブジェクトを展開する。
左向き矢印	オブジェクトを縮小する。

## 関連トピック

- 「Project Explorerツールバーを使う」
- 「Project Explorerの記号や色文字」

## Project Explorerツールバーを使う

Project Explorerのどのタブでも、ツールバーにアイコンが表示されます。これらのアイコンは、階層で選択したノードに親や子として追加できる、プロジェクトエレメントを表します。階層で選択したオブジェクトの種類によって、選択できるアイコンが異なります。



デフォルトとして、ツールバーに表示されるのは、選択したノードの子として追加できる全てのプロジェクトエレメントです。親として可能なプロジェクトエレメントを表示させるには、Shiftキーを押します。

#### ツールバーを使い、階層に子オブジェクトを追加するには:

1. Project Explorerで、階層のオブジェクトまたは他のプロジェクトエレメントを1つ選択する。

選択したエレメントの子として追加できるプロジェクトエレメントのアイコンが、ツールバーで選択可能となる。

2. 選択可能なアイコンを1つクリックして、選択したエレメントの新しい子を作成する。
3. 子オブジェクトに適切な名前をつけて、Enterを押す。

#### ツールバーを使い、階層に親オブジェクトを追加するには:

1. Project Explorerで、階層のオブジェクトまたは他のプロジェクトエレメントを1つ選択する。
2. Shiftを押して、選択したエレメントの親として追加できるプロジェクトエレメントを表示させる。
3. 選択可能なアイコンを1つクリックして、選択したエレメントの新しい親を作成する。
4. 親オブジェクトに適切な名前をつけて、Enterを押す。

#### 関連トピック

- 「表示オプションの設定」
- 「Project Explorerの記号や色文字」

---

## 第38章 Event Viewer画面について

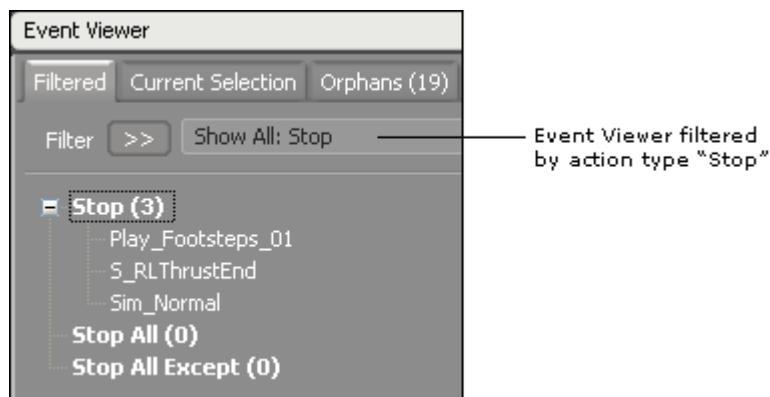
概要 .....	748
Event Viewerで作業する .....	748

## 概要

Wwiseでは、プロジェクト階層の様々なサウンド構造、ミュージック構造、モーション構造に、具体的なアクションを適用する時は、アクションイベントと呼ばれるものを使います。一般的なゲームで、アクションイベント数が数百個にもなることがあるので、目的のイベントを素早く見つけられることは重要です。Event Viewerの、ソートツールやフィルタツールを利用して、現在のプロジェクト用に作成した様々なイベントを検索できます。

Event Viewerには、以下の3つのタブがあり、それぞれ異なるフィルタを使ってイベントが表示されます。

- **Filtered (フィルタ)** - 全てのイベントが、アルファベット順に表示される。Show All - Sorted (ソートして全て表示) オプションを使って、リストをアクションタイプ別にソートしたり、特定のアクションタイプを含むイベントだけが表示されるようにフィルタを設定したりできる。フィルタ後に表示されたイベントを、フォルダ横のプラス (+) やマイナス (-) 記号をクリックして展開または縮小をして、検索できる。



- **Current Selection (現在の選択)** - Project Explorerの、Audioタブで選択したオブジェクト（単数または複数）に関連するイベントのリストが表示される。
- **Orphans (オーファン)** - オーファン（孤立）イベント、つまり作成されたものの、現在は特定オブジェクトに関連していないイベントが、表示される。

また、イベントの削除、Event Editorの表示、およびEvent Viewerのイベント（単数または複数）を、Soundcasterや、SoundBank Viewerなど、Wwiseの他のビューにドラッグ&ドロップすることもできます。



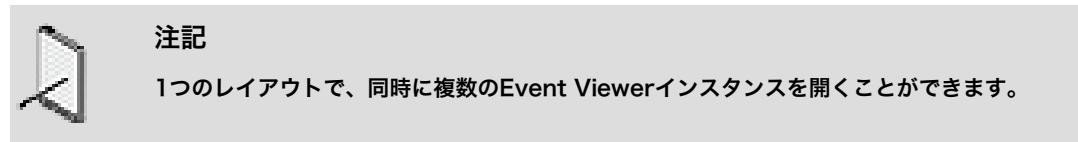
## Event Viewerで作業する

Event Viewerに、プロジェクトで作成した全てのイベントが表示されます。1つのプロジェクトに大量のイベントが入っていることもあるので、イベントの編集や再

生などが必要な時に、そのイベントを簡単に検索できるようなソート機能やフィルタ機能が必要となります。

Event Viewerの、様々なツールや表示設定の内容を、以下のセクションで説明します。

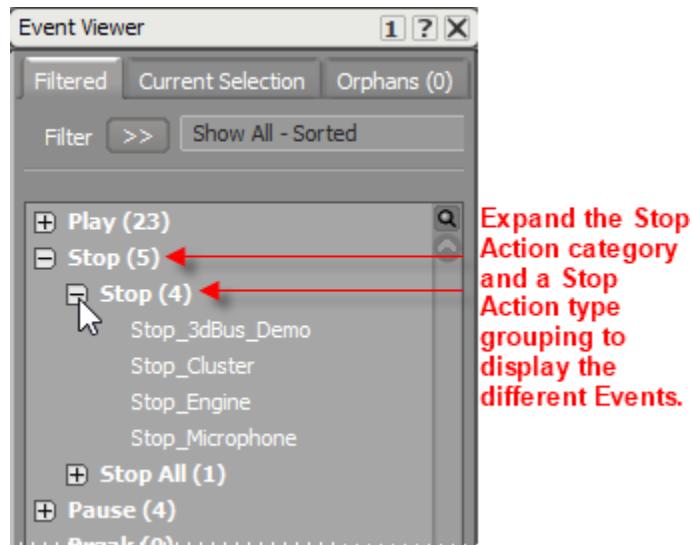
- ・「Event Viewerで情報を探す」
- ・「イベントリストのソート」
- ・「イベントリストのフィルタ設定」



### Event Viewerで情報を探す

Event Viewerの情報は、イベントリストに設定したソートやフィルタの条件によって、表示方法が変わります。フィルタやソートの設定で、Event Viewerの中に複数のレベルができる場合もあります。様々なレベルの情報を検索する時に、アクションカテゴリーやタイプの横にある、プラス (+) やマイナス (-) 記号をクリックすれば、そのグループを展開・縮小できます。

様々なレベルの情報を検索する時に、アクションカテゴリーやタイプの横にある、プラス (+) やマイナス (-) 記号をクリックすれば、そのグループを展開・縮小できます。



### Quick Navigation

Event Viewerの様々なレベルを素早く移動 (Navigate) するには、以下のキーボードショートカットを使用できます。

Keyboard Shortcut	To
上向き矢印	イベントリストで、上に移動する。

Keyboard Shortcut	To
下向き矢印	イベントリストで、下に移動する。
右向き矢印	Event Actionの、カテゴリまたはタイプのグループを展開する。
左向き矢印	Event Actionの、カテゴリまたはタイプのグループを縮小する。

## Event Viewerの表示設定の変更

状況や、個人的な好みに応じて、Event Viewerの表示設定を変更できます。例えば、全てのグループを展開・縮小させて表示させたり、リストにフィルタやソートをかけた時に毎回、全てのグループを自動的に展開させたりできます。

### Event Viewerの表示設定を変更するには:

1. Event Viewerで、グループ名を1つ右クリックする。

ショートカットメニューが表示される。

2. 以下のオプションを1つ選択する。

- **Auto Expand (自動展開)** : リストをフィルタまたはソートした時に毎回、全グループを自動的に展開させる。
- **Expand All (全て展開)** : 全てのカテゴリーやタイプのグループを、展開させる。
- **Collapse All (全て縮小)** : 全てのカテゴリーやタイプのグループを、縮小させる。

### 関連トピック

- 「[イベントリストのソート](#)」
- 「[イベントリストのフィルタ設定](#)」

### イベントリストのソート

デフォルトとして、Event Viewerで開くのは、Filteredタブで、プロジェクトの全てのイベントがアルファベット順に表示されます。特定イベントを素早く検索するには、Show All - Sorted (ソートして全て表示) フィルタを選択すれば、イベントがEvent Action別にソートされます。つまり、全てのPlayアクションが1つのグループにまとめられ、全てのStopアクションが、別のグループにまとめられて、表示されます。

### イベントリストをソートするには:

1. Filteredタブをクリックする。

デフォルトで、全てのイベントがアルファベット順に並べられる。

2. Filterボタンをクリックして、フィルタやソートのオプションを表示させる。
3. Show All - Sorted (ソートして全て表示) オプションをクリックする。

プロジェクトの全てのイベントが、アクションカテゴリ別にソートされ、さらにアクションタイプ別にソートされます。

## 関連トピック

- ・「イベントリストのフィルタ設定」
- ・「Event Viewerで情報を探す」
- ・「Event Viewerの表示設定の変更」

### イベントリストのフィルタ設定

イベントの検索をさらに洗練させるには、イベントリストにフィルタを設定します。イベントリストに設定できるフィルタを、以下に示します。

- ・「アクションタイプを条件に、リストをフィルタする」
- ・「現在の選択を条件に、リストをフィルタする」
- ・「オーファンイベントを条件に、リストをフィルタする」

#### アクションタイプを条件に、リストをフィルタする

特定のActionタイプだけを表示させるフィルタを、イベントリストに設定できます。例えば、Stop All（全てストップ）アクションだけ、またはMute（ミュート）アクションだけを表示させるフィルタなどを設定できます。

#### アクションタイプを条件に、イベントリストをフィルタするには

1. Filteredタブをクリックする。

デフォルトで、全てのイベントがアルファベット順に並べられる。

2. Filterボタンをクリックして、フィルタやソートのオプションを表示させる。
3. アクションカテゴリーを1つ選択して、リストでアクションタイプを1つクリックする。

選択したフィルタオプションに従って、イベントがフィルタされる。

## 関連トピック

- ・「現在の選択を条件に、リストをフィルタする」
- ・「オーファンイベントを条件に、リストをフィルタする」
- ・「Event Viewerで情報を探す」
- ・「イベントリストのソート」
- ・「Event Viewerの表示設定の変更」

### 現在の選択を条件に、リストをフィルタする

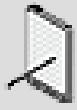
リストにフィルターを設定して、Project Explorerの、Audioタブで選択したオブジェクト（単数または複数）に関連する具体的なイベントを見つけることができます。

#### 現在の選択を条件に、イベントリストをフィルタするには:

1. Project Explorerの、Audioタブで、オブジェクトを単数または複数を選択する。

2. Event Viewerで、Current Selection（現在の選択）タブをクリックする。

選択したオブジェクトに関連するイベントが、イベントリストに表示される。



#### 注記

Project Explorerの、Audioタブで、複数のオブジェクトが選択された場合は、イベントがオブジェクト別にソートされます。

### 関連トピック

- ・「アクションタイプを条件に、リストをフィルタする」
- ・「オーファンイベントを条件に、リストをフィルタする」
- ・「Event Viewerで情報を探す」
- ・「イベントリストのソート」
- ・「Event Viewerの表示設定の変更」

#### オーファンイベントを条件に、リストをフィルタする

オーファン（孤立）イベント、つまりどのオブジェクトにも関連付いていないアクションを含むイベントを見つけるために、リストをフィルタできます。プロジェクト全体のオーファンイベントの合計数が、Orphansタブのタイトルに、角括弧表示されます。

#### オーファンイベントを条件に、リストをフィルタするには：

1. Event Viewerで、Orphans タブをクリックする。

全てのオーファンイベントが、イベントリストに表示されます。

### 関連トピック

- ・「アクションタイプを条件に、リストをフィルタする」
- ・「現在の選択を条件に、リストをフィルタする」
- ・「Event Viewerで情報を探す」
- ・「イベントリストのソート」
- ・「Event Viewerの表示設定の変更」

---

## 第39章 Property Editor画面を使う

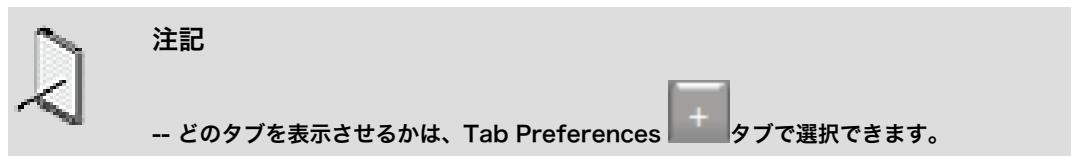
概要 .....	754
Property Editorを使って作業する .....	770

## 概要

Property Editorは、様々なプロパティ設定や動作設定を表示して、プロジェクトにある各種オブジェクトやGame Sync（ゲームシンク）の全体的な特徴を設定できる画面です。

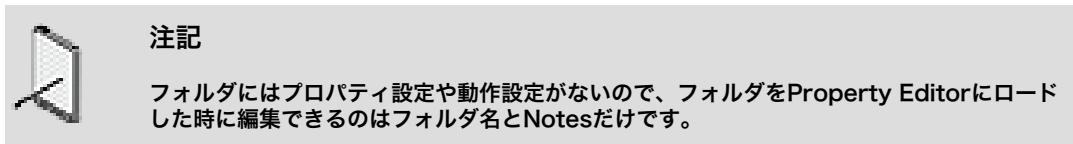
Property Editorにオーディオオブジェクトをロードすると、オブジェクトタイプに応じて以下のタブの多くが表示されます。

- ・「General Settingsタブ」
- ・「Source Settings」
- ・「エフェクト」
- ・「Positioning」（Actor-Mixer Hierarchy、Interactive Music Hierarchy、ほとんどのMaster-Mixer Hierarchy オブジェクト用）
- ・「Real-Time Parameter Controls」（RTPC / リアルタイムパラメータコントロール）
- ・「States」
- ・「Auto-ducking」（Audio Bus専用）
- ・「HDR」（Audio Bus専用）
- ・「Transitions」（ミュージックオブジェクト専用）
- ・「Stingers」（ミュージックオブジェクト専用）
- ・「Mixer プラグイン」
- ・「MIDI」（Actor-Mixer HierarchyやInteractive Music Hierarchyのオブジェクト）
- ・「Advanced Settings」（Auxiliary Bus以外の全てのオブジェクト）
- ・「All Properties」



Property Editorにゲームシンクをロードすると、編集するゲームシンクの種類によって、表示されるプロパティ設定が変わります。表示されるプロパティを、ゲームシンクの種類別に、下表に示します。

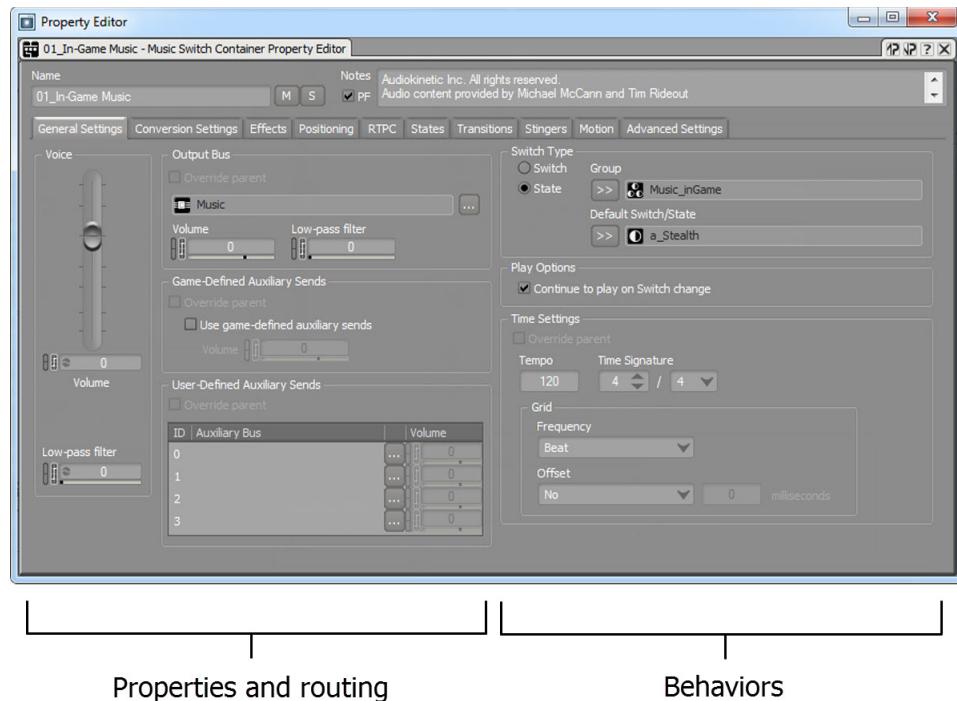
Game Sync	Property Editorの内容
Switch Group	Name、Note、Switch to Game Parameter値のマッピング。
Switch	Name、Notes。
State Group	Name、Notes、State間のトランジション設定。
State	Name、Notes、State設定値のコピー。
Game Parameter	Name、Notes、バインドするBuilt-in Parameter、Game ParameterのMinimum、Maximum、Default値、Interpolationモード値。
Trigger	Name、Notes。



## General Settingsタブ

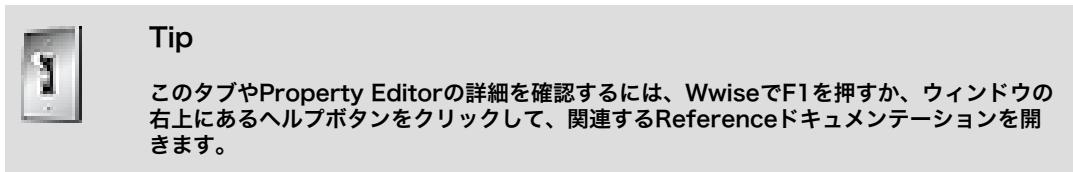
General Settingsタブは、Property Editorにオブジェクトをロードした時のメインのタブです。プロジェクト階層の様々なオブジェクトの一般的な特徴を確認したり設定したりできます。例えばボリュームやオブジェクトのピッチなど、具体的な数値をここで設定できます。また、階層内の様々なオブジェクトの具体的な再生設定も定義できます。

General Settingsタブの画面はプロパティ設定と動作設定に分けられています。プロパティ設定は左側に、動作設定は右側にあります。プロパティ設定は、さらに絶対プロパティと相対プロパティに分けられています。絶対プロパティは左側に、相対プロパティ、つまり累積されるプロパティは右側に表示されます。



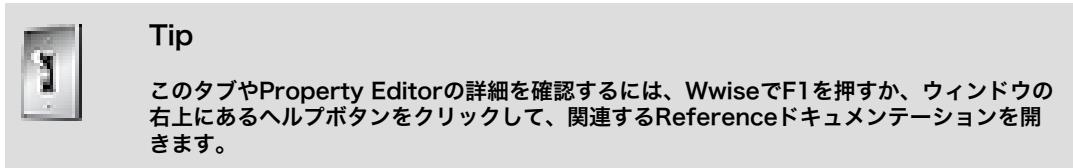
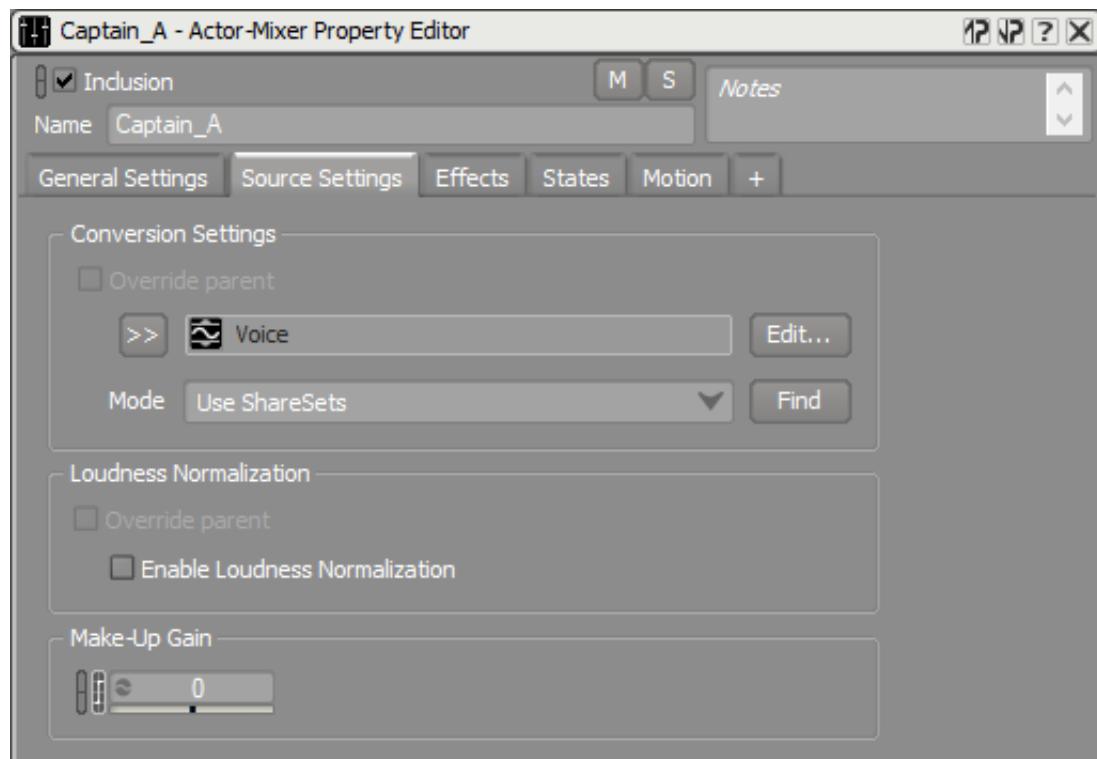
相対プロパティと絶対プロパティの詳細は「[プロジェクト階層内のプロパティについて](#)」を参照してください。

また、General Settingsタブはコンテキスト（内容）に対応するタブなので、表示されるプロパティ設定や動作設定の項目は編集するオブジェクトの種類によって変わります。



## Source Settings

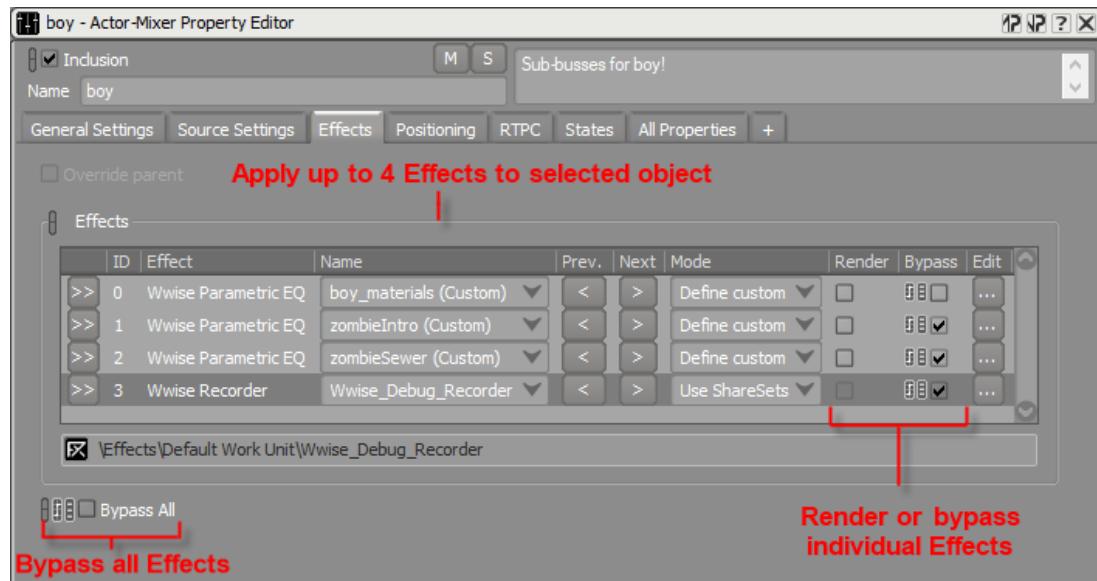
Source Settingsタブで、オブジェクトのコンバージョン設定ShareSetの選択ができます。ほかにも、ラウドネスノーマライゼーションを有効にしたり、ボリュームを調整するためのメイクアップゲインを設定したりできます。



## エフェクト

Effects（エフェクト）タブで、プロジェクト階層の様々なオブジェクトやバスに対して、1つ以上のエフェクトを適用できます。エフェクトを適用する時に、ShareSet（ShareSet）を適用するのか、エフェクトのカスタムインスタンスを適用するのかを決めます。また、エフェクトのバイパス（Bypass）、レンダリング（Render）、編集（Edit）も可能です。

## Property Editor画面を使う

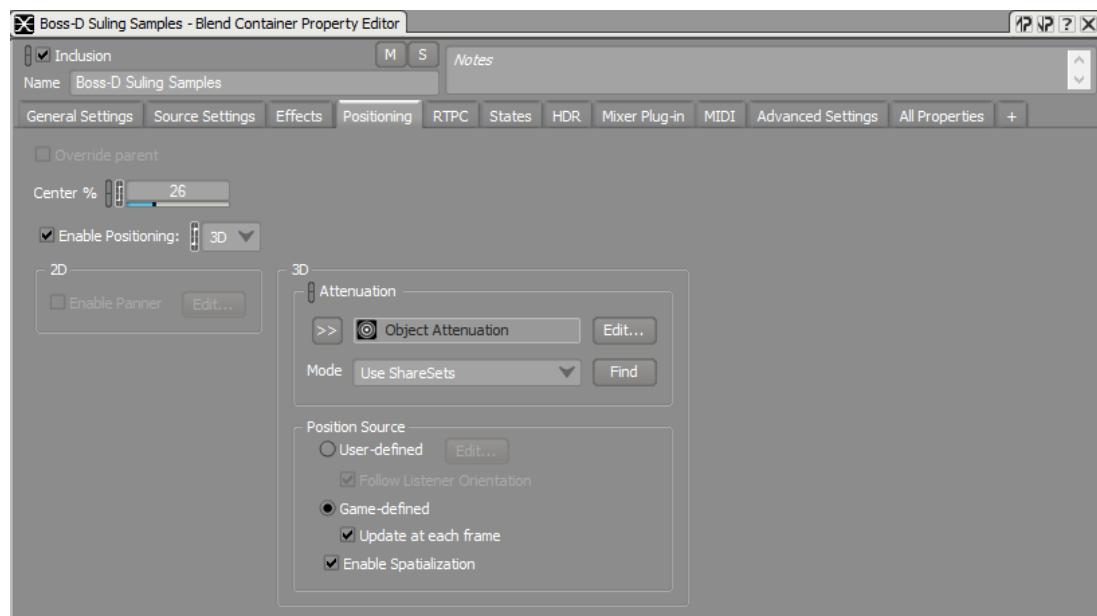


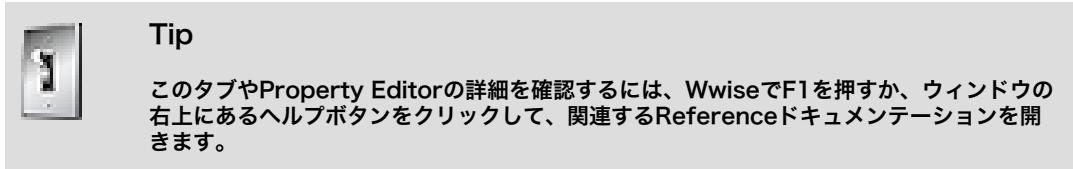
### Tip

このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

## Positioning

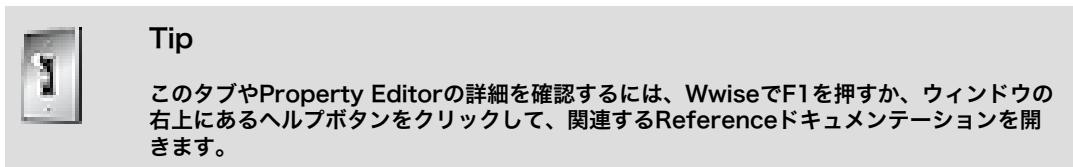
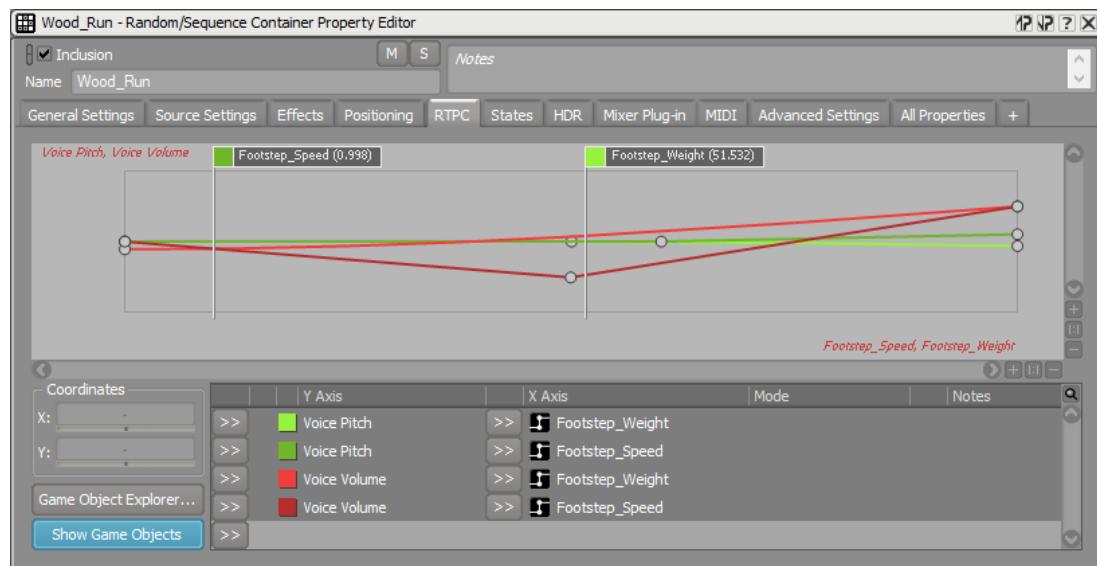
Positioningタブには、プロジェクトオブジェクトのポジショニング特性を確認して設定するための様々な項目が表示されます。まず、対象オブジェクトが2Dポジショニングと3Dポジショニングのどちらを使うのかを決め、選択したポジショニングタイプの実装方法を設定します。オブジェクトのオーディオスループットについて、センタースピーカーから再生する割合(%)も指定できます。





## Real-Time Parameter Controls

RTPC (Real-time Parameter Controls) タブには、ゲーム中に発生するリアルタイムパラメータ値の変化に基づいて、特定のサウンドプロパティをリアルタイムに編集できる設定機能があります。パラメータ値を表示するグラフの、Y軸にWwise側のプロパティ値が、X軸にゲーム側のパラメータ値が表示されます。グラフのカーブを1つだけ編集することも、同時に全てのカーブを編集することもできます。



## States

Statesタブで、Stateが入っているState Groupを現在のオブジェクトに追加できます。その後に[追加型プロパティ](#)であるピッチやボリュームをStateごとにカスタマイズして、そのオブジェクトの特徴を更に細かく定義できます。プロパティは、選択中のオブジェクトの種類によってある程度変わりますが、Propertiesボタンをクリックして追加したり削除したりできます。

## Property Editor画面を使う

The screenshot shows the 'Blend Container Property Editor' window for '24h New York City Ambience'. The 'Properties...' button is highlighted with a red box. Below it, the 'State' tab is selected, showing a table titled 'Customize additive properties per State' for the 'Game\_States' category. The table has columns for State name, Voice Volume, Voice Pitch, Voice Low-pass Filter, Voice High-pass Filter, Make-up Gain, Transposition, and Velocity Offset. The rows list various states like Defeat, Falling, Neutral, Rising, Victory, Hostile, Burn, Gang, Alley, Hangar, Street, Objective, DefuseBomb, NeutralizeHostile, and RescueHostage, each with its corresponding values.

State	Voice Volume	Voice Pitch	Voice Low-pass Filter	Voice High-pass Filter	Make-up Gain	Transposition	Velocity Offset
Defeat	4	-660	0	0	0	0	0
Falling	6	1160	0	0	0	-1	0
Neutral	0	0	0	0	0	0	0
Rising	2	0	0	0	5	1	0
Victory	5	1210	0	100	10	10	0
Hostile	0	-50	0	0	0	-5	0
Burn	0	0	0	0	3	0	0
Gang	0	0	21	0	0	0	0
Alley	0	0	0	60	5	0	0
Hangar	0	0	74	0	0	0	8
Street	0	0	0	0	25	0	25
Objective	10	100	0	0	16	0	0
DefuseBomb	0	15	0	0	5	0	0
NeutralizeHostile	33	200	0	0	0	0	0
RescueHostage	0	0	0	0	0	0	0

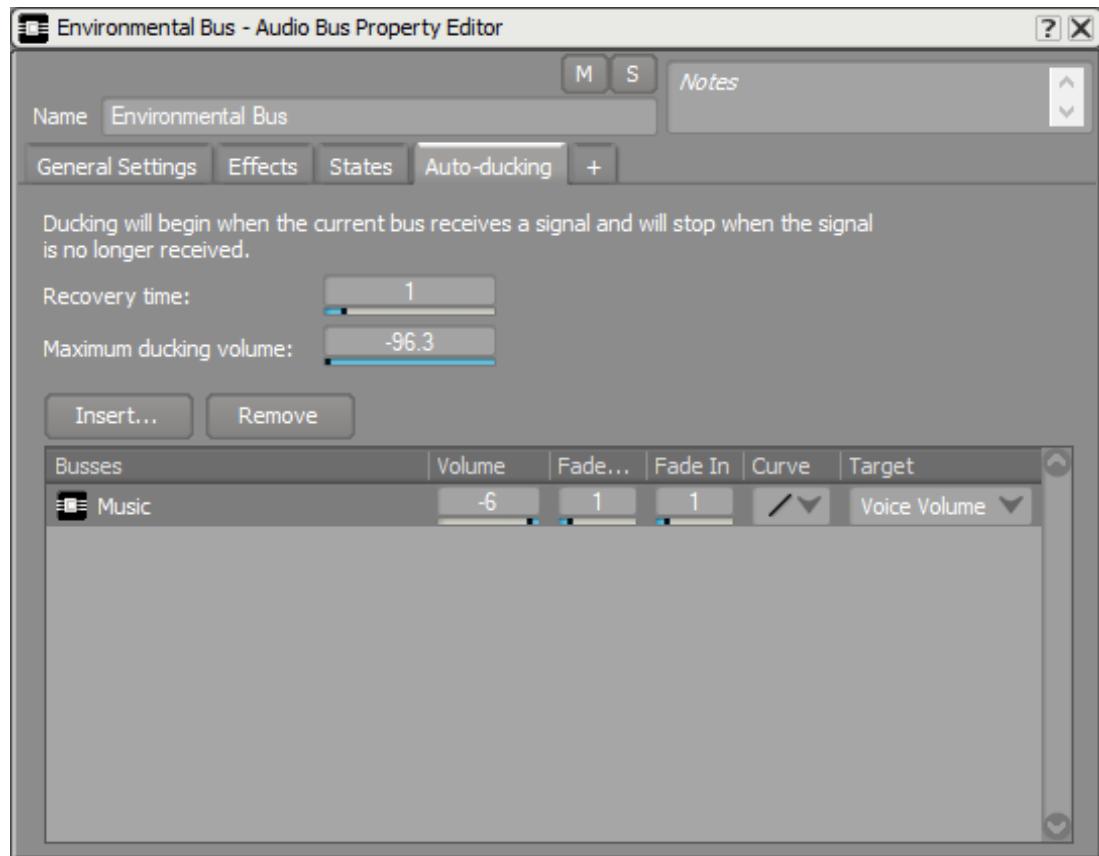


### Tip

このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

## Auto-ducking

Audioバス用のAuto-Ducking（自動ダッキング）タブでは、1つのオーディオ信号の音量レベルを下げて、別の同時オーディオ信号がより顕著になるようにすることができます



#### Tip

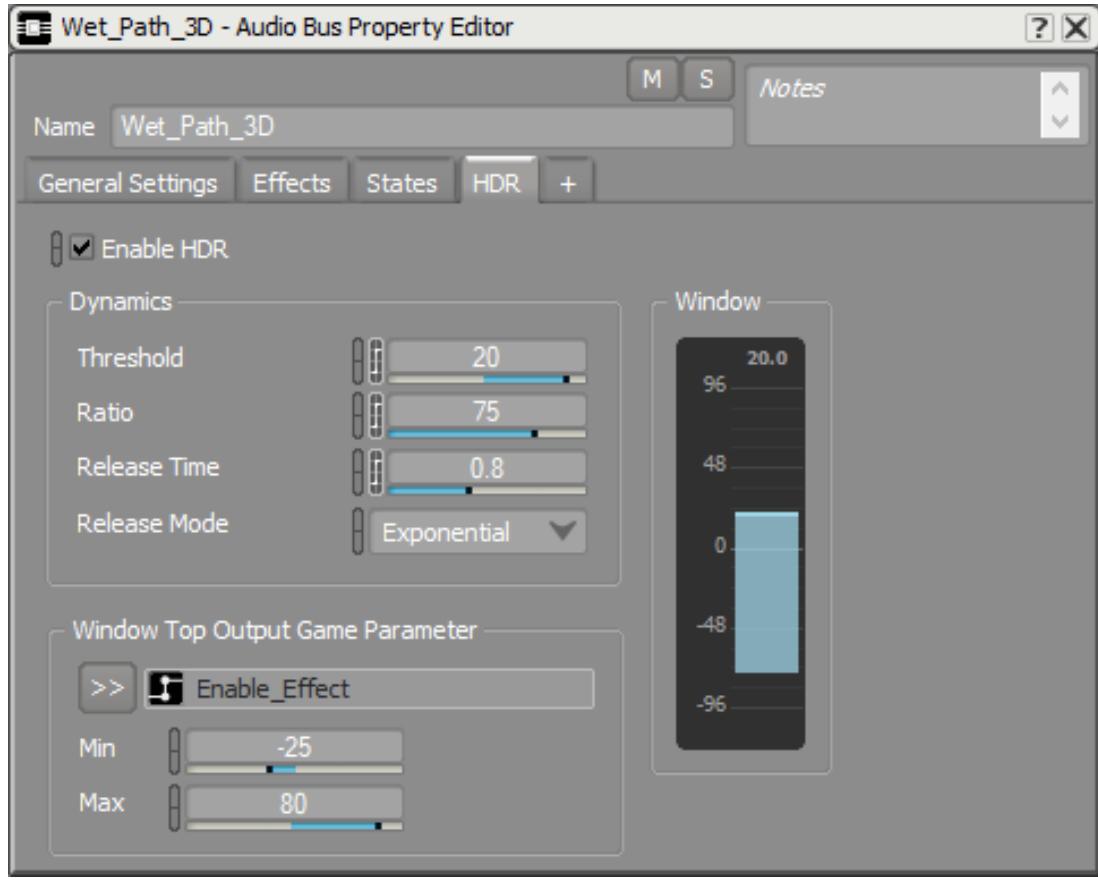
このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

