

audiokinetic

Wwise

用户指南

2019.1.1



Wwise

用户指南

Wwise 2019.1.1 Revision 3471
Copyright © 2019 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.

目录

欢迎使用 Wwise 2019.1.1	x
I. 从何处入手?	1
1. Wwise 路线图	3
从哪里开始	4
Wwise 路线图指南	4
文档约定	4
Wwise 帮助文档	5
与支持部门联系	7
2. 入门	8
启动和退出 Wwise	9
探索 Wwise 界面	10
Wwise 界面基础知识	12
个性化您的工作空间	21
提供工作效率	35
II. 设置工程	46
3. 处理工程	48
概述	49
管理工程	51
定义工程设置	54
定义工程的默认用户设置	76
排查工程的问题	77
学习附录 —— 有关声障和声笼的详细信息	82
工程管理技巧与经验总结	84
4. 管理多平台	86
概述	87
使用 Platform Manager	87
5. 管理语言	91
概述	92
使用 Language Manager	92
6. 管理 Workgroup	96
概述	97
将工程分成 Work Units	100
查看工程文件的状态	107
结合版本控制系统使用 Wwise	109
解决工程中的不一致现象	109
使用 Workgroup 插件管理工程文件	111
Workgroup 技巧与经验总结	122
7. 管理工程中的媒体文件	125
概述	126
导入的过程	126
导入媒体文件	133
替换媒体文件	149
管理文件导入问题	151
重新组织 Originals Folder 中的媒体文件。	153
清除缓存	154
在外部编辑器中编辑音频文件	155

使用插件创建 Source	156
媒体文件管理技巧和经验总结	157
8. 构建 Actor-Mixer Hierarchy	158
概述	159
什么是声音对象?	159
在创建 Actor-Mixer Hierarchy 时为对象分组	161
构建 Actor-Mixer Hierarchy	164
工程层级结构中的属性介绍	168
通过随机化属性值来改善播放	176
构建 Actor-Mixer Hierarchy 的技巧和经验总结	177
9. 建立输出总线的结构	181
概述	182
定义总线的属性	186
构建总线层级结构 —— 示例	190
III. 使用声音和振动来提升游戏体验	192
10. 定义对象播放行为	194
概述	195
定义对象的播放行为	195
定义 Random/Sequency Container 的播放行为	201
定义 Switch Container 的内容和行为	211
定义 Blend Container 的内容和行为	216
对象播放相关的技巧和经验总结	225
11. 定义定位	227
定义定位概述	228
理解 Wwise 中的定位	228
使用 Speaker Panning	231
使用 3D 空间化对象	232
Speaker Panning 和 3D Spatialization 交叉淡变	235
为总线应用定位	235
3D 定位图解	236
应用基于距离的衰减	247
使用动画路径定义空间定位	259
将音频信号传送到中置扬声器	268
定位技巧和经验总结	269
理解声道配置	275
声像摆位规则：扬声器对耳机	277
使用 Ambisonics	278
12. 管理优先级	284
概述	285
了解 Wwise 如何为对象排列优先级	286
限制对象的播放实例	289
定义播放优先级	292
管理音量较低的对象	293
优先级相关技巧和经验总结	295
13. 管理效果器	297
概述	298
使用效果器	298
使用效果器实现环境音效	306
效果器技巧和经验总结	308

14. 管理 Motion	310
概述	311
为游戏创建振动	313
为振动来构建输出结构	321
振动相关技巧和经验总结	322
IV. 与游戏互动	324
15. 管理 Event	326
概述	327
创建 Event	331
处理 Event	337
事件技巧和经验总结	340
16. 管理动态对话	341
概述	342
理解 Dynamic Dialogue 系统	342
创建 Dialogue Event	345
处理 Dialogue Event	354
Dialogue Event 技巧和经验总结	357
17. 使用 State	360
概述	361
使用 State	363
为 State Group 内的 State 之间的切换设置过渡	366
将 State 指派给对象和总线	367
State 技巧和经验总结	371
18. 使用 Switch	372
概述	373
使用 Switch	374
将 Game Parameter 值映射到 Switch	376
Switch 技巧和经验总结	377
19. 使用 RTPC	379
概述	380
管理 RTPC 中使用的 Game Parameter	382
使用 Game Parameter 控制属性值	386
使用 LFO	393
使用包络	394
使用时间	397
查看 Game Object	398
使用旁链压缩	398
RTPC 技巧与最佳实践	404
20. 使用 Trigger	406
概述	407
使用 Trigger	408
21. 将 State 和 State Group 用于动态对话	410
概述	411
使用 State Group	411
V. 创建互动音乐	415
22. 理解互动音乐	418
概述	419
理解互动音乐	419
互动音乐技巧和经验总结	420

23. 构建 Interactive Music Hierarchy	423
概述	424
什么是 Music Segment?	424
容器类型	426
关于 Interactive Music Hierarchy 中的属性	427
将对象添加到 Interactive Music Hierarchy	428
添加父对象	429
管理 Interactive Music Hierarchy 中的音乐对象	430
构建 Interactive Music Hierarchy 的技巧和经验总结	431
24. 定义音乐对象播放行为	432
概述	433
定义音乐对象的 Time Settings	433
定义 Music Playlist Container 的播放行为	434
定义 Music Switch Container 的内容和行为	437
互动音乐播放的技巧和经验总结	440
25. 使用 Music Track 和 Music Segment	442
概述	443
将 Music Track 添加至 Music Segment	444
定义 Music Track 的播放行为	445
将子音轨添加至音轨	447
将 Sub-Track 关联至 Switch/State	448
添加音轨	449
从段落中移除音轨和子音轨	449
在 Music Editor 中按间距对齐	450
使用片段	450
预览 Music Segment	453
使用 Cue	454
26. 处理 MIDI	460
创建 MIDI 内容	461
导入 MIDI 文件	461
理解 MIDI 内容和 MIDI 目标	461
混合 MIDI 和音频内容	462
理解 MIDI 速度	463
更改 MIDI 的播放速度	463
27. 创建 MIDI 乐器	465
设计 Synth One 乐器	466
设计简单的采样 MIDI 乐器	466
理解 MIDI 音符跟踪	466
理解 MIDI 筛选器	467
了解 MIDI 事件	467
在 MIDI 事件上添加淡入和淡出	468
使用 MIDI 数据控制对象属性值	468
使用 MIDI Keymap Editor	469
使用 MIDI 键盘测试乐器	469
将 MIDI 从 DAW 连通到 Wwise	470
28. 使用 Transition	472
概述	473
理解 Transition	473
添加 Transition	475

复制和粘贴 Transition	476
移除 Transition	476
设置源和目标属性	477
使用 Transition Segment	479
互动音乐中 Transition 的技巧和经验总结	481
29. 使用 Stinger	483
概述	484
添加 Stinger	485
定义 Stinger 的播放设置	486
移除 Stinger	487
预览 Stinger	488
VI. 完善工程	490
30. 管理输出	493
概述	494
指定对象的输出连线	494
使用用户定义的辅助发送	495
使用游戏定义的辅助发送	497
使用响度归一化或补偿增益来调节音量	497
理解声部管线	498
内置音频设备	500
理解 Secondary Output	501
理解 HDR	506
使用 HDR	509
深入了解 HDR	517
完成终混	521
31. 管理平台和语言版本	536
概述	537
跨平台制作	537
对工程进行本地化	555
版本技巧和经验总结	559
32. 创建模拟	562
概述	563
创建模拟	564
管理模拟的播放	567
使用 Game Sync 模拟	569
在模拟中细调属性	572
创建模拟的技巧和经验总结	574
33. 在 Wwise 中管理内存	576
概述	577
理解内存管理器的组件	577
设置内存池的大小	578
排查内存问题	579
优化内存池	580
内存管理技巧和经验总结	584
34. 性能分析	586
概述	587
连接到本地/远程游戏系统	595
从声音引擎捕获数据	597
使用 Performance Monitor 进行监控和故障排查	606

使用 Voice Inspector 分析声部	608
使用 Game Object Explorer 跟踪对象和听者	614
通过 Game Object 3D Viewer 检查对象	618
通过 Game Sync Monitor 评估游戏同步器	623
性能分析技巧和最佳实践	624
35. 管理 SoundBank	626
概述	627
理解 SoundBank 如何加载到游戏中	629
构建 SoundBank	632
管理 SoundBank	647
定义 SoundBank 的自定义属性	649
为工程生成 SoundBank	656
使用 CopyStreamedFile 工具	659
SoundBank 的管理策略	660
SoundBank 技巧和经验总结	672
36. 管理 File Package	676
概述	677
使用 File Packager 工程	677
管理工程内的 File Package	679
可下载内容 (DLC) 概述	683
生成 File Package	684
在命令行中使用 File Packager 参数	684
File Packager 技巧和最佳实践	687
VII. 使用 Wwise	689
37. 认识 Project Explorer 视图	692
概述	693
Project Explorer 中的可视元素	695
在 Project Explorer 中工作	695
38. 认识 Event Viewer 视图	699
概述	700
使用 Event Viewer	700
39. 认识 Property Editor 视图	705
概述	706
使用 Property Editor	723
40. 认识 Contents Editor 视图	726
概述	727
使用 Contents Editor	734
41. 认识 Transport Control 视图	741
概述	742
设置播放属性	744
在 Transport Control 中平移对象	746
播放/暂停/停止内容	746
在播放期间使用 Game Sync	747
42. 认识 Schematic View 视图	752
概述	753
自定义 Schematic View 视图	753
使用 Schematic View 视图	755
43. 了解坐标图视图	758
概述	759

更改坐标图的显示	760
在坐标图视图中使用控制点	764
使用坐标图视图中的曲线	766
44. 认识时间线	770
概述	771
使用时间线进行定位	773
使用 Music Segment Editor 时间线	774
45. 使用 Search、Query 和 Reference	777
概述	778
搜索工程中的元素	778
查找引用特定对象的工程元素	780
使用 Query	784
查询 —— 技巧和经验总结	791
46. 使用 Preset	792
概述	793
使用 Preset	793
47. 使用控制设备	797
概述	798
将控制设备连接到 Wwise	798
创建 Control Surface Session	799
理解控制设备绑定	799
创建控制设备绑定	800
理解 Control Surface 的 View Group	803
处理控制 Control Surface Session 中的冲突	804
使用 Control Surface 工具栏	804
VIII. 附录	806
A. 下混行为	808
标准配置	809
下混表	810
B. 正则表达式快速参考指南	814
C. 快捷键	816
可缩放的编辑器	817
Attenuation Editor 视图	817
音频文件管理	817
Contents Editor 视图	818
Game Object 3D Viewer 视图	818
Game Profiler	818
全局 —— 上下文（活跃视图和选定对象）	818
Music Segment Editor 视图	819
Position Editor (3D Automation)	819
Project Explorer 视图	820
RTPC 坐标图视图	820
Schematic View 视图	820
Soundcaster 视图	820
Transport Control 视图	820
词汇表	822

欢迎使用 Wwise 2019.1.1

欢迎使用 Wwise® 版本 &version;。它是 Audiokinetic 推出的中间件解决方案，可以帮助您为电子游戏创建高品质音频和振动内容。通过紧密集成高级 Wwise 设计工具、强大的声音引擎和用于控制它们的全面的 SDK，Wwise 可以提高您的工作效率，提升您的创意作品品质。

Wwise 是专门为了满足游戏开发管线的特定需求而设计，是面向设计师、作曲家和程序员的独特解决方案。通过让您在开发游戏画面的同时开发游戏音频、音乐和振动，Wwise 可以在游戏开发的各个阶段为复杂音频的设计和制作提供便利。

关于本指南

本指南适用于在游戏开发行业中从业的设计师、集成师、编剧、作曲家和程序员。本指南假定您拥有计算机和音频常识。文中解释了一些基本概念，但仅与 Wwise 软件相关。本指南为您提供有关 Wwise 中最常见任务的具体信息和详细步骤。

要了解有关文档结构组织的详细信息，请参阅[Wwise 帮助文档](#)页面。

前期准备

在使用 Wwise 前，您可以阅读《Wwise 基础知识》文档，以更好地理解 Wwise 的主要概念和工作流程。在仔细阅读《Wwise 基础知识》文档后，便可以开始阅读帮助文档每一部分中的各章内容。这些章节提供有关使用 Wwise 的概念和程序信息、示例以及技巧和经验总结。

使用其它 Wwise 文档和支持

Wwise 帮助只是您可以使用的整套材料和资源的一部分。请查阅[Wwise 路线图指南](#)了解有关其它指南和资源的信息。



部分 I. 从何处入手?



1. Wwise 路线图	3
从哪里开始	4
Wwise 路线图指南	4
文档约定	4
Wwise 帮助文档	5
与支持部门联系	7
2. 入门	8
启动和退出 Wwise	9
探索 Wwise 界面	10
Wwise 界面基础知识	12
个性化您的工作空间	21
提供工作效率	35

第 1 章 Wwise 路线图

从哪里开始	4
Wwise 路线图指南	4
文档约定	4
Wwise 帮助文档	5
与支持部门联系	7

从哪里开始

Audiokinetic 为您提供了一套综合的材料和资源。在开始使用 Wwise 之前，您应查阅路线图以全面了解可供利用的内容。我们希望这些材料和资源能够有助于您将 Wwise 整合到内容制作流程中去。

Wwise 路线图指南

有关 Wwise 提供的不同资料和资源的完整指南，请参阅下表。

	发布说明	有关 新功能信息 、产品局限和解决方法，请参阅发布说明。
	Wwise 安装和迁移指南	使用该安装和迁移指南来安装或重新安装 Wwise 及其组件应用程序，以及将工程从一个版本迁移到另一个版本。
	Wwise 视频教程	所有用户都可观看我们提供的 视频教程 ，了解 Wwise 概念以及如何在 Wwise 中执行特定任务。
	Wwise 在线帮助	有关 Wwise 中所有选项和视图的屏幕信息，请查阅 在线帮助 。
	Wwise 用户指南	有关基本信息和高级的专项信息，请参阅该用户指南。
	Wwise 游戏模拟器 (Game Simulator) 帮助	有关如何创建脚本以在各种平台上测试游戏场景的信息，请参阅游戏模拟器帮助。
	Wwise SDK 文档	有关 集成声音引擎 、 SoundFrame® 、插件等的信息，请参阅 SDK 文档。
	Wwise 示例工程 (Sample Project)	要深入了解 Wwise 中的实用声音设计示例，请查看示例工程和相应文档。
	Audiokinetic 问答论坛 (Q&A)	请访问我们的 Audiokinetic 问答论坛 以询问和回答问题，这个论坛是 Wwise 用户和专家组成的社区。
	Wwise 技术支持中心	访问我们的 在线技术支持中心 。

文档约定

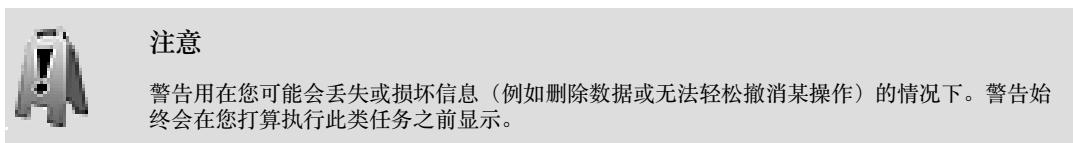
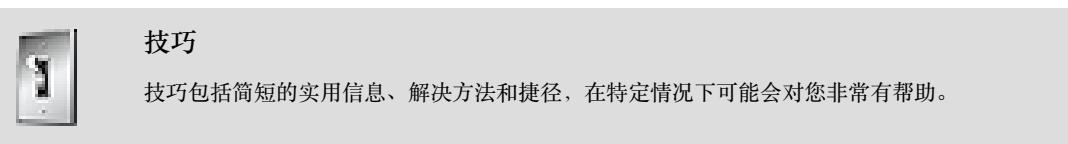
所有 Wwise 用户文档中的信息都会使用以下约定来显示：

- [视觉识别符号](#)
- [键盘和鼠标约定](#)

视觉识别符号

以下图标有助于识别某些类型的信息：





键盘和鼠标约定

Wwise 会用到鼠标左键、中键和右键。除非另有说明，一般使用的是鼠标左键。下表说明了与有关鼠标和键盘有关的术语。

此术语	使用鼠标来解释该术语
点击	快速按下并释放鼠标左键。除非另有说明，请始终使用鼠标左键。
右键点击	快速按下并释放鼠标右键。
双击	快速点击两次鼠标左键。
Shift+点击、Ctrl+点击、Alt+点击	在点击鼠标按钮时按住 Shift、Ctrl 或 Alt 键。
拖动	在移动鼠标时按住鼠标左键。这相当于拖放操作。
Alt+键、Ctrl+键、Shift+键	按住第一个键的同时按第二个键。例如，“按 Ctrl+Z”意味着按住 Ctrl 键的同时按 Z 键。

Wwise 帮助文档

Wwise 帮助是上下文相关帮助系统，包含说明 Wwise 中各个视图、字段和选项的参考主题以及基于任务的信息。文档分为多个部分，可为您提供有关 Wwise 的全面且易于查找的信息。

- **第 I 部分“从何处入手？”**——包括 Wwise 文档集的概述、Wwise 中许多概念的介绍和帮您快速上手 Wwise 的新手入门信息，以及一些有助于组织工作并高效使用 Wwise 的信息。
- **第 II 部分“设置工程”**——介绍如何设置工程（单人或多人群组环境下）、如何为您的工程定义或调节平台、如何处理工程中的素材以及如何建立 Wwise 素材的层级结构并充分加以利用，包括概念和操作步骤信息。
- **第 III 部分“使用声音和振动来提升游戏体验”**——介绍如何使用各种属性和行为、效果器、振动、空间定位设置和播放优先级来增强丰富多彩、身临其境的游戏环境，包括概念和操作步骤信息。
- **第 IV 部分“与游戏互动”**——介绍如何使用事件、动态序列以及 Wwise 游戏同步器驱动游戏中的声音，包括概念和操作步骤信息。
- **第 V 部分“创建互动音乐”**——包括 Wwise 中互动音乐概念的概述和介绍，以及介绍如何为游戏创建互动音乐，含大量操作步骤信息。
- **第 VI 部分“完善工程”**——包括有关如何创建最终混音、故障排查和模拟工程的不同方面并为游戏生成 SoundBank 和文件包的信息。
- **第 VII 部分“使用 Wwise”**——说明特定 Wwise 视图的功能，以帮助您上手使用软件。

- Wwise 参考文档（包含在《Wwise 用户指南》中）—— 提供有关 Wwise 中所有选项和视图的完整信息，并且可通过上下文帮助进行访问。
- Wwise 源/效果器插件（不包含在《Wwise 用户指南》中）—— 提供 Wwise 随附源和效果器插件的各个选项的说明。
- Wwise 工具（不包含在《Wwise 用户指南》中）—— 提供有关 Wwise 随附的各种工具（包括 SoundFrame（声音构架）和 Multi-Channel Creator）的信息。
- **第 VIII 部分“附录”**—— 提供有关 Wwise 下混方式、正则表达式使用和默认键盘快捷方式的补充参考信息。
- **词汇表**—— 提供按字母排列的 Wwise 和音频相关术语，以及相应定义。

相关主题

- **设置文档偏好**

访问 Wwise 上下文相关帮助

在您查找有关 Wwise 中特定选项或视图的信息时，只需在视图中打开帮助并阅读相应信息即可。

访问 Wwise 上下文相关帮助的方法如下：

1. 执行以下操作之一：

- 点击任何视图或对话框右上角的 Help（帮助）图标。
- 按 F1。

此时将会打开 Wwise 帮助并显示有关您当前所在视图或对话框的信息。

访问 Wwise 帮助

要查看整个帮助系统，您可以打开 Wwise 帮助起始页并从该页面前往其它主题。

打开 Wwise 帮助的方法如下：

1. 从菜单栏中，点击 Help > Wwise Help（帮助 > Wwise 帮助）。

此时将会打开 Wwise 帮助的起始页。

更改 Wwise Help 语言

Wwise Help（包括上下文相关帮助和 Property Help）可设为 English（英语）、Japanese（日语）、Korean（韩语）或 Simplified Chinese（简体中文）。

更改 Wwise Help 语言的方法如下：

1. 在菜单栏中，依次单击 Help > Documentation Language（帮助 > 文档语言）。

这时将显示语言子菜单。

2. 选择语言。

这时将按所选语言打开 Property Help 视图、Wwise Help 和 Wwise SDK 文档。

与支持部门联系

使用 Audiokinetic [Customer Portal](#)（客户门户）提交支持工单。



技巧

您还可以通过我们网站的 [Q&A](#)（问答论坛）在 Wwise 用户社区中提出问题。

第 2 章 入门

启动和退出 Wwise	9
探索 Wwise 界面	10
Wwise 界面基础知识	12
个性化您的工作空间	21
提供工作效率	35

启动和退出 Wwise

Wwise 支持在 Microsoft Windows® 7、8、10 以及 macOS 上运行。为了让您快速访问 Wwise，Wwise Launcher 应用可以管理所有 Wwise 版本的安装和迁移。如果尚未安装 Wwise，首先需要点击 Audiokinetic 网站 [下载](#) 页面中的 Download Wwise 来安装 Wwise Launcher。之后，Wwise 的安装将非常简单，只需点击所需 Wwise 版本的按钮即可。

启动 Wwise 的方法如下：

1. 执行以下操作之一：

- 在桌面上双击 Wwise Launcher 图标。
- 从 Windows 启动菜单中，选择 Wwise Launcher。

Wwise Launcher 将打开。

2. 单击 Log in 登录到 Audiokinetic 客户门户来获取您帐户的所有权限。

3. 如果以前使用过 Wwise，那么您可以转到 Projects 选项卡。

如果从未使用过 Wwise，请转至 Wwise 选项卡。

4. 对于现有工程，点击 Open with Wwise #####.#.#.#####（对应于您选择的版本）。工程将在该版本的 Wwise 中打开。

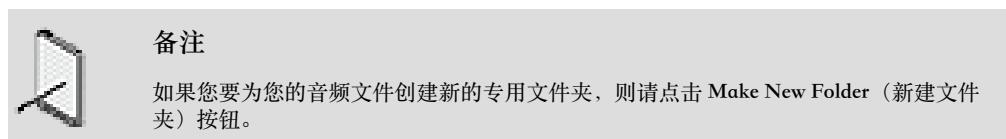
否则，请点击 Launch Wwise，打开您需要的版本。该版本的 Wwise 将打开，并显示 Project Launcher。

5. 选择一个工程或点击 New... 来创建一个新工程。

在前一种情况下，Wwise 会加载所选工程供您编辑。

后一种情况下，New Project 对话框将打开，您需要填写以下信息。

- 在 Name（名称）字段中键入工程的名称。
- 要更改工程文件的保存位置，请执行以下步骤：



- 点击 Browse (浏览) 按钮 (...)。
- 此时将会打开 Browse For Folder (浏览文件夹) 对话框。
- 前往您希望 Wwise 储存工程文件的文件夹。
- 点击 OK。
- 要更改原始音频文件的保存目录，请执行以下操作：
 - 点击浏览按钮 (...)。
 - 此时将会打开 Browse For Folder (浏览文件夹) 对话框。
 - 前往您希望 Wwise 存储文件原始副本的文件夹，用来存储已导入至工程的音频。
 - 点击 OK。

d. 点击 OK。

Wwise 打开新的工程。

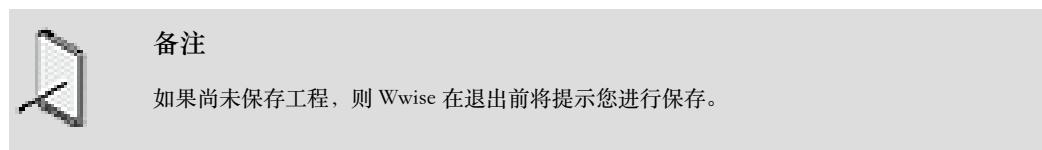
退出 Wwise

完成工作会话后，保存您的工作并退出该应用程序。

退出 Wwise 的方法如下：

1. 在菜单栏中点击 Project > Exit。

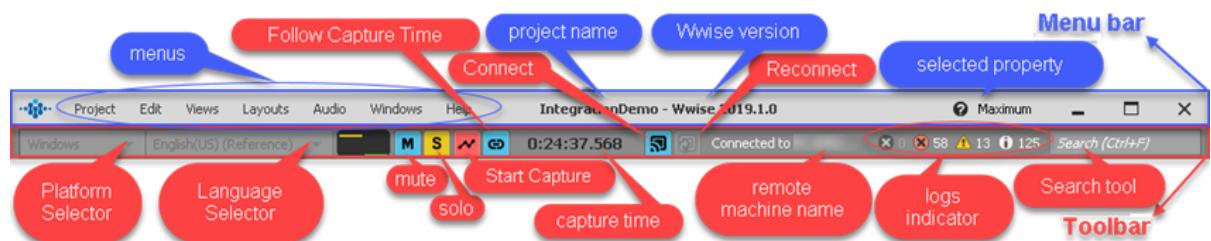
Wwise 将关闭工程并退出。



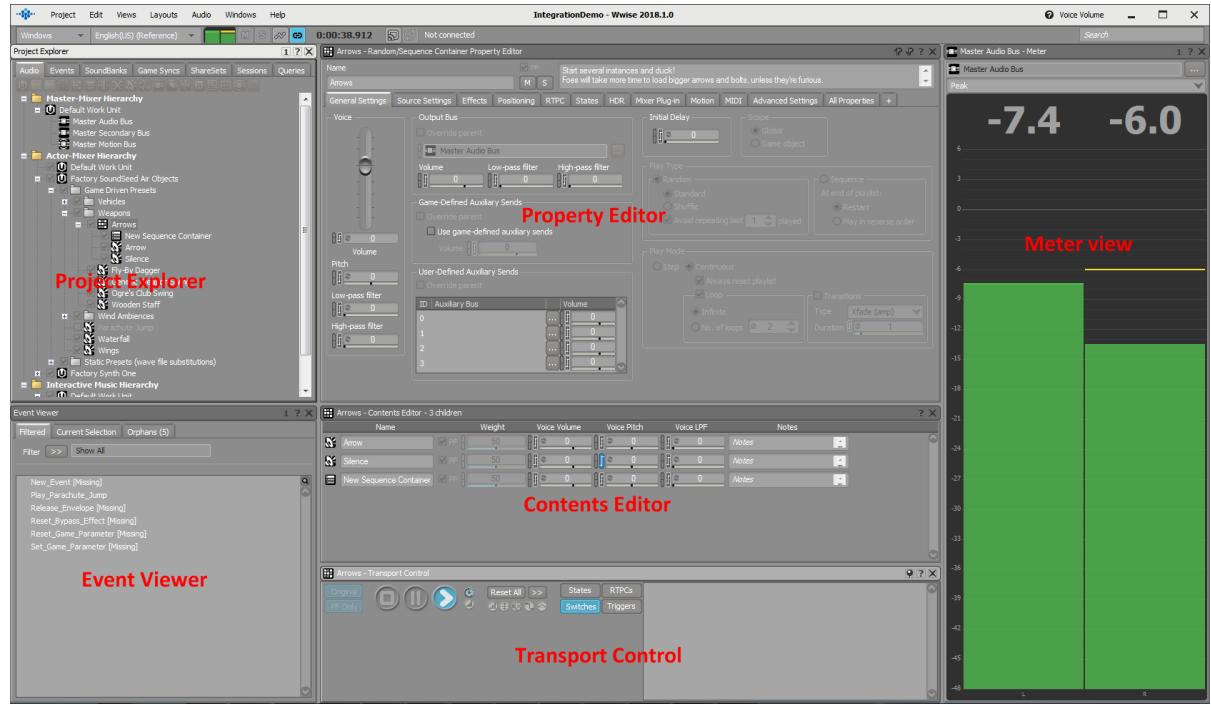
探索 Wwise 界面

Wwise 界面分为多个不同的视图。各个视图都有其特定的目的，您可以访问其中的各种工具或选项，以管理和定义游戏的内容。这些视图组合在一起即成为布局，方便您处理特定任务或作业所需的工作。Wwise 中提供了不同的布局。有关详细信息，请参阅[处理布局](#)。

各个布局的顶端是菜单栏和工具栏。通过菜单栏可以访问所有基本信息（如工程名称）和基本命令（如保存工程、更改布局、打开视图或打开属性的帮助条目）。通过工具栏可以快速访问某些工具，如 Platform 或 Language Selector、Capture 工具、Remote Platform Connector 和 Search 工具。



当您首次启动 Wwise 时会显示 Designer 布局。



该布局包含以下视图：

- **Project Explorer**: 这是主要区域，您可在其中管理和组织 Wwise 工程的各种元素。Project Explorer 包含以下选项卡：
 - **Audio**: 一种层次树视图，非常像 Windows 资源管理器和 Mac Finder，可以在其中组织工程中的资源（asset）。Audio 选项卡包含三个主要层级：Master-Mixer Hierarchy、Actor-Mixer Hierarchy 和 Interactive Music Hierarchy。
 - **Events (事件)** : 显示工程中的事件，包括操作和对白事件。
 - **SoundBanks (音频包)** : 显示工程中的所有 SoundBank。
 - **Game Syncs (游戏同步器)** : 显示工程中的所有 Switch (切换开关)、State (状态)、Game Parameters (游戏参数) 和 Trigger (触发器)。
 - **ShareSets (共享集)** : 显示工程中的所有效果和衰减工程集。
 - **Sessions (会话)** : 显示工程中的所有 Soundcaster (声音选角器) 会话。
 - **Queries (查询)** : 显示工程中的所有查询。
- **Event Viewer (事件查看器)** : 显示已为当前工程创建的不同事件。Event Viewer 包含三个不同的选项卡：Filtered (已筛选)、Current Selection (当前选中) 和 Orphans (孤立)，各个选项卡可以不同的方式筛选事件。
- **Property Editor (属性编辑器)** : 包含很多属性和行为选项，您可使用它们定义声音、音乐或总线结构内特定对象的综合特性。
- **Contents Editor (内容编辑器)** : 显示已加载至属性编辑器内的父对象内包含的若干个对象。您还可以通过 Contents Editor 快速访问各个对象相关的最常用属性，如音量和音高。
- **Transport Control (播放控制)** : 播放您的对象。Transport Control 包含与音频播放相关的传统控件，如播放、停止和暂停。

- Meter view (电平表视图)：为各个通道显示彩色电平值，可以选择三种不同类型之一，即峰值、真实峰值和 RMS 值。对于 Ambisonics 总线，Meter 视图还可显示 3D Meter（显示方向数据）。

Wwise 界面基础知识

以下章节说明标准操作系统功能和 Wwise 特定功能。

- [了解 Wwise 中的命名规范](#)
- [使用文本框](#)
- [使用列表](#)
- [使用滑杆](#)
- [使用表格](#)
- [理解 Wwise 中的视觉元素](#)
- [Wwise 中的撤销和恢复操作](#)

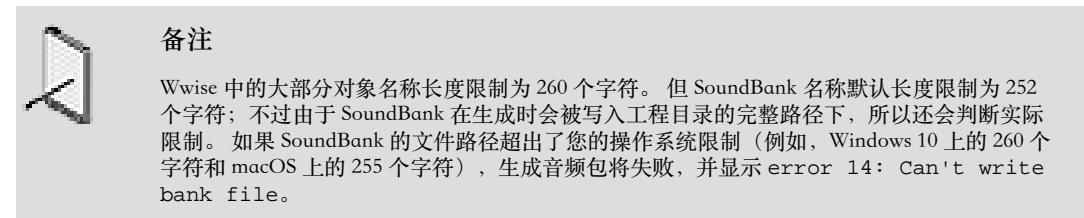
了解 Wwise 中的命名规范

在 Wwise 中命名工程和对象时，您可使用任何 Unicode 支持的字符。

但为了遵循特定游戏引擎的命名限制，在 Wwise 中命名以下元素时会有一些限制：

- SoundBanks (声音包)
- Event
- Dialogue events (对白事件)
- Effect ShareSets (效果共享集)
- Switch Groups (切换开关组)
- Switches (切换开关)
- State Groups (状态组)
- States (状态)
- RTPC (实时参数控制) 游戏参数
- Triggers
- Work Units

这些元素仅接受数字、无重音罗马字符和下划线。另外，名称开头不得为数字。

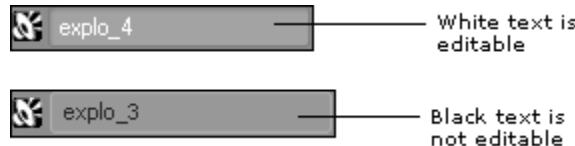


使用文本框

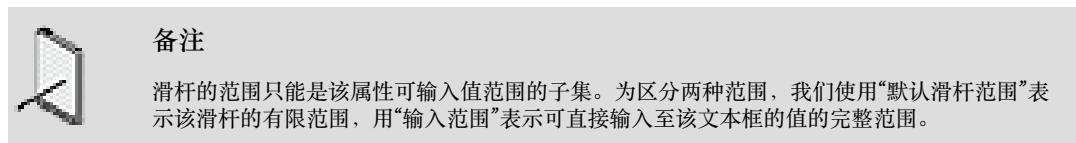
Wwise 中的大部分视图包含字段栏或文本框，您可在其中输入适当的值或对象的信息。文本框的名称表示栏中所显示信息的类型。根据视图的不同，文本框的名称可位于视图的旁边、上方或下方。



在界面的有些区域，文本显示于文本框内，但无法进行编辑。为帮助您区分可编辑与不可编辑文本，Wwise 使用不同的颜色显示文本。当文本显示为白色时，可以对其进行编辑。文字变黑时将不允许编辑；但需要时，您可以右键单击文本并选择 Copy Text 来复制它。



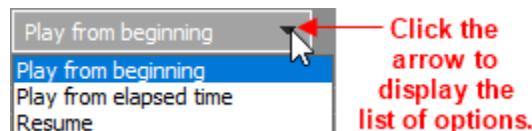
包含属性值的大部分文本框还在下方包含一个垂直滑杆或推子，您可进行拖动以更改这些文本框内的值。有关滑杆的详情，请参阅 [使用滑杆](#)。



如果您要将属性值恢复至其默认设置，则请按住 Ctrl 并点击文本框。

使用列表

Wwise 中包含两种类型的列表：下拉式列表和快捷方式列表（有时称为上下文列表）。下拉列表（本帮助文档中简称为“列表”）指包含很多预定义选项的下拉栏。要显示列表，请点击下拉栏右侧的箭头。



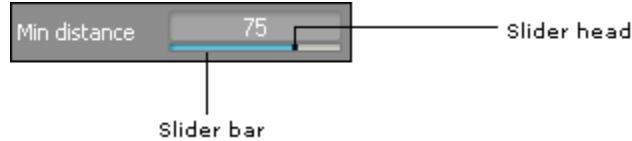
选择器按钮 (>>) 用于显示一组选项或操作。快捷方式菜单可能包含相关信息栏，以显示所选的选项。点击该按钮，以显示菜单选项。



使用滑杆

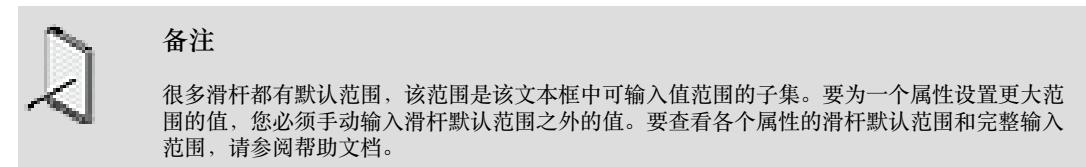
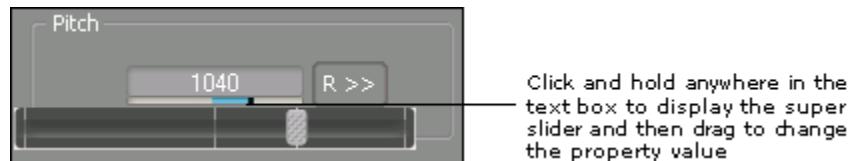
滑杆是用于从连续取值范围内设置某个值的控件。包含属性值的大部分栏或文本框在下方还含有一个水平滑杆。

水平滑杆包含一个滑块和一个滑动槽。滑块是一个小黑点，表示当前的属性值。滑动槽是一个蓝色条，表示当前值大小，取值在允许范围内。



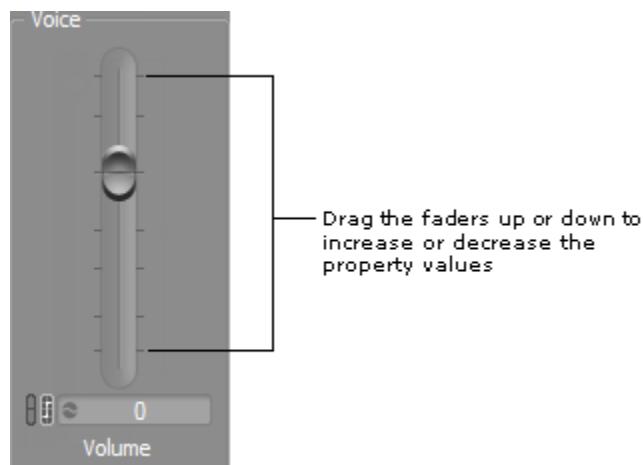
默认情况下，滑杆将根据属性默认值相对取值范围所处的位置，显示在滑杆的不同位置。默认情况下，根据属性默认值相对取值范围所处的位置，滑杆也会起始于不同的位置，并向不同的方向移动。

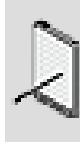
您可向左或向右拖动滑杆，以增加或减小属性值。在文本框中点击并按住鼠标键时，会显示大型滑杆，以方便您微调该属性值。松开鼠标按钮时，较大的滑杆会消失。如果您需要更精确地定义该数值，则可按住 Shift 键同时拖动滑杆，以更小的增量来增加或减小该数值。



使用推子

某些属性（如音量）使用垂直滑杆或推子（而不是水平滑杆）更改值。音量控件使用垂直滑杆以更好地模拟硬件和软件混音器中的推子。您可向上或向下拖动滑杆，以增加或减小这些属性值。如果需要更精确地定义该值，则可以按住 Shift 键同时点击推子上方或下方，以更小的增量来增加或减小该数值。





备注

您可以按住 Ctrl 键同时点击滑杆或文本框的任意位置，以将属性值恢复为其默认值。

使用表格

表格通过使用数个行和列来排列数据。第一列中的信息与其它列中的信息直接相关。在 Wwise 中，表格显示事件、预设等相关信息。

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.

可以使用向上和向下箭头键浏览表格中的这些条目。大多数情况下，您可以调整列的大小、根据各个不同列标题排序信息，并可在表格内直接编辑属性值。

调整表格中的列宽的方法如下：

1. 将鼠标指针置于列分隔线上。
指针变为双箭头。
2. 执行以下操作之一：
 - 将分隔线拖至右侧可增加列宽。
 - 向左拖动分隔线可减小列宽。

在表格中配置列标题的方法如下：

1. 将鼠标指针置于列标题区上。
2. 右键点击标题区。

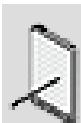
快捷方式菜单会显示其独有选项：Configure Columns...。

3. 选择 Configure Columns...。

随即显示 Configure Columns... 对话框，可以在其中指定要显示的列标题及它们的顺序。

在表格中排序信息的方法如下：

1. 在表格中点击列标题。
表格内容将根据该列中的信息按升序排序。
2. 再次点击列标题，按降序排序信息。



备注

如果表格内条目的顺序起着决定性作用，则您将无法按列标题排序。您只能将这些条目从一个位置拖至另一个位置，来手动对它们重新排序。

在表格中筛选元素的方法如下：

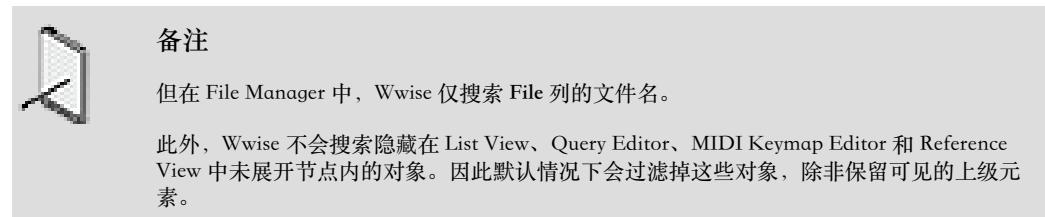
1. 点击右上角的 Search 图标。默认快捷方式：Ctrl+F3。

随即显示标准字母数字搜索栏。

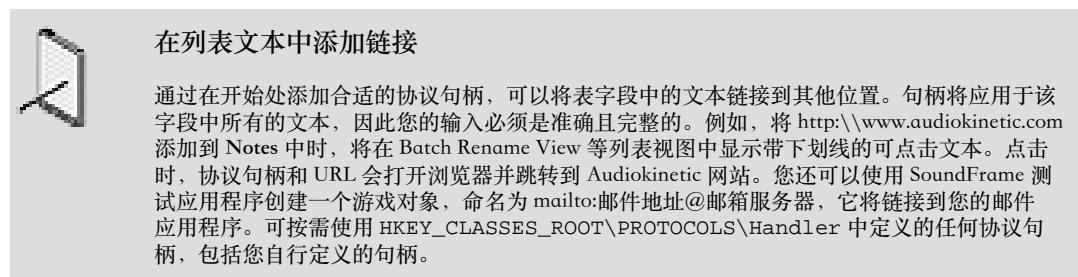


2. 在栏中输入任意字符串，包括数字、字母、下划线或空格。

Wwise 将不显示各列中都没有匹配字符的元素。列出的元素越少，越容易找到要找的。



3. 要关闭搜索栏并重置筛选器，请点击 Search 图标左侧的 Close 图标，或按下 Esc 键。



理解 Wwise 中的视觉元素

Wwise 使用多种图标或视觉标识，以帮助在界面中呈现不同的元素。以下表格说明了 Wwise 中所用的不同视觉元素。

图标	名称	代表
	Physical Folder (实文件夹)	物理文件夹。该文件夹代表磁盘上的实际目录。所有 Wwise Work Unit 都存放在 Physical Folder 内。
	Virtual Folder (虚拟文件夹)	可以在整个 Project Explorer 中添加虚拟文件夹，用来将项目元素分组。
Master Mixer 图标		
	Audio Bus (音频总线)	一种声音对象分组，您可通过它处理游戏中不同的结构。例如，您可将所有环境声结构编组为一个音频总线，将所有玩家角色声音结构编组为另一个音频总线。
	Auxiliary Bus (辅助总线)	A sub-grouping of sound objects anywhere in the project for adjusting volume, channel configuration, positioning, and RTPC, as well as applying Effects, States, or mixer plug-ins before routing back to a parent Audio Bus. 闪电、HDR 和声部混音调整无法在辅助总线上实现。
角色混音器图标		

图标	名称	代表
	Audio Source (音频源)	音频文件与对象之间的独立抽象层。它链接到工程中导入的音频文件，并且音频源层级也是设置 Conversion Settings (针对活跃游戏平台) 的位置。
	Source Plug-in (源插件)	由 Wwise 外部的源插件创建的音频源。
	Sound SFX (音效声)	包含声音效果的声音对象。
	Sound Voice (语音声)	包含旁白或角色对话的声音对象。
	Work Unit (工作单元)	可以通过独有的 XML 文件将工程拆分成独立单元，以便您团队中的不同成员同时处理工程的不同部分。这些 XML 文档可通过版本控制系统轻松管理。
	Actor-Mixer (角色混音器)	角色混音器。若干个声音、容器或角色混音器所构成的层级结构。可以使用角色混音器将某一属性应用于其下的所有对象。
	Blend Container (混合容器)	该组中的对象或容器将被同时播放。在 Blend Container 中的声音和容器能够被编组到 Blend Track (混合轨) 中，然后通过使用 RTPC 把这些声音的属性映射到游戏参数值上。在同一条 Blend Track 上的各个对象之间也可以基于游戏参数值来进行交叉淡入淡出。
	Random Container (随机容器)	该组中的对象或容器将被随机播放。
	Sequence Container (序列容器)	该组中的对象或容器将根据特定顺序或播放列表播放。
	Switch Container (切换容器)	该组中的对象或容器会被组织成与一系列切换开关或状态相符，当游戏调用相应的切换开关或状态时就会播放。
Interactive Music (互动音乐) 图标		
	Music Track (音乐轨)	可以包含多个独立音乐片段音乐对象，并以波形形式显示它们，使您能够在音乐片段中以视觉方式进行调整。
	Music Track - Random Step (音乐轨 - 随机步进)	每次播放其父段落时，将按随机顺序播放其子音乐轨。
	Music Track - Sequence Step (音乐轨 - 序列步进)	每次播放其父级段落时，都将按照顺序播放其子音乐轨。
	Music Track - Switch (音乐轨 - 切换开关)	将根据指定 Switch/State Group 中相关联的 Switch/State 来播放其子音乐轨。
	Music Segment (音乐段落)	包含音乐轨的音乐对象，可使用同步点进行调整，在互动音乐中实现音乐编排。
	Music Playlist Container (音乐播放列表容器)	由若干个段落按特定方式构成的一个编组，会按随机顺序或特定顺序播放这些段落。
	Music Switch Container (音乐切换容器)	由若干个音乐段落和容器构成的一个编组，它们被组织成与一系列切换开关或状态相符，当游戏调用相应的切换开关或状态时就会播放。
其它工程元素图标		
	Event (事件)	触发游戏中音频或振动的方法，使用一个或一系列动作（如播放、静音和暂停）来控制若干个 Wwise 对象。
	Dialogue Event (对白事件)	一种适用于对白、其他声音或振动的音频触发方式，在游戏中使用 Switch/State Group 以及 Switch/State 组合成 Path (路径)，并指派给 Wwise Actor-Mixer 中的对象。
	SoundBank (音频包)	事件、Wwise 对象和媒体的编组，可在游戏中特定时刻一起加载至游戏平台内存中。
	Switch Group (切换开关组)	将相关的切换开关进行分组，用来管理游戏内指定元素在不同条件下的替代选项。

图标	名称	代表
	Switch (切换开关)	切换选项，用于游戏内指定元素在特定条件下播放声音。
	State Group (状态组)	将相关状态进行分组，用来管理游戏环境中的全局更改。
	State (状态)	对游戏音频属性所做的全局偏置或调整，使其符合游戏中物理及环境条件的变化。
	Game Parameter (RTPC) (游戏参数 (RTPC))	游戏中的参数，如赛车游戏中的速度和 RPM，可使用 RTPC 映射至 Wwise 属性值。
	Trigger (触发器)	可由游戏中的特定操作触发播放的一小段音乐。
	Effect ShareSet (效果共享集)	音频效果插件设定，可用于增强游戏中音频效果。这些设定保存为共享集，可在工程间共享。
	Attenuation ShareSet (衰减共享集)	基于音源与听者之间相对距离而进行的音量衰减设定。这些设定保存为共享集，可在工程间共享。
	Conversion Settings ShareSet (转码设置共享集)	转码设定包括采样率、音频格式和通道数量，可帮助定义音频输出的整体品质。这些设定已保存为共享集，可在工程间应用及共享。
	Soundcaster Session (声音选角器会话)	按特定顺序集合一组声音、音乐、振动和事件模块，并与相关的游戏同步器设置一同保存，用来对游戏音频进行模拟。
	Mixing Session (混音会话)	将一组总线或对象，与其各自属性一起在调音台界面内保存，用来对游戏的混音进行微调。
	Query (查询)	一组特定的搜索条件，用于查找特定对象或对象元素。
Property Editor 分隔器		
	No split	不分隔。Property Editor 将在一个标签页中显示信息。
	Column split	列分隔。Property Editor 将在纵向面板中分左右两个标签页显示信息。
	Row split	行分隔。Property Editor 将在横向面板中分上下两个标签页显示信息。
属性值标志		
	Link	链接。属性值已链接到其它有效游戏平台的值。
	Unlink	取消链接。属性值没有链接到其它有效游戏平台的值。
	Partial Unlink	部分取消链接。当前平台的属性值已链接到其它有效平台，但其它平台的若干个相应值已取消链接。
	Link Mixed	有些选定的对象具有不同的链接状态。有些可能是链接的，而另一些则是取消链接或部分取消链接的。
	RTPC 已禁用	该属性值未绑定至游戏内参数值。
	RTPC 已启用	游戏内参数值已绑定至该属性值。这意味着，例如游戏赛车的速度可直接绑定至 Wwise 中的音调属性。当游戏中的赛车速度提高时，Wwise 中的音调也将实时提高。
	RTPC 部分启用	Multi Editor 中只有部分对象为该属性绑定了游戏参数值。Property Editor 或 Contents Editor 中不会看到这个标识。
	Randomizer 已启用	随机化器效果已应用到的属性值。

图标	名称	代表
	Randomizer 已禁用	尚未应用随机化器效果的属性值。
	随机化器 Mixed	Multi Editor 中只有部分对象为该属性值启用了 Randomizer 效果。Property Editor 或 Contents Editor 中不会看到这个标识。
	State 已禁用	此属性值未与 State 绑定。
	State 已启用	State Group 已与此属性值绑定。也就是说，所述属性（如 Volume）可能会随应用的 State 变化。
	State 混合情形	State Group 绑定到了 Multi Editor 中加载的一个或多个对象（并非全部）的属性值。Property Editor 或 Contents Editor 中不会看到这个标识。
属性底色		
	"Unsupported feature (不支持的功能)"	如果某一属性被显示为蓝色，则表示该功能在当前平台中不支持。
预设图标		
	Save Preset (保存预设)	该命令可将视图内所有值的当前状态保存至预设。
	Load Preset (加载预设)	该命令可加载之前保存的预设。
"Source Control (版本控制)"		
	"Source Control (版本控制)"	已选中控件。在已选中控件周围显示的焦点边框（在使用 Wwise Classic 主题时显示为白色）。
视图图标		
	Help (帮助)	该命令可显示特定视图、窗口或对话框的在线帮助。
	View Settings (视图设置)	显示包含一系列设置的对话框，用于定义在视图中显示哪些内容。
	Floating Indicator	表示未停靠视图。若要移动而不停靠视图，请在此处或标题栏上任意区域单击并拖动。若要停靠视图，请单击并拖动浮动视图的选项卡。
	Sync Group	同步组。该命令用于将视图切换至特定同步组 (1-4)。 该选项仅适用于 Project Explorer 和 Event Viewer 视图。
	Close View	该命令可关闭浮动视图。
Voice Inspector 图标		
	Project Value	工程值。在未连接游戏时，影响声部管线音量的基准音量值。
	SoundBank Value	音频包值。在连接游戏时，影响声部管线音量的基准音量值。
	Live Value	实时值。在连接游戏时，影响声部管线音量的属性值改变。
	API	影响声部管线音量的 API 调用。
	HDR	影响声部管线音量的 HDR 窗口改变。
	Ducking	闪避。影响声部管线音量的闪避（降低音量）行为。
	Fade	淡变。淡入或淡出：仅会在执行操作的过程中未显示时显示。
	Attenuation Distance	衰减距离。影响声部管线音量的发声体与听者之间的距离。

图标	名称	代表
	Blend Track (混合轨)	混合音轨。影响声部管线音量的混合音轨。
	Cone Attenuation	声锥衰减。影响声部管线音量的声锥角度衰减。
	Ray	射线。影响声部管线音量的发声体与听者之间的射线距离。
	声笼 (Occlusion)	声笼。影响声部管线音量的射线声笼。
	声障 (Obstruction)	声障。影响声部管线音量的射线声障。
	Volume	音量。影响声部管线音量的 Volume 属性/滑杆变化。
	Send Volume	发送音量。影响声部管线音量的 Aux Bus Send Volume 属性/滑杆变化。
	Output Bus Volume	输出总线音量。影响声部管线音量的 Output Bus Volume 属性/滑杆变化。
	Bus Volume	总线音量。影响声部管线音量的 Bus Volume 属性/滑杆变化。
	Make-up Gain	补偿增益。影响声部管线音量的音量补偿增益，在所有其他音量调节之后应用于声部。 备注 与其他音量调节不同，Make-up Gain 是在声部进行所有其他音量调节之后才应用于信号的。
	Volume Mute	音量 - 静音。Event 中影响声部管线音量的 Mute 动作。
	Volume Pause	音量 - 暂停。Event 中影响声部管线音量的 Pause 动作。
	Low Pass	低通。影响声部管线的 LPF 设置。
	Send Low Pass	发送 - 低通。影响声部管线的辅助发送总线 LPF 设置。
	Output Bus Low Pass	输出总线 - 低通。影响声部管线的输出总线 LPF 设置。
	High Pass	高通。影响声部管线的 HPF 设置。
	Send High Pass	发送 - 高通。影响声部管线的辅助发送总线 HPF 设置。
	Output Bus High Pass	输出总线 - 高通。影响声部管线的输出总线 HPF 设置。
	Multiple Properties	多项属性。影响声部管线 Volume、LPF 和 HPF 的多项属性。
	Send Multiple Properties	发送 - 多项属性。影响声部管线的辅助发送总线的多项属性。
	Output Bus Multiple Properties	输出总线 - 多项属性。影响声部管线的输出总线的多项属性。

Wwise 中的撤消和恢复操作

您可撤消您在 Wwise 中执行的大部分操作，如更改属性值、移动对象，或创建事件。如果您错误地撤消了某一操作，则可以恢复至上一个操作之前的值或状态。

要撤消某个操作, 请点击 **Edit > Undo ><操作名称>**, 或按下 **Ctrl+Z**。您最多可撤消前 200 个操作。

要恢复某一操作, 请点击 **Edit > Redo ><操作名称>**, 或按下 **Ctrl+Y**。您可为每个撤消操作调用一个恢复命令。

个性化您的工作空间

可通过多种方式自定义 Wwise 工作环境。以下表格总结了可用的自定义类型。

自定义区域	描述
处理视图	定义固定视图和浮动视图的大小和位置。
处理布局	定义显示哪种布局、布局中显示哪种视图, 以及显示位置。
设置颜色	定义 Wwise 工程的色块标志。
设置用户偏好	定义您独有的用户偏好。

处理视图

Wwise 界面由许多视图构成。视图是界面中独立的窗口, 包含一组与特定任务相关的信息或命令。

由两种不同类型的视图: 浮动视图和固定视图。固定视图停靠在布局中, 而浮动视图就像窗口一样, 可在布局上浮动。这两种类型的视图都可重新定位和调整大小。

显示视图的方法如下:

1. 在菜单栏中, 点击 **Views > <视图名>**。

选择的视图显示为浮动视图。

切换浮动视图的方法如下:

1. 在菜单栏中, 点击 **Windows > <视图名>**。

选择的视图显示在堆叠的浮动窗口前方, 成为活跃视图。

相关主题

- [同步一组视图](#)
- [理解视图的标题栏图标](#)
- [管理布局内的浮动视图](#)
- [调整布局内视图的大小](#)
- [在布局中添加/移除视图](#)
- [处理布局](#)

理解视图的标题栏图标

视图在标题栏的右侧可包含一系列图标。各个图标执行与该视图相关的不同命令。根据视图的不同, 会显示不同的图标组合。以下表格对视图的标题栏中显示的各个图标进行说明。

图标	名称	描述
	Floating Indicator	表示未停靠视图。若要移动而不停靠视图，请在此处或标题栏上任意区域单击并拖动。若要停靠视图，请单击并拖动浮动视图的选项卡。
	Save Preset (保存预设)	保存预设。将当前视图内所有值得当前状态保存至预设。
	Load Preset (加载预设)	加载预设。加载之前保存的预设。
	Sync Group	同步组。将视图同步至特定的同步组 (1-4)。将视图同步至某一同步组时，该组内所有视图将总是显示相同的选项卡/所选对项。例如，比如说您将三个不同布局内的三个不同的 Project Explorer 全部同步至组 1。如果您切换至其中一个 Project Explorer 的 Game Syncs 选项卡，则该组内所有 Project Explorer 视图也将自动切换至 Game Syncs 选项卡。 该选项仅适用于 Project Explorer 和 Event Viewer 视图。
	Help (帮助)	帮助。显示该特定视图、窗口或对话框的在线帮助。
	View Settings (视图设置)	视图设置。显示包含一系列设置的对话框，用于定义在视图中显示哪些内容。
	Close View	关闭视图。关闭浮动视图。

管理布局内的浮动视图

在使用特定布局时，您可随时打开许多不同的浮动视图。您可使用窗口管理器高效地管理这些浮动视图。在窗口管理器中，您可轻松激活特定的浮动视图、最大化、最小化和恢复视图、移动视图及关闭视图。