## HDR

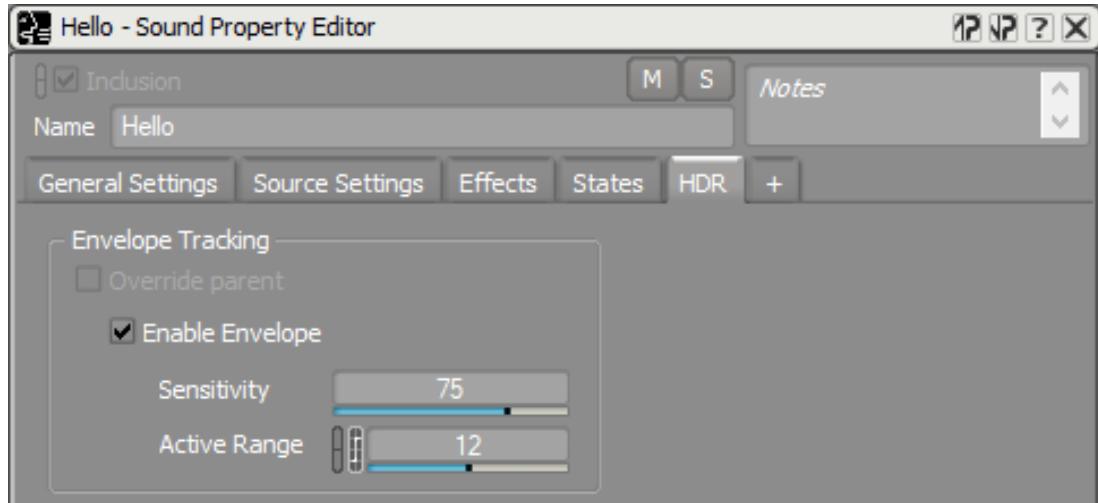
HDR（ハイダイナミックレンジ）オーディオは、自然界で見られるような非常に幅広いダイナミックレンジに渡る様々なレベル値を使って、ミックスするテクニックです。また、HDRは、この幅広いレンジのレベルを、使用するサウンドシステムのデジタルアウトプットに適したレンジにダイナミックにマッピングする、ランタイムシステムでもあります。HDRタブで、選択中のバスや、限定的な範囲でオーディオオブジェクトの、HDR動作やプロパティを定義できます。

--

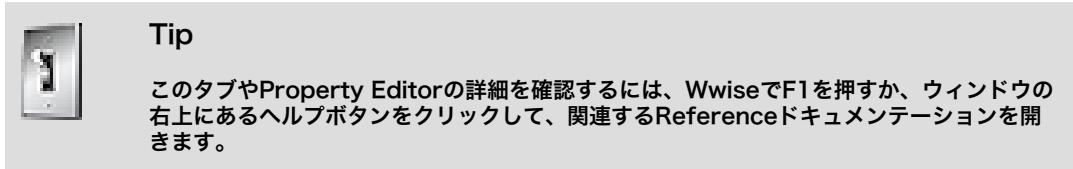
## Property Editor画面を使う



Audio BusのHDRタブ

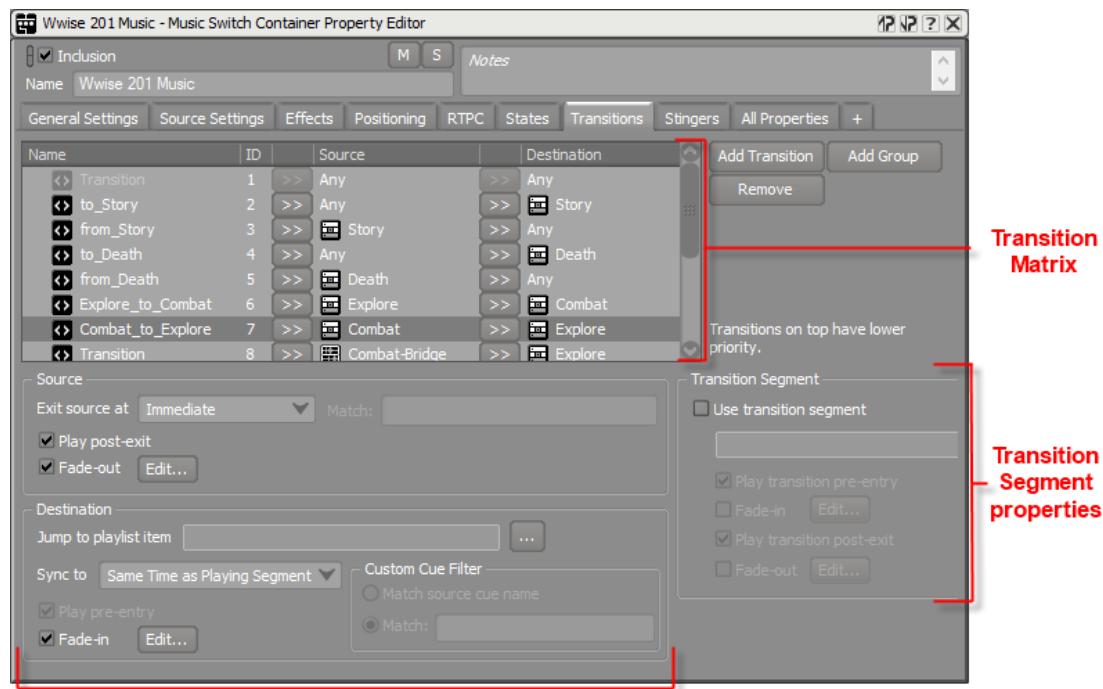


オーディオオブジェクトのHDRタブ

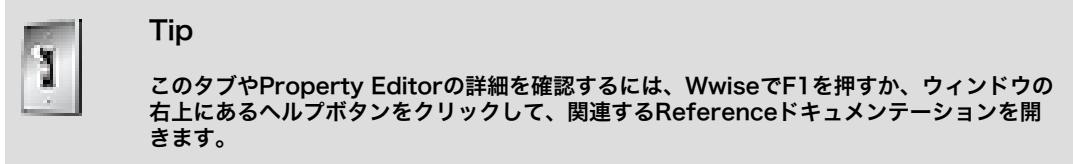


## Transitions

Transitionsタブの中心となるのが、Transitionマトリックス (Matrix / 表) で、Music Switch ContainerやPlaylist Containerの中の各オブジェクトが、同じコンテナ内の他の各オブジェクトにトランジションする時のルールを、この表で設定します。また、トランジションごとに、ソース (Source) とデスティネーション (Destination) のプロパティをカスタマイズできる機能や、トランジションセグメントの有無やそのプロパティを指定する機能もあります。



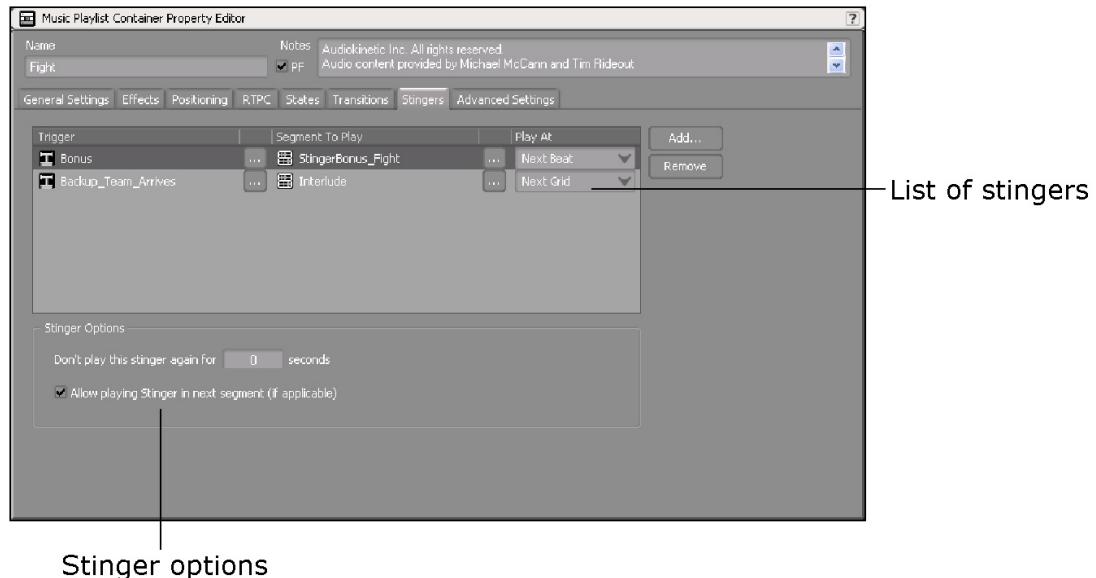
Source and Destination properties



## Stingers

Stingers (ステインガー) タブで、Triggerリストを作成してから、これらのTriggerで再生させるセグメントをアサインして、再生するタイミングを指定します。また、ステインガーごとに、再生 (Play) オプションも設定できます。

## Property Editor画面を使う



Stinger options



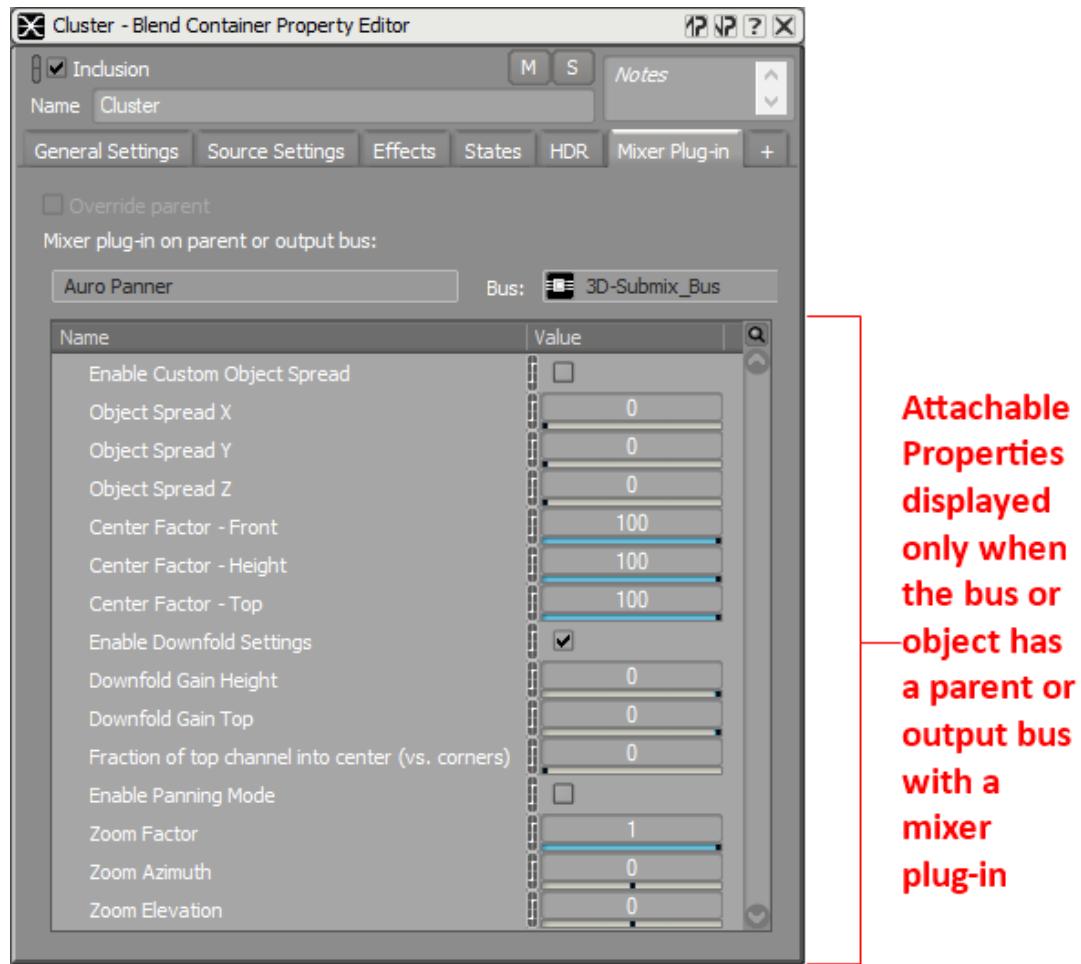
### Tip

このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

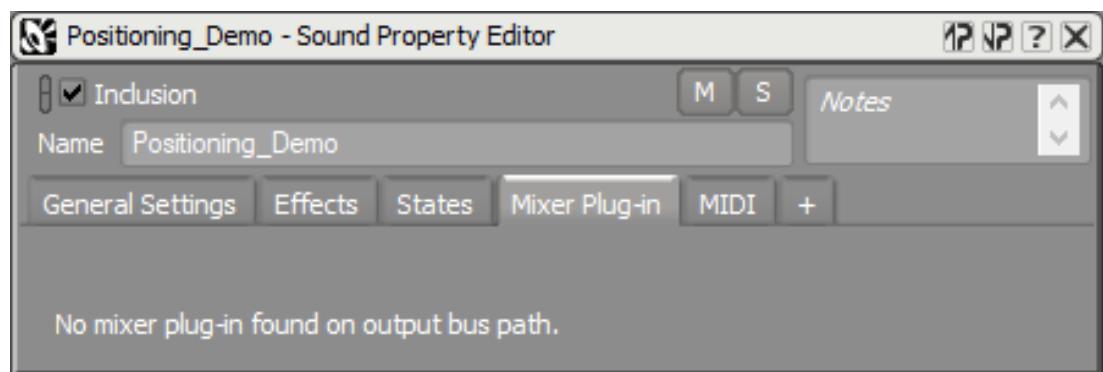
## Mixer プラグイン

Auro Pannerなどのミキサープラグインは、Mixer Plug-inタブでバスに適用します。ミキサープラグイン付きのバスを通る全てのバスやオーディオオブジェクトが、Mixer Plug-inタブに付随するプロパティが表示されるので、必要に応じて調整できます。編集可能なプロパティは、プラグインによって異なります。

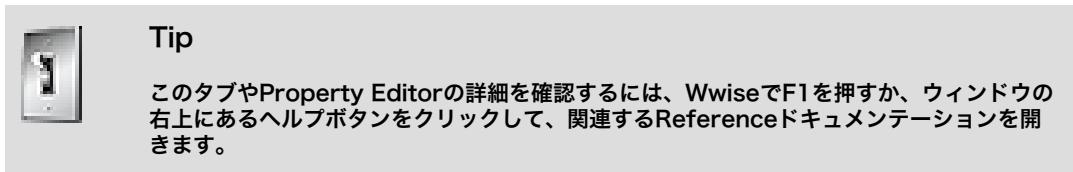
## Property Editor画面を使う



オブジェクトのバス出力パスの途中に適用したプラグインに添付できるプロパティを表示した、Mixer Plug-inタブ

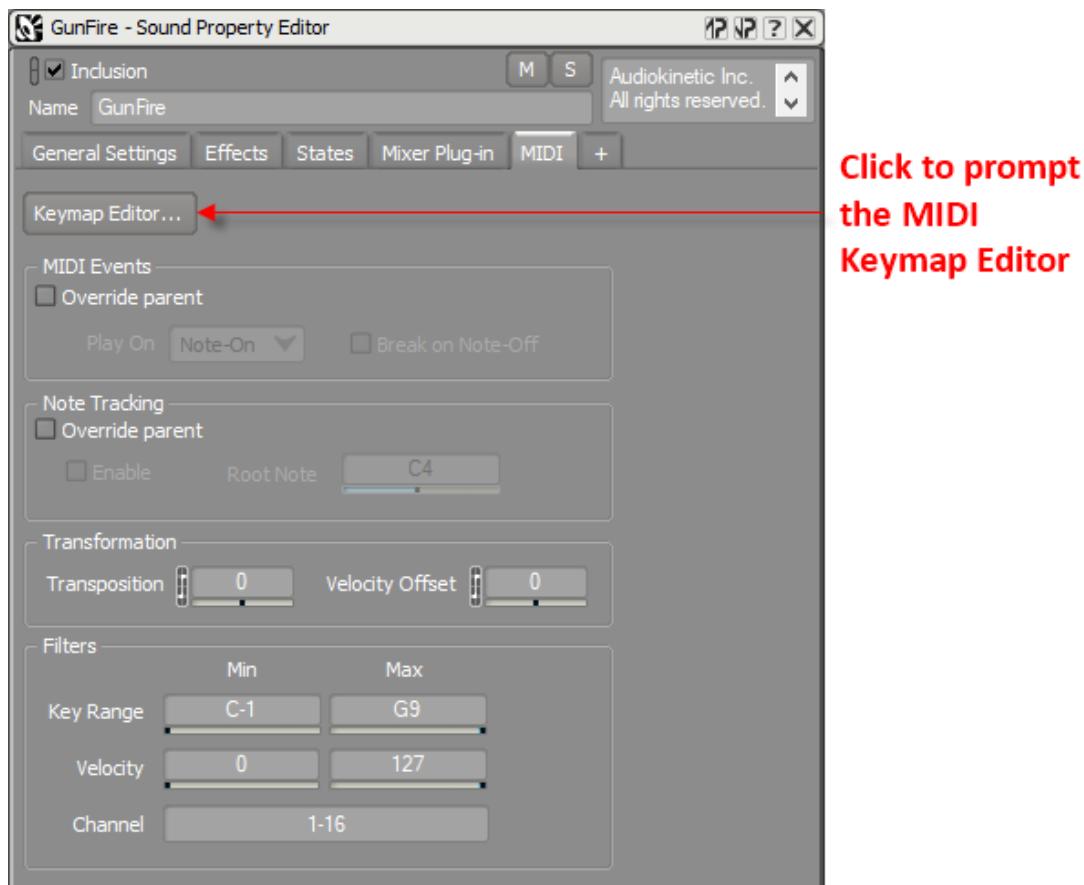


オブジェクトのバス出力パスにミキサープラグインが適用されていない場合の、Mixer Plug-inタブ



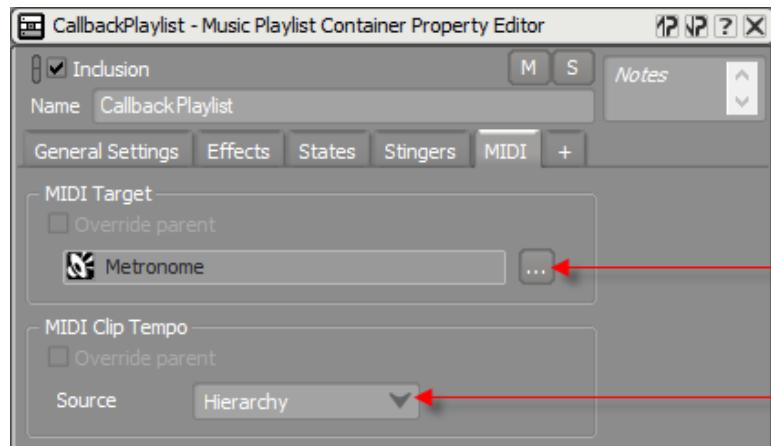
## MIDI

Actor-Mixer HierarchyにあるオブジェクトのMIDIタブで、このオブジェクトをMIDIインストゥルメントとして定義する一連のコントロール機能が入っています。

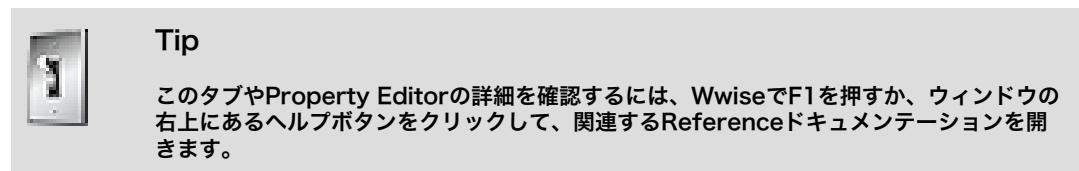


Actor-Mixer HierarchyオブジェクトのMIDIタブ

Interactive Music Hierarchyに位置するオブジェクトのMIDIタブで、そのオブジェクトのMIDIコンテンツのターゲットやテンポを指定できます。Actor-Mixer Hierarchyの再生可能なオブジェクトであれば、ターゲットとして指定できます。テンポは、このインタラクティブミュージックオブジェクトの全てのMIDIコンテンツに適用するグローバル設定値として指定することも、MIDIファイルから取得することもできます。

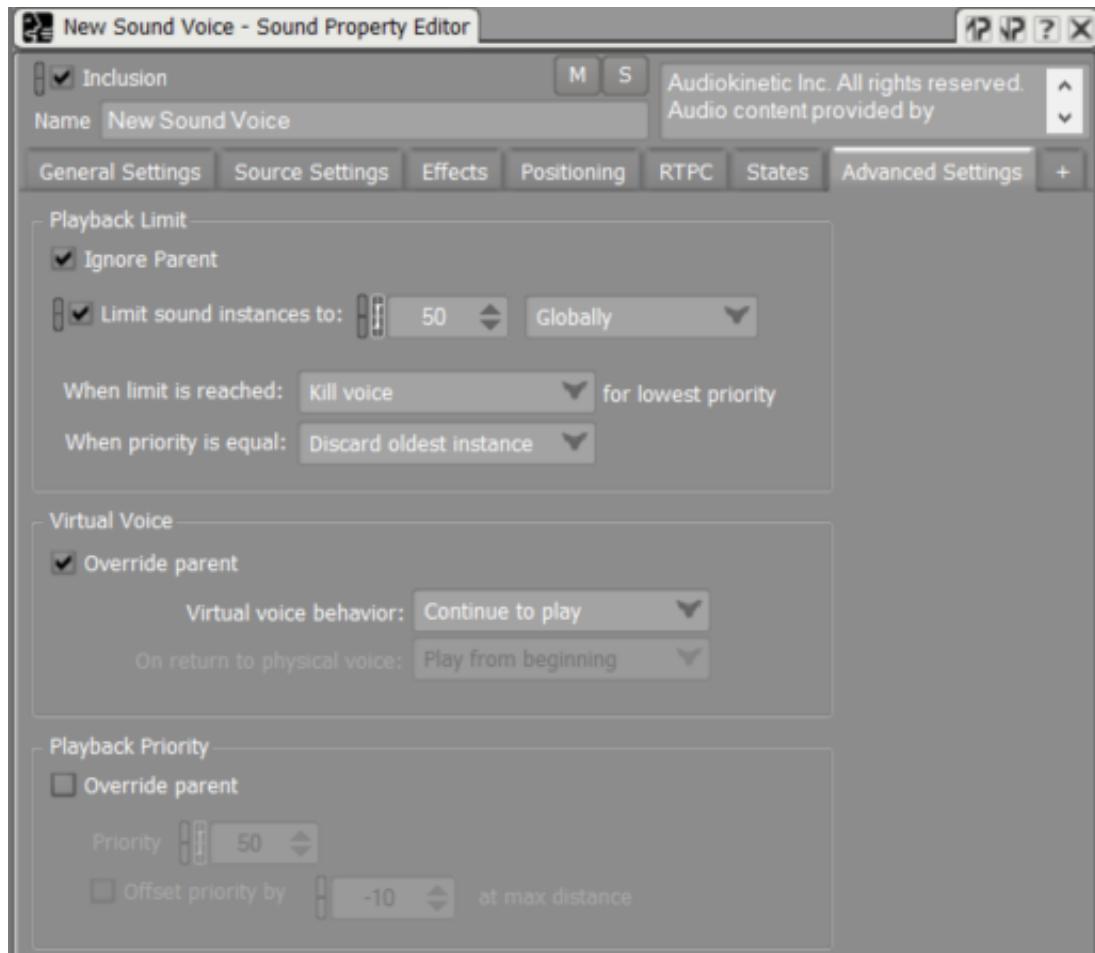


Interactive Music HierarchyオブジェクトのMIDIタブ



### Advanced Settings

Advanced Settings (詳細設定) タブには、いくつかの詳細な動作設定があり、ゲーム中に同時に再生するサウンド、ミュージック、モーションオブジェクトの最大数や、どのオブジェクトのプライオリティが高いのか、また、あるオブジェクトのボリュームが既定のボリューム閾値を下回った時の対処などを、効率的に管理できるようになっています。



### Tip

このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

## Advanced Settingsとダイナミックミキシングを使う

サウンド構造の詳細設定は、オブジェクトのProperty EditorのAdvanced Settingsタブにあり、同時に再生できるサウンド数の制限や、聞こえない時のサウンドの動作などを設定できますが、それぞれ、以下のようなメリットがあります：

- ・パフォーマンス向上と、ミキシングの整理
- ・メモリやCPUの節約

## Advanced Settingsとミキシング

Advanced Settingsの調整を、制作プロセスの遅い段階に後回ししないでください。具体的には、ミキシング前またはミキシング中に、少なくとも1回目の確認を行うようにします。早い段階でミキシングに手を加えすぎて、後でオーディオがリソースを消費しすぎていることに気づき、Advanced Settingsを積極的に使い、オーディオ処理を制約範囲内に収めようとすると、ミックスが大々的に変わってしまう恐れがあります。

Playback Limit（再生制限）は、実はミキシングを助ける機能です。ダイナミックミキシングの一種として利用すれば、ユーザーがサウンドに圧倒されずに重要なサウンドにフォーカスできます。この他にも、バスダッキング、Set Voice Volume Action、State、RTPCなどを使ってミックスを整理できます。

### Playback Limit、Priority、Volume Threshold以下の動作

Playback Limit（再生制限）をサウンド構造に適用すると、同時再生できる最大サウンド数を、ゲームオブジェクト毎に設定したり（Actor-Mixer階層や<e1>Interactive Music</e1>階層の場合）、グローバル（全体）に設定したり（<e2>Master-Mixer Hierarchy</e2>のバスの場合）できます。再生制限のロジックは、再生するサウンドの数に基づいています。しかし、サウンド・デザイナーは階層全体に、様々なオブジェクトへ別々の制限を適用することができます。そして、希望によって、Ignore Parent オプションを選ぶことで別々の階層を作ることができます。つまり、再生制限のそれぞれのレベルはサウンド・デザイナーに、強力かつ簡単に何を再生するかのコントロールを提供します。

再生制限の条件を確認してから、サウンド再生が試みられます。サウンドを再生しようとする時に、既に再生制限に達していれば、このサウンドまたは他のサウンドが停止されます。最初の判断材料がサウンドのプライオリティです。2つのサウンドのプライオリティが同じケースでは、サウンドエンジンが最古または最新のサウンドインスタンスを停止しますが、これはWhen limit is reached（制限に達した場合）や、When priority is equal（プライオリティが同等の場合）の項目に設定されたプロパティで決まります。



#### 注記

Playback Limitによってキル（kill）され消されたサウンドは、再生カウントが制限以下に戻っても、再びスタートすることはできません。無限にループするアンビエントサウンドなどでは、特に注意してください。

Priority設定は、Playback Limitと一緒に機能します。制限システムのバランスを保つために、階層内のプライオリティを調整してください。絶対に制限システムでキルしてはいけないサウンド、例えばナレーション、BGM、ループするアンビエントサウンドなどは、最高のプライオリティを付与します。さらに、サウンドとリスナーの距離によって最終的なプライオリティが左右されることもあります。

状況がunder Volume Threshold（閾値以下）の設定動作は、Playback LimitやPriorityと無関係です。これは、聞こえなくなったサウンドの動作を設定しているだけです。サウンドが聞こえるかどうかを判断するために、Wwiseはメタデータボリューム、つまり階層、バス、State、RTPC、そしてSet Voice Volumeアクションの全ボリュームを合算した結果を見るだけです。WAVデータの分析は一切行われません。

### ダイナミックミキシングのテクニック

以下のダイナミックミキシングテクニックが、Advanced Settings タブで設定できます。

- バスやアクターミキサー構造の再生を制限して、大事なサウンド用にスペースを確保する

例えば、激しいアクションや爆発などプレイヤーが集中すべきエレメントがある場面では、アンビエントサウンドやフォーリーサウンドの数を減らします。アンビエントやフォーリーが銃撃や爆発とかち合うバスを見つけて、このバスに制限をかけ、前者のプライオリティを下げます。

- 距離を基準としたオフセットを、プライオリティに適用する

例えば、様々なアンビエンスサウンドのうち、接近しているものにフォーカスがあたるよう、距離単位のプライオリティを適用した再生制限を行います。**Offset priority by**オプションで、最大距離におけるプライオリティオフセットを指定して、オフセット値を0から**at max distance**までの間で補間します。

- 重要でないサウンドのボリュームをダッキングする

再生制限を全く使えない場合もあります。例えば、あるレベルの最初にアンビエントサウンドを無限ループ再生し始めた場合は、再生制限によって停止させてしまうと、状況が落ち着いてからも再び再生されることはないので、避けるべきです。このようなケースや、その他の妥当だと思われるケースでは、ゲームのオーディオで他の重要な分野にアクションがあった時に、適切なテクニックを使ってアンビエントサウンドのボリュームをダッキングさせると良いでしょう。例えば、ナレーション中やコンバット中は、アンビエンスをシャットダウンすることもできます。また、グレネードが真横で爆発したら、ランプの電池音などは聞こえなくとも構わないでしょう。この処理は、メタデータのサイドチェインととらえることができます。ボリュームの増減を、コントロールバスのバスダッキング機能でトリガーさせたり、特定イベント内の**Set Voice Volume**アクションでトリガーさせたりします。

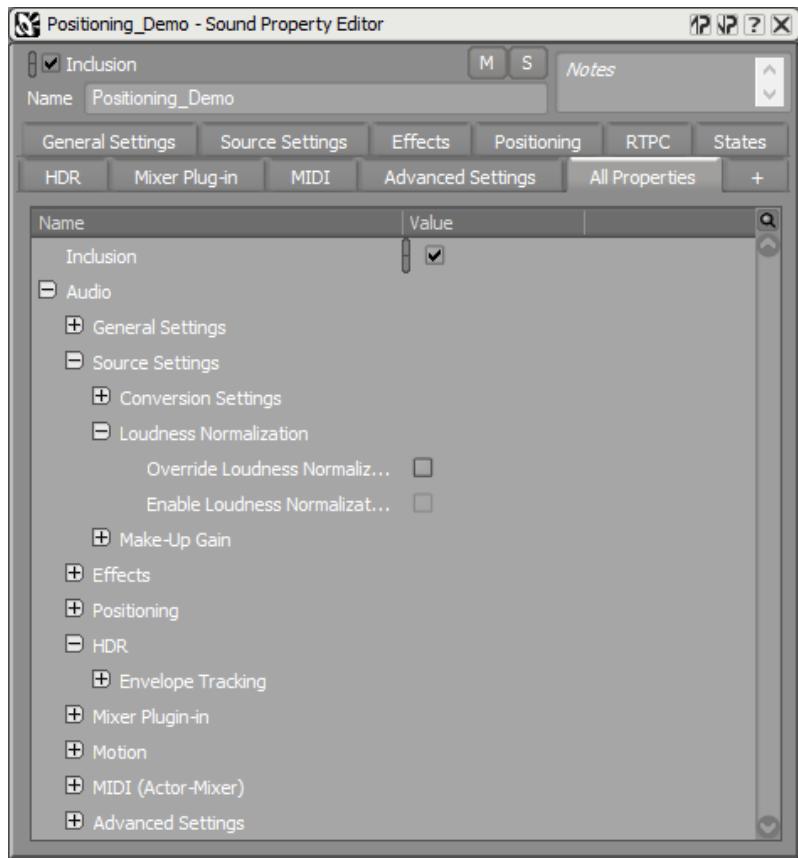
ミキシング内容を整理して、特定の状況では重要でないサウンドのボリュームが下がるように設定できたら、それらのサウンドの**under Volume Threshold**動作を微調整して、聞こえない時の消費CPUやメモリを最小限に抑えます。

- コード側のダイナミックミキシングシステムの実装

なお、再生制限、プライオリティ設定、そしてプライオリティオフセットは、RTPC経由でゲームに露出させることもできます。設定値をゲームで起きている内容によって変化させるシステムを実装するのも、作成者の自由です。

### All Properties

All Propertiesタブでオブジェクトのプロパティを編集します。前のタブに関するページで説明した通り、オブジェクトタイプによって編集可能なプロパティが異なります。



All the object's properties, across all tabs, are accessible in the All Properties tab of the Property Editor

 **Tip**  
このタブやProperty Editorの詳細を確認するには、WwiseでF1を押すか、ウィンドウの右上にあるヘルプボタンをクリックして、関連するReferenceドキュメンテーションを開きます。

## Property Editorを使って作業する

Property Editorにプロジェクトの様々なオブジェクトやゲームシンクのプロパティや動作を設定するための選択肢、フィールド、スライダ、リスト、ボタン、グラフビューなどが表示されます。このようなツールの使い方は、以下のセクションを参照してください。

- 「テキストボックスの使用」
- 「リストの使用」
- 「スライダの使用」
- 「43章グラフビューについて」

Property Editorでプロパティ値を調整する時に、Randomizerを適用したり、この値を全てのプラットフォームでリンクしたり、アンリンクしたりできる場合があります。Wwiseではこれらの機能をアイコンで表示します。また、プロパティ値が RTPCを使ってGame Parameterにアサインされているかどうかも表示されます。ここで使われるアイコンを下表に示します。

アイコン	名前	内容
	Inclusion	このオブジェクトが含まれるかどうかを示す。選択すると、このオブジェクトが含まれる。選択しないと、含まれないことを示す。デフォルトでは全プラットフォームに適用します。プラットフォーム専用のカスタマイゼーションを設定したり決定するには <a href="#">Link indicator</a> （チェックボックス左側）を使います。
	Link	チェックなしの場合は、Property Editorのプロパティや動作を設定できない。 このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしている。
	Unlink	このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしていない。
	Partial Unlink	このプロパティ値は、現在のプラットフォームでは他のアクティブなプラットフォームにリンクしているが、他の1つ以上のプラットフォームにおいて、この設定値がリンクしていない。
	Link Mixed	選択されたオブジェクトによっては異なるリンクステータスがあります。リンクされていたり、リンクされていなかったり、一部だけリンクされています。
	RTPC Disabled RTPC無効	このプロパティ値は、ゲーム内のパラメータ値に結び付いていない。
	RTPC Enabled RTPC有効	このプロパティ値は、ゲーム内のパラメータ値に結び付いている。例えば、ゲーム内の車のスピードを、Wwiseにおけるピッチのプロパティに直接、結びつけることができる。ゲーム中に車のスピードが加速すると、Wwise側のピッチもリアルタイムで上昇する。
	RTPC Mixed	ゲーム内のパラメータ値は、Multi Editorにロードされたオブジェクトの一つ以上のすべてではないプロパティ値にを関連付けられています。Property Editor やContents Editorではこのインジケーターが見られません。
	Randomizer Enabled	Randomizerエフェクトが適用されたプロパティ値。
	Randomizer Disabled	Randomizerエフェクトが適用されていないプロパティ値。
	Randomizer Mixed	Randomizer エフェクトは Multi Editor にロードされた1つ以上のすべてではないプロパティ値に適用されます。Property Editor やContents Editorではこのインジケーターが見られません。
	State Disabled	このプロパティ値は、ステートに結びついていない。
	State Enabled	ステートグループがこのプロパティ値に結びついている。ボリュームなど、ここで対象となっているプロパティが、適用されるステートによって変化することを意味します。

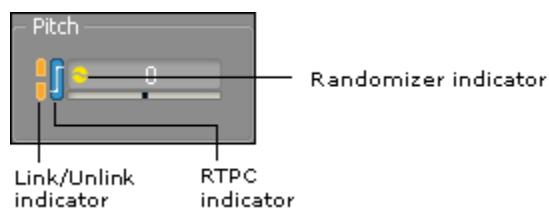
## プロパティ値のランダム化

ゲームサウンドのリアリズムを強化するために、Property Editorにある大部分のプロパティ値に、Randomizerを適用できます。Randomizer機能で、あるプロパティ値の有効範囲を指定します。このオブジェクトが再生されると毎回、Wwiseが、指定した範囲内の異なるプロパティ値を選択します。それぞれのプロパティ値にRandomizerを適用することで、このオブジェクトは、再生される度に、聞こえ方が変化します。

Randomizerの詳細と使い方については「[ランダム化したプロパティ値で、再生を強化](#)」を参照してください。

### プロパティ値のリンク・アンリンク

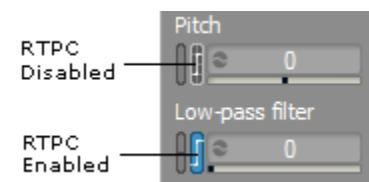
Wwiseでは、1つのオブジェクトに関連する様々なプロパティ値の大部分を、リンク・アンリンクできるので、同時に複数のプラットフォーム向けにオーサリングできます。デフォルトでは、複数のアクティブプラットフォームにおいて、すべてのプロパティ値はリンクされていますので、同じ値に設定できます。あるプロパティ値をアンリンクすると、そのプラットフォームでカスタマイズできるようになります。



プロパティ値のリンク・アンリンクについては「[プラットフォームごとに、オブジェクトプロパティをカスタマイズする](#)」を参照してください。

### プロパティ値をGame Parameterにアサインする

前述の通り、Game ParameterをWwiseの特定プロパティ値にアサインできます。この機能が、RTPC (Real-time Parameter Controls) です。Wwiseでは、RTPCとしてアサインされたプロパティが分かるように、視覚的フィードバックを表示します。Game Parameterにアサインされたプロパティ値は、そのRTPCインジケータが青く表示されます。



RTPCの使い方については「[Game Parameterを使って、プロパティ値をコントロールする](#)」を参照してください。

### プロジェクトエレメントのプロパティを表示する

Project Editorには、プロジェクトの1つのプロジェクトエレメントに関連する、全てのプロパティが表示されます。Property Editorに特定のオブジェクトをロードするには、ビューに表示されたオブジェクトをダブルクリックしますが、これができるビューは、Project Explorerの、Audioタブや、Contents Editor、Capture Logなどです。Property Editorにゲームシンクをロードするには、Project Explorerの、Game Syncsタブで、以下のいずれかをダブルクリックします。

- Switch Group

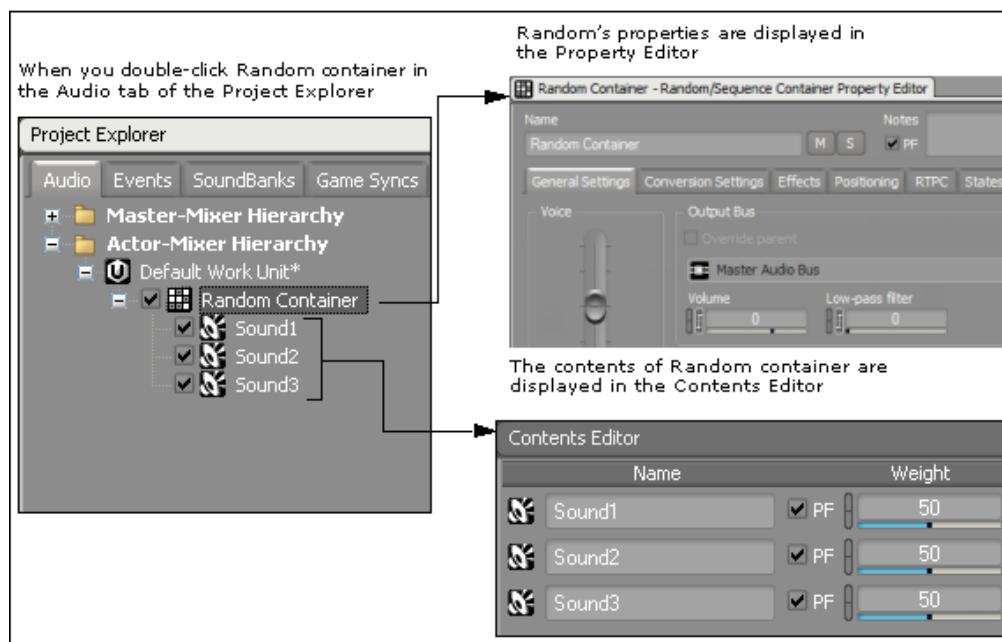
- Switch
- State Group
- State
- Game Parameter (Game parameter)
- Trigger

Designerレイアウトで、プロジェクトエレメントのプロパティを表示するには:

1. Designerレイアウトで、以下のいずれかを行う。
  - Project Explorerの、Audioタブで、オブジェクトを1つダブルクリックする。
  - Project Explorerの、Game Syncsタブで、ゲームシンクを1つダブルクリックする。

以下の通りになります。

- Property Editorに、このオブジェクトのプロパティが表示される。
- Contents Editorに、このオブジェクトまたはゲームシンクに入っている全ての子オブジェクトが、表示される。



---

## 第40章 Contents Editor画面について

概要 .....	775
Contents Editorで作業する .....	782

## 概要

Contents Editorに表示されるのは、Property Editorにロードした親オブジェクトの中にあるオブジェクト（単数または複数）です。Project Editorには、様々な種類のオブジェクト構造やその他のプロジェクトエレメントが含まれるので、Contents Editorは、コンテキスト（内容）に合わせて、つまりロードされたオブジェクトタイプに合わせて、レイアウトを変えて表示します。

Contents Editorに、サウンド構造、ミュージック構造、モーション構造、またはその他のプロジェクトエレメントをロードすると、各オブジェクトに関連する最も一般的なプロパティ、例えばボリュームなど数種類のプロパティが、すぐにアクセスできるようになります。これらのプロパティを、Contents Editorで調整できるので、子オブジェクトを1つずつ、Property Editorにロードしなくても編集できます。また、Contents Editorには、プラットフォーム別にオブジェクトを含める（Include）・排除する（Exclude）ツールや、プレイリストやSwitchの動作を設定するツール、さらにオーディオソース、モーションソース、ソースプラグインを管理するツールなどがあります。

Contents Editorで表示するサウンド構造、ミュージック構造、モーション構造、その他のプロジェクトエレメントの専用ビューを、以下のセクションで説明します。

- [「Sound Objects」](#)
- [「Random Container」](#)
- [「Sequence Container」](#)
- [「Switch Container」](#)
- [「Blend Container」](#)
- [「Actor-Mixer」](#)
- [「Folder」](#)
- [「Bus」](#)
- [「Event」](#)
- [「Switch Group」](#)
- [「State Group」](#)
- [「Music Track」](#)
- [「Music Segment」](#)
- [「Music Playlist Container」](#)
- [「Music Switch Container」](#)

### Sound Objects

Property Editorにサウンドオブジェクトをロードすると、このオブジェクトのソースが全て、Contents Editorに表示されます。サウンドオブジェクトには、以下のような複数のソースが含まれます。

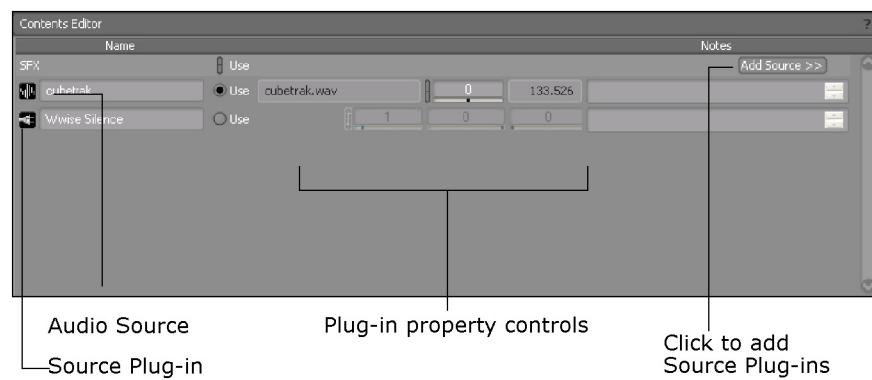
- **Takes** - 同じサウンドオブジェクトの様々なバージョンのことで、採用するソースを決める前に試聴やテストができる。これらのオーディオソースは、オーディ

## Contents Editor画面について

オファイル、サイレンス、プラグインのどれかにリンクするか、この3種の組み合わせにリンクする。

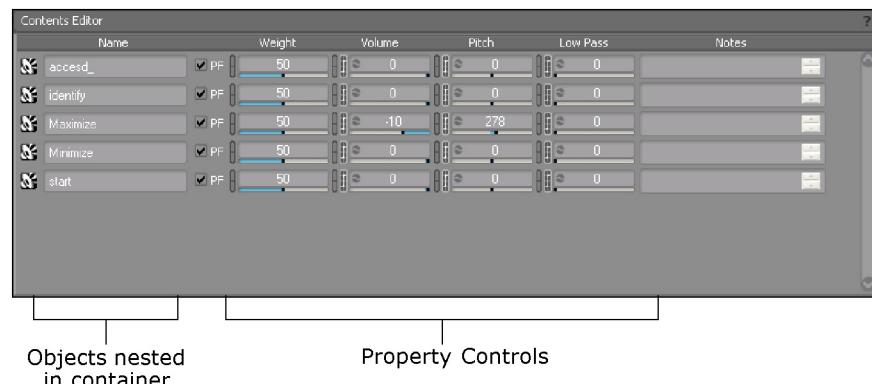
- **Languages** - プロジェクトのローカリゼーション用の様々なランゲージバージョン。 Wwiseにおけるランゲージバージョンの使い方は「[プロジェクトのローカリゼーション作業](#)」を参照。

Contents Editorのソース向けの設定機能やプロパティは、選択したソースタイプによって変わります。



## Random Container

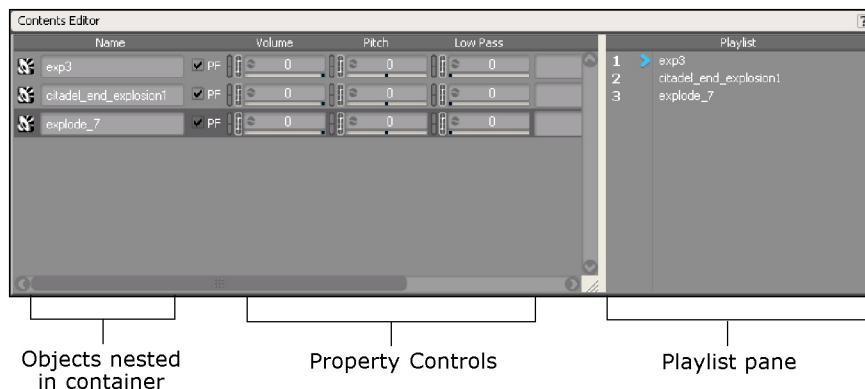
Property EditorにRandom Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示され、オブジェクト別に、プロパティを編集できます。



## Sequence Container

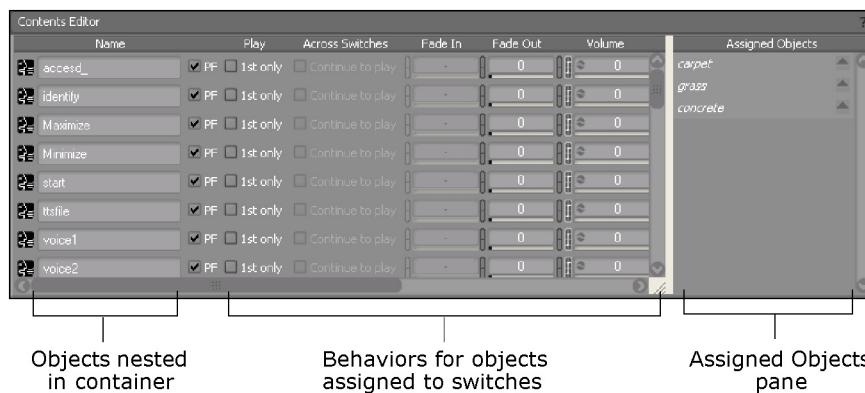
Property EditorにSequence Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示されます。オブジェクト別にプロパティを編集したり、このSequence Containerのプレイリストを作成したりできます。

## Contents Editor画面について



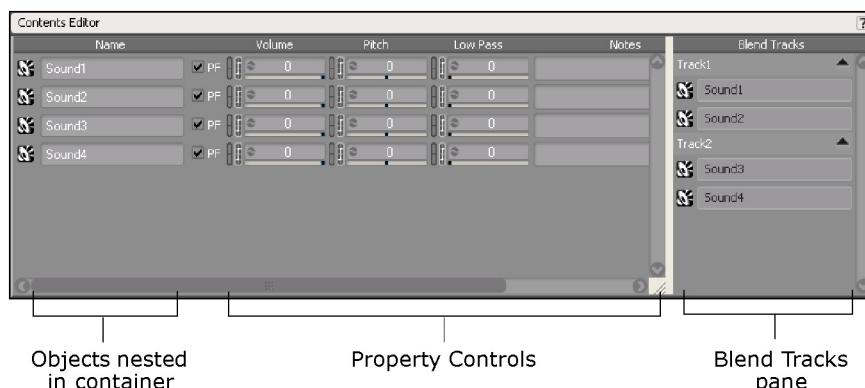
## Switch Container

Property EditorにSwitch Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示されます。子オブジェクトのプロパティを編集したり、Switchにオブジェクトをアサインしたり、ゲームがSwitchを呼び出した時のオブジェクトの動作を、オブジェクトごとに設定したりできます。



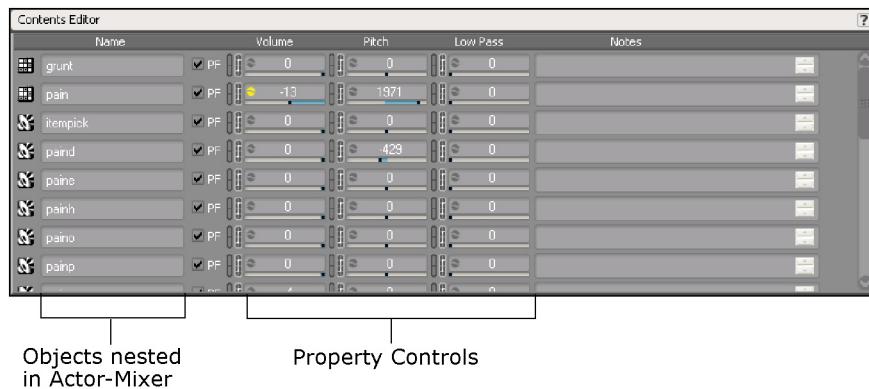
## Blend Container

Property EditorにBlend Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示されます。これらの子オブジェクトのプロパティを編集したり、様々なブレンドトラック（Blend tracks）にアサインしたりできます。



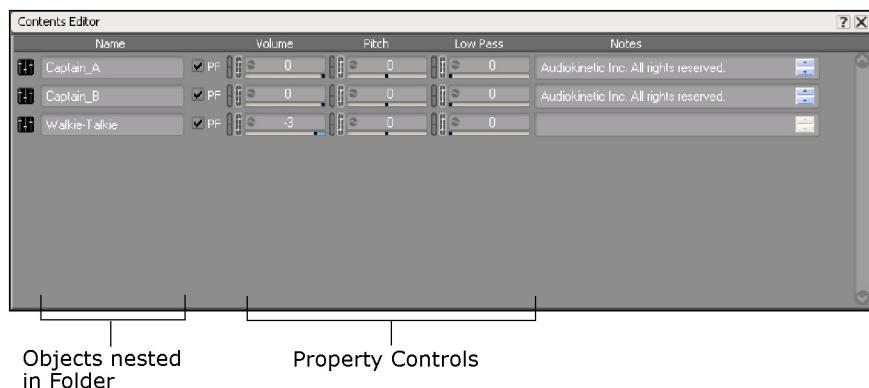
## Actor-Mixer

Property Editorにアクターミキサーをロードすると、その子オブジェクトが全て、Contents Editorに表示されます。アクターミキサーの子オブジェクトのプロパティを、このビューで編集できます。



## Folder

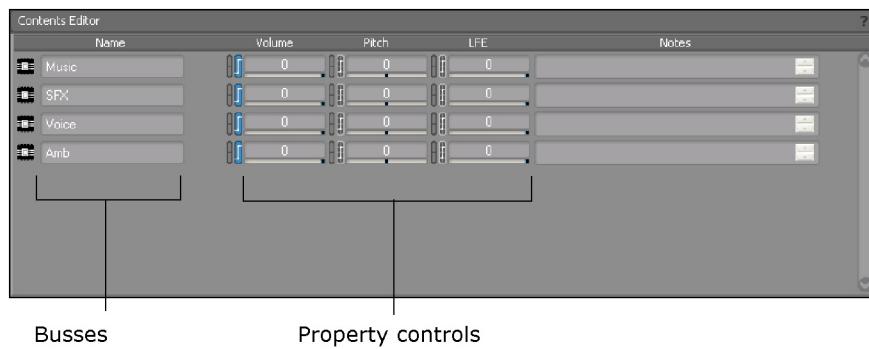
Property Editorにフォルダをロードすると、その子オブジェクトが全て、Contents Editorに表示されます。フォルダの子オブジェクトのプロパティを、このビューで編集できます。



## Bus

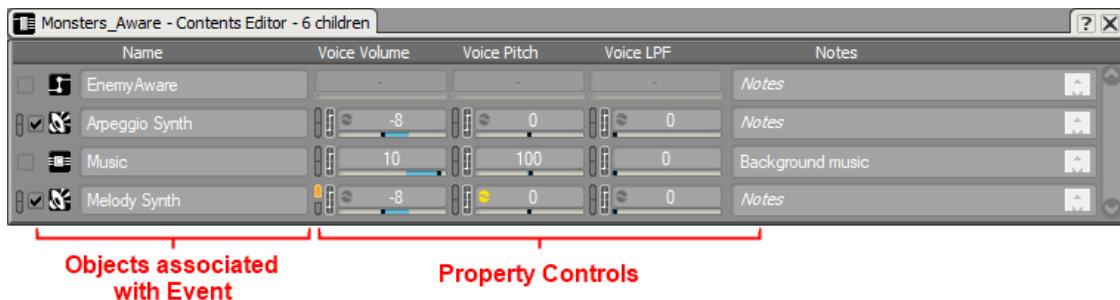
Property Editorにマスターバスまたは親バスをロードすると、その子バスが全て、Contents Editorにロードされます。チャイルドバスのプロパティを、このビューで編集できます。

## Contents Editor画面について



### Event

Event Editorにイベントをロードすると、このイベントに関連するオブジェクトが全て、Contents Editorに表示されます。これらのオブジェクトのプロパティを、編集できます。



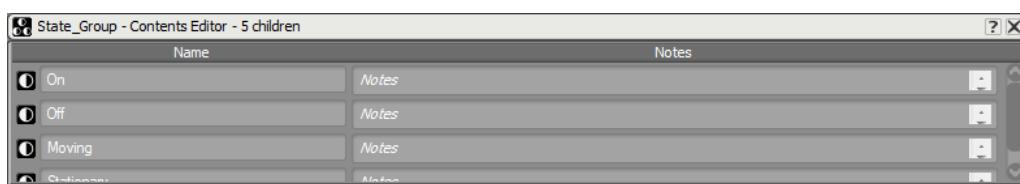
### Switch Group

Property EditorにSwitch Groupをロードすると、このSwitch Groupの中のSwitchが全て、Contents Editorに表示されます。これらのSwitchの名前 (Name) を編集したり、備考列 (Notes) に記入したりできます。



### State Group

Property EditorにStateやState Groupをロードすると、このStateやState Groupの中のStateが全て、Contents Editorに表示されます。これらのStateのプロパティを、編集できます。



## Music Track

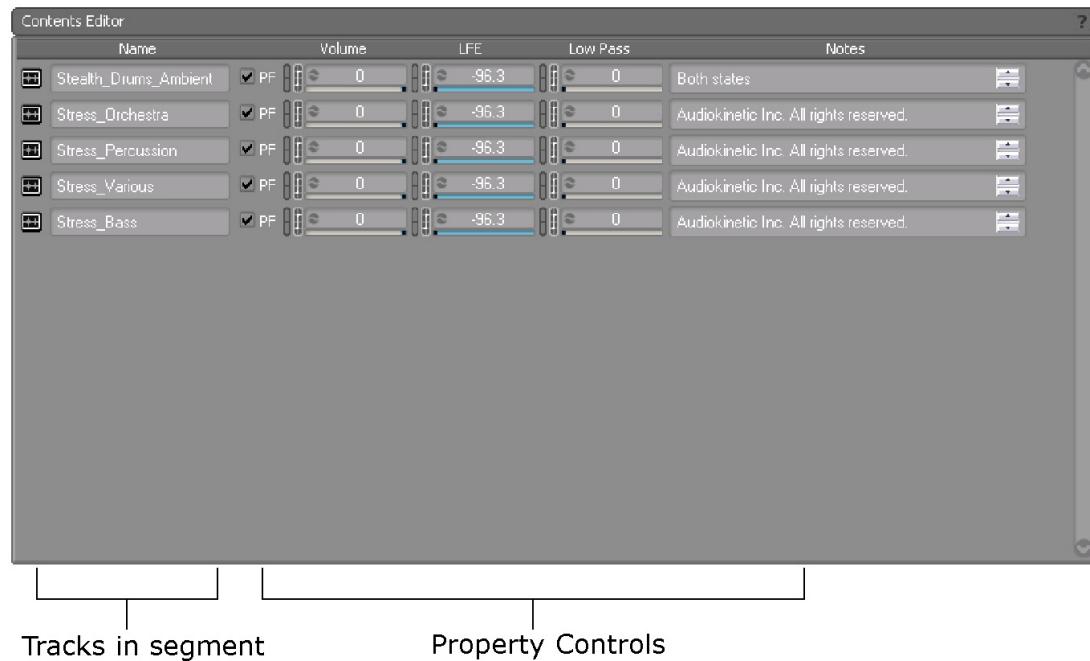
Property Editorにミュージックトラックをロードすると、これがContents Editorに表示されます。プロパティが表示されますが、ここで編集できるのは備考列(Notes)だけです。



## Music Segment

Property EditorにMusic Segmentをロードすると、そのトラックが全て、Contents Editorに表示され、ここでオブジェクトのプロパティを編集できます。

## Contents Editor画面について



## Music Playlist Container

Property EditorにMusic Playlist Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示され、ここでオブジェクトのプロパティを編集できます。



## Music Switch Container

Property EditorにMusic Switch Containerをロードすると、その子オブジェクトが全て、Contents Editorに表示され、ここでオブジェクトのプロパティを編集できます。



## Contents Editorで作業する

Contents Editorには、オブジェクト、State、Switchなどの、プロパティや動作を設定するための、フィールド、リスト、オプション、スライダなどが表示されます。このようなツールの使い方は、以下のセクションを参照してください。

- ・「テキストボックスの使用」
- ・「リストの使用」
- ・「スライダの使用」

Contents Editorで、オブジェクトを個別に再生したり、ビューに表示された様々なオブジェクトの並べ替え、コピー、ペースト、削除ができます。このビューで設定できるオプションやプロパティは、現在ロードされているオブジェクトによって変わります。多くのビューでは、一部のプロパティスライダの横にアイコンが表示され、特定のプロパティがアサインされているかどうかが、表示されます。ここで使われるアイコンを下表に示します。

アイコン	名前	内容
Link	Link	このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしている。

## Contents Editor画面について

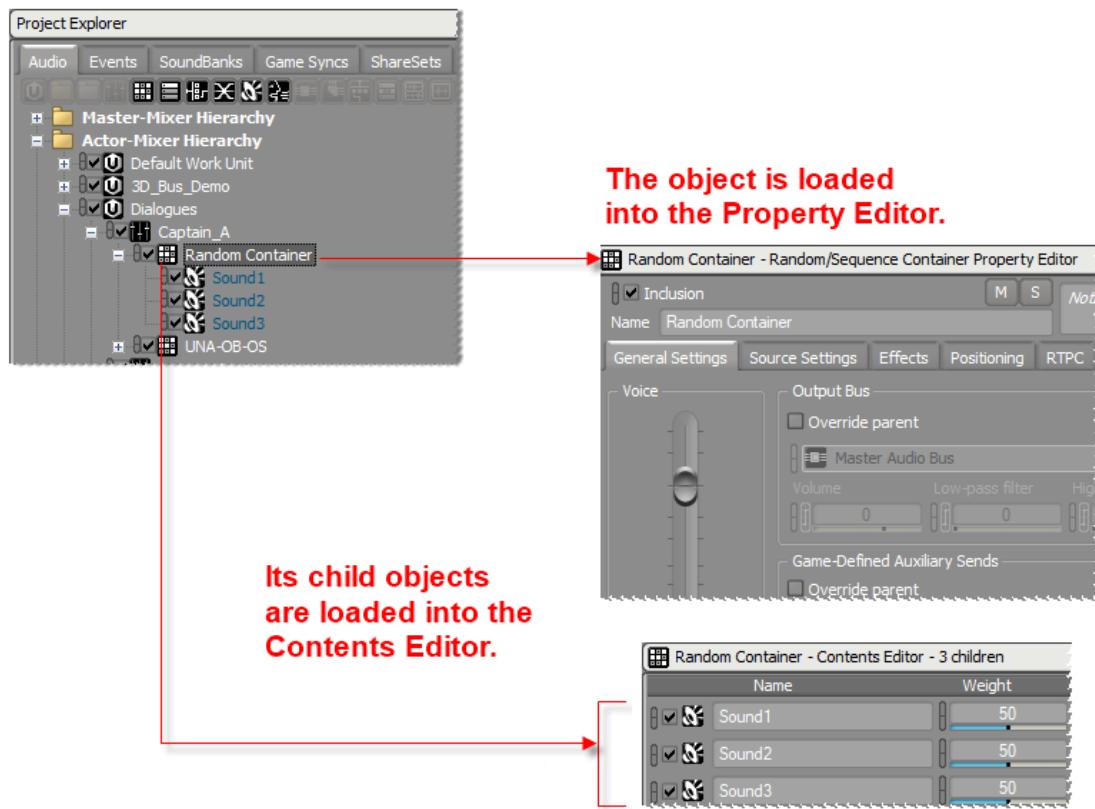
---

アイコン	名前	内容
	Unlink	このプロパティ値は、他のアクティブなゲームプラットフォームの設定値とリンクしていない。
	Partial Unlink	このプロパティ値は、現在のプラットフォームでは他のアクティブなプラットフォームにリンクしているが、他の1つ以上のプラットフォームにおいて、この設定値がリンクしていない。
	Link Mixed	選択されたオブジェクトによっては異なるリンクステータスがあります。リンクされてたり、リンクされていなかったり、一部だけリンクされています。
	RTPC Disabled RTPC無効	このプロパティ値は、ゲーム内のパラメータ値に結び付いていない。
	RTPC Enabled RTPC有効	このプロパティ値は、ゲーム内のパラメータ値に結び付いています。例えば、ゲーム内の車のスピードを、Wwiseにおけるピッチのプロパティに直接、結びつけることができる。ゲーム中に車のスピードが加速すると、Wwise側のピッチもリアルタイムで上昇する。
	RTPC Mixed	ゲーム内のパラメータ値は、Multi Editorにロードされたオブジェクトの一つ以上のすべてではないプロパティ値にを関連付けられています。Property Editor やContents Editorではこのインジケータが見られません。
	State Disabled	このプロパティ値は、ステートに結びついていない。
	State Enabled	ステートグループがこのプロパティ値に結びついている。ボリュームなど、ここで対象となっているプロパティが、適用されるステートによって変化することを意味します。
	Randomizer Enabled	Randomizerエフェクトが適用されたプロパティ値。
	Randomizer Disabled	Randomizerエフェクトが適用されていないプロパティ値。
	Randomizer Mixed	Randomizer エフェクトは Multi Editor にロードされた1つ以上のすべてではないプロパティ値に適用されます。Property Editor やContents Editorではこのインジケータが見られません。

## Contents Editorにオブジェクトを表示させる

階層のオブジェクトを1つ、Property Editorにロードすると、その子オブジェクトが全て、Contents Editorに表示されます。例えばProject ExplorerのAudioタブでRandom Containerを選択してProperty Editorにロードすると、その子セグメントがContents Editorにロードされます。

**Double-click an object in the Audio tab of the Project Explorer.**



また、Contents Editorのオブジェクトを1つダブルクリックすると、このプロジェクト階層を下方向に移動します。

そのまま下方向にプロジェクトツリーを移動して、オブジェクトに関連付いているソース（単数または複数）の段階まで下がることができます。

**Double-click an object in the Contents Editor**

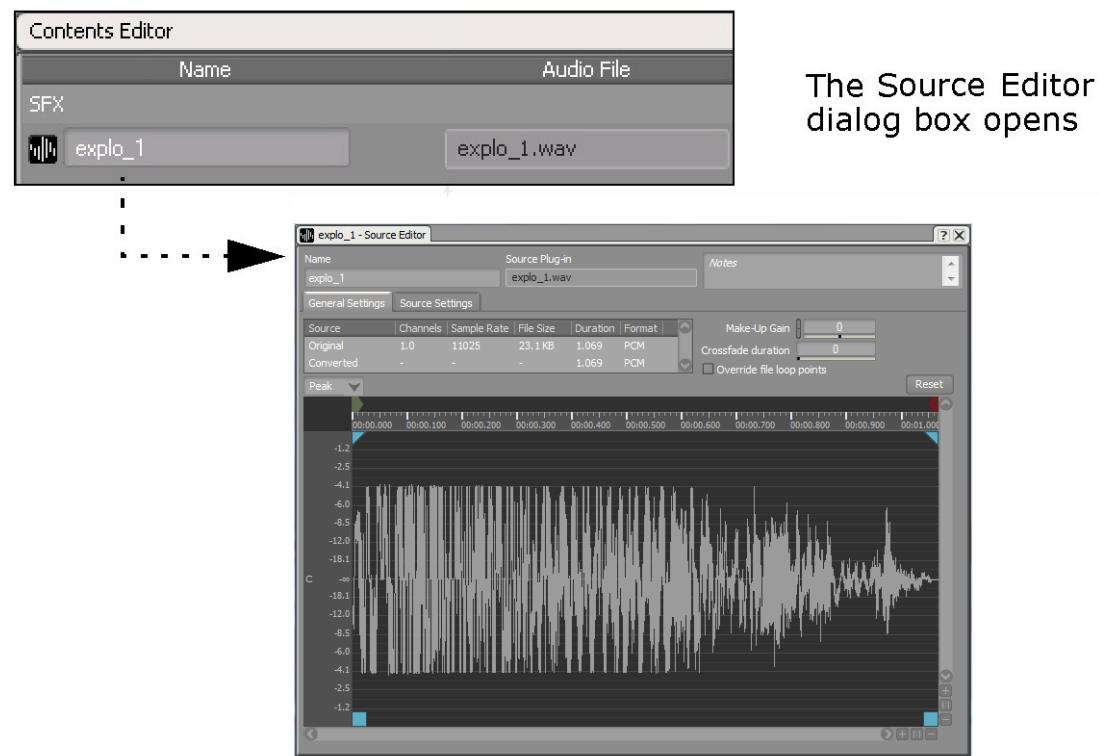




### Contents Editorのソースの設定を変える

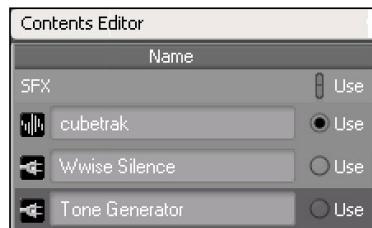
Contents Editorで、ソースレベルまで到達すると、オーディオソースやソースプラグインの設定を変更できます。オーディオソースを1つダブルクリックすると、Source Editorダイアログボックスが開きます。

Double-click a source in the Contents Editor

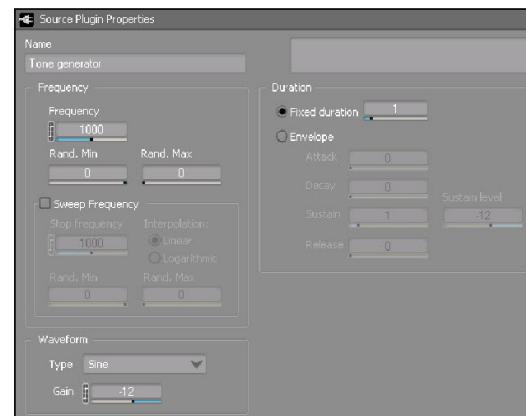


ソースプラグインを1つダブルクリックすると、プラグインプロパティがProperty Editorに表示されるので、編集できます。

Double-click a source in the Contents Editor



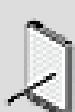
The Source Plug-ins properties are displayed in the Property Editor



### Contents Editorにオブジェクトを追加する

Property Editorにオブジェクトをロードすると、同時に、このオブジェクトのコンテンツ（中身）が、Contents Editorに表示されます。また、Project Explorerにあるオブジェクトを、Contents Editorにドラッグすることで、Contents Editorにこのオブジェクトを直接追加できます。なお、Contents Editorにオブジェクトをドラッグして入れると、このオブジェクトが、その階層における現在の位置から、Contents Editorに表示されたオブジェクトの親の下に、移動するので、注意してください。このオブジェクトを移動させたくない場合は、Ctrlを押しながらドラッグして、Contents Editorに入れます。これで、オブジェクトがコピーされ、Contents Editorの他のオブジェクトに追加されます。

また、オブジェクトにソースを追加するには、Contents Editorに直接インポートする方法や、ソースプラグインを追加する方法もあります。



#### 注記

Contents Editorに表示されるオブジェクトは、設定した順番に並べられます。Contents Editorに追加されたオブジェクトは、Project ExplorerのAudioタブで、親の下にアルファベット順に並べられる。

### Contents Editorのオブジェクトを並べ替える

Contents Editorに表示されたオブジェクトやソースの順番を、必要に応じて変更できます。この操作は、Project Explorerに表示される順番に、影響しません。

#### オブジェクトの順番を変更するには:

1. Contents Editorで、順番を変えたいオブジェクトをドラッグして、新しい位置に移動する。

このオブジェクトが挿入される位置の上に、赤いインジケータが表示される。

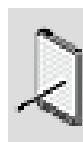
## 関連トピック

- 「[Contents Editorにオブジェクトを追加する](#)」
- 「[Contents Editorのペイン間で、オブジェクトをドラッグする](#)」
- 「[リストの展開・縮小](#)」
- 「[オブジェクトを削除する](#)」
- 「[Contents Editorのオブジェクトやソースを試聴する](#)」

## Contents Editorのペイン間で、オブジェクトをドラッグする

Switch Container、Sequence Container、またはBlend Containerで作業をしている場合は、Contents Editorが2つのペインに分割されます。左側のペインにコンテンツの子オブジェクトが、右側のペインにリストが、表示されます。オブジェクトをペイン間でドラッグすることで、プレイリストを作成したり、オブジェクトをSwitchやブレンドトラックにアサインしたりできます。

あるオブジェクトをSwitchにアサインするには、このオブジェクトを、オブジェクトペインから、Assigned Objects（アサインされたオブジェクト）ペインにドラッグします。また、Project ExplorerのAudioタブに表示されたオブジェクトを、Switchにドロップすることもできます。オブジェクトとSwitchの使い方については「[Switch Containerの中身と、再生方法の設定](#)」を参照してください。



### 注記

あるSwitchが空で、このSwitchにオブジェクトがアサインされていない場合は、イタリック体で表示されます。

Sequence Containerのプレイリストを作成するには、オブジェクトのペインから、Playlistペインに、オブジェクトをドラッグして入れます。プレイリストやSequence Containerの使い方については「[プレイリストの作成](#)」を参照してください。

ブレンドトラックにオブジェクトをアサインするには、オブジェクトのペインから、Blend Tracksペインに、オブジェクトをドラッグして入れます。また、Project ExplorerのAudioタブに表示されたオブジェクトを、ブレンドトラックにドロップできます。ブレンドトラックの使い方については「[ブレンドトラックのオブジェクトの追加、削除](#)」を参照してください。

## リストの展開・縮小

Contents Editorの各種ペインに表示される複数のオブジェクトで作業しやすくするために、リストの展開と縮小ができます。この機能は、以下の場合に利用できます。

- **Switch Container - Assigned Object**リストの展開・縮小。
- **Blend Container - Blend Tracks**リストの展開・縮小。
- **ランゲージ - ランゲージ別のソースリスト**の展開・縮小。

## リストを展開・縮小するには:

1. 以下のいずれかを行う:

- ・縮小するリストの横の矢印をクリックする。
- ・リストが縮小されます。
- ・展開するリストの横の矢印をクリックする。

リストが展開される。

## ランゲージリストを、全て展開・縮小するには

1. 1つのランゲージのタイトルバーを右クリックする。  
ショートカットメニューが開く。
2. 以下のいずれかを行う:
  - ・全てのランゲージリストを、展開するには、**Expand All**をクリックする。
  - ・全てのランゲージリストを、縮小するには、**Collapse All**をクリックする。

## 関連トピック

- ・「[Contents Editorのオブジェクトを並べ替える](#)」
- ・「[Contents Editorのペイン間で、オブジェクトをドラッグする](#)」
- ・「[Contents Editorにオブジェクトを追加する](#)」
- ・「[オブジェクトを削除する](#)」
- ・「[Contents Editorのオブジェクトやソースを試聴する](#)」

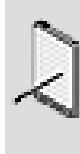
## オブジェクトを削除する

不要となったオブジェクトやソースを、Contents Editorで削除できます。テスト用に複数のソースをインポートして試した後に、どのバージョンを使うかが決まれば、残りを削除できます。Contents Editorでオブジェクトやソースを削除すると、このオブジェクトまたはソースがプロジェクトから削除されます。プロジェクトの.cacheフォルダにある、コンバージョン後の関連オーディオファイルは、自動的に削除されません。オーファン（孤立）ファイルを削除するには、オーディオキャッシュをクリアが必要があります。このようなオーファンファイルの対処については「[キャッシュクリア](#)」を参照してください。

## Contents Editorで、オブジェクトを削除するには:

1. 削除するオブジェクトのアイコンを選択する。
2. 以下のいずれかを行う:
  - ・Deleteキーを押す。
  - ・このオブジェクトを右クリックして、ショートカットメニューでDeleteを選択する。

選択したオブジェクトまたはソースが、Contents EditorとProject ExplorerのAudioタブから削除される。



### 注記

Contents Editorのオブジェクトを削除すると、プロジェクトからこのオブジェクトが削除される。プレイリストや、Switchリストから、アイテムを1つ削除しても、このオブジェクト自体は削除されません。リストから消されるだけです。

## 関連トピック

- 「Contents Editorのオブジェクトを並べ替える」
- 「Contents Editorのペイン間で、オブジェクトをドラッグする」
- 「リストの展開・縮小」
- 「Contents Editorにオブジェクトを追加する」
- 「Contents Editorのオブジェクトやソースを試聴する」

## Contents Editorのオブジェクトやソースを試聴する

Contents Editorに表示されたオブジェクトやソースは、全て、Transport Controlを使って1つずつ試聴できます。あるサウンドオブジェクトやモーションオブジェクトに、どのソースを使うのかを決める時や、作成したプレイリストを試聴したい時に、便利です。

### Contents Editorのオブジェクトを試聴するには:

1. Transport Controlにオブジェクトを1つロードする。
2. Transport Controlの、Playアイコンをクリックする。

選択したオブジェクトが再生される。

3. Transport Controlの他の機能を使い、再生を一時停止したり、停止したりする。

Transport Controlを使った試聴については[41章Transport Control画面について](#)を参照。

### Contents Editorのソースを試聴するには:

1. Transport Controlに、サウンド、ボイス、モーション、またはミュージックのオブジェクトを、1つロードする。
2. 複数のソースがある場合は、再生したいソースの横の、Useオプションを選択する。
3. Transport Controlの、Playアイコンをクリックする。

選択したソースが再生される。

4. Transport Controlの他の機能を使い、再生を一時停止したり、停止したりする。

Transport Controlを使った試聴については[41章Transport Control画面について](#)を参照。

## 関連トピック

- ・「Contents Editorにオブジェクトを追加する」
- ・「Contents Editorのオブジェクトを並べ替える」
- ・「Contents Editorのペイン間で、オブジェクトをドラッグする」
- ・「リストの展開・縮小」
- ・「オブジェクトを削除する」

---

## 第41章 Transport Control画面について

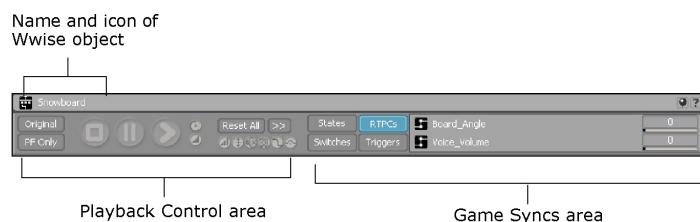
概要 .....	792
再生方法の設定 .....	794
Transport Controlでオブジェクトをピン付けする .....	797
コンテンツの再生、一時停止、停止 .....	798
再生中にゲームシンクを使う .....	798

## 概要

サウンド、ミュージック、モーションの各種プロパティを調整する時に、作業内容を試聴できなければなりません。Wwiseでサウンド、ミュージック、コンテナ、またはイベントオブジェクトを選択すると、自動的にTransport Controlにロードされるのでここで試聴できます。オブジェクトの名前とアイコンが、タイトルバーに表示されます。