管理布局内的浮动视图的方法如下：

1. 在菜单栏中，点击 Windows > Windows...。

随即打开 Windows 对话框。

2. 从 View Name 列表中，选择若干个视图，然后点击以下任一选项：

- Activate：激活所选浮动视图，并将其显示在所有窗口的前方。
- Minimize：最小化所选的浮动视图。
- Maximize：最大化所选的浮动视图。
- Restore：将之前已执行最小化或最大化操作的所选浮动视图恢复至原始大小和位置。
- Move To...：将所选的浮动视图移至显示范围内的新位置。
- Close Window(s)：关闭所选的浮动视图。

3. 完成后，请点击 Close 以关闭 Windows 对话框。

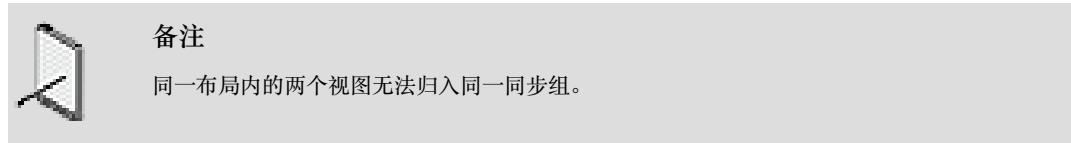
相关主题

- [处理视图](#)
- [调整布局内视图的大小](#)
- [在布局中添加/移除视图](#)

- 将出厂布局恢复至其默认配置
- 理解视图的标题栏图标

同步一组视图

由于您可能在多个布局中创建了不同的 Project Explorer 和 Event Viewer 实例，所以可能会需要同步这些视图。这表示在视图内执行的任何选择或移动都会同步至其它视图。



同步一组视图的方法如下：

1. 从 Project Explorer 或 Event Viewer 标题栏中，点击 Sync Group 图标。
2. 从快捷菜单中，选择以下任一选项：
 - No sync group: 不将视图同步至任何组。
 - Sync Group 1: 将视图同步至组 1。
 - Sync Group 2: 将视图同步至组 2。
 - Sync Group 3: 将视图同步至组 3。
 - Sync Group 4: 将视图同步至组 4。
3. 要同步其它布局中的 Project Explorer 和 Event Viewer，只需切换布局并重复步骤 1-2。

相关主题

- 处理视图
- 调整布局内视图的大小
- 管理布局内的浮动视图
- 理解视图的标题栏图标
- 在布局中添加/移除视图
- 在布局中停靠/取消停靠视图
- 将出厂布局恢复至其默认配置

处理布局

可将一组视图排列在一起，以创建布局。Wwise 包含 8 个不同的默认布局，这些布局已经过优化，以帮助您执行某些任务或作业。您可根据需要处理的任务，使用不同的布局。

Wwise 中提供以下布局：

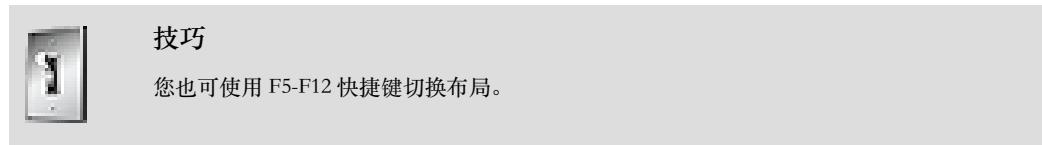
- Designer: 用于管理、构建和定义构成游戏中音效、对话和振动部分的资源。
- Profiler: 用于监视和分析游戏中出现的各音频和振动元素的性能。
- SoundBank: 用于创建、管理和生成工程的SoundBank。
- Mixer: 用于在 Wwise 中创建原型，并对各种对象进行混音。

- **Schematic:** 用于查看工程结构的图形概览。
- **Interactive Music:** 用于管理、构建和定义构成游戏交互音乐部分的音乐素材。
- **Voice Profiler:** 用于监控和分析游戏中的各个声部。
- **Game Object Profiler:** 用于从游戏对象的角度来监视和分析音频表现。
- **Dynamic Dialogue:** 用于管理和构建游戏中驱动动态对话的对白事件。

切换布局的方法如下：

1. 从菜单栏中，点击 **Layouts > Layout Name**。

随即显示新布局。



相关主题

- [处理视图](#)
- [调整布局内视图的大小](#)
- [在布局中添加/移除视图](#)
- [在布局中停靠/取消停靠视图](#)
- [将出厂布局恢复至其默认配置](#)

调整布局内视图的大小

您可在布局内，通过调整已有视图的大小来修改默认布局。

调整固定视图的大小的方法如下：

1. 将鼠标指针放置在两个或多个视图之间的分隔栏上。
分隔栏变为高亮显示，鼠标指针变为双头箭头。
2. 拖动分隔栏，以调整高亮显示的分隔栏两侧所有视图的大小。

相关主题

- [在布局中添加/移除视图](#)
- [在布局中停靠/取消停靠视图](#)
- [将出厂布局恢复至其默认配置](#)
- [处理布局](#)
- [处理视图](#)

在布局中添加/移除视图

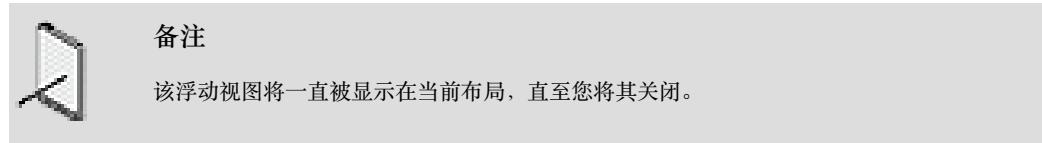
虽然 Wwise 中的出厂默认布局已经过优化，能够尽可能提升工作流程的效率，但您仍可以在不布局中添加或移除视图。无论是浮动视图还是固定视图，都属于布局的一部分，直至您将其移除。

向布局添加新视图的方法如下：

1. 在菜单栏中，点击 Views > <视图名>。

选择的视图在布局内显示为浮动视图。

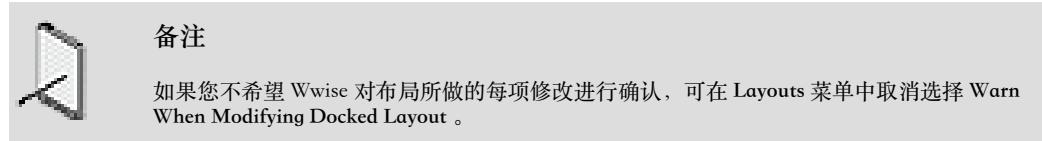
2. 您可调整浮动视图的大小，并可将其移至显示范围内的任意位置。



从布局中移除视图的方法如下：

1. 在浮动视图或固定视图的标题栏中，点击 Close 图标 (X) 可关闭浮动视图。

如果视图停靠在布局内，则 Wwise 将提示您确认布局修改。



2. 点击 Yes。

停靠视图会从布局中移除，余下视图的大小会调整为充满剩余空间。

相关主题

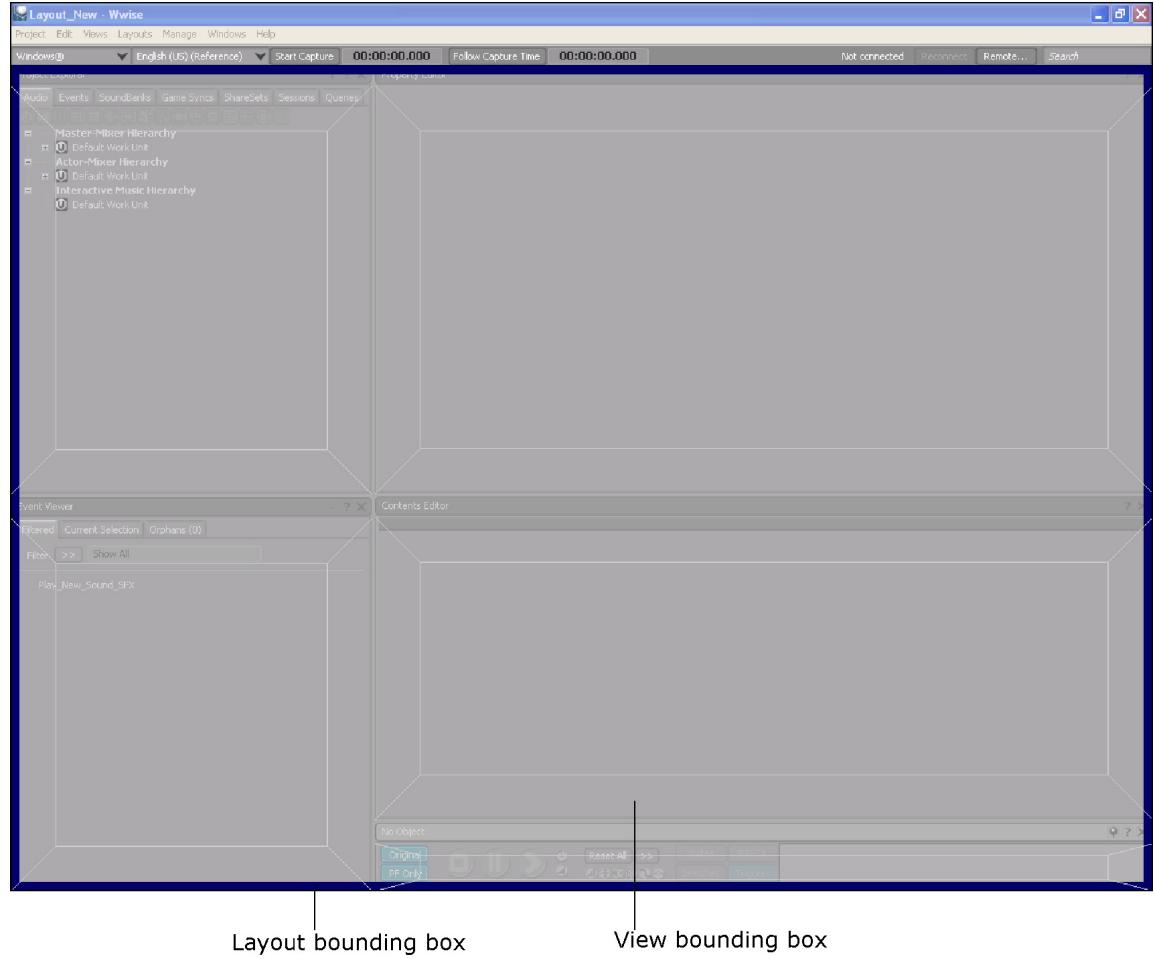
- [在布局中停靠/取消停靠视图](#)
- [调整布局内视图的大小](#)
- [将出厂布局恢复至其默认配置](#)
- [处理布局](#)
- [处理视图](#)

在布局中停靠/取消停靠视图

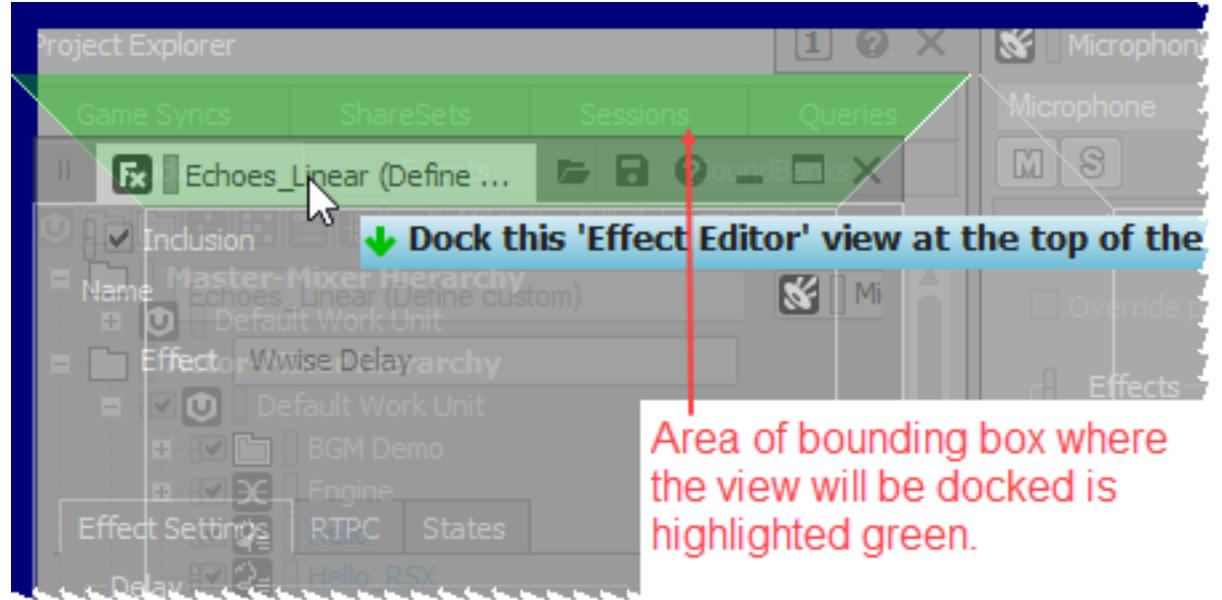
虽然默认布局已经过设计，能够尽可能提升工作流程效率，但您仍可编辑布局内视图的内容和位置。

将视图停靠至布局时，可以选择将新视图停靠在多个位置。当 Wwise 进入“编辑布局”(edit layout) 模式时，会根据布局内区分视图的各种分隔栏将布局划分为不同的区域。同时会围绕以下范围创建单独的边框：

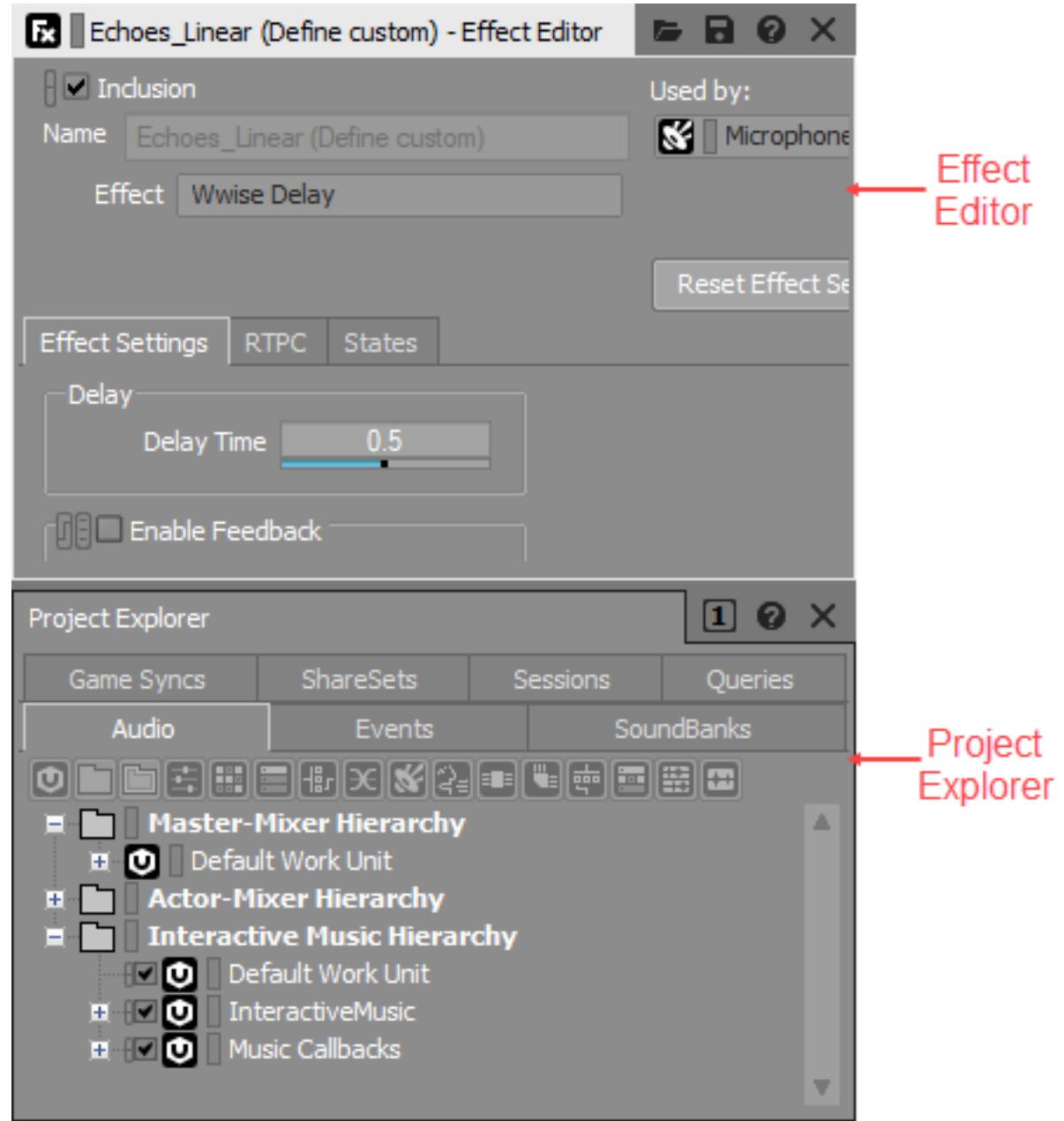
- 整体布局（由粗蓝线表示）。
- 布局内的各个视图。



每个边框的四边表示可将新视图停靠在布局内的不同位置。例如，您希望将 Effect Editor 添加至 Designer 布局中 Project Explorer 的正上方。可以打开 Effect Editor，然后将其拖至 Project Explorer 边框的顶部。



该操作将先前 Project Explorer 的区域分隔为两部分，Effect Editor 位于上半部，Project Explorer 位于下半部。



如果您希望 Effect Editor 占据布局的整个宽度，则可以使用顶部布局边框，而非视图边框。

在布局内停靠浮动视图的方法如下：

1. 在浮动窗口内单击视图的标题选项卡，并拖动视图。

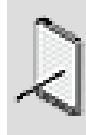
此时界面进入“编辑布局”模式。

2. 将浮动视图拖至布局内所要停靠的边框区域。

边框区域高亮显示为绿色。

3. 释放鼠标按钮。

Wwise 将提示您确认布局修改。



备注

如果您不希望 Wwise 对布局所做的每项修改进行确认，可在 **Layouts** 菜单中取消选择 **Warn When Modifying Docked Layout**。

4. 点击 Yes。

浮动视图在布局内指定位置变为固定视图。

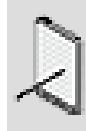
5. 现在可以调整固定视图的大小，以获得您所希望的布局。

在布局中取消停靠视图的方法如下：

1. 对于要取消停靠的视图，点击其标题栏，然后开始拖动视图。

Wwise 进入“编辑布局”模式。

2. 将视图拖至布局内的任意位置，确保没有边框高亮显示为绿色。

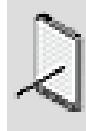


备注

如果某一边框高亮显示为绿色，则该视图会移动并停靠至布局内的新位置。

3. 释放鼠标按钮。

Wwise 将提示您确认布局修改。



备注

如果您不希望 Wwise 对布局所做的每项修改进行确认，可在 **Layouts** 菜单中取消选择 **Warn When Modifying Docked Layout**。

4. 点击 Yes。

固定视图即变为浮动视图。标题栏左侧会显示浮动标志，右侧会显示标题选项卡。

相关主题

- [调整布局内视图的大小](#)
- [在布局中添加/移除视图](#)
- [将出厂布局恢复至其默认配置](#)
- [处理布局](#)
- [处理视图](#)

将出厂布局恢复至其默认配置

对布局进行编辑后，您可能会决定将其恢复至默认布局配置。Wwise 可以随时恢复默认布局。请注意，目前无法保存已编辑的布局。因此，当您恢复出厂默认布局时，将丢失对布局所做的更改。

要将出厂默认布局恢复至其默认配置的方法如下：

1. 从菜单栏中，点击 **Layouts > Reset Factory Layouts**。

Reset Factory Layouts 对话框将打开。

2. 选择要恢复至其默认配置的布局。
3. 点击 **OK**。

此时选择的出厂布局将恢复至其默认配置。

相关主题

- [调整布局内视图的大小](#)
- [在布局中添加/移除视图](#)
- [处理布局](#)
- [处理视图](#)

设置颜色

每个 Wwise 对象的右侧均设有色块，默认设为空白（灰色）。您可以根据需要将其更改为其他颜色（共 26 种）。这样方便快速查看和理解工程组织结构。

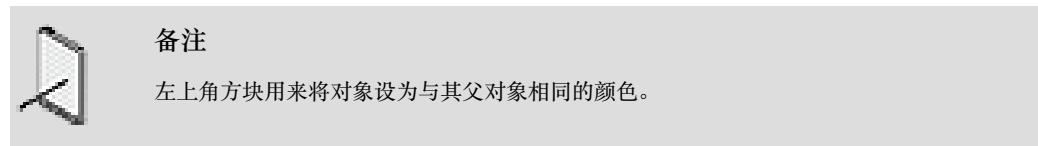
设置对象色块的方法如下：

1. 在 Project Explorer（工程资源管理器）内，选择一个或多个对象。
2. 打开[快捷菜单](#)，并单击 **Set Color...**（设置颜色...）。

这时将显示颜色选择器。



3. 单击所需颜色对应的方块。



所选对象的色块将被填充。

此外，还可将 Multi Editor（多项编辑器）内所加载全部对象的色块设为同一种颜色。

色块用例如下：

- 状态：您可以按照各项的工作状态来组织工程。比如：
 - 已锁定 = 黄色

- 待处理 = 绿色
- 已受阻 = 红色
- 已完成 = 蓝色
- 团队：您可以按照团队职责来组织工程。比如：
 - 脚本作者 = 黄色
 - 本地化人员 = 绿色
 - 作曲家 = 红色
 - 声音设计师 = 蓝色
- 类型：您可以按照声音的类型来组织工程。比如：
 - 环境声 = 黄色
 - 对白 = 绿色
 - SFX = 红色

设置用户偏好

您可对用户偏好进行设置，以自定义 Wwise 的以下方面：

- 启用确认信息
- 选择 External Audio Editors
- 设置输出缓冲区延迟
- 设置首选音频设备
- 设置音频通道配置
- 设置音乐轨预读时间
- 设置文档偏好

启用确认信息

当您在 Wwise 中执行特定任务时，确认信息可提供实用信息。使用 Wwise 时，您可通过在特定确认对话框中选择“Don't ask again”选项，来禁用这条信息。如果要再次显示所有这些信息，则请在用户偏好中重新启用它们。

启用确认信息的方法如下：

1. 从菜单栏中，点击 Project > User Preferences。

User Preferences 对话框将打开。

2. 在 Confirmation Messages 分组框中，点击 Reset。

此时显示确认对话框。

3. 点击 OK。

从现在起，在 Wwise 中执行相关操作前将显示确认信息。

选择 External Audio Editors

在 Wwise 中，您可以选择直接在任何音频文件编辑器中编辑音频文件。但首先，您需要将各个程序添加至可用编辑器列表。

选择外部音频编辑软件的方法如下：

1. 从菜单栏中，点击 Project > User Preferences。

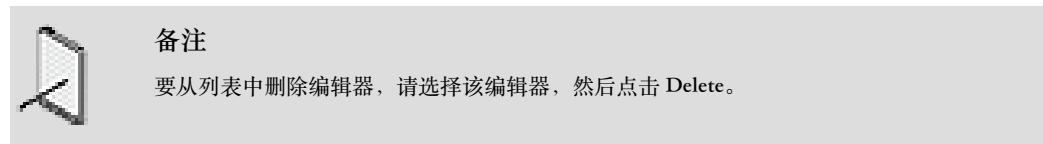
User Preferences 对话框将打开。

2. 在 External Editors 分组框中，点击 Add...。

此时将会打开 Open File 对话框。

3. 选择音频编辑器可执行文件 (EXE)，然后点击 Open。

此时会将编辑器添加至可用音频编辑器列表。



4. 点击 OK，保存您的设置。

从现在开始，就可以直接从 Wwise 中打开已添加的编辑器。

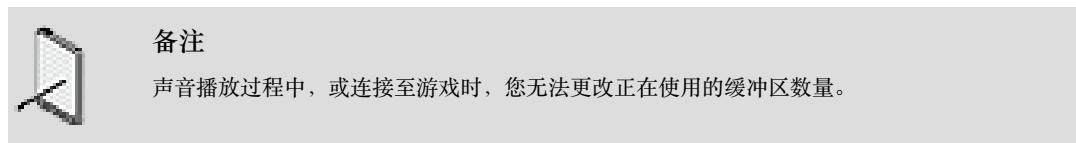
选择默认外部音频编辑器的方法如下：

1. 在编辑器列表中，选择要设为默认选项的编辑器。
2. 点击 Set As Default。 (如果列出的编辑器少于两个，此按钮将不可用。)

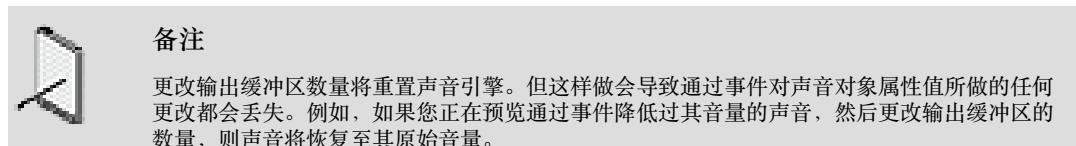


设置输出缓冲区延迟

当您在 Wwise 中播放声音时，Wwise 声音引擎会使用预填充的缓冲区，以减少播放延迟。默认情况下，声音引擎使用其中四个缓冲区。但您可以调整声音引擎使用的缓冲区数量，以纠正使用 Wwise 时可能会遇到的某些性能问题。



如果您在 Wwise 中播放声音时，发现有 voice starvation 的现象，则请尝试增加输出缓冲区的数量。而如果您希望减小播放延迟，则请尝试减少缓冲区的数量。



设置输出缓冲区延迟的方法如下：

1. 在菜单栏中，点击 Audio > Audio Preferences。

Audio Preferences 对话框将打开。

2. 在 Sound Engine 区域框中，从列表中选择输出缓冲区的数量。括号中的数字表示相应的延迟。
3. 点击 OK，保存您的设置。

从现在开始，在 Wwise 中播放声音时，声音引擎将使用您指定的输出缓冲区数量。

设置首选音频设备

默认情况下，Wwise 中播放的所有声音都将由 Windows 默认播放设备输出。您可以直接在 Wwise 中更改输出设备，以及在设计工具中试听时使用哪个 Audio Device 共享集。

更改输出设备的方法如下：

1. 在菜单栏中，点击 Audio > Audio Preferences。

Audio Preferences 对话框将打开。

2. 在 Hardware Devices 中，选择要使用的设备。
3. 点击 OK，保存您的设置。

从现在起，声音引擎将始终把相应总线输出到所选设备。

设备列表中列出了当前活跃的 Audio Device 插件所支持的所有设备。如果一个硬件设备可以通过不同的插件访问，则该设备可能会在列表中出现多次。当设置为默认值时，设备将由总线使用的 Audio Device 插件选择。如果该 Audio Device 插件在 Windows 上不可用，将改为使用 Windows 的默认音频设备。这个选项仅适用于当前用户，而非整个游戏。

设置音频通道配置

系统默认通道配置

- 从菜单栏中，点击 Audio > System Default Channel Configuration。
- 默认情况下，Wwise 使用 Windows 控制面板中的扬声器配置方案。选择该选项，即使用 Windows 控制面板中选择的数值。

立体声声道配置（扬声器）

- 在菜单栏中点击 Audio > Stereo Channel Configuration (Speakers)。
- 有关声像摆位规则（扬声器、耳机）的详细信息，请查阅 [声像摆位规则：扬声器对耳机](#)。

立体声声道配置（耳机）

- 在菜单栏中点击 Audio > Stereo Channel Configuration (Headphones)。

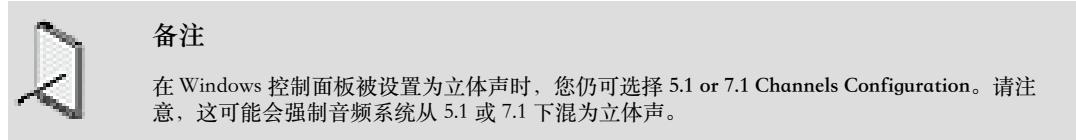
- 有关声像摆位规则（扬声器、耳机）的详细信息，请查阅 [声像摆位规则：扬声器对耳机](#)。

5.1 Channel Configuration

- 在菜单栏中点击 **Audio > 5.1 Channel Configuration**。

7.1 Channel Configuration

- 在菜单栏中点击 **Audio > 7.1 Channel Configuration**。



设置音乐轨预读时间

在 Wwise 中编辑创作时，音频总是以流播放（streaming）形式播放。如果您在 Music Track Property Editor 中已指定流播放并已定义音乐轨，则在流播放过程中将使用这些值。如果您未选择流播放选项，则默认音乐轨预读时间将使用 200 毫秒。要在播放过程中提高灵活性及避免同步和 voice starvation 问题，您可以在为用户偏好中设置音乐轨预读时间。

设置音乐轨预读时间的方法如下：

- 在菜单栏中，点击 **Audio > Audio Preferences**。

User Preferences 对话框将打开。

- Sound Engine 区域框中，在 **Music Track look-ahead time (ms)** 文本框里输入预读时间量（单位为毫秒）。
- 点击 **OK**，保存您的设置。

从现在开始，在 Wwise 中播放尚未定义流播放设置的音乐轨的音乐时，声音引擎将使用该预读时间量寻找流播放数据。

设置文档偏好

Wwise Help 可设置为打开本地 CHM 文件或打开 audiokinetic.com 提供的帮助文档在线版本。这两种文档均提供中文、英文和日文版本。

设置 Wwise 文档偏好的方法如下：

- 从菜单栏中，点击 **Project > User Preferences**。

User Preferences 对话框将打开。

- 在 **Documentation** 分组框中，从 **Source** 选项中选择 CHM，以使用存储在本地的 Wwise 帮助文件获取上下文帮助，也可选择 audiokinetic.com 以使用 Audiokinetic 在线网络文档获取上下文帮助。
- 从 **Language** 选项中选择 English, >Japanese> 或 Simplified Chinese。

选中语言将在 Wwise Help (包括上下文帮助) 和 Wwise 用户指南中应用。

4. 点击 OK, 保存您的设置。

提供工作效率

随着对 Wwise 熟悉程度的加深，您可以开始使用一些更高级的功能。您可通过以下工具快速访问某些操作和命令，从而提高您的工作效率：

- [使用键盘导航](#)
- [使用键盘快捷方式](#)
- [使用快捷菜单](#)
- [使用批量重命名](#)

使用键盘导航

Wwise 设计工具可完全通过“单击”、“拖动”和“滚动”等鼠标操作来操控。不过，有时使用键盘在设计工具中导航会更为便捷。

若要使用按键在 Wwise 内导航，可按下：

- Tab：从某一控件转到下一控件；
- Shift + Tab：返回上一控件；
- Enter：激活所选控件；
- 方向键：选择列表选项或切换单选按钮。



备注

在 Project Explorer (工程资源管理器) 中高亮显示要重命名的对象时，可使用 Tab 和 Shift + Tab 按钮转到下一和上一对象。若跳转对象可以重命名，则会将其高亮显示。若其不可重命名，则停止高亮显示和键盘导航。

当然，您也可以使用键盘来完成各种 Wwise 快捷操作（详见[“使用键盘快捷方式”](#)）。

使用键盘快捷方式

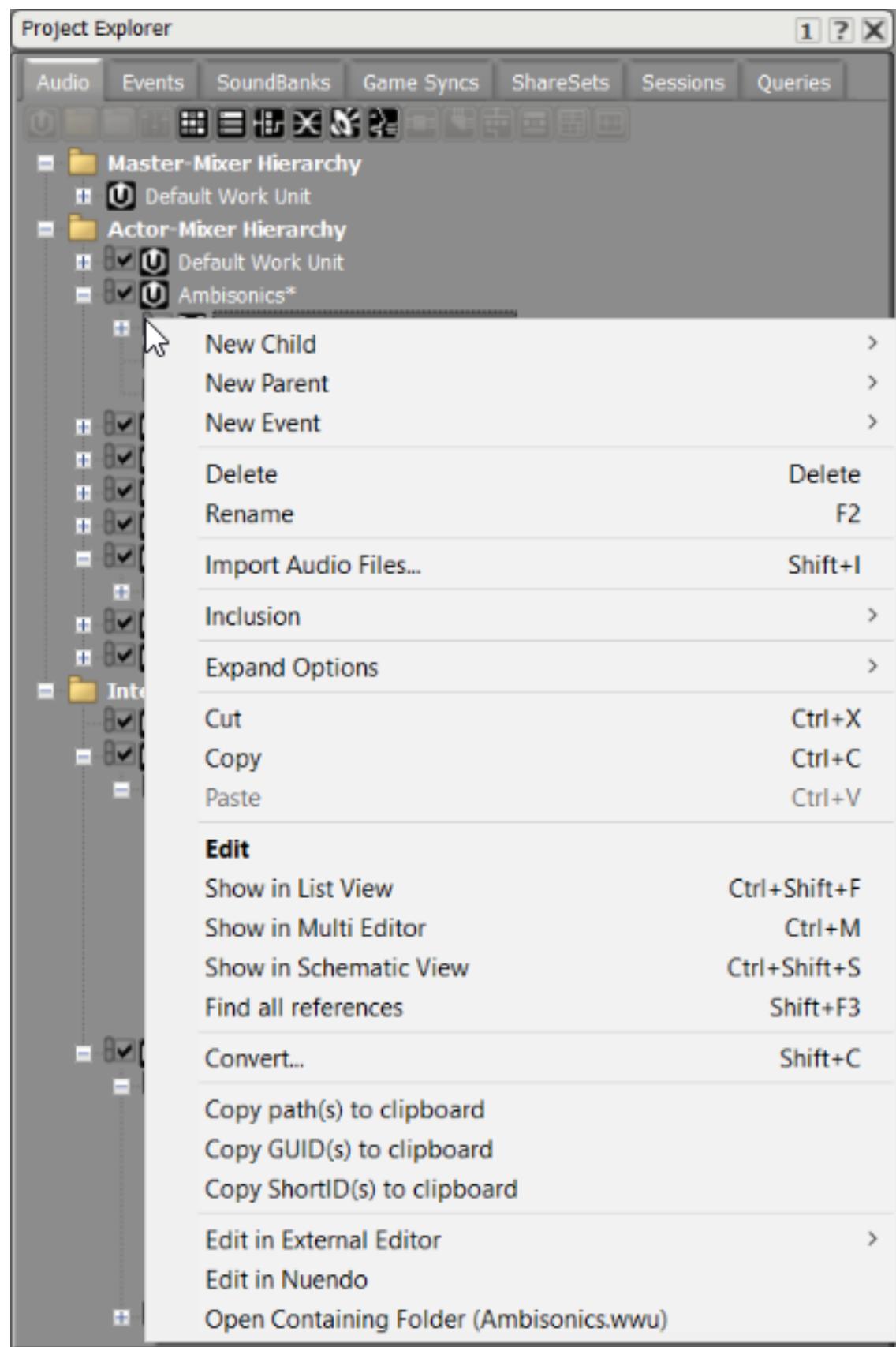
Wwise 中的许多命令或操作在键盘上都有对应的按键或按键组合。您可使用键盘快捷方式来替代鼠标，以执行任意操作或命令。例如，您可同时按下 Ctrl 和 S 键以保存您的工程，而无需使用鼠标点击 File > Save。

要获取键盘快捷方式的完整列表，请参阅[附录 C, 快捷键](#)。

使用快捷菜单

快捷菜单是 Wwise 界面中与对象、工程元素或特定区域相关的一组命令。您可通过右键点击某个对象或界面的特定区域来访问这些菜单。

例如，当您在 Project Explorer 中右键点击某个对象时，会显示一个包含多个命令的快捷菜单。您可使用这些命令完成剪切、复制或粘贴某个对象、创建父对象或子对象、创建与特定对象相关联的事件、转换声音对象等操作。



标准快捷菜单命令

虽然这些菜单是与上下文相关的，但很多命令会显示在多个菜单中。除剪切、复制、粘贴、删除和重命名等标准 Windows 命令外，通过右键点击界面中不同的对象或区域，您还可访问很多下列 Wwise 特定命令。

- **Convert to Sound Voice… / Convert to Sound SFX…**: 将对象由 Sound SFX 转换为 Sound Voice（反之亦可）。有关详细信息，请参阅[声音对象类型转换](#)。
- **Set Color…**: 打开颜色菜单，以便选择对象色块的颜色，从而清晰地组织工程。
- **Edit**: 将选择的对象或工程元素加载至其相应的编辑器内。
- **Find in Project Explorer**: 在 Project Explorer 中高亮显示选择的对象或工程元素。使用该命令时，您还必须指定 Project Explorer 所属的 Sync Group（如果有的话）。
- **Show in List View**: 在 List View 中显示所选对象。
- **Show in Multi Editor**: 在 Multi Editor 中显示所选对象。
- **Show in Schematic View**: 在 Schematic View 中显示所选对象。
- **Find All References**: 显示包含当前对象的直接引用的工程元素列表。直接引用元素列表显示在 Reference 视图中。
- 仅在按住 Shift 键同时打开快捷菜单时，以下选项才可见：
 - **Copy path(s) to clipboard**: 将所选对象（可多选）的完整路径复制到剪贴板。
 - **Copy GUID(s) to clipboard**: 将所选对象（可多选）的专有 GUID 复制到剪贴板。
 - **Copy ShortID(s) to clipboard**: 将所选对象（可多选）的专有 ShortID 复制到剪贴板。
 - **Edit in external editor**: 打开外部编辑器列表（如果在 User Preferences 中已设置）。
 - **Edit in Nuendo**: 打开 Nuendo 中的元素。只有安装了 Nuendo Game Audio Connect 插件并正确设置了 Nuendo Connection Settings …（可在 Project 菜单中打开），且 Nuendo 当前正在运行时，该选项才可用。
 - **Create in Nuendo**（仅适用于音乐片段对象）：弹出 Steinberg Hub，可以在其中创建新的 Nuendo 项目。Nuendo 将弹出窗口，询问您 Do you want to import the transferred Segment into this Project（是否要将转换的片段导入此工程？）点击 Yes 将音乐片段添加到 Project Zone。
 - **Open containing folder**: 将在 Windows 资源管理器（或 Mac Finder）中打开该对象的父级元素位置。

使用批量重命名

通过 Batch Rename 视图，执行一个步骤就能将可编辑的对象重命名，或更改其相关联的备注，该命令功能强大，可同时执行替换、移除和插入机制。单独重命名对象操作耗时长、单调乏味，并容易出错，批量重命名则消除了上述弊端。

以下页面提供了使用 Batch Rename 视图的说明和示例。

- [打开 Batch Rename 视图](#)
- [指定批量重命名设置](#)

- [应用批量重命名更改](#)

打开 Batch Rename 视图

可以从多个位置打开 Batch Rename 视图，视图中可能加载也可能不加载要重命名的对象。

显示不包含对象的 Batch Rename 视图的方法如下：

1. 选择 Views > Batch Rename 或使用快捷方式。默认快捷方式：Ctrl + F2。

Batch Rename 视图打开一个空白的 Preview 面板。

显示加载已选择对象的 Batch Rename 视图的方法如下：

1. 在 Project Explorer（或其它对象视图，如 List View）中选择要对其执行重命名操作的可编辑对象。
2. 从快捷菜单（右键点击）中或使用键盘快捷方式选择 Batch Rename...。

此时打开 Batch Rename 视图，其 Preview 面板已加载所选对象。

将对象添加至 Batch Rename 视图的方法如下：

1. 打开 Batch Rename 后，在 Project Explorer（或其它对象视图）中选择要对其执行重命名操作的可编辑对象。
2. 从快捷菜单中选择 Batch Rename...，也可使用键盘快捷方式，或将选择的对象拖至 Preview 面板上。

选择的对象将显示在 Preview 面板中，并替换之前选择的对象。



相关主题

- [指定批量重命名设置](#)
- [应用批量重命名更改](#)

[指定批量重命名设置](#)

[打开 Batch Rename 视图](#) 后，在 Batch Rename 视图的 Settings 面板中选择的更改动作，将应用至 Preview 面板中列出的对象。

Settings 面板包含以下三部分：

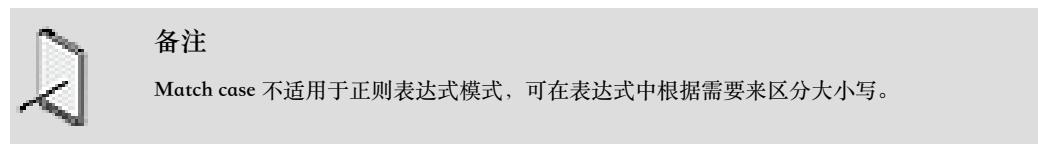
- Replace：替换内容，请参阅 [替换内容](#) 了解详细信息。
- Remove：移除内容，请参阅 [移除内容](#) 了解详细信息。
- Insert：插入内容，请参阅 [插入内容](#) 了解详细信息。

替换内容

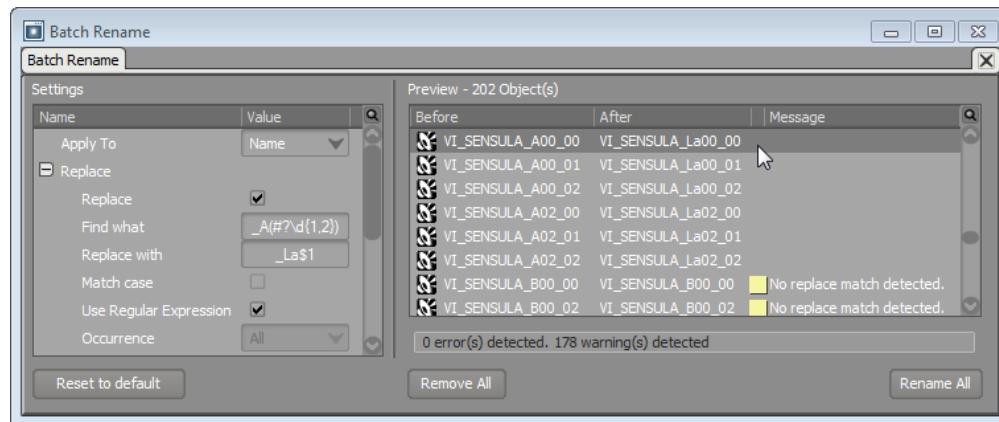
Replace 部分是 Batch Rename Settings 面板内三个部分中的第一个部分。在此部分中，用户可使用从非常简便到极为复杂的替换方式，来替换对象名称或备注中的文本。

替换内容的方法如下：

1. 请在 Apply To 列表中指定该重命名操作将应用至所列对象的名称还是备注。
2. 选择 Replace。
3. 在 Find what 栏中以任意正则表达式模式输入文本进行查找。
4. 在 Replace with 栏中以任意正则表达式模式输入替换文本。
5. 如果您希望将 Find what 条目设置为仅查找具有相同大小写（大写或小写）字母的文本，则请选择 Match case。



6. 如果您希望 Wwise 将 Find what 和 Replace with 条目理解为正则表达式语法，则请选择 Use Regular Expression。
7. 在 Occurrence 列表中指定为特定文本找到多个匹配项时，该执行何种操作。
 - First：仅包含首个匹配项。
 - Last：仅包含最后一个匹配项。
 - All：包含对象名称或备注中的各个匹配项。
8. 检查您所做的更改，然后点击 Rename All 以替换内容。



示例：替换正则表达式

以 Wwise Sample Project 为例。在其 Actor-Mixer Hierarchy 中，>MIDI Work Unit 包含很多 MIDI 对象，其部分名称以与其相应的绝对音符命名：C00、D00、E00、F00、G00、A01、B01、C02 等。假设您要将该工程传输至您在法国的录音室，但在那里习惯使用固定唱名音符：Do、Re、Mi、Fa、Sol、La、Si、Do 等。如果我们能将所有这些对象重命名，则法国录音室设计师将对此深表谢意。可以：

1. 按下 Ctrl 并点击 MIDI Work Unit (MIDI 工作单元)。

MIDI Work Unit 层级展开，显示其众多对象。

2. 在 Project Explorer 中选择所有 Kalimba、Sansula、Voice 和 Wood Sansula 对象。然后打开快捷菜单并选择 Batch Rename...。

此时显示 Batch Rename 视图及在其 Preview 面板中选择的对象。

3. 在 Settings 面板中，将 Apply To 设置为 Name (如尚未设置)。

Preview 面板在 Before 列下列出当前对象名称。

4. 启用 Replace，在下面几行中启用 Use Regular Expression。

Find what 和 Replace with 栏激活，将接受条目并解释为正则表达式。

5. 在 Find what 栏中输入 _C(#?\d{1,2})。然后点击 Preview 面板，以查看预计结果。

所有名称中包含 _C，有或没有 #，后接一个或两个数字的对象都将被列在 After 列中，但匹配的字符将不显示。针对一些对象会显示错误信息“Resulting name is already used by a sibling object or is otherwise reserved”（重命名结果已被同级对象使用，或不可用）。而针对所有其它对象将显示“No replace match detected”信息。

6. 在 Replace with 栏中输入 _Do\$1。然后点击 Preview 面板，以查看预计结果。After 列现在在 C 的位置列出 Do，而并非仅将匹配的 C 字符移除。

7. 点击 Rename All 以应用更改。

对于各个匹配对象，会在 Preview 面板中显示名称更新和 Successfully renamed 信息。

我们可对余下的六个音符重复该操作。

关于我们的正则表达式、_C(#?\d{1,2})，及其替换、_Do\$1 是如何实现的，以下表格提供了更为详细的说明。

Before	After	说明
VI_VOICE_33_C#6	VI_VOICE_33_Do#6	<ul style="list-style-type: none">_C 与名称中的 _C 在字面上匹配，就像非正则表达式替换那样。#?, 与 # 相匹配或不匹配，# 是匹配项。\d{1,2}，一个或两个数字，6 是匹配项。#?\d{1,2} 两边的括号将匹配项 # 和 6 放入一组。 <p>在替换中：</p> <ul style="list-style-type: none">匹配项替换为 _Do，并且第一组 \$1 的值为 #6。
VI_WOOD_SANSULA_C00_HI	VI_WOOD_SANSULA_Do00_HI	<ul style="list-style-type: none">_C 与名称中的 _C 在字面上匹配，就像非正则表达式替换那样。#?, 与 # 相匹配或不匹配，此例无匹配项。\d{1,2}，一个或两个数字，00 是匹配项。

Before	After	说明
		<ul style="list-style-type: none"> • <code>#?\d{1,2}</code> 两边的括号将匹配项 00 放入一组。 <p>在替换中：</p> <ul style="list-style-type: none"> • 匹配项替换为 _Do， 并且 • 第一组 \$1 的值为 00。

相关主题

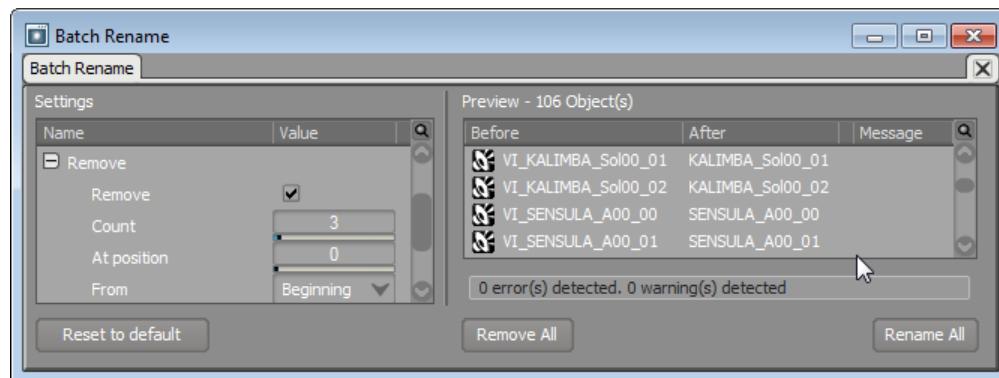
- [打开 Batch Rename 视图](#)
- [指定批量重命名设置](#)
- [应用批量重命名更改](#)

移除内容

Remove 部分是 Batch Rename Settings 面板内三个部分中的第二个部分。在此部分中，用户可使用简单的字符定位系统移除对象名称或备注中的文本。

移除内容的方法如下：

1. 请在 Apply To 列表中指定该重命名操作将应用至所列对象的名称还是备注。
Preview 面板中的 Before 和 After 列将显示所选择的对象内容。
2. 启用 Remove。
Count (计数)、At position (定位) 和 From (起点) 将被激活。
3. 在 Count 栏中输入整数，或将其滑杆移至所需值以指定要移除的字符数量。
4. 指定要执行移除操作的字符位置。为此请在 At position 栏中输入整数，或将滑杆移至所需值。
5. 选择应从哪个方向计数 At position：
 - (名称或备注的) Beginning (起始位置)
 - (名称或备注的) End (结束位置)
6. 检查您所做的更改，然后点击 Rename All 以移除内容。



示例：从匹配项中移除

以 Wwise Sample Project 为例，假设 Sample Project 的主管决定通过删除额外信息来缩短工程的对象名称。在本例中，Kalimba、Sansula、Voice 和 Wood Sansula 混合容器下的所有 SFX 都带有前缀“VI_”，同虚拟乐器一样。这不是定义这些对象的关键因素，因此我们决定将其移除。

1. 打开 Batch Rename 视图，然后从 Project Explorer 中将这些混合容器的所有 SFX 拖至 Preview 面板。

此时列出 106 个以“VI_”开头的对象。（如未列出，请检查 Apply To 列表是否设置为将此批量重命名应用至对象的 Name，而非其 Notes。）

2. 移动至 Settings 的第二个层级，然后启用 Remove。
3. 在 Count 中输入 3。
4. 将 At position 指定为 0。
5. 在 From 列表中选择 Beginning。

这包含名称的前三个字符，即“VI_”。（如果您仅希望移除“VI”与乐器名称之间的下划线，则 Count 应为 1，而 At position 为 2。）

6. 点击 Preview 面板。

After 列进行更新，以显示所有名称中将不再包含“VI_”，如 SENSUSLA_A00_00。

7. 点击 Rename All 以应用更改。

Batch Rename 中各个对象的名称都会更新，并会在 Preview 面板中显示 Successfully renamed 信息。

相关主题

- [打开 Batch Rename 视图](#)
- [指定批量重命名设置](#)
- [应用批量重命名更改](#)

插入内容

Insert 部分是 Batch Rename Settings 面板内三个部分中的第三个部分。在此部分中，用户可使用简单的替换和字符定位，在对象名称或备注中插入文本和 printf 格式化数字。

插入内容的方法如下：

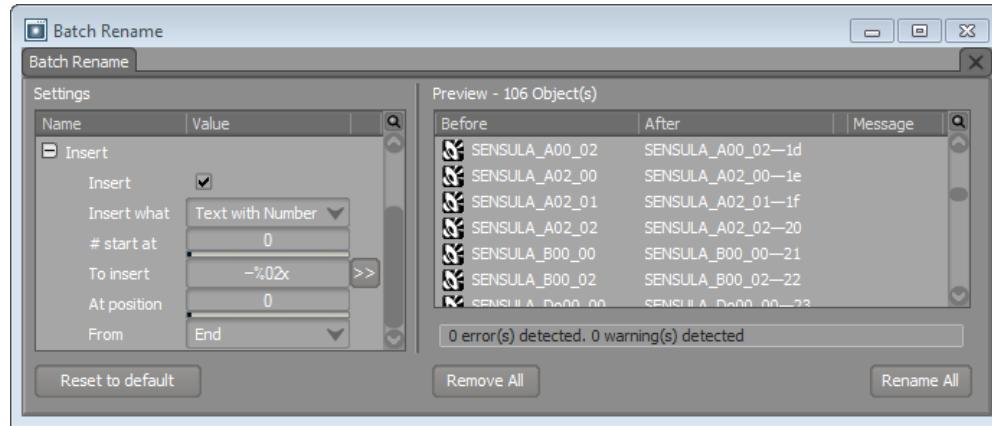
1. 请在 Apply To 列表中指定该重命名操作将应用至所列对象的名称还是备注。
2. 启用 Insert。
3. 在 Insert what 中指定插入的类型：
 - Text：文本。To insert 中仅识别纯文本条目。
start at 将变为无效。
 - Text with Number：带数字的文本。To insert 中识别文本条目，以及 C++ printf 格式化数字。

start at 将变为有效。

4. 如果您选择 Text with Number，则在 # start at 栏中输入一个整数，或将滑杆移至所需的值，来给输入的 C++ printf 格式化数字指定开始计数的数值。

如果您选择 Text，则该栏应处于无效状态。

5. 在 To insert 栏中指定插入文本。请注意，如果您指定了 Text with Number，则 Wwise 需要在何处输入相应的 C++ printf 格式化数字。
 - 另外也可以使用选择按键来选择预定义的文本，输入常用的 C++ printf 格式化数字。
6. 指定应要插入字符的位置。为此请在 At position 栏中输入整数，或将滑杆移至所需值。
7. 选择应从哪个方向计数 At position：
 - (名称或备注的) Beginning (起始位置)
 - (名称或备注的) End (结束位置)
8. 检查您所做的更改，然后点击 Rename All 以插入内容。



示例：插入数字

让我们再次以 Wwise Sample Project 为例。假设在 Sample Project 中，我们想要追踪 Actor-Mixer Hierarchy 里 MIDI Work Unit 内 Kalimba, Sansula, Voice 和 Wood Sansula 这几个 Blend Container 的所有虚拟乐器 SFX。不同的虚拟乐器应该有其自己的容器。在[示例：从匹配项中移除](#)中，我们已经从这些虚拟乐器的 SFX 对象名称中删除了组别前缀“VI_”，因为它不是识别这些虚拟乐器 SFX 的关键字。QA 团队已决定如果对象以独有的识别数字开头，则会更容易对其进行跟踪，因此我们将执行以下操作：在这些 SFX 对象前面插入独有的数字。

1. 在 Project Explorer 中选择所有容器内的 SFX 对象，然后点击 Batch Rename 快捷方式：Ctrl+F2（默认情况下）。

此时显示 Batch Rename 视图及在其 Preview 面板中选择的对象。

2. 确保 Name 是所选的 Apply to 选项，在 Settings 面板中启用 Insert。

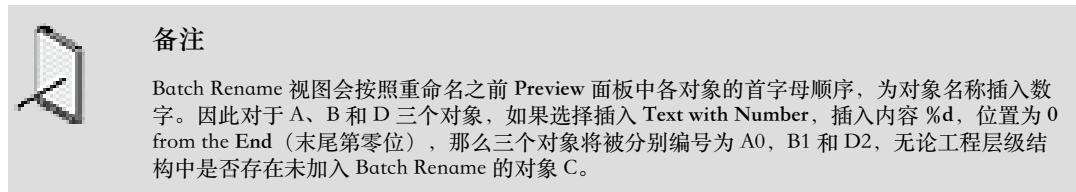
插入栏 (Insert、Insert what、To insert、At position 和 From) 激活。

3. 选择 Text with Number。

start at 栏激活。

4. 保持 #start at 为默认值 0。
5. 在 To Insert 栏中，输入一个英文破折号，后接一个小写十六进制 printf 符号并补零，以确保其至少包含两位：—%02x
6. 将 At position 指定为 0。
7. 在 From 列表中选择 End。
8. 点击 Preview 面板。
9. After 列更新，显示的对象名称后缀带有一个英文破折号（用来与原始对象名称作适当分隔）和两位十六进制数字，如 SENSULA_A02_01—1f。
10. 点击 Rename All 以应用更改。

Batch Rename 中各个对象的名称都会更新，并会在 Preview 面板中显示 Successfully renamed 信息。



相关主题

- [打开 Batch Rename 视图](#)
- [应用批量重命名更改](#)

应用批量重命名更改

指定批量重命名设置 后，在应用这些设置前，最好能检查一下 Preview 面板，以查看指定的重命名设置所造成的影响。

查看预览的方法如下：

1. 查看检测到的错误计数，以及面板下方栏中的警告。
2. 按信息类型列将对象排序（点击 After 右侧没有标题的列标）。

首先列出的是包含错误信息的对象然后列出警告信息，最后列出没有信息的对象。

3. 察看 Before 和 After 列，仔细查看包含信息的对象。

After 列会显示应用 Settings 面板设置之后的效果。

如果有必要的话（警告不一定表示必须要进行更改），找出需要修改的部分。

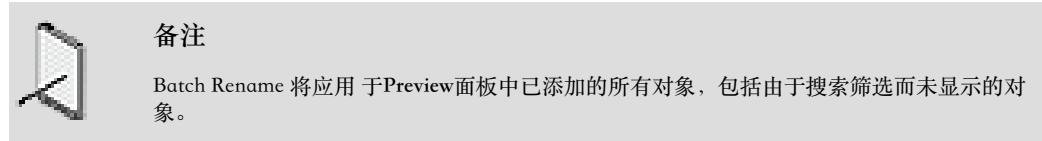
4. 根据需要更改 Settings 面板条目，然后点击 Preview 面板。