Transport Controlは、以下の2つのエリアに分かれています。

- 「再生コントロールのエリア」
- 「ゲームシンクのエリア」

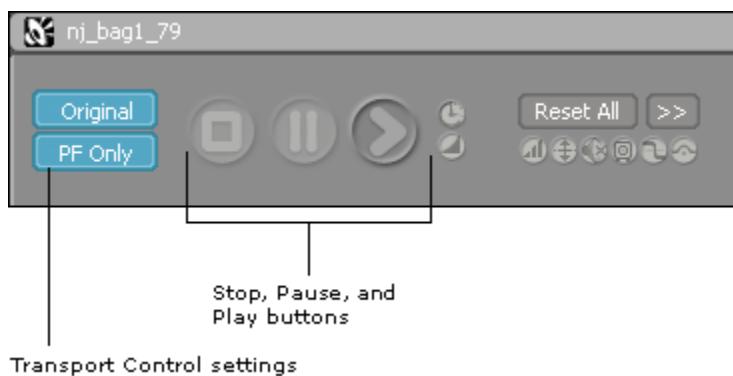


### 注記

ダイアログイベントはTransport Controlにはロードされませんが、引数バスにアサインされているオブジェクトはロードされます。

## 再生コントロールのエリア

Transport Controlの、再生コントロールのエリアに、オーディオ再生に関連する従来のコントロール機能である、再生、停止、一時停止などのボタンがあります。また、Transport Controlの設定機能を使い、オブジェクトの再生方法も変えられます。設定を選択したり選択解除したりすることで、オブジェクトのオリジナルとコンバージョン後のどちらのバージョンを再生するのか、また、現在のプラットフォームから除外されたオブジェクトを再生するのかどうかを指定できます。



### 注記

アクターミキサーとバスは再生できないので、選択されている時も、Transport Controlにはロードされません。

## Transport Control画面について

再生コントロールのエリアには、再生中のオブジェクトに特定のプロパティや動作が既に設定されていると、色が変わるインジケータがいくつかあります。Transport Controlにある、プロパティやアクションパラメータに関するインジケータを、下表に示します。

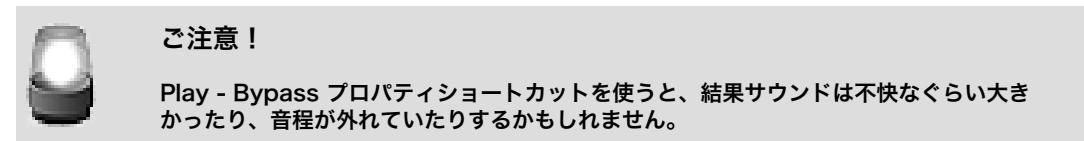
アイコン	名前	Indicates
	Delay	Event、Random Container、またはSequence Containerの中のイベントに対して、ディレイが適用された。
	Fade	Event、Random Container、またはSequence Containerの中の、オブジェクトに対して、フェードが適用された。
	Set Voice Volume	イベントの中のオブジェクトに、ボリューム設定アクションが適用された。
	Set Pitch	イベントの中のオブジェクトに、ピッチ設定アクションが適用された。
	Mute	イベントの中のオブジェクトに、ミュートアクションが適用された。
	Set Low-Pass Filter	イベントの中のオブジェクトに、ローパスフィルタ設定アクションが適用された。
	Enable Bypass	イベントの中のオブジェクトに、バイパス許可アクションが適用された。

これらのオブジェクトプロパティの調整については、以下のセクションを参照してください。

- 「コンテナ内のオブジェクトを、全て再生」
- 「Event Actionのプロパティの設定」

また、設計の目的のためにPlay - Bypass ショートカットが使えることを覚えておいてください。

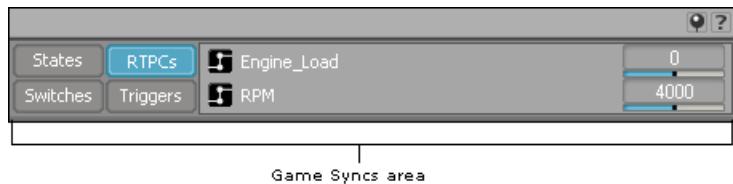
プロパティをバイパスしながら、Wwiseにロードしたオブジェクトを実行するためにPlayをクリックする間、Shiftキーを押したままにしてください。それは、DAWsで見つかるようなPFL (Pre-Fader Listen)のようであり、Wwiseは階層のプロパティ（ボリューム、ピッチ、フィルター、遅延、エフェクト、Auxiliary Sends、Attenuation Curves、RTPC カーブ、States ポジショニング、バスルーティングのようなものを含む）なしでオブジェクトを再生しますが、その動作（フェード、トリム、ループポイント、ランダム・シーケンス再生などのよう）は残ります。



Transport のPlay ボタンをクリックすると、Play Cursor の位置からクリップが再生されます。Play Cursor の実行中にPlay ボタンをクリックすると、最初のPlay Cursorと同じ開始位置から同時に再生される別のPlay Cursorが追加されます。

## ゲームシンクのエリア

従来の再生コントロール機能に加えて、Transport Controlのゲームシンクのエリアに、選択中のオブジェクトに関連する全てのState、Switch、RTPC、Triggerが含まれます。Transport Controlを小規模のシミュレータとして、サウンド、ミュージック、モーションのテストや、ゲーム中の変化のシミュレーションに利用できます。再生中に、StateやSwitchを切り替えたり、Game Parameterにマッピングされた値を試聴したりできます。



ゲームシンクの使い方については、以下のセクションを参照してください。

- [17章Stateの活用](#)
- [「SwitchやStateの中身の管理」](#)
- [「RTPCで使う、Game Parameterの管理」](#)
- [「Triggerの使い方」](#)

## 再生方法の設定

Transport Controlの様々なコントロール機能を有効にして、以下を行えます。

- [「オーディオファイルのオリジナルを再生する」。](#)
- [「再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する」。](#)
- [「Transport Controlをリセットする」](#)

### オーディオファイルのオリジナルを再生する

インポートしたオーディオファイルをコンバージョンする時に、このオーディオファイルのオリジナル (Original) バージョンもWwiseで保存するので、いつでも試聴できます。オリジナルバージョンは、プラットフォーム用のコンバージョンを行っていません。デフォルトとして、Transport Controlではオリジナルサウンドを再生しますが、コンバージョン後のバージョンも再生できます。

#### オリジナルサウンドを再生するには:

1. Transport Controlで、Originalをクリックする。  
ボタンが青くなる。
2. Playアイコンをクリックする。

Transport Controlのオブジェクトのコンバージョン前のオリジナルサウンドが再生される。



#### 注記

コンバージョン後のサウンドを再生するには、Originalボタンをクリックして選択を解除します。

### 関連トピック

- 「再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する」
- 「Transport Controlをリセットする」

### 再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する

オーディオ構造、ミュージック構造、モーション構造を作成する過程で、プラットフォーム（単数または複数）で特定のオブジェクトを含める（Include）・除外する（Exclude）ことを設定できます。プラットフォームに関する設定については「[1つのプラットフォームで、プロジェクトのエレメントを除外する](#)」を参照してください。サウンド、ミュージック、モーションオブジェクトを再生する時に、現在のプラットフォームに含まれるコンテンツだけを再生するのか、または、Transport Controlにロードした全てのサウンド、ミュージック、モーションオブジェクトを再生するのかを、決めます。

### プラットフォーム専用のコンテンツを再生するには:

- ツールバーの、Platform Selectorリストで、試聴するオブジェクトのプラットフォームを選択する。
- Transport ControlでInc. Only.

Inc. Onlyボタンが青くなり、現在のプラットフォームにあるオブジェクトやイベントだけが、Transport Controlで再生される。



#### 注記

全てのオブジェクトやイベントを再生するには、Inc. Only をもう一度クリックして、このオプションを解除します。

### 関連トピック

- 「再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する」
- 「オーディオファイルのオリジナルを再生する」

### Transport Controlをリセットする

Transport Controlでオブジェクトを再生する時に、これらのオブジェクトの様々なプロパティ、動作設定、ゲームシンクをアクセスできるので、ゲーム中のエキスペリエンスをシミュレーションするために活用できます。ゲームに接続した状態では、他のゲームシンク、エフェクト、イベントによって、オブジェクトの既定プロパティが、影響を受けることもあります。Transport Controlのプロパティインジ

ケータが、再生中にまだ有効な動作やアクションを、フィードバックとして提供します。オブジェクトを以前の設定に戻すには、Reset（リセット）機能を使います。



#### 注記

イベント再生インスタンスごとに、Event Actionのプロパティ値が、オブジェクトのプロパティに加算されます。Event Actionを使って作業する場合は、オブジェクトを再生する前に、これらのアクションプロパティをリセットして、累積されたプロパティをクリアしてデフォルトに戻してください。Event Actionが、オブジェクトのプロパティに与える影響については「[イベントの使い方](#)」を参照してください。

### Transport Controlプロパティをリセットするには:

1. Transport Controlで、セレクタアイコンをクリックする。  
Resetメニューが表示される。
2. Resetメニューで、以下のいずれかを選択する。
  - **Reset All**: 全オブジェクトを元の設定に戻す。
  - **Reset All Random and Sequence Containers**: オブジェクト用にトリガーした全てのランダムアクションやシーケンスアクションを解除する。
  - **Reset All Game Parameters**: オブジェクト用にトリガーした全てのGame Parameterを解除する。
  - **Reset All Set Mute**: オブジェクト用にトリガーした全てのミュートアクションを解除する。
  - **Reset All Set Pitch**: オブジェクト用にトリガーした全てのピッチアクションを解除する。
  - **Reset All Set Volume**: オブジェクト用にトリガーしたボリュームアクションを解除する。
  - **Reset All Set Bus Volume**: オブジェクト用にトリガーしたボリュームアクションを解除する。
  - **Reset All Set Low-pass Filter**: オブジェクト用にトリガーした全てのLPFアクションを解除する。
  - **Reset All Bypass Effect**: オブジェクト用にトリガーした全てのエフェクトバイパスアクションを解除する。
  - **Reset All States**: オブジェクトの全てのSet Stateアクションを解除する。
  - **Reset All Switches**: オブジェクトに設定した全てのSet Switchアクションを解除する。
  - **Reset All Music Tracks Force Usage**: Soundcasterを使った特定トラックの強制再生を解除する。
  - **Reset Position**: Attenuation Preview内のリスナーのポジションをデフォルトポジションに戻す。



#### Tip

Resetメニューで選択せずに、全てのオブジェクトをデフォルト設定に戻すには、Reset Allをクリックする。

## 関連トピック

- 「再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する」
- 「オーディオファイルのオリジナルを再生する」

## Transport Controlでオブジェクトをピン付けする

Property Editorに現在あるオブジェクトが、自動的に、Transport Controlにロードされます。Project Explorerで他のオブジェクトやイベントを選択すると、デフォルトで、Transport Controlでも置き換えられます。次に何を選択するかに関わらず、Transport Controlにロードしたオブジェクトを、そのまま維持したい時は、このオブジェクトをピン付けする (Pin) と、他のオブジェクトがロードされるのを防ぎます。



### ピン付け機能の活用例

例えば、Random Containerの試聴中に、このコンテナの中のあるオブジェクトが、リバーブエフェクトを受ける時のサウンドが、気になるとします。その場合はまず、Transport Controlでこのオブジェクトをピン付けします。次に、Property Editorで親オブジェクトにリバーブを適用して、エフェクトの結果を、このオブジェクトだけで試聴します。もしこのオブジェクトをピン付け (Pin) していないと、コンテナの全てのオブジェクトがランダムモードで再生されるので、試聴したいオブジェクトが再生されないかもしれません。あるオブジェクトをピン付けすれば、親の設定を変えてから、このオブジェクトだけを、好きなだけ、試聴できます。

### Transport Controlにロードされたオブジェクトをピン付けするには:

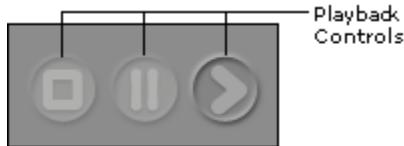
- Transport Controlにオブジェクトを1つロードする。
- Transport ControlでPinをクリックする。

Pinアイコンが赤くなり、Property Editorに他のオブジェクトやイベントをロードしても、Project Explorerで他のオブジェクトを選択しても、このオブジェクトまたはイベントが、Transport Controlに残る。



## コンテンツの再生、一時停止、停止

オブジェクトを試聴するには、Transport Controlの標準的な再生機能を使います。



また、Transport Controlビューのメインコントロールボタンに設定されたショートカットキーを使えば、様々なプロパティ値をリアルタイムで編集しながら、素早く再生を開始、一時停止、停止できます。全てのキーボードショートカットを見るには[付録C ショートカット](#)を参照してください。

**Transport Controlで、サウンドを再生するには:**

1. Transport Controlにオブジェクトを1つロードする。
2. 以下のいずれかを行う。
  - Playアイコンをクリックする。
  - 対象のコンテンツが、最後まで再生される。
  - 再生を一時停止するには、Pauseアイコンをクリックする。
  - Pauseアイコンが黄色になる。再生を再開するには、Pauseをもう一度クリックする。
  - 再生を停止するには、Stopアイコンをクリックする。

### 関連トピック

- 「[オーディオファイルのオリジナルを再生する](#)」
- 「[再生時に、一部のオーディオコンテンツやモーションコンテンツを含める・除外する](#)」
- 「[Transport Controlでオブジェクトをピン付けする](#)」
- 「[再生中にゲームシンクを使う](#)」

## 再生中にゲームシンクを使う

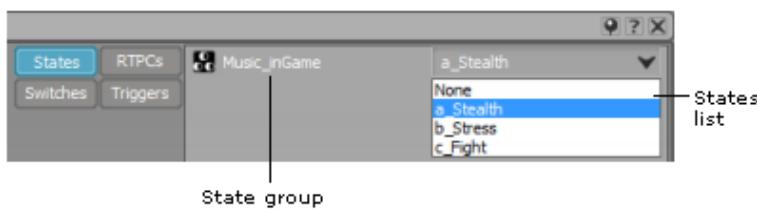
Transport Controlから、プロジェクトで作成したゲームシンクをアクセスできます。ステート、スイッチ、RTPCが、サウンド、ミュージック、モーションオブジェクトに適用される様子を試聴する方法を、以下に示します。

- 「[再生中に、Stateを有効にする](#)」
- 「[再生中に、Switchをアサインする](#)」
- 「[再生中に、Game Parameter値を変更する](#)」
- 「[再生中に、Triggerを呼び出す](#)」

## 再生中に、Stateを有効にする

Transport Controlにオブジェクトをロードすると、このオブジェクトが登録しているState GroupやStateのリストが表示されるので、これらを選択して、再生中に、ゲーム中のStateの発生や変化をシミュレーションできます。つまり、オブジェクトを再生しながら、Stateに設定したプロパティを試聴できるほか、別のStateに切り替えてStateが変化する様子も試聴できます。Stateを作成したり、Transport Controlで視聴できるStateのプロパティやトランジションを設定したり、Stateにオブジェクトをアサインする方法については、以下のセクションを参照してください：

- ・「Stateの活用」
- ・「オブジェクトやバスにStateをアサインする」



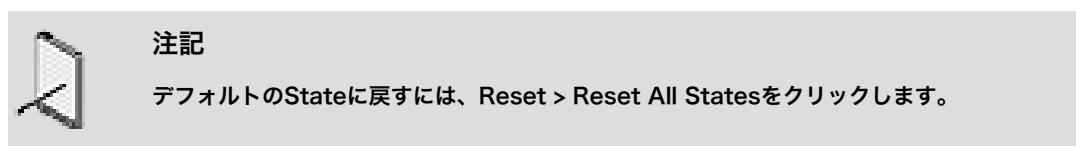
### 再生中にStateを有効にするには：

1. Transport Controlにオブジェクトを1つロードする。
2. Stateリストで、適用するStateを1つ選択する。

このStateに登録しているTransport Controlの全てのサウンドに、このStateが、再生中に適用される。

3. Playアイコンをクリックする。

再生中に、選択したStateが適用される。オブジェクトの再生中に、引き続きStateを切り替えて、ゲームをシミュレーションできる。



### 関連トピック

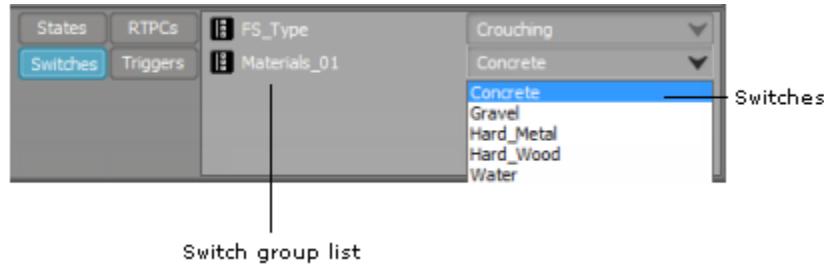
- ・「再生中に、Switchをアサインする」
- ・「再生中に、Game Parameter値を変更する」

## 再生中に、Switchをアサインする

Transport Controlにオブジェクトをロードすると、このオブジェクトがアサインされているSwitch GroupやSwitchのリストが表示されるので、これらを選択して、再生中に、ゲーム中のSwitchの切り替わりをシミュレーションできます。つまり、オブジェクトを再生しながら、Switchを切り替えて、変化する様子を試聴

できます。Switchの作成と使い方については、以下のセクションを参照してください。

- ・「[Switchの使い方](#)」
- ・「[Switch Containerの種類の設定](#)」



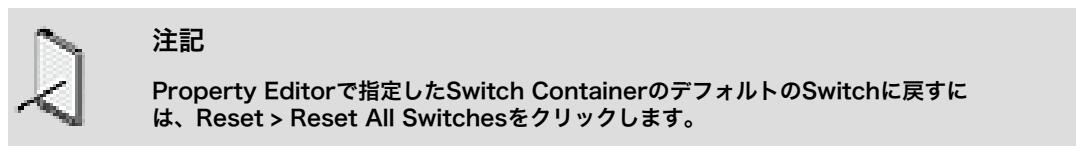
### 再生中にSwitchをアサインするには:

1. Transport Controlにオブジェクトを1つロードする。
2. Switchリストで、適用するSwitchを1つ選択する。

選択中のSwitch Groupに登録されたSwitch Containerが、選択したSwitchに対応するオブジェクトを再生する。

3. Playアイコンをクリックする。

再生中に、選択したSwitchが適用される。オブジェクトの再生中に、引き続き Switchを切り替えて、ゲームをシミュレーションできる。



### 関連トピック

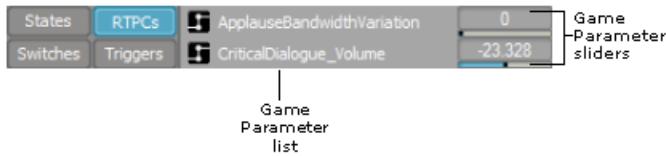
- ・「[再生中に、Stateを有効にする](#)」
- ・「[再生中に、Game Parameter値を変更する](#)」

### 再生中に、Game Parameter値を変更する

Transport Controlにオブジェクトをロードすると、関連するRTPCが、ゲームシンクのエリアに表示されます。スライダを使って、オブジェクトの再生中にGame Parameterを変えられます。既にGame Parameter値をWwiseのプロパティ値にマッピングしてあるので、Game Parameter値を変えると、自動的にオブジェクトのプロパティ値も変わります。このようにして、Game Parameterがゲーム中に変化した時の様子をシミュレーションして、設定したプロパティマッピングがゲーム上で効率的に作用するかどうかを、検証できます。Game Parameterを作成してプロパティ値をマッピングする方法については、以下のセクションを参照してください。

- ・「[RTPCで使う、Game Parameterの管理](#)」

- 「Game Parameterに対して、Wwiseプロパティを設定する」



プロパティが変化する様子を、シミュレーションの再生中に試聴できます。

### 再生中にGame Parameter値を変更するには:

- Transport Controlにオブジェクトを1つロードする。
- ゲームシンクのエリアで、RTPCsをクリックする。

このオブジェクトにマッピングされているGame Parameterが全て表示される。

- Playアイコンをクリックする。

オブジェクトの再生中に、RTPCスライダを使ってGame Parameter値を変えながら、変化に反応するサウンドの様子を確認する。



#### 注記

Game Parameterをデフォルト設定に戻すには、Reset > Reset All Game Parametersをクリックします。



#### 注記

Game Parameterの線形補間をバイパスするには、Ctrlを押しながらTransport ControlのRTPCスライダを使うか、Property EditorのRTPCタブのGame Parameterカーソルを使います。

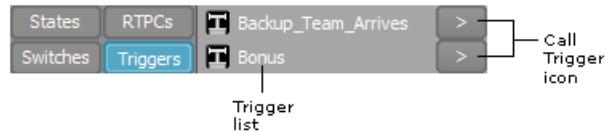
### 関連トピック

- 「再生中に、Switchをアサインする」
- 「再生中に、Stateを有効にする」

### 再生中に、Triggerを呼び出す

オブジェクト自体を試聴するほかに、Triggerをリストから選択して、ステインガーを試聴できます。ステインガーとは、再生中の音楽に重ねてミックスされる、短いミュージックフレーズのことです。ゲーム中の重要ポイントでTriggerがステインガーを呼び出して、再生中のミュージックに重ねて再生する様子を、シミュレーションできます。Triggerを作成して、それぞれのステインガーを作成する方法については、以下のセクションを参照してください。

- 「Triggerの使い方」
- 「29章ステインガーの活用」



### 再生中にTriggerを呼び出すには:

1. Transport Controlにミュージックオブジェクトを1つロードする。
2. Playアイコンをクリックする。  
Transport Controlにロードされたミュージックオブジェクトが再生される。
3. ゲームシンクのエリアで、Triggersボタンをクリックして、Triggerリストを表示させる。



4. Call Triggerアイコンをクリックする。

該当するスティンガーが、再生中のミュージックオブジェクトに重ねて再生される。他のTriggerも選択して、それぞれのスティンガーを再生させて、ゲーム中のミュージックをシミュレーションできる。

### 関連トピック

- 「コンテンツの再生、一時停止、停止」
- 「Triggerの使い方」
- 29章スティンガーの活用

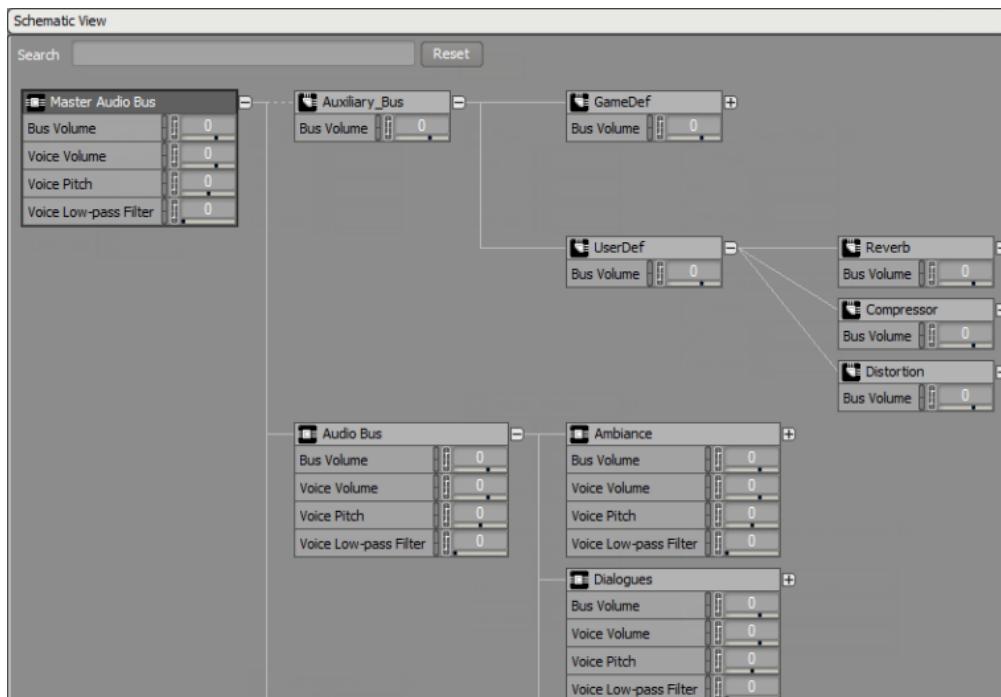
---

## 第42章 Schematic View画面について

概要 .....	804
Schematic Viewのカスタマイズ .....	804
Schematic Viewを活用する .....	806

## 概要

Schematic View（スケマティックビュー）に、Wwiseプロジェクトの構造が図式化して表示されます。Schematic Viewは、プロジェクトの全体像を把握する時や、プロジェクトのオブジェクトを探している時や、オブジェクトごとにプロジェクト構造を分析したい時に、使用します。Schematic Viewに表示されるのは、各プロジェクトオブジェクトを表すアイコンとオブジェクト名のほか、オブジェクトの相互関係を示すラインやノードです。また、Schematic Viewをカスタマイズすれば、プロジェクトオブジェクトの詳細情報の表示方法を変えられます。



## Schematic Viewのカスタマイズ

Schematic Viewでは、プロジェクトオブジェクトが図式化され、ビューをカスタマイズするための特別なツールがあります。また、オブジェクトを素早く見つけるための検索機能もあります。

Schematic Viewをカスタマイズするためには、以下を行います。

- 「プロジェクトオブジェクトやコネクタの表示方法」
- 「Schematic Viewの表示オプションの設定」

### プロジェクトオブジェクトやコネクタの表示方法

Schematic Viewでは、プロジェクトオブジェクトが各種アイコンで示され、相互関係がコネクタで示されます。プロジェクトオブジェクト同士をつなぐコネクタの意味を、下表に示します。プロジェクトオブジェクトを表すアイコンについては「[Wwiseのアイコン](#)」を参照してください。

アイコン	名前	内容
	Solid line	親と子オブジェクトをつなぐ線。
	Dashed line	バスと子オブジェクトをつなぎ、ルーティングを示す線。
	Plus sign (white)	クリックしてスキーマを展開して、オブジェクトの子を全て表示させるアイコン。
	Plus sign (yellow)	オブジェクトの子が全て表示されていない時にクリックして、全ての子を表示させるアイコン。
	Minus sign	クリックして縮小して、オブジェクトの子を全て隠すアイコン。

## Schematic Viewの表示オプションの設定

Schematic View Settingsを使えば、スキーマに表示されるプロジェクトオブジェクトの情報を、カスタマイズできます。

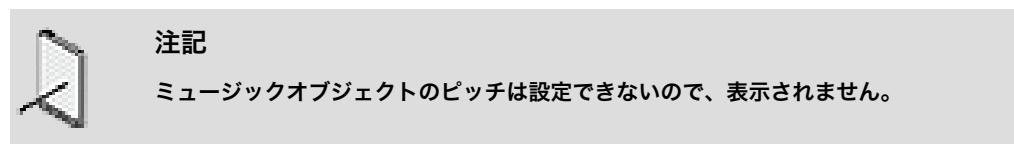
### Schematic Viewの表示内容を設定するには:

1. Schematic Viewの右上にある、オプションを示すアイコンをクリックする。

Schematic View Settingsダイアログボックスが開く。

2. 情報タイプを、以下から1つ以上選択する。

- **Icon Strip:** オブジェクトのプロパティを表すアイコンの列を表示する場合。オブジェクトプロパティに変更があれば、アイコンが白くなり、上にマウスを配置すると、詳細情報が表示される。プロパティのデフォルト値が設定されたままであれば、グレイアウト（灰色表示）表示となる。
- **Mute/Solo:** 各オブジェクトのミュートボタンとソロボタンを表示する場合。
- **Bus:** オブジェクトをルーティングするバスを表示する場合。
- **Conversion Settings (コンバージョン設定):** オブジェクトが使用するコンバージョン設定ShareSetを表示する場合。
- **Effect:** オブジェクトに適用されるエフェクトを表示する場合。
- **Positioning Type:** オブジェクトに適用されるポジショニングタイプ（2D、3D User-defined、3D Game-defined）を表示する場合。
- **Game Parameters:** オブジェクトに（RTPCを経由して）影響するGame Parameterを表示する場合。
- **State Group:** オブジェクトが登録されているStateやState Groupを表示する場合。
- **Advanced Settings:** オブジェクトの詳細設定の変更があったときに、それを表示する場合（再生リミットやボリューム閾値など）。
- **Volume:** オブジェクトに設定したボリュームを表示する場合。
- **Pitch:** オブジェクトに設定したピッチを表示する場合。



- **Low Pass:** オブジェクトに設定したLPF（ローパスフィルタ）を表示する場合。

### 3. OKをクリックする。

選択した情報内容が、プロジェクトオブジェクトごとに、Schematic Viewで表示される。

## 関連トピック

- ・「プロジェクトオブジェクトやコネクタの表示方法」
- ・「プロジェクトオブジェクトを検索する」
- ・「1つのプロジェクトオブジェクトを表示させる」
- ・「プロジェクトオブジェクトを編集する」

## Schematic Viewを活用する

Schematic Viewに表示される様々なプロジェクトオブジェクトを対象に、検索、分析、作業するための様々なオプションがあります。

Schematic Viewで作業をする時に、以下の機能を利用できます。

- ・「プロジェクトオブジェクトを検索する」
- ・「1つのプロジェクトオブジェクトを表示させる」
- ・「プロジェクトオブジェクトを編集する」

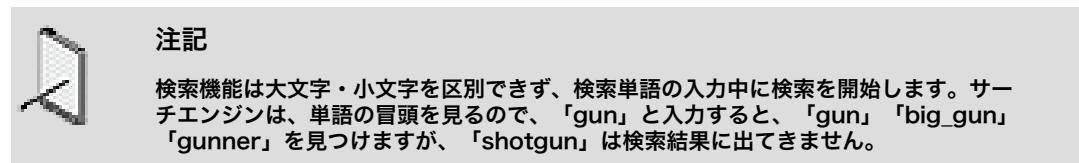
以下のセクションに出てくるコマンドの多くは、ショートカットキーが設定されています。全てのショートカットを見るには[付録C ショートカット](#)を参照してください。

### プロジェクトオブジェクトを検索する

Schematic Viewの、検索 (Search) 機能で、プロジェクトオブジェクトを素早く検索できます。

### Schematic Viewでプロジェクトオブジェクトを検索するには:

1. Searchフィールドで、探しているプロジェクトオブジェクトの名前を入力する。  
入力した名前に合うプロジェクトオブジェクトを、Wwiseがハイライト表示します。
2. Searchフィールドをクリアして、Schematic Viewをリセットするには、Resetをクリックする。



## 関連トピック

- ・「Schematic Viewの表示オプションの設定」

- ・「プロジェクトオブジェクトやコネクタの表示方法」
- ・「1つのプロジェクトオブジェクトを表示させる」
- ・「プロジェクトオブジェクトを編集する」

### 1つのプロジェクトオブジェクトを表示させる

Schematic Viewで1つのプロジェクトオブジェクトをハイライト表示させることで、表示されるプロジェクトスキーマをシンプルにできます。

#### Schematic Viewでプロジェクトオブジェクトをハイライト表示するには:

1. Schematic Viewで、プロジェクトオブジェクトを1つ右クリックする。  
ショートカットメニューが表示される。
2. ショートカットメニューで、Show in Schematic Viewを選択する。  
選択したオブジェクトのバスのルーティングが表示される。このオブジェクトがハイライト表示され、無関係のオブジェクトが非表示となる。
3. Schematic Viewをリセットするには、Clearをクリックする。



### 関連トピック

- ・「プロジェクトオブジェクトやコネクタの表示方法」
- ・「Schematic Viewの表示オプションの設定」
- ・「プロジェクトオブジェクトを検索する」
- ・「プロジェクトオブジェクトを編集する」

### プロジェクトオブジェクトを編集する

Schematic Viewに表示されるプロジェクトオブジェクトを編集するには、以下の2通りがあります。

- ・「プロジェクトオブジェクトを直接、編集する」
- ・「Property Editorでプロジェクトオブジェクトを編集する」

### プロジェクトオブジェクトを直接、編集する

Schematic Viewの各プロジェクトオブジェクトの下に表示されるコントロール機能を使って、オブジェクトを編集できます。設定方法は、Property Editorと全く同じです。コントロール機能の使い方については「[Wwiseのインターフェースの基礎](#)」を参照してください。



Schematic View Settingsを使って、コントロール機能を表示させます。コントロール機能の設定方法については「[Schematic Viewの表示オプションの設定](#)」を参照してください。

### Property Editorでプロジェクトオブジェクトを編集する

オブジェクトプロパティを直接編集したい場合は、そのプロジェクトオブジェクトのProperty Editorを、Schematic Viewから簡単に開けます。

### あるプロジェクトオブジェクトのProperty Editorを開くには:

1. Schematic Viewで、プロジェクトオブジェクトを1つ右クリックする。

ショートカットメニューが表示される。

2. ショートカットメニューで、Editを選択する。

このプロジェクトオブジェクトのProperty Editorが開く。



### 関連トピック

- 「プロジェクトオブジェクトやコネクタの表示方法」
- 「Schematic Viewの表示オプションの設定」
- 「プロジェクトオブジェクトを検索する」
- 「1つのプロジェクトオブジェクトを表示させる」

---

## 第43章 グラフビューについて

概要 .....	810
グラフビュー表示設定を変更する .....	811
グラフビューのコントロールポイントの使い方 .....	816
グラフビューのカーブの使い方 .....	818

## 概要

Wwiseには、グラフが表示されるビューがいくつかあります。グラフで、2つの変数の関係の設定、空間的ポジショニングの設定、パフォーマンスのモニタリングなどができます。

Wwiseには、数種類のグラフがあります。使える機能は基本的に同じですが、いくつかの違いがあります。グラフが組み込まれているビューを、以下に示します。

- Project Settings (Obstruction/Occlusion tab)
- Property Editor (RTPC tab)
- Blend Tracks Editor
- Attenuation Editor
- Position Editor
- Switch Group Property Editor
- Music Fade Editor
- Performance Monitor
- Game Sync Monitor
- Wwise Motion Generator Property Editor
- Wwise Reflect - エフェクトエディタ
- SoundSeed Air Source Editor

各グラフの目的を、下表に示します。

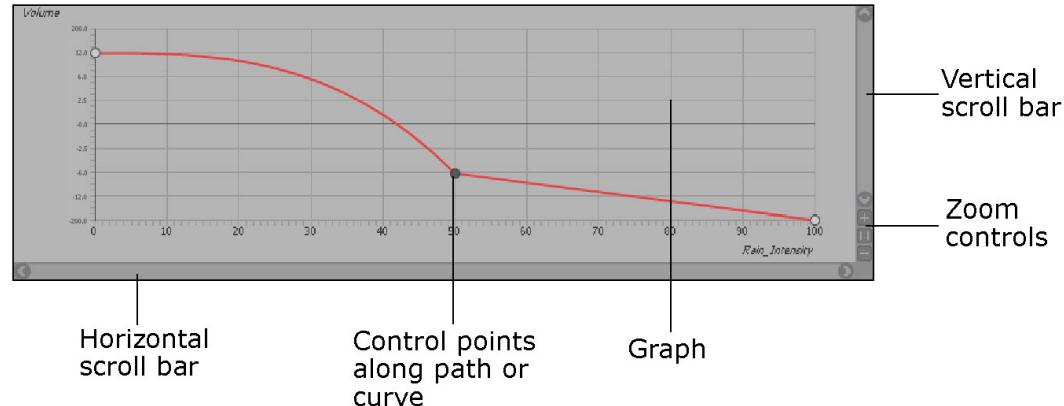
Use this graph	To
Project Settings (Obstruction/ Occlusion tab)	サウンドオブジェクトのボリュームやLPFと、オブストラクション やオクルージョンの要因との関係を示すグラフ。
Property Editor (RTPC tab)	Game Parameter値と、Wwiseのプロパティ値の関係を示すグラフ。
Blend Tracks Editor	オブジェクトを複数のブレンドトラックに分けて、RTPCカーブを適用したり、クロスフェードを適用したり、両方を適用したりするためのグラフ。
Attenuation Editor	特定のプロパティと、リスナーからエミッターソースまでの距離との関係を定義して、現実的な減衰、つまりサウンドの衰退を作成するグラフ。
Position Editor	3D環境内のパスを設定して、サラウンドスピーカー環境の中にサウンドを配置するグラフ。
Switch Group Property Editor	Switch Groupの様々なSwitchを、具体的なGame Parameterにマッピングするグラフ。
Music Fade Editor	ミュージックトランジションで、あるミュージックピースから別のものに変わる時の、フェードイン・フェードアウトのカーブを設定するグラフ。
Performance Monitor	ゲーム、シミュレータ、またはサウンドキャスターによってゲームエレメントがトリガーされる時の、サウンドエンジンのパフォーマンス情報を表示するグラフ。  Performance Monitorの情報は読み込み専用で、分析目的の利用に限定。このため、コントロールポイントの追加・削除や、カーブの調整ができない。

Use this graph	To
Game Sync Monitor	ウォッチ中のゲームオブジェクトに適用されるRTCP値をトラッキングするグラフ。 Game Sync Monitorの情報は読み込み専用で、分析目的の利用に限定。このため、コントロールポイントの追加・削除や、カーブの調整ができない。
Wwise Motion Generator Property Editor	ゲーム機のモーターで発生させるモーションの強度を、プラットフォーム別に定義する複数のカーブを作成する。
Wwise Reflect - エフェクトエディタ	距離に基づいた5種類のカーブを編集して、減衰によるボリューム変更、スプレッド率、そして音をカットオフする周波数の上限と下限を定義する。
SoundSeed Air - Woosh Source Plugin Editor (Object Path Graph)	Woosh使用シーンで、オブジェクトがたどる軌跡を定義するパスと、サウンドが遠ざかる時の減衰も定義するパスを作成する。
SoundSeed Air (Wind/Woosh) Source Plugin Editor	生成されるWindサウンドやWooshサウンドのプロパティを定義する、複数の自動化カーブを作成する。
Voice Monitor	一定の時間の間に再生されるWwiseオブジェクトのボイスのボリュームをカーブで確認する。

サウンドパスの作成、パフォーマンスのモニタリング、セグメントを使った作業などの際は、タイムフレーム（時間枠）も定義する必要があります。この場合、グラフビューに、タイムライン（時間軸）が関連付けられます。Wwiseでタイムラインを使った作業については「[概要](#)」を参照してください。

### グラフの共通エレメントの説明

Wwiseの様々なグラフビューは、それぞれ少しずつ違いますが、ズームコントロール、スクロールバー、カーブやパス（軌跡）を形成するコントロールポイントなど、いくつかの共通エレメントがあります。グラフビューにある共通エレメントを、下図に示します。



## グラフビュー表示設定を変更する

グラフビューの表示設定を変えて、コントロールポイントの位置を詳細に決めたり、特定エリアを詳しく調べたりできます。グラフビューの表示を変更するには、以下のツールを使います：