面板更新，显示更改之后的效果。

5. 继续调整这些设置，直至不再出现错误信息，并对预览结果感到满意（在 After 列中）。

对所有列出的对象应用重命名设置的方法如下：

1. 点击 Rename All，以更改 Preview 面板的 Before 列中列出的对象名称或备注。



所有列出的对象名称或备注将更改为其 After 列值。

信息类型和 Message 列将更新，显示绿色方块和 Successfully renamed 信息。

相关主题

- [打开 Batch Rename 视图](#)
- [指定批量重命名设置](#)



部分 II. 设置工程



3. 处理工程	48
概述	49
管理工程	51
定义工程设置	54
定义工程的默认用户设置	76
排查工程的问题	77
学习附录 —— 有关声障和声笼的详细信息	82
工程管理技巧与经验总结	84
4. 管理多平台	86
概述	87
使用 Platform Manager	87
5. 管理语言	91
概述	92
使用 Language Manager	92
6. 管理 Workgroup	96
概述	97
将工程分成 Work Units	100
查看工程文件的状态	107
结合版本控制系统使用 Wwise	109
解决工程中的不一致现象	109
使用 Workgroup 插件管理工程文件	111
Workgroup 技巧与经验总结	122
7. 管理工程中的媒体文件	125
概述	126
导入的过程	126
导入媒体文件	133
替换媒体文件	149
管理文件导入问题	151
重新组织 Originals Folder 中的媒体文件。	153
清除缓存	154
在外部编辑器中编辑音频文件	155
使用插件创建 Source	156
媒体文件管理技巧和经验总结	157
8. 构建 Actor-Mixer Hierarchy	158
概述	159
什么是声音对象?	159
在创建 Actor-Mixer Hierarchy 时为对象分组	161
构建 Actor-Mixer Hierarchy	164
工程层级结构中的属性介绍	168
通过随机化属性值来改善播放	176
构建 Actor-Mixer Hierarchy 的技巧和经验总结	177
9. 建立输出总线的结构	181
概述	182
定义总线的属性	186
构建总线层级结构 —— 示例	190

第3章 处理工程

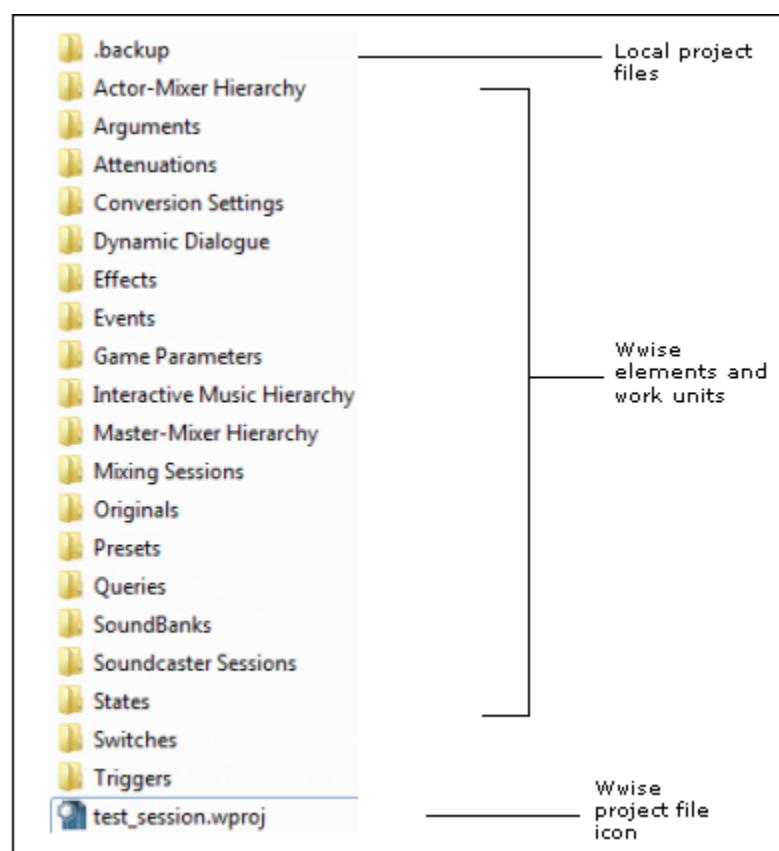
概述	49
管理工程	51
定义工程设置	54
定义工程的默认用户设置	76
排查工程的问题	77
学习附录——有关声障和声笼的详细信息	82
工程管理技巧与经验总结	84

概述

在 Wwise 中，工程包括所有素材，以及您为各个平台和语言的素材设置的属性和行为。该工程还包括 Wwise 元素，例如事件、预设、日志、模拟以及您生成的 SoundBanks（声音包）。工程包括您的所有工作，如果您与他人合作，则还包括同事的工作。

每款游戏只能使用一个 Wwise 工程。您可以将工程划分成多个独立的工作单元。有关使用工作单元的信息，请参阅[将工程分成 Work Units](#)。

在 Wwise 中，工程开发的第一站是 Project Launcher（工程启动器），在此您可以创建和打开工程。在创建工程时，会在您的电脑或网络上您选择的位置中创建一系列的文件夹。



Wwise 工程文件夹结构包括各种文件夹，其中包含不同工程元素的 XML 文件，以便更轻松地管理工程版本和多个用户。典型的工程文件夹包括以下内容：

- **.cache** —— 缓存文件夹，导入到工程中的 SFX、语音和插件素材的转码版本。请勿在版本控制系统中包含此文件夹。
- **Actor-Mixer Hierarchy** - 声音以及振动结构文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Attenuations** - 衰减共享集文件夹，包含工程默认工作单元以及用户创建工作单元。

- **Conversion Settings** - 转换设置共享集文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Dynamic Dialogue** - 对话事件文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Effects** - 效果器共享集文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Events** - 事件文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Game Parameters** - 游戏参数文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Interactive Music Hierarchy** - 音乐结构的件夹，包含工程默认工作单元以及用户创建工作单元。
- **Master-Mixer Hierarchy** - 信号输出结构文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Mixing Sessions** - 混音会话文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Original** -- 原始音频文件夹，包含导入到工程中的素材原始版本的全部拷贝。
- **Presets** - 预设文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Queries** - 查询文件夹，包含工程默认工作单元以及用户创建工作单元。
- **SoundBanks** - 音频包文件夹，包含工程默认工作单元以及用户创建工作单元。
为工程生成 SoundBank 后，会显示一个新的 Generated SoundBank（已生成的声音包）文件夹。
- **Soundcaster Sessions** - 声音选角器会话文件夹，包含工程默认工作单元以及用户创建工作单元。
- **States** - 状态文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Switches** - 切换开关文件夹，包含工程默认工作单元以及用户创建工作单元。
- **Triggers** - 触发器文件夹，包含工程默认工作单元以及用户创建工作单元。
- **IncrementalSoundBankData.xml**: 列出生成的所有 SoundBank（含名称、唯一哈希值和时间戳）。Wwise 使用此文件来确定 SoundBank 有无更改。若无更改，则 Wwise 将不重新生成 SoundBank，同时在 Generating SoundBanks（生成音频包）对话框的 Created（创建日期）列中显示 Up to Date（最新）。此文件不应由版本控制系统来管理。
- **.validationcache** —— 已对照当前 XML schema 版本进行了验证的所有工程文件的列表。通过跟踪记录经过验证的文件，Wwise 每次加载工程时无需再验证这些文件。这极大地缩短了 Wwise 加载工程所需要的时间。此文件不应由版本控制系统来管理。
- **.wsettings** —— 为当前工程定义的默认转码和杂项对象设置。这些设置会按各个用户分别保存。此文件不应由版本控制系统来管理。
- **.wproj** —— Wwise 工程文件。双击工程图标可打开工程。

创建工程后，您可以根据游戏设计将工程划分为不同工作单元，并为各种资源创建结构。同时，您还可以在 Master-Mixer Hierarchy 中为工程建立输出结构，并创建工程所需的 Game Sync（游戏同步器）。

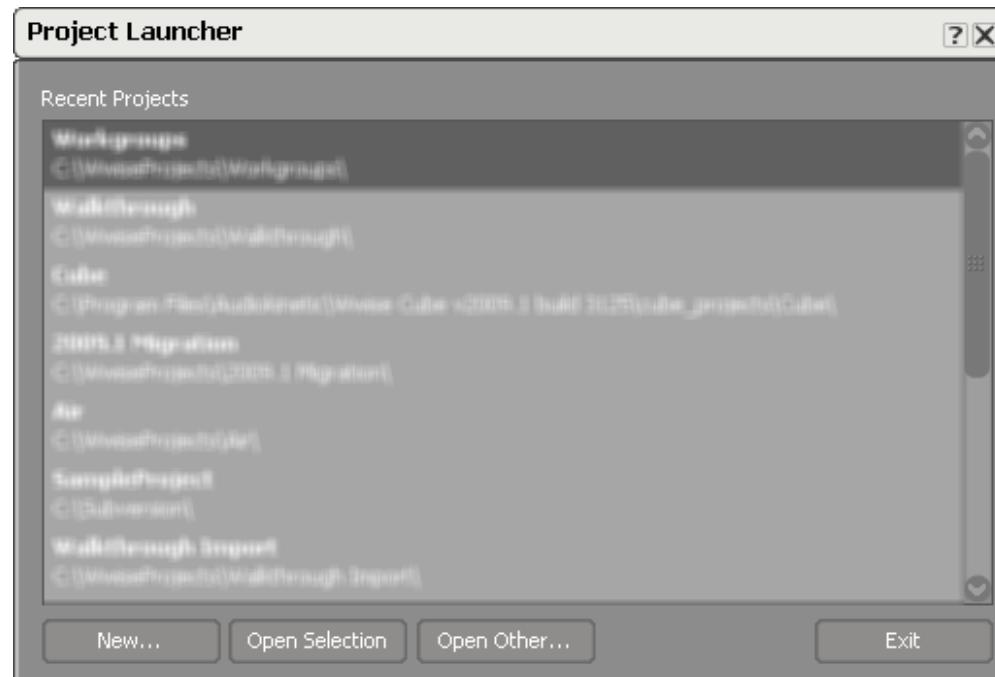
工程中除这些文件夹外，还可能有其它文件，例如完好度报告和 SoundBank 工程头文件。

Wwise 工程及其文件夹的设计使它易于集成到外部版本控制管理工具中去。有关如何管理多个工程用户的详细信息，请参阅[第 6 章 管理 Workgroup](#)。

管理工程

由于一个 Wwise 工程包含游戏音频的所有声音、属性和 SoundBanks，因此认真管理工程非常重要。和其它 Windows 程序一样，您可以在 Windows 资源管理程序中管理工程文件夹和执行复制、移动和删除工程等标准任务。但是，要创建和保存工程，需要在 Wwise 中操作。

从 Wwise Launcher 中直接打开 Wwise 时，Project Launcher 将会显示，您可以在其中创建新工程或打开现有工程。



在 Wwise 中打开工程后，您可以在 Project 菜单中访问一系列命令，包括打开、关闭、创建和保存工程等操作。许多命令都设有键盘快捷键。快捷方式的完整列表请参阅[附录 C, 快捷键](#)。

创建新工程

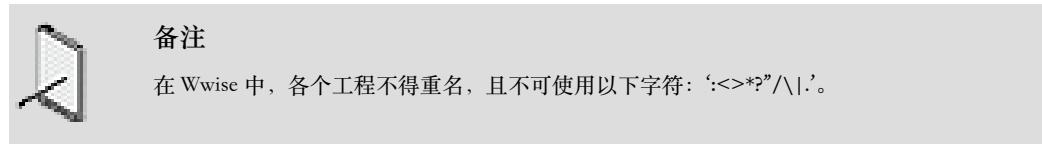
您需要使用 Project Launcher 来新建第一个 Wwise 工程。在已加载的 Wwise 工程中，可以通过 Project > New... 菜单选项新建工程。

创建新工程的方法是：

1. 执行以下操作之一：
 - 在 Project Launcher 中，点击 New...（新建...）
 - 在 Wwise 菜单栏，点击 Project > New。
 - 按 Ctrl + N。

此时将打开 New Project 对话框。

2. 在 Name (名称) 中为新工程输出名称。

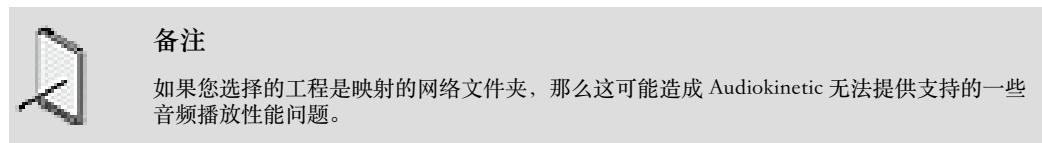


在输入有效的工程名称时，Project folder 中的工程路径将更新，并且 OK 将被激活。

3. 在 Location (位置) 中，执行以下操作之一：

- 在路径列表中，选择以前用过的工程的路径。
- 点击浏览按钮 [...] 并选择想要创建新工程的位置。

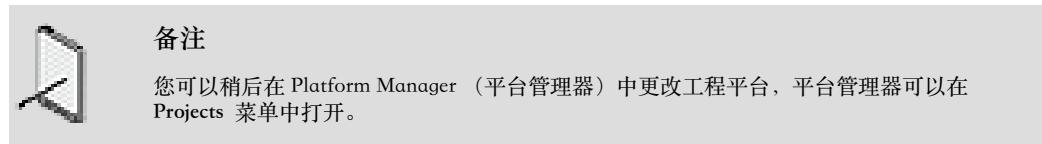
您为工程文件夹选择路径显示在 Project folder 中。



4. 在 Original files (原始文件) 中，执行以下操作之一：

- 从列表中选择 Use Default Originals Directory (使用默认原始音频目录)。此文件夹位于 Wwise 工程文件夹中。
- 点击浏览按钮 [...] 并选择想要创建 "Originals" 文件夹的位置。

5. 仔细查看平台列表。点击 Add... 可[添加平台](#)，点击 Remove 可[移除平台](#)，或者点击 Rename 可更改您选择的任何所列平台的名称。



6. 选择您要加入工程的所有素材组。

7. 点击 OK。

新工程将被创建，并位于所选位置的 Wwise 工程文件夹中。

相关主题

- [打开和关闭工程](#)
- [保存工程](#)
- [使用 Platform Manager](#)

打开和关闭工程

打开 Wwise 时，您最近处理的工程会显示在 Recent Projects 列表中。您可以打开列表中的某个工程，或者转而浏览文件系统去打开另一个工程。记住，您一次只能打开一个工程。如果您已有一个工程打开，则需要首先关闭已打开的工程。

打开一个工程之前，Wwise 会先对其进行验证。则如果存在 XML 语法错误，工程则将无法加载。如果存在工程不一致性，则 Project Load Log (工程加载日志) 对话框

将打开，其中包括不一致性列表和可能的解决办法。如果您接受 Wwise 建议的解决办法，则这些办法会被执行，但直至您保存工程时才会被保存。有关如何处理这些工程相关问题的详细信息，请参阅[解决工程中的不一致现象](#)。

如果您正在处理一个非常大的工程，而且加载时间很长，则可以卸载特定的工作单元，只保留正在处理的工作单元。当工作单元从工程中卸载后，Wwise 每次打开工程时将不再加载它们。这可以极大地加快打开工程的速度。您随时可以将工作单元重新加载到工程中。关于在工程中加载或者卸载工作单元的更多信息，请参阅[在工程中加载/卸载 Work Unit](#)。

从 Project Launcher 中打开工程的方法是：

1. 执行以下操作之一：

- 在 Project Launcher 的 Recent Projects 列表中，选择一个工程并单击 Open Selection。
- 点击 Open Other（打开其它）以前往您想打开的工程。

Loading Project 对话框于是会显示进度条，然后选定的工程便会在 Designer Layout 中打开了。



技巧

您可以双击 Recent Projects 列表中的工程来将其打开。

在 Wwise 中打开工程的方法是：

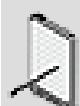
1. 执行以下操作之一：

- 在菜单栏中，点击 Project > Open。
- 按 Ctrl + O。

此时 Open Folder 对话框打开，您可以在其中浏览到您要打开的工程。

2. 前往包含 Wwise 工程文件的工程文件夹。
3. 选择 WPROJ 文件并点击 Open。

此时选定的工程打开了。



备注

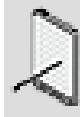
您还可以通过前往工程文件夹并双击 Wwise 工程图标来打开工程。如果您系统中目前安装了多个 Wwise 版本，则工程图标将始终启动 Wwise 的最新版本。

关闭工程的方法是：

1. 执行以下操作之一：

- 在菜单栏中，点击 Project > Close。
- 按 Ctrl + F4。

工程于是被关闭，您将返回到 Project Launcher。



备注

如果您已删除了工程中的对象，那么关闭工程时不会删除相关音频文件。要删除这些落单文件，您需要清除音频缓存。有关详细信息，请参阅[清除缓存](#)。

相关主题

- [创建新工程](#)
- [保存工程](#)
- [解决工程中的不一致现象](#)
- [在工程中加载/卸载 Work Unit](#)

保存工程

对当前工程进行了更改时，标题栏中工程名称旁边会显示星号，并且在 Project Explorer 中已更改的 Work Unit 旁边也将显示星号。定期保存工程是个好习惯。

在某些情况下，如果您修改了只读的 Work Unit，则工程的某些部分可能无法保存。在开始处理 Work Unit 之前，请确保更改可以被保存。在使用版本控制系统管理工程文件时，请记住需要 checkout 工程文件才能保存这些文件。有关保存由版本控制系统管理的工程文件的详细信息，请参阅[在使用 Perforce 时保存工程](#)。



备注

对于 Project Load Log 对话框中列出的工程不一致错误，在修复后，除非您手动保存该工程，否则修复将不会被保存。有关这些错误的详细信息，请参阅[解决工程中的不一致现象](#)。

保存工程的方法是：

1. 执行以下操作之一：
 - 在 Wwise 菜单栏中，点击 Project > Save。
 - 按 Ctrl + S。

您对工程所做的更改于是保存下来了，然后您将返回到工程。

相关主题

- [创建新工程](#)
- [打开和关闭工程](#)

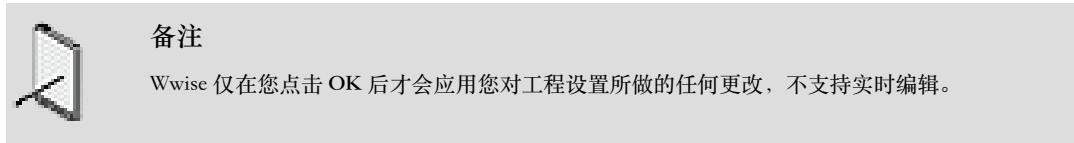
定义工程设置

在 Wwise 中，您可以在 Project Settings 对话框中进行工程设置。

Project Settings 对话框包含以下选项卡：

- [General](#)，用于设置版本控制插件，每个平台的音量阈值，Originals 工程资源文件夹的位置，自动采样率检测设置，以及事件名称创建设置。
- [Conversion](#)，用于设置工程的默认转换设置共享集，以及自动采样率检测设置。

- [SoundBanks](#)，用于为工程定义 SoundBank 设置，包括是否生成内容文件、头文件以及 SoundBank 中 Event 的最大衰减距离信息，是否使用 SoundBank 名称，SoundBank 的存储位置，以及 SoundBank 生成前后执行的用户定义的自定义步骤（可选）。
- [Log](#)（日志），用于管理 Conversion 和 SoundBank 日志中显示的警告、错误和消息。
- [Obstruction/Occlusion](#)（声障/声笼），用于定义工程中的声障和声笼音量、LPF 和 HPF 曲线。
- [External Source](#)（外部源），用于指定外部音频源的输入输出路径，这个外部源将结合 External Source 插件一起使用。
- [Network](#)（网络），用于指定在Wwise 设计工具与游戏之间通信时使用的端口号。
- [Custom Properties](#)（自定义属性），用于定义 Actor-Mixer Hierarchy 的 Sound 和 Audio Source 对象属性。



定义工程的常规设置

在 Project Settings 对话框的 General 选项卡上，您可以执行以下任务：

- [指定工程的 Volume Threshold](#)
- [配置版本控制插件](#)
- [定义 Originals 文件夹设置](#)
- [定义缓存文件夹设置](#)
- [定义事件创建设置](#)

指定工程的 Volume Threshold

Volume Threshold（音量阈值）是电平阈值，低于此电平值的声部将按照 Property Editor 中 Advanced Settings 所定义的行为进行处理。您可以在 Project Settings 对话框中指定默认音量阈值。如果没有使用 Wwise API 另行设置 Volume Threshold，则会使用此值。关于管理底层音频和振动对象，以及设置对象达到音量阈值时的行为，更多信息请参阅 [管理音量较低的对象](#)

为工程设置 Volume Threshold 的方法如下：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project（工程）菜单中，选择 Project Setting。
 - 按 Shift+K。
2. 在 Platforms 分组框中，指定各平台下声音降低到多大音量时，将开始按照 Property Editor 的 Advanced Settings 中定义的行为进行处理。



3. 点击 OK 保存设置并关闭 Project Setting 对话框。

指定最大声部数

与指定工程音量阈值非常相似，在工程中同时播放的声部数量有限。如果工程中的声部数量超过了这个指定值，则也会查找当前所有播放对象 Property Editor 中的 Advanced Settings。

在此，Wwise 将根据哪些对象具有最高优先级来决定保留哪些声部。因此，超出虚声部数量并且具有最低优先级的对象将被虚拟化或淘汰。

相关主题

- [配置版本控制插件](#)
- [定义 Originals 文件夹设置](#)
- [定义缓存文件夹设置](#)
- [定义 Sample Rate Automatic Detection（采样率自动检测）设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)
- [定义事件创建设置](#)
- [将设置从一个平台复制到另一个平台](#)

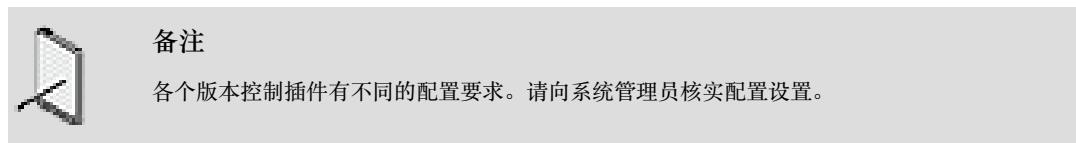
配置版本控制插件

如果使用版本控制系统管理工程素材和文件，那么您可以选择工程的 Workgroup 插件。如果 Wwise 的当前版本支持此插件，则您还可以配置此插件用于工作区域。

Wwise 附带以下两个版本控制插件：

- [Perforce®](#) —— 请参阅[Perforce](#)了解有关兼容版本的详细信息。
- [Subversion](#) —— 请参阅[Subversion](#)了解有关兼容版本的详细信息。

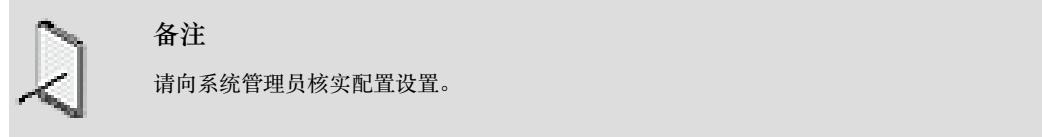
有关 Wwise 支持的 Perforce 和 Subversion 版本的详细信息，请参阅 [支持的 Perforce / Subversion 版本](#) 或 [Wwise SDK 文档](#)。



定义工程版本控制插件的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project（工程）菜单中，选择 Project Setting。

- 按 Shift+K。
2. 从 Workgroup 分组框的 Plug-in 下拉菜单中，为您的工程选择 <Source Control> 插件。
 3. 要配置该插件，请点击 Config。
- 此时将打开 Source Control Plug-in Configuration 对话框。
4. 在 Source Control Plug-in Configuration 对话框的字段中输入必需的信息。



5. 点击 OK 保存配置设置并关闭 Plug-in Configuration（插件配置）对话框。
6. 点击 OK 保存设置并关闭 Project Setting 对话框。

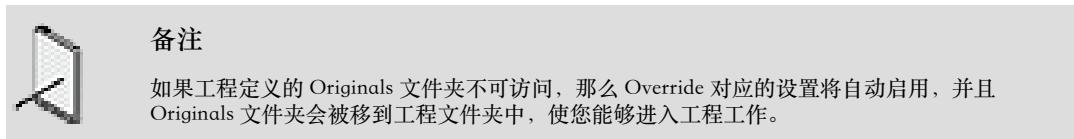
相关主题

- [指定工程的 Volume Threshold](#)
- [定义 Originals 文件夹设置](#)
- [定义缓存文件夹设置](#)
- [定义 Sample Rate Automatic Detection（采样率自动检测）设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)
- [定义事件创建设置](#)

定义 Originals 文件夹设置

工程的 Originals 文件夹包含您导入到工程中的原始音频文件的副本。它的位置是在创建工程时定义的。在 General 选项卡中，您可以指定 Originals 文件的位置。您可以选择一个位置供整个工程使用，也可以选择将这些文件保存到其它位置供您自己使用。在以下情形中，此选项可能会非常有用：

- 在远程工作时，您没有访问 Originals 文件夹的权限。
- 您没有更改 Originals 文件夹内容的权限。
- 需要为 Originals 文件夹创建一个临时位置，同时不更改工程 Originals 文件夹的位置。



有关 Originals 文件夹的详细信息，请参阅[媒体文件结构](#)。

定义工程 Originals 文件夹的位置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Settings。
 - 按 Shift+K。
2. 要定义工程的 Originals 文件夹位置，则需要执行以下操作之一：

在 Original Audio Files 分组框中，输入或粘贴原始文件夹的路径。该路径为绝对路径或相对于工程文件夹的相对路径。默认路径为“Originals”。

点击 Browse 按钮 (...) 以前往您要存储 Originals 音频文件夹的位置。

3. 要定义 Originals 文件夹位置供您个人使用，请执行以下操作之一：
 - 在 Original Audio Files 分组框中，选择 Override location for current user 选项。
 - 点击 Browse 按钮 (...) 以前往您要存储 Originals 音频文件夹的位置。
4. 点击 OK 保存设置并关闭 Project Setting 对话框。

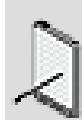
相关主题

- [指定工程的 Volume Threshold](#)
- [配置版本控制插件](#)
- [定义缓存文件夹设置](#)
- [定义 Sample Rate Automatic Detection \(采样率自动检测\) 设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)
- [定义事件创建设置](#)

定义缓存文件夹设置

工程缓存文件夹包含了音频文件转码及 SoundBank 生成期间由 Wwise 生成的中间数据。在创建一个新工程时，该文件的初始位置位于工程目录中的“.cache/”。在 Project Setting 对话框的 General 选项卡中，您可以更改缓存文件的位置。您可以选择一个位置供整个工程使用，也可以选择将这些文件保存到其它位置供您自己使用。在以下情形中，此选项可能会非常有用：

- 远程工作时，没有访问缓存文件夹的权限。
- 您没有权限改变缓存文件夹的内容。
- 需要为 Originals 文件夹创建一个临时位置，而又不更改工程 Originals 文件夹的位置。



备注

如果工程定义的缓存文件夹不可访问，那么 Override 对应的设置将自动启用，并且缓存文件夹会被移到工程文件夹中，使您能够进入工程工作。

如果您选择不沿用缓存文件夹的位置，则下次打开 Wwise 工程时，缓存文件夹将位于新的位置。然而其它用户仍将访问位于原始位置的文件夹。禁用此选项将恢复访问位于工程位置的缓存文件夹。有关缓存文件夹的详细信息，请参阅[媒体文件结构](#)。



注意

多个用户不可同时访问同一个缓存文件夹。

定义工程缓存文件夹的位置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Settings。
 - 按 Shift+K。
2. 要定义工程的缓存文件夹位置，则需要执行以下操作之一：
 - 在 Cached Audio Files 分组框中，输入或粘贴缓存文件夹的路径。该路径为绝对路径或相对于工程文件夹的相对路径。默认路径为“.cache”。
 - 点击 Browse 按钮 (...) 以前往您要存储缓存文件夹的位置。
3. 要定义缓存文件夹位置供个人使用，请执行以下操作之一：
 - 在 Cached Audio Files 分组框中，选择 Override location for current user 选项。
 - 点击 Browse 按钮 (...) 以前往您要存储缓存文件夹的位置。
4. 点击 OK 保存设置并关闭 Project Setting 对话框。

相关主题

- [指定工程的 Volume Threshold](#)
- [配置版本控制插件](#)
- [定义 Originals 文件夹设置](#)
- [定义 Sample Rate Automatic Detection \(采样率自动检测\) 设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)
- [定义事件创建设置](#)

定义事件创建设置

在 Wwise 中创建事件的办法有很多。如果创建的事件没有目标对象，Wwise 会以通用名称为新事件命名。但如果是为选定对象创建事件，则可以指定 Wwise 如何为新事件命名。

定义工程的 Event Creation Setting 的方法如下：

1. 通过执行以下操作之一来打开 Project Setting 对话框：

- 在 Project (工程) 菜单中, 选择 Project Settings。
 - 按 Shift+K。
2. 选择是否为工程或当前用户指定 Event Creation Setting:
- 要为工程设定事件创建设置, 选择 Define settings for project (定义工程设置)。
 - 要为当前用户设定 Event Creation Setting, 选择 Override settings for current user。
3. 选择所需的事件创建设置:
- 启用 Add action name, 即可将操作名称包含在事件名称中。
 - 启用 Modify case, 即可设置事件名称的大小写。

见下文了解更多详情。

4. 点击 OK 保存设置并关闭 Project Setting 对话框。

默认情况下, 将使用对象本身的名称作为新的事件名称。以下选项可以修改新的事件名称:

- **Add action name:** 启用时, 事件的操作名将添加到事件名中。
 1. set as prefix - 操作名称将作为对象名称的前缀: <action_name>_<object_name>
 2. set as suffix - 操作名称将作为对象名称的后缀: <object_name>_<action_name>
- **Modify case:** 如果启用, 则将更改事件名称的大小写。
 1. all lowercase -- 全部小写。事件名称全部为小写字母。
 2. all uppercase -- 全部大写。事件名称全部为大写字母。



相关主题

- [指定工程的 Volume Threshold](#)
- [配置版本控制插件](#)
- [定义 Originals 文件夹设置](#)
- [定义 Sample Rate Automatic Detection \(采样率自动检测\) 设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)
- [定义缓存文件夹设置](#)

定义工程的转码设置

在 Project Settings 对话框的 Conversion 选项卡上, 您可以执行以下任务:

- 指定默认转码设置
- 定义 Sample Rate Automatic Detection (采样率自动检测) 设置

指定默认转码设置

为工程创建 Conversion Settings ShareSets (转换设置共享集) 后，您可以指定哪个共享集将用作默认设置。

默认 Conversion Settings ShareSet 将应用于以下情况：

- 创建新对象时 - 当新对象是顶层父对象时，会使用默认共享集。如果新对象是另一对象的子对象，它将继承父对象的 Conversion Settings。
- 在 SoundBank 生成过程中 - 如果某个对象尚未被指定 Conversion Setting ShareSet，则生成 SoundBanks 之前将使用默认共享集来转换对象。

指定默认 Conversion Setting ShareSet 的方法如下：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Settings。
 - 按 Shift+K。
2. 切换至 Conversion 选项卡。
3. 在 Default Conversion Settings 分组框中，点击 Browse 按钮 (...)。

此时将会打开 Project Explorer —— Browser。

4. 选择您希望作为工程默认转换设置的 Conversion Settings。
5. 点击 OK。

相关主题

- 指定工程的 Volume Threshold
- 配置版本控制插件
- 定义 Sample Rate Automatic Detection (采样率自动检测) 设置
- 定义工程的 SoundBank 设置
- 管理在日志中出现的消息
- 定义工程的声障和声笼曲线
- 指定外部源的输入/输出位置
- 定义工程的默认用户设置
- 排查工程的问题

定义 Sample Rate Automatic Detection (采样率自动检测) 设置

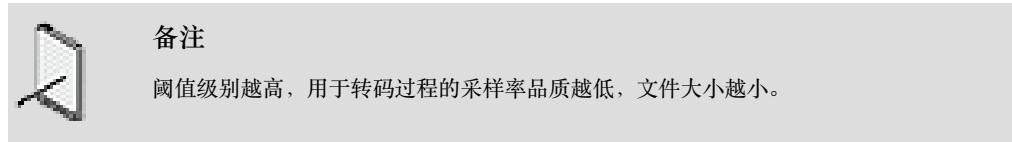
确定媒体文件的最佳采样率既麻烦又耗时。为了加速进度，您可以使用 Wwise 根据快速傅立叶变换 (FFT) 算法来分析各个文件。Wwise 基本使用 FFT 来生成媒体文件的频谱分析，方法是每次使用 Hanning 窗口分析一部分声波。截止音量或阈值识别用于确定最佳采样率的频率，您的文件将按照该最佳采样率进行转码。作为 Project Settings 的一部分，您可以定义 FFT 算法使用的 Hanning 窗口的大小以及高、中、低

三种不同质量设置的阈值电平。当您选择 Auto High（自动高）、Auto Medium（自动中）或 Auto Low（自动低）作为采样率转码方法时将使用这些阈值设置。



定义工程的自动采样率检测设置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project（工程）菜单中，选择 Project Setting。
 - 按 Shift+K。
2. 在 Sample Rate Automatic Detection 中的 FFT window size 列表中，选择 FFT 算法分析声波时使用的 Hanning 窗口大小。
3. 在 Volume Thresholds 分组框中，为以下各质量选项指定 FFT 算法使用的截止音量电平：
 - **Low quality** – 低品质。一个截止音量电平，用于识别可确定最佳采样率的频率，在对文件转码时将使用该最佳采样率。当选择 Auto Low 选项作为转码采样率时使用低品质阈值。
 - **Medium quality** – 中等品质。一个截止音量电平，用于识别可确定最佳采样率的频率，在对文件转码时将使用该最佳采样率。当选择 Auto Medium 选项作为转码采样率时使用中等品质阈值。
 - **High quality** – 高品质。一个截止音量电平，用于识别可确定最佳采样率的频率，在对文件转码时将使用该最佳采样率。当选择 Auto High 选项作为转码采样率时使用高品质阈值。



4. 点击 OK 保存设置并关闭 Project Setting 对话框。

相关主题

- [指定工程的 Volume Threshold](#)
- [配置版本控制插件](#)
- [定义 Originals 文件夹设置](#)
- [定义缓存文件夹设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)
- [排查工程的问题](#)

定义工程的 SoundBank 设置

在为每种平台和语言生成 SoundBanks 前，您需要定义 SoundBank 设置。以下 SoundBank 设置可以在工程级别定义：

- 定义 SoundBank 工程设置 —— 确定生成的 SoundBank 中所包含的信息。
- 为已保存的 SoundBank 指定新位置的方法是：—— 确定硬盘或网络中保存 SoundBank 的位置。
- 定义在生成 SoundBank 之前/或之后要执行的步骤 —— 确定在刚要生成 SoundBank 前将执行的任务。
- 定义在生成 SoundBank 之前/或之后要执行的步骤 —— 确定在刚刚生成 SoundBank 后将执行的任务。

虽然这些设置在工程级别定义，但您可以创建自定义用户设置，覆盖这些工程设置。有关不沿用 SoundBank 工程设置的详细信息，请参阅[定义 SoundBank 的自定义属性](#)。

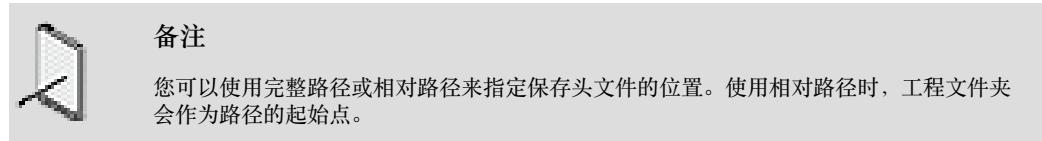
定义 SoundBank 工程设置

在生成 SoundBanks 前，您需要确定哪些信息是生成过程的一部分、如何加入这些信息以及生成什么格式。您选择的设置将取决于游戏如何访问 SoundBanks 中的数据和媒体。

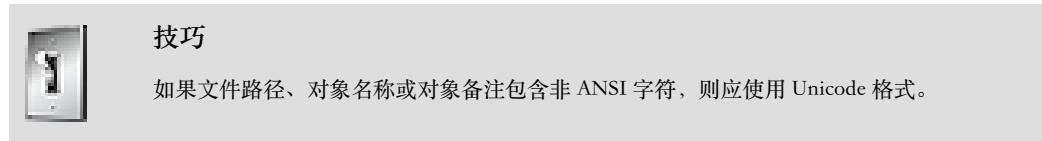
定义 SoundBank 工程设置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Setting。
 - 按 Shift+K。
2. 切换到 SoundBanks 选项卡。
3. 在 SoundBank Settings 分组框中，选择以下任一选项来定义 SoundBanks 自定义设置：
 - Allow SoundBanks to exceed maximum size -- 允许 SoundBanks 超出最大体积。即使超出指定的最大体积时，仍将生成 SoundBank。
 - Generate SoundBank content file -- 生成 SoundBank 内容文件。用于创建列出各个 SoundBank 的内容的文件。内容文件包括有关 Event、Buss、State 和 Switch 的信息，以及流播放音频文件和内存音频文件的完整列表。
 - Generate header file -- 生成头文件。创建将 Event、状态、切换开关和游戏参数名称映射到 ID 的头文件。
 - Max attenuation -- 最大衰减距离。在 SoundBanksInfo.xml 文件中包含各个 Event 的最大衰减距离信息。
 - Estimated duration -- 预计时长。在 SoundBanksInfo.xml 文件中包含各个 Event 的预计最大和最小时长，以及声音是无限循环播放还是只播放一次。
 - Use SoundBank Name: 使用 SoundBank 名称。使用 SoundBank 名称（勾选时）或 ID（不勾选时）来命名生成的 .bnk SoundBank 文件，以及在一个 SoundBank 中引用另一个 SoundBank。
4. 如果选择生成头文件，则必须确定是否保存它。为此，请执行以下操作之一：
 - 点击文本框并直接在其中输入路径。

- 双击文本框或单击浏览按钮 [...]，在弹出的浏览器中前往要选择的位置。



5. 如果选择生成 SoundBank 内容文件，则可以选择所需的文本文件格式和 SoundBank content file format 选项。



6. 点击 OK 以应用这些设置。

相关主题

- [为已保存的 SoundBank 指定新位置的方法是：](#)
- [定义在生成 SoundBank 之前/或之后要执行的步骤](#)
- [定义工程的常规设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)

为已保存的 SoundBank 指定新位置的方法是：

在生成工程的 SoundBanks 时，默认保存文件夹为：

ProjectName\GeneratedSoundBanks\Platform\

如果此位置对您不方便，那么可将它改为电脑或网络中的任何目录。

在为保存的 SoundBank 指定位置时，可使用完整路径或相对路径。使用相对路径时，工程文件夹会作为路径的起始点。例如，下列完整路径和相对路径指定的位置是相同的：

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

为已保存的 SoundBank 指定新位置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Setting。
 - 按 Shift+K。
2. 切换到 SoundBanks 选项卡。
3. 在 SoundBank Paths 分组框中，通过下列操作之一来指定路径：

- 直接在文本框中输入路径。
 - 点击 **Browse...** 并使用浏览器前往您选择的位置。
4. 点击 **OK** 以使您所做的任何更改生效。

相关主题

- [定义 SoundBank 工程设置](#)
- [定义在生成 SoundBank 之前/或之后要执行的步骤](#)
- [定义工程的常规设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)

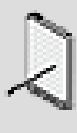
定义在生成 SoundBank 之前/或之后要执行的步骤

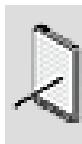
根据工作流程，在生成 SoundBank 之前或之后您可能需要立即执行特定步骤或任务。例如，在生成特定 SoundBank 文件前想从版本控制系统中 check out 它们；或者在生成后想立即将流播放文件复制到 SoundBanks 目录中。

在 Wwise 中，这些任务类型是通过创建命令行来定义的。Wwise 中有一个专门的命令行编辑器，方便您构建数目不限的命令行。为了进一步简化过程，编辑器中包含命令行中可使用的所有 Wwise 专用环境变量和其它 Windows 环境变量的列表。

可用于编写自定义命令行的专用 Wwise 变量如下：

命令行变量	描述
<code>\$(AllowExceedMaximum)</code>	指定当 SoundBanks 超过指定最大大小时是否生成该 SoundBanks。 当选择了 Allow SoundBanks to exceed maximum （允许 SoundBanks 超过最大大小）选项时，此变量设为 True。
<code>\$(ContentFileFormat)</code>	指定生成的 SoundBank 内容文件的文件类型。可能的值为： <ul style="list-style-type: none"> ANSI Unicode
<code>\$(GenerateContentFile)</code>	指定是否创建文件来列出各个 SoundBank 的内容。内容文件包括有关 Event、Buss、State 和 Switch 的信息，以及流播放音频文件和内存音频文件的完整列表。 当选择了 Generate SoundBank content files （生成 SoundBank 内容文件）选项时，此变量设为 True。
<code>\$(GenerateHeaderFile)</code>	指定是否生成将 Event、状态、切换开关和游戏参数名称映射到 ID 的头文件。 当选择了 Generate Header File （生成头文件）选项时，此变量设为 True。
<code>\$(GenerateMaxAttenuationInfo)</code>	指定是否为 Event 生成最大衰减距离信息。 当选择了 Metadata Options: Max attenuation 选项时，此变量设为 True。
<code>\$(GenerateEstimatedDuration)</code>	指定是否生成 Event 的预计最大和最小时长以及时长类型信息。 启用 Metadata Options: Estimated Duration 选项时，此变量将设置为 True。
<code>\$(HeaderFileFullPath)</code>	头文件的完整路径，具体为：\$(HeaderFilePath)\Wwise_IDs.h
<code>\$(HeaderFilePath)</code>	保存头文件的路径或位置。

命令行变量	描述
	此路径来自 Header file path (头文件路径) 文本框。
<code>\$(InfoFilePath)</code>	当前平台的信息文件的完整文件名。
<code>\$(IsRunningFromCmdLine)</code>	指定 Wwise 启动的命令行中是否带有 <code>-generatesoundbanks</code> 选项。
<code>\$(LanguageList)</code>	传送到命令行的语言的列表或者 SoundBank Manager 中的选定语言的列表。  备注 列表采用空格隔开。
<code>\$(Platform)</code>	当前平台的名称。
<code>\$(SoundBankList)</code>	传送到命令行的 SoundBank 的列表或者 SoundBank Manager 中的选定 SoundBank 的列表。  备注 列表采用空格隔开。并用双引号括起整个参数中的列表。
<code>\$(SoundBankPath)</code>	保存当前平台的 SoundBanks 的路径或位置。
<code>\$(UseSoundBankNames)</code>	指定选用 SoundBank 名称（设为 <code>true</code> 时）或 ID（设为 <code>false</code> 时）来用于命名生成的 SoundBank (BNK) 文件，以及在 SoundBanks 中用于引用其它 SoundBank 中的媒体。 当选择了 Use SoundBank names (使用声音包名称) 选项时，此变量设为 <code>true</code> 。
<code>\$(WwiseExeDriveLetter)</code>	Wwise 可执行程序 (Wwise.exe) 所在的电脑盘符。
<code>\$(WwiseExePath)</code>	Wwise 可执行程序 (Wwise.exe) 的路径或位置。
<code>\$(WwiseExeProcessID)</code>	Wwise 可执行程序 (Wwise.exe) 的进程 ID (数字形式)。
<code>\$(WwiseProjectDriveLetter)</code>	Wwise 工程所在的电脑盘符。
<code>\$(WwiseProjectName)</code>	当前工程的名称。
<code>\$(WwiseProjectPath)</code>	Wwise 工程的路径或位置。
<code>\$(WaapiWampPort)</code>	WA API 为 WAMP 协议使用的端口。
<code>\$(WaapiHttpPort)</code>	WA API 为 HTTP 协议使用的端口。



备注

此外，所有环境变量都可以使用，如 `$(WWISESDK)`。

为了尽可能提高灵活性，Wwise 支持为以下类型的步骤指定命令行：

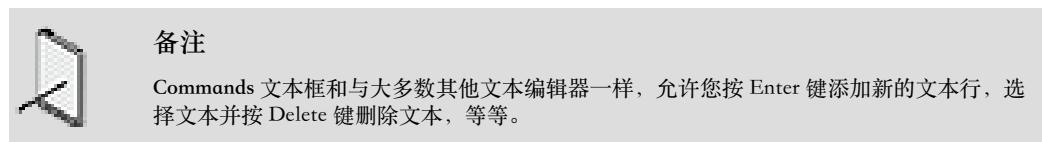
- **Global opening step** -- 全局初始步骤。适用于所有平台并且在任何其它步骤前执行的命令行。
- **Platform-specific pre-generation step** -- 针对平台的生成前步骤。适用于特定平台并且在生成 SoundBanks 前执行的命令行。
- **Platform-specific post-generation step** -- 针对平台的生成后步骤。适用于特定平台并且在生成 SoundBanks 后执行的命令行。
- **Global closing step** -- 全局结束步骤。适用于所有平台并且在所有其它步之后执行的命令行。

在默认情况下，各个工程都包含平台专有的生成后步骤（post-generation step）命令行，该命令行将流播放文件复制到 SoundBank 目录下。然而，您可以通过执行一个不同的命令行将任何类型的任务自动化。Wwise 还自带另一个出厂命令行，它使用 File Packager 来生成包含 SoundBank 中所有数据和媒体的文件包。有关 File Packager 的详细信息，请参阅第 36 章 管理 File Package。有关加载出厂命令行的详细信息，请参阅[加载出厂设定/自定义命令行](#)。

您还可以将您创建的命令行保存到文件（WCMDLINE）中，以便今后在同一工程中使用，跨工程使用，或者与其它用户共享。有关保存命令的详细信息，请参阅[保存自定义命令行到文件](#)。

定义在生成 SoundBank 前执行的用户任务的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project（工程）菜单中，选择 Project Setting。
 - 按 Shift+K。
 2. 切换到 SoundBanks 选项卡。
 3. 要创建全局 Pre-Generation Step（生成前步骤），请点击浏览按钮 [...] 以打开 Global Opening Step Editor。
- Pre-Generation Step Editor（生成前步骤编辑器）打开。
4. 在 Description 文本框中，输入名称，名称要能清晰描述将要执行的步骤或任务。
 5. 点击 Commands 文本框，开始创建您的命令行。



6. 如果要在命令中插入内置宏和环境变量，则请执行以下操作：

在 Macros（宏）分组框中，选择以下其中一个选项：

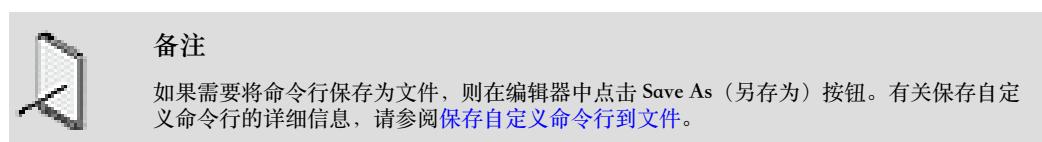
- Built-in Macros -- 内置宏。显示可用于 Wwise 命令行中的一列 Wwise 专用变量。
- Environment Variables -- 环境变量。显示可用于 Wwise 命令行中的一列 Windows 专用环境变量。

要添加变量到命令行中的话，要执行以下操作中的一项：

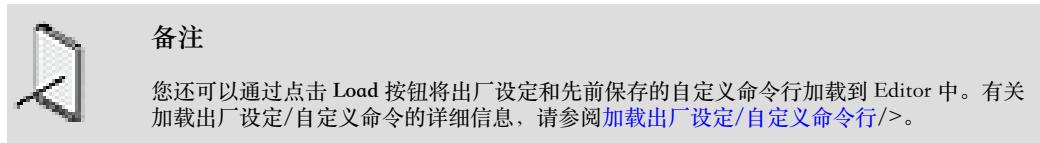
- 双击列表中的变量。
- 从列表中选择变量，然后点击 Insert（插入）。

根据需要，继续添加变量到命令行。

7. 如果需要执行第二个全局生成前步骤，则只需转到第一行末尾，按 Enter 键，然后即可开始创建新的命令行。
8. 点击 OK 保存命令行并关闭 Pre-Generation Step Editor。



- 要创建针对平台的生成前步骤，针对各个平台重复步骤 3 至 8。

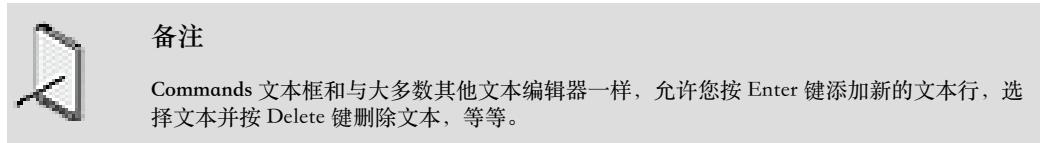


定义在生成 SoundBank 后执行的用户任务的方法是：

- 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在 Project (工程) 菜单中，选择 Project Setting。
 - 按 Shift+K。
- 切换到 SoundBanks 选项卡。
- 在 Post-Generation Step 分组框中，Copy Streamed Files 命令行是默认添加的。要修改此命令行或添加其他命令行，请单击相应的浏览按钮 [...]。

此时将打开 Post-Generation Step Editor。

- 在 Description 文本框中，输入名称，名称要能清晰描述将要执行的步骤或任务。
- 在 Commands 文本框中，单击当前命令行的末尾，然后按 Enter。现在可以开始创建新的命令行了。



- 如果要在命令中插入内置宏和环境变量，则请执行以下操作：

在 Macros (宏) 分组框中，选择以下其中一个选项：

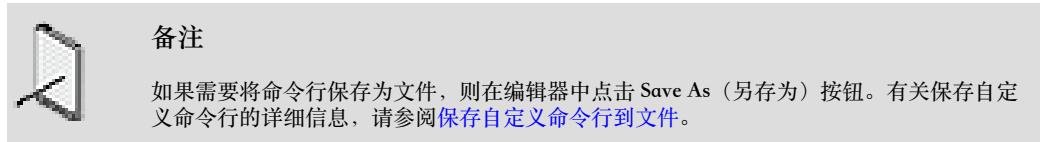
- Built-in Macros -- 内置宏。显示可用于 Wwise 命令行中的一列 Wwise 专用变量。
- Environment Variables -- 环境变量。显示可用于 Wwise 命令行中的一列 Windows 专用环境变量。

要添加变量到命令行中的话，要执行以下操作中的一项：

- 双击列表中的变量。
- 从列表中选择变量，然后点击 Insert (插入)。

根据需要，继续添加变量到命令行。

- 如果需要执行第二个全局生成前步骤，则只需转到第一行末尾，按 Enter 键，然后即可开始创建新的命令行。
- 点击 OK 以保存命令行并关闭 Post-Generation Step Editor。



- 对于全局结束步骤或其它各个平台，重复执行步骤 3 至 8。



备注

您还可以通过点击 Load 按钮将出厂设定和先前保存的自定义命令行加载到 Editor 中。有关加载出厂设定/自定义命令的详细信息，请参阅[加载出厂设定/自定义命令行](#)。

相关主题

- [加载出厂设定/自定义命令行](#)
- [保存自定义命令行到文件](#)
- [定义 SoundBank 工程设置](#)
- [为已保存的 SoundBank 指定新位置的方法是：](#)
- [定义工程的常规设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)
- [定义工程的默认用户设置](#)

加载出厂设定/自定义命令行

Wwise 附带一些已经创建的命令行，包括将流播放文件复制到 SoundBank 目录的命令行，以及将流播放文件和 SoundBanks 打包到一个文件包（package）的命令行。这些被称作出厂设定命令行。您可以加载这些出厂设定命令行或您先前保存到文件中的自定义命令行。

加载出厂设定/自定义命令行的方法是：

1. 在 Pre/Post Generation Step Editor 中，点击 Load...。
2. 在快捷菜单中，点击以下某个选项：
 - From Factory Folder - 打开 Explorer / Finder 并转至 Wwise 出厂设定命令行所在的目录。
 - From Last Location - 打开 Explorer / Finder 并转至上次加载命令行的目录。

此时将会打开 Open（打开）对话框。

3. 选择您要加载的命令行并点击 Open。

命令行将被加载到编辑器中。

相关主题

- [保存自定义命令行到文件](#)
- [定义在生成 SoundBank 之前/或之后要执行的步骤](#)

保存自定义命令行到文件

您可以将您创建的自定义命令行保存为文件，以便今后在同一工程中使用，跨工程使用，或者与其它用户共享。

保存命令行为文件的方法是：

1. 在 Pre-Post Generation Step Editor 中，编写命令行。
2. 完成时，点击 Save As。
Save As 对话框打开。
3. 前往您想保存命令行的文件夹，对它命名，然后按 Save。
命令行将保存为 WCMDLINE 文件，可以随时复用。

相关主题

- [加载出厂设定/自定义命令行](#)
- [定义在生成 SoundBank 之前/或之后要执行的步骤](#)

管理在日志中出现的消息

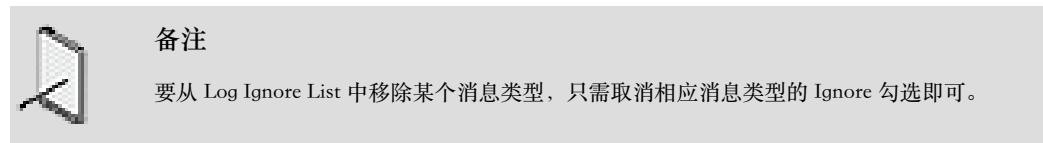
有时候，生成过程中不需要在 SoundBank 或 Conversion 日志中显示特定警告或消息。若不想在 SoundBank 和 Conversion 日志中显示这些类型的警告和消息，可选择将其添加到 Log Ignore List。如果您希望日志中重新显示这些消息时，则只需将这些消息类型从 Log Ignore List 中移除即可。在 Project Settings 对话框中从工程级别管理此消息列表。



添加消息到 Log Ignore List 的方法是：

1. 在菜单栏中，点击 Project > Project Setting。
Project Settings 对话框打开。
2. 切换到 Logs 选项卡。
3. 选择您要添加到 Log Ignore List 的消息。
4. 点击 OK 以关闭 Project Setting 对话框。

下次生成日志时，Wwise 会忽略选定类型的消息，并且不会在日志中显示它们。



更改消息的严重性的方法是：

1. 在菜单栏中，点击 Project > Project Setting。
Project Settings 对话框打开。

2. 切换到 Logs 选项卡。
3. 通过使用 Severity 列中的下拉控件更改严重性。
4. 点击 OK 以关闭 Project Setting 对话框。

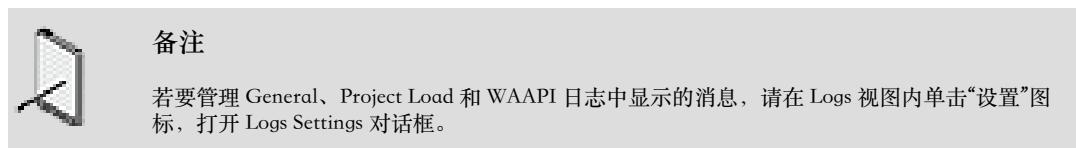
下次生成日志时，将应用选定的严重性。

限制出现日志中消息数量的方法是：

1. 在菜单栏中，点击 Project > Project Setting。

Project Settings 对话框打开。

2. 切换到 Logs 选项卡。
3. 启用 Limit number of messages displayed to: 选项并输入您想要在日志中显示的最大消息数量。注意，此数字用于限制日志中的所有消息类型。
4. 点击 OK 以关闭 Project Setting 对话框。



相关主题

- [定义工程的常规设置](#)
- [定义工程的 SoundBank 设置](#)
- [定义工程的声障和声笼曲线](#)
- [指定外部源的输入/输出位置](#)

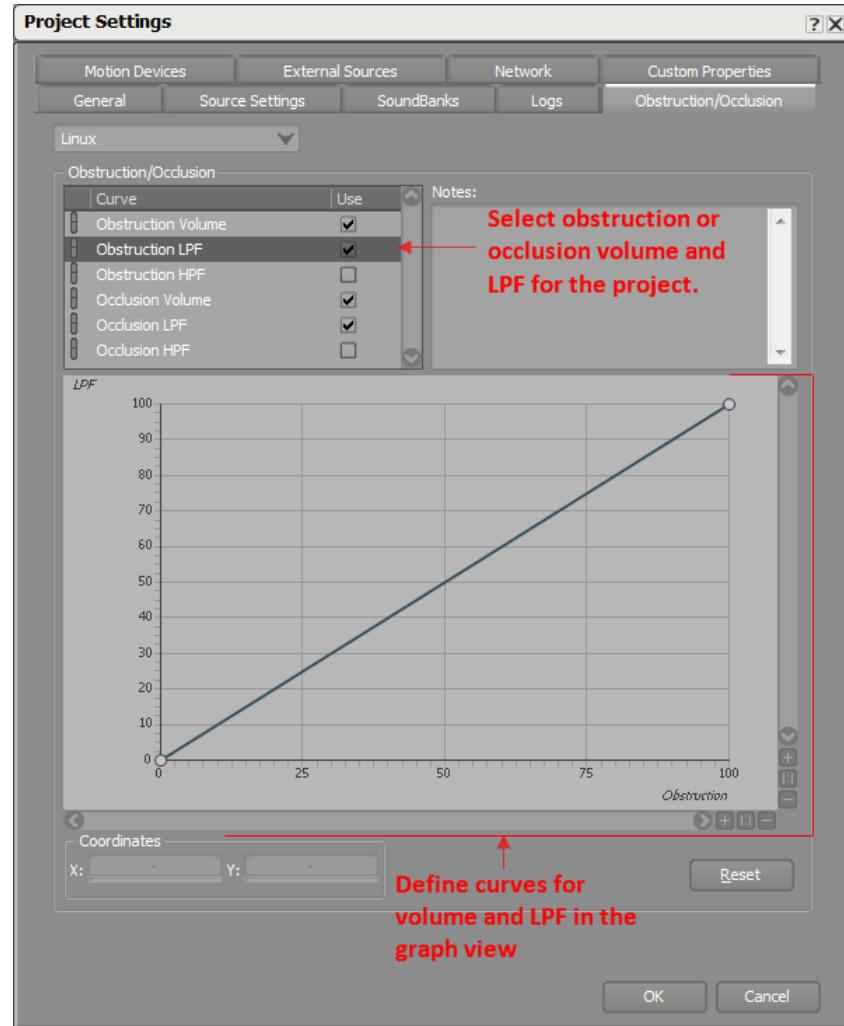
定义工程的声障和声笼曲线

在 Wwise Project Settings 对话框的 Obstruction/Occlusion 选项卡中，可以为工程中各个平台的声音对象定义声障和声笼设置。Obstruction 是指游戏几何空间中的对象（例如墙壁或立柱）部分挡住了声源与听者之间的空间。Occlusion 是指游戏几何空间中的对象完全挡住了声源与听者之间的空间时的声学现象。

游戏程序员通过编程的方法定义游戏的几何空间，在这种情况下可能就会产生声障和声笼条件。Wwise 自身不计算声障和声笼的程度。物理计算必须由游戏完成，并将结果传递给下面的函数：[SetObjectObstructionAndOcclusion\(\)](#)。

有关程序员如何编写声障和声笼的详细信息，请参阅 Wwise SDK 文档的 [环境中的声障和声笼](#)一节。

通过 Wwise，您可以定义平台专有的音量和 LPF 曲线，在游戏中，当声音对象受到声障和声笼的影响时，这些音量和曲线将作用于声音对象。您定义的值是相对值，会加到已应用于游戏对象的任何音量和 LPF 值上。



要创建更多详细和复杂的声障和声笼曲线，您可以定义各条曲线段的形状。曲线段是两个控制点之间的部分。您可选择各种曲线形状，包括线性曲线、恒定曲线、对数曲线、幂数曲线和 S 曲线。

有关声障和声笼的更多背景信息，请参阅[学习附录——有关声障和声笼的详细信息](#)。

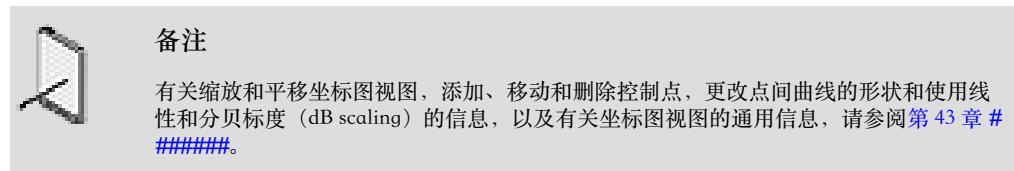
定义工程的声障和声笼设置的方法是：

1. 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在菜单栏中，点击 Project > Project Setting。
 - 按 Shift+K。
2. 切换到 Obstruction/Occlusion 选项卡。
3. 在 Platform 列表中，选择您正在为其声障和声笼设置定义曲线的平台。
4. 要为选定平台上的声障或声笼音量、LPF 和 HPF 指定不同的设置，右键点击 Link 指示图标，然后在快捷菜单中选择 Unlink（断链）。



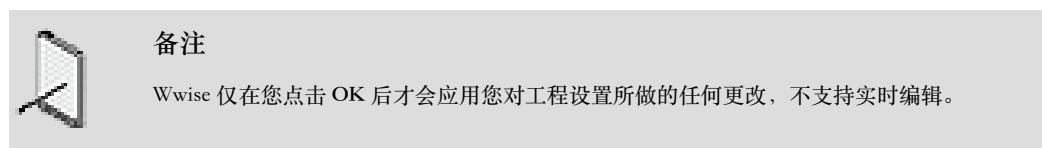
指示图标变成橙色，这表明您为断链属性定义的设置将仅用于选定平台。

5. 要为游戏对象的声障和声笼定义音量、LPF 和 HPF 曲线，请执行以下操作：
 - 启用 Use 复选框来高亮显示相应曲线。
 - 要在曲线上创建点，双击曲线上的点。
 - 要删除曲线上的点，请选择该点并按 Delete 键。
 - 要删除您目前定义的所有点并从头开始，则点击 Reset（重置）。



6. 要在曲线上加注释，请点击 Notes 文本框输入注释。
7. 完成后，点击 OK。

此时 Project Setting 对话框将关闭，并保存工程声障/声笼属性曲线。



相关主题

- [缩放和平移坐标图视图](#)
- [定义坐标图视图的坐标显示方式](#)
- [指定控制点之间曲线的形状](#)
- [创建新工程](#)
- [排查工程的问题](#)
- [定义工程的默认用户设置](#)
- [定义工程的常规设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [指定外部源的输入/输出位置](#)

指定外部源的输入/输出位置

如果您计划使用 External Source plug-in（外部源插件），则必须指定外部源列表文件的位置。此文件是一个非常简单的 XML 文件，它指定以下内容：

- 与 Wwise 中创建的模板相关联的外部音频文件位置。
- 用于转换每个文件的 Conversion Settings。

您还必须指定保存经过转码后的源的文件夹，以便 Wwise 声音引擎在运行时可以使用它们。

指定外部音频源的输入/输出路径的方法是：

1. 在菜单栏中，点击 Project > Project Setting。

Project Settings 对话框打开。

2. 切换到 External Sources 选项卡。
3. 在 Input Path 分组框中，执行以下操作之一来指定外部源列表文件 (WSOURCES) 所在的文件夹：
 - 点击 External Sources List，然后输入外部资源列表文件所在的路径。
 - 点击游戏平台的相应浏览按钮 [...], 前往该平台外部资源列表文件的文件夹，然后点击 Open。
4. 对工程中的各个有效平台重复执行步骤 3。
5. 在 Output Path 分组框中，执行以下任一操作来指定转换后的外部源保存的文件夹：
 - 点击 External Sources Output Folder 列表，并输入要保存音频文件的路径。
 - 点击对应于其中一个游戏平台的 Browse 按钮 [...], 前往您想为该平台保存音频文件的文件夹，然后点击 OK。
6. 对各个平台执行步骤 5。
7. 点击 OK 以关闭 Project Setting 对话框。

相关主题

- [定义工程的常规设置](#)
- [定义工程的 SoundBank 设置](#)
- [管理在日志中出现的消息](#)
- [定义工程的声障和声笼曲线](#)
- [定义工程的默认用户设置](#)

指定网络端口

将 Wwise 设计工具连接到游戏进行性能分析和实时编辑时，Wwise 与游戏之间的通信将通过您为其打开的网络端口进行处理。您可以在 Wwise Project Settings 对话框的 Network 选项卡中自行设置 Wwise 设计工具使用的端口。在游戏初始化 Wwise 通信模块时，可以对游戏开放和使用的端口进行定制。

您可以定义特定的端口号（固定），也可以让操作系统自动选择一个端口号（动态）。具体选择取决于端口是由 Wwise 设计工具开启还是由游戏开启：



注意

在 Project Settings 中更改此端口时，请务必确保使用的值和代码中初始化 Wwise 通信模块时相同（特别是传递给 `AK::Comm::Init()` 的 ports.uDiscoveryBroadcast 结构成员）。

Game Discovery Broadcast Port (游戏探查广播端口)

此端口由集成到游戏中的 Wwise 通信模块开启。Wwise 设计工具为了发现网络上可以连接的游戏，会发送消息广播，游戏需要通过这个端口侦听网络中的消息广播。Wwise 设计工具将 Game Discovery (游戏探查) 消息广播至该端口，因此游戏中和 Wwise 工程中应使用相同的端口，这点十分重要。由于游戏和 Wwise 设计工具需要确定该端口，因此该端口不得为动态（不得设置为 0）。



注意

您可以选择使用 1 到 65535 之间的任意端口号；但是请验证您定义的端口号不会与同一计算机或游戏机上运行的其它应用程序存在冲突。

Game Discovery Response Poart (游戏探查响应端口)

此端口由 Wwise 设计工具开启。在此，游戏对 Game Discovery 消息做出响应。在定义此端口的端口号时，记住以下重要注意事项：

- 如果将此端口设置为 0，则等于告诉 Wwise 设计工具自行请求动态的端口号，又称为临时端口号。这意味着操作系统不会使用特定端口号，而是自动选择一个端口号。自动选择端口号是默认行为，建议使用动态端口号以避免与其它应用程序发生冲突。
- 如果需要使用固定端口号（即不是动态/临时端口号），则要避免使用 Game Discovery Broadcast Port 的同一端口号，除非您不打算连接到与创作应用同机运行的游戏。
- 您可以选择使用 0 到 65535 之间的任意端口号；但是请验证您定义的端口号不会与同一计算机或游戏机上运行的其它应用程序存在冲突。

自定义游戏中的通信端口

如前所述，Game Discovery Broadcast Port 由通信模块在游戏中开启。游戏中，通信模块还开启了三个其它端口，它们在初始化期间进行自定义。由于设计工具中没有声明这些端口，请参阅 [Wwise SDK 文档](#) 的“初始化通信”一节了解详细信息。

自定义通信端口的方法是：

- 通过执行以下操作之一来打开 Project Setting 对话框：
 - 在菜单栏中，点击 Project > Project Settings。
 - 按 Shift+K。
- 切换到 Network 选项卡。
- 根据需要编辑端口号。
- 点击 OK。

此时 Project Setting 对话框将关闭。从现在起将使用新端口与游戏进行通信。

相关主题

- [连接到本地/远程游戏系统](#)
- [排查远程连接问题](#)

定义工程的默认用户设置

在 Wwise 中，您可以使用 Default User Settings 对话框为工程定义一系列默认用户设置。在 Wwise 中，您可以使用 Default User Settings 对话框为工程定义一系列默认用户设置。较好的习惯是在工程开发早期定义这些设置，以便能将它们应用于您在工程中创建的所有对象。

在项目开发中更改默认对象设置只会影响在更改后创建的对象。此功能可以简化您的工作流程：您可以先设置一类对象（例如 SFX 对象）的默认属性，然后更改这些设置用于另一类对象（例如语音对象）。

这些默认设置存储在当前工程目录下的 WSETTINGS 文件中。这些设置可由每位用户单独定义，因此处理工程不同部分的用户能够设置不同的默认设置。

为工程中的对象定义默认用户设置的方法是：

1. 通过执行以下操作之一打开 Default User Settings 对话框：

- 在 Project (工程) 菜单中，选择 Default Setting。
- 按 Shift+D。

2. 要为工程中的声音和音乐对象设置默认通路，则需要执行以下操作：

在 Routing 中的 Audio Output 部分单击浏览按钮 (...)。

此时将会打开 Project Explorer —— Browser。

选择一条音频总线作为工程中声音和音乐对象的默认总线。

点击 OK。

新的声音和音乐对象将自动通过此音频总线进行输出。

3. 为所有新建的对象定义默认音量，在 Voice Volume (声部音量) 文本字段中指定值。

4. 要重设工程中使用的默认 Conversion Setting 共享集，请执行以下操作：

在 Default Conversion Settings 中勾选 Override Project Settings 复选框。

点击 Browse (浏览) 按钮 (...)。

此时将会打开 Project Explorer —— Browser。

浏览 Conversion Settings 层级结构并选择一个共享集，用作工程中的默认 Conversion Settings。

点击 OK。

所有新父级对象将自动使用所选的 Conversion Settings 共享集。

5. 完成后，点击 OK。

此时 Default Setting 对话框将关闭，并保存默认用户设置。

相关主题

- [定义工程设置](#)
- [设置用户偏好](#)

排查工程的问题

在 Wwise 中，一般可通过以下两种方式探查工程内出现的问题：

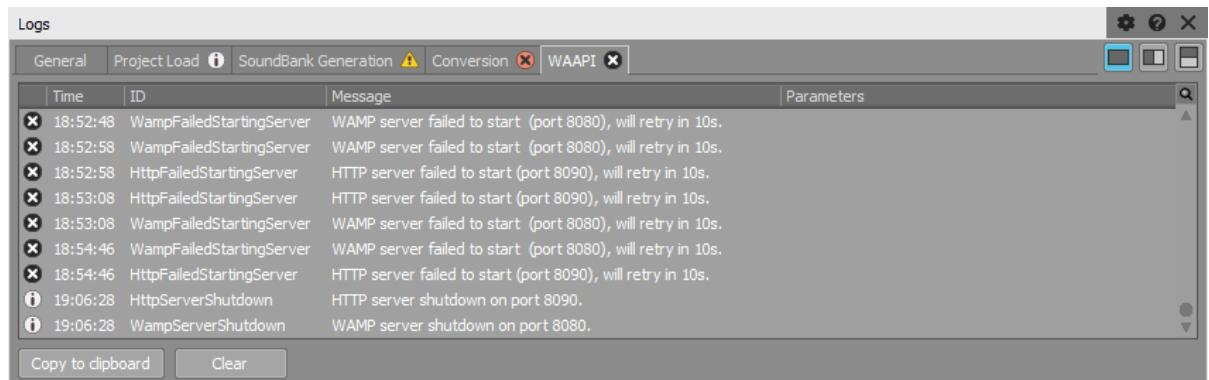
- [使用 Logs 视图](#)
- [使用完好度报告](#)

使用 Logs 视图

通过 [Wwise 工具栏](#)的日志标志，可快速查看工程中显示的消息、警告和错误。如需了解更多详细信息，请将鼠标悬停在标志上。



若要查看所列日志条目的全部信息，请单击标志以便打开 Logs（日志）视图。其中的五个选项卡涵盖了 Wwise 工程中可能遇到的所有问题。



对于包含日志条目的选项卡，每个都会以图标显示其中最高的严重级别。您必须排除所有 Error（错误），因为其会影响操作的完整性。Fatal Error（严重错误）甚至会导致无法完成操作。Warning（警告）可能会影响操作的完整性，所以要加以评估。Message（消息）应该问题不大，但其提供有宝贵的信息，便于了解 Wwise 工程的运行情况。若日志中没有任何条目，则选项卡不会显示“严重级别”图标。



技巧

Logs 视图选项卡中的所有列标题都可单击排序，以便高效地管理条目。比如，可按严重级别的高低来对 Severity（严重性）列进行排序（从低到高或从高到底）。

使用完好度报告

您可以通过生成完好度报告来检查工程的状态。完好度报告中显示有关平台、音频文件 SoundBank 和插件问题的信息。通过检查此报告，您可以处理许多工程问题，例如：

- 缺失的媒体文件
- 缺少源，例如缺失语言版本
- 插件问题
- SoundBanks 缺少对象
- SoundBanks 的渲染和旁通效果问题
- SoundBanks 的渲染和 RTPC

除了显示工程问题外，完好度报告还提供包括下列问题的信息：

- **Platform:** 平台。发生问题的平台。
- **Type:** 类型。受影响的对象类型。
- **Object name:** 对象名称。受影响的对象或元素的名称。
- **Status:** 状态。问题的描述。
- **Comments/Suggestions:** 备注/建议。有关如何处理问题的信息和建议。
- **Hierarchy:** 层级结构。层级结构中受影响对象的位置。

相关主题

- [生成完好度报告](#)
- [完好度报告问题](#)

生成完好度报告

在 Wwise 工程开发期间，您可以随时生成完好度报告。生成 SoundBank 之前，建议先生成完好度报告。如果您也使用了版本控制软件，则应在提交各 Work Unit 中的重大更改之前生成该报告。

为简化特定问题类型的查找，您还可以通过指定要生成的信息类型来筛选此报告。您可以限制报告显示下列任意或全部内容：

- 音频文件和源
- 层级结构
- 参考源
- 优化

在生成报告后，您可以仔细查看报告中的各个问题，然后双击各个问题来执行建议的修复办法。例如，双击内容为“*No group assigned to the Switch Container*”（无编组被分配到切换容器）的状态消息，将打开 Property Editor，在此您可以将 Switch Group 或 State Group 分配给 Switch Container。

生成完好度报告的方法是：

1. 在 Wwise 菜单栏中，点击 Views > Integrity Report。

此时将打开 Integrity Report。

2. 要设置完好度报告中包括哪些平台的问题，请在 Platforms 中勾选所需平台，可选平台是在 [Platform Manager](#) 中定义的。
3. 要设置完好度报告中包括哪些语言的问题，请在 [Languages](#) 中选择 Languages 选项并勾选所需语言。
4. 要筛选工程问题类型，则要选择以下若干项内容：

音频文件和源

层级结构

引用失效

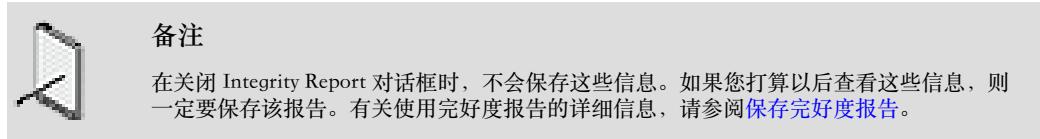
优化

5. 点击 **Generate**。

此时进度条将显示生成过程的状态。

6. 当 Wwise 完成生成完好度报告时，点击 **Close**（关闭）。

此时 Integrity Report- Completed（完好度报告——完成）对话框将关闭，并显示完好度报告。



7. 双击问题以打开相应的对话框，在该对话框中您可以修复或修改问题状态。

完好度报告问题

下表列出了工程中检测到的可能问题，以及如何纠正这些问题。

问题	信息/建议
找不到原始音频文件。	执行以下操作之一： <ul style="list-style-type: none">• 将缺失的文件拖到 Originals 文件夹。• 使用 Audio File Importer（音频文件导入程序）导入缺失的文件。
音频源缺失。	双击消息以在 SFX Project Explorer 的 Audio 选项卡中和语音的 Contents Editor 中查看缺失的源。
平台上不支持源插件。	双击消息以打开 Contents Editor，在此您可以选择当前平台支持的源插件。
没有安装源插件。	安装源插件。
缺少语言源。	使用 Audio File Importer 导入缺少的语言源。
选定的编解码器不支持此源的声道配置。	在转码前，Wwise 将自动下混源。双击消息并更改编解码器（音频格式）或声道设置。
平台上不支持音频效果插件。	双击消息以打开 Property Editor，在此您可以选择当前平台支持的效果插件。
没有安装音频效果插件。	安装缺少的音频效果插件。
Render Effect（将效果器渲染到 SoundBank 中）已经应用于非音频文件的“使用中的”源。	源插件不支持渲染效果。双击消息以执行以下操作之一： <ul style="list-style-type: none">在 Property Editor 中，取消选中渲染效果选项。在 Contents Editor 中，选择所引用音频文件状态为“In Use”（使用中）的音频源。

问题	信息/建议
必须渲染选定的效果。	双击以打开 Property Editor 并执行以下操作之一： 为效果器选择 Render 选项。 更改使用的效果器。
先对效果做渲染可以节省 CPU，因为在游戏中将不会改变其参数。	双击打开 Property Editor 并选择 Render 复选框。
已旁通此对象上的效果。将忽略渲染效果。	由于不能同时支持这两个操作，因此请验证渲染效果对您项目的重要性。如果想应用渲染效果选项，则要取消选中 Bypass Effect（旁通效果）。
已为此对象的 RTPC 指定 Bypass Effect 属性。将忽略渲染效果。	由于不能同时支持这两个操作，请验证 Render effect 操作和 RTPC 驱动的 Bypass effect 操作对项目的重要性。如果您想应用 Render 效果器选项，则需取消选中对象上的 RTPC 驱动的 Bypass 效果。
RTPC 已应用于该对象的效果器。将忽略渲染效果。	由于不能同时支持这两个操作，因此请验证已经应用的 RTPC 或渲染效果对项目的重要性。如果您要应用渲染效果，则要取消选中 RTPC。
流 XMA 媒体不支持区域循环。将忽略所有用户定义的循环区域。	执行以下操作之一： <ul style="list-style-type: none">• 双击消息以打开 Property Editor 并取消选中 Loop 复选框。• 在 Conversion Setting 对话框中，更改文件的音频格式。
XMA 文件的循环次数设置不能超过 254。	执行以下操作之一： <ul style="list-style-type: none">• 双击消息以打开 Property Editor，并在 No. of loops（循环次数）文本框中，减小循环次数。• 在 Conversion Setting 对话框中，更改文件的音频格式。
不支持超过 8MB 的内存 XMA 文件。	执行以下操作之一： <ul style="list-style-type: none">• 对声音采用流播放。• 降低采样率或压缩品质。
此虚声部行为需要 Seek Table（文件寻址表）。	双击消息以打开 Conversion Settings 对话框并执行以下操作之一： <ul style="list-style-type: none">• 在 Vorbis Encoder Parameters（Vorbis 编码器参数）对话框中启用 Seek Table。• 更改文件格式。
此虚声部行为不需要 Seek Table。	为减少内存占用，您可以弃用 Seek Table。双击消息以打开 Conversion Settings 对话框并禁用 Vorbis Encoder Parameters 对话框中的 Seek Table。
Vorbis 格式的音乐对象需要 Seek Table。	双击消息以打开 Conversion Settings 对话框并执行以下操作之一： 在 Vorbis Encoder Parameters（Vorbis 编码器参数）对话框中启用 Seek Table。 更改文件格式。
此容器中的子对象没有分配给 Switch。	双击消息以打开 Project Explorer 和相应 Property Editor，在此您可以执行以下操作之一： 移除未分配的对象。 将对象指派给 Switch。
没有组指派给 Switch Container。	双击以打开 Property Editor，并将 Switch Group 或 State Group 指派给 Switch Container。
Switch Container 中没有对象指派给了 Switch。	双击以打开 Contents Editor，并将对象指派给 Switch Container 中的 Switch。
没有组指派给 Music Switch Container。	双击以打开 Property Editor，并将 Switch Group 或 State Group 指派给 Music Switch Container。
没有音乐对象指派给此容器中的 Switch。	双击以打开 Music Switch Association Editor（音乐切换关联编辑器），在此您可以将音乐对象指派给 Switch。

问题	信息/建议
Sequence Container 的播放列表中没有指派好的对象。	双击以打开 Contents Editor，在此您可以将对象指派给播放列表。
音乐播放列表容器的播放列表中没有指派好的对象。	双击以打开 Music Playlist Editor（音乐播放编辑器），并将音乐对象指派给播放列表。
Blend Container 中的混合轨没有指派好的对象。	双击以打开 Blend Track Editor，在此您可以执行以下操作之一： 添加对象到混合轨中。 移除混合轨。
Music Segment 内的音乐轨中不包含音频文件。	双击以打开 Music Segment Editor（音乐段落编辑器），在此您可以执行以下操作之一： 添加片段到音乐轨中。 删除音乐轨。
SoundBank 在引用不存在的对象。	从 SoundBank 中删除事件或对象。
Event Action 中没有指派对象。	双击并打开 Event Editor 来执行以下操作之一： 给这些 Action 指派对象。 移除 Event Action。
在 Dialogue Event 中的路径没有指派对象。	双击以打开 Dialogue Event Editor（对白事件编辑器）并执行以下操作之一： 将对象指派给路径。 移除路径。
此精确到采样点的容器中的部分子对象没有使用与父容器相同的输出属性。	父容器采样可能不再精确到采样点。双击以打开 Schematic View（原理图视图），对于精确到采样点的容器中的子对象，当它们有任何不沿用父对象输出的设定时，您可以在此取消这些设定。
精确到采样点的容器中的部分子对象没有使用与父容器相同的效果属性。	父容器采样可能不再精确到采样点。双击以打开 Schematic View，对于精确到采样点的容器中的子对象，当这些对象有任何不沿用父对象效果器的设定时，您可以在此取消这些设定。
此精确到采样点的容器中的部分子对象没有使用与父容器相同的高级设置属性。	父容器采样可能不再精确到采样点。双击以打开 Schematic View，对于精确到采样点的容器中的子对象，当这些对象有任何不沿用父对象的高级设定时，您可以在此取消这些设定。
此精确到采样点的容器的部分容器没有达到精确到采样点。	父容器采样可能不再精确到采样点。双击以打开 Project Explorer，并将精确到采样点的容器中的非精确到采样点的子容器更改为精确到采样点。
此精确到采样点的容器拥有子 Switch Container。	父容器采样可能不再精确到采样点。双击以打开 Project Explorer，从而移除 Switch Container。
此精确到采样点的容器中的音频源与容器中其它音频源的声音数量不一样。	父容器采样可能不再精确到采样点。验证此容器的所有音频源都具有相同数量的声音。
某些音频文件时间短于 0.05 秒钟。它们不可用于 crossfade（交叉淡变）过渡。	双击以打开 Project Explorer，找到由于音频文件过短而不适合交叉淡变过渡的关联声音对象，并执行以下操作之一。 将时长更长的另一个音频文件拖放到声音对象上。 在父对象的 Property Editor 中，选择另一个过渡类型。
多个 Switch Container 中有多个对象指派给了某个 Switch，这些 Switch Containers 没有正确地进行交叉淡变。	双击以打开 Project Explorer，并在相关联的 Property Editor 中确保只有一个对象指派给一个 Switch，以便实现交叉淡变过渡。
触发速率间隔小于 0.021 秒的容器可能无法按预期播放。	双击打开 Property Editor，将触发速率间隔改为 0.021 秒以上。
此源只包含 LFE 声道。	
此源包含 LFE 声道。	

保存完好度报告

保存完好度报告是一个好办法，这样可以方便您查看报告，从而帮助解决工程中的问题或者为问题保留一份记录。

保存完好度报告的方法是：

1. 在完好度报告中，点击 Save Report（保存报告）。
- Save As 对话框打开。
2. 前往您要保存报告的文件夹。
3. 将默认报告名称替换成最适合此报告内容的名称，然后点击 Save（保存）。

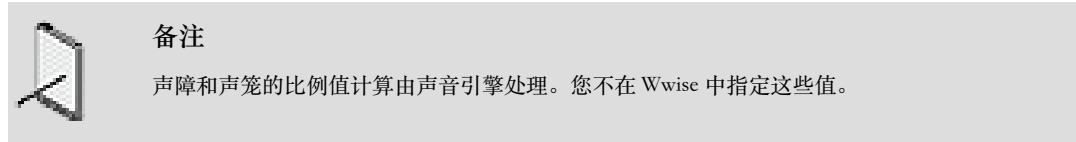
完好度报告将作为文本文件保存在您指定的位置。

学习附录 —— 有关声障和声笼的详细信息

Obstruction（声障）和 Occlusion（声笼）是为遮挡声源与听者之间的空间而设置的条件。声障指部分遮挡，声笼指全部遮挡。这些条件由游戏程序员设置，他们通过编程的方法来定义游戏的几何空间。Wwise 自身不计算声障和声笼的程度。这些物理计算必须由游戏完成，然后使用以下函数将结果传送到声音引擎：[SetObjectObstructionAndOcclusion\(\)](#)。

在 Wwise 中，您只需按游戏中声音对象受到声障和声笼的影响的情况对声音对象应用音量和 LPF 即可。您定义的值是相对值，会加到已应用于游戏对象的任何音量和 LPF 值上。

在这些条件的实现中，对声障和声笼指定比例值，以增强真实感。这些比例基于游戏引擎的实时定位信息，可以反映游戏对象出入遮挡区域的移动情况。例如，特工角色可以穿过布满大型雕塑的画廊。声障的比例可以根据特工从各个雕塑背后通过时所处的位置进行变化。同样，当特工悄悄溜出完全封闭的杂物间时，声笼的比例也可以随之进行变化。



要想更好地理解发生声障和声笼的条件，请参阅以下主题：

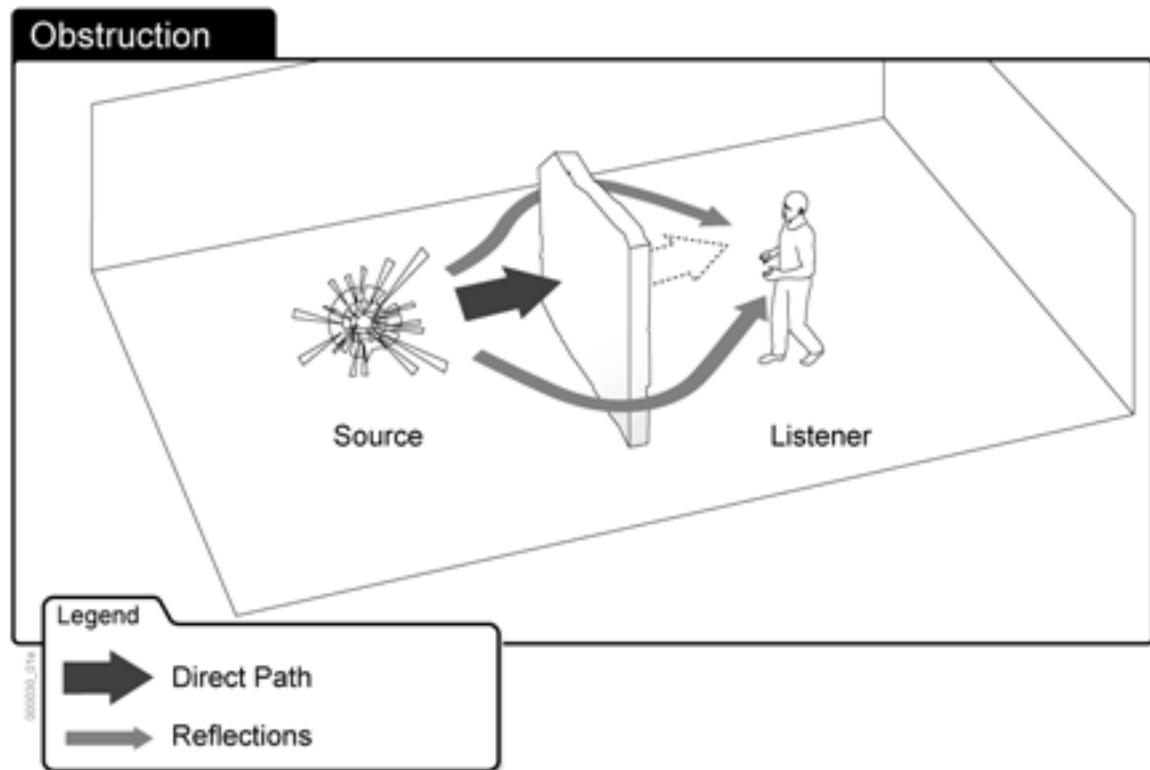
- [理解声障](#)
- [理解声笼](#)

理解声障

声障是指游戏几何空间中的对象（例如墙壁或立柱）部分遮挡声源与听者之间空间时的声学现象。例如，在特工游戏中，尽管玩家躲在柱子后面，但仍可以听到前方的枪声。

在声障条件下，听者可以清楚地听到声音的反射声，但直达声听起来很模糊。声音的改变通过降低音量、增加 LPF 或者同时调整两者来实现。

下图演示声障的条件。

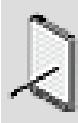
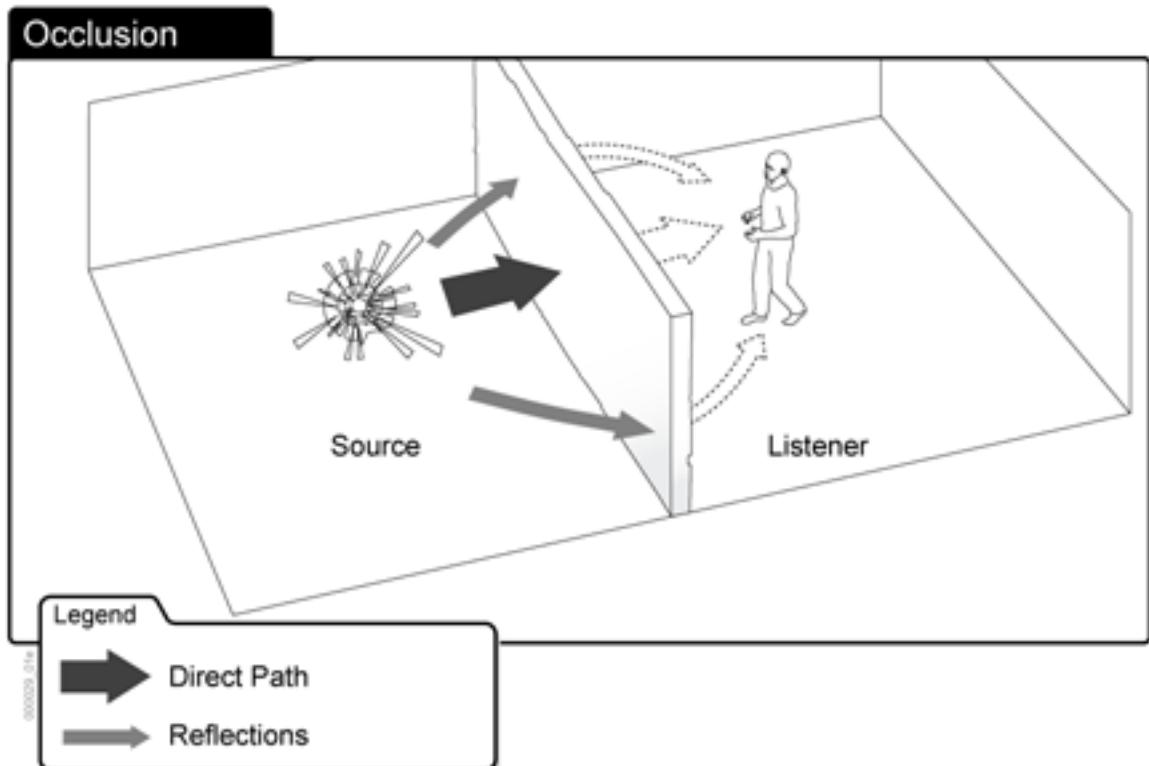


理解声笼

Occlusion 是指游戏几何空间中的对象完全挡住了声源与听者之间的空间时的声学现象。例如，在特工游戏中，虽然隔着一堵墙壁，但玩家仍可以听到隔壁的枪声。

在声笼条件下，直达声和反射声都很模糊。与声障一样，您可以更改声音的音量和 LPF。

下图演示声笼的条件。



备注

声障和声笼设置可以与 Environmental Effect（环境效果）并行使用，根据音源的位置来动态改变声音属性。关于使用环境效果的更多信息，请参阅 [理解发送](#)。

工程管理技巧与经验总结

您最好仔细阅读以下章节，一系列的技巧和经验总结可帮助您更好地管理您的工程。

使用完好度报告

定期生成完好度报告是帮助您处理任何工程问题的好办法，在以下情况尤其如此：

- 在生成 SoundBank 之前 —— 允许您在将音频和振动打包到 SoundBank 中之前，先解决工程里的问题。
- 提交 Work Unit 中的重大更改之前 —— 允许您确保所做的更改没有导致工程出现问题。

使用默认设置

您可以使用默认设置来"multi-edit"（同时编辑）工程的默认通路和音量。当您更改这些值时，更改后添加的所有对象都将采用新的默认值。因此，针对在更改这些设置后添加的对象，您可能会有不同的默认值。

定义声障和声笼曲线

总是以线性方式来定义这些曲线是一个好方法，这可以最大限度地降低工程中的 CPU 和内存占用。总是以线性方式来定义这些曲线是一个好方法，这可以最大限度地降低工程中的 CPU 和内存占用。

将大型工程分割成多个工作单元

如果是多人协作完成一个大型工程，可能需要使用 Work Unit 将工程划分成更细的部分。工作单元是独特的 XML 文件，其中包含与工程中特定部分或元素相关的信息。这些工作单元可帮助您组织和管理工程中的各种不同元素。如果是团队协作，还可通过版本控制系统来管理这些工作单元，以便其他团队成员同时来处理此工程。将工程划分成 Work Unit 后，您可以卸载其中一部分，只保留正在处理的 Work Unit。从工程中卸载 Work Unit 的主要目的是提升工程加载速度，减少内存使用量，并提高 Wwise 的整体性能。如果工程很大，那么您可以通过卸载一个或多个工作单元来大幅提高性能。

第 4 章 管理多平台

概述	87
使用 Platform Manager	87

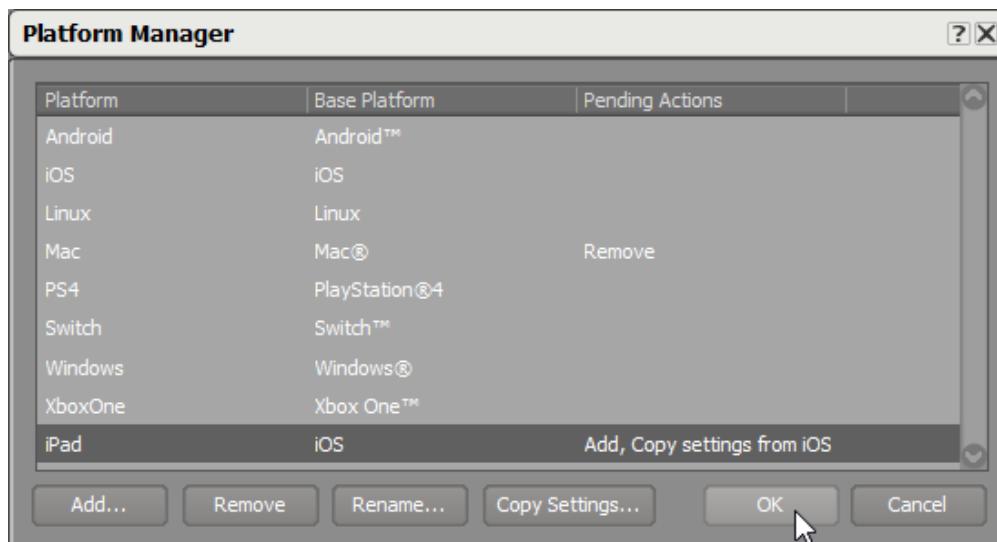
概述

一款游戏经常会针对多个平台发布多个版本。因此具备为工程指定目标平台的能力至关重要。在 Wwise 中，Platform Manager（平台管理器）是用来实现这个功能的。

在 Wwise 菜单栏中，选择 Project（工程）> Platform Manager… 来调用 [Platform Manager](#)。默认快捷键：Shift+Alt+P。

使用 Platform Manager

Platform Manager（见下文）用于为游戏指定平台。平台由平台名称和基础 SDK 组成。对于任何列出的 SDK，可以随心所欲地指定任意数量的平台。



Platform Manager

Wwise 包含一系列基础平台，它们涵盖与 Wwise 一起使用的基本 SDK。您可以根据与工程相关的所有基础平台自由地创建自己的平台。但是，为了求同存异，您还可能需要添加多个拥有同一基础平台的平台。

比如，可以在工程中同时开发 iPhone 和 iPad 平台。虽然从本质上说它们都属于 iOS 平台，但它们并不完全一致。为了方便不同工程类型的管理，可以创建两个基于 iOS 的不同平台：一个专门针对 iPad，另一个专门针对 iPhone。

相关主题

- [添加、删除和复制平台](#)

添加、删除和复制平台

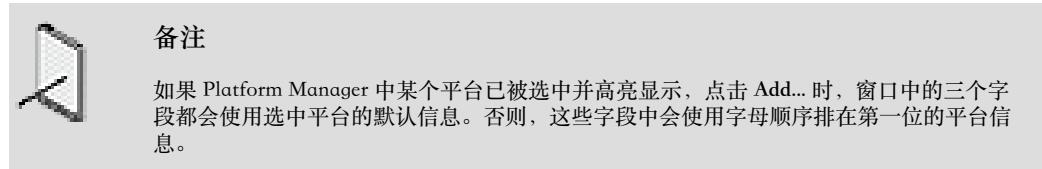
在 Wwise 菜单栏中，选择 Project > Platform Manager…。

此时将显示 Platform Manager。

添加平台

1. 点击 Add... (添加...)。

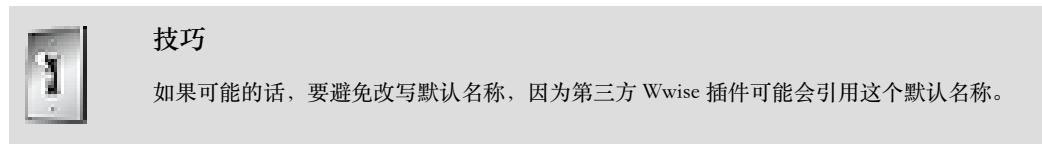
Add Platform 对话框将显示。



2. 从 SDK 列表中选择 Base Platform (基础平台)。

名称字段相应更新了。

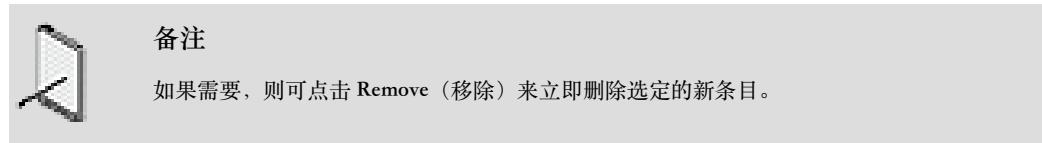
3. 如果需要, 则可以选择 Override Default Name (改写默认名称)。这将激活名称字段, 在此可以指定新平台的名称。名称必须使用字母和数字 (不含空格, 但可使用下划线), 并且不能和已确定的平台名称重名。



4. 或者, 选择一个现有平台来Copy settings from (复制其设置)。

5. 点击 OK 确认新平台信息并关闭 Add Platform 对话框。

新平台的名称、基础平台和待处理操作 (“Add”, 可能还有“Copy settings from...”) 将在 Add Platform 对话框底部新插入行中显示。



6. 点击 OK 以确认新添加的平台。

此时将显示一条警告, 说明这些平台更改不可撤消。

7. 点击 Yes (是) 以继续进行待执行操作。

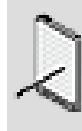
此时将显示另一条警告, 提醒您在工程重新加载后验证关于平台的更改, 特别是路径名称。

8. 点击 OK 以重新加载工程。

此时工程将关闭, 然后使用新的平台设置重新加载。根据工程的大小, 这可能需要若干秒钟。

移除平台

1. 从列表中选择要移除的平台。
2. 点击 Remove (删除)。
3. 选定平台的 Pending Actions 列将显示 Remove。



备注

要取消 Remove 操作，可以关闭 Platform Manager 或点击 Cancel。

4. 点击 OK 以激活被选中平台的移除操作。

此时将显示一条警告，说明这些平台更改不可撤消。

5. 点击 Yes (是) 以继续进行待执行操作。

此时将显示另一条警告，提醒您在工程重新加载后验证关于平台的更改，特别是路径名称。

6. 点击 OK 以重新加载工程。

此时工程将关闭，然后使用新的平台设置重新加载。根据工程的大小，这可能需要若干秒钟。

复制平台设置

您可以选择将从以前定义的平台中把平台专有的设置复制到另一个平台。

如果多个平台的许多设置相同，那么这种复制操作可以为您节省大量的时间。

1. 点击 Copy settings… (复制设置…)。

Copy Platform Settings (复制平台设置) 对话框将弹出。

2. 在 From (源) 列表中选择源平台。
3. 在 To (目标) 列表中选择目标平台。
4. 点击 OK 以确认新添加的平台。

此时将显示一条警告，说明这些平台更改不可撤消。

5. 点击 Yes (是) 以继续进行待执行操作。

此时将显示另一条警告，提醒您在工程重新加载后验证关于平台的更改，特别是路径名称。

6. 点击 OK 以重新加载工程。

工程先关闭掉，然后重新加载新的工程[从源平台复制到目标平台的平台相关设置](#)。根据工程的大小，这可能需要若干秒钟。



注意

确保为工程保留一份最新的备份，以便您稍后决定恢复这些更改。



警告

此操作不可撤消，并且会消除剪贴板上的 Wwise 相关内容和撤消操作 (Undo) 的历史记录。

相关主题

- 将设置从一个平台复制到另一个平台

第 5 章 管理语言

概述	92
使用 Language Manager	92

概述

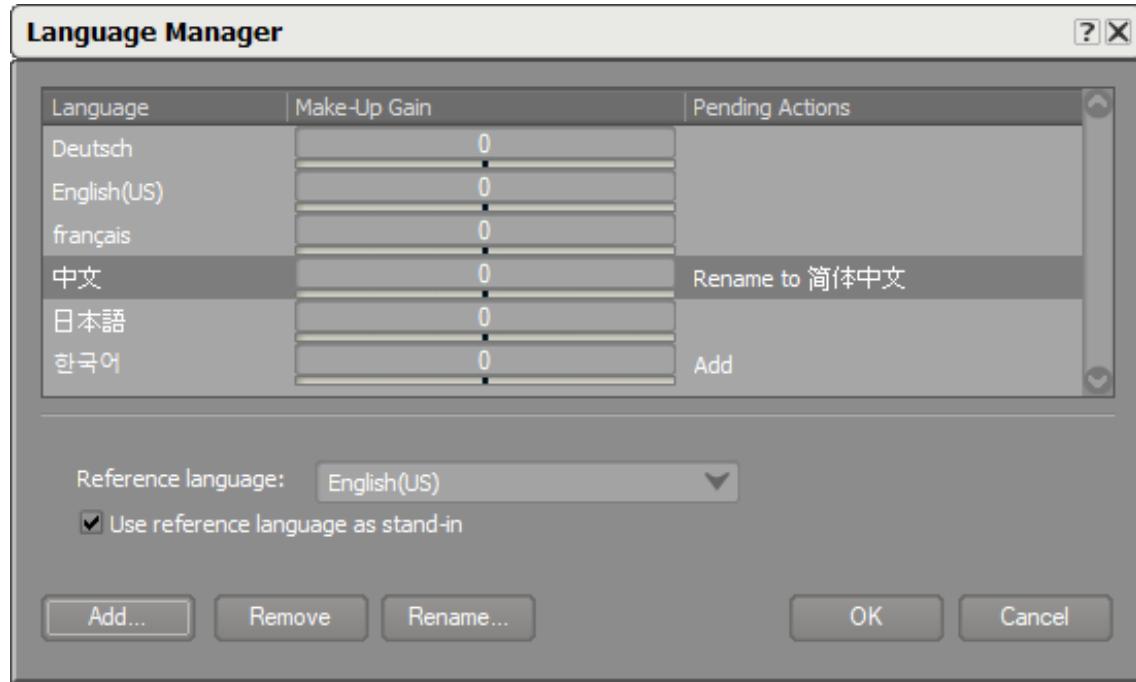
一款游戏通常都会发行多种语言版本，因此，能够为一个工程指定多种语言就显得至关重要。在 Wwise 中，Language Manager 是用来实现这个功能的。

在 Wwise 菜单栏中，选择 Project > Languages… 来调用 Language Manager。默认快捷键为：Shift+J。

使用 Language Manager

在确定要本地化的语言后，您可以为工程创建这些语言。这些工程语言显示在工具栏的 Language Selector 和语音对象的 Contents Editor 中。由于在创建列表时不是所有语言文件都可用，因此您可以选择参考语言来替代尚不可用的语言。参考语言的 Conversion Settings 也可以作为其他语言文件的默认 Conversion Settings。

Language Manager 可让您定义工程的语言，以及指定作为参考语言的语言。



创建工程的语言

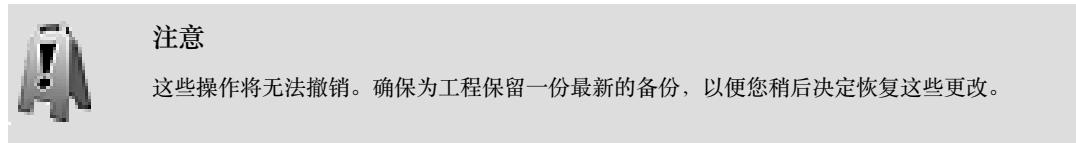
您可以在 Language Manager 中创建工程语言。在创建语言时，您还可以为语言文件定义音量偏置。

在 Wwise 菜单栏中，选择 Project > Languages…。此时将显示 Language Manager。

在此，您可以进行以下操作：

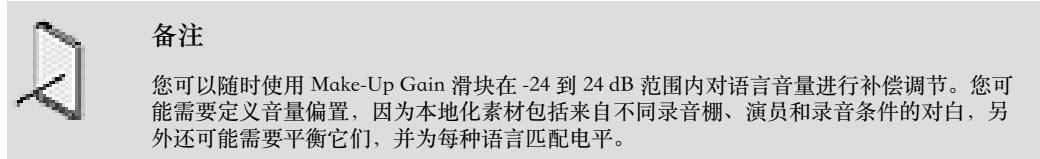
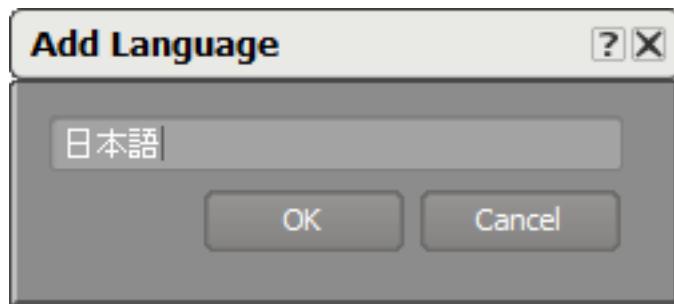
- [添加语言](#)；
- [移除语言](#)；

- 移除语言；及
- 设置参考语言。



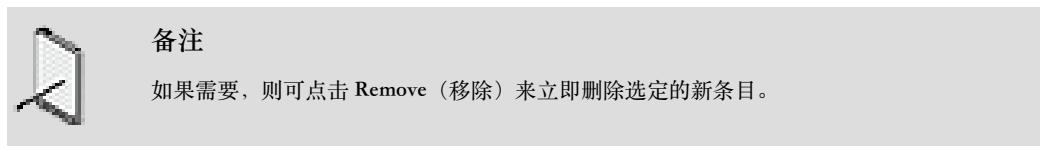
添加语言

1. 点击 Add... (添加...)。
- Add Language 对话框将显示。
2. 输入语言名称，可以使用任何 OS 支持的字符。



3. 点击 OK 确认新增语言，并关闭 Add Language 对话框。

新增语言将显示在 Language Manager 底部，Pending Action 列将显示待处理操作 (“Add”)。



4. 点击 OK 以确认新添加的语言。

此时将出现消息框，警告您将清空 Wwise 剪贴板中的内容和撤消历史记录。

5. 点击 Yes (是) 以继续进行待执行操作。

另一则警告将弹出，提醒您确认语言变更，尤其是 GeneratedSoundBanks 和 Original 文件夹中的语言路径变更。

6. 点击 OK 以接受更改。

Process Log 将打开，并显示指定操作已完成的消息。工程的 Language Selector 列表中也将显示最新添加的语言。

移除语言

1. 从列表中选择要移除的语言。
2. 点击 Remove (删除)。
3. 选定语言的 Pending Actions 列将显示为 Remove。



4. 点击 OK 以激活被选中语言的移除操作。

此时将显示一条警告，说明这些语言更改不可撤消。

5. 点击 Yes (是) 以继续进行待执行操作。

另一则警告将弹出，提醒您确认语言变更，尤其是 GeneratedSoundBanks 和 Original 文件夹中的语言路径变更。

6. 点击 OK 以接受更改。

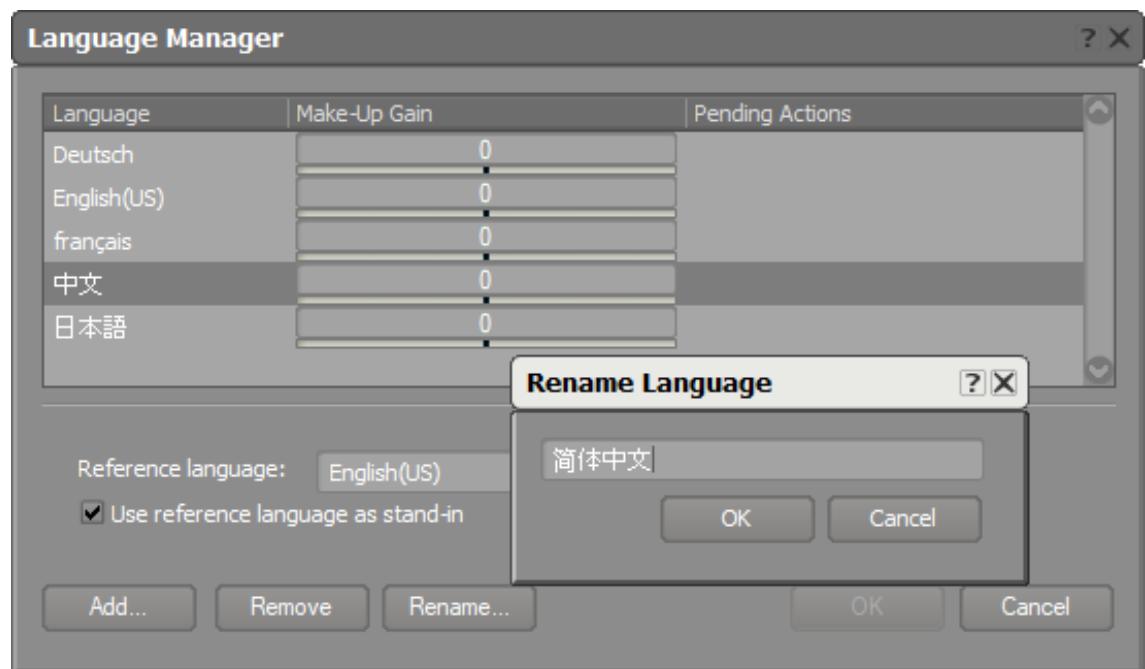
Process Log 将打开，并显示指定操作已完成的消息。工程的 Language Selector 列表中将会删除指定语言。

Renaming Languages (重命名语言)

1. 点击 Rename...。

此时将打开 Rename 对话框。

2. 输入新的语言名称，可以使用任何 OS 支持的字符。



3. 点按 OK 确认新的语言名称，并关闭 Rename Language 对话框。

新的语言名称将显示，待处理操作将显示为“Rename language”。

4. 点击 OK 以确认新添加的语言。

此时将出现消息框，警告您将清空 Wwise 剪贴板中的内容和撤消历史记录。

5. 点击 Yes (是) 以继续进行待执行操作。

另一则警告将弹出，提醒您确认语言变更，尤其是 GeneratedSoundBanks 和 Original 文件夹中的语言路径变更。

6. 点击 OK 接受更改。

Process Log 将打开，并显示指定操作已完成（也可能有错误）的消息。工程 Language Selector 列表中将显示新的语言名称。

更改 Reference Language

1. 从 Reference language 列表中选择一种语言。
2. 您也可以启用 Use reference language as stand-in (将参考语言作为临时替代文件)，这样，对于本地化音频文件尚未到位的语言，会将参考语言的音频文件暂时纳入其音频包中。
3. 点击 OK 确认新参考语言及其临时替代状态。

此时将出现消息框，警告您将清空 Wwise 剪贴板中的内容和撤消历史记录。

4. 点击 Yes (是) 以继续进行待执行操作。

另一则警告将弹出，提醒您确认语言变更，尤其是 GeneratedSoundBanks 和 Original 文件夹中的语言路径变更。

5. 点击 OK 接受更改。

Process Log 将打开，并显示指定操作已完成（也可能有错误）的消息。

相关主题

- [对工程进行本地化](#)

第 6 章 管理 Workgroup

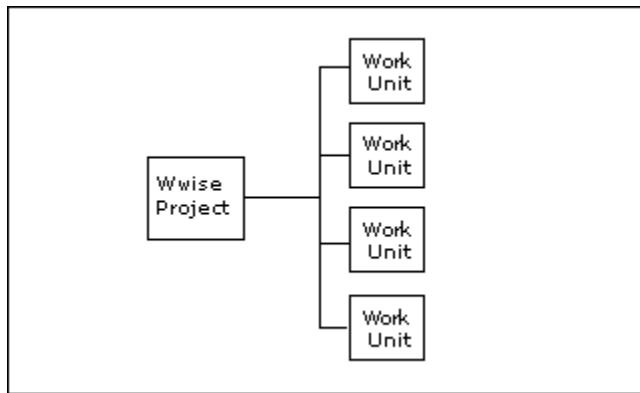
概述	97
将工程分成 Work Units	100
查看工程文件的状态	107
结合版本控制系统使用 Wwise	109
解决工程中的不一致现象	109
使用 Workgroup 插件管理工程文件	111
Workgroup 技巧与经验总结	122

概述

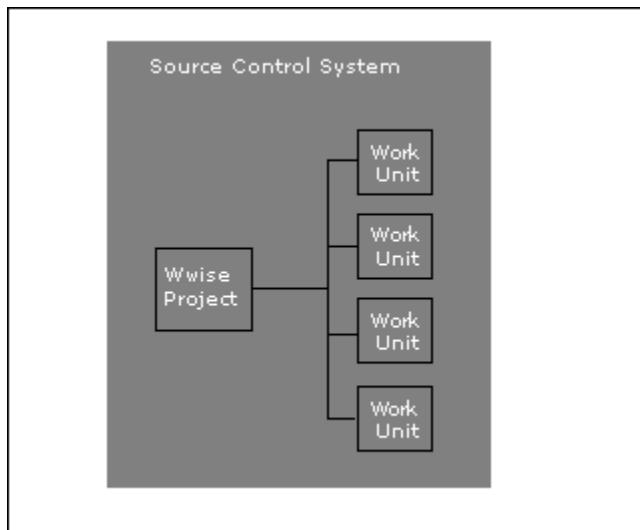
在当今的游戏环境中，随着游戏日趋复杂并且面临着上市的压力，声音设计师、作曲家、音频集成师和音频程序员能够协同处理同一工程至关重要。在 Wwise 中，可以使用 Workgroup（工作组）来做到这一点。

每款游戏只能使用一个 Wwise 工程。

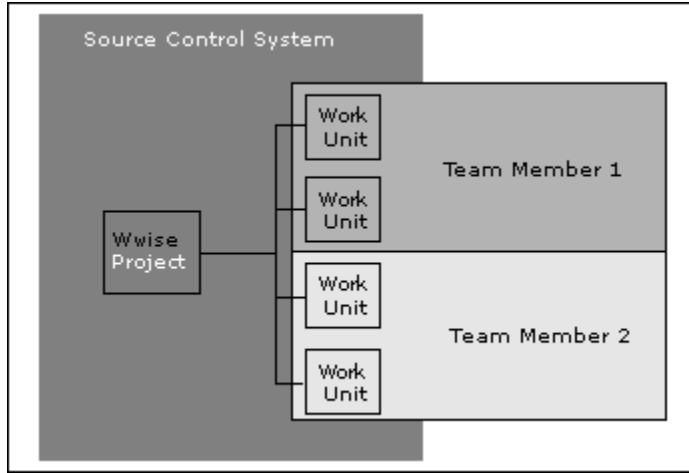
为了让多人高效地协同处理同一工程，必须将工程分解成多个部分。在 Wwise 中，这些部分被称为工作单元。



然后可通过版本控制系统来管理这些工作单元文件。

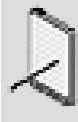


工作组中的每个成员于是可并行地处理工程的同一部分或不同部分。



在大多数情况下，为了避免出现棘手和频繁的合并问题，您需要不同的成员处理工程的不同部分。然而在某些情况下，两个或多个成员必须并行处理同一工作单元。当这些文件重新 check-in 到版本控制系统时，您很可能需要处理合并冲突。请参阅版本控制文档，了解有关如何妥善处理合并冲突的详细信息。

虽然 Wwise 不是一款版本控制系统，但是您可以使用它的开放架构轻松地集成现有版本控制系统。它可以让您直接在 Wwise 中管理工程素材并执行许多版本控制功能。有关 Wwise 版本控制插件的详细信息，请参阅[使用 Workgroup 插件管理工程文件](#)。



备注

为了创建工作组插件，您的版本控制系统需要能够使用第三方集成的 API 来支持这些集成。Wwise 中安装了 Perforce 和 Subversion 插件。

什么是 Work Unit?