- 「[グラフビューのズームとパン](#)」
- 「[グラフビューの単位の設定](#)」

- ・「グリッドラインの表示・非表示」
- ・「カーソルの表示・非表示」

## グラフビューのズームとパン

パンやズームのツールを使い、パス（軌跡）やカーブ上のポイントの細かい配置や、特定エリアの情報の詳細な調査ができます。ズームイン・ズームアウトや、上下左右へのパンが可能です。

全てのパンやズームの機能に、キーボードショートカットが設定されています。

To	ショートカット
ズームイン	Z+[マーキー選択]
現在のマウス位置を中心に、ズームイン	Ctrl+[マウスホイールを上に]
現在のマウス位置を中心に、ズームアウト	Ctrl+[マウスホイールを下に]
ズームイン中に、ズームをリセット	Z+[クリック]
ズームイン中に、上にパンニング	マウスホイールを上に
ズームイン中に、下にパンニング	マウスホイールを下に
ズームイン中に、左にパンニング	Shift+[マウスホイールを上に]
ズームイン中に、右にパンニング	Shift+[マウスホイールを下に]
ズームイン中に、フリーハンドでパンニング	X+[ドラッグ]

全てのショートカットを見るには[付録C ショートカット](#)を参照してください。

### グラフビューで、ズームインするには:

1. グラフビューのツールバーで、Zoom Inアイコンをクリックする。

グラフビューの中心に向かって、グラフがズームインする。

### グラフビューで、ズームアウトするには:

1. グラフビューのツールバーで、Zoom Outアイコンをクリックする。

グラフビューの中心から、グラフがズームアウトする。

### グラフビューで、パンとズームをリセットするには:

1. グラフビューのツールバーで、Resetアイコンをクリックする。

設定されているポイントが全て表示されるように、グラフビューがリセットされる。

### グラフビューで、パンするには:

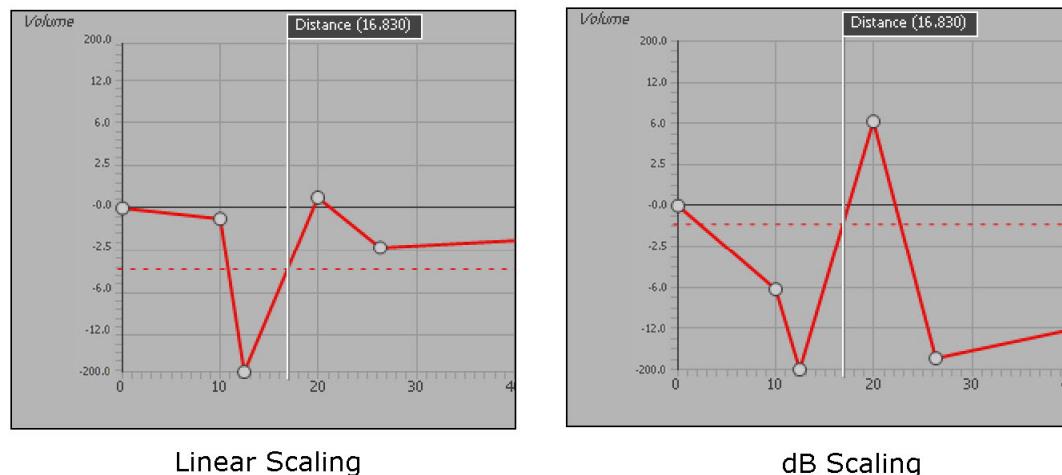
1. グラフビューで、ズームインする。
2. 以下のいずれかを行う:
  - ・上にパンするには、マウスホイールを上に動かす。

- ・下にパンするには、マウスホイールを下に動かす。
- ・右にパンするには、Shiftを押しながらマウスホイールを下に動かす。
- ・左にパンするには、Shiftを押しながらマウスホイールを上に動かす。
- ・フリーハンドでパンするには、xを押して、グラフィュー内でマウスをドラッグする。

### グラフィューの単位の設定

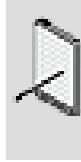
デシベル単位で設定されたカーブを表示する時に、厳密な「Linear scaling（リニアスケーリング / 均等目盛り）」のグラフで表示するのか、より普通（つまり、対数 / Logarithmic）のサウンドの減少に従った「dB Scaling（デシベルスケーリング / db目盛り）」で表示するかを決めます。「Linear Scaling」のグラフでは、Y軸の単位が均等に割り付けられています。「dB Scaling」の場合は、人間の聴覚がサウンドをどう解釈するかを、デシベル数によってY軸で再現します。つまり、ゼロに近いほど目盛り間のギャップが大きくなり、デシベル数が低くなるほど間隔が狭くなります。

同じボリュームカーブを「Linear scaling」と「dB Scaling」で表示した2つのグラフを、以下に示します。



「dB Scaling」はY軸上のデシベル数の目盛りに影響するので、グラフィュー内のコントロールポイントがどう動くかにも、影響します。例えば、「dB Scaling」グラフの「-5dB」の変化によって、「0dB」にあるポイントの方が、「-80dB」にあるポイントよりも、大きく動きます。一方、「Linear Scaling」グラフでは、「-5dB」の変化は、Y軸上で常に同じ変化となります。

多くの場合、「dB Scaling」の方が、あるGame Parameter値においてサウンドがどのように聞こえるかを、正確に表します。ただし、Game Parameterをそのまま、Wwiseプロパティに反映させて関係を設定する場合は、「Linear Scaling」を使います。例えば、バス「Voices」のボリュームを、ゲームの中に存在するボリュームスライダにマッピングして、ゲームプレイヤーがこれを上下してボイスのボリュームを増減できるとします。この場合は、サウンドのボリュームとゲームのボリュームを、直接マッピングする必要があります。X軸のボリュームが均等の目盛り（Linear）なので、Y軸も均等の目盛り（Linear）にする必要があります。



### 注記

目盛り (Scaling) の方式によって、Y軸の目盛りの数値が変わるので、同時に複数のカーブをグラフに表示する時は、全てのカーブの目盛り方が同じでなければ、Y軸の目盛りの数値が表示されません。

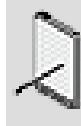
## グラフビューの、目盛り方式を設定するには:

1. グラフビューで、カーブを1つ右クリックする。  
ショートカットメニューが表示される。
2. 目盛り (Scaling) 方式を1つ、以下から選択する。
  - **dB scaling** (デシベル目盛り) : 人間の聴覚によるサウンドの解釈に合わせて、サウンドの普通 (つまり、対数 / Logarithmic) の減少に従い、デシベルでカーブを表示させる場合。
  - **Linear scaling** (均等目盛り) : 厳密な均等方式で、カーブを表示させる場合。

選択した方式の目盛りに従って、グラフビューにカーブが表示される。

## グリッドラインの表示・非表示

グラフビューによっては、グリッドラインを表示させて、コントロールポイントをより正確に配置できます。グリッドラインの表示・非表示の設定は、グラフ中のポイント自体に影響しません。



### 注記

デフォルトとして、グリッドラインが表示されます。

## グラフビューで、グリッドラインを非表示にするには:

1. グラフビュー内を右クリックする。  
ショートカットメニューが表示される。
2. 以下のいずれかを選択する。
  - **Display Grid (Vertical)** (縦グリッド表示) : 縦グリッドを非表示にする場合。
  - **Display Grid (Horizontal)** (横グリッド表示) : 横グリッドを非表示にする場合。
3. 上記1~2を繰り返して、残りのグリッドラインを非表示にする。

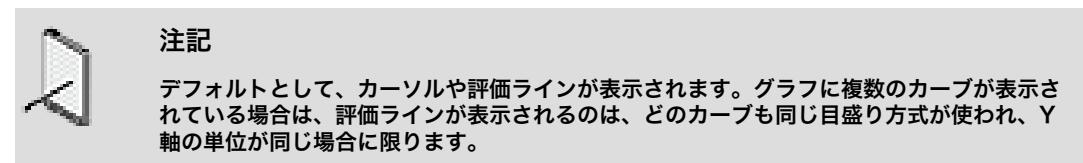
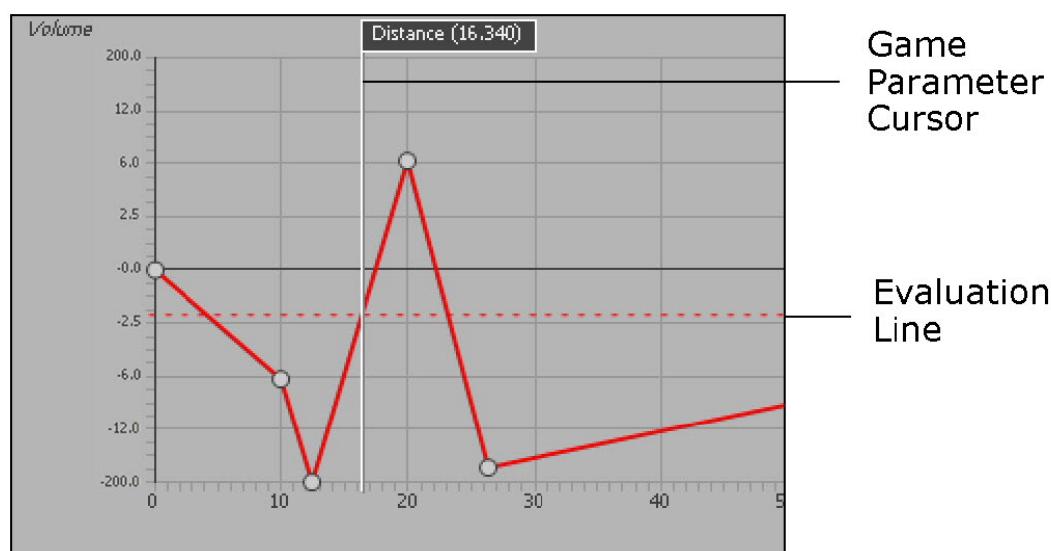
## グラフビューで、グリッドラインを表示するには:

1. グラフビュー内を右クリックする。  
ショートカットメニューが表示される。
2. 以下のいずれかを選択する。
  - **Display Grid (Vertical)** (縦グリッド表示) : 縦グリッドを表示する場合。

- Display Grid (Horizontal) (横グリッド表示) : 横グリッドを表示する場合。
3. 上記1~2を繰り返して、残りのグリッドラインを表示させる。

### カーソルの表示・非表示

RTPCのグラフィビューでは、表示されているカーブごとに、Game Parameterカーソルと、評定ラインが表示されます。Game Parameterカーソルの上部に添付されるタグには、設定されているGame Parameterの名前と、グラフ上の現在の数値が表示されます。再生中にこのカーソルを前後にドラッグすれば、Game Parameter値が変化する様子をシミュレーションできます。評定ラインは、特定のGame Parameter値に対応するプロパティ値を決める参考として表示されます。これらのカーソルの表示・非表示の設定は、グラフ中のポイント自体に影響しません。



### グラフィビューで、カーソルを非表示にするには:

1. グラフィビュー内を右クリックする。  
ショートカットメニューが表示される。
2. Show cursors (カーソル表示) を選択する。

このグラフィビューで、カーソルが非表示になる。

### グラフィビューで、カーソルを表示するには:

1. グラフィビュー内を右クリックする。  
ショートカットメニューが表示される。

## 2. Show cursors (カーソル表示) を選択する。

このグラフビューで、カーソルが表示されます。

# グラフビューのコントロールポイントの使い方

グラフビューのコントロールポイントを使って、カーブやパス（軌跡）の形状を変え、2つの変数値の関係を具体的に設定します。カーブ上のポイントを、いつでも追加、移動、削除できます。また、複数のポイントを同時選択して、カーブの一部分、または複数のカーブを対象に、一括変更や削除も可能です。

コントロールポイントに関するあらゆる情報を、以下のセクションで説明します。

- 「コントロールポイントの追加」
- 「コントロールポイントの選択」
- 「コントロールポイントの移動」
- 「コントロールポイントの削除」

## コントロールポイントの追加

カーブ上のどこでも、ポイントを追加してカーブの形状を変えたり、複数の変数値の関係を具体的に設定したりできます。それぞれのグラフビューで設定する情報タイプが異なるので、コントロールポイントの追加方法も、多少違います。様々なグラフビューにおいて、ポイントを追加する方法を、下表に示します。

Graph	ポイントを追加する方法	2点の間にポイントを挿入する方法
Project Settings (Obstruction/Occlusion tab)	該当なし	カーブ上の2点の間をダブルクリック
RTPC/Blend Tracks Editor	該当なし	カーブ上の2点の間をダブルクリック
Position Editor	ダブルクリック	カーブ上の2点の間を、Ctrlを押しながらダブルクリック
Attenuation Editor	該当なし	カーブ上の2点の間をダブルクリック
Switch Group Property Editor	該当なし	カーブ上の2点の間をダブルクリック
Music Fade Editor	N/A	N/A
Performance Monitor	N/A	N/A
Game Sync Monitor	N/A	N/A
Wwise Motion Generator Source Plugin Editor	該当なし	カーブ上の2点の間をダブルクリック
SoundSeed Air - Woosh Source Plugin Editor (Object Path Graph)	ダブルクリック	カーブ上の2点の間を、Ctrlを押しながらダブルクリック
SoundSeed Air (Wind/Woosh) Source Plugin Editor	該当なし	カーブ上の2点の間をダブルクリック

## コントロールポイントの選択

グラフビューでコントロールポイントを移動または削除する前に、それを選択します。一度に選択できるコントロールポイントは、1つ、または複数、または全て

です。グラフビューに同時に複数のカーブが表示されている場合は、複数のカーブで、複数のポイントを選択して、移動または削除できます。

### グラフビューで、コントロールポイントを選択するには:

1. グラフビューで、コントロールポイントを1つクリックして選択する。

選択されたコントロールポイントが黒色になる。

### グラフビューで、複数のコントロールポイントを選択するには:

1. グラフビューで、以下のいずれかを行う。

- ・選択したい複数のポイントの上で、マーキー選択をドラッグする。
- ・Ctrlを押しながら、選択したい複数ポイントをクリックする。

選択されたポイントが黒色になる。

### グラフビューで、全てのコントロールポイントを選択するには:

1. グラフビュー内をクリックして、ビューを有効にする。
2. Ctrl+Aを押す。

このグラフビュー内の全てのポイントが選択され、黒色になる。

### コントロールポイントの移動

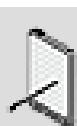
グラフビューに表示されるカーブ（単数または複数）の、1つ以上のコントロールポイントを、グラフビュー内であればどこにでも移動でき、カーブ形状を変更できます。ただし、グラフビューを完全にズームアウトさせた状態では、グラフビューの外枠を出ることはできません。

### グラフビューで、コントロールポイントを移動するには:

1. グラフビューで、コントロールポイントを1つまたは複数を選択する。

選択されたコントロールポイントが黒色になる。

2. これらのポイントを、グラフビューの枠内で、ドラッグする。



#### 注記

矢印キーを使って、またはXY座標 (XY Coordinates) フィールドに数値を直接入力して、ポイントを移動することもできます。複数のポイントが選択されている場合は、XY フィールドに入力した数値によって、全てのポイントがそれぞれの位置からオフセットされます。

### コントロールポイントの削除

カーブやパス（軌跡）のポイントを消したい場合は、削除できます。ただし、削除できないポイントもあります。各種グラフビューで削除できるポイントを、下表に示します。

Graph	削除できるポイント
Project Settings (Obstruction/Occlusion tab)	最初と最後のポイント以外の、全てのポイント。
RTPC/Blend Tracks Editor	最初と最後のポイント以外の、全てのポイント。
Attenuation Editor	最初と最後のポイント以外の、全てのポイント。
Position Editor	最初のポイント以外の、全てのポイント。
Switch Group Property Editor	最初と最後のポイント以外の、全てのポイント。
Music Fade Editor	該当なし - Music Fade Editorにはコントロールポイントがない。
Performance Monitor	該当なし - Performance Monitorにはコントロールポイントがない。
Game Sync Monitor	該当なし - Game Sync Monitorにはコントロールポイントがない。
Wwise Motion Generator Source Plugin Editor	最初と最後のポイント以外の、全てのポイント。
SoundSeed Air - Woosh Source Plugin Editor (Object Path Graph)	最初のポイント以外の、全てのポイント。
SoundSeed Air (Wind/Woosh) Source Plugin Editor	最初と最後のポイント以外の、全てのポイント。

### カーブのコントロールポイントを削除するには:

1. カーブまたはパスのポイント（単数または複数）を選択する。

選択されたコントロールポイントが黒色になる。

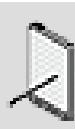
2. Deleteキーを押す。

選択したコントロールポイントがカーブから削除される。

## グラフビューのカーブの使い方

1つのグラフビューの中で、同時に複数のカーブで作業することもあるので、以下の基本的なタスクの習得が重要です。

- ・「グラフビューで、カーブを表示する」
- ・「グラフビューで、カーブをピン付けするには」
- ・「コントロールポイント間の、カーブ形状の指定」



#### 注記

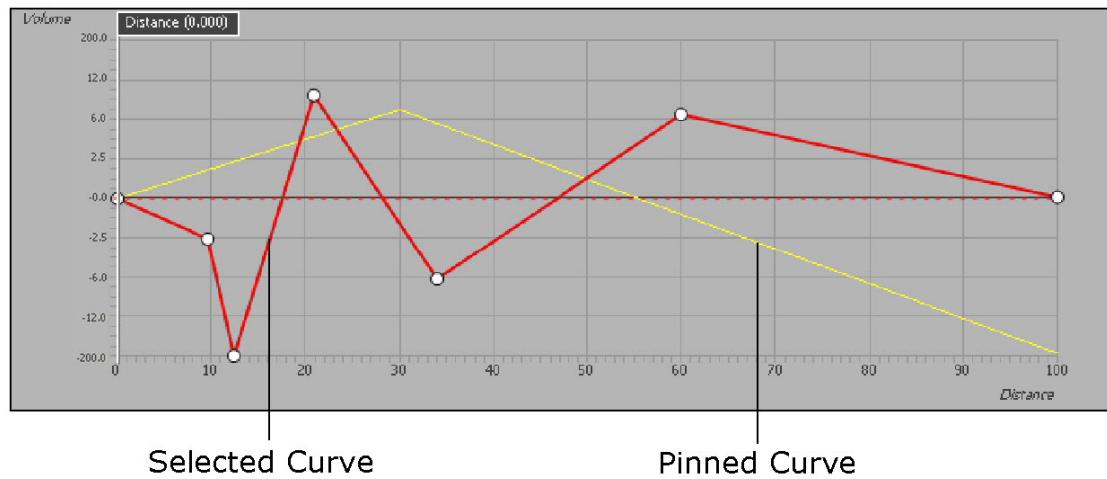
RTPCカーブは、あるプロパティから別のプロパティへ、またあるオブジェクトから別のオブジェクトへ、コピーできます。RTPCカーブのコピーについては「[RTPCカーブのコピー](#)」を参照してください。

### グラフビューで、カーブを表示する

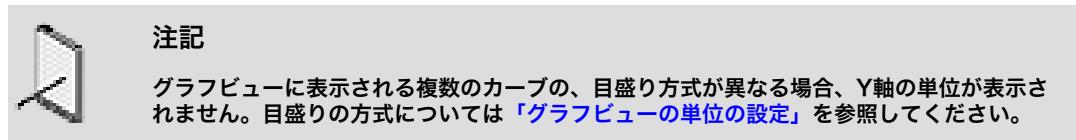
Wwiseのグラフビューの多くで、パスやGame ParameterとWwiseプロパティの間の関係を表すカーブを複数、表示できます。カーブリストでカーブを1つ選択すると、それがグラフビューに表示されます。また、RTPC Editorや、Attenuation Editorのグラフビューなどのように、複数カーブの表示に対応しているグラフであれば、Ctrlを押しながら、複数のカーブをクリックして、全てを同時にグラフビューに表示させることも可能です。

あるカーブを選択しなくても、グラフビューに表示させたままにするには、そのグラフビューにピン付け (Pin) できます。カーブをピン付けすると、そのアウトランが常にグラフビューに表示されます。しかし、このカーブのコントロールポイントは表示されません。カーブを選択しなければ、そのコントロールポイントの設定を変えられません。

選択されたカーブ (Selected curve) と、ピン付けされたカーブ (Pinned curve) の違いを、RTPCグラフビューを例に、下図に示します。



グラフビューに複数のカーブが表示されていると、XY軸の単位を表す数値が、表示されなくなることがあります。これは、カーブのGame Parameterの単位や、プロパティの単位が、異なる場合に起こります。例えば、同時にピッチカーブとボリュームカーブを表示すると、単位が「セント」と「デシベル」で違うので、Y軸の単位表示が消えます。



### グラフビューで、カーブを表示するには:

1. グラフビューで、カーブリストのカーブを1つ選択する。  
カーブが、グラフビューに表示される。
2. 他にも複数のカーブを表示させるには、Ctrlを押しながら、リストでカーブを選択する。

選択した複数のカーブが、グラフビューに表示されます。それぞれのカーブの表示単位が異なる場合は、X軸やY軸に、単位が表示されない。

### グラフビューで、カーブをピン付けするには:

1. カーブリストで、グラフビューにピン付けするカーブの、ピンアイコンをクリックする。

ピンアイコンが青色に変わり、このカーブがグラフビューに表示されます。

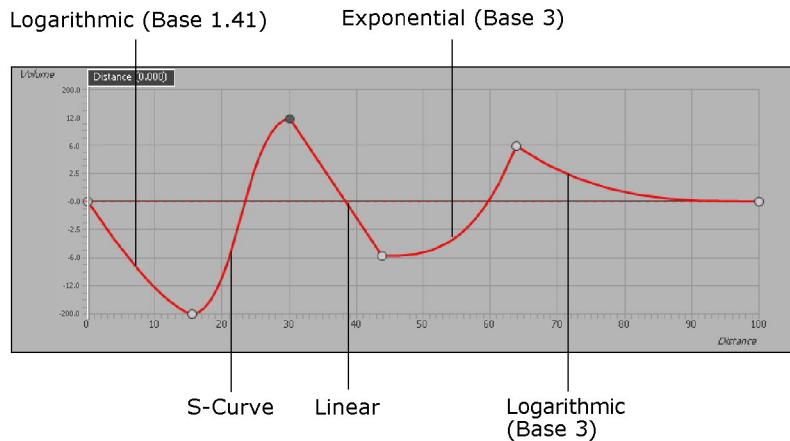
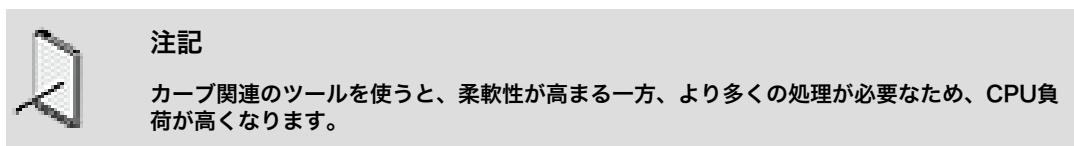
他のカーブを選択すると、ピン付けされたカーブのアウトラインが、ピン付けを解除するまで、グラフビューに表示されたままとなります。

### 関連トピック

- 「コントロールポイント間の、カーブ形状の指定」
- 「グラフビューのズームとパン」
- 「グラフビューの単位の設定」

### コントロールポイント間の、カーブ形状の指定

Wwiseでは、プロジェクトのカーブに対する操作性と柔軟性を高めるために、カーブセグメントの形状を変更できます。カーブセグメントとは、カーブ上の任意の2点のコントロールポイントの間の部分を指します。カーブ形状として、直線、定数、対数関数、指数関数、S型カーブなど、様々な選択肢があります。



### コントロールポイント間の、カーブ形状を指定するには:

1. グラフビューで、カーブのセグメントを1つ、右クリックする。  
ショートカットメニューが表示される。
2. メニューで、以下のオプションを1つ選択する。
  - Logarithmic (Base 3): 対数関数、底3
  - Sine (Constant Power Fade In): このサインカーブは、カーブの「入」部分で使う時だけ、コンスタンツパワークロスフェード (Constant power crossfade) を提供する。
  - Logarithmic (Base 1.41): 対数関数、底1.41

- Inverted S-Curve: 逆Sカーブ
- Linear: 直線
- Constant: 定数
- S-Curve: Sカーブ
- Exponential (Base 1.41): 指数関数、底1.41
- Sine (Constant Power Fade Out): このサインカーブは、カーブの「出」部分で使う時だけ、コンスタントパワークロスフェード (Constant power crossfade) を提供する。
- Exponential (Base 3): 指数関数、底3

選択したカーブ形状が、カーブのこのセグメントに適用される。

3. 必要に応じて、カーブの他のセグメントでも、カーブ形状を選択する。

## 関連トピック

- 「グラフビューで、カーブを表示する」
- 「グラフビューのズームとパン」
- 「グラフビューの単位の設定」
- 「コントロールポイントの追加」
- 「コントロールポイントの移動」
- 「コントロールポイントの削除」

---

## 第44章 タイムラインについて

概要 .....	823
ポジショニングにおけるタイムラインの使い方 .....	826
Music Segment Editorのタイムラインを使う .....	827

## 概要

Wwiseには時間を示すタイムラインが表示されるビューがいくつかあり、時間関連の情報が様々な目的のためにプロジェクトで使えます。タイムラインが表示されるビューを、以下に示します。

- Music Segment Editor
- Position Editor
- Performance Monitor
- Game Sync Monitor
- Voice Monitor

各種のタイムラインで使える機能は基本的に同じですが、いくつかの違いがあります。各タイムラインの目的を、下表に示します。

Use this timeline	To
Music Segment Editor	インタラクティブミュージックのコンポーネントをアレンジして、同期するため。
Position Editor	サウンドパスの期間を設定するため。
Performance Monitor	Performance Monitorの、キャプチャーされた時間関連の情報を確認するため。
Game Sync Monitor	ウォッチ（観察）中のゲームオブジェクトの時間に基づくRTCP値の変化を確認するため。
Voice Monitor	Wwiseオブジェクトのボイスのボリュームレベルを時間軸で確認するため。

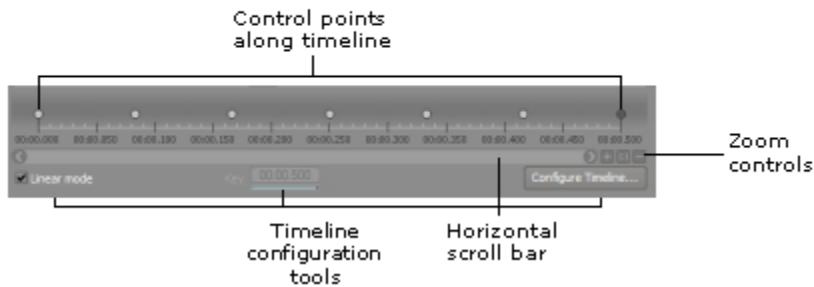
サウンドパスの作成や、パフォーマンスマニタリングの際に、タイムラインがグラフビューと連携して使われます。グラフビューについては「[グラフビューのコントロールポイントの使い方](#)」を参照してください。

## タイムラインのエレメントの説明

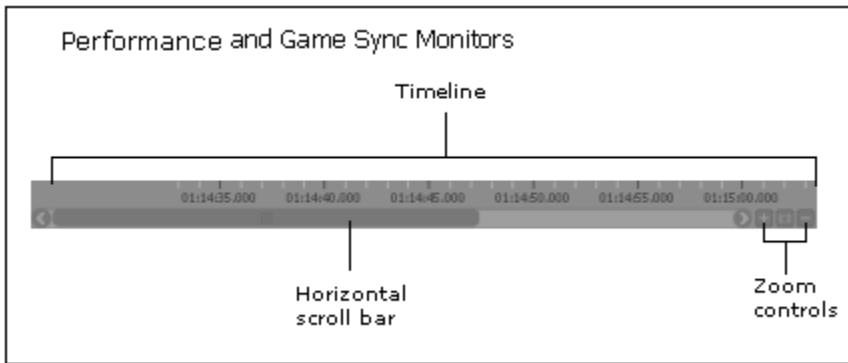
Wwiseには、それぞれ特徴のある3種類のタイムラインがあるので、個別に説明します。Position Editor (3D User defined)機能では、グラフビューで追加設定したコントロールポイントが、タイムライン上にも反映されます。タイムラインには、各コントロールポイントが時間的にどの位置になるのかが表示されます。ポイントの位置を、空間と時間の両方で設定することで、サウンドのポジションと、サウンドがパス（軌跡）上を移動するのにかかる時間を、決めることができます。Position Editor (3D User-defined)のタイムラインに表示される各種エレメントを、下図に示します。

## タイムラインについて

Position Editor Timeline

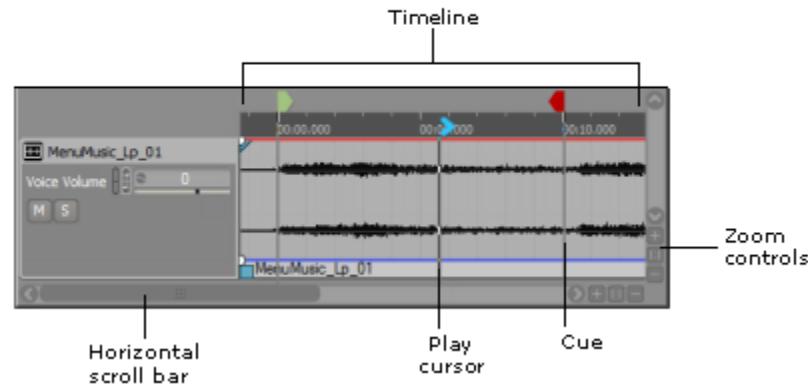


Performance Monitorではタイムラインを使って、サウンドエンジンからキャプチャーされて、Capture Logにログされた、各アクションやノティフィケーションを探せます。Performance Monitorのタイムラインに表示される各種エレメントを、下図に示します。



Music Segment Editorでは、タイムラインに沿って、ミュージッククリップがトラック内に表示されます。タイムラインと、波形自体を見ながら、ミュージックのトランジション、State変更、ステインガーなどにとって最適なセクションを確認します。タイムライン上の正確な位置に、エントリキュー、エグジットキュー、そして無制限の数のカスタムキューを、配置できます。

Music Editor Timeline



## タイムラインのパンとズーム

パンやズームのツールを使えば、情報を詳しく調査して、コントロールポイントやキーを追加する時間を、非常に細かく指定できます。可能な操作は、ズームイン・ズームアウトと、左右へのパンです。Music Segment Editorのタイムラインでは、これらのツールをXY軸の両方向で提供します。

全てのパンやズームの機能に、キーボードショートカットが設定されています。

To	ショートカット
ズームイン	Z+[マーキー選択]
現在のマウス位置を中心に、ズームイン	Ctrl+[マウスホイールを上に]
現在のマウス位置を中心に、ズームアウト	Ctrl+[マウスホイールを下に]
ズームイン中に、ズームをリセット	Z+[クリック]
ズームイン中に、左にパンニング	Shift+[マウスホイールを上に]
ズームイン中に、右にパンニング	Shift+[マウスホイールを下に]
ズームイン中に、上にパンニング	Ctrl+Shift+[マウスホイールを上に]
ズームイン中に、下にパンニング	Ctrl+Shift+[マウスホイールを下に]
フリーハンドでパンニング	X+[ドラッグ]

全てのショートカットを見るには付録C ショートカットを参照してください。

### タイムラインで、ズームインするには:

1. タイムラインのツールバーで、Zoom Inアイコンをクリックする。

タイムラインの中心に向かって、タイムラインがズームインします。

### タイムラインで、ズームアウトするには:

1. タイムラインのツールバーで、Zoom Outアイコンをクリックする。

タイムラインの中心から、タイムラインがズームアウトする。

### タイムラインで、パンとズームをリセットするには:

1. タイムラインのツールバーで、Resetアイコンをクリックする。

コントロールポイントを全て表示できるように、タイムラインがリセットされる。

### タイムラインで、パンするには:

1. タイムラインで、ズームインする。
2. 以下のいずれかを行う:
  - ・右にパンするには、マウスホイールを下に動かす。
  - ・左にパンするには、マウスホイールを上に動かす。

- ・フリーハンドでパンするには、Xを押して、タイムライン内でマウスをドラッグする。

## 関連トピック

- ・[「ポジショニングのタイムラインの設定」](#)
- ・[「Music Segment Editorのタイムラインの設定」](#)
- ・[「タイムラインで、Music Trackの長さを延長・短縮」](#)

## ポジショニングにおけるタイムラインの使い方

空間を移動するオブジェクトのパス（通り道）を作成する時に、そのパスを通り抜けるのにかかる時間を設定する必要があります。各コントロールポイントが時間的にどこにくるのかを、Position Editorのタイムラインで指定します。グラフィューと同様に、タイムラインでもズームやパンの機能を使ってコントロールポイントの位置を正確に設定できます。また、作成したパスごとにタイムラインの長さを変えられます。

- ・[「ポジショニングのタイムラインの設定」](#)

### ポジショニングのタイムラインの設定

タイムラインのプロパティや動作を、設定できます。例えば、タイムラインの長さや、新規コントロールポイントを追加された時に、タイムライン上のコントロールポイントの動作などを指定できます。タイムラインの長さによって、選択中のパスの長さが、自動的に決まります。作成する全てのパスの長さが異なるように、タイムラインを設定することもできます。

タイムラインが、Linear（均等）モードの場合は、タイムライン上の各ポイントの動作が既定されているので、設定できるのは、タイムラインの長さだけです。

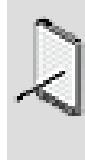
### Timelineを設定するには:

1. Position Editor (3D User-defined)で、Configure Timeline（タイムライン設定）ボタンをクリックする。  
Configure Timelineダイアログボックスが開く。
2. Lengthフィールドに、タイムラインの長さを、mm:ss.ms（分:秒:ミリ秒）単位で入力する。



3. タイムラインの長さを変更する場合、Non-linear（非均等）モードであれば、以下のオプションを1つ選択する。
  - ・Stretch proportionally（プロポーションを維持してストレッチ）：既存の複数のコントロールポイントの、タイムライン上の位置の相対関係を維持したまま、位置を変える場合。

- **Preserve key values** (主要な値を保持) : 既存のコントロールポイントの位置を、変えない場合。



#### 注記

Preserve key valuesオプションを使って、タイムラインの長さを短くする場合は、一部のコントロールポイントが削除される可能性があります。ポイントが削除されてしまう場合は、事前に確認メッセージが表示されます。

4. Insert Key Everyフィールドに、タイムラインに新規ポイントを挿入する時、最後の既存コントロールポイントとの間に空ける時間の長さを、入力する。
5. OKをクリックして、設定変更を承諾する。

新規設定に従って、タイムラインが再設定される。

### 関連トピック

- 「[タイムラインのパンとズーム](#)」
- 「[Music Segment Editorのタイムラインの設定](#)」
- 「[タイムラインで、Music Trackの長さを延長・短縮](#)」

## Music Segment Editorのタイムラインを使う

Music Segment Editorのタイムラインは、インタラクティブミュージックプロジェクトの様々なアレンジを管理するための、要となります。セグメント、トラック、クリップなどを使って作業する時に、タイムラインから重要なフィードバックを取得できます。

Music Segment Editorのタイムラインを使って作業をする時に、実行できるタスクを、以下に示します。

- 「[Music Segment Editorのタイムラインの設定](#)」
- 「[タイムラインで、Music Trackの長さを延長・短縮](#)」

### Music Segment Editorのタイムラインの設定

ミュージックを最適な方法でアレンジするために、以下の単位をタイムラインに表示できます。

- **Seconds (秒)** - タイムラインを秒単位で表示する場合。このビューにズームインすると、秒の表示が、1/10秒、1/100秒、1/1000秒と細かくなる。
- **Bars and Beats (バー、ビート)** - タイムラインをバーやビートで表示する場合。拍子にアラインさせる必要のあるミュージック素材を使う場合は、この設定にする。タイムラインの単位は、該当する、Property Editorで指定された、拍子記号 (Time Signature) に基づいて決まる。

### Music Segment Editorのタイムラインを、秒単位にするには:

1. タイムラインを右クリックして、Secondsを選択する。

タイムラインの表示が、秒単位に再設定されます。

## タイムラインについて



Music Segment Editorのタイムラインを、BarやBeat単位にするには：

1. タイムラインを右クリックして、Bars and Beatsを選択する。

タイムラインの表示が、バーとビートの単位に再設定されます。



## 関連トピック

- 「タイムラインのパンとズーム」
- 「ポジショニングのタイムラインの設定」
- 「タイムラインで、Music Trackの長さを延長・短縮」

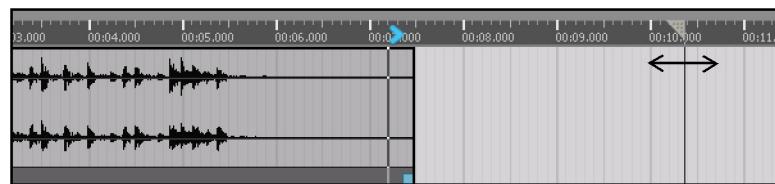
## タイムラインで、Music Trackの長さを延長・短縮

タイムライン上でクリップを使って作業をしている時に、タイムラインの長さを変更できます。

Music Segment Editorで、タイムラインのサイズを変更するには:

- 1.