Wwise 工作组的基础是工作单元。工作单元是独特的 XML 文件，其中包含与工程中特定部分或元素相关的信息。这些工作单元可帮助您组织和管理工程中的各种不同元素。如果是团队协作，还可通过版本控制系统来管理这些工作单元，以便其他团队成员同时来处理此工程。

在创建工程时，Wwise 中将为下列每个元素创建一个默认工作单元：

- Actor-Mixer Hierarchy (角色混音器层级结构)
- Audio Devices
- Attenuations
- Control Surface Session
- Conversion Settings (转码设置)
- Dynamic Dialogue (动态对话)
- Effects (效果器)
- Events (事件)
- Game Parameters

- Interactive Music Hierarchy
- Master-Mixer Hierarchy
- Mixing Session
- Modulators (调制器)
- Presets (预设)
- Queries (查询)
- SoundBanks (声音库)
- Soundcaster Sessions (声音选角器会话)
- States (状态)
- Switches (切换开关)
- Triggers

这些默认工作单元位于工程目录下的相应文件夹中。每个都命名为“Default Work Unit.wwu”。默认 Work Unit 已经创建，这样您就可以直接新建对象、事件、状态等，而不必先为每个工程元素创建 Work Unit。

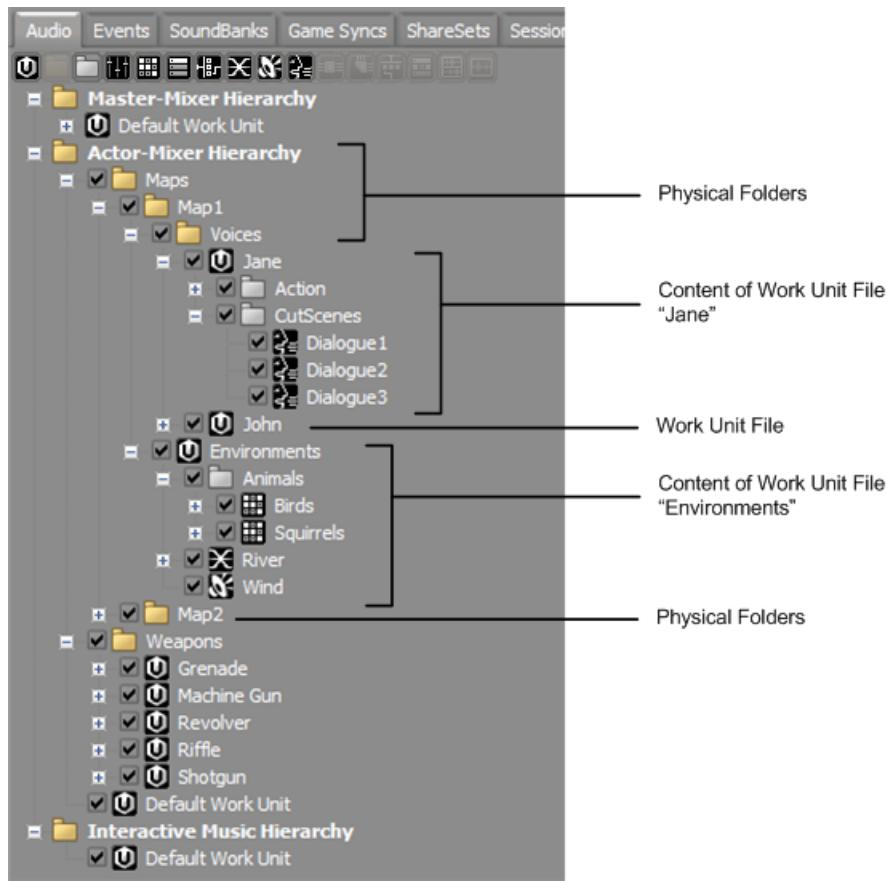
随着项目进展或者更多人员加入项目团队，您可能需要将不同的工程元素分成新的工作单元。例如，您可以为 StatesLevel1、StatesLevel2 和 StatesLevel3 状态创建三个不同的工作单元。

如果您决定创建新的工作单元，则可以将默认工作单元保留为空白。然而，Default Work Units 是关键的工程文件，不得重命名或删除。如果您重命名或删除这些文件，Wwise 则将在您下次打开工程时重新创建它们。

工作单元可以按照实子文件夹来组织存储在磁盘上。Project Explorer 中将复制此实文件夹结构。

您可以为除 Preset 和 Master-Mixer Hierarchy 之外的所有其他工程元素创建新的工作单元。Master-Mixer Hierarchy 中的总线必须始终保留在它们的默认工作单元中。

下面的例子显示了如何创建和组织工作单元，从而划分 Actor-Mixer Hierarchy 中的声音结构。



您可以在 Project Explorer 中的不同选项卡中创建和管理工作单元的内容。有关将工程分成工作单元的详细信息，请参阅[将工程分成 Work Units](#)。

将工程分成 Work Units

在划分工程前，首先必须确定组织工程中不同元素的最佳方法，以简化工作流程和减少冲突。由于划分工程的方法有很多，因此值得抽出时间来为您的工程确定最佳的划分方式。

在创建工作单元后，您可以通过将声音结构、事件、SoundBank 等拖到各自的工作单元中来划分工程中的工作。您还可以从您的工程版本中卸载特定工作单元，只保留您正在处理的工作单元，以缩短工程加载时间和降低内存占用。

管理工程中的工作单元包括以下任务：

- [在工程中创建 Work Unit](#)
- [将工程元素指派到 Work Unit](#)
- [在工程中加载/卸载 Work Unit](#)

在工程中创建 Work Unit

您可以为工程中的以下元素创建新的工作单元：

- Actor-Mixer Hierarchy 中的声音结构
- Interactive Music Hierarchy 中的音乐结构

- 动作 Event 和对白 Event
- SoundBanks
- 切换开关、状态、游戏参数和触发器
- 效果器和衰减共享集
- Soundcaster 和 Mixing Desk 会话
- Queries

每个工作单元只可包含为它创建的工程元素。例如，Event 工作单元中只可包含事件，效果器工作单元中只可包含效果器，以此类推。

如果您打算使用版本控制系统管理工程文件，可以将所有工作单元（包括默认工作单元）添加到版本控制系统的中央资源库（repository）或文档库（depot）中去。如果在使用工作组插件时创建工作单元，则会提醒您将文件添加到资源库中。

- 创建新的工作单元

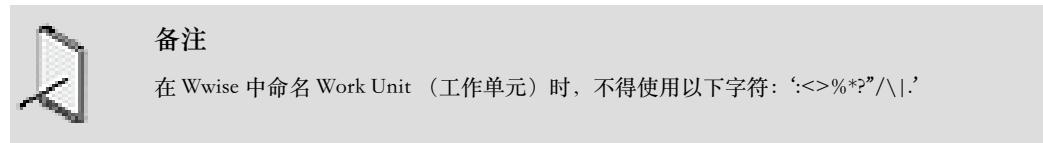
1. 在 Project Explorer 中，右键点击实文件夹之一（例如：Actor-Mixer Hierarchy 或 Interactive Music Hierarchy）。

此时将会显示快捷菜单。

2. 点击 New Child > Work Unit。

Work Unit 对话框打开。

3. 在 Name 字段中，输入Work Unit 的名称。



4. 点击 OK。

新的工作单元于是创建了。



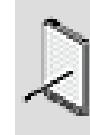
相关主题

- [将工程元素指派到 Work Unit](#)
- [重命名用户创建的 Work Unit](#)
- [删除用户创建的 Work Unit](#)
- [Workgroup 技巧与经验总结](#)

将工程元素指派到 Work Unit

在默认情况下，工程中的所有元素（包括声音结构、事件和 SoundBank）会自动指派到各自的默认工作单元。在创建工作单元后，您可以将工程中的不同元素指派到

新创建的工作单元。您只需将工程元素拖到特定的工作单元，就可以将它指派到工作单元中。



备注

在将工程元素移到新的工作单元或者在现有工作单元之间移动之前，您应确保工程团队中的其他成员当前没有在处理会受到影响的工程元素。

将工程元素指派到 Work Unit

1. 在 Project Explorer 中，点击其中一个选项卡。（包括 Audio、Events、SoundBanks、Game Syncs、ShareSets、Sessions 和 Queries 选项卡。）
2. 将 Default Work Unit 中的若干工程元素拖至您创建的 Work Unit 中。

此时工程元素将指派到新的工作单元。

相关主题

- [在工程中创建 Work Unit](#)
- [Workgroup 技巧与经验总结](#)
- [管理实文件夹](#)
- [Nested Object（嵌套工作单元）](#)
- [在工程中加载/卸载 Work Unit](#)
- [重命名用户创建的 Work Unit](#)
- [删除用户创建的 Work Unit](#)

管理实文件夹

在工程特定对象类别的实际磁盘文件夹中（如 Actor-Mixer Hierarchy 文件夹），可以在根目录下新建子文件夹，用于组织 Work Unit 文件。如果您有数百个工作单元，并且不希望它们位于同一文件夹中，这样就尤其有用。

实文件夹和虚文件夹是两个截然不同的实体。实文件夹代表磁盘中的实际目录，而虚文件夹包含在工作单元文件中，是该工作单元的一部分。两者均用于组织内容，但 Physical Folder 是 Work Unit 的上级文件夹，而 Virtual Folder 是 Work Unit 的下级文件夹。

在 Project Explorer 中可直接创建实文件夹，方法是在其他实文件夹中使用上下文菜单或使用 Project Explorer 工具栏按钮。

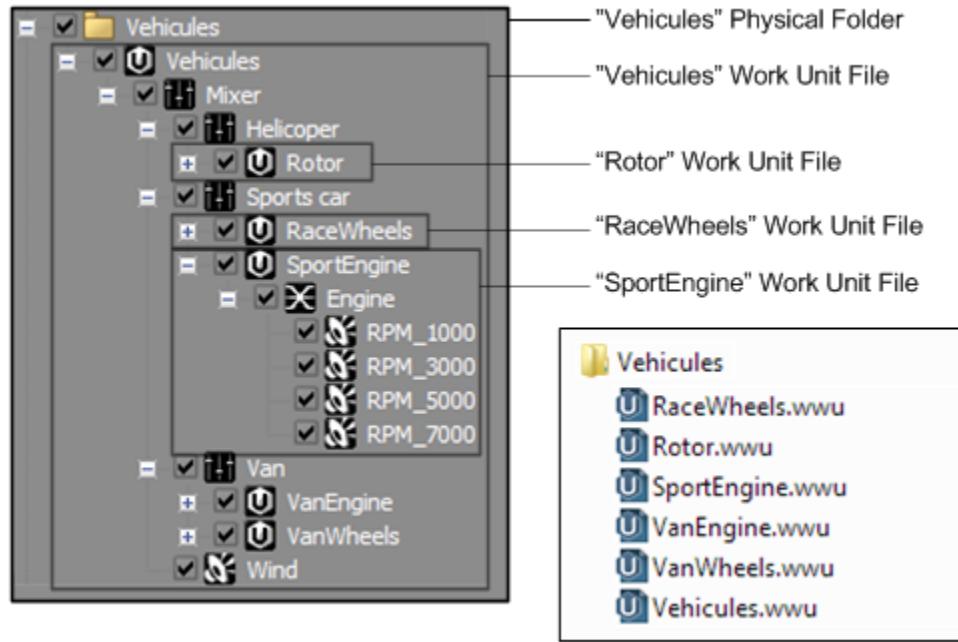
实文件夹只可包含其他实文件夹或工作单元文件。

Nested Object（嵌套工作单元）

工作单元可嵌套在其他工作单元内。如果这样做，工作单元中的内容则将委托给嵌套工作单元而不是父工作单元。这样可以将多个工作单元放置到一个通用的角色混音器对象下，为嵌套工作单元提供共同的行为和混音属性。

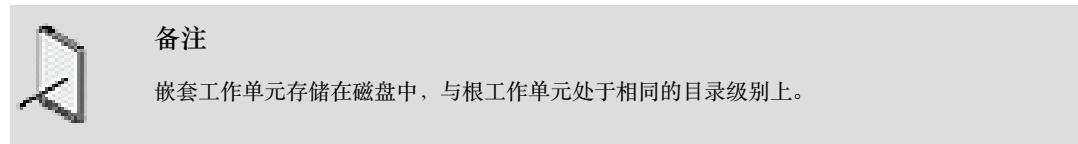
嵌套工作单元支持更细的文件划分粒度，从而可以减少合并时的潜在冲突。

嵌套工作单元保存在包含它们的实文件夹中。同一个实文件夹内的工作单元不能重名。



以下对象类型下可以创建嵌套工作单元：

- 实文件夹
- Work Units
- 虚拟文件夹
- Actor-Mixer (角色混音器)

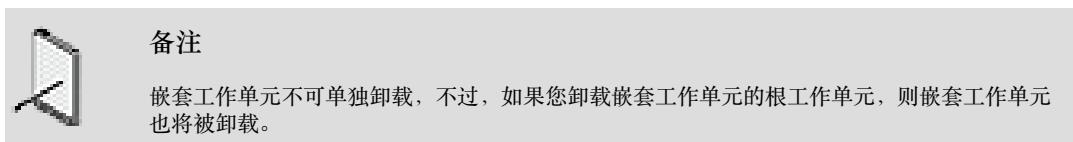
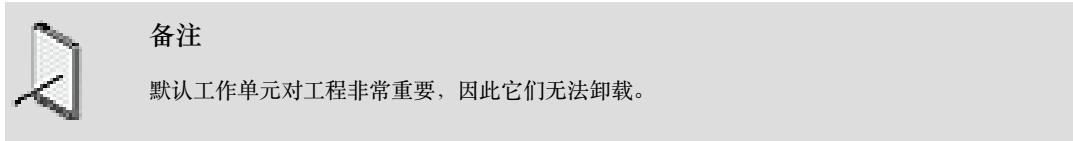


在工程中加载/卸载 Work Unit

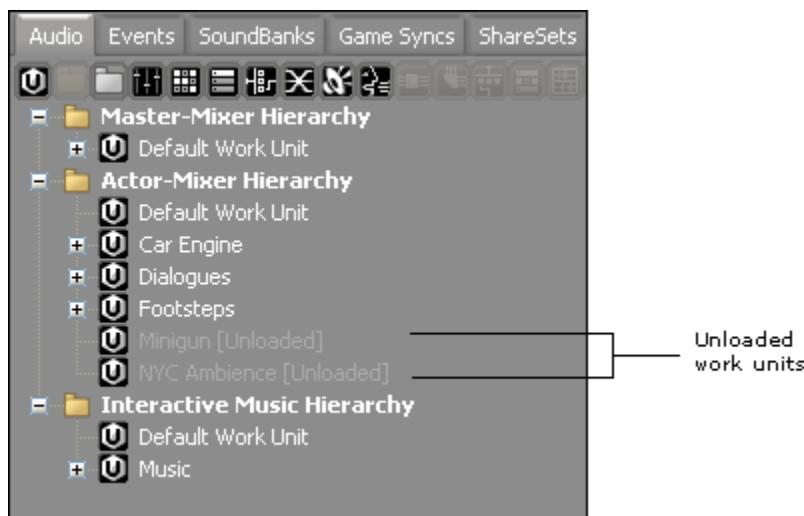
在工程分成多个工作单元后，您可能希望卸载其中部分工作单元，而只保留您正在处理的工作单元。从工程中卸载工作单元的主要原因是加快工程加载时间，降低内存占用，提高总体性能。如果工程很大，那么您可以通过卸载一个或多个工作单元来大幅提高性能。

可从工程中卸载用户创建的下列工作单元类型：

- Actor-Mixer Hierarchy 工作单元
- Interactive Music Hierarchy 工作单元
- Event 工作单元



从工程中卸载工作单元后，它们在 Project Explorer 中将显示为灰色。直到将它们重新载入工程后才可使用它们。



您可以继续处理已加载工作单元中的对象和事件。然而，您可能会发现，部分对象和事件包含针对已卸载对象和/事件的引用。例如，在以下工程元素中可能有针对已卸载对象或事件的引用：

- Dialogue events (对白事件)
- Event
- 插播乐句
- Music Transitions
- Queries
- SoundBanks
- Soundcaster 和混音会话

在这些情况下，已卸载的元素将标记为 [Unloaded]，并以黄色高亮显示。

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	>>			

您仍可添加/移除对象，修改属性值，生成 SoundBank 等，但在 Wwise 中无法播放这些对象和事件。

从工程中卸载工作单元时，工作单元并不会立即卸载。只有在关闭工程时，才会将您选择要卸载的工作单元从工程中卸除。当您下次打开工程时，Wwise 便不会加载已经卸载的工作单元了。如果您需要访问已卸载工作单元的内容，则可以随时轻松地将它重新加载回工程中。如果对工程作出的任何更改会影响已卸载工作单元内的对象或事件，当这些工作单元重新加载到工程中时，Project Load Log 中将列出这些更改。



备注

当工作单元从工程中卸载时，Wwise 无法区分未使用的和没有加载到工程中的源文件。其结果是，直至所有工作单元重新加载到工程中并且 Wwise 可以执行全面验证之前，表面上看起来工程未使用的所有源文件将在 File Manager 的 Usage 列中标为“Unknown”。

从工程中卸载工作单元的方法是：

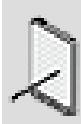
1. 在 Project Explorer 中，选择您要卸载的工作单元。
2. 右键点击选定项，并从菜单中选择 **Unload Work Unit (at next project load)**。
3. 按 **Ctrl+S** 保存工程。
4. 点击 **Project > Project Name** 重新打开工程。

工程打开时，以前已经卸载的工作单元将不再加载。

将工作单元重新加载到工程中的方法是：

1. 在 Project Explorer 中，选择您要重新加载到工程中的已卸载工作单元。
2. 右键点击选定项，并从菜单中选择 **Load Work Unit**。

这些工作单元于是重新加载到工程中了。



备注

如果对工程作出的任何更改会影响已卸载工作单元内的对象或事件，当这些工作单元重新加载到工程中时，Project Load Log 中将列出这些更改。

相关主题

- [在工程中创建 Work Unit](#)
- [将工程元素指派到 Work Unit](#)

- 重命名用户创建的 Work Unit
- 删除用户创建的 Work Unit

重命名用户创建的 Work Unit

在某个时候，您可能需要对您在工程中创建工作单元重命名。您可以在 Project Explorer 中对工作单元重命名，也可以在 File Manager 中进行。不可在 Windows 资源管理器或 Mac Finder 中对工作单元重命名或删除工作单元，否则将可能导致完好度错误或丢失工程数据。



注意

Default Work Units 是关键的工程文件，不得重命名或删除。如果这样做了的话，Wwise 则将在您下次打开工程时重新创建它们。

对工作单元进行重命名的方法是：

1. 在 Project Explorer 中，按 F2 或使用 Rename 快捷菜单项。

此时可编辑的文本框将显示在工作单元名称上方。

2. 在编辑框中，输入新名称，然后按 Enter。

此时将显示 Workgroup Operation 对话框。

3. 验证要应用的操作，并点击 Continue。

如果您使用版本控制插件，则要验证一下看看操作是否成功了。

4. 点击 OK 以关闭 Process Log。

相关主题

- [什么是 Work Unit?](#)
- [在工程中创建 Work Unit](#)
- [将工程元素指派到 Work Unit](#)
- [删除用户创建的 Work Unit](#)

删除用户创建的 Work Unit

您可能需要删除您创建的并且在工程中不再需要的工作单元。您可以在 Project Explorer 中删除工作单元，也可以在 File Manager 中删除它们。不得在 Windows 资源管理器或 Mac Finder 中删除工作单元，否则可能造成完好度错误或丢失工程数据。

如果您从版本控制系统中删除了某个工作单元，则该工作单元内的工程元素将不再存在于工程中，这可能会造成完好度错误。如有必要，当您打开工程时，这些错误以及如何解决它们的说明将一起显示出来。有关工程不一致性的详细信息，请参阅[解决工程中的不一致现象](#)。



注意

Default Work Units 是关键的工程文件，不得重命名或删除。如果这样做了的话，Wwise 则将在您下次打开工程时重新创建它们。

删除工作单元的方法是：

1. 执行以下操作之一：
 - 在 Project Explorer 中，选择要删除的工作单元，然后按 Delete 或使用快捷菜单中的 Delete 功能。
 - 此时将显示 Workgroup Operation 对话框。
2. 验证要应用的操作，并点击 Continue。

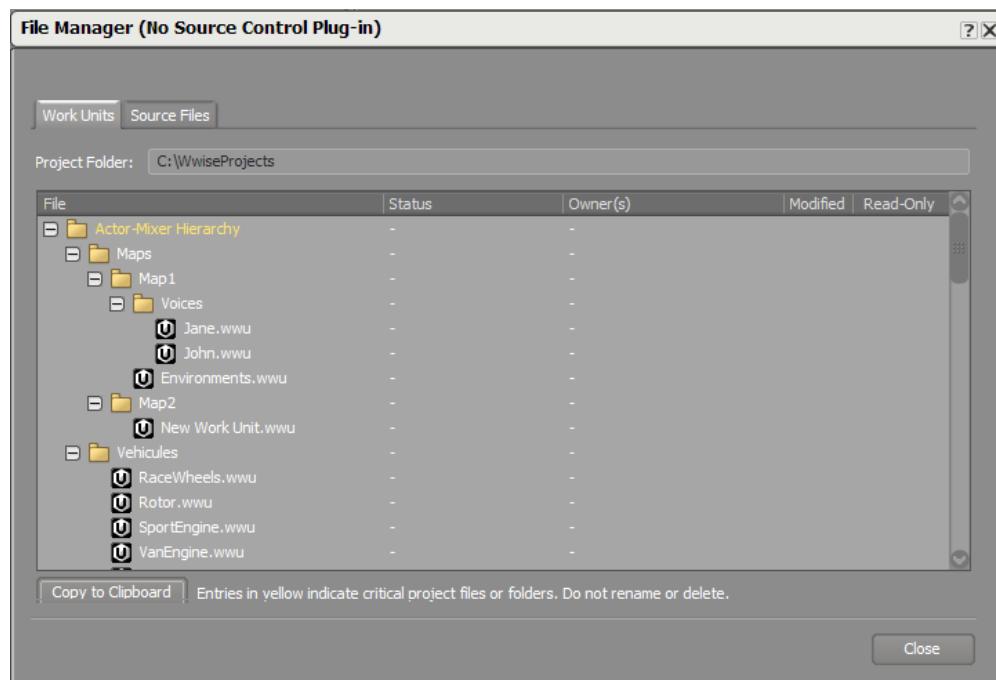
如果您使用版本控制插件，则要验证一下看看操作是否成功了。
3. 点击 OK 以关闭 Process Log。

相关主题

- [什么是 Work Unit?](#)
- [在工程中创建 Work Unit](#)
- [将工程元素指派到 Work Unit](#)
- [重命名用户创建的 Work Unit](#)

查看工程文件的状态

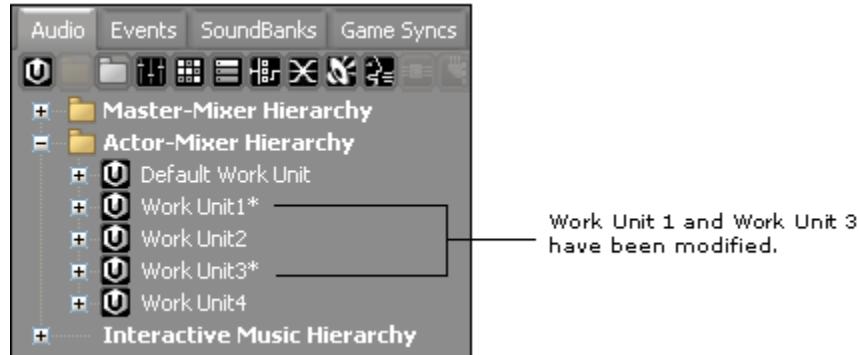
随着工程不断取得进展，您需要跟踪工程中的各种文件。File Manager 显示工程中每个音频文件和工作单元以及工程文件本身的信息。您可以使用 File Manager 验证工程文件或源文件的状态和所有者，以及确定工程中的音频源是否正在使用源音频文件。



当工作单元被修改或者标记为只读后，该工程文件的相应列中将显示选中标记。此类反馈可帮助您跟踪您已经修改的和不可修改的文件。

File Manager 还显示工程正在使用的文件的信息，以便您定期清理不再使用的文件。然而当工作单元从工程中卸载后，Wwise 不可再分辨没有使用的和尚未加载到工程中的源文件。这些情况下，在未经 Wwise 完成验证之前，工程中可能未使用的所有源文件将在 Usage 列中标记为 Unknown。

Wwise 还提供反馈信息，显示哪些文件已经直接在 Project Explorer 中进行了修改。与工程一样，Wwise 使用星号 (*) 显示已经修改并且需要保存的工作单元。



注意

如果您更改只读工作单元，则在保存工程时将不保存这些工作单元文件。有关保存工程的详细信息，请参阅[保存工程](#)。

查看工程文件状态的方法是：

1. 执行以下操作之一：

- 在菜单栏中，点击 Project > File Manager...。
- 按 Shift+F1。

此时将打开 File Manager 对话框。

2. 仔细查看工程文件以验证：

- 它们是否经过了修改。
- 它们是否为只读文件。

3. 切换到 Source File 选项卡，并查看源文件以验证：

- 它们是否经过了修改。
- 它们是否为只读文件。或者
- 它们是否正在被工程中的若干个音频源使用。



注意

Usage 列中显示的信息不一定总是最新的。在删除被标记为 Unused 的任何文件之前，最好关闭并重新打开工程，以确保信息是最新的。

4. 点击 OK 以关闭 Project File Status 对话框。

相关主题

- [保存工程](#)

- [解决工程中的不一致现象](#)
- [Workgroup 技巧与经验总结](#)

结合版本控制系统使用 Wwise

在工程开发环境中，您可能已经使用版本控制系统来有效地管理素材和其它类型的工程文件。所有 Wwise 工程文件（包括各个工作单元）都基于 XML，这意味着用户也可以使用版本控制系统轻松地管理这些文件。

工程中的以下文件可通过版本控制系统进行管理：

- Wwise 工程文件——WPROJ 文件。
- Wwise 工作单元——WWU 文件，包括默认工作单元。
- Originals 文件夹——此文件夹中包含导入到 Wwise 中的原始声音文件。
- 生成的 SoundBank——为各个平台和语言生成的工程 SoundBank 文件。



注意

位于工程目录中的 .cache 文件夹是 Wwise 的一个本地工作文件夹。.cache 文件夹的内容不得添加到版本控制系统，因为这可能造成 Wwise 中发生异常行为。

在游戏的整个开发过程中，用户都可以在 File Manager 中查看工程文件（WPROJ）、工作单元文件（WWU）和音频文件的状态。如果您使用 Perforce、Subversion 或其它工作组插件，则可以直接在 Wwise 中执行版本控制功能。有关在 Wwise 中使用工作组插件的详细信息，请参阅[使用 Workgroup 插件管理工程文件](#)。

当您是工作组的成员，并使用版本控制系统管理工程文件时，应该时刻注意其他人也正在处理这一工程，因此可能会有一些合并冲突需要解决。因此，经常同步和合并您的工作，并频繁与团队成员沟通交流您正在做的工作非常重要。有关经验总结的完整列表，请参阅[Workgroup 技巧与经验总结](#)。

解决工程中的不一致现象

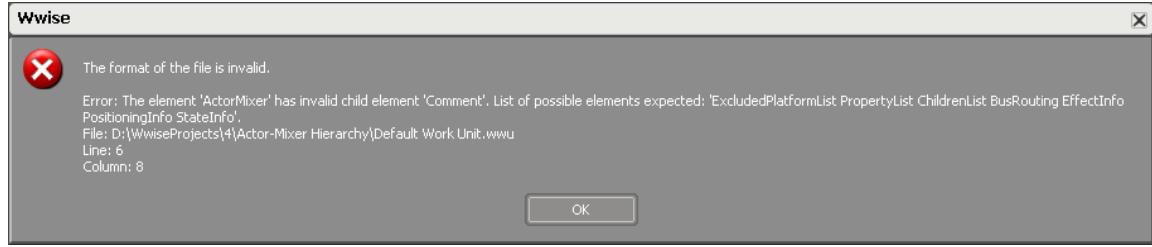
当多人同时处理一个工程时，更改特定文件可能导致工程文件出错或不一致。为了减少这些问题的影响，Wwise 在您每次打开工程文件时执行两种类型的工程验证：

- XML 语法和 Schema 错误验证
- 工程一致性验证

这些验证可以帮助您解决错误，防止发生更多的工程问题。

XML 语法和 Schema 错误

当工程文件 check-in 或合并到版本控制系统时，您可能需要更新 XML 代码本身来解决冲突。如果在更新期间导致 XML 语法或 Schema 错误，工程文件则将变为无效，并且 Wwise 将无法打开工程。此时将显示一个消息框，对错误进行描述，并指明错误所在的具体文件和位置。

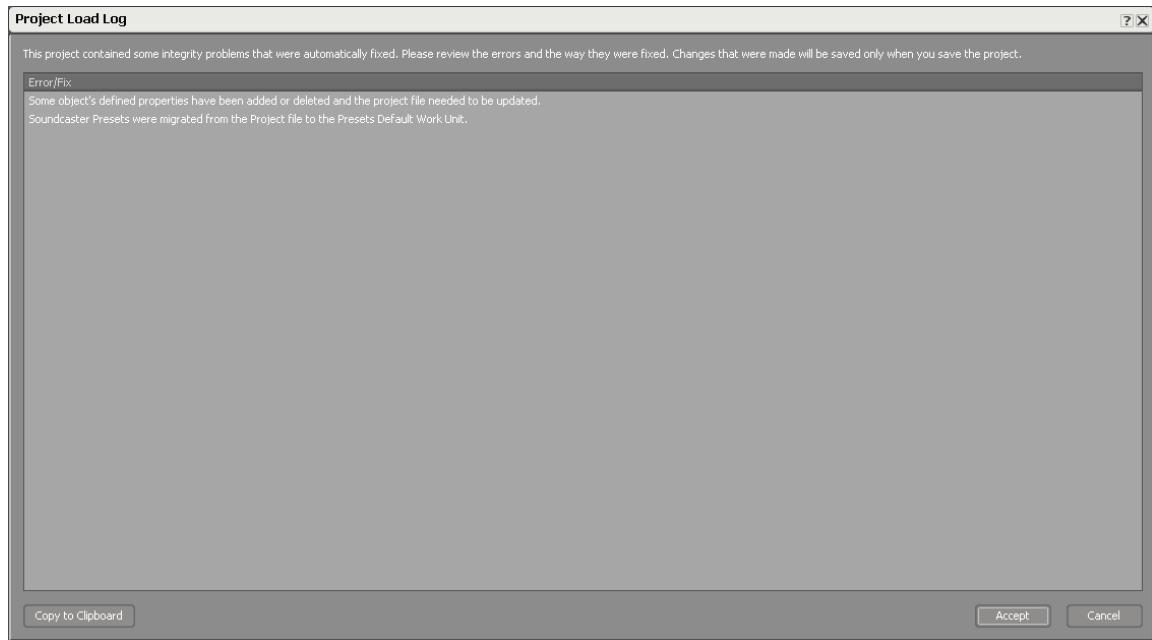


要解决问题，您必须查看错误中列出的各个工作单元文件。

工程不一致问题

如果工程文件中不存在 XML 语法错误，那么 Wwise 将继续验证是否存在任何工程不一致性或错误。例如，如果在 States 工作单元中删除某个 State，但声音或音乐结构工作单元之一中的某个对象仍在使用它，则会发生工程不一致。

如果 Wwise 检测到任何工程问题，必要时它则将显示有关各个问题的信息以及如何解决该问题的说明。



您可以选择全盘接受这些修复方案，也可以选择拒绝这些方案，然后恢复工程的旧版本或者尝试直接修正 XML 文件中的不一致性。

解决工程不一致性的方法是：

1. 在 Wwise 中打开工程。

如果找到工程不一致性，则必要时 Project Load Log 将打开，其中包含工程问题的完整列表以及 Wwise 建议的修复办法。

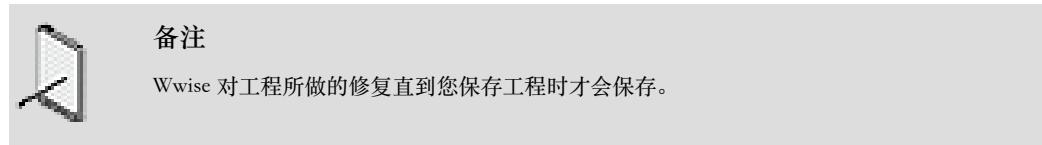
2. 执行以下操作之一：

- 点击 Accept 以接受建议的修复办法。

Wwise 将修复不一致性并打开工程。如果您接受这些修复办法，则应逐一审查这些办法，评估一下是否需要手动修复问题。

- 点击 **Cancel** 以拒绝建议的修复办法。

此时 Wwise 将关闭工程。您可以恢复到工程的旧版本，也可以尝试直接修正 XML 文件中的不一致性。



使用 Workgroup 插件管理工程文件

Wwise 的开放架构可以方便您创建 Workgroup 插件来集成版本控制软件。Workgroup 插件在 Wwise 和版本控制软件之间创建一个链接，以便您直接在 Wwise 中管理您的文件，执行版本控制功能，例如 check-in 和 check-out。

由于各个版本控制系统工作原理不同，因此可用的特定功能和您采用的工作流程将取决于您使用的系统。

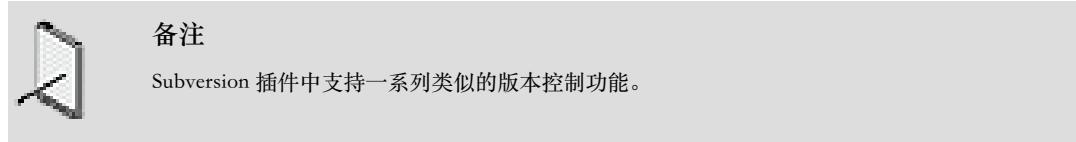
为了帮助您迅速入门，Wwise 为软件配置管理系统 Perforce® 和 Subversion 提供了两个功能齐全的 Workgroup 插件示例。Perforce 和 Subversion 均不采用锁定-修改-解锁模式，而采用复制-修改-合并模式。在这种模式下，工作组的每个成员读取资源库或文档库并在他们的工作站中创建一个客户端工作区 (workspace)，此空间是一个目录，里面包含文件和/或工程的个人工作副本 (working copy)。这样，团队的各个成员都可以并行工作，修改他们自己的个人副本。做 Prepare 时，各个成员可以将他们的个人副本合并到一个新的最终版本中去。如果存在冲突，版本控制系统则将协助您完成合并，但最终将由您负责确保正确解决所有问题。

如果您使用其它版本控制管理系统，则可以为 Wwise 创建您自己的工作组插件。有关创建和集成您自己的工作组插件的信息，请参阅 SDK 文档中的[如何创建版本控制插件 DLL](#)一节。

在使用版本控制插件时，Wwise 中可直接使用众多的版本控制功能。例如在使用 Perforce 插件时，可以在 Wwise 中执行下列版本控制操作：

- **Get latest version** —— 通过从文档库中检索最新版本来更新工作副本。
- **Submit changes** —— 将待提交的文件发送到 Perforce 服务器进行处理。
- **Check out** —— 在客户端工作区中创建文档库文件的工作副本。
- **Lock** —— 锁定文件，使其它客户端无法将文件的工作副本提交到服务器。
- **Unlock** —— 解锁文件，使其它客户端可以将文件的工作副本提交到服务器。
- **Mark for add** —— 将文件添加到文档库。当文件被标记为添加时，它将被添加到更改列表 (changelist)，然后必须将它提交到文档库。
- **Mark for delete** —— 将文件从文档库中删除。当文件被标记为删除时，它将被放置到更改列表中，然后必须将它提交到要删除它的文档库中。
- **Move** —— 在 Originals 目录中的文件夹之间移动源文件。
- **Rename** —— 重命名文档库中的文件。

- **Revert changes** — 放弃您对客户端工作区中的某文件所做的更改。
- **Resolve** — 调解文件两个修订版本之间的差异。
- **Diff** — 将客户端工作区中的文件与文档库中的文件进行比较。
- **File History** — 显示文件的修订历史记录。



您不仅可以在 Wwise 中访问这些版本控制命令，而且还能通过 Project Explorer 中特殊的叠加显示图标，来帮助快速识别工作单元文件的状态。例如，下表显示 Perforce Workgroup 插件中使用的叠加图标。

叠加图标	名称	描述
	Normal (not checked out)	文件处于最新状态，没有人将它 check-out。
	Marked for add Moved	文件已被标记为添加到 Perforce 文档库或者被移动过了因而带有 move / add (移动/添加) 状态。
	Checked out (Open for Edit)	文件已从 Perforce 文档库中登出。
	Checked out by another user	文件已被工作组中的另一个用户 check-out 了。
	Concurrently checked out	文件已被您和工作组中的另一个用户同时 check-out 了。
	Outdated (非文件的最新修订版)	服务器上存在一个更新的修订版本。您可以获取文件的最新版本。



支持的 Perforce／Subversion 版本

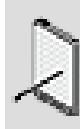
在使用任一工作组插件前，请查看以下信息，确保 Perforce 或 Subversion 的版本与该工作组插件兼容。

Perforce

Perforce 工作组插件通过 Perforce SDK 版本 2018.1 创建，但可与 Perforce 服务器的大多数版本兼容。如果您使用的是与工作组插件不兼容的不同版本的 Perforce，您可以 Wwise SDK 中的源和您正在使用的版本重新构建插件。有关 Perforce 插件的详细信息，请参阅 [Wwise SDK 文档](#)。

Subversion

Subversion 工作组插件通过 Subversion 版本 1.8.5 创建。如果您使用的是与工作组插件不兼容的不同版本的 Subversion，您可以 Wwise SDK 中的源和您正在使用的版本重新构建插件。有关重新构建 Subversion 插件的详细信息，请参阅 [Wwise SDK 文档](#)。



备注

Subversion 的 bin 文件夹在默认情况下位于 C:\Program Files\Subversion\bin 下，必须在系统的 PATH 环境变量中设置这个文件夹路径。Subversion 安装程序通常会将这条路径添加到 PATH 中，但是，如果从上文提到的 ZIP 文件手动安装 Subversion，您则必须将 Bin 文件夹手动添加到 PATH 中去。



技巧

Perforce、Subversion 和其他 Platform SDK 的更新都会在我们 SDK Documentation 的 [What's New](#) 部分发布。

以下各节描述了如何使用工作组插件管理素材。虽然它们主要针对 Perforce 工作组插件，但使用 Subversion 的工作流程与其相似。

- [在 Perforce 中设置工程文件](#)
- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

在 Perforce 中设置工程文件

在开始使用 Perforce 工作组插件前，您或者您的系统管理员需要执行以下操作：

- 在工作组每位成员的工作站中安装和设置版本控制管理软件的客户端版本。
- 在工作组每位成员的工作站中配置客户端工作区。
- 从文档库中获取工程文件的最新版本，并将它们保存在客户端工作区中。
- 在 Wwise 中选择并配置工作组插件。

有关安装和配置 Perforce 的详细信息，请联系系统管理员或参阅 Perforce 文档。

您可以在 Project Settings 对话框中选择并配置工作组插件。有关在 Wwise 中选择并配置工作组插件的详细信息，请参阅[配置版本控制插件](#)。

相关主题

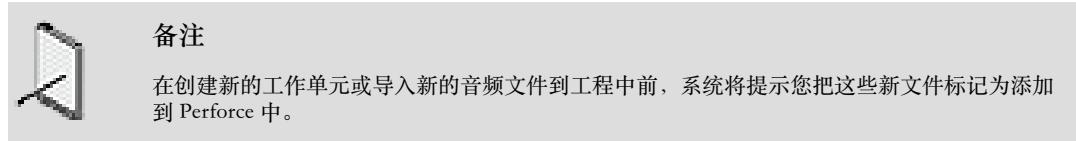
- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)

- 在使用 Perforce 时保存工程

将文件添加到 Perforce

在开始开发工程前，应将工程文件和音频源添加到文档库或资源库。将文件添加到 Perforce 包括以下两个步骤：

- 将文件添加到更改列表的方法是：.
- 将您的更改提交到文档库的方法是：.



将文件添加到更改列表的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 切换到以下某个选项卡：

- Work Units，将工作单元添加到 Perforce
- Sources，将音频文件添加到 Perforce。

3. 选择您要添加到 Perforce 的文件。

4. 右键点击选定的工程文件，并从菜单中选择 Mark for add。

此时 Process Log 对话框打开，显示选定文件已经添加到更改列表或者“opened for add（已为添加操作打开）”并可提交到文档库。

5. 点击 OK 以关闭 Process Log。

文件的状态设置为“add”。

6. 点击 Close 以关闭 File Manager。

在 Project Explorer 中，工程文件此刻便有了 Marked for add 叠加图标。

此时，这些文件只是刚被添加到更改列表中而已，您还需要提交更改列表，以将这些新文件 check-in 到文档库。有关将文件提交到文档库的详细信息，请参阅[将您的更改提交到 Perforce 文档库](#)。

7. 保存工程。

相关主题

- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)

- 在使用 Perforce 时保存工程

从 Perforce 文档库中 check-out 文件进行编辑

在编辑工程中的文件前，首先应该将它们 check-out，确保您的客户端工作区中有这些文件的最新版本。在 Perforce 中 check-out 文件时，这些文件变成可写状态。这意味着您可以保存对文件所做的更改。

在默认情况下，Perforce 不执行排他性 check-out，这意味着团队的其它成员可以同时处理同一文件。沟通是避免出现合并冲突的关键，解决合并冲突耗时又费力。

check-out 文件进行编辑的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 切换到以下某个选项卡：

- Work Units，将工作单元添加到 Perforce
- Sources，将音频文件添加到 Perforce。

3. 选择您要打开进行编辑的文件。

4. 右键点击选定的工程文件之一，然后从菜单中选择 Check out。

此时 Process Log 对话框将打开，其中显示文件已经登出或者“已经打开以待编辑”。

5. 点击 OK 以关闭 Process Log。

文件的状态设置为“edit”。

6. 点击 Close 以关闭 File Manager。

在 Project Explorer 中，工程文件上于是有了 Checked out 叠加图标。

相关主题

- [将文件添加到 Perforce](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

将您的更改提交到 Perforce 文档库

如果处理完特定文件或者希望团队中的其它成员可以访问您的更改，您则可以将文件重新提交到文档库。非常重要的是，您需要详细描述所做的更改，以便您和其他人能充分了解文件中更改的范围。在处理冲突时，这些备注可能会很有用。

在将文件重新提交到文档库时，Perforce 会将您的更改合并到文档库中已有的文件中。如果存在任何冲突，则必须先解决冲突，然后文件才能成功地提交到文档库。

将您的更改提交到文档库的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 点击 Submit Changes 按钮，然后点击以下其中某个选项：

- Work Units —— 仅发送工作单元到服务器进行登入。
- Sources —— 仅发送源文件到服务器进行登入。
- All —— 发送所有未完成的文件（工作单元和源文件）到服务器进行登入。

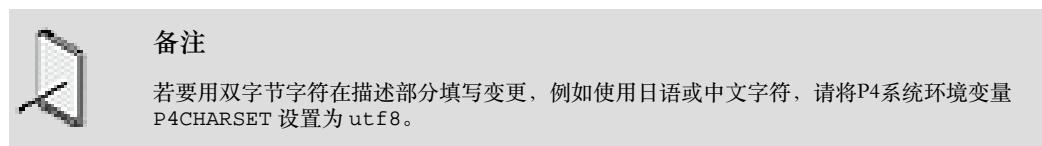
此时将打开 Submit Changes 对话框。

3. 如果您不想将特定文件 check-in 到服务器，则可取消选择这些文件，然后点击 OK。

4. 右键点击选定的工程文件之一，然后从菜单中选择 Submit Changes。

此时将打开 Description 对话框。

5. 输入您对文件所做的更改的详细描述。



6. 如果您想在提交更改后立即继续编辑文件，请选择 Check out file(s) after submit 选项。
7. 点击 OK。

此时 Process Log 对话框将打开，显示文件已经登入或“提交”。

8. 点击 OK 以关闭 Process Log。

文件的状态设置为“normal”。

9. 点击 Close 以关闭 File Manager。

在 Project Explorer 中，除非您选择了 Check out file(s) after submit 选项，否则工程文件将带有 Checked in 叠加图标。

将单个文件提交到文档库的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 切换到以下某个选项卡：

- Work Units，将工作单元添加到 Perforce
- Sources，将音频文件添加到 Perforce。

3. 选择您要提交的文件。

4. 右键点击选定的工程文件之一，然后从菜单中选择 Submit Changes。

此时将打开 Description 对话框。

5. 输入您对文件所做的更改的详细描述。



备注

若要用双字节字符在描述部分填写变更，例如使用日语或中文字符，请将P4系统环境变量 P4CHARSET 设置为 utf8。

6. 如果您想在提交更改后立即继续编辑文件，请选择 **Check out file(s) after submit** 选项。
7. 点击 **OK**。

此时 Process Log 对话框将打开，显示文件已经登入或“提交”。

8. 点击 **OK** 以关闭 Process Log。

文件的状态设置为“normal”。

9. 点击 **Close** 以关闭 File Manager。

在 Project Explorer 中，除非您选择了 **Check out file(s) after submit** 选项，否则现在工程文件将带有 Checked in 叠加图标。



技巧

您也可以通过在 Project Explorer 中右键点击工作单元，然后选择 **Workgroup > Submit Changes** 来将更改提交到 Perforce 文档库中。

相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

将您的文件与 Perforce 文档库中的文件进行比较

您可以将客户端工作区中的文件与文档库中文件的最新版本进行比较，以确定两个文件之间是否存在差异。在比较文件之前应该保存工程，以便您使用文件的最新版本进行比较。

将您的文件与文档库的文件进行比较的方法是：

1. 在菜单栏中，点击 **Project > File Manager...**。

File Manager 将打开。

2. 右键点击您要比较的工作单元文件，然后从菜单中选择 **Diff**。

默认的 Perforce 差异查看器 (diff viewer) 中显示两个文件之间的差异。

3. 查看两个文件之间的差异。
4. 查看完差异后，关闭 Perforce 以返回 Wwise。

相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

使用 Perforce 在 Originals 文件夹中移动源文件

您可以在工程的 Originals 文件夹中移动源文件，但在这样做之前，应确保其它团队成员没有在使用这些文件。如果您移动一个已被其他人 check-out 的文件，则那些人在试图将更改提交到文档库中去时将遇到问题。

当 Perforce 移动文件时，它将打开当前文件以待删除，并在新位置中创建该文件的一个完全副本。这些文件不会在文档库中自动更新，因此在移动结束后，必须将更改列表提交到文档库中。在移动文件前，必须 check-in 文档库。



在 Originals 目录内的文件夹之间移动文件的方法如下：

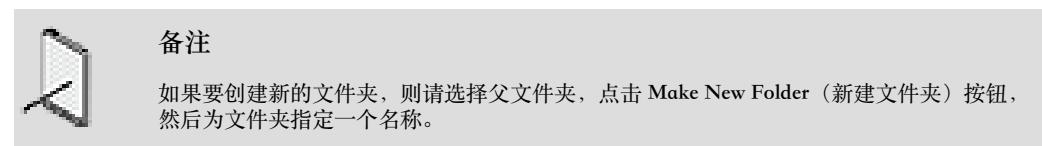
1. 保存当前工程。
2. 执行以下操作之一：
 - 在菜单栏中，点击 Project > File Manager...。
 - 按 Shift+F1。

File Manager 将打开。

3. 切换至 Source Files 选项卡。
4. 选择要移动的文件。
5. 右键点击选定项，然后从菜单中选择 Move（移动）。

此时将会打开 Browse For Folder（浏览文件夹）对话框。

6. 浏览文件夹结构并选择要移动其中文件的文件夹。



7. 打开 OK 以将文件移动到新文件夹。

此时 Process Log 对话框打开，显示文件的新副本通过分支命令已经添加到指定文件夹，原始文件已“打开以待删除”。

8. 点击 OK 以关闭 Process Log 对话框。

9. 选择列表中的两个条目：分支和删除。

10. 右键点击选定项，并从菜单中选择 Submit Change。

此时将打开 Description 对话框。

11. 输入有用的描述，然后点击 OK。

此时 Process Log 将打开，显示两个待提交的操作已提交到文档库。

12. 点击 OK 以关闭 Process Log。

文件于是放置于指定文件夹中了。

相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

使用 Perforce 时重命名文件

您可以对工程中的文件进行重命名，但在此之前，应确保其它团队成员没有在使用该文件。如果您对已被其他人 check-out 的文件进行重命名，他们在尝试将更改提交到文档库时则会遇到问题。

当 Perforce 对文件重命名时，它将打开当前文件以待删除，并使用新名称创建该文件的完全副本。这些文件不会自动在文档库中更新，因此重命名结束后，必须将更改列表提交到文档库中。在重命名文件前，必须先 check-in 到文档库中。



注意

勿对默认工作单元进行重命名，因为它们对工程至关重要。

对工程中的文件进行重命名的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 右键点击您要重命名的工作单元文件，然后从菜单中选择 Rename。

此时将打开 Rename File 对话框。

3. 在 Name 字段，为文件输入新名称。
4. 点击 OK。

此时 Process Log 对话框打开，显示已经使用分支命令创建有了新名称的文件副本，并且旧文件已打开以待删除。

5. 点击 OK 以关闭 Process Log。
6. 选择新旧文件。
7. 右键点击并从菜单中选择 Submit Changes。

此时将打开 Description 对话框。

8. 输入有关更改的有用描述，并点击 OK。

Process Log 打开，其中显示原始文件已删除，并添加了一个状态设为“Branch”的新文件。

9. 点击 OK 以关闭 Process Log。
10. 点击 Close 以关闭 File Manager。

此时 Project 对话框打开，提示您重新加载工程的最新版本，因为外部对工程进行了更改。

11. 点击 Yes。

工程的最新版本于是加载进来了，并且文件也被重命名了。

相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时从工程中删除文件](#)
- [在使用 Perforce 时保存工程](#)

使用 Perforce 时从工程中删除文件

当您不再需要工程中的某些文件时，可以删除它们。在 Perforce 中删除文件时，必须先打开文件以待删除，然后将更改列表提交到文档库。在删除文件前，应始终先保存工程，防止丢失您自上次保存以来所做的更改。



注意

切勿删除默认工作单元，因为它们对工程至关重要。

从工程中删除文件的方法是：

1. 在菜单栏中，点击 Project > File Manager...。

File Manager 将打开。

2. 选择您要删除的文件。
3. 右键点击选定的工程文件之一，然后从菜单中选择 **Mark for Delete**。

此时将显示一条消息，提示您确认删除文件。

4. 点击 **Yes**。

此时 Process Log 对话框打开，显示文件已经打开以待删除。

5. 点击 **OK** 以关闭 Process Log。

文件的状态设置为“delete”。

6. 点击 **Close** 以关闭 File Manager。

此时 Project 对话框打开，提示您重新加载工程的最新版本，因为外部对工程进行了更改。

7. 点击 **Yes**。

工程的最新版本于是加载进来了，并且文件也被删除了。

此时，文件只是被添加到了更改列表中，并且已被标记为待删除。您还需要提交更改列表才能从文档库中删除文件。

相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [在使用 Perforce 时保存工程](#)

在使用 Perforce 时保存工程

只要您 check-out 了打算更改的所有工程文件，保存您的工程就不会有任何问题。然而，如果您更改没有 check-out 的文件，Wwise 则将提示您 check-out 这些文件，然后它才能保存整个工程。

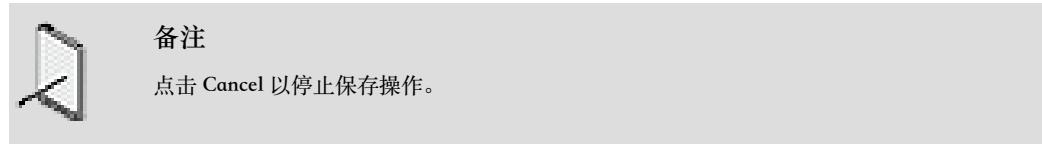
保存工程的方法是：

1. 执行以下操作之一：
 - 在菜单栏中，点击 **Project > Save**。
 - 按 **Ctrl+S**。

Wwise 试图保存工程中的所有文件。如果您更改没有 check-out 的文件，则将打开 Pending Source Control Operations 对话框。

2. 选择 **Check out modified files** 选项以检出已经更改的文件，以便 Wwise 保存它们。
3. 点击 **OK**。

Wwise check-out 必要的文件并保存工程。



相关主题

- [将文件添加到 Perforce](#)
- [从 Perforce 文档库中 check-out 文件进行编辑](#)
- [将您的更改提交到 Perforce 文档库](#)
- [将您的文件与 Perforce 文档库中的文件进行比较](#)
- [使用 Perforce 在 Originals 文件夹中移动源文件](#)
- [使用 Perforce 时重命名文件](#)
- [使用 Perforce 时从工程中删除文件](#)

Workgroup 技巧与经验总结

在结合 Wwise 使用版本控制系统之前，您可能需要阅读以下各节，它们为您提供一系列的技巧和最佳实践，可帮助您在整个音频开发过程中更好地管理团队和工程文件。

规划您的工程

- 将工程划分成小型工作单元。如果工程很庞大，并且您还将所有工程数据都保留在默认工作单元中的话，那么不仅会延长 Wwise 的响应时间，而且每当团队中有人做出更改后，其他团队成员则都需要进行合并，这可能会令发生合并问题和冲突的概率激增。通过使用工作单元将工程划分成小块，可以提高人们的工作效率，加快访问信息的速度，处理不同的领域，从而降低发生合并问题的可能性。
- 为全局默认工作单元指定负责人。某些工程元素（例如 Master-Mixer 层级结构和预置）无法进一步细分成更多的工作单元。因此好办法是让一个人管理或者至少其他人清楚对这些工程元素所做的更改。

基本工作流程

- 高效地管理全局工程元素。当您重命名或删除某个全局工程元素（例如状态或游戏参数）时，注意，您可能修改了工程中的许多其他对象，包括使用这些元素的所有声音对象和容器。当保存和登入这些更改时，您可能会影响其他人正在处理的众多工程文件。为限制此类更改的影响，应该尽早定义全局工程元素，此后再尽力避免更改这些元素。如果在初始设置后需要更改，则应执行以下操作：
 - 警告团队成员全局元素已被更改。
 - 要求团队全体成员登入他们的更改。
 - check-in 您的文件
 - 要求团队全体成员更新他们的工程文件。

- 通过执行此流程，团队成员只需更新就可以获得新文件，无需合并。
- 检查工程文件状态。在开始处理某个工作单元前，使用 File Manager 验证哪些文件是只读文件。如果您更改的是只读工作单元，则将无法把该特定文件保存到工程中。
- 定期备份本地文件。虽然中央资源库中的文件可能是预定备份计划的一部分，但本地机器上的副本不是。为防止数据丢失，定期备份工程文件是个好办法，特别是当您对文件做了大量更改时，尤其如此。
- 在登入前生成完好度报告。在登入特定工作单元前，生成完好度报告是确保没有任何工程错误的好办法。如果存在错误，您可以快速解决它们，然后登入工作单元。

同步您的文件

- 在启动新的工作会话前同步。您应该先将您的工程文件与服务器进行同步，然后再启动新的工作会话，以便您获得最新的修改。
- 同步前关闭 Wwise。在将工程文件与服务器机型同步之前，应关闭 Wwise 以防止丢失信息的可能性。如果 Wwise 一直是打开状态，则副本将保留在内存中。同步时磁盘中的文件将被修改，但工程的旧版本仍将保留在内存中。如果您保存当前已打开的工程，则会覆盖磁盘中已保存的其他人的更改。

这仅适用于您未使用工作组插件的情况。当您使用工作组插件进行同步时，将自动提示您重新加载工程的最新版本。

check-in 您的文件

- 经常提交。如果您有大更改，影响到团队的其他成员，应该经常将工程文件提交到服务器，以使其他成员可以使用您所做的更改。如果您等待太久，则还会增加发生冲突的概率。通过提交较小的、有针对性的更改，在必要时就容易恢复到工程的旧版本，并且容易在出现冲突时加以解决。
- 添加有用的注释。在提交或登入文件时，确保充分描述您所做的更改。

版本控制系统

- 了解您的版本控制系统。在使用版本控制系统管理 Wwise 工程文件之前，应充分了解它的工作原理。了解版本控制系统的复杂性可帮助您避免问题，构建高效的制作管线。

工程不一致性

- 熟悉 Wwise 工程 XML 结构。在合并文件前，花一点时间熟悉 Wwise 工程文件的 XML 结构。在一些情况下，您可能必须更新 XML 代码。如果您没有正确理解它，可能会毁坏您的工程。如果您的确要修改 XML，请确保在 Wwise 中打开工程，然后将文件重新登入到版本控制系统中。这将确保您对 XML 的更改是有效的，并且是您所需的。

Communication

- 鼓励经常和坦诚的沟通。在任何工作组环境中，成功的关键是经常与团队其他成员进行坦诚沟通。经常、坦诚的沟通可以减少冲突，缩短合并文件的时间，建立一个更加高效的制作管线。例如，在对工作单元执行会影响到团队其他成员的更改前，应叫他们登入他们的更改，然后等到您的指示后再同步并继续他们的工作。

文件使用

- 对于 File Manager 内 Usage 列中示为 Unused 的文件，最好的做法是在删除它们之前先关闭并重新打开工程，确保信息完全更新。

SoundBanks

- 为避免每次更改工程时必须编辑 SoundBank，您可以使用工作单元和/或文件夹按照您设置 SoundBank 的方式重新创建。由于 Wwise 在 SoundBank 元素与工程元素之间保持着有效的链接，因此这些 Work Units（工作单元）一旦添加到 SoundBank，您就无需再编辑这些 SoundBank 了，它们会自动更新。

第 7 章 管理工程中的媒体文件

概述	126
导入的过程	126
导入媒体文件	133
替换媒体文件	149
管理文件导入问题	151
重新组织 Originals Folder 中的媒体文件。	153
清除缓存	154
在外部编辑器中编辑音频文件	155
使用插件创建 Source	156
媒体文件管理技巧和经验总结	157

概述

在一个典型的游戏里，您可能拥有成千上万的素材，了解如何有效管理工程中这些素材至关重要。通常情况下，Wwise 会将素材分为两类：

- 导入到 Wwise 中的原始音频素材。
- 为各种游戏平台创建的这些素材的各个版本。

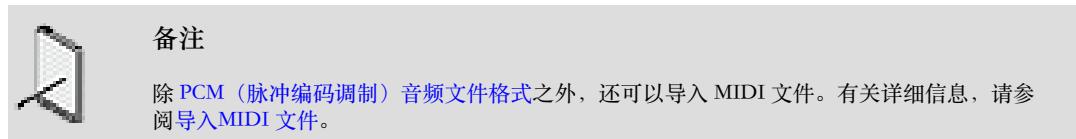
Wwise 会将这两类素材存储在您工程文件夹内的不同位置，以便独立管理它们。原始音频素材存储在Originals（原始音频）工程文件夹中。由于这些素材通常由团队中的多个成员所共享，因此该文件夹可位于您网络中的任何位置并通过版本控制系统轻松进行管理。

其他版本的素材存储在每个用户本地工程的缓存（cache）文件夹中。这样一来，各个用户都能够管理各自的平台版本，并且能够试用不同的转码设置。

这两类文件都可在 Wwise 中播放。每当在 Transport Control（播放控制）或 Soundcaster（声音选角器）中激活 Original（原始音频）这个控件时，Wwise 都将播放未转码的原始音频文件。在 Original 控件没有激活的情况下，如果存在转码了的文件，则 Wwise 将尝试播放该文件。不过请务必要注意，某些格式的转码文件在 Wwise 中不一定能播放。在针对特定平台对某个音频文件做转码时，转码的目的是要满足该平台的特定硬件要求。因此，如果 Wwise 设计工具的运行平台不支持该文件类型，可能会无法播放这些转码文件。

导入的过程

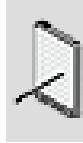
在将文件导入 Wwise 工程之前，您首先应了解 Wwise 支持的文件类型，以及您希望将这些媒体文件存储在什么位置。



支持哪些媒体文件？

要想了解 Wwise 支持哪些类型的媒体文件的话，您需要了解以下内容：

- [PCM（脉冲编码调制）音频文件格式](#)
- [Channel Configuration](#)
- [采样率](#)
- [Bit depth](#)



备注

尝试导入 Wwise 不支持的文件将会导致在 Import Conflict Manager（导入冲突管理器）中显示不可恢复的错误。有关如何处理不可恢复的错误的信息，请参阅[管理不可恢复的错误](#)。

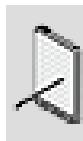
PCM（脉冲编码调制）音频文件格式

波形音频格式（WAV）文件和 Ambisonics 音频格式（AMB）文件可导入到 Wwise 中。通过读取文件头，Wwise 可以应用相应的声道配置。

Channel Configuration

Wwise 可导入具有多声道的媒体文件。

下表列出了 Wwise 支持的所有源声道配置以及 Wwise 使用的默认配置。



备注

Wwise 遵从 Microsoft 的声道顺序标准。导入的媒体文件中的声道顺序与 Wwise 中显现的顺序不同。下表显示的是媒体 Wwise 中的顺序。

声道名称——简称和全称	建议的角度
L 左	22° -30°
R 右	22° -30°
C 中置	0°
SL 左环绕	90° -110°
SR 右环绕	90° -110°
BL 左后	135° -150°
BR 右后	135° -150°
LFE Low-Frequency Effects（低频效果）	不适用

声道	Wwise 使用的声道配置
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
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

声道	Wwise 使用的声道配置
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

如果您希望 Wwise 在导入文件时保留某个特定的多声道配置，则必须在 WAV 文件头的 WAVEFORMATEXTENSIBLE（可扩展波形文件格式）定义结构格式块中定义它。如果 WAVEFORMATEXTENSIBLE 中的声道掩码所定义的声道在 Wwise 中不支持，则这些声道将会被自动更改为支持的值。例如，如果某个配置定义了左声道但没有定义右声道，则左声道（0x01）会更改为中置声道（0x04）。如果某配置定义了侧边（环绕）声道，但没有定义后置（后）声道，则侧边声道（0x100 和 0x200）将更改为后置声道（0x10 和 0x20）。

您可以使用 Wwise Multi-Channel Creator（Wwise 多声道创建工具）应用程序，将单个单声道音频文件和立体声音频文件合并为一个多声道文件。此独立工具是 Wwise 程序包安装内容的一部分并可在 Wwise 应用程序所在的目录下找到。您可以利用 Multi-Channel Creator 创建包含任何多声道配置的文件，以便能将这些配置导入到 Wwise 工程中。您还可以使用此工具将特定单声道文件标记为 0.1 组件，这样便可将这些文件作为独立的LFE（低频效果）导入到 Wwise 中。

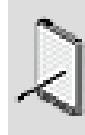
采样率

Wwise 可导入采样率高达 96 kHz（含 96 kHz）的媒体文件。

Bit depth

Wwise 可导入具有以下位深的媒体文件：

- 16 位
- 24 位



备注

Wwise 还可以导入 32 位浮点式 PCM（脉冲编码调制）文件；但是，转码格式最高仅支持 24 位。此外，在 Transport Control 中播放时，文件将转换为 16 位。

在导入时创建 Wwise 对象

在验证了 Wwise 确实支持要导入的文件后，您可以选定要为文件创建何种类型的对象。

Wwise 会区别对待 Sound SFX、Sound Voice，和 Music Track 对象。由于语音对象可以翻译为不同的语言，因此在 Wwise 中会以不同方式进行处理。此外，还存在其他

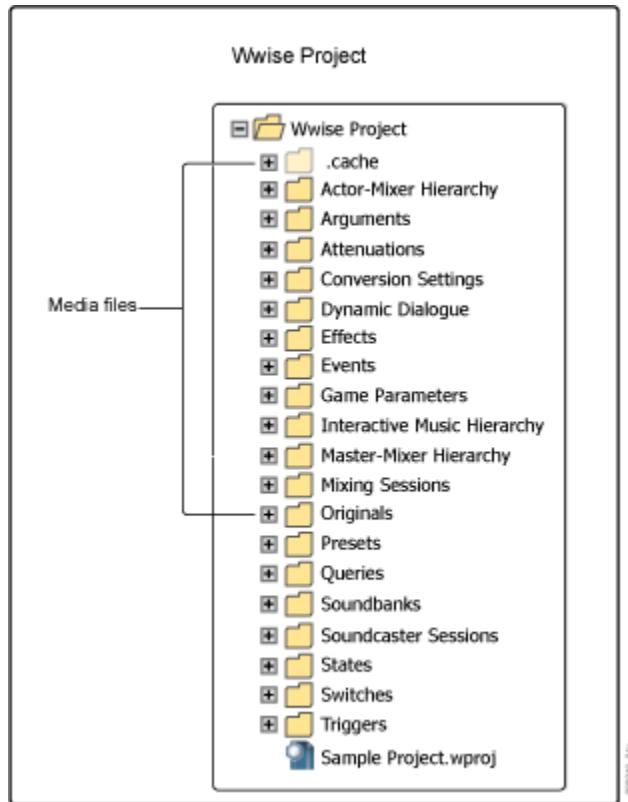
对象结构，例如 Container 和 Music Segment，可以用来对基础对象进行分组和组织。为了帮助区分这些对象，Wwise 使用不同的图标来表示每个对象类型。

图标	代表
	SFX —— 音效。为媒体文件导入 Actor-Mixer 时的默认对象类型。
	Voice —— 语音。为游戏中的对白而创建的对象，这些对白可能会翻译为多种语言的版本。 技巧 为 Reference Language 导入对象时，按住 Ctrl 和 Shift 键的同时将 WAV 文件拖动到 Actor-Mixer 中，则 Import As: (导入类型) 将自动设置为 Sound Voice，而不是默认的 Sound SFX。
	Random Container —— 随机容器。默认情况下，在将文件夹拖动到 Actor-Mixer Hierarchy 中时，Wwise 会为该文件夹创建 Random Container 对象。但是在弹出的 Audio File Importer 窗口中，可以在 Object Type/Action (对象类型/动作) 下拉菜单中为文件选择其他选项： <ul style="list-style-type: none">Object (对象)Virtual Folder (虚拟文件夹)Actor-Mixer (角色混音器)Sequence Container (序列容器)Switch Container (切换容器)Blend Container (混合容器)Sound SFX (音效声)
	Music Track —— 在 Interactive Music Hierarchy 中，将源文件 (WAV 或 AMB) 拖动到 Music Segment 时将会创建。 技巧 在 Project Explorer 的 Interactive Music Hierarchy 内，将 WAV 文件拖动到 Music Segment 中时同时按住 Ctrl 键，可以自动创建 Music Track 对象。
	Music Segment —— 将文件夹或源文件 (WAV 或 AMB) 拖动到 Interactive Music Hierarchy 时，将默认创建。 但在弹出的 Audio File Importer 窗口中，可以在 Object Type/Action 下拉菜单中为文件选择其他选项： <ul style="list-style-type: none">Object (对象)Virtual Folder (虚拟文件夹)Music Switch Container (音乐切换容器)Music Playlist Container (音乐播放列表容器) 备注 在将 WAV 文件拖动到现有的 Music Segment 上去时，Wwise 会为拖上来的各个 WAV 文件创建音乐轨。
	技巧 在将 WAV 文件拖动到 Project Explorer 的 Interactive Music 层级结构中的同时按住 Ctrl 键，可自动创建 Music Segment 对象。

媒体文件结构

了解媒体文件在 Wwise 工程中的存储方式是大有帮助的，特别是您在以团队合作的方式工作时更是如此。媒体文件存储在两个单独的文件夹中：

- Originals（原始音频）文件夹
- .cache（缓存）文件夹

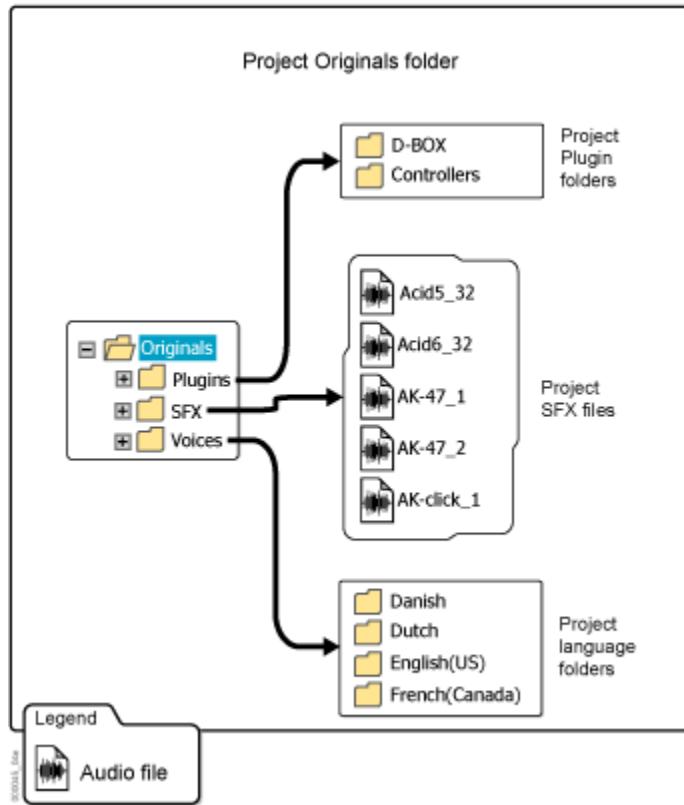


Originals（原始音频）文件夹

在将媒体文件导入到 Wwise 中时，这些媒体文件会复制到 Originals 工程文件夹中，并保持未转码和未更改的状态。通常，应该对此工程文件夹做版本控制，让协同处理该工程的人员都能访问它的存储位置。Originals 文件夹的位置是在工程创建时指定的，但您可以在 Project Settings（工程设置）对话框中更改此位置。当您在 Transport Control（播放控制）或 Soundcaster（声音选角器）中选择 Original 时，将会播放这些“原始音频”文件。

Originals 文件夹具有以下几个子文件夹：

- Plugins（插件）
- SFX（音效）
- Voices（语音）



所有导入到 Wwise 中的媒体文件都会按类型进行分类，以便 Wwise 能够有效管理这些文件。

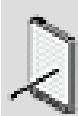
随着工程规模的增长，您可能会需要将这些文件夹中的文件组织到子文件夹中去。如果是这样的话，则在导入文件之前，您还需要注意以下限制：

- 组织到子文件夹中的媒体文件需要从相应的子文件夹内导入 Wwise。这使得 Wwise 可管理各个媒体文件的位置，方法是使用其完整路径和名称来保存媒体文件。
- 文件路径必须不能重复。也就是说，各个文件夹中的所有媒体文件名不得重名。但这并不妨碍两个媒体文件具有相同的文件名，只要这两个文件位于不同的文件夹中即可。

如果已经将媒体文件导入到工程中，那么您还可以在各文件夹之间移动源媒体文件。有关移动源媒体文件的详细信息，请参阅[重新组织 Originals Folder 中的媒体文件](#)。

.cache (缓存) 文件夹

默认情况下，这个缓存文件夹存储在 Wwise 工程中并且包含您为各个平台开发游戏时采用的转码后的媒体文件。当您在 Transport Control (播放控制) 或 Soundcaster (声音选角器) 中未选择 Original (原始音频) 按钮时，将会播放这些转码结果文件。



备注

您可以 Project Setting 对话框中修改缓存文件夹的位置。有关修改工程缓存文件夹位置的详细信息，请参阅[定义缓存文件夹设置](#)。

不过请务必要注意，某些格式的转码文件在 Wwise 中不一定能播放。在针对特定平台对某个音频文件做转码时，转码的目的是要满足该平台的特定硬件要求。因此，如果 Wwise 设计工具的运行平台不支持该文件类型，可能会无法播放这些转码文件。

要在 Wwise 中测试转码结果文件，您可以执行以下操作：

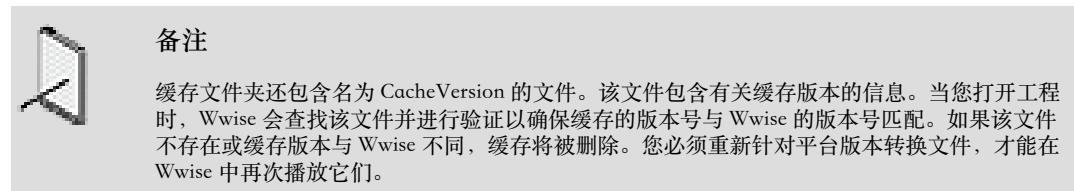
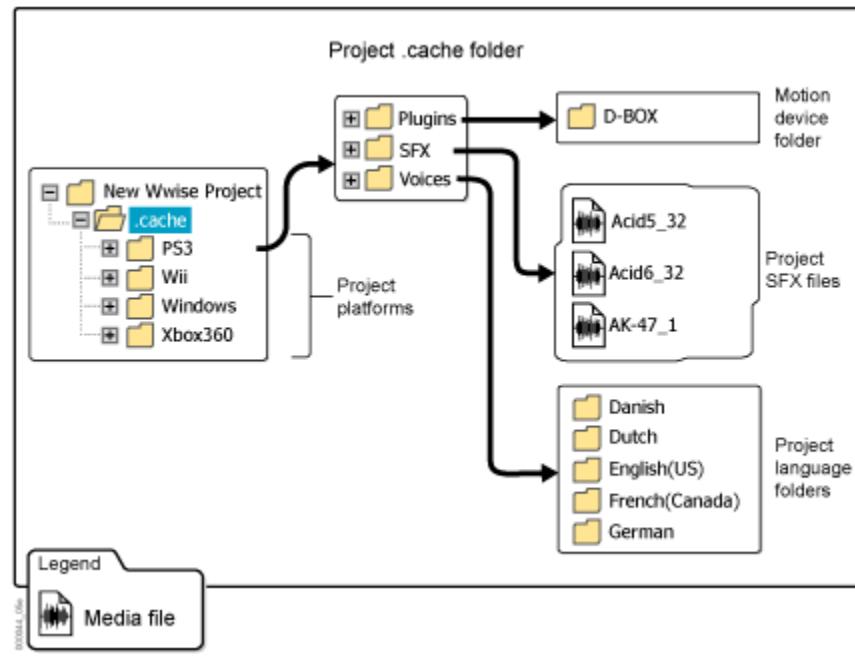
- PCM —— 选择 Windows 平台。
- ADPCM -- 自适应差分脉冲编码调制。选择 Windows 平台。
- Vorbis —— 选择 Windows 平台。



各个平台文件夹还可进一步包含以下子文件夹：

- Plugins (插件)
- SFX (音效)
- Voices (语音)

Wwise 为您管理这些文件夹的内容，因此您将不必再直接在这些文件夹中进行任何更改。



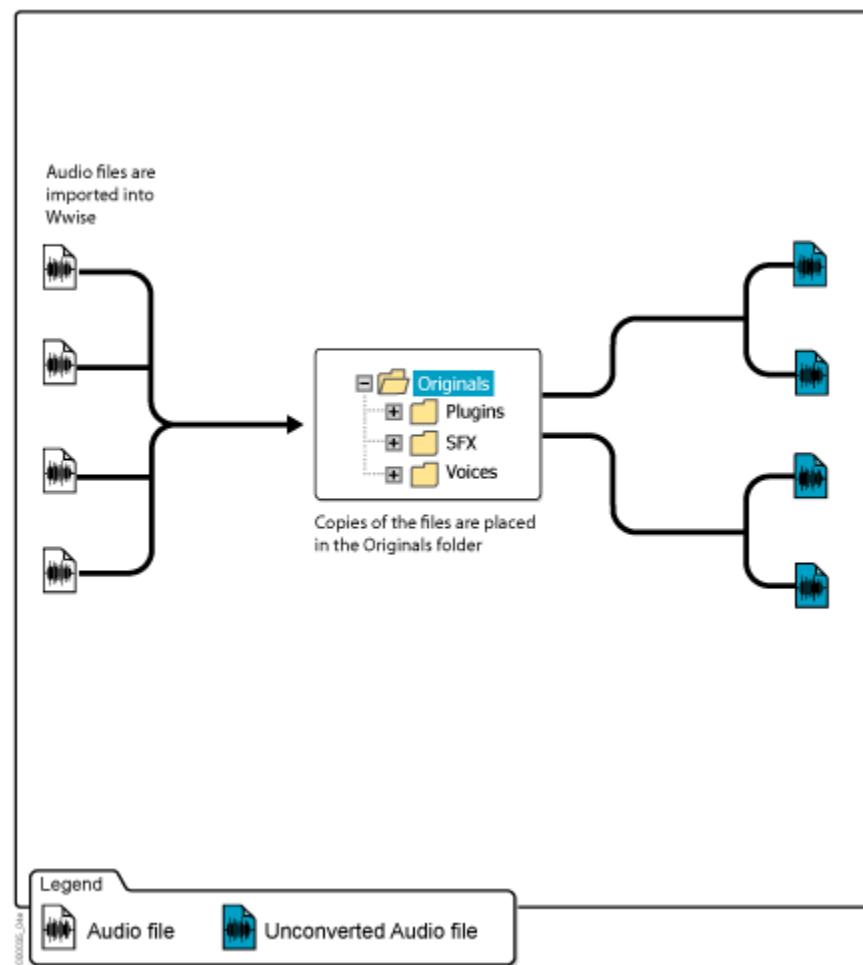
在导入过程期间会发生什么？

在导入过程期间，Wwise 会执行几个活动。对这些媒体文件分别执行转码操作。有关针对各平台对媒体文件做转码的详细信息，请参阅[对音频文件做转码](#)。

导入过程包括以下操作：

- 在将原始媒体文件复制到工程的 Originals 文件夹之前，Wwise 会对其进行验证。
- 为媒体文件创建音频源。
- 包含版本的声音对象或音乐对象会在 Wwise 中进行创建并在 Project Explorer 的 Audio 选项卡中以各自层级结构进行显示。

下图说明了在您将媒体文件导入到 Wwise 工程时会发生什么。



导入媒体文件

出于各种原因，您随时都会需要将音效或语音媒体文件导入到 Wwise，具体取决于您打算如何使用这些媒体文件。通常，您会在以下情况下导入文件：

- 在工程开始时或有文件要用时将媒体文件导入工程。

- 替换您之前导入的媒体文件。例如，您可能会需要替换工程之初使用的占位符文件或临时文件。
- 将多语言媒体文件导入工程以进行本地化。有关本地化的详细信息，请参阅[对工程进行本地化](#)。

在 Wwise 中，您可使用 **Audio File Importer**（音频文件导入器）将媒体文件导入工程，或者您可以使用快速导入快捷方式执行音频文件导入。

快速导入

下表列出了您可用于将媒体文件导入 Wwise 中的快捷方式。

操作	使用此快捷方式
导入音效媒体文件。	将文件拖动到 Wwise 中。
导入语音媒体文件。	按 Shift 并将文件拖动到 Wwise 中。
在不打开 Audio File Importer 的情况下导入音效媒体文件。	按 Ctrl 并将文件拖动到 Wwise 中。
在不打开 Audio File Importer 的情况下导入 Sound Voice 媒体文件。	同时按 Ctrl 和 Shift 并将文件拖动到 Wwise 中。 如果您使用此快捷方式导入语言，则请确保已在 Language Selector（语言选择器）中选择了引用语言。
将媒体文件作为新源导入 Contents Editor（内容编辑器）。	将文件拖动到 Contents Editor 中。



技巧

如果您使用电子表格程序来管理工程中成千上万的语音素材，则可以在 Wwise 中使用 **Voice Asset Importer**（语音素材导入器）快速创建相应的 Sound Voice 对象。有关将语音素材导入 Wwise 的详细信息，请参阅[导入文本文件中的语音素材](#)。

了解 Wwise 中的文件长度限制

您可以将文件名长度为 256 个字符的文件导入 Wwise。此文件长度限制实际上是 Windows 的限制并且与文件名的整条路径有关。例如，在目录根处的文件名最多可有 252 个字符。如果文件位于某文件夹中，则该文件夹名称的长度也需要考虑在内。在 Wwise 中导入媒体文件或对媒体文件做转码时，此限制可能会导致问题。

如果您尝试导入接近或达到最大限制的媒体文件，则可能会遇到某些导入问题。如果文件名长度太长，则将会显示以下错误之一：

- Not a valid WAV file.（不是有效的 WAV 文件。）
- Error copying the file to the originals folder.（将文件复制到 Originals 文件夹时出错。）

在某些情况下，Wwise 可以导入接近或达到最大限制的文件，但您可能会在文件转码时遇到问题。在此情况下，将会显示以下错误消息：

- Can't open source or output file.（无法打开源文件或输出文件。）

要解决这些问题，您需要缩短文件名的长度或缩短文件名路径的整个长度。

导入用于音效的媒体文件

在 Audio File Importer 中，您可以指定您希望将媒体文件作为音效对象导入。在导入这些文件时，音效对象将在 Project Explorer 的 Audio 选项卡中的选定位置加以创建。根据您导入的层级结构，这些对象可以是声音对象或音乐对象。

将音效媒体文件导入工程的方法如下：

1. 执行以下操作之一：

- 从 Wwise 菜单栏中，点击 Project > Import Audio Files。
- 右键点击 Actor-Mixer Hierarchy 或 Interactive Music Hierarchy（互动音乐层级结构）中的对象，然后从菜单中选择 Import Audio Files。

Audio File Importer 将打开。

2. 点击 Add Files（添加文件）或 Add Folders（添加文件夹）。

此时将会打开 Open File（打开文件）对话框。

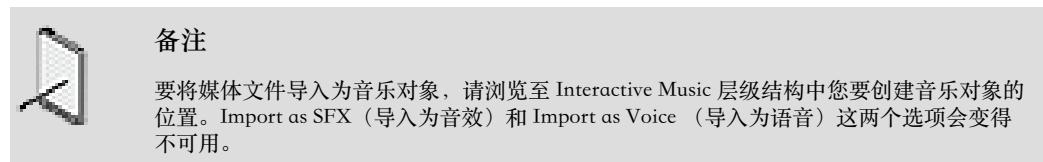
3. 浏览至您要导入的媒体文件的位置。

4. 选择文件或文件夹，然后点击 Open（打开）或 Select Folder（选择文件夹）。

所选文件或文件夹会加载到 Audio File Importer 中。

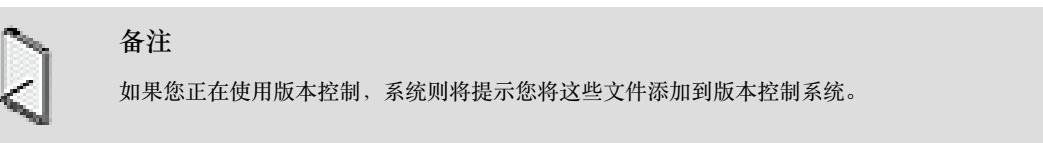
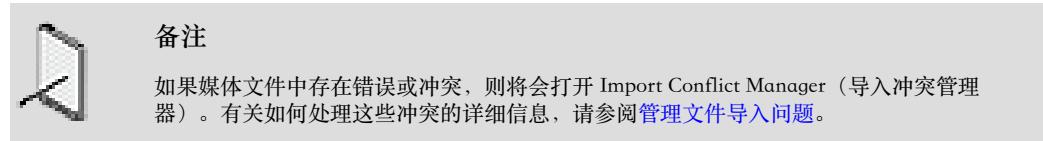
5. 要将媒体文件导入为声音对象，请选择 Import as Sound SFX（导入为音效声）选项。

6. 要更改将在 Wwise 中创建对象的位置，请点击 [...]，然后在层级结构中选择新的位置。



7. 点击 Import（导入）

此时将会打开 Importing（导入）对话框，您可以在其中查看导入过程的进度。



相关主题

- [导入用于旁白的媒体文件](#)
- [了解 Wwise 中的文件长度限制](#)

- 快速导入
- 替换媒体文件
- 管理文件导入问题
- 重新组织 Originals Folder 中的媒体文件。
- 清除缓存
- 在外部编辑器中编辑音频文件

导入包含媒体文件的文件夹

您可以直接将若干个包含媒体文件的文件夹结构导入工程中。导入文件夹结构时：

- Wwise 可为导入结构中的各个文件夹创建容器对象（可选）。
- Wwise 在将媒体文件复制到 Originals 文件夹时会复制文件夹结构。

将文件夹结构导入工程的方法如下：

1. 执行以下操作之一：

- 从 Wwise 菜单栏中，点击 Project > Import Audio Files。
- 右键点击 Project Explorer 层级结构，然后从菜单中选择 Import Audio Files。

Audio File Importer 将打开。

2. 点击 Add Folders。

此时将会打开 Select Folder 对话框。

3. 浏览至您要导入的媒体文件的位置。
4. 选择文件夹，然后点击 Select Folder。

所选文件夹将会加载到 Audio File Importer。

5. 在 Object Type/Action（对象类型/动作）列中，选择要为各个正在导入的文件夹创建对象类型。



技巧

您可以在 Audio File Importer 中选择多个文件夹条目，然后选择 Object Type/Action 以快速同时更改这些条目。

6. 点击 Import（导入）

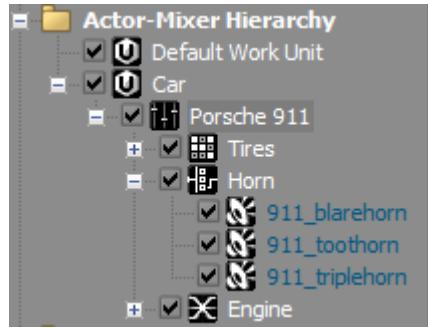
此时将会打开 Importing（导入）对话框，您可以在其中查看导入过程的进度。

使用模板导入媒体文件

当从文件夹中导入媒体文件时，您可以选择 Audio File Importer 将要使用的工程中的对象，为您正在导入的文件夹和媒体文件创建新的 Wwise 对象。在创建多个仅有媒体文件不同的类似结构时，这可能会非常有用。

典型示例是，创建多个车辆结构；它们都类似，但具有不同的媒体文件并且与特定车辆品牌或型号有关。以下是相应的工作流程：

1. 在工程中创建第一个车辆结构，然后将其作为模板来导入其它结构。此结构可包含任何容器和声音对象。



如果您对此结构满意，那么可将此车辆结构作为下一个车辆结构的模板。

2. 在计算机中，创建与您刚刚在 Wwise 中创建的车辆结构相对应的文件夹结构，以保存您的媒体文件。



现在，您有了一个可轻松导入的结构。

3. 从 Project 或快捷菜单中点击 Import Audio Files...（导入音频文件...）。默认快捷键为：Shift+I。

Audio File Importer 将打开。

4. 点击 Add Folders...（添加文件夹...）。

此时将会打开 Explorer（资源管理器）对话框。

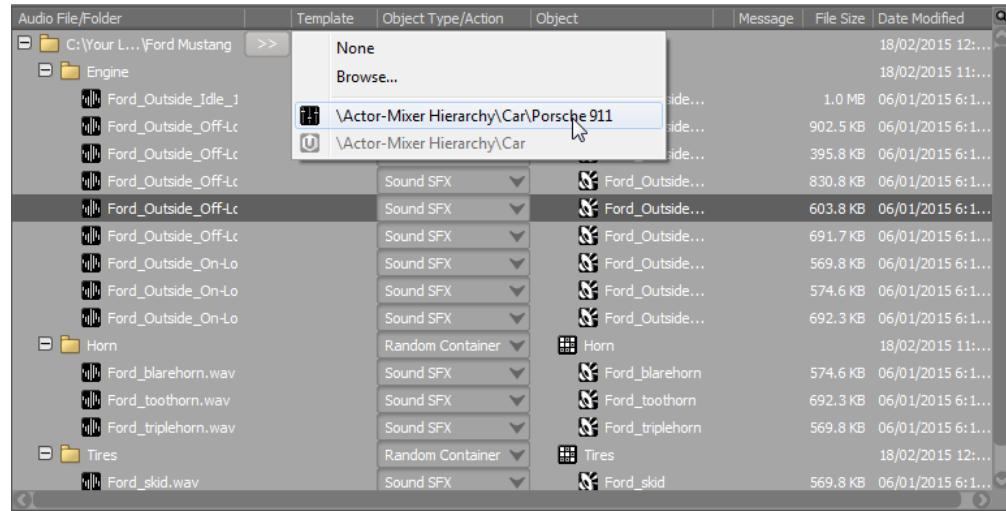
5. 前往您要导入的该文件夹并点击该文件夹（您刚刚创建的车辆结构的父文件夹）。

所选文件夹的名称将会显示在 Folder（文件夹）字段中。

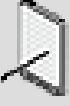
6. 点击 Select Folder（选择文件夹）。

返回 Audio File Importer，本地文件夹层级结构现在显示在 Audio File/Folder 面板中，并且在此结构中也可找到所有媒体文件。

7. 在 Template（模板）列的左侧，点击选择器 [>>] 并浏览找到您在第一步骤中创建的结构。模板即以设置完成。选择器 [>>] 还会列出最近用作模板的对象。



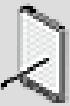
此时，Wwise 将尝试自动将您导入的媒体文件和文件夹与模板中的对象相匹配。匹配的文件夹将指定为与其对应模板相同的容器类型，而默认情况下没有匹配的文件夹则为随机容器。

 **备注**

您可以更改 Wwise 匹配这些导入项的严格程度，方法是从 Audio File Importer 左下方的 Template match mode（模板匹配模式）列表中进行选择。Match all（匹配全部）模式使用特殊算法以在导入文件和模板对象之间找到最匹配项，而 Perfect match only（仅完美匹配）模式会仅在导入文件和文件夹与模板对象的名称完全匹配时才会对它们进行匹配。如果 Wwise 发现导入项与模板不匹配，则 Message（消息）列将会显示"No matching template found"（未找到匹配模板）。

 **技巧**

模板匹配算法是基于名称进行比较的。因此，最佳做法是使文件夹层级结构中的名称与模板层级结构中的名称类似。

 **备注**

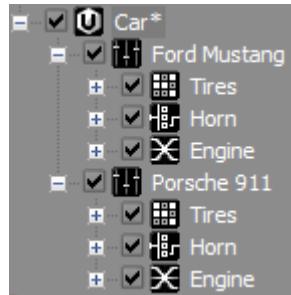
如果模板与导入的结构不匹配，则您可以完全删除模板，方法是点击选择器 [>>] 并选择 None（无）。您还可取消模板关联，方法是在 Object Type/Action 列中选择新的对象类型。

8. 点击 Import（导入）

 **备注**

如果与现有文件存在冲突，则 Import Conflict Manager 将会打开，以用来在完成导入前解决冲突问题。

Audio File Importer 关闭时，新的结构将会出现在 Project Explorer 层级中。



导入用于旁白的媒体文件

在导入用于对白或旁白的媒体文件时，您需要将它们指定为语音对象，以便 Wwise 之后在本地化过程中可以识别出这些对象。

在某些情况下，为了对工程进行本地化，您需要导入语音文件。在导入这些文件时，这些文件会保存在工程语言文件夹中，并且会在 Wwise 中为这些语言创建新源。在您能够执行此操作之前，您需要在 Language Manager 中定义工程的语言。在您设置了工程语言之后，您可以为语言源导入媒体文件。有关使用工程语言的详细信息，请参阅[对工程进行本地化](#)。



将媒体文件作为语音对象导入工程的方法如下：

1. 执行以下操作之一：
 - 从 Wwise 菜单栏中，点击 Project > Import Audio Files。
 - 右键点击 Actor-Mixer Hierarchy 中的 Sound Voice 或工作单元，然后从菜单中选择 Import Audio Files。
2. 点击 Add Files (添加文件) 或 Add Folders (添加文件夹)。

此时将会打开 Open File (打开文件) 对话框。
3. 浏览至您要导入的媒体文件的位置。
4. 选择这些文件，然后点击 Open (打开)。

所选文件于是加载到 Audio File Importer 中了。
5. 选择 Import as Sound Voice 选项。
6. 要更改将要在 Wwise 中创建的对象位置，请点击 [...]，然后在层级结构中选择一个新位置。
7. 点击 Import (导入)

此时将会显示 Importing 对话框，您可在其中查看导入过程的进度。



备注

如果媒体文件中存在错误或冲突，则将会打开 Import Conflict Manager。有关如何处理这些冲突的详细信息，请参阅[管理文件导入问题](#)。

相关主题

- [第 5 章 管理语言](#)
- [导入用于音效的媒体文件](#)
- [了解 Wwise 中的文件长度限制](#)
- [替换 Sound Voice 媒体文件](#)
- [替换音效文件](#)
- [管理文件导入问题](#)
- [快速导入](#)
- [重新组织 Originals Folder 中的媒体文件。](#)
- [清除缓存](#)
- [在外部编辑器中编辑音频文件](#)

从用制表符分割的文本文件导入媒体文件

您可以用制表符分割（也称为用制表符分隔）的文本文件导入大量媒体文件。用制表符分割的文本文件（TXT 或 TSV 扩展名）可通过 Microsoft Excel 或其它电子表格应用程序进行生成。可以导入ANSI、UTF-8 或 UCS-2 编码格式的文件。



导入较大的制表符分割的文件需要花费较长时间。

请注意，导入数以万计的媒体文件可能需要花费大量的时间。在配有 SSD 硬盘的机器上进行大文件的导入，可以减少导入时间。请注意，导入时请确保已退出密集使用磁盘的应用程序，如版本控制同步、云同步，甚至是病毒保护服务（如有可能）。

用制表符分割的文件可定义以下元素：

- 要导入的 WAV、AMB 或 MIDI 文件
- 用于包含文件的对象结构
- 任何属性值或引用，例如声部音量或输出总线
- 要创建的事件

用制表符分割的文件的第一行为后续行定义了列的内容。列的顺序对导入过程不会有影响。

对于制表符分割的文本文件，下表列出了所有可能出现的列标题，并对其进行了说明。



获取对象的 Absolute Path 或 GUID。

在制表符分割的文件中，可以用 Absolute path 和 GUID 明确指定对象。要获得对象的绝对路径或 GUID，请按住 Shift 键并右键点击对象以显示快捷菜单。选择 Copy path(s) to clipboard（将路径复制到剪贴板）或 Copy GUID(s) to clipboard（将 GUID 复制到剪贴板）。

列标题	内容
Audio File	<p>音频文件。指定要导入的 WAV 或 MIDI 文件的路径。文件将复制到 Originals 文件夹中，前提是文件夹中不存在该文件。路径可是绝对路径或相对路径。</p> <ul style="list-style-type: none"> • 绝对路径：包含音频文件名本身的完整路径。 例如 <code>C:\MyWaves\MyFolder\Originals\MySound.wav</code> • 相对路径：相对于制符表分割文件目录的路径。 例如：<code>..\..\MyWaves\MyFolder\Originals\MySound.wav</code> <p>如果创建容器，则此列可以留空。</p>
Audio File	<p>定义要导入的 base64 编码 WAV 音频文件。需要指定目标文件路径并使用竖线（ ）将其与 base64 编码数据分隔。指定的目标文件路径可以在 Originals 下的导入类型文件夹中获取。</p> <p>例如：<code>MySounds/MySound.wav/Uk1GRuRcAABXQVZFZm10IBAA...</code></p>
Object Path	<p>对象路径。定义要创建的对象的路径和名称。路径使用反斜杠，可以是绝对路径或相对路径。</p> <ul style="list-style-type: none"> • 绝对路径：完整路径，以反斜杠和对象类别开头。 例如：<code>\Actor-Mixer Hierarchy\MyWorkUnit\MyVirtualFolder\MySound</code> <div style="background-color: #f0f0f0; padding: 5px;">  备注 <p>如果绝对路径中指定的 Work Unit 不存在，则会导入失败。</p> </div> <ul style="list-style-type: none"> • 相对路径：相对于导入目标的路径，在 Audio File Importer 对话框中进行指定。 例如：<code>MyVirtualFolder\MySound</code> <p>如果指向对象的路径不存在，Wwise 将自动创建所有缺失的父级和对象。可以在路径中指定不同对象的类型，只要在对象名称前的 <> 中填写类型即可。如果未指定类型，导入流程将默认选择 Virtual Folder。支持的对象类型列表与以下描述的 Object Type 列的类型相同。</p> <p>如果此列留空，则对象的名字将从导入音频文件中获取，并且将会导入到 Audio File Importer 对话框中指定的目标。</p> <p>物体路径示例：<code>\<Actor-Mixer>Barrel\<Sound SFX>Minigun_Barrel_Stop</code></p>
Object Type	<p>用于指定对象类型，有下列选项可用：</p> <ul style="list-style-type: none"> • Virtual Folder（虚拟文件夹，适用于 Actor-Mixer Hierarchy 或 Interactive Music Hierarchy） • Actor-Mixer Hierarchy 对象： <ul style="list-style-type: none"> • Actor-Mixer（角色混音器） • Random Container（随机容器） • Sequence Container（序列容器） • Switch Container（切换容器） • Blend Container（混合容器） • Sound SFX（音效声） • Sound Voice（语音声） • Interactive Music Hierarchy 对象： <ul style="list-style-type: none"> • Music Switch Container（音乐切换容器）

列标题	内容
	<ul style="list-style-type: none"> • Music Playlist Container (音乐播放列表容器) • Music Segment (音乐段落) • Music Track (音乐轨) <p>如果此列留空，则会假定 Object Type 为 Virtual Folder，除非在 Object Path 中另行指定。</p> <div style="background-color: #f0f0f0; padding: 5px;">  备注 使用 Audio File Importer 时，无法同时创建 Actor-Mixer 层级结构对象和 Interactive Music 层级结构对象。在 Actor-Mixer Hierarchy 中，也无法同时创建 Sound SFX 对象和 Sound Voice 对象。但是，在使用 Wwise Authoring API 和 TabDelimitedImport 命令行选项时，则可以同时导入 Actor-Mixer Hierarchy 对象和 Interactive Music Hierarchy 对象。 </div>
Property[属性名] 或 @属性名 示例： <ul style="list-style-type: none"> • <i>Property[Voice Volume]</i> • <i>Property[Volume]</i> • <i>@Voice Volume</i> • <i>@Volume</i> 	<p>列标题指定要设置哪个属性。属性名称为显示名称，如 RTPC（实时参数控制）选项卡、Multi Editor 或 List View 中所示。</p> <p>如果没有文件或缓存版本与 Wwise 的版本不同，则在 Wwise 中再次播放媒体文件之前，该缓存将会被删除而您将不得不重新对平台版本文件做转码。https://www.audiokinetic.com/library/edge/?source=SDK&id=wobjects_index.html</p> <p>常见属性名称：</p> <ul style="list-style-type: none"> • Volume • Pitch • Lowpass • Highpass <p>通用属性显示名称：</p> <ul style="list-style-type: none"> • Voice Volume (声部音量)。 • Voice Pitch (声部音高) • Voice Low-pass Filter (声部低通滤波器) • Voice High-pass Filter (声部高通滤波器)。 <p>各个后续行指定要在创建对象上设置的属性值。</p> <p>枚举值必须指定与枚举名称有关的整数值，它可从 WWU 文件或 WObjects.xml 中找到。</p> <p>如果此列留空，则未设置任何值。</p>
Reference[引用名称] 或 @ReferenceName 或 ~ReferenceName (不建议使用) 示例： <ul style="list-style-type: none"> • <i>Reference[Output Bus]</i> • <i>Reference[OutputBus]</i> • <i>@Output Bus</i> • <i>@OutputBus</i> 	<p>列标题指定要设置哪个引用。Reference name 可以是引用名称或者显示名称，如 List View 和 Multi-Editor 中所示。</p> <p>请参阅 Wwise Objects Reference 来获取更多关于引用名称的信息。</p> <p>通用引用显示名称：</p> <ul style="list-style-type: none"> • Output Bus (输出总线) • User Auxiliary Send 0 (用户辅助发送 0) • User Auxiliary Send 1 (用户辅助发送 1) • User Auxiliary Send 2 (用户辅助发送 2) • User Auxiliary Send 3 (用户辅助发送 3) • Conversion Settings (转码设置) • Effect 0 (效果器 0) • Effect 1 (效果器 1)

列标题	内容
	<ul style="list-style-type: none"> Effect 2 (效果器 2) Effect 3 (效果器 3) Attenuation (衰减) MIDI Target (MIDI 目标) <p>各个后续行指定以下内容之一：</p> <ul style="list-style-type: none"> 绝对路径：工程中对象的绝对路径。 例如：<code>\Master-Mixer Hierarchy\Default Work Unit\Master Audio Bus\Sub Bus</code> GUID：对象的 GUID，可以在 WWU 文件中找到；在 Wwise 设计工具中，打开对象快捷菜单的同时按住 Shift 键也可以直接看到。 例如：<code>{10EC26E3-03EB-4470-8B30-4344B6F90438}</code> <p>如果此列留空，则未设置任何引用。</p>
Notes (备注)	为要创建的对象定义说明或备注。
Audio Source 说明或备注	为新建对象中的音频源文件进行说明或备注。
Event	<p>为导入的对象定义要创建的 Event 路径和名称。</p> <p>路径可是绝对路径或相对路径：</p> <ul style="list-style-type: none"> 绝对路径：事件的绝对路径，以 <code>\Events\</code> 开头，后跟工作单元名称。 例如：<code>\Events\MyWorkUnit\PlayEvent</code> 相对路径：事件相对于 Default Work Unit 的相对路径。 例如：<code>MyVirtualFolder\PlayEvent</code> <p>要指定动作类型，请在事件名称后附加 <code>@ActionName</code>。如果未指定，则将使用 <code>>Play></code> (播放) 动作。</p> <p>Action Type 示例：</p> <ul style="list-style-type: none"> <code>MyEvent@Play</code> <code>MyEvent@Stop</code> <code>MyEvent@Pause</code> <code>MyEvent@Resume</code> <code>MyEvent@Break</code> <code>MyEvent@Seek</code> <p>对于目标处不存在的任何路径元素，都将创建一个虚拟文件夹。</p> <p>若目标位置已存在指定 Event，则会将新的 Event Action 添加到现有 Event (如尚未添加完全一样的 Event Action)。</p> <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;">  技巧 要为单个对象创建多个事件，您可以多次添加 Event (事件) 列。 </div>
Dialogue Event (对白事件)	<p>为导入的对象定义要创建的 Dialogue Event 路径和名称。路径可是绝对路径或相对路径：</p> <ul style="list-style-type: none"> 绝对路径：对白事件的绝对路径，以 <code>\Dynamic Dialogues\</code> 开头，后跟工作单元名称。 例如：<code>\Dynamic Dialogues\MyWorkUnit\DialogueEvent</code> 相对路径：对白事件相对于 Default Work Unit 的相对路径。

列标题	内容
	<p>例如: <code>MyVirtualFolder\DialogueEvent</code></p> <p>Dialogue Event 列的主要用于创建或修改 Dialogue Event 中的 Path，等同于在 Dialogue Event Editor 中进行操作。要指定 Path，需要在 Dialogue Event 的名称后面附加 <code>@Path:</code>，然后添加 Switch/State 结构，使用英文句号连接。</p> <p>还可以指定 Path 的 Probability 和 Weight 值（范围从 0 到 100）。只需要添加 <code>/</code>，然后输入 <code>Probability:###</code> 或 <code>Weight:##</code>。有关详细信息，请参阅为路径设置概率和权重。如果未指定 Path，Probability 或 Weight 参数将不起作用。</p> <div style="background-color: #f0f0f0; padding: 5px;">  技巧 要为单个对象创建多个对白事件，您可以多次添加 Event（对白事件）列。 </div> <p>Dialogue Event 示例: . <code>VirtualFolder\Unit_Under_Attack@Path:<Switch Group:MyVirtFolder\FS_Type>Running.<State Group:Hostile>Gang.<Switch Group:VirtualFolder\Location>Alley/Probability:80/Weight:20</code></p>
Switch Assignment	<p>定义 Switch Container 与哪个 Switch Group 或 State Group 关联，仅适用于 Actor-Mixer Hierarchy 中的对象。还定义 Switch Container 中的哪一个子项将要分配给关联 Group 中的 Switch 或 State。</p> <p>Switch Group 或 State Group 需要在 <code><></code> 中声明，指定组类型及其名称，并用冒号分隔。要执行分配，可以在 <code>></code> 后面指定其子项。参见以下示例，了解更多详情。</p> <div style="background-color: #f0f0f0; padding: 5px;">  技巧 路径中，可以在 Switch 或 State Group 名称前指定 Virtual Folder 层级结构。 </div> <div style="background-color: #f0f0f0; padding: 5px;">  技巧 可以在与 Switch Container 关联的组名称前指定 Virtual Folder 层级结构。 </div> <div style="background-color: #f0f0f0; padding: 5px;">  技巧 要为单个对象创建多个 Switch Assignment，您可以添加多个 Switch Assignment 列。<segment 0594> </div> <p>Switch Assignment 示例: . <code><Switch Group:YourSwitchGroupName>YourSwitchName , <Switch Group:SS_FS_Type>Jump</code></p>

导入用制表符分割的文件的方法如下：

1. 通过以下方式打开 Audio File Importer：
 - 在 Project Explorer 中选择导入的目标对象，然后从其快捷菜单 (Shift+I) 中选择 Import Audio Files... 选项。
 - 从 Project 菜单中选择 Import Audio Files...。

此时将会打开 Audio File Importer。

2. 点击 Import Tab Delimited... (导入用制表符分割的文件...) 按钮。
3. 浏览找到该用制表符分割的文件，然后点击 OK (确定)。
4. 验证 Audio File Importer 列表中的导入内容。
5. 点击 Import (导入)

用制表符分割的文件示例

以下表格（使用了 Sample Project 中的细节，但有差别）即使用制表符分割的电子表格示例。尽管这些表格不能包罗万象，但却展示了使用不同方式导入相似项的多种可能性，例如 Object Type 列与 Object Path 中的对象类型的使用，以及绝对路径与相对路径的使用。

一次创建路径中的多个对象：演示如何使用一行表格创建整个路径结构中的多个容器。Each of the object Types specified in the object Path could have been entered in separate lines with entries in the object Type column.