エンドカーソル (End cursor) アイコンを、タイムライン上の適切な位置にドラッグする。



Drag end cursor icon to extend or shorten the segment tracks

## 関連トピック

- 「タイムラインのパンとズーム」
- 「ポジショニングのタイムラインの設定」
- 「Music Segment Editorのタイムラインの設定」

---

## 第45章 検索、クエリ、レファレンスの活用

概要 .....	830
プロジェクトエレメントを検索する .....	830
あるオブジェクトをレファレンスする、プロジェクトエレメントを検索する . 832	
クエリの使い方 .....	837
クエリに関する、コツとペストプラクティス .....	845

## 概要

Wwiseの1つのプロジェクトに、何千ものサウンドやモーションエフェクト、コンテナ、イベントなどのオブジェクトが入ることもあります。プロジェクトが大きくなればなるほど、プロジェクトエレメントを、素早く簡単に検索できることが重要になります。Wwiseでは、プロジェクトエレメントを探すための、効率的で強力な3種類の機能を提供します。

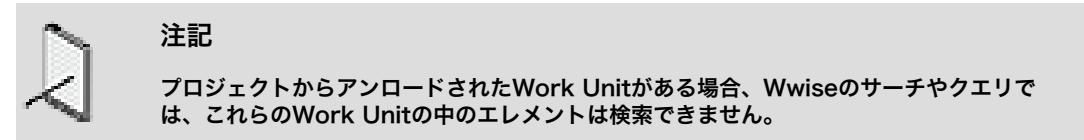
- ・ **検索 (Searches)** - 入力している最中から結果が表示される、素早いテキストベースの検索機能。
- ・ **クエリ (Queries)** - いくつかの条件や基準を設定できる、さらに強力な検索ツール。
- ・ **レファレンス (References)** - あるオブジェクトやエレメントを、直接レファレンス (参照) するプロジェクトエレメントを検索するための、検索ツール。

検索ツールは、素早く簡単に使えます。Search (検索) フィールドにテキストを入力するだけで、Wwiseが全てのプロジェクトエレメントの、名前と備考 (Notes) のフィールドを確認して、マッチするものを探します。結果 (Results) は、入力している最中から、Resultsリストに自動的に表示されます。

レファレンスツールも、とても素早く簡単に使えます。オブジェクトを右クリックして、Find All References (レファレンスを全て見つける) を選択するだけで、現在のオブジェクトへの、直接のレファレンス (参照) を含むオブジェクトなどのプロジェクトエレメントが、リストに表示されます。レファレンスのリストは、Referenceビューに表示されます。

一方、クエリツールの検索エンジンは、さらに洗練されて強力なものです。クエリに、具体的な基準 (Criteria) をいくつか設定することで、プロジェクトの特定エレメントを探し出せます。例えば、プリフェッヂ長さが100ミリ秒に設定された全てのストリーム用サウンドオブジェクトをクエリで探して、Show in Multi Editorコマンドを使い、全てを150ミリ秒に変更できます。クエリは、設定に多少の時間がかかりますが、完成すれば強力なツールで、開発サイクル中にいつでも、再利用できます。

これらの検索ツールを使いこなせば、どのようなエレメントでも、ほぼ確実にプロジェクト内から検出できます。



## プロジェクトエレメントを検索する

どのようなエレメントでも、検索ツールを使えば、プロジェクトで素早く簡単に探せます。検索ツールは、シンプルなテキストベースの検索機能です。Search (検索) フィールドにテキストを入力するだけで、Wwiseがプロジェクト全体を検索して、名前や備考 (Notes) がマッチする全てのエレメントを見つけ出します。マッチするものが検索された場合は、Resultsリストに自動的に表示されます。

検索ツールは、名前や備考 (Notes) フィールドにある全ての単語の冒頭を見て、マッチするものを探します。なお、Wwiseは必ず、スペース、アンダースコア、ダッシュなどの記号がある位置や、大文字・小文字の切り替えがある位置を、新しい単語が始まる位置と判断します。検索ツールは、この点を考慮してマッチする単語を探します。いくつかの例を、下表に示します。

入力した単語	検索結果	検出されない単語
Big	Big_Car Big Car Bigcar VeryBigCar	Verybigcar
Small Car	Small Fast Car Small_Car Very small car	Smallcar Verysmallcar
mediumcar	mediumcar VeryMediumcar Very_Mediumcar	Medium Car

List ビューから検索を実行できますが、Wwiseツールバーの右側にあるSearch フィールドからも、簡単に検索できます。結果のリストから、プロジェクトエレメントを選択して該当するEditor (編集画面) にロードできます。また、List ビューでは、リストのエントリ (項目) を複数選択して右クリックして、一括編集、コンバージョン、削除など様々なタスクを実行できます。

### プロジェクトのエレメントを検索するには:

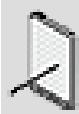
1. Wwiseのツールバーで、検索するプロジェクトエレメントの名前、備考 (Notes) 、またはその一部を、Search フィールドに入力する。

入力している最中から、名前または備考列の内容がマッチするプロジェクトエレメントのリストが、検索フィールドの下に表示される。

00:00:00.000			Not connected	Reconnect	Remote...	Footstep
Name	Type	Path				
Footstep_Crouching_Concrete_01	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Concrete\Footstep_...				
Footstep_Crouching_Concrete_02	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Concrete\Footstep_...				
Footstep_Crouching_Concrete_03	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Concrete\Footstep_...				
Footstep_Crouching_Concrete_04	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Concrete\Footstep_...				
Footstep_Crouching_Concrete_05	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Concrete\Footstep_...				
Footstep_Crouching_Gravel_01	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Gravel_02	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Gravel_03	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Gravel_04	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Gravel_05	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Gravel_06	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Gravel\Footstep_Cr...				
Footstep_Crouching_Hardmetal...	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Hard_Metal\Footste...				
Footstep_Crouching_Hardmetal...	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Hard_Metal\Footste...				
Footstep_Crouching_Hardmetal...	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Hard_Metal\Footste...				
Footstep_Crouching_Hardmetal...	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Hard_Metal\Footste...				
Footstep_Crouching_Hardmetal...	Sound	\Footsteps\Footstep_Types\FS_Crouching\FS_Hard_Metal\Footste...				

2. 以下のいずれかを行う：

- **Open Results in List View**（結果をリストビューで開く）をクリックすると、結果の一覧が、Listビューに表示される。
- **Open Results in Schematic View**（結果をスケマティックビューで開く）をクリックすると、結果の一覧が、Schematicビューに表示される。
- 目的のプロジェクトエレメントをクリックして、該当するEditor（エディタ画面）にロードする。
- 矢印キーを使ってResultsリストをスクロールして、Enterを押して、選択したエレメントを該当するEditor（エディタ画面）にロードする。



注記

Listビューで結果の一覧を表示して、エントリ（項目）を1つ選択すると、自動的に、Transport Controlにロードされるので、すぐに再生できます。また、リストのエントリを複数選択して右クリックして、一括編集、コンバージョン、削除など様々なタスクも実行できます。

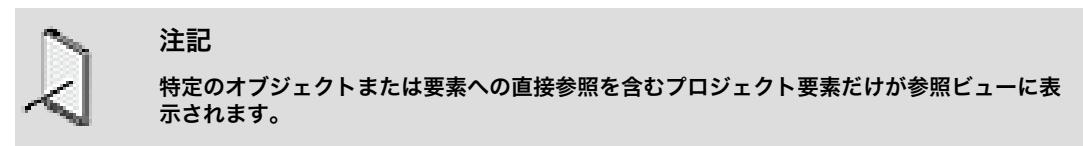
関連トピック

- 「クエリの作成」
- 「クエリを設定して実行する」
- 「クエリの削除」
- 「表の使用」
- 「Batch Rename機能を使う」

## あるオブジェクトをレファレンスする、プロジェクトエレメントを検索する

プロジェクトの開発中に、ある特定のオブジェクトを直接レファレンス（Reference）する、全てのエレメントを、プロジェクト内で検索したい場合も考

えられます。例えば、あるオブジェクトをレファレンスするイベントはどれか、またはあるイベントをレファレンスするSoundBankはどれかを、確認したい時などです。Wwiseでは、Find all references（レファレンスを全て検索）コマンドが、右クリックで表示させるほぼ全てのコンテキストメニューにあるので、ここから簡単に検索できます。特定のオブジェクトをレファレンスする全てのエレメントが、Referenceビューに表示されるので、ここから簡単に、表示されたプロジェクトエレメントを選択して開き、必要に応じて設定を変更して、次に進むことができます。



次の表は、すべてのWwiseプロジェクトエレメントとそれらを参照できる各要素を示しています。

アイコン	プロジェクトエレメント	から参照できます：
	Virtual Folder	<ul style="list-style-type: none"><li>• SoundBank</li></ul>
Master Mixerアイコン		
	Audio Bus	<ul style="list-style-type: none"><li>• Actor-Mixer</li><li>• Audio Bus</li><li>• Audio Source</li><li>• Containers</li><li>• Mixing Session</li><li>• Music Playlist Container</li><li>• Music Segment</li><li>• Music Switch Container</li><li>• Music Track</li><li>• Sound SFX</li><li>• SoundBank</li><li>• Soundcaster Session</li><li>• Source Plug-in</li></ul>
	Auxiliary Bus	<ul style="list-style-type: none"><li>• Actor-Mixer</li><li>• Audio Source</li><li>• Containers</li><li>• Mixing Session</li><li>• Music Playlist Container</li><li>• Music Segment</li><li>• Music Switch Container</li><li>• Music Track</li><li>• Sound SFX</li><li>• SoundBank</li><li>• Soundcaster Session</li><li>• Source Plug-in</li></ul>
Actor-Mixerアイコン		
	Audio source	<ul style="list-style-type: none"><li>• N/A</li></ul>
	Source Plug-in	<ul style="list-style-type: none"><li>• N/A</li></ul>
	Sound SFX	<ul style="list-style-type: none"><li>• Dialogue Event 「ダイアログイベント」</li></ul>

検索、クエリ、レファレンスの  
活用

アイコン	プロジェクトエレメント	から参照できます:
		<ul style="list-style-type: none"> <li>• Event</li> <li>• Mixing Session</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Sound Voice	<ul style="list-style-type: none"> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Work Unit	<ul style="list-style-type: none"> <li>• SoundBank</li> </ul>
	Actor-Mixer	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Blend Container	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Random Container	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Sequence Container	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Switch Container	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
Interactive Musicアイコン		
	Music Track (all types)	<ul style="list-style-type: none"> <li>• Mixing Session</li> <li>• SoundBank</li> </ul>

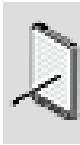
## 検索、クエリ、レファレンスの 活用

アイコン	プロジェクトエレメント	から参照できます：
	Music Segment	<ul style="list-style-type: none"> <li>• Mixing Session</li> <li>• Music Playlist Container</li> <li>• Music Switch Container</li> <li>• SoundBank</li> </ul>
	Music Playlist Container	<ul style="list-style-type: none"> <li>• Mixing Session</li> <li>• Music Switch Container</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
# 	Music Switch Container	<ul style="list-style-type: none"> <li>• Event</li> <li>• Music Track</li> <li>• Mixing Session</li> <li>• SoundBank</li> </ul>
他のプロジェクトエレメントのアイコン		
	Event	<ul style="list-style-type: none"> <li>• Event</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> <li>• Work Unit</li> </ul>
	Dialogue Event 「ダイアログイベント」	<ul style="list-style-type: none"> <li>• Event</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> <li>• Work Unit</li> </ul>
	SoundBank	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
	Switch Group	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Containers</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Music Switch Container</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	Switch	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Containers</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Music Switch Container</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	State Group	<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Containers</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Mixing Session</li> <li>• Music Switch Container</li> <li>• Sound SFX</li> <li>• SoundBank</li> <li>• Soundcaster Session</li> </ul>
	State	<ul style="list-style-type: none"> <li>• Actor-Mixer</li> </ul>

## 検索、クエリ、レファレンスの 活用

アイコン	プロジェクトエレメント	から参照できます：
		<ul style="list-style-type: none"> <li>• Audio Source</li> <li>• Containers</li> <li>• Dialogue Event 「ダイアログイベント」</li> <li>• Event</li> <li>• Music Switch Container</li> <li>• Sound SFX</li> <li>• Soundcaster Session</li> <li>• Source Plug-in</li> <li>• State</li> <li>• State Group</li> </ul>
	Game Parameter (RTPC) 「ゲームパラメータ (RTPC)」	<ul style="list-style-type: none"> <li>• Actor-Mixer</li> <li>• Audio Bus</li> <li>• Containers</li> <li>• Effect</li> <li>• Event</li> <li>• Sound SFX</li> <li>• Switch Group</li> </ul>
	Trigger	<ul style="list-style-type: none"> <li>• Event</li> <li>• Music Playlist Container</li> <li>• Music Segment</li> </ul>
	Effect ShareSet	<ul style="list-style-type: none"> <li>• Auxiliary Bus</li> <li>• Sound SFX</li> <li>• SoundBank</li> </ul>
	Attenuation ShareSet	<ul style="list-style-type: none"> <li>• Actor-Mixer</li> <li>• Audio Source</li> <li>• Containers</li> <li>• Sound SFX</li> </ul>
	Conversion Settings ShareSet	<ul style="list-style-type: none"> <li>• Actor-Mixer</li> <li>• Audio Bus</li> <li>• Containers</li> <li>• Effect</li> <li>• Event</li> <li>• Music Playlist Container</li> <li>• Music Segment</li> <li>• Music Switch Container</li> <li>• Sound SFX</li> <li>• Sound Voice</li> <li>• Switch Group</li> </ul>
	Soundcaster Session	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
	Mixing Session	<ul style="list-style-type: none"> <li>• N/A</li> </ul>
	Query クエリ	<ul style="list-style-type: none"> <li>• N/A</li> </ul>

多くの場合、レファレンス検索は1つの特定オブジェクトを条件としますが、複数のオブジェクトをレファレンスするエレメントを全て探すこともできます。



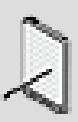
#### 注記

Referenceリストは自動的に更新されないので、プロジェクトに変更があった場合はリストで、Refreshボタンをクリックして、マニュアル操作で更新する必要があります。

あるオブジェクトをレファレンスする、プロジェクトエレメントを検索するには:

1. 以下のいずれかを行う:

- プロジェクトエレメントを1つ選択するか、複数を一括選択して、右クリックして、コンテキストメニューで、Find all references（レファレンスを全て検索）を選択する。
- Shift+F3を押してReferenceビューを開き、オブジェクト（単数または複数）を選択してReferences to:フィールドにドラッグする。
- Shift+F3を押して、Referenceビューを開き、References to:フィールドの横の、ブラウズボタンをクリックして、プロジェクトエレメントを選択してから、OKをクリックする。



#### 注記

Project Explorer、Property Editor、またはReference Viewで、オブジェクトやエレメントを選択した状態で、Shift+F3を押すと、選択されたオブジェクトへのレファレンスを含むオブジェクトを、Wwiseが自動的に検索します。

選択中のオブジェクト（単数または複数）へのレファレンスを含むプロジェクトエレメントのリストが、Referenceビューに表示される。

#### 関連トピック

- [「プロジェクトエレメントを検索する」](#)
- [「クエリの使い方」](#)

## クエリの使い方

プロジェクトの特定のオブジェクトを見つける必要がある時に、検索ツールでは不十分な場合は、Query Editorでクエリを作成します。Query Editorは、プロジェクト内を全て検索して目的のオブジェクトを見つけるための、洗練された強力なツールです。クエリは、目的に応じて、幅広くも細かくも設定できます。また、クエリを保存したり再ロードしたりすれば、ワークフローの効率も上がります。クエリを実行してから、Resultsペインに表示されたオブジェクトを全て、一括編集できます。例えば、Multi Editorでストリーミングをオン・オフすることや、Resultsペインに表示されたオブジェクト全体に対して、コンバージョン設定を変更することもできます。

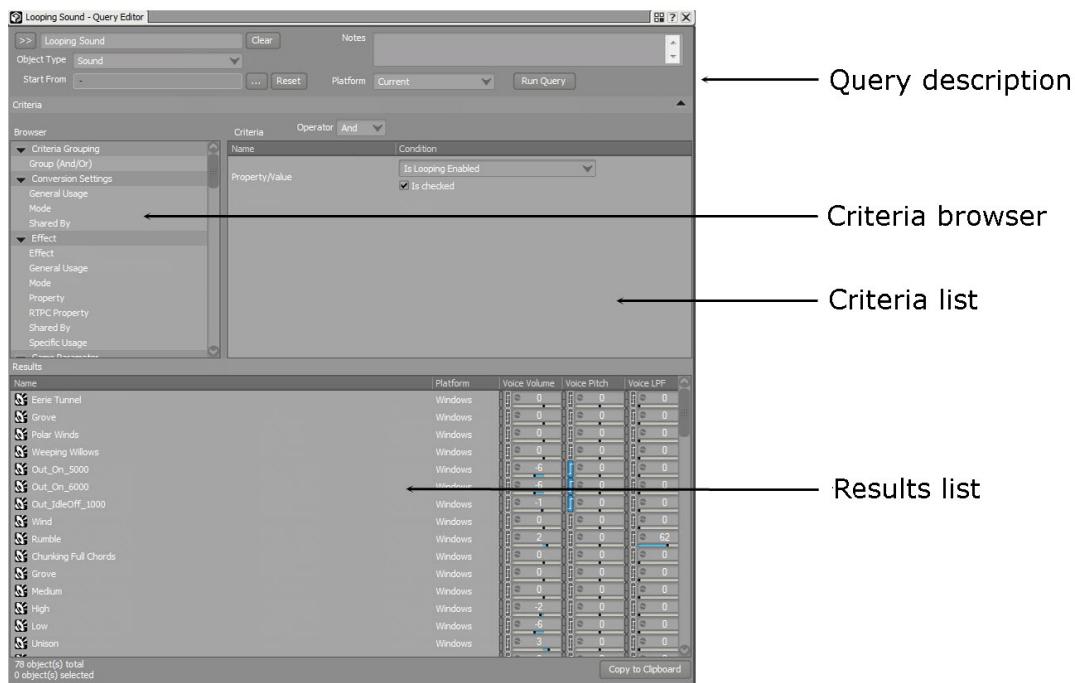
クエリを作成して実行する例を、以下に示します。

- 「Sword」という単語で始まるプロジェクト内のサウンドを全て見つけるには、サウンド用のクエリを作成して、サウンドタイプとしてSFXを指定して、名前を「sword\*」に設定する。

## 検索、クエリ、レファレンスの活用

- あるバスを経由してルーティングされるMusic Segmentを全て見つけるには、Music Segment用のクエリを作成して、アウトプットバスを指定する。
- LPFプロパティが10を超えるボイスを全て見つけるには、サウンド用のクエリを作成して、サウンドタイプをVoiceに指定して、プロパティとしてローパスフィルタ値を10より上に設定する。

Query Editorを構成するセクションを、以下に示します。



## クエリの作成

Wwiseでクエリを作成すると、プロジェクトでオブジェクトを探すために使う、固有の検索条件を設定することになります。クエリに名前を付けると、後から使えるように保存されます。

Wwiseでクエリを作成できるのは、以下の2つの場所です。

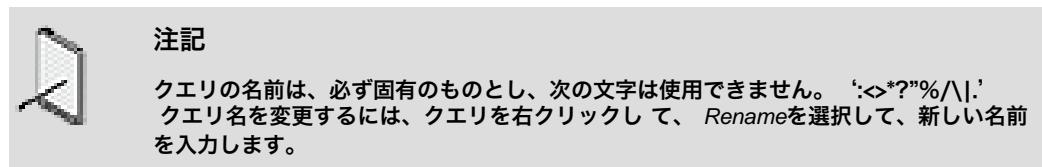
- Project Explorerの、Queriesタブ
- Query Editor画面

### 新規クエリを作成するには:

- Project Explorerで、Queriesタブに切り替える。
- 以下のいずれかを行う:
  - Work Unit (または仮想フォルダ) を1つ選択して、Project Explorerのツールバーで、**Query**アイコンをクリックする。
  - Work Unit (または仮想フォルダ) を1つ右クリックして、ショートカットメニューで、**New Child > Query**を選択する。

選択したWork Unit (または仮想フォルダ) に、新規クエリが追加されます。

3. デフォルトのクエリ名を、内容が分かる名前に変更して、Enterを押す。



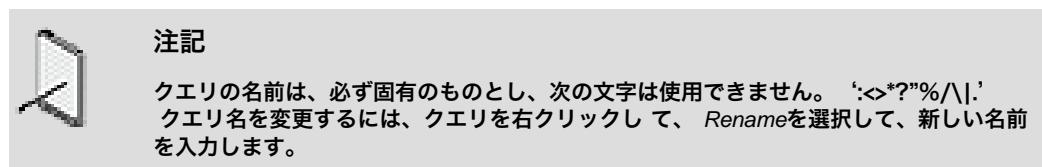
**Query Editorで、新規クエリを作成するには:**

1. Query Editorで、New...をクリックする。

ダイアログNew Queryが開きます。

2. 作成するクエリの、保存先となるWork Unitを選択する。

3. クエリ名を入力する。



4. OKをクリックする。

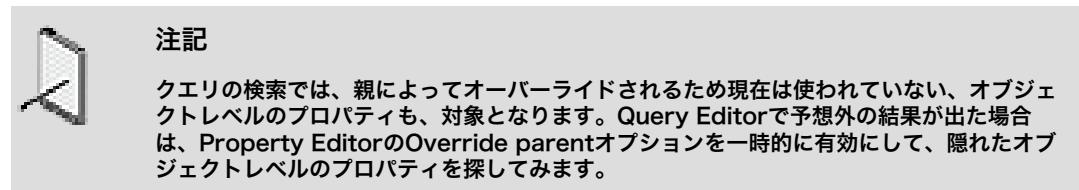
新規クエリが作成される。

## 関連トピック

- 「クエリを設定して実行する」
- 「クライティアグループを使った、クエリの詳細設定」
- 「ファクトリ設定のクエリを使用する」
- 「クエリの削除」
- 「プロジェクトエレメントを検索する」

## クエリを設定して実行する

クエリを最初に実行する時に、プロジェクトを検索するための条件を設定します。  
これらの条件は自動的に保存され、あとからまた同じクエリを実行する時に、再利用できます。



**クエリの内容を設定するには:**

1. Query Editorに、クエリを1つロードする。

2. クエリに関するその他の情報を、Notesフィールドに入力する。

3. Object Typeリストで、検索するオブジェクトやプロジェクトエレメントのタイプを選択する。特定のオブジェクトタイプを特定しない場合は、All Objects（全オブジェクト）を選択する。

4. 階層内のどの位置から検索を開始するのかを指定するには、以下を行う：

Start From Hereボックスの横の、ブラウズボタン (...)をクリックする。

Project Explorer - Browserダイアログボックスが開く。

Wwiseで検索を開始するフォルダまで、移動する。

OKをクリックする。



5. Platformリストで、以下のいずれかを選択する。

- Allは、プロジェクトの全てのプラットフォームを対象に、オブジェクトの検索を行う場合。
- Currentは、現在選択中のプラットフォームで使われるオブジェクトを検索する場合。

6. Browserリストで、クエリに使うクライテリアを全て選択する。以下を行う：

Criteriaリストにロードするクライテリアを1つ、ダブルクリックする。

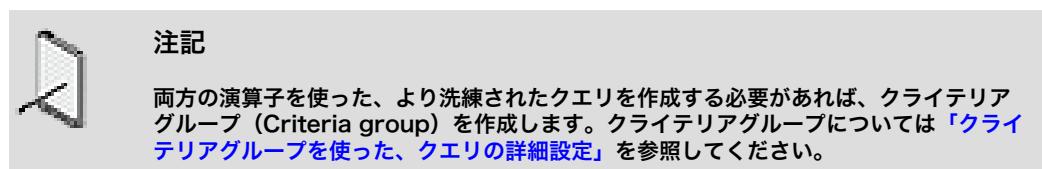
クライテリアの条件を入力、または選択する。クライテリアを入力する時に、単語の一部をワイルドカード演算子 (Wildcard operator) の「\*」で置き換えるてもよい。



7. Operator（演算子）リストで、以下のいずれかを選択する。

Andは、全てのクライテリアが「true」である時の結果を表示する場合。

Orは、1つ以上のクライテリアが「true」である時の結果を表示する場合。



8. Run Query（クエリ開始）をクリックする。

クエリの結果が表示される。列のヘッダをクリックすると、結果をソートできる。



### Tip

Resultsリストのエントリ（項目）を1つ選択すると、自動的に、Transport Controlにロードされるので、すぐに再生できます。また、Resultsリストのエントリを複数選択して右クリックして、Edit、Show in Multi Editor、Show in Schematic View、Convertなど様々なコマンドもアクセスできます。Resultsリストのエントリをダブルクリックすると、そのエントリが、Property Editorに表示されます。また、プロジェクトエレメントを、Resultsリストから、Project Explorerにドラッグできます。例えば、Resultsリストのサウンドオブジェクト（単数または複数）を、Project Explorerに表示されたアクターミキサーやコンテナに、ドラッグすることができます。

9. クエリ結果を他のアプリケーションで使うには、Copy to Clipboard（クリップボードにコピー）をクリックする。

10. クライティアや検索結果をクリアするには、Clearをクリックする。



### 注記

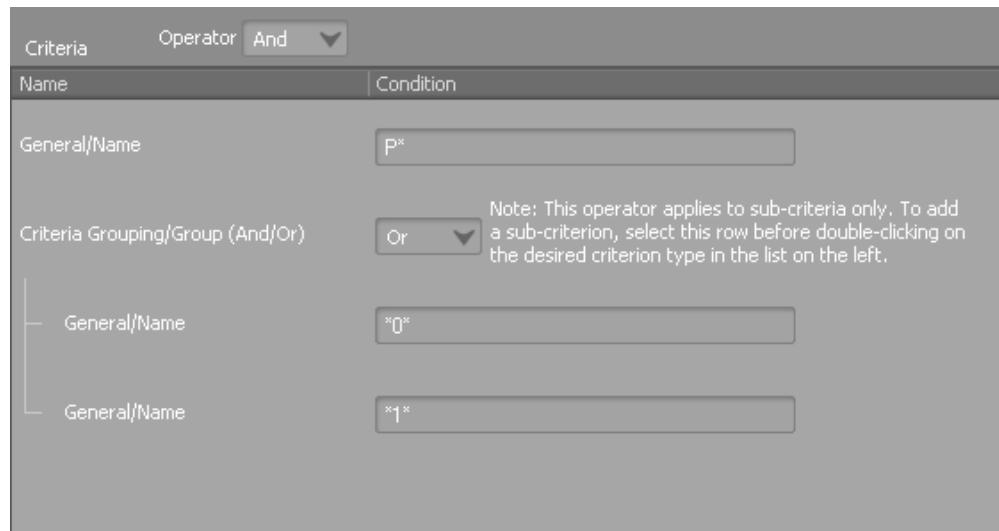
Wwiseのオブジェクトには、親オブジェクトのプロパティによって置き換えられた後も存続する、固有のプロパティ値があります。検索結果に誤検出が発生する場合、つまり検索クライティアを含まないオブジェクトが検出される場合は、これらのオブジェクトに、Override parentオプションでオーバーライドされている固有プロパティがある可能性があります。

## 関連トピック

- 「クエリの作成」
- 「クライティアグループを使った、クエリの詳細設定」
- 「ファクトリ設定のクエリを使用する」
- 「クエリの削除」
- 「プロジェクトエレメントを検索する」

### クライティアグループを使った、クエリの詳細設定

1つの演算子（Operator / オペレータ）を設定した基本的なクエリでは詳細が不充分で、探しているものが正確に見つからない状況も考えられます。このような場合には、クライティアグループ（Criteria group）を使って、さらに詳細な検索を設定します。クライティアグループは、条件（Criteria）のサブセットであり、別の演算子を設定できます。これを使えば、様々な演算子を設定した1つのクエリを作成できます。例えば、名前（Name）が「P」で始まり、かつ（AND）、名前に「1または0（1 OR 0）」が含まれる、全てのオブジェクトを探し出すクエリを作成できます。



1つのクエリ内にクライティアグループ（単数または複数）を設定して、特定のオブジェクトやその他のエレメントを、プロジェクトから探し出せます。クライティアグループを、さらに別のクライティアグループの中に入れて（ネスト化して）、非常に細かいクエリを設定することも可能です。

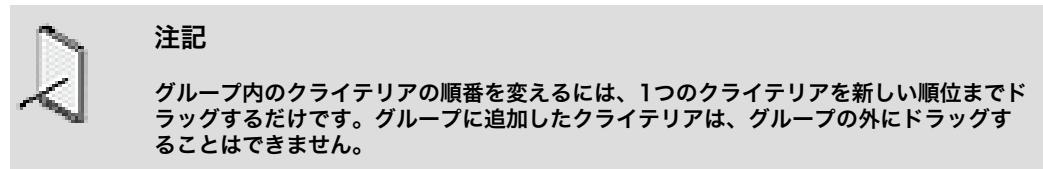
#### クエリ内に、クライティアグループを設定するには:

1. Query Editorに、クエリを1つロードする。
2. Browserリストで、Criteria Groupingセクションの、Group (And/Or)オプションをダブルクリックする。  
Criteriaリストに、新規クライティアグループが追加される。
3. Browserリストで、クライティアグループにクライティア（単数または複数）を追加するには、以下を行う：

Criteriaリストで、クライティアグループを選択する。

Criteriaグループに追加するクライティアを1つ、ダブルクリックする。

クライティアの条件を入力、または選択する。クライティアを入力する時に、单語の一部をワイルドカード演算子（Wildcard operator）の「\*」で置き換えるてもよい。



4. Criteria Group Operator（クライティアグループ演算子）リストで、以下のいずれかを選択する。

**And**は、グループ内の全てのクライティアが「true」である時の結果を表示する場合。

Orは、グループ内の1つ以上のクライテリアが「true」である時の結果を表示する場合。



#### 注記

グループ外にクライテリアを追加するには、グループの外側をクリックしてグループの選択を外し、Browserリストに表示されたクライテリアを1つ、ダブルクリックする。

#### 5. Run Query (クエリ開始) をクリックする。

クエリの結果が表示される。列のヘッダをクリックすると、結果をソートできる。



#### Tip

Resultsリストのエントリ（項目）を1つ選択すると、自動的に、Transport Controlにロードされるので、すぐに再生できます。また、Resultsリストのエントリを複数選択して右クリックして、Edit、Show in Multi Editor、Show in Schematic View、Convertなど様々なコマンドもアクセスできます。Resultsリストのエントリをダブルクリックすると、そのエントリが、Property Editorに表示されます。また、プロジェクトエレメントを、Resultsリストから、Project Explorerにドラッグできます。例えば、Resultsリストのサウンドオブジェクト（単数または複数）を、Project Explorerに表示されたアクターミキサー или コンテナに、ドラッグすることができます。

#### 6. クエリ結果を他のアプリケーションで使うには、Copy to Clipboard (クリップボードにコピー) をクリックする。

#### 7. クライテリアや検索結果をクリアするには、Clearをクリックする。

### 関連トピック

- ・「クエリを設定して実行する」
- ・「ファクトリ設定のクエリを使用する」
- ・「クエリの削除」

### ファクトリ設定のクエリを使用する

Wwiseには、事前に定義されたファクトリプリセットがあり、一般的なクエリを数多く準備しています。クエリを一から定義する代わりに使えば、時間を節約できます。

### ファクトリ設定のクエリを使用するには:

1. Project Explorerで、Queriesタブに切り替える。
2. Factory Queriesリストで、クエリを1つダブルクリックする。

このクエリが、Query Editorにロードされる。

3. 要件に従い、クライテリアの条件を入力、または選択する。クライテリアを入力する時に、単語の一部をワイルドカード演算子（Wildcard operator）の「\*」で置き換えてよい。
4. Run Query (クエリ開始) をクリックする。

クエリの結果が表示される。列のヘッダをクリックすると、結果をソートできる。



### Tip

Resultsリストのエントリ（項目）を1つ選択すると、自動的に、Transport Controlにロードされるので、すぐに再生できます。また、Resultsリストのエントリを複数選択して右クリックして、Edit、Show in Multi Editor、Show in Schematic View、Convertなど様々なコマンドもアクセスできます。Resultsリストのエントリをダブルクリックすると、そのエントリが、Property Editorに表示されます。また、プロジェクトエレメントを、Resultsリストから、Project Explorerにドラッグできます。例えば、Resultsリストのサウンドオブジェクト（単数または複数）を、Project Explorerに表示されたアクターミキサー или コンテナに、ドラッグすることができます。

5. クエリ結果を他のアプリケーションで使うには、Copy to Clipboard（クリップボードにコピー）をクリックする。
6. クライティアや検索結果をクリアするには、Clearをクリックする。

## 関連トピック

- [「クエリの作成」](#)
- [「クライティアグループを使った、クエリの詳細設定」](#)
- [「クエリの削除」](#)
- [「プロジェクトエレメントを検索する」](#)

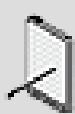
## クエリの削除

不要となったクエリは、削除できます。

### クエリを削除するには:

1. Project Explorerで、Queriesタブに切り替える。
2. 削除するクエリを右クリックして、Delete Selectionを選択する。

選択したクエリが削除される。



### 注記

誤ってクエリを削除した場合は、Ctrl+Z を押すか、Edit > Undoをクリックして、削除を取り消します。

## 関連トピック

- [「クエリの作成」](#)
- [「クライティアグループを使った、クエリの詳細設定」](#)
- [「ファクトリ設定のクエリを使用する」](#)
- [「クエリを設定して実行する」](#)
- [「プロジェクトエレメントを検索する」](#)

## クエリに関する、コツとベストプラクティス

Wwiseでクエリを作成する時に、設定できるオプション機能が多数あります。クエリを計画する際に効率的な戦略を立てると、必要な結果を正確に、迅速に、そして確実に入手できます。プロジェクトでクエリを構築する際に、以下の戦略を検討してみてください。

### オブジェクトタイプの詳細

ブラウザーリストのクライテリアの多くに、クエリの的をさらに絞るための条件設定があります。設定できる条件は、選択したオブジェクトのタイプによって異なります。例えば、Property Value（プロパティ値）というクライテリアに関する事例を、以下に示します。

- Object Type（オブジェクトタイプ）として「Audio Source（オーディオソース）」を選択した場合、Bit Depth（ビット深度）や、Sample Rate（サンプルレート）などの条件を選択できます。
- Object Type（オブジェクトタイプ）として「Event（イベント）」を選択した場合、Property Valueがこのオブジェクトタイプの有効なクライテリアではないので、何も選択できません（Wwiseでは、イベントにプロパティがありません）。

以上の通り、オブジェクトタイプとして何を選ぶかによって、クエリで設定できる条件が決まるので、非常に重要な選択です。

### クエリのスピードアップ

いくつかの単純な条件を設定することで、クエリをさらに速くできます。

- オブジェクトタイプを指定する。
- 特定のプラットフォーム専用のものを探している場合は、プラットフォームを指定する。
- Start From Here（ここからスタート）を使い、階層のできるだけ下からクエリを開始する。

### クエリで、ワイルドカードを活用

ワイルドカード（Wildcards）とは、キーワード検索をさらに柔軟にする記号のことで、検索する単語のパラメータを拡張させます。Wwiseのワイルドカード演算子であるアステリスク (\*) で、単語の一部を置き換えて、特定の文字や数字を含む、または始まる、または終わるような、様々なプロジェクトエレメントを探す時に、利用できます。例えば、名前に「LOOP」が含まれる全てのオブジェクトを探す時は、Condition（条件）のテキストボックスに、\*LOOP\*と入力します。一方、「LOOP」という言葉で始まる全てのオブジェクトを探すには、Conditionのテキストボックスに、LOOP\*と入力します。最後に、「LOOP」という言葉で終わる全てのオブジェクトを探すには、Conditionのテキストボックスに、\*LOOPと入力します。

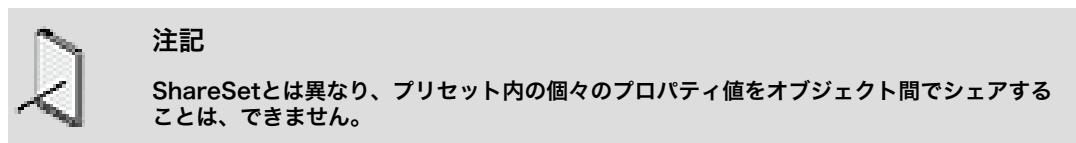
---

## 第46章 プリセットを使用する

概要 .....	847
プリセットを使用する .....	847

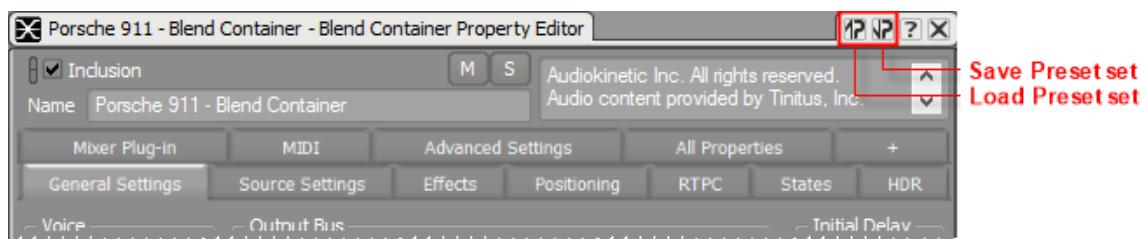
## 概要

プリセット (Presets) は、あるオブジェクトまたはエフェクトに関して設定したプロパティ値のセットのことです。特別なファイルとして保存して、同じプロジェクトで後から再利用できるものです。プリセットを使えば、プロジェクトの他のオブジェクトで再利用したい具体的なプロパティ設定を、再設定する必要がなくなります。一度プロパティ値を設定すれば、これをプリセットとして保存して、プロジェクトの他のオブジェクトに適用できます。これで作業効率が向上し、時間と労力の節約につながります。



## プリセットを使用する

プリセット用アイコンは、プリセットの保存やロードができる全てのビューで、タイトルバーに表示されます。



Load Preset (プリセットのロード) と、Save Preset (プリセットの保存) のアイコンを、下表に示します。

アイコン	名前
	Load Preset
	Save Preset

### プリセットを保存する

プリセットとして保存できるのは、Wwiseエレメントの以下の設定です。

- Actor-Mixer HierarchyやInteractive Music Hierarchyの、オブジェクトやソースに関して設定した、プロパティ値やプロパティ設定
- Positioning values
- Attenuation settings
- Effect values

プリセットを保存すると、Wwiseが、そのビューの全てのタブにある、全ての値を保存します。保存したプリセットは、以下のカテゴリグループに分けられます。

- Actor-Mixer

- Random Container、Sequence Container
- Switch Container
- Blend Container
- サウンドSFX、サウンドボイス
- オーディオソースプラグイン
- Music Switch Container
- Music Playlist Container
- Music Segment
- Music Track
- Positioning (3D User-defined)
- Positioning (2D)
- Attenuation
- エフェクト

Save Presetダイアログボックスを開くと、プリセットがフィルターされて、同じカテゴリのプリセットだけが表示されます。

### プリセットを保存するには:

1. ビューのタイトルバーで、Save Presetアイコンをクリックする。

Save Presetダイアログボックスが開く。

2. Nameフィールドに、プリセット名を入力する。
3. Notesフィールドに、このプリセットの内容を示す情報を入力する。
4. Saveをクリックするか、Enterを押す。

プリセットが保存され、同じプロジェクト内で、いつでも再利用できる。

### 関連トピック

- 「[プリセットをロードする](#)」
- 「[プリセットを削除する](#)」

### プリセットのコンテンツ (内容)

オブジェクトプロパティのプリセットを、プロジェクト階層の様々なレベルで保存できるので、プリセットによって、どのエレメントが保存されるのかを知っておくことが重要です。全てのプリセットと、そこに保存される情報を、下表に示します。

Preset	階層における位置	保存内容
オブジェクトのプロパティ、ソースのプロパティ	最上位オブジェクト	Property Editorに表示される、全てのタブの、全てのプロパティ値。
	子オブジェクト	Property Editorに表示される、全てのタブの、全てのプロパティ値。

Preset	階層における位置	保存内容
		そのオーバーライド設定の下では、何も Preset に保存されません。Preset に保存するときに有効にした場合、Preset はそれに対応するすべてのオーバーライドプロパティ値を含みます。しかし、オーバーライドオプションはデフォルトで有効になっていません。オーバーライドプロパティは、一旦 <b>Override parent</b> が有効になったときに適用されます。
Effect	-	対象エフェクトの、全てのプロパティ値。
Positioning (2D)	-	ポイントソースの位置。
Positioning (3D User-defined)	-	Position Editor (3D User-defined) に表示される、ポジショニングに関する全てのパス、設定、値。
Attenuation	-	全てのカーブ（グラフ）と、全ての減衰設定。