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

创建 Switch Container，并设置 Switch Group 和 Switch 分配：演示如何添加 Switch，然后为其分配 Switch Group 和 Switch。

Object Path	Object Type	Switch Assignment
\Actor-Mixer Hierarchy\My Default Work Unit\<Actor-Mixer>Minigun \<Switch Container>Shells\A	Sound SFX (音效声)	<Switch Group:SS_Surface>Dirt
Actor-Mixer Hierarchy\My Default Work Unit\<Actor-Mixer>Minigun \<Switch Container>Shells\B	Sound SFX (音效声)	<Switch Group:SS_Surface>Wood_Solid

通过 Dialogue Event 添加 Sound Voice：演示如何通过 Dialogue Event 和 Path 添加关联的 Sound Voice。

Audio File	Object Path	Object Type	Dialogue Event (对白事件)
..\..\My Voice Recordings \English(US)\UNA-GG-AL_01.wav	\Actor-Mixer Hierarchy\ Dialogues\Captain_A\UNA-GG-AL_01	Sound Voice (语音声)	\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 (语音声)	Unit_Under_Attack@Path:<State Group:Unit>Unit_A.<Switch Group:FS_Type>Running.<State

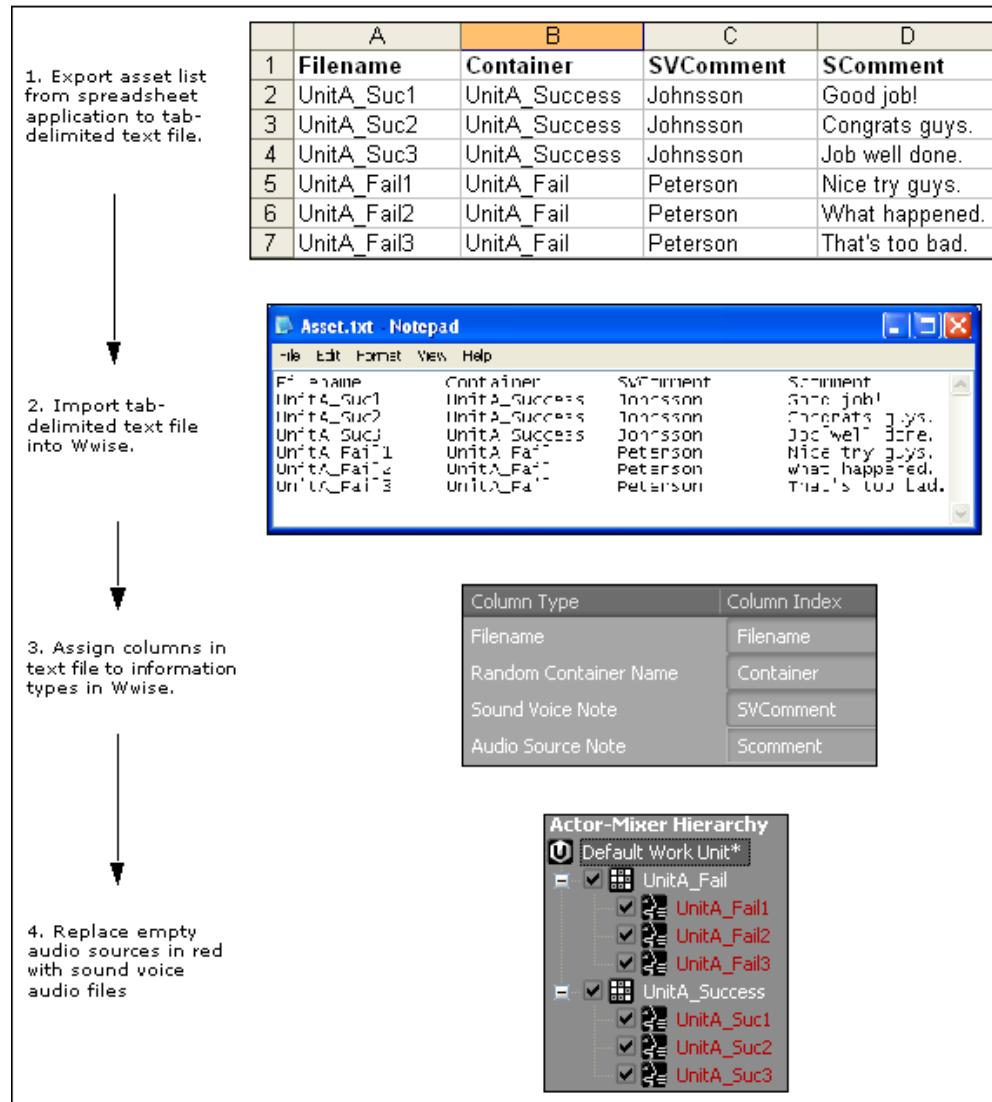
Audio File	Object Path	Object Type	Dialogue Event (对白事件)
			Group:Hostile>DefuseBomb.<State Group:Location>Alley Probability:20 Weight:20
..\..\My Voice Recordings\English(US)\UNA-GG-HG_01.wav	{D2F87A0D-7D5C-429DB13D-03478450C2A2}	Sound Voice (语音声)	Unit_Under_Attack@Path:<State Group:Unit>Unit_A.<Switch Group:FS_Type>Running.<State Group:Hostile>Gang.<State Group:Location>Hangar

导入文本文件中的语音素材

游戏中具有成千上万语音素材的情况并不少见，所以在 Wwise 中通过有效方法创建这些语音素材至关重要。由于这些素材通常在 Microsoft Excel 等外部程序中进行管理，因此 Wwise 可以通过导入用制表符分割的文件中的信息，在 Wwise 工程中重建这些素材。这不仅会加快创建过程，而且还可减少出错误的可能。

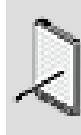


下图说明了将电子表格程序中的语音素材列表导入到 Wwise 中的工作流程。



尽管您的文本文件可能包含有关素材的多种信息，但在导入文件时 Wwise 仅使用以下四种类型的信息：

- **Filename** —— 文件名。此信息用于创建 Sound Voice 对象和其对应的音频源。最初音频源将为空，但它的的确包含对相同名称的音频文件的引用。也就是说，您可以在 Wwise 中像处理其它音频文件那样替换它。有关为 Sound Voice 对象替换媒体文件的详细信息，请参阅[替换 Sound Voice 媒体文件](#)。
- **Random Container name** —— 随机容器名称。如果文本文件中包含此信息，则将被用于创建相应 Sound Voice 对象的父随机容器。
- **Sound Voice notes** -- 声音对象备注。此信息会添加到 Sound Voice 对象的 Notes (备注) 字段。例如，您可能会需要添加说话角色的名称。
- **Audio source note** —— 音频源备注。此信息会添加到音频源的 Notes 字段。例如，您可能会需要包含所说的实际对话文本。



备注

在将语音素材导入 Wwise 时，仅以引用语言创建音频源。要在工程中创建其它语言版本的音频源，请参阅[导入语言文件](#)。

文本文件中信息的顺序和信息量并不重要，因为在导入文件之前，您必须指定要导入文本文件中的哪些列。唯一强制的信息是文件名，因为此名称会用于创建 Sound Voice 对象和音频源。如果文本文件中的某列尚未映射到四个信息类型之一，则它将被 Wwise 忽略。

为了帮助您确保您已正确指定了列，会显示将要创建的对象和将要导入的信息的预览。

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.
...



注意

在已导入文本文件中的语音素材之后，此操作无法撤消。

导入文本文件中的语音素材的方法如下：

1. 从 Project 菜单中，选择 >Import Voice Assets>（导入语音素材）。

此时将会打开 Voice Asset Importer。

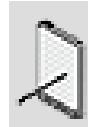
2. 点击 File to import（要导入文件）文本框旁边的浏览按钮 (...)。

此时将会打开 Open（打开）对话框。

3. 选择要导入的文本文件，然后点击 Open。

导入文件的位置将显示在 File to import 中。

4. 如果文本文件中有标题行并且您希望使用它来指派列，则请选择 Use header（使用标题）复选框。
5. 在 Header row 选择框中，指定文本文件中的标题信息行。
6. 在 Start import at row 选择框中，选择 Wwise 读取语音素材信息的起始行。
7. 从 Column Index（列索引）列表中，为以下各列类型选择文本文件中的相应列：



备注

文本文件中的 filename 列必须指派给相应的 Filename 信息类型，以便在 Wwise 中创建文本文件中的语音素材。所有其它信息为可选信息。

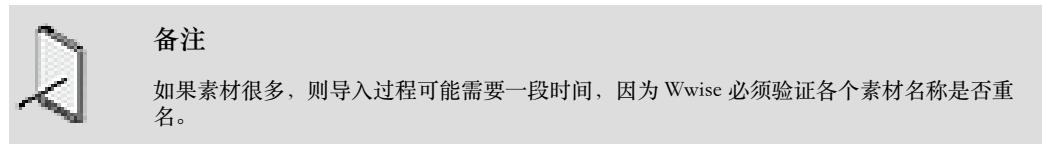
- Filename

- Random Container Name
 - Sound Voice Note
 - Audio Source Note
8. 检查 Voice Assets Preview (语音素材预览) 表中的信息，以确保您将文本文件中的列正确指派给了 Wwise 中的相应信息类型。
9. 如果您希望更改将创建新 Sound Voice 的位置，则请执行以下操作：
- 点击 Import destination (导入目标) 文本框旁边的 Browse 按钮。
 - 在 Project Explorer - Browser 中，在工程层级结构中选择一个新位置。
 - 点击 OK。

新路径将显示在 Import destination 文本字段。

10. 点击 Import (导入)

Wwise 中会创建 Sound Voice 对象，如果有相应的音频源和随机容器，则也会随之一起进行创建。如果在导入期间遇到问题，Wwise 则会将这些问题列在 Import Completed (导入完成) 对话框中。



相关主题

- 替换 Sound Voice 媒体文件
- 导入用于旁白的媒体文件
- 导入用于音效的媒体文件
- 管理文件导入问题
- 导入语言文件

替换媒体文件

在某些情况下，之前导入的文件可能存在技术问题，或您可能正在使用临时文件作为占位符文件，直到您收到打算在工程中使用的文件为止。在这些情况下，您需要替换之前已导入的媒体文件。您可以在 Audio File Importer 中或使用 Quick Replace (快速替换) 快捷方式来执行此操作。

快速替换

下表列出了您可以用于替换之前导入 Wwise 中的媒体文件的快捷方式。

操作...	使用此快捷方式:
更换现有音效文件并创建新声音对象。	拖放文件。
更换音效文件。	按 Alt 拖动文件。
更换现有语音文件并创建新的声音对象。	按 Shift 拖动文件。
更换语音文件。	按 Alt+Shift 并拖动文件。

操作...	使用此快捷方式: 如果您正使用此快捷方式替换语言，则请确保在 Language Selector 中选择了引用语言。
-------	---

替换音效文件

使用 Audio File Importer，您可以替换工程中的现有音效文件。

替换现有音效媒体文件的方法如下：

1. 执行以下操作之一：

- 从 Wwise 菜单栏中，点击 Project > Import Audio Files。
- 右键点击 Actor-Mixer Hierarchy 中的对象，然后从菜单中选择 Import Audio Files。

此时将会打开 Audio File Importer。

2. 点击 Add（添加）或 Add Folders。

此时将会打开 Open File 对话框。

3. 浏览至您要导入的媒体文件的位置。

4. 选择文件，然后点击 Open。

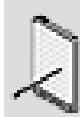
所选文件于是加载到 Audio File Importer 中了。

5. 在 Import Mode 组框中，选择 >Replace audio files>。

6. 在 Object Type 组框中，选择 >Import as Sound SFX。

7. 点击 Import（导入）

此时将会显示 Importing 对话框，您可在其中查看导入过程的进度。



备注

如果媒体文件中存在错误或冲突，则将会打开 Import Conflict Manager。有关如何处理这些冲突的详细信息，请参阅[管理文件导入问题](#)。

相关主题

- [导入媒体文件](#)
- [替换 Sound Voice 媒体文件](#)
- [管理文件导入问题](#)
- [重新组织 Originals Folder 中的媒体文件。](#)
- [清除缓存](#)
- [在外部编辑器中编辑音频文件](#)

替换 Sound Voice 媒体文件

您可以在 Audio File Importer 中以替换音频文件模式来替换工程中的现有语音媒体文件。在此模式中，转码设置选项不可用。如果您正在本地化您的工程，则您将需要使用 Localize Languages（本地化语言）模式。有关本地化的详细信息，请参阅[对工程进行本地化](#)。

替换工程中的语音媒体文件的方法如下：

1. 执行以下操作之一：

- 从 Wwise 菜单栏中，点击 Project > Import Audio Files。
- 右键点击 Actor-Mixer Hierarchy 中的 Sound Voice 或工作单元，然后从菜单中选择 Import Audio Files。

Audio File Importer 将打开。

2. 点击 Add。

此时将会打开 Open (打开) 对话框。

3. 浏览至您要导入的媒体文件的位置。

4. 选择这些文件，然后点击 Open (打开)。

所选文件于是加载到 Audio File Importer 中了。

5. 在 Import Mode 组框中，选择 >Replace audio files>。

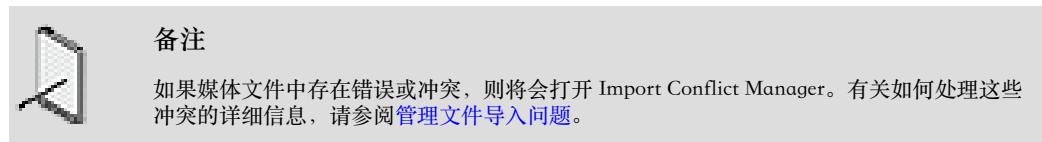
6. 在 Object Type 组框中，选择 >Import as Sound Voice>。

7. 从 Destination language (目标语言) 列表中，选择语言。



8. 点击 Import (导入)

此时将会显示 Importing 对话框，您可在其中查看导入过程的进度。



相关主题

- [导入媒体文件](#)
- [导入文本文件中的语音素材](#)
- [替换音效文件](#)
- [管理文件导入问题](#)
- [重新组织 Originals Folder 中的媒体文件。](#)
- [清除缓存](#)
- [在外部编辑器中编辑音频文件](#)

管理文件导入问题

在将媒体文件导入 Wwise 工程中时，您可能会遇到错误。可能出现两种错误：

- [管理可恢复错误](#)可在 Conflict Manager (冲突管理器) 中解决。

- **管理不可恢复的错误**无法在 Wwise 中解决。

管理可恢复错误

如果在尝试导入 Wwise 中已经存在的文件时，您并没有在 Audio File Importer 中选择 Replace Mode，则 Conflict Manager 将会打开。基于您计划执行的操作，您有三个选项。

- Replace -- 替换。使用将要导入的文件替换现有音频文件。
- Use Existing -- 使用现有。继续使用当前链接到音频源的文件。
- Cancel -- 取消。取消导入操作。

您可以逐个应用选项，或者将这些选项应用于错误列表中的所有文件。

替换包含可恢复错误的文件的方法如下：

1. 在 Conflict Manager 窗口中，执行以下操作：

要替换所有文件，请点击 Set All To（全部设置为）区域中的 Replace。

要替换单个文件，请点击 Error List 中的 Replace。

2. 点击 Import（导入）

导入列表中的文件会导入您指定的位置。

管理不可恢复的错误

当您在 Conflict Manager 中遇到不可恢复的错误时，您必须取消受影响文件的文件导入操作。

在您无法解决 Wwise 中不可恢复的错误时，查看可能出现的不可恢复的错误类型将非常有帮助。这些问题中的某些问题可以在 Wwise 之外得到解决，您可以尝试再次导入文件。

查看下表中所列的不可恢复的错误以及有关解决这些错误的建议。有关 Wwise 支持的文件的详细信息，请参阅 [PCM（脉冲编码调制）音频文件格式](#)。

原因	建议
Unsupported file format（不支持的文件格式）	将媒体文件的格式更改为波形音频格式 (.wav)。
Sample rate beyond specified range（采样率超过特定范围）	将音频文件的采样率更改到 1 - 96 kHz 范围内。
Bit rate beyond specified range（比特率超过特定范围）	将音频文件的比特率更改为 16 位或 24 位。
Unsupported input channel configuration（不支持的输入声道配置）	将声道配置的音频声道数减少到 0.1 和 7.1 之间或将媒体文件的格式更改为支持声道配置的格式。
Audio file not found（找不到音频文件）	对于 Sound Voice 对象，引用语言没有媒体文件。
Audio file name is over 256 characters（音频文件名称超过 256 个字节）	重命名媒体文件，以便文件名少于 256 个字节。

相关主题

- [PCM（脉冲编码调制）音频文件格式](#)
- [管理可恢复错误](#)
- [清除缓存](#)
- [在外部编辑器中编辑音频文件](#)

重新组织 Originals Folder 中的媒体文件。

随着工程中的源媒体文件数量的增加，您可能会需要将文件组织到一系列的子文件夹中。由于 Wwise 需要跟踪各个源文件的完整路径，所以您必须在 Wwise 的 File Manager（文件管理器）中执行移动操作。



在 Originals 目录内的文件夹之间移动文件的方法如下：

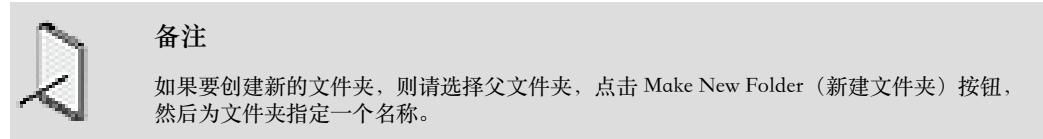
1. 保存当前工程。
2. 执行以下操作之一：
 - 在菜单栏中，点击 Project > File Manager...。
 - 按 Shift+F1。

File Manager 将打开。

3. 切换至 Source Files 选项卡。
4. 选择要移动的文件。
5. 右键点击选定项，然后从菜单中选择 Move（移动）。

此时将会打开 Browse For Folder（浏览文件夹）对话框。

6. 浏览文件夹结构并选择要移动其中文件的文件夹。



7. 打开 OK 以将文件移动到新文件夹。

此时将会打开 Process Log（流程日志）对话框并显示有关各个移动的文件的信息。

8. 点击 OK 以关闭 Process Log 对话框。

相关主题

- [导入媒体文件](#)
- [替换媒体文件](#)

- 管理文件导入问题
- 清除缓存
- 在外部编辑器中编辑音频文件

清除缓存

要高效管理您的转码媒体文件，您需要定期清除缓存文件夹，以删除不再使用的文件、过时文件或有问题的文件。您可以定义此操作的范围以有选择地清除缓存中的媒体文件。您可能会需要在以下情况下清除缓存：

- 在您已删除 Wwise 对象后——转码媒体文件或落单文件不再有用，但仍与对象关联，除非您手动清除这些文件，否则这些文件会一直保留在缓存文件夹中。在此情况下，您可能仅希望清除这些文件。
- 针对某平台转换文件后可能导致质量降低——您可能希望删除该平台的已转换文件并重新进行转换。
- 多语言文件的最终版本已经提交——您希望在缓存中仅保留新转码结果文件。

对于上述每种情况，您可以通过以下项来有选择地清除缓存文件夹：

- **Converted files** -- 转码结果文件。指定要清除哪些类型的文件。您可以清除缓存中的所有转码结果文件或仅清除落单文件。
- **Platforms** -- 平台。指定要为哪个平台清除文件。
- **Languages** -- 语言。指定要清除的多语言文件。



注意

清除缓存的操作无法撤消。

清除工程缓存的方法如下：

1. 从 Project 菜单中，选择 >Clear Audio File Cache>（清除音频文件缓存）。

此时将会打开 Clear Audio File Cache 对话框。

2. 在 Audio Files（音频文件）组框中，选择以下选项之一：

Only orphan files（仅落单文件）

All converted files（所有转码结果文件）

3. 在 Platforms 组框中，选择以下选项之一：

- Current platform（当前平台）
- All platforms（所有平台）

4. 在 Language 组框中，为 Sound Voice 对象选择以下选项之一：

- Current language（当前语言）
- All languages（所有语言）

5. 点击 OK。

此时将会打开 Clearing Audio File Cache 对话框，您可在其中看到清除过程。

在清除过程成功完成后，您指定的转码结果媒体文件将从缓存中移除。

相关主题

- [导入媒体文件](#)
- [替换媒体文件](#)
- [管理文件导入问题](#)
- [重新组织 Originals Folder 中的媒体文件。](#)
- [在外部编辑器中编辑音频文件](#)

在外部编辑器中编辑音频文件

在准备要用于游戏工程的音频文件时，您可以使用若干个音频编辑器来完成任务。但是在将这些音频文件整合到工程中后，您可能会需要进一步编辑这些文件。例如，您可能会需要剪掉过长演讲的末尾部分，或为一段口哨吹的小调加一个音符。Wwise 可以选择已经导入工程的任何音频文件，并直接在您喜欢的音频编辑器中打开该文件。这样一来，即使在文件已整合到工程中后，您也可以微调文件。



如果您有所需权限，则可以选择 Project Explorer 中所列的导入音频文件并直接在您所选的外部音频文件编辑器中打开这些文件。有关文件权限的详细信息，请参阅[结合版本控制系统使用 Wwise](#)。

在外部编辑器中编辑音频文件的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键点击声音对象。

此时将会显示快捷菜单。



2. 按 Ctrl-E 或从快捷菜单中，选择 Edit in External Editor（在外部编辑器中编辑）。

此时将会显示可用编辑器的列表。

3. 选择要用于编辑音频文件的编辑器。

对象的音频文件会在您选择的外部编辑器中打开。

4. 在您编辑完文件后，请在该编辑器中将文件保存到以下路径之一下面：

对于音效，请保存到 [ProjectName]\Originals\SFX\[Filename.wav] 下

对于语音，请保存到 [ProjectName]\Originals\Voices\[Language]\[Filename.wav] 下

编辑后的媒体文件现已能够在 Wwise 中播放了。



注意

请勿重命名您已编辑的文件。否则，Wwise 将无法检索该文件。

相关主题

- [导入媒体文件](#)
- [替换媒体文件](#)
- [管理文件导入问题](#)
- [重新组织 Originals Folder 中的媒体文件。](#)
- [清除缓存](#)

使用插件创建 Source

Wwise 的开放式体系结构使您能够通过创建源插件（Source Plug-in）来提升音频效果。Wwise 的开放式体系结构使您能够通过创建源插件来增强音频和振动。这些插件（例如合成器和物理建模）可轻松集成到 Wwise 中，并可用于创建声音和振动对象。您还可以修改插件属性以创建更广泛的音效和振动效果。

Wwise 自带不少的源插件。

可用的源插件

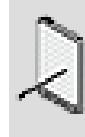
- Wwise Audio Input (Wwise 音频输入)
- Wwise External Source (Wwise 外部源)
- Wwise Silence (Wwise 空白信号源)
- Wwise Sine (Wwise 正弦波)
- Wwise Tone Generator (Wwise 乐音发生器)
- Wwise Motion Generator (不建议使用)
- Wwise Motion Source (Wwise 振动源)
- SoundSeed Air - Wind*
- SoundSeed Air - Woosh*

* —— 如果您计划为您的游戏开发、集成和发布 SoundSeed Air，则需要购买单独的授权。有关详细信息，请联系 Audiokinetic 销售团队，邮件地址是：sales@audiokinetic.com。

添加源插件的方法如下：

1. 将对象加载到 Property Editor 中。
2. 在 Contents Editor 中，点击 Add Source。

Source 菜单中将会显示可用的源插件列表。



备注

为特定平台创建的源插件，在所有其它平台的源插件列表中将显示为不可用。

3. 选择要添加的源插件。

源已添加到对象中，并在 Contents Editor 中显示为新条目。

4. 双击源插件以在 Source Plug-in Property Editor（源插件属性编辑器）中显示完整的属性列表。
5. 根据需要，编辑属性。

相关主题

- [将对象添加到 Contents Editor](#)
- [在 Contents Editor 中试听对象和源](#)

媒体文件管理技巧和经验总结

以下几节为您提供了一系列技巧和经验总结，以帮助您在整个音频开发过程中更好地管理工程中的媒体文件。

清除缓存

要节省空间并有效管理文件，您需要定期清除缓存。您应在以下情况下执行此操作：

- 在删除对象后。
- 在 File Manager（文件管理器）中删除文件后。

File Manager

另一种跟踪工程媒体文件的方法是查看 File Manager（文件管理器）的 Source Files（源文件）选项卡中的信息。File Manager 持续跟踪工程的 Originals 文件夹，以便您可以根据需要从工程中删除任何未使用的文件。但是，删除 File Manager 中的音频文件无法清除音频缓存中与这些文件相关联的转码结果文件。您将需要手动删除这些文件，方法是在 Clear Audio Cache 对话框中选择 Only orphan files 选项。有关 File Manager 的详细信息，请参阅[查看工程文件的状态](#)。

第8章 构建 Actor-Mixer Hierarchy

概述	159
什么是声音对象?	159
在创建 Actor-Mixer Hierarchy 时为对象分组	161
构建 Actor-Mixer Hierarchy	164
工程层级结构中的属性介绍	168
通过随机化属性值来改善播放	176
构建 Actor-Mixer Hierarchy 的技巧和经验总结	177

概述

一个典型游戏中可能有成千上万的素材需要管理，允许用计算机上管理众多文件一样的方式来组织它们就显得十分重要。在 Wwise 中，您可以将工程素材编组，从而更加轻松地管理它们。将对象分组并创建父/子关系，即可构建层级结构，这样不仅能组织工程素材，还允许设置各编组的属性和行为。如果多个对象需要共享属性或行为，实用的方法是将它们编为一组，这样既节省时间又优化了开发流程，可谓事半功倍。

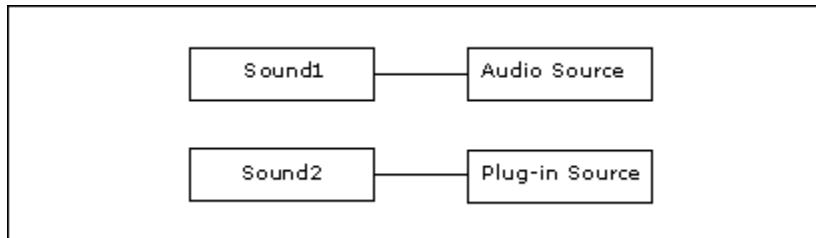
通过结合使用各种对象类型，可在工程中对素材进行分组，并构建 Actor-Mixer Hierarchy（角色混音器层级结构）。以下页面进一步阐述了如何使用这些对象类型。

- [什么是声音对象？](#)
- [在创建 Actor-Mixer Hierarchy 时为对象分组](#)
- [构建 Actor-Mixer Hierarchy](#)
- [工程层级结构中的属性介绍](#)
- [通过随机化属性值来改善播放](#)
- [构建 Actor-Mixer Hierarchy 的技巧和经验总结](#)

什么是声音对象？

声音对象是构建 Actor-Mixer Hierarchy（角色混音器层级结构）的基础，它们代表您为工程创作的各个音频素材。声音对象可以是用于对白的 Sound Voice（语音），也可以是用于 Actor-Mixer Hierarchy 内所有其他声音的 Sound SFX（音效）。

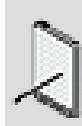
每个声音对象都包含一个源（source），即游戏中播放的实际音频内容。Wwise 中有两种源：音频源和插件源。



音频源最常见，它作为一个独立层次，将导入的音频文件和声音对象分隔开来。音频源会链接到工程中导入的音频文件。

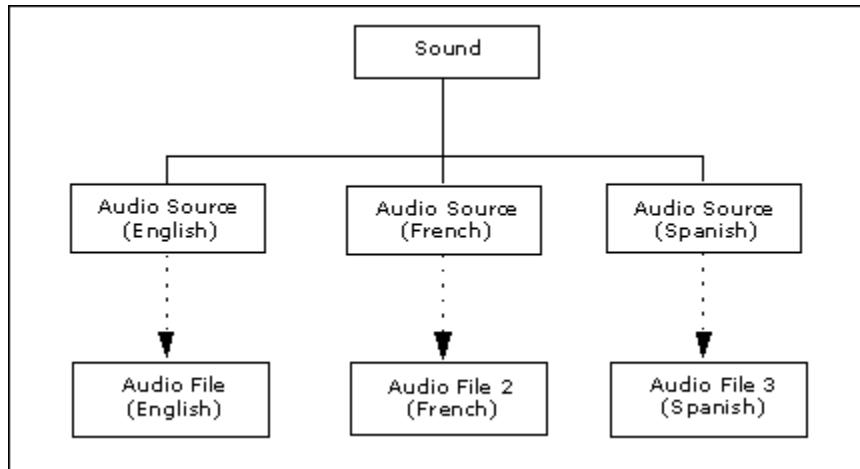


为了更加灵活有效，更适应多语言和多平台开发，Wwise 允许声音对象包含多个源。这些音频源可以链接到不同的音频文件，需要试听不同声音或不同语言版本时会非常有用。

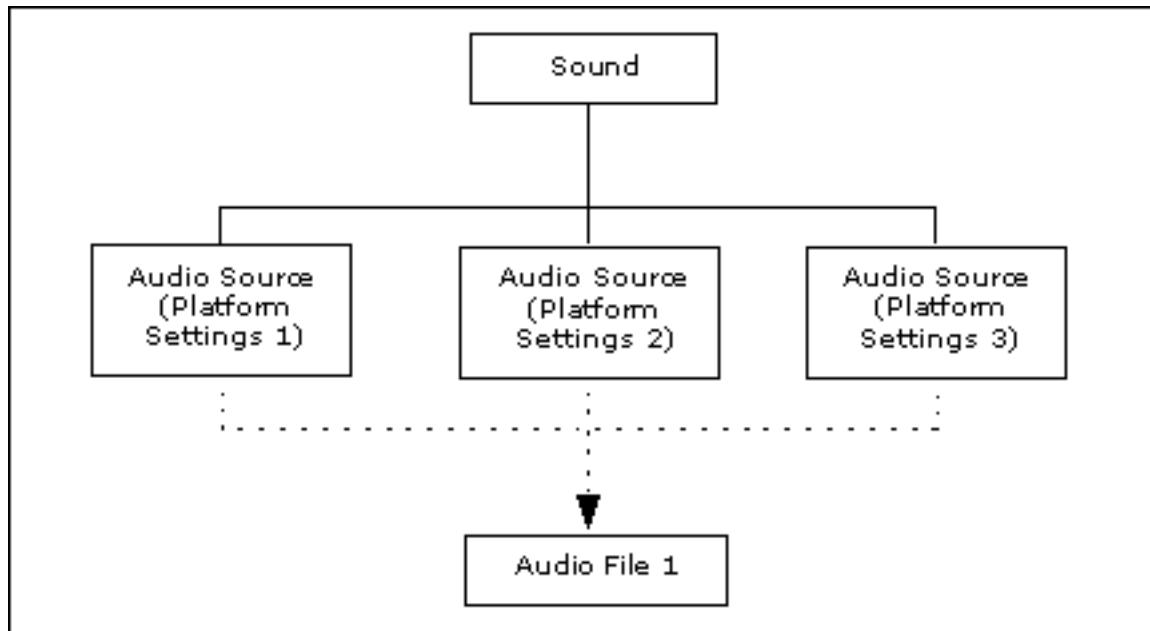


备注

音频源只有被选中，才会包含在生成的 SoundBank (声音包) 中。



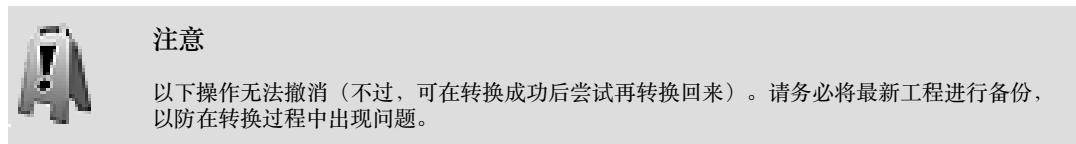
即使各目标平台使用不同采样率和其他 Conversion Setting，音频源也可以仅关联同一音频文件。



声音对象类型转换

声音对象有两种类型：Sound SFX (音效) 对象和 Sound Voice (语音) 对象。有时需要将一种对象转换为另一种对象。比如，在通过[音频文件导入](#)操作导入大量文件后发现类型有误。

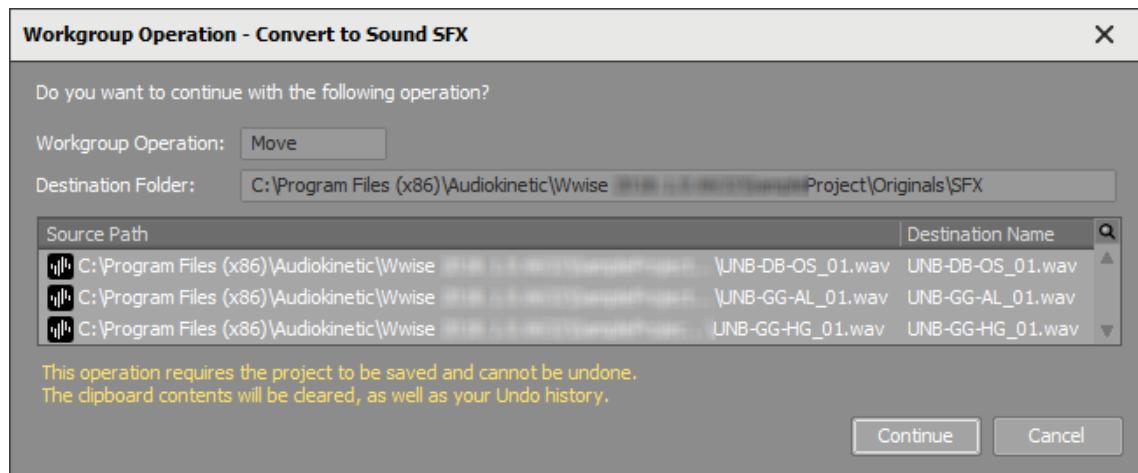
两种声音对象类型的音频源文件分别存储在对应文件夹中（即 Voices 和 SFX）。在将一种声音对象类型转换为另一种类型时，Wwise 会确保同时移动不同源文件夹内的关联源文件。不过对于 Sound Voice 对象，Wwise 仅会移动参考语言文件。



将一种声音对象类型转换为另一种类型：

1. 在 Actor-Mixer Hierarchy 内，选择要转换的 Sound SFX 对象或 Sound Voice 对象。
2. 右键单击并打开[快捷菜单](#)。
3. 单击 Convert to Sound SFX... / Convert to Sound Voice...（转换为音效.../转换为语音...）。

这时将显示 Workgroup Operation - Convert to Sound SFX/Sound Voice（工作组操作 - 转换为音效/语音）模式窗口。同时，列出声音对象详细信息，如目标文件夹和音频源路径。



4. 单击 Continue（继续），确认转换为另一种声音对象类型。

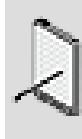
这时将运行操作，并在 Process Log（处理日志）中显示移动是否成功。

在创建 Actor-Mixer Hierarchy 时为对象分组

将不同对象编为一组，就能统一设置整个组的属性和行为了。为对象分组实际就是在为工程构建层级结构。

在 Wwise 中，可以用两种对象将其它对象分组：

- 容器类型 —— Container（容器）主要用来按照特定行为（例如随机、顺序或切换）来播放一组对象。
- 在 Actor-Mixer Hierarchy 内使用 Container 和 Actor-Mixer —— Actor-Mixer（角色混音器）通常用来设定一组对象的整体属性，例如音量和音高。



备注

也可以用 Virtual Folder（虚拟文件夹）将对象分组，但其本身不具有任何属性和行为。

由于 Container 和 Actor-Mixer 在游戏中用途不同，所以其属性、行为和能否播放都有差异。下表总结了 Container 和 Actor-Mixer 的区别。

功能	Container	Actor-Mixer
设置属性值	✓	✓
设置对象行为	✓	
播放对象	✓	

容器类型

鉴于游戏中的音频需要按不同行为播放，您可以选择不同 Container（容器）类型：Random（随机）、Sequence（序列）、Switch（切换）和 Blend（混合）。每个类型都具有不同的设置，用来定义游戏声音的播放行为。

图标	代表	描述
	Random Container	随机容器。其中的对象/容器将随机播放。
	Sequence Container（序列容器）	序列容器。其中的对象/容器将按特定顺序或播放列表播放。
	Switch Container	切换容器。其中的对象/容器可以映射到一系列 Switch（切换开关），它们对应游戏中特定条件的不同选项。具体播放哪个对象由游戏当前激活的 Switch 决定。
	Blend Container	混合容器。其中的对象/容器将同时播放。此容器内的对象可以编入 Blend Track（混合轨），在其中，对象属性可以通过 RTPC（实时参数控制）映射到游戏参数，即由参数值来控制对象属性。在 Blend Track 内的两个对象之间也可基于游戏参数的值来施加交叉淡变。

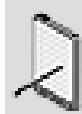
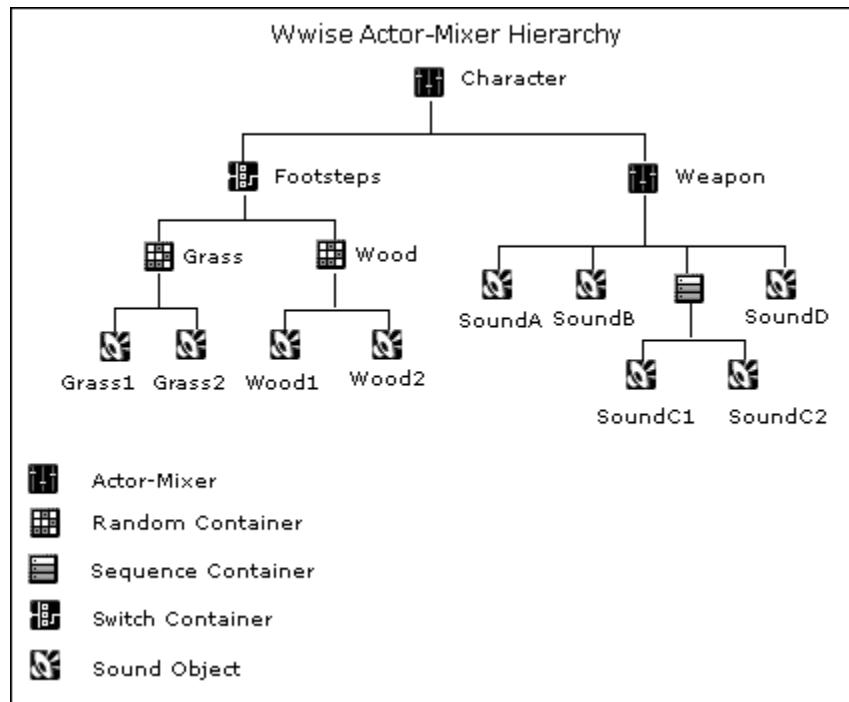
在 Actor-Mixer Hierarchy 内使用 Container 和 Actor-Mixer

Container（容器）和 Actor-Mixer（角色混音器）都可以用来将工程层级结构内的素材进行分组，但二者的层级位置和作用都不同。

容器处于 Actor-Mixer Hierarchy 中的第二层级，也就是说，它们可作为父对象和子对象。您可以使用容器来为声音和其他容器容器进行分组。通过在容器内“嵌套”其他容器，可以得到多样化的结果，模拟真实声音表现。

Actor-Mixer 也可以对工程层级结构内的对象进行分组，但它比容器高一个层级。也就是说，Actor-Mixer 可以是容器的父对象，但反之不行。Actor-Mixer 可以是任意数量的声音、容器以及其他 Actor-Mixer 的父对象。可以使用 Actor-Mixer 将大量的 Wwise 对象分组到一起，从而统一设置整个组的属性。

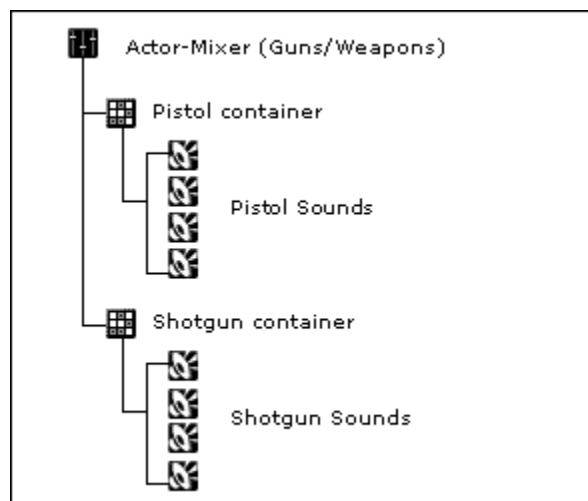
下图说明了如何使用 container 和 Actor-Mixer 对工程中某个角色的声音素材进行分组。

**备注**

对工程中的振动素材进行分组时，也可以用 Container 和 Actor-Mixer 来构建类似的层级结构。

组织素材——示例

假设您正在研发一款第一人称射击游戏，里面使用了七种不同武器，您希望每种武器的声音具有类似的属性。可以将所有与武器相关的音频分组到一个 Sequence Container（序列容器）或 Random Container（随机容器）中。之后，可以将所有武器容器分组到一个 Switch Container（切换容器）中，这样就能用它统一控制所有武器的音量和音高等属性了。



您可以在工程早期阶段根据游戏设计来建立素材结构，同时也需要考虑工程中的其它元素，例如 Work Unit（工作单元）、Routing（信号通路）以及 Game Syncs（游戏同步器）等。从工程整体角度考虑问题有助于更有效地为对象分组。关于如何为对象分组，请参阅“构建 Actor-Mixer Hierarchy 的技巧和经验总结”章节中的[在 Actor-Mixer Hierarchy 中为对象分组](#)

关于如何建立通路结构、Work Unit 以及新建 Game Sync，请参阅以下几节：

- [第 6 章 管理 Workgroup](#)
- [概述](#)
- [第 17 章 使用 State](#)
- [第 18 章 使用 Switch](#)
- [第 19 章 使用 RTPC](#)
- [第 20 章 使用 Trigger](#)
- [第 21 章 将 State 和 State Group 用于动态对话](#)

构建 Actor-Mixer Hierarchy

创建工程后，就可以开始为 Project Explorer（工程浏览器）内 Audio（音频）选项卡中的素材创建结构了。除音乐以外，素材所在的结构称为 Actor-Mixer Hierarchy（角色混音器层级结构）。> 您可以将对象添加到 Actor-Mixer Hierarchy 中，并将它们分组到不同层级，来构建对象间的相互关系。下表列出了允许添加的对象类别。

对象	图标	描述
Sound		声音对象。该 Wwise 对象代表各音频素材，并链接到音频源。共有两种声音对象：
		Sound SFX —— 音效声。即音效对象。
		Sound Voice —— 语音。即语音对象。
容器		容器。该对象可以包含 Sound 或其 Container，并且会根据特定行为播放整组对象。您可以设置 Container 属性，来影响其中的对象。Container 分为四种：
		Random Container —— 随机容器。其中的对象/容器将随机播放。
		Sequence Container —— 序列容器。其中的对象/容器将按照特定播放列表播放。
		Blend Container —— 混合容器。其中的对象/容器将同时播放。此容器内的对象可以编入 Blend Track（混合轨），在其中，对象属性可以通过 RTPC（实时参数控制）映射到游戏参数，即由参数值来控制对象属性。Blend Track 内，两个对象间也可以基于游戏参数值来进行交叉淡变。
		Switch Container —— 切换容器。其中的对象/容器可以映射到一系列 Switch（切换开关），它们对应游戏中特定条件的不同选项。
Actor-Mixer		角色混音器。一种高层对象，可以用来将其它对象编为一组。Actor-Mixer 的属性会影响其子对象的属性。 也可以用 Virtual Folder（虚拟文件夹）将 Actor-Mixer 中的对象进一步分组。
虚拟文件夹		虚拟文件夹。一种高层元素，可以包含其它对象。Virtual Folder 不能作为Container 或 Sound 的子对象。

对象	图标	描述
Work Units		工作单元。一种高层元素，用来分割工程内容，以便多人同时处理同一工程。工程层级结构内的所有素材以及其它 Wwise 元素（例如状态、效果器和 SoundBanks）都位于 Work Unit 内。
Physical Folders		实文件夹。一种高层元素，可以包含一组工程中其它的 Physical Folder 或 Work Unit。实文件夹不能作为容器或声音的子对象。

将对象添加到 Actor-Mixer Hierarchy

创建层级结构要从 Work Unit（工作单元）开始。您可以直接从 Default Work Unit（默认工作单元）开始创建层级结构，在团队中，更常见的方式是先新建 Work Unit，再添加对象至其中。关于 Work Unit 和工作组的详细信息，请参阅[第 6 章 管理 Workgroup](#)。

构建对象结构时，可以采用以下方式之一：

- 先设置好工程结构，然后将音频文件导入其中。
- 先导入音频文件，再将其组织到工程结构中。

关于如何导入音频文件以及在 Actor-Mixer Hierarchy（角色混音器层级结构）中创建新对象的详细信息，请参阅[第 7 章 管理工程中的媒体文件](#)。

使用 Project Explorer 工具栏创建子对象的方法如下：

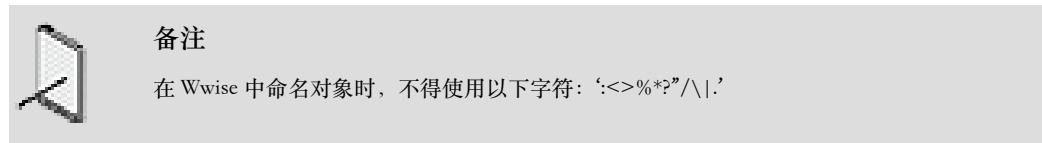
1. 在 Project Explorer（工程浏览器）的 Audio（音频）选项卡中，选择要新建对象的 Work Unit。

Project Explorer 工具栏中的一系列图标将变为激活状态。

2. 从图标列表中，点击要添加的对象的图标。

该对象于是添加到选中的 Work Unit 的 Actor-Mixer Hierarchy 中去了。

3. 将默认名称替换为最适合该对象的名称。



现在可以继续在层级结构中添加其它对象了。请提前花些时间熟悉理解对象之间的关系，以便有依据地组织它们，在工程后续进程中，这样会帮您节省大量的时间。

在 Actor-Mixer Hierarchy 中新建子对象的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键点击要新建对象的 Work Unit。
2. 在快捷菜单中选择 New Child（新建子项）。

子菜单将打开，并显示允许添加的对象列表。

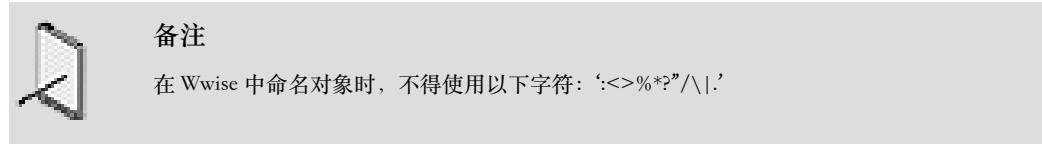
在此层级中，允许添加以下对象：

- Virtual Folder（虚拟文件夹）

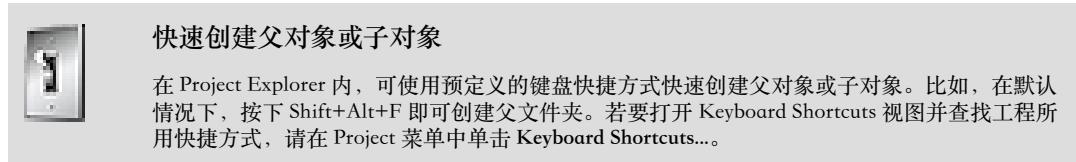
- Actor-Mixer (角色混音器)
 - Switch Container (切换容器)
 - Random Container (随机容器)
 - Sequence Container (序列容器)
 - Blend Container (混合容器)
 - Sound SFX (音效声)
 - Sound Voice (语音声)
3. 从对象列表中，点击要添加的对象。

该对象于是添加到选中的 Work Unit 的 Actor-Mixer Hierarchy 中去了。

4. 将默认名称替换为最适合该对象的名称。



现在可以继续在层级结构中添加其它对象了。请提前花些时间熟悉理解对象之间的关系，以便有依据地组织它们，从而帮您在后续工作中节省大量时间。



相关主题

- [添加父对象](#)
- [复制和粘贴对象](#)
- [移动对象](#)
- [剪切/删除对象](#)
- [在工程中创建 Work Unit](#)
- [将工程元素指派到 Work Unit](#)

添加父对象

将第一个对象添加到 Work Unit (工作单元) 中后，可以继续将其它对象添加到 Actor-Mixer Hierarchy (角色混音器层级结构) 中，并为它们创建父子关系。父对象包含其它对象，创建父对象后，您可以将现有对象移至其下。创建父子关系的好处是，更改父对象属性和定义父对象的行为时，可以影响其子对象。关于 Actor-Mixer Hierarchy 中各属性的详细信息，请参阅[工程层级结构中的属性介绍](#)。

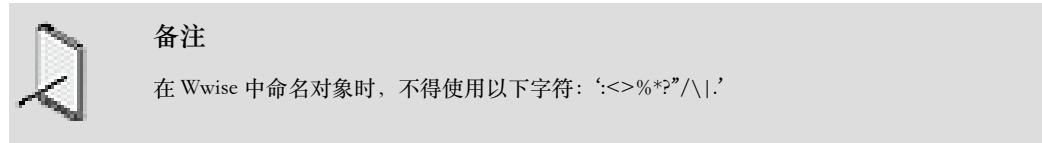
使用 Project Explorer 工具栏新建父对象的方法如下：

1. 在 Project Explorer (工程浏览器) 的 Audio (音频) 选项卡中，选择要添加父级的对象。
2. 按住 Shift 键后，工具栏将显示所选对象允许添加的父级对象图标。

3. 在列表中，点击所需的对象图标。

此对象将作为所选对象的父级添加至 Actor-Mixer Hierarchy 中。

4. 将默认名称替换为最适合该对象的名称。



新建父对象的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键点击需要新建父级的对象。
2. 在快捷菜单中，选择 New Parent (新建父项)。

子菜单将打开，并显示允许添加的对象列表。

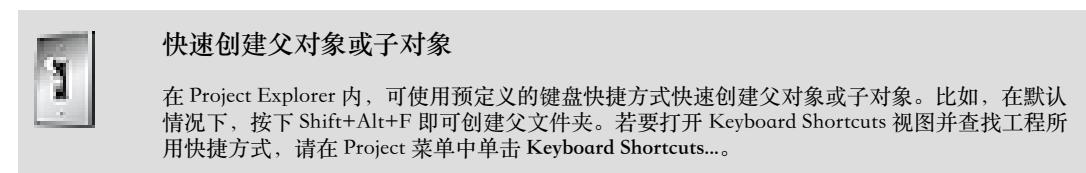
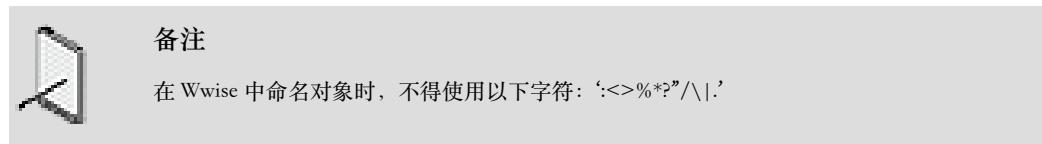
根据所选对象的结构层级不同，允许添加的父对象包括：

- Switch Container (切换容器)
- Random Container (随机容器)
- Sequence Container (序列容器)
- Blend Container (混合容器)
- Virtual Folder (虚拟文件夹)
- Actor-Mixer (角色混音器)

3. 在列表中，点击要添加的对象。

新父对象将被添加到 Actor-Mixer Hierarchy 中，并按对象类型默认命名。

4. 将默认名称替换为更合适的名称。



相关主题

- [将对象添加到 Actor-Mixer Hierarchy](#)
- [复制和粘贴对象](#)
- [移动对象](#)
- [剪切/删除对象](#)
- [在工程中创建 Work Unit](#)

- 将工程元素指派到 Work Unit

管理 Actor-Mixer Hierarchy 中的对象

在 Actor-Mixer Hierarchy (角色混音器层级结构) 中，您可以用快捷菜单和标准 Windows 快捷方式来重命名、剪切、复制和粘贴对象。更改层级结构时，请牢记以下几点：

移动对象

操作	结果
移动对象	如果将对象移动到层级结构中的新位置，该对象的属性和行为将受其新父对象影响。
	如果移动的对象与某事件相关联，则移动后，该对象仍与该事件关联。

复制和粘贴对象

操作	结果
复制对象	如果您复制对象并粘贴到新位置，该对象的属性和行为将受其新父对象影响，其子对象也会受到影响。
	如果复制的对象与某事件相关联，复制后，新对象将与该事件无关。

剪切/删除对象

操作	结果
剪切或删除对象	如果剪切或删除某对象，则其子对象也将被删除。
	但是，与其关联的已转码音频文件不会被删除，不再与对象关联的已转码音频文件称为 Orphan File (落单文件)。要删除这些落单文件，您需要清除音频缓存。关于删除落单音频文件的详细信息，请参阅 清除缓存 。
	如果您删除或剪切一个与某事件相关联的对象，则将会导致事件中缺失对象。

工程层级结构中的属性介绍

鉴于您将在游戏中建立复杂的结构，所以首先要了解不同的属性将如何影响层级结构内的各对象。在 Wwise 中，对象的属性分为两类：

- 相对属性——在层级结构中累积的属性，例如音高和音量，可以在每一层级上定义，而最终属性值由累计总和决定。了解各相对属性的最大或最小值限制至关重要，各属性的值域如下所示：
 - 音量：(-200 至 +200) 单位为 dB
 - 音高：(-2,400 至 2,400) 单位为音分
 - 低通滤波器：(0 至 100) 用百分比表示
 - 高通滤波器：(0 至 100) 用百分比表示

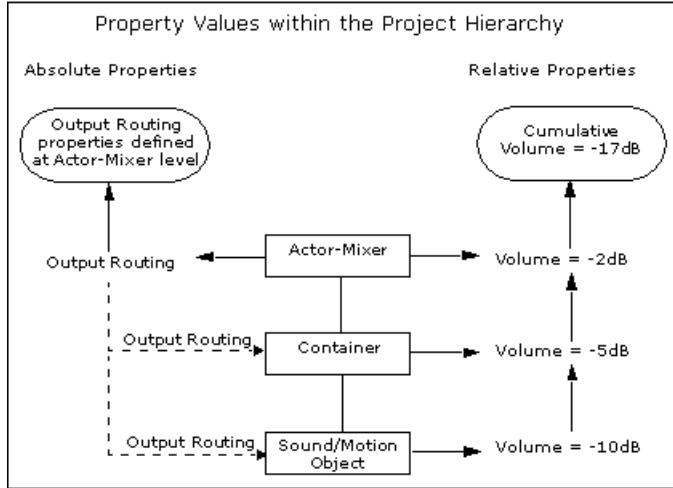
备注

并非所有相对属性都适用于所有对象，也并非在所有平台上都能用。

- 绝对属性——在特定层级（通常为最高层级）上定义的属性，可以传递给其下所有子音乐对象。例如，对象的输出通路。虽然是在最高层级设置，但实际上它们对

其下各层级都生效。在每个层级上都可以选择 Override（不沿用）父对象的绝对属性。

下图说明了工程层级结构中，这两种属性值的作用。



将素材导入工程，并用不同对象类型将它们组织到逻辑结构中之后，就可以设置各素材的属性和行为了。

Wwise 中每个对象都具有自己的一组属性和行为，您可以用它们来定义和区分对象。属性（例如音量和音高）可以定义各对象的特征，而行为（例如循环和流播放）则决定对象的播放方式。

您也可以为层级结构内的所有对象统一设置属性和行为，从而控制整个对象组的属性，并决定组内的哪些对象被播放，以什么顺序播放。

您还可以将某些效果器应用在不同的对象上，来创造您想要的独特体验。需要注意的是，在最高层级上添加效果器时，它们实际会应用在其下的各层级上。

属性标志

您可能注意到 Property Editor（属性编辑器）和 Contents Editor（内容编辑器）中，某些属性值的旁边带有几个标志。这些标志用于指示属性值是否链接到了其他活跃平台、是否关联了 RTPC 或 State（状态）、是否应用了 Randomizer。



下表说明了各标志所示的情况：

标志	名称	描述
	Link	链接。属性值已链接到其它有效游戏平台的相应值。
	Unlink	取消链接。属性值未链接到其它有效游戏平台的相应值。

标志	名称	描述
	Partial Unlink	部分链接。当前平台的属性值已链接，但其它有效平台的若干个相应值已取消链接。
	RTPC - 开启	属性值已通过 RTPC 绑定到游戏中的参数值。
	RTPC - 关闭	属性值未与游戏中的参数值绑定。
	Randomizer - 开启	该属性值的 Randomizer 已开启。
	Randomizer - 关闭	该属性值的 Randomizer 未开启。