## プリセットをロードする

保存したプリセットは、プロジェクトの他のオブジェクトやエフェクトに適用できます。保存したプリセットは、以下のカテゴリグループに分けられます。

- Actor-Mixer
- Random Container、Sequence Container
- Switch Container
- Blend Container
- サウンドSFX、サウンドボイス
- オーディオソースプラグイン
- Music Switch Container
- Music Playlist Container
- Music Segment
- Music Track
- Positioning (3D User-defined)
- Positioning (2D)
- Attenuation
- エフェクト

Load Preset ダイアログボックスを開くと、プリセットがフィルターされて、同じカテゴリのプリセットだけが表示されます。

### プリセットをロードするには:

1. ビューの、タイトルバーで、Load Preset アイコンをクリックする。  
Load Preset ダイアログボックスが開く。
2. リストで、プリセットを1つ選択する。
3. Load をクリックするか、Enter を押す。

オブジェクトまたはエフェクトに、プリセットが適用される。

## 関連トピック

- ・「[プリセットを保存する](#)」
- ・「[プリセットを削除する](#)」

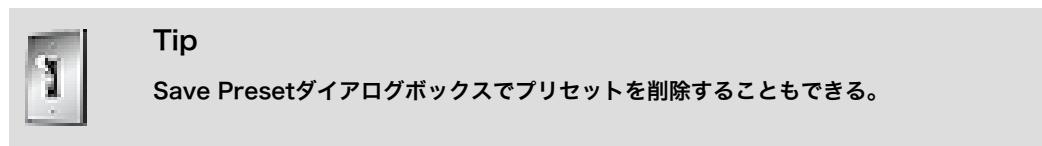
## プリセットを削除する

不要となったプリセットは、削除できます。

### プリセットを削除するには:

1. ビューの、タイトルバーで、Load Presetアイコンをクリックする。

Load Presetダイアログボックスが開く。



2. Presetリストで、削除するプリセットをクリックする。

ではDeleteボタンが選択可能となる。

3. Deleteをクリックする。

このプリセットが、Presetリストから削除され、プロジェクトからも削除されます。

## 関連トピック

- ・「[プリセットを保存する](#)」
- ・「[プリセットをロードする](#)」

---

## 第47章 コントロールサーフェスの使用

概要 .....	852
コントロールサーフェスデバイスをWwiseに接続する .....	852
Control Surface Sessionを作成する .....	853
コントロールサーフェスのバインディングを理解する .....	853
コントロールサーフェスのバインディングを作成する .....	854
コントロールサーフェスのView Groupsを理解する .....	857
Control Surface Sessionのコンフリクト対応 .....	858
コントロールサーフェスのツールバーの使用 .....	859

## 概要

Wwiseの機能やプロジェクトプロパティを、コントロールサーフェスからコントロールできます。WwiseはMIDIプロトコルやMackieプロトコルに対応しています。また、iOSやAndroidのデバイスで、TouchOSCアプリケーションと合わせてMIDIに対応するTouchOSC Bridgeを使うと、Wwiseでも使えます。

## コントロールサーフェスデバイスをWwiseに接続する

Wwiseは、コントロールサーフェスの以下の2種類のプロトコルに対応しています。

- MIDI Protocol
- Mackie HUI MIDI Mapping Protocol (MCU Pro)

### 始める前に確認すること

- デバイスがコンピュータに物理的に接続されていること。
- デバイスの電源がONであること。
- デバイス用の正しいドライバがインストールされていること。



#### 注記

WwiseはOpen Sound Control (OSC) プロトコルにネイティブ対応していません。ただし、TouchOSCなどのMIDI対応アプリケーションをTouchOSC Bridgeなどと合わせて使えます。

### デバイスをWwiseに接続するには:

1. Projectメニューで、Control Surface Devicesを選択する。
2. Addボタンをクリックする。
3. デバイスに名前をつける。
4. OKをクリックする。

デバイスがリストに追加される。

5. Receive From列で、MIDI INに使うデバイスを選択する。

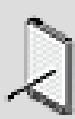
Connectedというメッセージが表示される。

6. Send To列で、MIDI OUTに使うデバイスを選択する。

Connectedというメッセージが表示される。

7. Closeをクリックする。

デバイスを使う準備が完了する。



#### 注記

コントロールサーフェスのデバイス設定はコンピュータにローカル保存されます。

## Control Surface Sessionを作成する

Control Surface Sessionで、ハードウェアのコントロールをWwiseの機能やプロジェクトプロパティにどのように結び付けるかを定義します。Control Surface Sessionで、バインディングのリストを定義します。1つのバインディングで、1つのハードウェアコントロール（ボタン、スライダ、ノブ、キーなど）と、1つのWwiseエレメント（プロパティやコマンド）を結び付けます。



### 注記

Control Surface Sessionはプロジェクト内に保存され、プロジェクトを使用中のどのコンピュータでも利用できます。

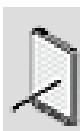
### Control Surface Sessionを作成するには:

1. Viewメニューで、Control Surface Bindingsビューを開く(Ctrl+Shift+Q)。
2. [>>]ボタンをクリックしてセレクタメニューを開く。
3. New...を選択する。
4. セッション名を入力する。
5. OKを押す。

セッションが作成されてロードされる。

1つのWwiseプロジェクトで、以下のように目的に合わせて複数のControl Surface Sessionを作成できる。

- Wwiseプロジェクトの複数のユーザが、それぞれ異なるハードウェアデバイスを使用するとき。
- 使用シナリオが異なるとき。
- ユーザによって、使用目的や好みが異なるとき。



### 注記

1つのプロジェクトで複数のセッションを作成できますが、一度にアクティブにできるセッションは1つだけです。

## コントロールサーフェスのバインディングを理解する

Control Surfaceのバインディングによって、1つのハードウェアコントロール（ボタン、スライダ、ノブ、キーなど）と、1つのWwiseエレメント（プロパティやコマンド）が結び付けられます。

バインディングには、必ず以下の3つのエレメントが含まれます。

- Property/Command:
  - オブジェクトプロパティ: 対象オブジェクトの調整するプロパティの名前。
  - オブジェクトコマンド: 対象オブジェクトに対して実行するコマンドまたはアクション。

- **グローバルコマンド:** グローバルに（特定オブジェクトに限定せず）実行するコマンドまたはアクション。
- **Object/Index:** 対象オブジェクトを指定する。
- **Controller Assignment:** ハードウェアのコントロール部分の、MIDIメッセージID。

バインディングは、対象オブジェクトを定義する方式がそれぞれ異なる、以下の3種類のグループに分けて保存されます。

- **Global:** 対象オブジェクトを、バインディング内で直接、指定する。
- **Current Selection:** 対象オブジェクトは、Wwiseで最後に選択したオブジェクトとなる。
- **View Groups:** ターゲットオブジェクトは、バインディンググループがロードされているビュー（Mixing Desk や Mixing Session、Soundcasterや Soundcaster Session）によって定義される。ビューにロードされたオブジェクトごとに、インデックスが指定される。

## 関連トピック

- 「キーボードショートカット用のバインディングを作成する」
- 「特定オブジェクトのプロパティ値を調整するバインディングを作成する」
- 「現在の選択を調整するバインディングを作成する」

## コントロールサーフェスのバインディングを作成する

バインディングは、以下の通り複数のシナリオ用に作成できます。

- 「キーボードショートカット用のバインディングを作成する」
- 「現在の選択を調整するバインディングを作成する」
- 「特定オブジェクトのプロパティ値を調整するバインディングを作成する」

### キーボードショートカット用のバインディングを作成する

Keyboard Shortcutマネージャからアクセスできるグローバルコマンドを、グローバルバインディングを使ってトリガーすることもできます。グローバルコマンドを、Keyboard ShortcutマネージャとControl Surface Sessionのバインディングの両方でトリガーできます。

Control Surface Bindingsビューでグローバルバインディングを作成するには：

1. Globalグループを選択する。
2. Add & Learn Bindingボタンをクリックする。

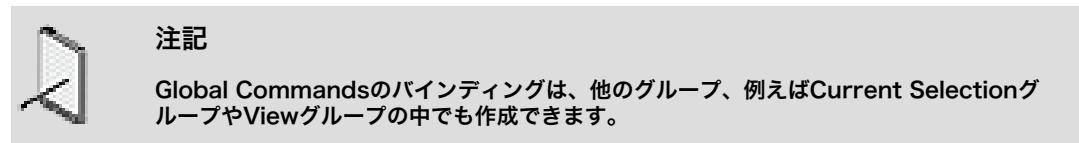
バインディングのエントリーが追加され、Learnボタンが有効になる。

UIで選択できるProperties/Commandsの表示が、緑になる。

3. [>>]ボタンをクリックして、Property/Commandを選択する。
4. メニューの**Global commands...**を選択する。
5. 実行するグローバルコマンドまで移動する。
6. OKをクリックする。
7. バインディングに利用したいハードウェアコントロールを使う。

Learnボタンが無効になる。

バインディングが作成され、利用できるようになる。



### 特定オブジェクトのプロパティ値を調整するバインディングを作成する

プロジェクト内の特定オブジェクトのプロパティ値を対象とするコントロールサーフェスのバインディングを作成できます。コントローラをGame Parameterなどのオブジェクトにアサインして結び付けるのに便利です。

### Game Parameterのシミュレーション値をコントロールするバインディングを作成するには:

1. Globalグループを選択する。
2. Add & Learn Bindingボタンをクリックする。

バインディングのエントリーが追加され、Learnボタンが有効になる。

UIで選択できるProperties/Commandsの表示が、緑になる。

3. [>>]セレクタをクリックして、Property/Commandを選択する。
  - Transport Controlで、Game ParameterのSimulation Valueをクリックする。
  - [>>]ボタンをクリックして、選択したいプロパティまで移動する:
    - メニューの**Object Properties...**を選択する。
    - Browse for Games Syncs > Game Parameter > Simulation Value.
    - OKをクリックする。
4. バインディングに利用したいハードウェアコントロールを使う。

Learnボタンが無効になる。

バインディングが作成され、利用できるようになる。

### 現在の選択を調整するバインディングを作成する

Control Surface Session で、現在選択中のオブジェクトのプロパティ値をハードウェアコントローラのコントロール（スライダ、ノブ、ボタン）に結び付けるバインディングを設定できます。

コントロールサーフェスの4つのスライダを以下にアサインするのが、典型的な設定です。

- Voice Volume.
- Voice Pitch.
- Voice Low-pass Filter.
- Voice High-pass Filter.

このバインディングを作成すると、選択中のオブジェクトのこれらのプロパティに、4つのスライダが自動的に結び付けられます。モーターフェーダ（Mackie Control Universal Proなど）を使う場合は、現在の選択対象（Current Selection）が変わると、同時にフェーダが現在の数値に自動的に移動します。

**選択中のオブジェクトのボイスボリュームを調整するバインディングを作成するには:**

1. Current Selectionグループを選択する。
2. Add & Learn Bindingボタンをクリックする。

バインディングのエントリーが追加され、Learnボタンが有効になる。

UIで選択できるProperties/Commandsの表示が、緑になる。

3. [>>]セレクタをクリックして、Property/Commandを選択する。

- Property Editorで、Voice Volumeをクリックする。
- [>>]ボタンをクリックして、選択したいプロパティまで移動する:
  - メニューのObject Properties...を選択する。
  - Browse for Audio > General Settings > Voice Volume.
  - OKをクリックする。

4. バインディングに利用したいハードウェアコントロールを使う。

Learnボタンが無効になる。

バインディングが作成され、利用できるようになる。

Current Selectionバインディングを使って、選択中のオブジェクトに対するコマンドをトリガーさせることもできます。例えば、いくつかのボタンを以下に割り当てることができます。

- Mute
- Solo
- Play
- Stop

**選択中のオブジェクトをミュートするバインディングを作成するには:**

1. Current Selectionグループを選択する。

## 2. Add & Learn Bindingボタンをクリックする。

バインディングのエントリーが追加され、Learnボタンが有効になる。

UIで選択できるProperties/Commandsの表示が、緑になる。

## 3. [>>]セレクタをクリックして、Property/Commandを選択する。

- [>>]ボタンをクリックする。

- メニューのObject commands > Muteを選択する。

## 4. バインディングに利用したいハードウェアコントロールを使う。

Learnボタンが無効になる。

バインディングが作成され、利用できるようになる。

コントローラのボタンが点灯する場合は、ミュート中にボタンが点灯して、ミュート中でないときは消える。

## コントロールサーフェスのView Groupsを理解する

View Groupsを使って、あるビューの内容（オブジェクト）をバインディンググループの内容にバインディングできます。例えば、Mixing Deskのオブジェクトを、特定のコントロールサーフェスに割り当てられます。

View Groupにビューのオブジェクトに関連したバインディングが含まれ、ビューのインデックスを使って表示されます。

View Groupsに対応しているビューと、インデックスへのアサイン方法を、以下に示します。

- Mixing Desk: 縦ストリップ1本に対して、インデックス1つ。
- List View、Query Editor、Reference View、Master-Mixer Console: 1列に対して、インデックス1つ。
- Property Editor、Effect Editor、Source Editor、Modulator Editor: 現在のオブジェクトに対して、インデックス1のみ。
- Soundcaster: モジュール1つに対して、インデックス1つ。

## View Groupを作成するには:

1. Control Surface Bindingsビューで、View Groupsフォルダをクリックする。
2. Add Groupボタンをクリックする。
3. グループに名前をつける。
4. OKを押す。
5. Add Bindingボタンを使って、グループのバインディングを作成する。

詳しくは「[コントロールサーフェスのバインディングを作成する](#)」を参照してください。

Mixing Desk、List View、Soundcasterなどで使えるコントロールサーフェスのView Groupの例を、以下に示します。

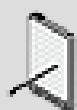
- **Mixing View Group:**

- Binding: Property:Voice Volume - Index:1 - Key:MyController Ch.1 CC 0
- Binding: Property:Voice Volume - Index:2 - Key:MyController Ch.1 CC 1
- Binding: Property:Voice Volume - Index:3 - Key:MyController Ch.1 CC 2
- Binding: Property:Voice Volume - Index:4 - Key:MyController Ch.1 CC 3
- Binding: Command:Solo - Index:1 - Key:MyController Ch.1 CC 32
- Binding: Command:Solo - Index:2 - Key:MyController Ch.1 CC 33
- Binding: Command:Solo - Index:3 - Key:MyController Ch.1 CC 34
- Binding: Command:Solo - Index:4 - Key:MyController Ch.1 CC 35
- Binding: Command:Mute - Index:1 - Key:MyController Ch.1 CC 64
- Binding: Command:Mute - Index:2 - Key:MyController Ch.1 CC 65
- Binding: Command:Mute - Index:3 - Key:MyController Ch.1 CC 66
- Binding: Command:Mute - Index:4 - Key:MyController Ch.1 CC 67

このView Groupによって、以下の割り当てが設定されます。

- ハードウェアのスライダ4つを、4つのオブジェクトのVoice Volumeに割り当てる。
- ハードウェアのボタン4つを、4つのオブジェクトのSoloに割り当てる。
- ハードウェアのボタン4つを、4つのオブジェクトのMuteに割り当てる。

### View GroupをMixing Deskに関連付けるには:

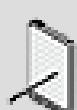


#### 注記

以下の手順を行うには、Control Surface Sessionを1つ起動して、View Groupを1つ以上作成しておく必要があります。

1. Viewメニューで、Mixing Deskビューを開く(Ctrl+Shift+M)。
2. Mixing Deskの右上のタイトルバーにある[>>]ボタンをクリックする。
3. セレクタメニューで、使うView Groupを選択する。

View Groupがロードされる。

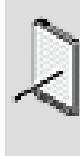


#### 注記

ビューにロードされたView Groupは、Wwiseツールバーにも表示されます。ツールバーでView Groupのオン・オフができます。

## Control Surface Sessionのコンフリクト対応

複数のバインディングが起動している場合、それぞれのコントローラ割り当てがコンフリクトを起こすことがあります。コントロールサーフェスのシステムで、同じキーのバインディングを複数、同時にロードすることはできません。



## 注記

Control Surface Sessionのバインディングは、上から下へ、順にロードされます。コンフリクトが発生すると、コンフリクトするバインディングのうち、最初のバインディングだけがロードされます。最初にあるバインディングが優先されます。

コンフリクトがあると、以下が表示されます。

- Wwiseツールバーの、グループ名の横に黄色の三角マークを表示。
- Control Surface Sessionビューの、バインディングの横に黄色のメッセージを表示。

コンフリクトの対処方法は、以下の通りいくつかあります。

- Control Surface Sessionビューでバインディングを並び替えて、優先順位を変更する。
- バインディング項目のコンテキストメニューを使い、コンフリクトを解消する。
- コンフリクトを起こしているバインディングの1つに、別のコントローラをアサインする。



## Tip

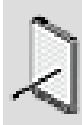
1つのコントローラ割り当てに、同じグループの複数のバインディングをアサインすると便利な場合もあります。例えば、様々なエフェクトのプロパティをバインディングするEffect View Groupを作成することができます。Effect Editorは同時に1つのエフェクトしかロードできないため、バインディング同士のコンフリクトは発生しません。

## コントロールサーフェスのツールバーの使用

コントロールサーフェスのツールバーに表示される項目を、以下に示します。

項目	内容
戻るボタン	直前に選択したオブジェクトに移動する(Backspace)。
進むボタン	リストで次の項目に移動する(Shift+Backspace)。
現在選択中のオブジェクト名	現在選択中のオブジェクトを表示する。
Last Property Changed	Control Surfaceバインディングで変更された最後のオブジェクト、プロパティ、および値(スライダ)を表示します。
ロード中のバインディンググループ	これらのボタンは、アクティブ/非アクティブの <a href="#">グループ</a> を示します： <ul style="list-style-type: none"> <li>Global : ターゲットオブジェクトはバインディングで定義されます。</li> <li>Current Selection : ターゲットオブジェクトは最後に選択されたオブジェクトです。</li> <li>View Group : ターゲットオブジェクトは、指定されたビューを介して指定されたグループにロードされます。たとえば、ボタンにYourGroup:Soundcasterと書かれている場合、プロジェクトには "YourGroup" という定義されたビューグループがあり、Soundcaster内での使用が指定されています。複数のView Groupが異なるビューで定義されている可能性があるため、複数のボタンがある可能性があります。</li> </ul>

項目	内容
	<p>グループ名をクリックすると、グループのアクティブ状態が切り替わる。</p> <ul style="list-style-type: none"><li>• アクティブ：ブルー</li><li>• 非アクティブ：グレイ</li></ul>



### 注記

コントロールサーフェスのツールバーは、Control Surface BindingsビューにControl Surface Sessionがロードされている場合のみ、表示されます。

**audio**kinetic

## パート VIII. 付録



---

A. ダウンミキシングの動作 .....	863
標準的なコンフィギュレーション .....	864
ダウンミックステーブル .....	865
B. 正規表現クイックレフェレンスガイド .....	869
C. ショートカット .....	871
ズームできるEditor画面 .....	872
Attenuation Editor .....	872
オーディオファイルの管理 .....	872
Contents Editor .....	873
Game Object 3D Viewer .....	873
Game Profiler .....	874
グローバル（全体） - 起動中のビューや選択中のオブジェクト .....	874
Music Segment Editor .....	874
Position Editor (User-defined)画面 .....	875
Project Explorer .....	875
RTPC Graph View .....	875
Schematic View .....	875
Soundcaster .....	875
Transport Control .....	876

---

## 付録A ダウンミキシングの動作

### 目次

標準的なコンフィギュレーション .....	864
ダウンミックステーブル .....	865

ボイスやBus（インプット側）を、パンニングやポジショニングがなくチャンネルコンフィギュレーションの異なるバスにルーティングする場合は、Wwiseが以下に説明するルールに基づいてダウンミックスします。

## 標準的なコンフィギュレーション

標準的なコンフィギュレーションは典型的なコンフィギュレーションを指し、ステレオや5.1などチャンネル名から各チャンネルに対応するスピーカーのおおよその配置が分かるものです。例えば"Height Back Left"はリスナーの左後方の地面より上に配置されます。

### LFE

**LFE** (.1) は絶対にダウンミックスしません。インプットとアウトプットの両コンフィギュレーションでLFEチャンネルがある場合は、インプットのLFEチャンネルの信号をアウトプットにコピーします。アウトプットにLFEチャンネルがなければ、インプットのLFEチャンネルの信号をドロップします。

### サラウンドとサイドの違い

4.xや5.Xのコンフィギュレーションでは、フロント以外のスピーカーペアは1つしかないので、サイドチャンネルとバックチャンネルの区別がありません。両者ともサラウンドを意味します。このようなコンフィギュレーションではサラウンドの頭文字のSを使いサラウンドチャンネルをSLやSRと呼び、一貫性を保ちます。また、やや偶然の結果として4.xや5.XのコンフィギュレーションのサラウンドチャンネルであるSLとSRは、バックチャンネルのBLやBRではなく6.Xや7.XのサイドチャンネルのSLやSRにそれぞれマッピングされます。5.1のサラウンドスピーカーのDolby推奨の角度と7.1のサイドスピーカーの角度がマッチするので不都合はありません。

詳細については <http://www.dolby.com/us/en/guide/surround-sound-speaker-setup/5-1-setup.html> と <http://www.dolby.com/jp/ja/guide/surround-sound-speaker-setup/7-1-setup.html> を参照してください。



#### 注記

4.xや5.xコンフィギュレーションのサラウンドチャンネルをWwiseの設定で6.xや7.xコンフィギュレーションのバックチャンネルにルーティングする方法は、現時点ではありません。

### ノーマライゼーション

Wwiseのダウンミックスでノーマライゼーションを行いませんが、これはノーマライゼーション係数の予測しにくい動作を考慮したためです。このためチャンネル数が多いコンフィギュレーションからチャンネル数の少ないコンフィギュレーションにダウンミキシングすると、ラウドネスゲインが発生することが珍しくありません。BusボリュームのStateやRTPCを活用して、デザイナー好みに合うノーマライゼーションを行うようにしてください。

### ハイトチャンネルのあるコンフィギュレーション

ハイトチャンネル（HXX）のあるコンフィギュレーション用のダウンミックステーブルは、現時点ではありません。ハイトチャンネルはLFEチャンネルと同様に

扱われます。つまり、アウトプット側で存在すれば直接マッピングされ、それ以外の場合はドロップされます。

例えば7.0.4インプット (FL-FR-C-SL-SR-BL-BR-HFL-HFR-HBL-HBR) を5.0.2アウトプット (FL-FR-C-SL-SR-HFL-HFR) にルーティングした場合は、以下のように処理されます:

- To 5.xテーブルの通りにインプットのBLとBRがアウトプットのSLとSRにダウンミックスされます。
- インプットのHFLとHFRがアウトプットのHFLとHFRにパスされます。
- インプットのHBLとHBRはアウトプットで存在しないのでドロップされます。

### その他のコンフィギュレーションタイプ

AnonymousやAmbisonicsのコンフィギュレーションはダウンミックスを行いません。ルーティング動作は標準コンフィギュレーションのLFEチャンネルと同じです。もし対応するチャンネルがアウトプット側にも存在すればコピーされ、それ以外の場合はドロップされます。

## ダウンミックステーブル

ダウンミックスの式はAC-3標準に基づいています。それぞれの式は、信号を直接ダウンミックスした場合（例えば7.1からステレオ）でも数段階に分けた場合（例えば7.1から5.1からステレオ）でも同じ結果になるように設定されています。

### ダウンミックステーブルの読み方

以下のテーブルは5.1インプットをステレオアウトプットにルーティングする時に使うダウンミックス式を示しています。

Input Channels	L	R
FL	1	0
FR	0	1
C	$1/\sqrt{2}$	$1/\sqrt{2}$
SL	$1/\sqrt{2}$	0
SR	0	$1/\sqrt{2}$

最終的なステレオシグナルの左右チャンネルは以下の式のインプットのウェイト付けした合計です:

$$\begin{aligned} L_{OUT} &= FL + 1/\sqrt{2} * C + 1/\sqrt{2} * SL \\ R_{OUT} &= FR + 1/\sqrt{2} * C + 1/\sqrt{2} * SR \end{aligned}$$

### レファレンステーブル

上記の公式と同様に、ダウンミックスした各フォーマットのアウトプットシグナルは以下のレファレンスダウンミックス表の各アウトプットチャンネル列の合計で計算できます。

- To Mono

- To Stereo
- To 3.x
- To 4.x
- To 5.x
- To 6.x
- To 7.x

### To Mono

Input Channels	C
FL	$1/\sqrt{2}$
FR	$1/\sqrt{2}$
C	1
SL	$1/2$
SR	$1/2$
BL	$1/2$
BR	$1/2$
Top	1
HFL	$1/2$
HFC	$1/\sqrt{2}$
HFR	$1/2$
HBL	$1/(2\sqrt{2})$
HBR	$1/(2\sqrt{2})$

### To Stereo

Input Channels	L	R
FL	1	0
FR	0	1
C	$1/\sqrt{2}$	$1/\sqrt{2}$
SL	$1/\sqrt{2}$	0
SR	0	$1/\sqrt{2}$
BL	$1/\sqrt{2}$	0
BR	0	$1/\sqrt{2}$
Top	$1/\sqrt{2}$	$1/\sqrt{2}$
HFL	$1/\sqrt{2}$	0
HFC	$1/2$	$1/2$
HFR	0	$1/\sqrt{2}$
HBL	$1/2$	0
HBR	0	$1/2$

### To 3.x

Input Channels	L	R	C
FL	1	0	0
FR	0	1	0

Input Channels	L	R	C
C	0	0	1
SL	$1/\sqrt{2}$	0	0
SR	0	$1/\sqrt{2}$	0
BL	$1/\sqrt{2}$	0	0
BR	0	$1/\sqrt{2}$	0
Top	$1/\sqrt{3}$	$1/\sqrt{3}$	$1/\sqrt{3}$
HFL	$1/\sqrt{2}$	0	0
HFC	0	0	$1/\sqrt{2}$
HFR	0	$1/\sqrt{2}$	0
HBL	$1/2$	0	0
HBR	0	$1/2$	0

### To 4.x

Input Channels	L	R	SL	SR
FL	1	0	0	0
FR	0	1	0	0
C	$1/\sqrt{2}$	$1/\sqrt{2}$	0	0
SL	0	0	1	0
SR	0	0	0	1
BL	0	0	1	0
BR	0	0	0	1
Top	$1/2$	$1/2$	$1/2$	$1/2$
HFL	$1/\sqrt{2}$	0	0	0
HFC	$1/2$	$1/2$	0	0
HFR	0	$1/\sqrt{2}$	0	0
HBL	0	0	$1/\sqrt{2}$	0
HBR	0	0	0	$1/\sqrt{2}$

### To 5.x

Input Channels	L	R	C	SL	SR
FL	1	0	0	0	0
FR	0	1	0	0	0
C	0	0	1	0	0
SL	0	0	0	1	0
SR	0	0	0	0	1
BL	0	0	0	1	0
BR	0	0	0	0	1
Top	$1/\sqrt{5}$	$1/\sqrt{5}$	$1/\sqrt{5}$	$1/\sqrt{5}$	$1/\sqrt{5}$
HFL	$1/\sqrt{2}$	0	0	0	0
HFC	0	0	$1/\sqrt{2}$	0	0
HFR	0	$1/\sqrt{2}$	0	0	0
HBL	0	0	0	$1/\sqrt{2}$	0
HBR	0	0	0	0	$1/\sqrt{2}$

### To 6.x

Input Channels	L	R	SL	SR	BL	BR
FL	1	0	0	0	0	0
FR	0	1	0	0	0	0
C	$1/\sqrt{2}$	$1/\sqrt{2}$	0	0	0	0
SL	0	0	1	0	0	0
SR	0	0	0	1	0	0
BL	0	0	0	0	1	0
BR	0	0	0	0	0	1
Top	$1/\sqrt{6}$	$1/\sqrt{6}$	$1/\sqrt{6}$	$1/\sqrt{6}$	$1/\sqrt{6}$	$1/\sqrt{6}$
HFL	$1/\sqrt{2}$	0	0	0	0	0
HFC	$1/2$	$1/2$	0	0	0	0
HFR	0	$1/\sqrt{2}$	0	0	0	0
HBL	0	0	0	0	$1/\sqrt{2}$	0
HBR	0	0	0	0	0	$1/\sqrt{2}$

### To 7.x

Input Channels	L	R	C	SL	SR	BL	BR
FL	1	0	0	0	0	0	0
FR	0	1	0	0	0	0	0
C	0	0	1	0	0	0	0
SL	0	0	0	1	0	0	0
SR	0	0	0	0	1	0	0
BL	0	0	0	0	0	1	0
BR	0	0	0	0	0	0	1
Top	$1/\sqrt{7}$						
HFL	$1/\sqrt{2}$	0	0	0	0	0	0
HFC	0	0	$1/\sqrt{2}$	0	0	0	0
HFR	0	$1/\sqrt{2}$	0	0	0	0	0
HBL	0	0	0	0	0	$1/\sqrt{2}$	0
HBR	0	0	0	0	0	0	$1/\sqrt{2}$

## 付録B 正規表現クイックレファレンスガイド

Anchors	
^	文字列の行頭、または複数行パターンの行頭
\$	文字列の行末、または複数行パターンの行末
\b	ワードの区切り
\B	ワードの区切り以外

Character Classes (文字クラス)	
\s	空白文字
\S	空白文字以外
\d	数字
\D	数字以外
\w	ワード
\W	ワード以外
\x	16進数の数字 16進数コード文字のマッチに使い、例えば • 「\xA9」でコピーライト印の「0xA9」 • や、「\x5B」でオープン角括弧の「0x5B」など。

Quantifiers (量指定子)	
*	空白文字
+	空白文字以外
?	数字
{3}	ちょうど3
{3,}	3以上
{3,5}	3、4、または5

Escape Sequences	
\	続く文字をエスケープする ^ [ . \$ { * ( \ + )   ? < >

Special Characters (特別文字)	
\n	改行
\r	キャリッジリターン
\t	タブ

Groups and Ranges (グループと範囲)	
.	改行(\n)以外の文字
\r	キャリッジリターン
(a b)	aまたはb

# 正規表現クリックレファレンス ガイド

Groups and Ranges (グループと範囲)	
(...)	グループ
(?:...)	パッシブ (キャプチャしない) グループ
[abc]	範囲 (aまたはbまたはc)
[^abc]	(aまたはbまたはc) 以外
[a-q]	aからqまでの小文字
[A-Q]	AからQまでの大文字
[0-7]	0から7までの数字

String Replacement (文字列の置換)	
\$n	n番目の非パッシブグループ
\$2	/^(abc(xyz))\$/ の中の「xyz」
\$1	/^(?:abc)(xyz)\$/ の中の「xyz」
\$`	マッチする文字列の前
\$'	マッチする文字列の後
\$+	最後にマッチする文字列
\$&	マッチする文字列全体

---

## 付録C ショートカット

### 目次

ズームできるEditor画面 .....	872
Attenuation Editor .....	872
オーディオファイルの管理 .....	872
Contents Editor .....	873
Game Object 3D Viewer .....	873
Game Profiler .....	874
グローバル（全体） - 起動中のビューや選択中のオブジェクト .....	874
Music Segment Editor .....	874
Position Editor (User-defined)画面 .....	875
Project Explorer .....	875
RTPC Graph View .....	875
Schematic View .....	875
Soundcaster .....	875
Transport Control .....	876

Wwiseで使う多くのコマンドや操作に、1つのキー、または複数のキーの組み合わせがショートカットキーとして設定されています。これらのアクションやコマンドを実行する時に、マウス操作の代わりに使えます。

ここで紹介するコマンドの多くは、Keyboard Shortcut Managerを使って再マッピングできます。

## ズームできるEditor画面

To	ショートカット
ズームイン	Z+[マーキー選択]
現在のマウス位置を中心に、横にズームイン	Ctrl+[マウスホイールを上に]
現在のマウス位置を中心に、横にズームアウト	Ctrl+[マウスホイールを下に]
現在のマウス位置を中心に、縦にズームイン	Ctrl+Shift+[マウスホイールを上に]
現在のマウス位置を中心に、縦にズームアウト	Ctrl+Shift+[マウスホイールを下に]
ズームイン中に、パンをリセットしてズーム	Z+[クリック]
ズームイン中に、上にパンニング	マウスホイールを上に
ズームイン中に、下にパンニング	マウスホイールを下に
ズームイン中に、左にパンニング	Shift+[マウスホイールを上に]
ズームイン中に、右にパンニング	Shift+[マウスホイールを下に]
ズームイン中に、フリーハンドでパンニング	X+[ドラッグ]
ズームイン中に、パンをリセット	X+[クリック]
ポイント挿入	ダブルクリック
選択したポイントを移動	矢印
ポイントを細かく移動	Shift+[ ドラッグ]
連続していない複数のポイントを選択	Ctrl+[クリック]
全てのポイントを選択	Ctrl+A
次のポイントに移動して選択	タブ
前のポイントに移動して選択	Shift+Tab
選択をX軸またはY軸にロック	Alt+[ ドラッグ]

## Attenuation Editor

「ズームできるEditor画面」を参照してください。

## オーディオファイルの管理



To	ショートカット
新規Sound Voiceを作成	Shift+[ ドラッグ]
Audio File Importerを開かず、SFXやミュージックの、オーディオファイルをインポート	Ctrl+[ ドラッグ]
Audio File Importerを開かず、サウンドボイスのオーディオファイルをインポート	Ctrl+Shift+[ ドラッグ]

## ショートカット

---

To	ショートカット
サウンドSFXやミュージックの、ソースを強制的に置換	Alt+[ドラッグ]
サウンドボイスを強制的に置換	Shift+Alt+[ドラッグ]

## Contents Editor

To	ショートカット
上・下の選択肢に移動	上・下向き矢印
オブジェクトを選択に追加（上・下）	Shift+[上・下向き矢印]
すぐ右にある設定項目に移動	タブ
すぐ左にある設定項目に移動	Shift+Tab
すぐ下にある設定項目に移動	Ctrl+Tab
すぐ上にある設定項目に移動	Ctrl+Shift+Tab

## Game Object 3D Viewer

以下のショートカットは、Game Object 3D Viewerが起動中の時に限り、利用できます。

To	ショートカット
Game Object 3D Viewer Settingsダイアログボックスを開く	V
カメラ位置をデフォルトにリセット（Reset）	Ctrl+R
キャプチャ中に、選択したゲームオブジェクトをカメラに追わせる（Follow）	Ctrl+F
ビューのフレームに全てのオブジェクトやリスナーを入れる	Ctrl+A
Game Object 3D Viewer Filterダイアログボックスを表示	Alt+F
前に移動（カメラ設定がファーストパーソンや、リスナーの時）	W
後に移動（カメラ設定がファーストパーソンや、リスナーの時）	S
右に移動（カメラ設定がファーストパーソンや、リスナーの時）	A
左に移動（カメラ設定がファーストパーソンや、リスナーの時）	D
ビューアー内の動きを加速	Shift
Camera User 1に切り替える	Ctrl+1
Camera User 2に切り替える	Ctrl+2
Front cameraビューに切り替える	F
Top cameraビューに切り替える	T
First Personビューに切り替える	P
Listener 0ビューに切り替える	0
Listener 1ビューに切り替える	1
Listener 2ビューに切り替える	2
Listener 3ビューに切り替える	3
Listener 4ビューに切り替える	4
Listener 5ビューに切り替える	5
Listener 6ビューに切り替える	6

## ショートカット

---

To	ショートカット
Listener 7ビューに切り替える	7

## Game Profiler

To	ショートカット
Profiler Settingsダイアログボックスを開く	Alt+G
すぐ上の青い円に移動	Ctrl+[上向き矢印]
すぐ下の青い円に移動	Ctrl+[下向き矢印]
タイムカーソルでログエントリを強制的にキャプチャ	Shift+[クリック]

## グローバル（全体） - 起動中のビューや選択中のオブジェクト

To	ショートカット
プロパティ値の設定	
デフォルト値にリセット	Ctrl+[クリック]
スーパースライダを使って微調整	Shift+[ドラッグ]
スーパースライダを使って複数の設定値をオフセット	Alt+[ドラッグ]

## Music Segment Editor

To	Press
セグメントの冒頭にジャンプ	Home
セグメントの終わりにジャンプ	End
最初のトラックにジャンプ	Shift+Home
最後のトラックにジャンプ	Shift+End
左上にジャンプ	Ctrl+Home
右下にジャンプ	Ctrl+End
再生カーソルをエントリキューまで移動	0
再生カーソルをエグジットキューまで移動	1
再生カーソルをカスタムキューまで移動	2~8
再生カーソルをセグメント冒頭まで移動	。(period)
バー、ビートにスナップ	Q
キーにスナップ	W
クリップ、ループにスナップ	E
カーソル位置で分割	S
ズームイン	+
ズームイン（選択）	Z+[マーキー選択]
ズームアウト	-
パン	X+[マウスドラッグ]
再生カーソル位置にキーを挿入	Insert
エントリキュー設定	Ctrl+[ルーラーをクリック]
エグジットキュー設定	Alt+[ルーラーをクリック]
新規カスタムキュー設定	Shift+[ルーラーをクリック]
エントリ・エグジットのキーを選択位置に移動	P

「[ズームできるEditor画面](#)」も参照してください。

## Position Editor (User-defined)画面

Position Editor (User-defined) のショートカットは、以下のセクションに分けられます。

- ・ [「Pathリストで」](#)

パスエディタやタイムラインビューのショートカットについては [「ズームできるEditor画面」](#) を参照してください。

### Pathリストで

To	ショートカット
上下に移動	上・下向き矢印

## Project Explorer

To	ショートカット
ノード展開	右向き矢印
ノード縮小	左向き矢印
上に移動	上向き矢印
下に移動	下向き矢印

## RTPC Graph View

「[ズームできるEditor画面](#)」を参照してください。

## Schematic View

To	ショートカット
選択変更、プロジェクト内の移動、階層の展開・縮小	矢印
兄弟オブジェクトのグループの、最初の子に移動	Home
兄弟オブジェクトのグループの、最後の子に移動	End
Master Audioに移動	Ctrl+Home
選択項目をPropertyEditorで編集	Enter
Playback	[スペースバー]
Searchフィールドに移動	Ctrl+F
プロジェクト階層に移動 (Searchフィールドから)	Enter

## Soundcaster

To	ショートカット
Play - Bypass プロパティ	Shift + クリック Play
全て一時停止・全て再開	Ctrl+Shift+[スペースバー]
Originalモードのトグル	Alt+O

## ショートカット

---

To	ショートカット
Inc. Onlyのトグル	Alt+I
全てリセット	Ctrl+Alt+R

## Transport Control

To	ショートカット
再生・停止	[スペースバー]
Play - Bypass プロパティ	Shift+スペースバー (どこからでも) Shift + クリック Play (Transportで)
一時停止・再開	Ctrl+[スペースバー]
Originalモードのトグル	Alt+O
Inc. Onlyのトグル	Alt+I
ピン付き (Pin) ・ピンなし (Unpin)	Ctrl+Alt+P
Transport Controlにロードしてピン付け	Alt+[クリック]
State (State) のトグル	Alt+S
Switch (sWitch) のトグル	Alt+W
RTPCのトグル	Alt+R
Triggerのトグル	Alt+T
全てリセット	Ctrl+Alt+R

---

# 用語集

この用語集の目的はHelpドキュメンテーションで使われるテクニカル用語を定義することです。デザイン用語やプログラミング用語を定義する[一般用語](#)と、Wwise独自のオブジェクトやコンセプト（audiokineticが特許を有するものも含む）を定義する[Wwise用語](#)に分かれています。

## 一般用語

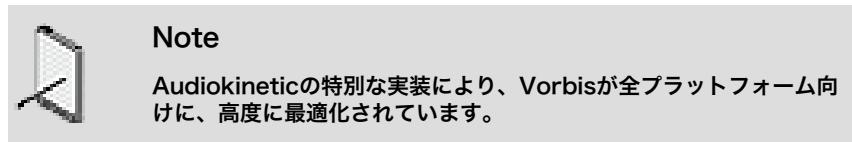
AAC	Mac および iOS プラットフォーム上でのWwiseで使用できる知覚コーディング音声圧縮方式。AACは、同様のビットレートにおいてMP3より優れた音質を実現すると言われている。圧縮は変数・コンテンツ依存で、品質設定は "quality (品質)" スライダーで制御可能。iOSでは、ハードウェア支援のコーデックが使用可能な場合、これによりAACがデコードされる。iOSハードウェアは、一度に一つのAAC 音声のみデコードできることに注意。
AC-3	オーディオコーディングアルゴリズムの圧縮標準。詳しくは <a href="#">Digital Audio Compression Standard</a> の詳細ドキュメンテーションを参照。
ADPCM (Adaptive Differential Pulse Code Modulation)	サウンド信号とそのサウンド信号から作られた予測の差分を量子化する、オーディオファイル変換・符号化（エンコード）方式。ADPCM量子化ステップは適応的なので、信号を直接量子化するPCM（パルス符号変調）符号化とは異なる。基本的にADPCMは音質を犠牲にしてメモリやCPUを非常に効率的に使う。このため、主にモバイルプラットフォームで採用される。
Ambisonics（アンビソニックス）	サラウンドサウンドテクニックの1つで、リスナーの上下リレーションに加えて水平面もカバーする。B-formatサウンドフィールド（音場）表現を使い、スピーカー設定とは独立している。
ADSR (Attack-Decay-Sustain-Release)	サウンドのアタックタイム（Wwiseではカーブも）、ディケイタイム、サステインレベル、リリースタイムの設定値によって定義されるエンベロープの形。
ATRAC9	PS4 向けの知覚符号化方式のことで、非常に優れた知覚音質を維持しながら、さまざまなビットレートでのオーディオファイル符号化を可能にする。
ビット深度	デジタルオーディオファイル内の各サンプルを記述するために使用されるビット数。PCMオーディオでは、ビット深度が信号の最大可能ダイナミックレンジを決定する。
Bit Rate（ビットレート）	1秒間に送受信できるデータ量、特にビット数のこと。通常、ビットレートが大きくなるほど、より多くのファイルデータが処理され音質が向上する。

Cent (セント)	半音の100分の1に等しいピッチの単位。1オクターブは、1200セントより構成される。
dBFS	デシベルフルスケール (Decibels full scale) のことであり、デジタルシステムにおけるデシベル振幅レベルのことで、(PCM符号化のように) 最大使用可能レベルがある。
Delay Line	信号を一定のサンプル数で保留する処理ユニット。
Depot	Perforceのソースコントロールサーバー上にある、ファイルの中央レポジトリのこと。サーバーに送信される全てのファイルの全バージョンを格納する。
Dithering (ディザリング)	量子化に先立って信号に加えられるノイズで、量子化処理に起因する歪みやノイズ変調を軽減する。ノイズレベルが若干増加するが、スペクトル整形ディザは明らかな増加を最小限に抑えることができる。ノイズは歪みより不快感が少なく、低レベルの信号をよりはつきり聞こえるようにする。
Downmix (ダウンミックス)	マルチチャンネルソース内の全チャンネルを、ステレオまたはモノラルなどの少数チャンネルと互換性のあるバージョンにミックスする処理。
Dry Signal (ドライシグナル)	完全に未処理の原音信号から成る出力。
Echo Density (エコーデンシティ)	リバーブ (残響) アルゴリズムによって1秒あたりに生成されるエコー量。
Frequency	単位時間あたりの周波数。
Gain (ゲイン)	信号のパワーまたは振幅の変化。
Harmonics (高調波)	基本周波数の倍音。例えば、周波数が50 Hzの場合、第二高調波は100 Hz、第三高調波は150 Hzとなる。
Headroom (ヘッドルーム)	アンプやオーディオデバイスの通常動作レベルとクリッピングレベルのレベル差 (dB単位)。
HPF	カットオフ周波数よりも低い周波数を減衰させる再帰型フィルタ。このフィルタの単位は適用されたハイパスフィルタリングエフェクトの割合を示し、0はハイパスフィルタリング無効 (信号は影響を受けない)、100は最大減衰を意味する。
HMD (Head Mounted Device)	バーチャルリアリティ向けに開発されたデバイスで、ユーザーの頭の動きに基づいて視覚的や聴覚的なインプットを提供する。
HDR (ハイダイナミックレンジ) オーディオ	自然界で見られるような非常に幅広いダイナミックレンジに渡る様々なレベル値を使って、ミックスするテクニックです。また、HDRは、この幅広いレンジのレベルを、使用するサウンドシ

	システムのデジタルアウトプットに適したレンジにダイナミックにマッピングする、ランタイムシステムでもあります。
HRTF - Head Related Transfer Function (頭部伝達関数)	サウンドが様々な物理的メディアを経由して、空間の点から耳に受け取られるまでの方式。HRTFのペアで、空間の特定ポイントから発生しているようなバイノーラルサウンドの合成が可能となります。
Inharmonicity (インハーモニティ)	部分高調波のピッチと高調波のピッチ間のわずかなオフセット（基本周波数の整数倍）。
Intensity Stereo (インテンシティステレオ)	高周波コンテンツのフェーズアライメント情報を排除してモノシグナルへとマージしつつ、ステレオシグナルを再構築できるようインテンシティ情報を維持する、ロッシーオーディオコーディングテクニック。
Latency (レイテンシ)	コンピュータ内におけるオーディオ信号の内部処理や生成に固有の遅延。
LFE	Low Frequency Effect (低域効果音)のこと。10Hz#120Hzの重低音用オーディオチャンネル名。LFEチャンネルは、x.1メディアファイルとしてWwiseへのインポートが必須の完全に独立したチャンネルである。
Limiting (リミッティング)	入力/出力の関係を非常にフラット (10:1以上) にする極端な圧縮形態。これにより、信号レベルにハードリミットが設定される。
LPF	カットオフ周波数よりも高い周波数を減衰させる再帰型フィルタ。このフィルタの単位は適用されたローパスフィルタリングエフェクトの割合を示し、0はローパスフィルタリング無効 (信号は影響を受けない)、100は最大の減衰を意味する。
Modal density	モードは、オーディオ信号の周波数領域表現におけるピークのこと。モード密度を上げることにより、たいていの音響空間シミュレーションにおいて、リバーブ (残響) のリアリズムを向上することができる。モード密度が低いと共鳴音が生じうる。
Noise Shaping (ノイズシェーピング)	デジタルシグナルのビット深度の削減によって起きるノイズを最小限に抑えるテクニック。
Nyquist Frequency (ナイキスト周波数)	正確にサンプリング可能な最高オーディオ周波数のこと、サンプリング周波数 (サンプリングレート) の半分に相当。ナイキストサンプリング定理は、正確に元の信号を再現するには、サンプリング周波数が、少なくともサンプル中の最高周波数の2倍でなければならないことを示した。
Opus	音声と汎用オーディオの両方に最適化された、低レイテンシのオーディオコーデック。音質を損なうことなく他のコーデックより優れた圧縮率を発揮します。
Outro (アウトロ)	サウンドの最後のセクションのこと、イントロの逆である。

Passband (パスバンド)	基本的に減衰無しでフィルタを通過する周波数帯域。
PCM - Pulse code modulation (パルス符号変調)	別個の2進数表現すなわちパルス符号を選択する、オーディオファイル変換・符号化（エンコード）方式。暗号化された2つのポイント間を測定して、最も近いポイントに関連する数値を選択することにより量子化する。
PCM frame (PCMフレーム)	PCMフレームは、特定の時点における全チャンネルのサンプルで構成されている。各フレームは、1/サンプリング周波数秒。
Pitch	オブジェクトの再生スピード。
Prediction filter (予測フィルタ)	ロッショーオーディオ圧縮テクニックとして使えるスピーチシグナルのモデル化メソッドで、スピーチシグナル動作が予測可能であることに基づいている。
Quantization (量子化)	オーディオファイルの値の範囲をサブ範囲に分割するプロセスのことであり、各サブ範囲は、割り当てられた値によって表される。
Recursive Filter (リカーシブフィルタ)	現在の入力値に加えて、事前計算された出力値を使用するフィルタ。フィードバックフィルタとも呼ばれる。
Ripple (in passband) ((パスバンド) リップル)	パスバンド内の最高減衰と最低減衰間の差。
RMS power (RMSパワー)	二乗平均平方根 (Root Mean Square)。信号の平均振幅の尺度のこと、ほとんどの場合において、ピーク振幅を使用するより優れた近似値を提供する。この値は、所定の時間ウインドウ内サンプルの平方を合計し、その結果の平方根を取ることにより得られる。
Side-Chaining (サイドチェイン)	オーディオ信号レベルをモニターし、このオーディオ情報を、別のオーディオ信号のリアルタイムな制御に使用する。このテクニックは、ファイナルミックスの重要度の低いサウンドの音量を自動的に制御し、重要なサウンドを強調するために使用される。
ソースコントロール	ファイル (Wwiseプロジェクト、ワークユニック、オーディオファイルなど)への変更の管理のこと、ファイルの各リビジョンをリビジョン識別番号でタグ付けし、タイムスタンプ、ファイルの変更者名に関連付ける。ほとんどのバージョン管理システムでは、リビジョンの比較、復元およびマージが可能。
標準的なコンフィギュレーション	独立した標準的なチャンネルで構成されたチャンネルコンフィギュレーションのこと、例えばステレオ、4.0、5.1、7.1、7.1.4など。アンビソニックスは、標準的なコンフィギュレーションではない。
タブ区切りファイル	情報がタブで区切られたカラムに配置されている特別な種類のブレーンテキストファイル。

Tap	信号がDelay Lineから抽出されるポイント。
VBAP (Vector Base Amplitude Panning)	1つまたは複数のバーチャルソースが異なる方向から発生していくような場合の正確な3Dポジショニングを、複数のラウドスピーカーを使って行うための、一般的なアルゴリズム。
Volume (ボリューム)	オーディオアウトプットの、振幅または強度
Vorbis	非常に優れた知覚音質を保ちながら様々なビットレートでのオーディオファイルエンコードを可能にする知覚符号化方式。データ圧縮効率と知覚音質のバランスは、Quality Factor (品質係数) 設定の使用またはチャンネルあたりの最大、最小および平均ビットレート指定により制御される。



**WAVEFORMATEXTENSIBLE** 2つ以上のチャンネルを持つフォーマットの波形（ウェーブフォーム/ waveform）オーディオデータのフォーマット定義をする構造体。WAVファイルの特殊なマルチチャンネル設定を保持するには、WAVヘッダファイルで、チャンネル情報をWAVEFORMATEXTENSIBLEチャンネルマスクの一部として定義する必要がある。

Wet Signal (ウェット信号)	処理済みのサウンドのみから成る出力。
World Builder (ワールドビルダー)	ゲームの仮想環境を作成するために使用されるアプリケーション。
XMA	ハードウェアサポートのXbox One向け知覚符号化オーディオ圧縮方式。XMAは、Windows Media Audio Proのコンソール最適化バージョン。圧縮は変数・コンテンツ依存、品質設定は「圧縮品質 (Compression Quality)」スライダより制御可能。最新版のXMA 2.0は、XMAデータのシークを容易にするシークテーブル作成のためのブロックサイズパラメータを備える。

## Wwise用語

Absolute Property (絶対プロパティ)	例えば、ポジショニングや再生の優先度など、通常最上位の親オブジェクトにて定義され、各親オブジェクトの子オブジェクトへ自動的に継承されるプロパティ。最上位の親プロパティは、階層内の異なるレベルでこれらのプロパティを定義することによりオーバーライド（上書き）可能。
Actor-Mixer	1つ以上のサウンド、モーションオブジェクト、コンテナ、Actor-Mixerなどで構成される階層構造。アクターミキサーによって、その下位にある全てのオブジェクトのプロパティ制御が可能。

Additive Type Property (追加型プロパティ)	RTPCやStateなど、様々な設定で複数回、変更できるWwiseのプロパティ。最終的な値は、適用された全てのプロパティオフセットの合計となる。
オーディオ入力 (Audio Input)	オーディオインプットとは、ゲームにより生成されるオーディオコンテンツをWwiseパイプライン経由で送信し、サウンドエンジンで処理するためのサンプルソースプラグインのこと。
Attenuation	サウンドオブジェクト、音楽オブジェクト、モーションオブジェクトがサウンドエミッタから離れるにつれ、その音量が低下すること。
Attenuation ShareSet	リスナーからの距離に基づく、サウンドのボリュームに関する減衰設定。設定はShareSetとして保存され、オブジェクト間で共有可能。
Audio Bus	オプショナルのバスであり、マスター Audio Busの下に（他の Audio BusやAUX Busと共に）入れて、サウンドミックスを整理したり提供したりするのに活用できる。これらのバスは名前変更、移動、削除が可能であり、エフェクトも適用できる。
Audio Source	オーディオファイルとサウンドオブジェクト間の独立した抽象化レイヤ。プロジェクトにインポートされるオーディオファイルにリンクされている。オーディオソースは、プロジェクトにインポートされるオーディオファイルへのリンク状態を保つので、いつでも参照可能。
Auto-ducking	1つのオーディオ信号の音量レベルを下げて、別の同時オーディオ信号を引き立てること。
Auxiliary Bus	リバーブやディレイなどのエフェクトを適用し、環境効果音のシミュレートやダイナミックミキシング（サイドチェイン）を実現するために通常使用される特殊なタイプのAudio Bus。
Auxiliary Send	Auxiliary Bus（補助バス）にオーディオ信号を送るために使用されるオーディオ信号ルーティングテクニック。SDK のAPI使用時、AUXセンドは、サウンドオブジェクトごと、またはゲームオブジェクトごとに制御可能。
Blend Container	同時に再生されるオブジェクトやコンテナ（単数または複数）のグループ。このコンテナ内のオブジェクトは、RTPC使用による Game Parameter値へのプロパティマッピングが実行されるブレンドトラックにグループ分けが可能。Game Parameter値に基づいて、ブレンドトラック内のオブジェクト間にクロスフェードを適用することも可能。
Cache (キャッシュ)	開発中プラットフォーム向けの全ての変換ファイルを含むプロジェクトフォルダ。デフォルトでは、このフォルダはローカルに保存されるが、保存場所は変更可能。複数ユーザーによるキャッシュフォルダへの同時アクセスは避けること。

Child Object	階層構造内のオブジェクトで、より上位にあるオブジェクトまたは親オブジェクト内に存在。
Clip	オーディオソースを表すミュージックオブジェクト。クリップは Music Track内に整理される。
Compressor	事前定義されたしきい値（スレッショルド）を超える入力信号の任意の部分を弱めることにより、信号のダイナミックレンジを減らすオーディオエフェクトプラグイン。
Container	サウンド、モーションオブジェクト、コンテナなどのオブジェクト（単数または複数）のグループで、定義された特定のビヘイビアに従って再生される。Wwiseに数種類のコンテナがあり、例えば <a href="#">Random Container</a> 、 <a href="#">Sequence Container</a> 、 <a href="#">Switch Container</a> 、 <a href="#">Blend Container</a> 、 <a href="#">Music Switch Container</a> 、 <a href="#">Music Playlist Container</a> などである。
Control Surface Session	保存されたプロジェクトエレメントで、WwiseがMIDIやMackieプロトコルを使って外部コントロールサーフェス機器とどうやり取りするかを定義する。互換性のある接続済み機器のバインディングが設定されている場合は、Control Surface Sessionを開いて機器から直接Wwise機能を制御できる。
Conversion Settings	プラットフォームごとにオーディオファイルの全体的な品質、メモリ、CPU使用率を決定する、サンプルレート、オーディオフォーマット、チャンネル数などオーディオファイルパラメータのグループ。
Convolution Reverb	コンサートホール、建物、道路、車両の室内、部屋、野原、森やその他の実空間の音響をシミュレートするためにIR（インパルス応答）を使用するオーディオエフェクト。
Cue	エントリポイントまたはエグジットポイントなどのキーポイントを示すためにMusic Segmentに付加されるマーカー。
Custom Cue	重要なシンクポイントを示すためにMusic Segmentに付加されるユーザー作成のマーカー。エントリキーおよびエグジットキーはカスタムキーとはみなされない。
dB Scaling	デシベルで測定されたプロパティを表す際に、グラフビューのY軸を対数スケールで表示するオプション。
DC Offset	「直流（Direct Current）」とは、音の波形の中心のこと。オフセットは、波形の中心がある0.0位置からのズレ率のこと。
Default Work Unit	特定のエレメントのために作成され、このエレメントに関連したプロジェクト内全ての情報を含むXMLファイル。例えば、イベント（Events）用Default Work Unit.wwuファイルには、イベントに関連する全ての情報が、State（States）用Default Work Unit.wwu ファイルには、Stateに関連する全ての情報が含まれている。プロジェクトが作成されると、Default Work Unitも作成される。

Definition File	ゲーム内の全てのイベントを一覧表示するテキストファイルのこととで、SoundBank (SoundBank) によって分類される。
Delay	指定した期間のオーディオ信号を遅延させることによりエコーを加えるオーディオエフェクトプラグイン。
Dialogue Event 「ダイアログイベント」	ゲーム中で起りうる様々な条件に一致する引数や引数値で表された一連のルールや条件を使用して、ゲームにオーディオをトリガーする方法。これらの引数値は引数バスに編成され、Wwise内のオブジェクトに割り当てられる。ゲームプレイ中にダイアログイベントが呼び出されると、ゲームは現状とダイアログイベントで定義された条件を照合し、適切なダイアログを再生する。
Effect Chain	特定の順序でオブジェクトまたはバスに適用された一連のエフェクト。
Effect Instance	カスタマイズされた一連のエフェクトプロパティで、保存して他のオブジェクトやバスに適用することができる。エフェクトインスタンスプロパティをオブジェクト間で共有することも可能。
Effect ShareSet	ゲームのオーディオを強化するために使用できるオーディオエフェクト設定。設定はShareSetとして保存され、オブジェクト間で共有可能。
Empty Event	アクションやオブジェクトを含まないイベント。
Environmental Effect	ゲームオブジェクトに生成されるサウンドのプロパティセットを、そのオブジェクトのゲーム中ジオメトリにおける位置に応じて変更するエフェクト。
Event (Action Event)	Wwiseオブジェクト（単数または複数）に適用され、Play、Mute、Pauseなどのアクションまたは一連のアクションを使用してゲーム内にオーディオをトリガーする方法。
Expander	Wwise Expanderプラグインエフェクトは、事前定義されたしきい値（スレッショルド）を下回る入力信号の任意の部分を弱めることにより、信号のダイナミックレンジを拡張する。信号が穏やかでしきい値以下である場合、Expanderはゲインリダクション（ゲイン低下）を開始。信号がしきい値と等しい場合や、しきい値より大きい場合、信号にゲインリダンクションは適用されない。
External Source	ランタイム時にサウンドオブジェクトをオーディオファイルに関連付けるソースプラグイン。本来であれば多大なオーバーヘッドが必要となるような、膨大な数のダイアログラインの管理に役立つ。ランタイムメモリをいくらか節約するのに役立ち、ゲームのDLCコンテンツ生成の際に要求されることがあるオーディオファイルの置換を簡素化する。
File Managerを使う	プロジェクトファイルとインポートされたオリジナルソースファイルに関する情報を表示し、必要に応じてソース管理プラグイン機能を制御するダイアログボックス。

Flanger	2つの同一信号をミックスするオーディオエフェクトのこと。元の信号に、これをわずかに遅延させた信号を遅延の程度を徐々に変化させながら加え、スイープコムフィルタエフェクトを生み出すことができる。
Flat View	レイアウトにドッキングされたビュー。
Focus	スプレッド値によって生じた複数のバーチャルエミッタを集約するために使う値(%)。Focusが0%のときは、バーチャルエミッタに変更はないが、数値が高くなると、それぞれのバーチャルポイントが元のソースチャンネルの周りに近づいていく。
Folder	Actor-Mixer Hierarchyの上位レベルの構造で、ほかのアクターミキサーやコンテナなどの構造を管理するために使用。
Game Object	インターフェース、Triggerまたはサウンドなどのエレメントをattachできるゲーム内のエンティティ。
Game Parameter	例えば、カーレーシングゲームのスピードやRPMなど、RTCPを使用してWwiseプロパティ値にマップできるゲームのパラメータ。
Game Sync	ゲーム内の状況によって呼び出され、それに応じてオーディオやモーションを変化させるState、Switch、RTPCs、Trigger、引数などを含むWwiseエレメントのグループ。
Game Unit	ゲームジオメトリの計算に使用される長さの基本単位。例えば、ステルスFPSで使用されるゲームユニットは「メートル」であるが、宇宙征服ゲームの場合は「光年」など。
Guitar Distortion	オーディオ波形の形状を変更し、元の信号に存在しない周波数成分を導入するオーディオエフェクト。WwiseのGuitar Distortionエフェクトは、典型的なギターのディストーションサウンドを得るために、一般的に使用される「ストンプボックス」ディストーションのビヘイビアを模倣。
Harmonizer (ハーモナイザー)	入力信号にワンピッチまたはツーピッチ高いボイスを追加するオーディオエフェクト。
Imported folder	プロジェクトの隠しキャッシュフォルダのことで、特別なインポート変換プロセスを経てプロジェクトにインポートされたオーディオファイルのコピーを格納している。
Indicator	特定のプロパティ値ステータスを示すWwiseインターフェース固有のアイコン。例えば、RTCPインジケータはプロパティ値にRTCPが関連付けられているかどうかを示す。
Initialization Bank	バス階層、State、SwitchやRTCPsに関する情報を含むプロジェクトのあらゆる一般情報を格納する特殊なタイプのSoundBank。プロジェクトごとに1つのみの初期化バンクがあり、デフォルトでは“Init.bnk”と言う名前が付けられている。

	ゲームの起動時、初期化バンクが最初にロードされる必要がある。最初にロードされないと、次のSoundBankをロードできないことがある。
Input Range	プロパティに入力可能な値の全範囲であり、Slider Rangeと対照的。
Integrity Report	Wwiseが生成するレポートことで、エラーや問題を対応策とともに表示する。
Interactive Music	ゲーム内アクションに対してモジュール性および応答性のあるミュージックスコア作成のための作曲および編曲法。
Invalid Events	プロジェクトから削除されているかSoundBank (SoundBank) に格納されているイベント。
IR (Impulse Response)	例えばコンサートホールなど、ある場所の実際の音響特性を測定することにより得られるオーディオファイル。インパルス反応は、特定の場所の音響特性を入力信号へ適用できるよう、コンボリューションリバーブ (Convolution Reverb) エフェクトで使用されている。
Layout	特定のタスクやジョブに関連する作業を容易にするためにグループにまとめられた一連のビュー。
Listener	ゲーム内にある仮想マイクやモーションセンサーのことで、3D環境をシミュレーションするために特定スピーカーにサウンドを割り当てたり、特定モータにモーションを割り当てたりするのに役立つ。
Look-ahead time	ストリーミング分野において、サウンドエンジンがストリーミングデータをシークするために予約される時間のこと。
Master-Mixer Hierarchy	プロジェクト階層の最上部にあるバスまたは一連のバスで、これにより、多数の異なるサウンド、ミュージックやモーションの構造を、ゲーム内のメインカテゴリーに従って再グループ化が可能。例えば、全てのボイス構造を1つのAudio Bus下位に、全てのサウンドエフェクトを別のAudio Bus下位にグループ分けできる。
マスターAudio Bus	マスターAudio Busは最上位にあり、Master Mixer Hierarchy のDefault Work Unitにある。最終的なオーディオアウトプットは、時として膨大なサブバスのネットワークを経由して、最後にMaster Audio Busやそれに適用されるEffectの設定によって決まる。
マスターセカンダリバス	マスターセカンダリバスとは、マスター・ミキサー階層のデフォルトワークユニットの下で一番上のレベルにあるバスで、その最終アウトプットがセカンダリであるものを示し、ゲームコントローラなどはその例である。

Matrix Reverb	ゲーム制作向けに最適化されたユニークなリバーブエフェクトであり、品質とパフォーマンスのバランスを取り、リアルタイム編集とRTCPマッピング機能などを備える。
Meter Effect	信号レベルに変更を加えず測定し、必要に応じて、このレベルをGame Parameterとして出力するオーディオエフェクト。特定のバスの測定レベルがRTCPを介して別のバスの音量を変化させるサイドチェインの実現に最も効果的です。
Meter (Peak)	各チャンネルのオーディオ信号レベルを表示する一連のメーター。オーディオバスと補助バス(AUXバス)は出力信号を示すが、ダイナミックエフェクト(すなわち、コンプレッサーやリミッター)は、一般的にオーディオ入力、オーディオ出力と適用されたゲインリダクションを表示する。
Mixing Desk	バスやオブジェクトのさまざまなプロパティを单一のビューに再グループ化する柔軟かつ強力なミキシングコンソールで、ゲームのオーディオミックスをリアルタイムに微調整するために使用される。
Mixing Session	ミキシングデスク内で使用される任意のWwiseオブジェクトのセットで、保存して、隨時再使用が可能。
Modulator Envelope	WaveformをMIDIや通常再生で調節するテクニックで、事前定義されたADSRの形(エンベロープ)を使う。
Modulator LFO	Waveformやそのプロパティを一定期間の間にMIDIや通常再生で調節するテクニックで、LFO(Low Frequency Oscillator)を使う。
モーションオブジェクト	Wwiseでプロジェクトのために作成した個々のモーションアセット。一般的に、これらのオブジェクトがコンソールのゲームコントローラの、ランブル(振動)機能をコントロールする。各モーションオブジェクトには、ゲーム中で生成される実際のモーションを定義する1つまたは複数のソースが含まれている。
Music Clip	Music Trackの基本的なコンポーネントであり、単一のWAVファイルを表す矩形領域として表示される。
Music Playlist Container	ランダムまたはシーケンスに再生されるMusicオブジェクトやコンテナ(单数または複数)のグループ。
Music Segment	マルチトラックのミュージックオブジェクトであり、Interactive Music Hierarchyの基本単位である。
Music Switch Container	呼び出されるSwitchまたはStateに応じて再生されるMusicオブジェクトやContainer(单数または複数)のグループ。
Music Track	個々のミュージッククリップのアレンジが入ったMusicオブジェクトであり、Music Segment内で視覚的に整列できるようウェーブフォームで表示されている。

	Wwiseに数種類のMusic Trackがあり、例えばRandom Music Track、Sequence Music Track シーケンスマジックトラック、Switch Music Trackなどである。
Nested Object	別のオブジェクト内に存在するオブジェクト。
Nested Work Units	別のワークユニット内にネストされたワークユニット。プロジェクトファイルの粒度をより細かにし、ソース管理下のファイルをマージする際にWorkgroup環境において起こりうる競合を減らすことができる。
Noise Gate	エキスパンダ (Expander) エフェクトプラグインで作成されるエフェクトのことで、出力信号からほとんど完全にサウンドを除去する。ノイズゲートは、高めのエキスパンダーレシオ (10:1以上) を設定し、ゲインがこの程度まで抑制されたサウンドに対してゲートを閉めることにより作成できる。
Object	サウンド、モーション、Actor-Mixer、コンテナなどWwise内のエレメントのことで、サウンド、ボイス、モーション、ミュージックを格納したり、グループにまとめたり、設定したりするために使う。
Obstruction	例えば柱などのゲーム中の物体が、サウンドオブジェクトとリスナー間の空間を部分的にブロックする時に発生する状態。
Occlusion	例えば壁などゲーム中の物体が、サウンドオブジェクトとリスナー間の空間を完全にブロックする時に発生する状態。
Originals folder	プロジェクトにインポートされたオーディオファイルの未処理コピーを格納するフォルダ。このフォルダは通常ソース管理下に置かれている。
Orphan file	サウンドオブジェクト、モーションオブジェクト、ミュージックオブジェクトにもはや関連付けられていないオーディオファイル。サウンドオブジェクトを削除しても、オーファンファイルは自動的には削除されない。これらのファイルを削除するには、オーディオキャッシュフォルダをクリアする必要がある。
Output buffer latency	オーディオ再生中に導入される遅延であり、Wwiseが使用する出力バッファ数によって決定される。
Parametric EQ	様々なフィルタを適用し、サウンドスペクトル形成を可能にするオーディオエフェクトプラグイン。
Parent Object	子オブジェクトを含む階層構造内のオブジェクト。
Peak Limiter	オーディオ信号のダイナミックレンジを制御するオーディオエフェクト。ピーク検出で算出された事前定義のしきい値 (スレッショルド) を一時的に超えるオーディオ信号の一部を弱めることによりこの制御が行われる。

Physical Folder	ハードディスク上のWwiseプロジェクトルート下にあるディレクトリであり、プロジェクトで使用される別の物理フォルダまたはWork Unitを含めることができる。物理フォルダを、コンテナ、モーション、またはサウンドの子オブジェクトにすることはできない。
Physical Voice	オーディオとモーションがサウンドエンジンにより再生・処理されるゲームの物理環境。音量レベルが極めて低くなると、サウンドとモーションオブジェクトは仮想環境へ移行し、そこでサウンドエンジンにより管理およびモニターされるが、オーディオ処理はされない。
Pitch Shifter	得られるオーディオ信号の持続時間に影響を与えずピッチを変更するオーディオエフェクト。
Point Source	まるで単一の点から出ているかのようにサウンドやモーションを発するソース（発生源）。
Post-exit	インタラクティブミュージックのトランジションに使用可能なエグジットキュー後のセグメント領域。
Pre-entry	インタラクティブミュージックのトランジションに使用可能なエントリキュー前のセグメント領域。
Prefetch Time	ストリーミングにおいては、残りのファイルデータをフェッチするのに必要な遅延時間をカバーする小さなバッファを指す。
Preset	オブジェクト、エフェクトとサウンド伝播のカスタマイズされたプロパティセットであり、保存していつでも再利用できる。
Query クエリ	特定のオブジェクトやプロジェクトエレメントを探すために使用される検索条件のセット。
Random Container	ランダムに再生される単数または複数のサウンド、モーションオブジェクト、またはコンテナのグループ。
Randomizer	プロパティ値に適用されるWwiseの特殊なエフェクトであり、オブジェクトが再生されるたびにランダム使用が可能な値の範囲を定義することができる。
Random Music Track	親セグメントが再生される度に、サブトラックがランダムに再生されるMusic Track。
Relative Property	プロジェクト階層内の各レベルで定義可能な音量やピッチなどのプロパティ。これらのプロパティ値は、累積的、つまり、親プロパティ値が子プロパティ値に追加される。
Reverb	特定の部屋や空間の音響特性をシミュレートするオーディオエフェクトプラグイン。
RoomVerb	特定の部屋や空間の音響特性をシミュレートする汎用性の高い高品質のリバーブエフェクトプラグイン。

RTPC	Real Time Parameter Controls. Game Parameter値をWwise内プロパティにマッピングして、オーディオをゲーム内に駆動するインターラクティブメソッド。RTPCは、ゲームパラメータをスイッチグループにマッピングすることによりスイッチ変更を駆動するためにも使用される。
サンプルレート	オーディオ信号をデジタル信号へ変換する時、または、デジタル変換中の1秒あたりのサンプリング頻度。
Send Volume	Auxiliary Busに送信されるオーディオ信号のレベルまたは振幅。
Sequence Container	1つまたは複数のサウンド、モーションオブジェクトと/またはコンテナのグループであり、特定のプレイリストに従って再生される。
Sequence Music Track シーケンスマジック トラック	親セグメントが再生される度に、サブトラックが順番に再生されるMusic Track。
ShareSet ShareSet	エフェクトや減衰などの属性を定義するためにオブジェクト間で共有可能なプロパティセット。
Silence Source	サウンドやモーションを生成しない指定された期間のプラグインソース。
Slider Range	スライダを使用しプロパティに入力できる値のデフォルト範囲で、手動でフィールドに入力できる範囲とは対照的で、Slider Rangeを変更します。
SoundBank	Eventデータや、Sound、Music、Motionのそれぞれの構造データや、メディアファイルなどをグループにまとめたもので、ゲーム内の特定の時点でゲームのプラットフォームメモリにロードされる。
Soundcaster	必要に応じて挿入・削除の可能なオブジェクトまたはイベントのゲーム内またはWwiseにおけるトランスポートコントロール、オーディションとリアルタイムミキシングが行えるWwiseのビュー。
Soundcaster Session	サウンドキャスター（Soundcaster）で作成された特定のシミュレーションで使用されるWwiseオブジェクトとイベントを格納する保存済みプロジェクトエレメント。
SoundFrame	外部アプリケーションがWwiseとシームレスに通信できるようにするユニークなプラグインインターフェイス。SoundFrameで連携可能なツールには、SFTestとCarSimなどがある。
サウンドオブジェクト	Wwiseプロジェクトのために作成される個々のオーディオアセット。各Soundオブジェクトにはソース（単数または複数）が格納されており、これにより、実際にゲーム中で再生されるオーディオコンテンツを定義する。なおWwiseドキュメンテーションで

	Soundと大文字の"S"で記載する場合はSoundオブジェクトを指し、Sound SFXまたはSound Voiceを意味する。
Sound SFX	アクターミキサー階層内のサウンドオブジェクトであり、サウンド、ミュージックおよびアンビエンスを格納する。
Sound Voice	ダイアログまたはゲームボイスオーバーを含むサウンドオブジェクト。
Source Plug-in	Wwise外部のプラグインにより作成されたオーディオまたはモーションソース。
Spatialization	ゲームの3D 環境内サウンドまたはミュージックオブジェクトの実際の場所や位置を決定するWwise機能。
Spread	オーディオやモーションが隣のスピーカーやモーターに広がる量または割合を指定する機能で、サウンドやモーションが距離に合わせて、スプレッド値が低い時は点音源、高い値では完全に拡散した伝搬へと変わる。マルチチャンネルサウンドのスプレッドは、各チャンネルで別々に行われる。
State	ゲーム中の物理的および環境的状況変化に関連するゲームオーディオのプロパティへのグローバルなオフセットまたは調整。
State Group	ゲーム環境内で発生するグローバルな変更を管理するために、グループにまとめられた関連性のあるStateの集まり。
Stereo Delay	内蔵フィルタでデュアルチャンネルディレイを生み出すオーディオエフェクト。1つのチャンネルから別のチャンネルへ遅延信号を送信してステレオエフェクトを作り出すフィードバックコントロールとクロスフィードコントロールがある。
Stinger	再生中のミュージックに重ねてミキシングされる短いミュージックフレーズのこと。
Switch	Switchは、ゲームの特定エレメントに対して存在する全ての選択肢を表し、それぞれの選択肢に対応するオブジェクトを管理するために使います。例えば、コンクリート上を走っているキャラクターが草の上へ移動すると、Switch Container内の足音は、地面の変化に応じて変化する必要がある。
Switch Container	一連のスイッチまたはステートのことで、それぞれにはゲーム内環境の特定の変化に対応するサウンド、モーション、オブジェクトまたはコンテナのグループを含んでいる。 例えば、キャラクターの足音用Switch Containerには、草地、コンクリート道、木の床などゲーム中でキャラクターがその上を歩く可能性のあるあらゆる地面のSwitchが含まれている。
Switch Group	ゲーム内の特定エレメントのために存在する様々な選択肢を管理するために、グループにまとめられた関連性のあるSwitchの集まり。

Switch Music Track	親セグメントが再生される度に、結びつけられたSwitch Groupの設定に従い、サブトラックが再生されるMusic Track。
Time Stretch	得られるオーディオ信号のピッチに影響を与える持続時間を変更するオーディオエフェクト。
Transition	インタラクティブミュージックにおいて、トランジションはソースMusic SegmentとデスティネーションMusic Segment間のスムーズな橋渡しを意味する。
Transition Time	同じStateやState Group内で、1つのStateから別のStateへの移行に使用される時間。トランジション時に、2つのStateプロパティの補間が発生する。
Tremolo	ユニポーラ搬送波信号で入力信号の振幅を変調するオーディオエフェクト。
Trigger	ゲーム内で発生する事象に応答してステインガーを起動するゲームシンク。
Virtual Folder	フォルダとして表示され、ワークユニットまたはその子オブジェクトのうちの1つに含まれる組織オブジェクトであり、Virtual Folder、Actor Mixer、コンテナ、モーションやサウンドなど他のオブジェクトを配置することができる。仮想フォルダを、コンテナ、モーションまたはサウンドの子オブジェクトにすることはできず、ハードディスク上に対応するディレクトリを持たない。
Virtual Voice	サウンドとモーションが管理およびモニターされる仮想環境であるが、ここでオーディオ処理は行われない。オブジェクトは音量レベルが音量しきい値（ボリュームスレッシュルド）以下になると、仮想音声へ移行する。
Voice	個別または不連続の再生インスタンスであり、オーディオまたはモーションのいずれか。
Voice Starvation	サウンドエンジンがオーディオデータを適時にプラットフォームハードウェアバッファへ提供できない場合、キャプチャーログ（Capture Log）に表示されるエラーメッセージの一種。ホストCPUの過度の使用があるとこのタイプの問題が発生する。例えば、プラットフォームCPUがミキシングしようとしているソースや同時に使用しているオーディオエフェクトが多すぎる場合など。
Volume Threshold	特定の音量レベルのことで、これを下回るとサウンド、ミュージックとモーションオブジェクトのビヘイビアを具体的に定義できる。例えば、音量しきい値を下回る音声は、再生の継続、停止または仮想音声リストへ送信のいずれかが可能。これらのビヘイビアはオブジェクトの「プロパティエディタ（Property Editor）」にある「詳細設定（Advanced Settings）タブ」で定義できる。

Watch ゲーム内のゲームオブジェクトとリスナーのモニターに使用可能な個別プロセス。

Workgroup 同一のWwiseプロジェクトに同時に取り組んでいる人々。

Work Unit プロジェクト内の特定のセクションまたはエレメントに関連する情報を含む個別のXMLファイル。