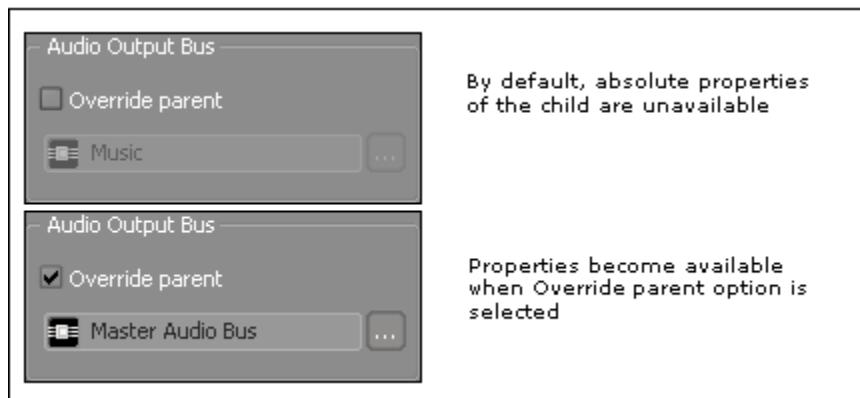
您可以右键点击特定属性值，来启用或禁用 Link 和 Randomizer 功能。可以双击 RTPC、State 和 Randomizer 标志，来打开对应视图/选项卡并查看定义的设置。

关于为属性值建立/取消 Link、使用 RTPC 以及随机化属性值的详细信息，请参阅以下几节：

- [根据平台自定义对象属性](#)
- [使用 Game Parameter 控制属性值](#)
- [通过随机化属性值来改善播放](#)

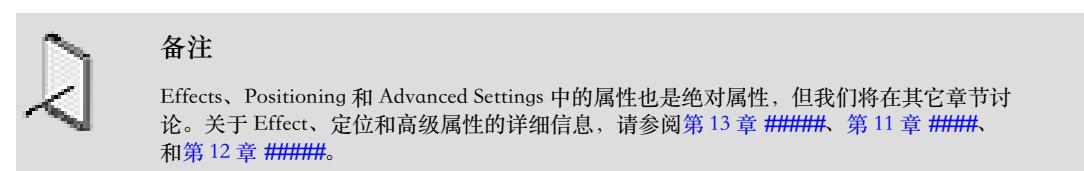
设置绝对属性

绝对属性会自动传递给各级子对象，因此应该在层级结构的顶层设置它们。如果特定子对象需要特有的属性值，可以 Override（不沿用）父对象属性，并设置其独有的属性。默认情况下，不能为子对象设置绝对属性，但勾选 Override parent（不沿用父项）选项后，绝对属性就可以修改了。



层级结构中，可以为对象设置以下绝对属性。

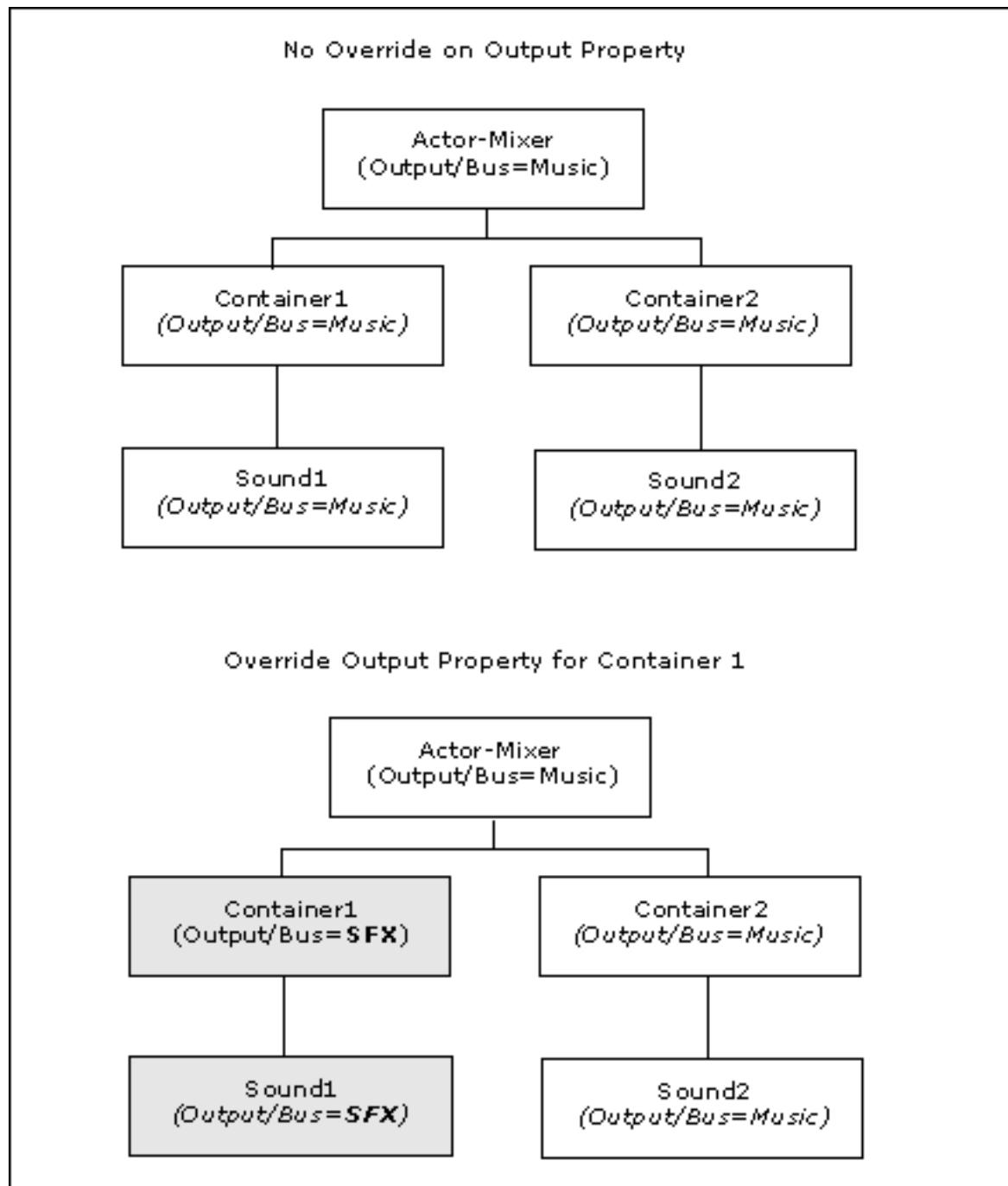
- [指定对象的输出连线](#)



Override 父级属性

如果要为子对象指定特有绝对属性值，可以 Override（不沿用）父对象属性，并为其设置独有属性。请注意，如果对象的属性被 Override，其子对象的对应属性也会取消沿用。

下图说明了Override 命令如何影响当前对象及其所有子对象的属性设置。



Override 父对象属性的方法如下：

1. 将子对象加载到 Property Editor 中。
2. 为需要的绝对属性选择 **Override parent** 选项。

该属性组将变为允许编辑。

3. 根据需要修改属性即可。



相关主题

- [设置绝对属性](#)
- [指定对象的输出连线](#)
- [定义相对属性（音量、音高、LPF、HPF）](#)

定义相对属性（音量、音高、LPF、HPF）

相对属性可以在各对象层级上定义。与绝对属性不同，相对属性是累积属性，也就是说，父对象属性值会与子对象属性值累加。

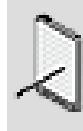
还可以为各相对属性值启用 Randomizer（随机化器），进一步提高游戏声音的真实感。有关随机化属性值的详细信息，请参阅[通过随机化属性值来改善播放](#)。

有些对象只具有一部分相对属性，有些平台也只有部分相对属性可用。下表列出了各对象和平台可用的相对属性。

对象/平台	Volume (音量)	Pitch (音高)	LPF (低通滤波器) 和 HPF (高通滤波器)
声音对象	✓	✓	✓
Containers (所有容器)	✓	✓	✓
Actor-Mixer (角色混音器)	✓	✓	✓
Music Track (音乐轨)	✓		✓
Music Segment (音乐段落)	✓		✓
Music Container (所有音乐容器)	✓		✓

为对象设置相对属性的方法如下：

1. 将对象加载到 Property Editor (属性编辑器) 中。
2. 键入值或拖动相应滑杆来设置以下属性：
 - [Volume \(音量\)](#)
 - [Pitch \(音高\)](#)
 - [Low-Pass Filter \(低通滤波器\)](#)
 - [High-Pass Filter \(高通滤波器\)](#)



备注

要同时编辑多个对象的相对属性，请在 Project Explorer（工程浏览器）中选中它们，右键点击并选择 Show in Multi Editor（批量编辑器）。

相关主题

- [设置绝对属性](#)
- [Low-pass 和 High-pass Filter 值与截止频率的关系](#)
- [通过随机化属性值来改善播放](#)
- [定义对象的播放行为](#)

[Low-pass 和 High-pass Filter 值与截止频率的关系](#)

为减少运行时 CPU 占用，并让 Low-pass Filter（低通滤波器）和 High-pass Filter（高通滤波器）使用的模型与 Volume（音量）、Pitch（音高）和 LFE（低频效果）相同（即在层级结构中是相对属性），所以 Low-pass 和 High-pass Filter 属性经过归一化，被映射至 0% 和 100% 之间。

LPF 值与截止频率的对应关系示例

假设一个声音结构由 Sound SFX（音效声）和父级 Container（容器）组成，其中：

- Sound 的 LPF 值 = 15%
- Container 的 LPF 值 = 30%

这样运行时 Sound 的最终 LPF 为 45%，对应的截止频率（Cutoff Frequency）为 1,922 Hz。

如果从开始就将 Low-pass Filter 值换算成 Wwise 内部的截止频率，大致结果如下：

- Sound 的 LPF 截止频率 = 13,500 Hz
- Container 的 LPF 截止频率 = 7,000 Hz

可以看出，如果不采用百分比的形式，而是直接设置各对象上的截止频率，那么组合结构的最终截止频率会变得很不直观。

Wwise 中 LPF 和 HPF 值对应的截止频率

下表是 Wwise 中 LPF/HPF 值（范围 0 到 100）与实际截止频率（单位 Hz）的大致对应关系。仅作为 Wwise 中 LPF 和 HPF 相对值体系的简介；实际截止频率可能稍有不同。

LPF 值:	截止频率 (Hz) :	HPF 值:
0	20,000	100
1	19,567	99
2	19,133	98
3	18,700	97
4	18,267	96
5	17,833	95

LPF 值:	截止频率 (Hz) :	HPF 值:
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
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

LPF 值:	截止频率 (Hz) :	HPF 值:
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
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

LPF 值:	截止频率 (Hz) :	HPF 值:
94	28	6
95	26	5
96	24	4
97	22	3
98	20	2
99	18	1
100	17	0

通过随机化属性值来改善播放

要使各对象每次在游戏中播放时听起来都不同，您可以为 Wwise 中的诸多属性值启用 Randomizer（随机化器）。每次播放时，Randomizer 会从您指定的范围内随机选择一个值，从而让对象属性值每次都稍有不同。

使用 Randomizer 时须执行以下操作：

- 启用/禁用 Randomizer
- 编辑 Randomizer 属性

启用/禁用 Randomizer

凡带有 Randomizer（随机化器）图标的属性都可以启用 Randomizer。默认情况下，Randomizer 处于禁用状态，但可以很容易地启用它并修改属性，制造需要的效果。

启用 Randomizer 的方法如下：

1. 对于带有 Randomizer 的属性，右键点击其文本框或滑杆。

快捷菜单将会显示。

2. 在菜单中，选择 **Enable Randomizer**（启用随机化器）。

Randomizer 窗口将弹出，而文本框中的 Randomizer 图标将变成黄色。



技巧

也可以双击 Randomizer 图标来打开随机化器窗口。

相关主题

- [通过随机化属性值来改善播放](#)
- [编辑 Randomizer 属性](#)

编辑 Randomizer 属性

对于特定属性，Randomizer 只会在一定范围内随机取值，因此它仅需要两个属性：Min（最小值）和 Max（最大值）。它们不是绝对值，而是作为偏置值与现有属性值叠加。

编辑 Randomizer 属性的方法如下：

1. 双击属性的 Randomizer 图标。

Randomizer 窗口将会打开。

2. 在 Min 文本框中，为属性键入您想要的最小可能值。Min 值将作为偏置值与现有属性值叠加。
3. 在 Max 文本框中，为属性键入您想要的最大可能值。最大值会作为偏移值作用于现有属性值。



相关主题

- [通过随机化属性值来改善播放](#)
- [启用/禁用 Randomizer](#)

构建 Actor-Mixer Hierarchy 的技巧和经验总结

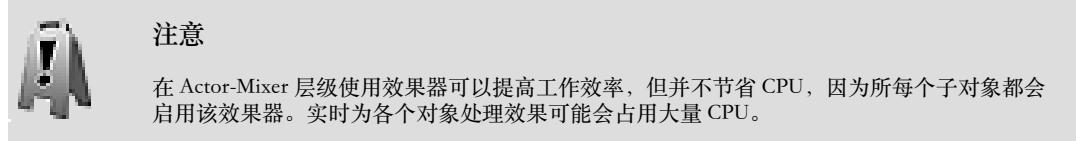
在 Wwise 中创建层级结构的方式非常灵活，在项目伊始就制定统一的策略可以节省后期的时间和精力。当然，处理音频工程的方法多种多样；了解这些概念可以让您的游戏获得最佳效果。

在 Actor-Mixer Hierarchy 中为对象分组

在构建层级结构之前，您需要思考组织对象的最佳方法，既要节省创作时间，又能很好地管理工程内存占用。关于如何对工程中的不同对象进行有效分组，以下是一些建议，请您根据实际需求选用。

Actor-Mixer（角色混音器）是节省内存和 CPU 的利器，因为其部分属性（例如定位和 RTPC）可以被所有子对象共享。所以组织对象的时候，可以考虑把一组对象置于 Actor-Mixer 下，这样可以：

- 共享属性设置，方便统一处理。
- 共享 Overrides（不沿用）属性，避免为多个子对象单独勾选 Override 和指定新值。



要优化内存占用，可以考虑把一组对象置于 Actor-Mixer 下，方便共享以下属性：

- 定位
- RTPC（实时参数控制）

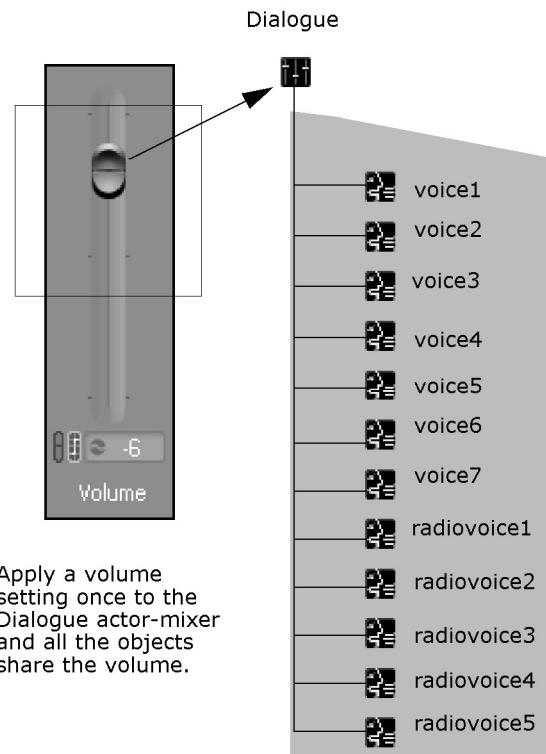
- States (状态)
- 随机化器

假设有一个 Actor-Mixer，其中含有 10 个声音，您希望将它们的 positioning（空间定位）设置为 3D。可以为这些声音逐个勾选 Override Parent（不沿用父选项），并设置为 3D。但更好的办法是，直接将 Actor-Mixer 的 positioning 属性设置为 3D，这样能节省 10 倍的运行时内存占用。即使只希望对部分声音进行声像摆位，将 Actor-Mixer 的定位属性设为 3D Emitter（3D 发声体）时仍可达到优化内存的效果。这时可为子对象忽略 Actor-Mixer 父对象的属性，并只将声像摆位应用于特定声音。因为，经过声像摆位的声音并不需要额外占用内存。

虽然 Actor-Mixer 通常是最好的选择，但在某些情况下，也可以选择采用 container（容器）属性，优化内存占用。例如，如果只需设置某个 container 内对象的空间定位，比如 Random Container（随机容器）中的脚步声，可以为该 container 设置定位属性来减少内存占用量，而无需设置父级 Actor-Mixer。但如果希望结构中的所有对象共享定位属性，则应在 Actor-Mixer 上设置。

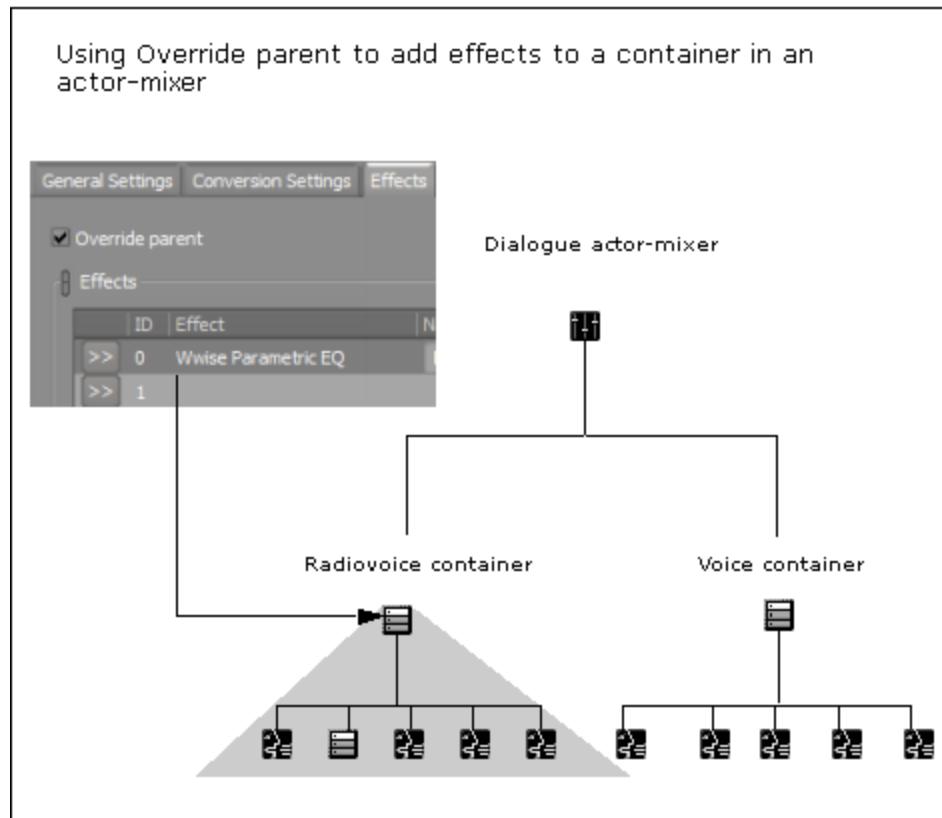
为对象分组——示例

下面介绍如何运用刚刚讨论的概念，有效地在 Actor-Mixer 中为对象分组。本例中，您会处理游戏的对话素材，其中一些是角色语音，另一些是对讲机语音。可以将整组素材放在名为 Dialogue（对话）的 Actor-Mixer 中，因为它希望所有对话素材共享音量等属性。



为 Actor-Mixer 设置好音量后，接下要为对讲机语音添加 Parametric EQ（参数均衡器）效果器。您可以逐个编辑对讲机语音对象，为每一个都 override 父级 Actor-

Mixer 设置，再逐个添加效果器。但还有更高效的办法：将所有对讲机语音文件放到 container 中，然后为 container 勾选 override 父级设置并添加效果器。



实时混音和对象属性

当您连接到游戏或游戏模拟器时，可以在 Wwise 中实时修改以下相对属性的值：

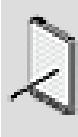
- Volume
- Pitch
- Low-Pass Filter（低通滤波器）
- Center %
- RTPC 值
- State 和 Switch 变换
- Triggers
- Attenuation 控件
- 部分音频和源效果器插件属性

备注

一般情况下，任何可映射至 Game Parameter 的属性都可在游戏中实时修改。

为了能够做到这一点，您需要加载您想在 Transport Control 或 Soundcaster 中修改其属性的对象。如果未加载对象，则更改将不会生效，因为声音引擎中未注册该对

象。对于无法加载到 Transport Control 中的 Actor-Mixer，可通过加载 Actor-Mixer 的子对象在声音引擎中注册其父对象。在注册对象后，对象将在您连接到游戏时保持注册状态。



备注

记住，如果您将对象固定在 Transport Control 中，则其它对象要到您取消固定第一个对象后才可加载。然而，如果您已将对象加载到 Soundcaster 中，则此对象将在声音引擎中注册。

相对属性和性能

在不同平台上，Wwise 的某些相对属性（如音高）可能会对性能造成影响。Wwise 中管理音高的机制是基于采样率的，改变声音的音高会增加 CPU 占用，因为必须对文件进行重新采样。

第9章 建立输出总线的结构

概述	182
定义总线的属性	186
构建总线层级结构——示例	190

概述

总线 (bus) 是一种用来对工程中对象进行分组的机制，目的是管理混音和最终声音输出，我们将在[Managing Output](#)中进行详细介绍。但是，在工程完成之前，清楚各总线的角色以及它们在整个工程结构中的工作方式至关重要。

就像需要为工程素材定义结构来以高屋建瓴的方式管理素材一样，您还需要为工程组织输出。通过在称为 Master-Mixer Hierarchy (主混音器层级结构) 的层级结构中对输出总线进行分组，您可以为工程中的通路定义相对属性、状态、RTPC 以及效果器。

您应花些时间来思考如何以最佳方式组织工程通路以优化混音过程。例如，您应通过将类似氛围音乐或炮火声的声音连接到相应总线来简化游戏音频的混音。

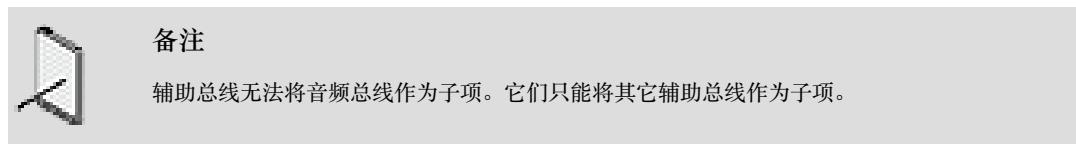
首先，您必须清楚 Master-Mixer Hierarchy 中的默认结构。它包含多条主总线，每条总线都有自己的角色和层级结构：

- [Master Audio Bus 层级结构](#)
- [Secondary Bus Hierarchy](#)

Master Audio Bus 层级结构

Master Audio Bus (主音频总线) 层级结构是输出总线的结构，工程中的声音和音乐连接到它上面。它包含三个不同层级的功能：

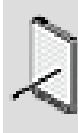
- **Master Audio Bus** —— 主音频总线。层级结构中的顶层元素，它决定了音频的最终输出。虽然允许重命名或者删除已创建的 Master Audio Bus，但您无法移动它们。您还可以在主音频总线上加效果器。
- **音频总线** —— 若干条可选总线，可分组到 Master Audio Bus 之下，以有助于组织和提供混音。您可重命名、移动和删除这些总线；还可以在总线上加效果器。
- **辅助总线** —— 若干条可选总线，可分组到任何辅助总线或音频总线之下。与音频总线类似，您可以重命名、移动、复制和删除辅助总线；您还可以对辅助总线应用效果器。工程中任何位置的声音对象都可发送到辅助总线以调整音量、声道配置、定位和 RTPC 以及应用效果器、状态或混音器插件。在辅助总线中无法做闪避、HDR 混音和声部调节。



以下 Wwise 界面图标有助于您轻松识别音频总线和辅助总线：

图标	代表
	音频总线
	辅助总线

默认情况下，Actor-Mixer Hierarchy 中的声音会被连通到 Master Audio Bus。但是，在您建立输出结构时，您可以系统地将对象连通到您创建的总线上去。



备注

您可以在 Default User Settings (默认用户设置) 对话框中为工程更改默认音频通路。有关这些默认设置的详细信息, 请参阅[定义工程的默认用户设置](#)。

Secondary Bus Hierarchy

Secondary Bus (二路输出) 属于音频总线层级结构, 用于为主输出 (电视或扬声器) 以外的其它输出信号内容进行混音。您可以为工程所需的各类型输出创建足够多的 Secondary Audio Bus (二路输出总线)。例如, 二路输出可以用作游戏手柄扬声器、聊天耳机或者 DVR-bypass 输出。和主输出结构一样, 二路输出结构也可以包含任意数量的子总线和辅助总线。

连通到 Master Secondary Bus 层级结构的声音将使用以下两种方法之一发送到二路输出:

- 直接将声音的 Output Bus (输出总线) 属性设置为 Secondary Bus 层级结构内的任何总线。对于通常只与一个二路输出相关联的声音, 这是首选方法。例如, 玩家开枪的声音, 网球拍击球的声音、PDA (掌上电脑) 声音、玩法反馈等。
- 将声音连线至 Master Audio Bus 层级结构内的任何总线, 并添加用户或游戏定义的辅助发送, 输出至 Secondary Bus 层级结构内的辅助总线。如果希望同一声音在电视和多个输出 (例如用手柄扬声器模拟间谍相机或广播) 中同时听到, 则这是首选方法。

总线层级只是混音时使用的层级结构。要将该混音结构与特定输出相关联, 请在对应的 Master Bus 上选择相应的 Audio Device Shareset。请参阅 Wwise 随附的[内置音频设备](#)列表。

必须要注意的是, 对于游戏中多个玩家可能同时使用的输出端 (如游戏控制器、耳机等), 相关混音层级结构会针对每个玩家进行复制。对于特定类型的输出, 工程定义的仅仅是特定的混音“处方”。实际信号输出到该结构的哪个副本则取决于听者和游戏对象的关系, 是由程序员设置的。

默认情况下, 创作期间设计工具会将所有声音输出到主声卡, 方便您预览声音。<不过, 您可以从 Audio (音频) 菜单打开 Audio Preferences (音频首选项) 窗口, 然后选择不同的硬件设备进行测试。

在 Master-Mixer 层级结构中添加总线

为了创建音频通路结构, 您可以将音频总线添加到 Master Audio Bus 上。您还可以创建其他主总线, 以便为其他输出进行混音。使用振动功能时, 还应为振动设备创建主总线, 并将所有声音 (振动实际只是低频声音) 输出至该层级结构的总线中。在已在主总线之下创建子总线之后, 您可以创建任意数目的父总线和子总线, 以建立通路结构。

在 Master-Mixer 层级结构中添加子总线的方法如下:

1. 在 Project Explorer 的 Audio 选项卡中, 右键点击您要为其创建新父总线子的总线。
2. 从菜单中, 选择以下选项之一:

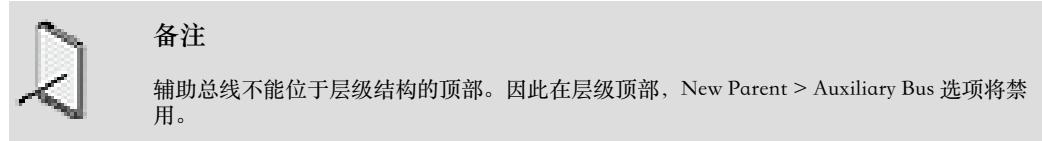
- New Child > Audio Bus (新建子项 > 音频总线)。
- New Child > Auxiliary Bus (新建子项 > 辅助总线)。

新子总线已添加到 Master-Mixer 层级结构中。

在 Master-Mixer 层级结构中创建父总线的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键点击您要为其创建新父总线的总线。
2. 从菜单中，选择以下选项之一：
 - New Parent > Audio Bus (新建父项 > 音频总线)。
 - New Parent > Auxiliary Bus (新建父项 > 辅助总线)。

新父总线于是添加到 Master-Mixer 层级结构中去了。



在 Master-Mixer 层级结构中添加新主总线的方法如下：

1. In the Audio tab in the Project Explorer, right-click the Default Work Unit in Master-Mixer Hierarchy.
2. 从菜单中，选择以下选项之一：
 - New Child > Audio Bus (新建子项 > 音频总线)。

新主总线已添加到 Master-Mixer 层级结构中。

复制和粘贴总线

添加总线的另一种方法是，复制现有总线并将其粘贴至所需位置。

在 Master-Mixer Hierarchy 中复制和粘贴音频总线的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键单击 Master-Mixer Hierarchy 中的相应总线。
2. 在菜单中，选择 Copy (复制) 或按下 Ctrl+C。
3. 选择要将复制的总线粘贴至 Master-Mixer Hierarchy 中的哪条总线。



4. 在菜单中，选择 Paste (粘贴) 或按下 Ctrl+V。

这时将创建一条与所复制总线的设置完全相同的新总线，并在其名称结尾后缀 _## 以示区分。

相关主题

- 移动 Master-Mixer 层级结构中的总线
- 删除 Master-Mixer 层级结构中的总线

移动 Master-Mixer 层级结构中的总线

在已添加总线之后，您可能会需要更改总线的位置以产生总线之间的不同关系。

在 Master-Mixer Hierarchy 中将总线移动到另一个位置的方法如下：

1. 将总线拖动到所需位置。

总线及其子总线都会移动到新位置。通过更改位置，移动后的总线现在将受其新父属性影响。



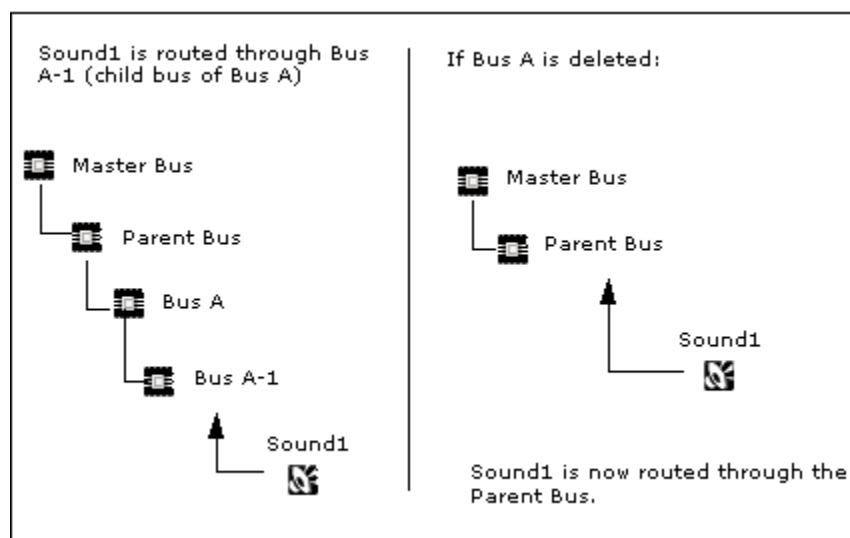
相关主题

- 在 Master-Mixer 层级结构中添加总线
- 删除 Master-Mixer 层级结构中的总线

删除 Master-Mixer 层级结构中的总线

如果您不小心创建了错误的总线或不再需要特定总线，则可以删除它。在删除一条总线时，其所有子总线也会被删除。连接到该总线上的对象现在已重新指派给该层级结构中的下一个父对象。对象重新指定通路后，其不沿用父对象（Override Parent）的属性保持不变。

下图说明了在删除某条总线后，会对该总线上传输的声音对象造成的影响。



删除 Master-Mixer 层级结构中的总线的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，选择要删除的总线，然后执行以下操作之一：
 - 按 Delete 键。
 - 在总线的快捷菜单中，选择 Delete Selection（删除选中项）。
2. 总线及其子总线可能会被删除，之前连接到删除的总线上的对象现在会连接到父总线上进行传输。

相关主题

- [在 Master-Mixer 层级结构中添加总线](#)
- [移动 Master-Mixer 层级结构中的总线](#)

定义总线的属性

您可以使用总线属性来对游戏中的音频或振动进行全局更改。在定义总线属性时，您可以执行以下操作：

- [通过现有声音生成振动](#)
- [定义总线的相对属性](#)
- [闪避信号](#)
- [将音乐替换为玩家自己的音乐](#)

由于总线是控制的最后一级，所以您所做的更改都将影响其下方整组对象。

与对象一样，您也可以向总线上添加效果器、使用 RTPC、指派状态以及设置高级属性。有关详细信息，请参阅以下几节：

- [将效果器作用于总线](#)
- [使用 Game Parameter 控制属性值](#)
- [将 State 指派给对象和总线](#)
- [第 12 章 管理优先级](#)

定义总线的相对属性

您可为层级结构中的各条总线定义相对属性。相对属性是累积的，也就是说，总线的属性值会累加到其下的子对象中。

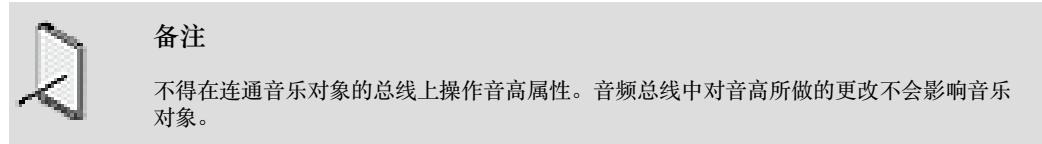
以下是您可以为总线修改的相对属性：

- Bus Volume —— 总线音量。用于直接调整总线的音量大小。
- Voice Volume —— 声部音量。用于调整总线内播放的音频对象的音量大小。
- Voice Pitch —— 声部音高。总线中播放的音频对象的播放速度。

您可以在 Multi Editor（多项编辑器）中同时为多条总线编辑这些属性。从 Project Explorer 的 Audio 选项卡中以及从快捷菜单中选择多条总线，选择 Show in Multi Editor。在连接到游戏的同时，您还可以实时微调这些属性。

定义总线的相对属性的方法如下：

1. 将一条总线加载到 Property Editor。
2. 键入一个值或拖动相应滑杆来为其中一个相对属性设置一个值：



相关主题

- [将效果器作用于总线](#)
- [闪避信号](#)
- [将音乐替换为玩家自己的音乐](#)
- [使用 Master Mixer Console 进行混音微调](#)

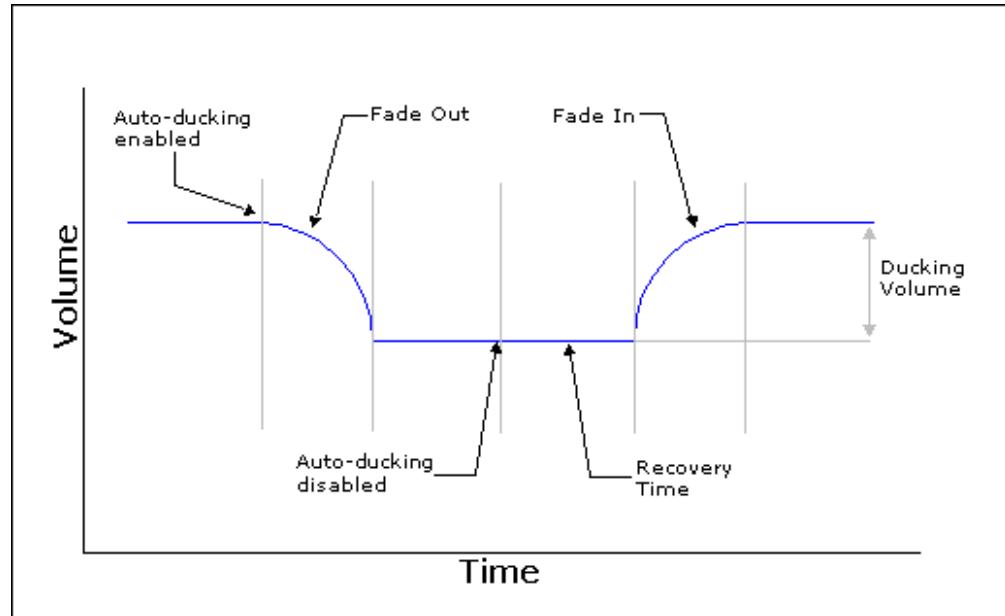
闪避信号

在游戏的不同时刻，您希望某些对象比其它对象更加突出。例如，您可能会需要在游戏角色说话时将音乐的音量降低。您可以使用闪避来决定一个信号相对于另一个信号的重要性。闪避使您能够自动降低通过一条总线的所有对象的音量电平，以便让另一条总线更为突出。

您可以通过修改以下属性和行为来控制信号闪避方式：

- Ducking volume (闪避音量)
- Fade out (淡出)
- Fade in (淡入)
- Curve shape (曲线形状)
- Recovery time (恢复时间)
- Maximum ducking volume (最大闪避音量)

下图说明了属性和行为对闪避的信号的影响方式和位置。



闪避某信号的方法如下：

1. 将一条总线加载到 Property Editor。
2. 在 Auto-ducking (自动闪避) 组框中，点击 Insert (插入)。

此时将会打开 Project Explorer - Browser 以及可被该总线要求闪避的总线列表。



备注

总线无法要求自己或其直接父总线闪避。

3. 从列表中，选择在当前总线收到信号时您要求闪避的总线。
4. 点击 OK。

所选总线已经添加到了闪避列表中。

5. 通过修改以下设置，定义闪避总线的属性和行为：
 - Volume：音量。在当前总线收到信号时，所选总线要降低的音量。
 - Fade Out：淡出。从原始音量淡出至闪避音量的时间。
 - Fade In：淡入。逐渐返回原始音量的时间。
 - Curve：曲线。用于定义淡出和淡入的曲线形状。
6. 在 Recovery time (恢复时间) 文本框中，指定您要当前总线信号结束到做出闪避的信号淡入开始的时间量。
7. 在 Maximum ducking volume (最大闪避音量) 文本框中，指定当前总线在被若干条总线要求闪避后可以衰减的最大量。



备注

闪避在 Auxiliary Bus 中不适用。

相关主题

- [将效果器作用于总线](#)
- [定义总线的相对属性](#)
- [将音乐替换为玩家自己的音乐](#)
- [使用 Master Mixer Console 进行混音微调](#)
- [使用旁链压缩](#)

将音乐替换为玩家自己的音乐

Xbox One™、PlayStation® 4、iOS 以及 Android 平台允许游戏玩家将游戏音乐替换为自己的音乐。对于所有平台，用户音乐播放时，您必须为要静音的所有总线启用 **Mute for Background Music** 选项。可以选择多条总线，这对音乐总线而言没有限制。

为平台的背景音乐选项指派总线：

1. 将音频总线加载到 Property Editor。
2. 选择 **Mute for background music** 选项。

现在，在用户通过游戏机上的音乐播放器播放音乐时，此总线将会静音。

平台专有的详细信息

平台不同，Mute for background music 的行为会稍稍有点不同。而且，在初始化声音引擎时，需要进行额外编程

- **Android:** Mute / Unmute（静音/取消静音）操作将仅在用户从音乐播放器应用程序切换至游戏时才会出现。也就是说，如果用户音乐自己结束播放了，那么就无需“Unmute”。
- **iOS:** 如果在声音引擎初始化设置中设置了 `AudioSession` 标志（flag）“`MixOther`”，则只有在用户从音乐播放器应用程序切换至游戏时才会做 Mute / Unmute 操作。也就是说，如果用户音乐自己结束播放了，那么就无需“Unmute”。在 iOS 8 和更高版本中，如果使用的是 `AVAudioSessionCategoryAmbient` 类别，则出现任何应用程序的音频中断时，游戏音乐都会发生静音和取消静音。
- **Xbox One 和 PS4:** 制造商添加的 DVR 功能允许玩家录制游戏并发布录制的游戏记录。这带来了一些法律问题，涉及到游戏音频中有受版权保护的音乐，和用户可替换音乐中受版权保护的音乐。虽然游戏工作室有权在其游戏中使用该音乐，但终端用户无权以任何形式发布该音乐。因此，TCR 要求不得录制背景音乐。此问题的经济高效的解决方案（CPU 占用）是将音乐和音乐外的游戏声音分开混音。使用 `Secondary Output`（二路输出）功能可完成该操作。

对于禁止录制的声音，XBox One 和 PS4 提供了专门接收它们的虚拟设备。您需要新建一条 Master Bus，用来为 DVR-bypass Audio Device 进行特殊混音。然后更改该新的总线上的 `Audio Device` 属性，使其指向 DVR-bypass 设备。随后声音可以正常输出到新的总线或子总线中。有关详细信息，请参阅[理解 Secondary Output](#)。



PS4 的重要备注

如果要使用 **Mute for Background Music** 功能，必须在代码中添加二路输出（使用 `AddOutput`）。如果不使用此输出，用户音乐开始播放时 PS4 系统将不会发出通知。

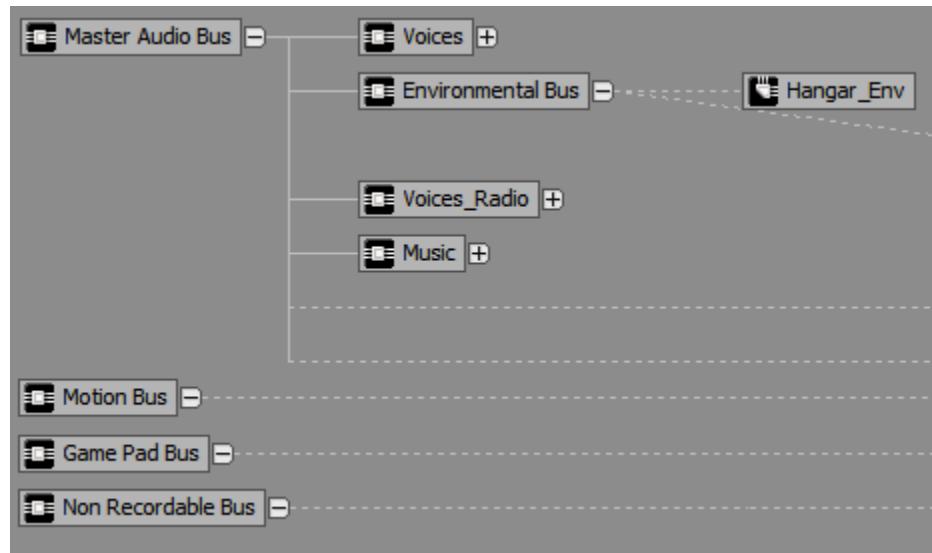
相关主题

- [将效果器作用于总线](#)
- [定义总线的相对属性](#)
- [闪避信号](#)
- [使用 Master Mixer Console 进行混音微调](#)
- [指定对象的输出连线](#)
- [内置音频设备](#)
- [集成 Secondary Outputs](#)

构建总线层级结构 —— 示例

Wwise 为构建总线层级结构提供了很大的灵活性。也就是说，组织工程的声音结构时条条大路通罗马。尽管如此，以下简单示例仍为您提供了一些开发总线结构的常见做法，无论工程的素材和要求的具体组合形式。

在以下 Schematic View 屏幕截图中，我们看到工程中的 Master Audio Bus 下有四个 Audio Bus，同时还有三个 Secondary Master Bus。



对于 Master Audio Bus 结构，我们有以下总线：

- **Environmental Bus** -- 环境总线。此总线会基于不同环境因素（例如混响）对玩家可能听到的各种声音进行分组，例如玩家或其它角色走在砾石地板、木地板或水泥地板上的脚步声。
- **Music** -- 音乐。此总线会对所有音乐进行分组，无论这些音乐是在游戏场景中播放的，还是在游戏外操作 UI 菜单时播放的。
- **Voices** -- 语音。此总线会对大多数角色对话进行分组。
- **Voices_Radio** -- 语音_无线电。这款游戏中有大量的对话，还有很多语音需要特殊设置来表现无线电中发出的声音效果，所以我们还在 Voices 这条总线之外添加了这条总线。组织这条总线的方式可能有很多，一种方式是将其设置为 Voices 总线

的子总线。但是，尽管这两条总线在概念上类似，但这条总线所需的声音输出以及隐含的混音方式告诉我们，更简单的做法是将其直接定义为 Master Audio Bus 下的单独总线。

为了更方便地调节位于大型飞机库中的声音，我们还添加了一条辅助总线：Hangar_Env。这样做后，当游戏场景移入飞机库时，我们可以将声音（手动或通过游戏调用）发送至该总线，我们可在此处运用玩家在此类环境中可能会听到的开放式回声的混响。

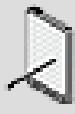
我们还有另外几条主总线，每条都作为一个二路输出的信号源：

- **Motion Bus - 振动总线。**此总线用于接收所有 Motion 信号（也称为 rumble），并将它们输出到相应的控制器。本例中不存在子总线，尽管对于振动也允许进行更复杂的混音。
- **Non Recordable Bus - 非录音总线。**系统允许有一条不会被 DVR (Digital Video Recorder) 录音的独立输出。希望防止玩家内录和发布版权音乐的话，这是理想的解决方案。对于防止玩家内录和发布版权音乐，这是理想的解决方案。
- **Game Pad Bus - 游戏控制器。**控制器的扬声器可输出玩家将更直接地听到或至少是能明显区分的声音。它非常适合像角色头部撞墙发出的沉闷的声音这样的情况。



技巧

建议不要同时将同一声音发送到二路总线和音频总线！每个系统都有特定的延迟量。即使是几毫秒的差异，同一声音的两条输出间的延迟也将会产生明显的不和谐。



备注

您可以使用声音对象（包括音乐文件）生成振动数据。有关使用现有声音对象生成振动的详细信息，请参阅[通过现有声音生成振动](#)。



部分 III. 使用声音和振动来提升游戏体验



10. 定义对象播放行为	194
概述	195
定义对象的播放行为	195
定义 Random/Sequency Container 的播放行为	201
定义 Switch Container 的内容和行为	211
定义 Blend Container 的内容和行为	216
对象播放相关的技巧和经验总结	225
11. 定义定位	227
定义定位概述	228
理解 Wwise 中的定位	228
使用 Speaker Panning	231
使用 3D 空间化对象	232
Speaker Panning 和 3D Spatialization 交叉淡变	235
为总线应用定位	235
3D 定位图解	236
应用基于距离的衰减	247
使用动画路径定义空间定位	259
将音频信号传送到中置扬声器	268
定位技巧和经验总结	269
理解声道配置	275
声像摆位规则：扬声器对耳机	277
使用 Ambisonics	278
12. 管理优先级	284
概述	285
了解 Wwise 如何为对象排列优先级	286
限制对象的播放实例	289
定义播放优先级	292
管理音量较低的对象	293
优先级相关技巧和经验总结	295
13. 管理效果器	297
概述	298
使用效果器	298
使用效果器实现环境音效	306
效果器技巧和经验总结	308
14. 管理 Motion	310
概述	311
为游戏创建振动	313
为振动来构建输出结构	321
振动相关技巧和经验总结	322

第 10 章 定义对象播放行为

概述	195
定义对象的播放行为	195
定义 Random/Sequency Container 的播放行为	201
定义 Switch Container 的内容和行为	211
定义 Blend Container 的内容和行为	216
对象播放相关的技巧和经验总结	225

概述

游戏内出现的各种不同情况会要求声音以不同形式播放。为了适应这些不同的场景，用户可以通过 Wwise 来设置 Actor-Mixer Hierarchy 中各个对象的播放行为。但为了增强播放的灵活性和控制度，您还可以通过 Wwise 将对象编组到不同类型的容器内：

- [创建随机容器](#)
- [创建序列容器](#)
- [定义 Switch Container 的内容和行为](#)
- [创建混合容器](#)

所选的容器类型将决定容器内对象播放的方式。例如，Random Container 随机播放其内容，Sequence Container 按播放列表播放其内容，Switch Container 根据游戏内当前切换开关或状态播放其内容，Blend Container 则使用混合方式或基于 RTPC 来同时播放容器内所有对象。

使用不同容器的组合，您可以高效、创意地为游戏管理各种情形下如何播放声音，从而提升游戏的体验。

定义对象的播放行为

工程内的每个对象都具有一组特定行为。这些行为决定每次调用对象时的播放次数，以及是在内存中存储对象，还是直接从 DVD、CD 或硬盘进行流播放。

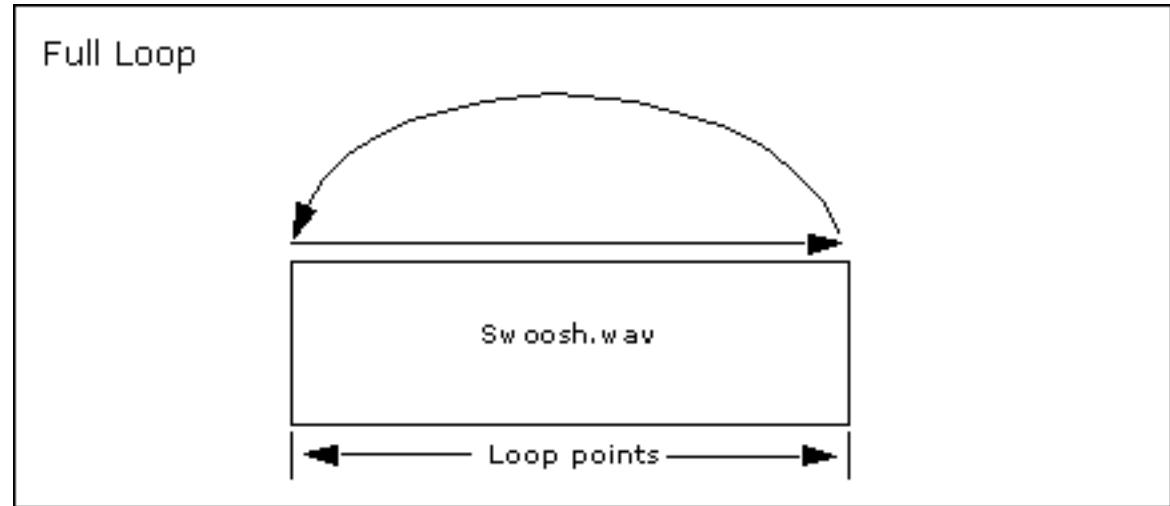
您可以为对象定义以下行为：

- [创建循环](#)
- [用流来播放媒体](#)

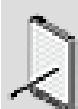
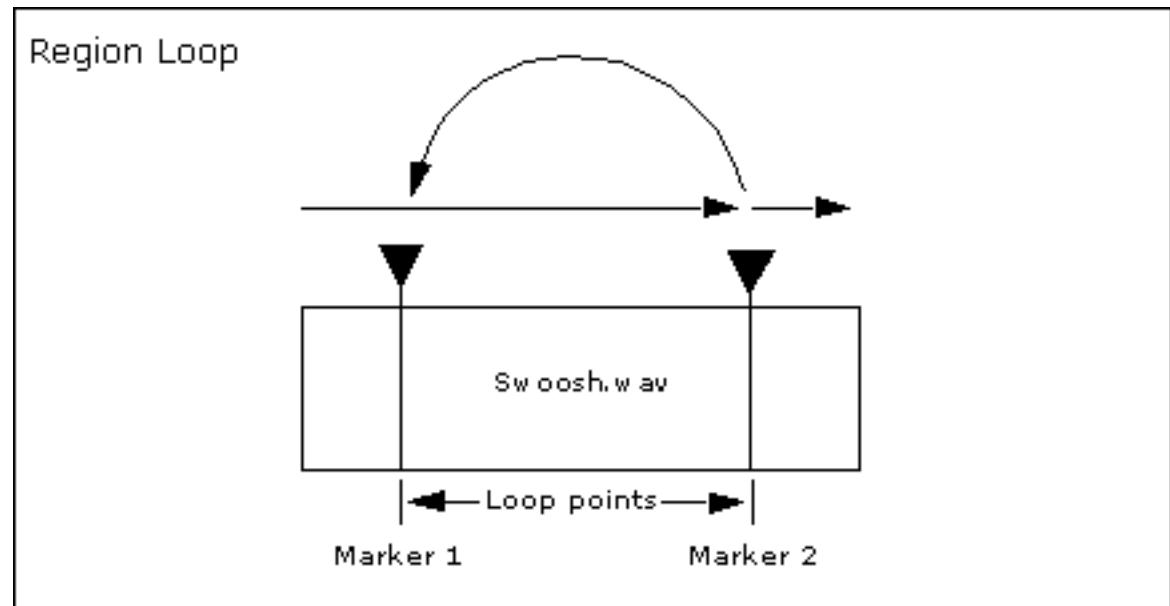
创建循环

默认情况下，对象从头至尾播放一次。但您可能会需要多次播放对象。这种情况下可以创建循环。当您决定循环某个对象时，必须定义循环的次数。可以选择无限循环对象，也可以指定循环次数。

默认情况下将循环整个对象，但如果音频文件中添加了 loop region（循环范围）标记，则可以循环对象的一部分。在完整循环中，整个对象将从头至尾播放。播放到对象结尾后，会返回至开头继续播放。它会重复播放，直至完成指定的次数。



在特定范围循环中，仅循环标记范围之内的部分。对象从头播放，直至循环范围的结尾。然后返回至标记范围的开头，并重复播放循环范围，直至完成指定的循环次数。循环结束后，将播放对象的最后部分。如果是无限循环，则永远不会播放对象的最后部分。Wwise 仅支持一个循环范围，因此请确保在音频文件中仅定义了一个范围。



备注

范围标记（或 Audition 中的 start/end 采样）必须在第三方应用程序中创建，如 Adobe® Audition® 和 Sound Forge®。也可以使用其它第三方应用程序，但这些应用程序不在 Wwise 官方支持范围内。

循环播放声音的方法如下：

1. 将对象加载到 Property Editor 中。
2. 选择 Loop 选项。

Loop 参数变为可用。

3. 选择以下选项之一：

Infinite 无限重复，整个对象或文件中定义的循环范围将无限重复。

No. of Loops 次数，指定整个对象或特定循环范围的播放次数。

4. 如果您选择 No. of Loops 选项，请输入对象或特定循环范围的播放次数。

相关主题

- [定义对象的播放行为](#)
- [用流来播放媒体](#)

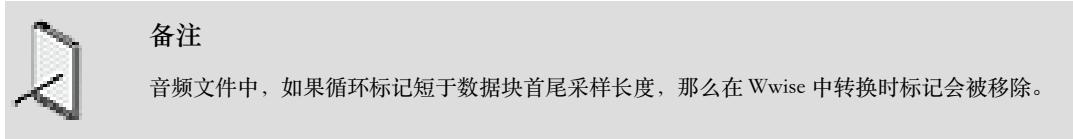
循环播放压缩音频文件

由于压缩文件格式（如 ADPCM）要求文件长度和范围标记与数据块的边界采样对齐，因此转码过程中会添加零本采样来补齐。循环声音时，这些额外的零采样会导致问题。为了避免问题，Wwise 会提示点用户重新转换循环的声音，以确保：

- 补零不会发生在文件结尾。
- 循环范围标记会与数据块边界采样对齐。

Wwise 在重新转码过程中使用变调，以确保这些文件符合压缩格式的要求。循环会精确到采样点，且文件的采样率不会改变。

您可能还会注意到，循环的 XMA 文件在 Windows 和 Xbox One 中听起来有些不同。发生这种情况是由于 Xbox One 使用硬件解码文件，在循环点之间可应用短淡变。但在 PC 中，会使用 XMA 编码器生成单独的 WAV 文件。



使用 Source Editor

Source Editor 可用于编辑以下类型的对象：

- 音频源（WAV 或 AMB 文件）
- 源插件（比如 Wwise Silence、Wwise Sine 和 Wwise Tone Generator 等）

相关主题

- [编辑音频源](#)

编辑音频源

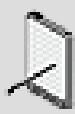
在 Source Editor（源编辑器）中，可通过以下操作对源文件（WAV/AMB）进行非破坏性处理：

- 裁剪源的开头或结尾
- 应用淡入或淡出
- 设置循环点
- 设置交叉淡变，来获得平滑的循环
- 添加或编辑文件标记

所有这些操作都会离线处理，并非在游戏中实时进行。它们会被存储在工程的 Work Unit（工作单元）中。这些操作是可逆的，允许撤消（依次选择 Edit > Undo 或按下 Ctrl+Z）。

修剪音频源的内容

修剪 WAV 文件中的内容可以节省 WAV 数据的空间。可以用于移除 WAV 文件开头或结尾的无声段落。修剪内容将发生在源素材的转码过程中，属于非破坏性的操作。



备注

编辑功能（如修剪、循环和淡变）不适用于交互音乐层级中的 Audio Source 对象，因为 Music Segment 和 Music Clip 对象提供了这些功能。

修剪源的方法如下：

1. 将 Trim Start（位于波形左下角的蓝色方形手柄）拖向右侧。
2. 将 Trim End（位于波形左下角的蓝色方形手柄）拖向左侧。

编辑淡入和淡出

淡入和淡出可通过拖动淡变手柄（波形上方的蓝色三角）在 Source Editor 中创建。淡变将仅对转换的文件生效，而且在播放音频源的过程中不会产生额外的处理负担。

创建淡入和淡出

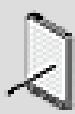
1. 将 Fade-in（蓝色三角手柄，位于波形的左上角）手柄拖向右侧。
2. 将 Fade-out（蓝色三角手柄，位于波形的右上角）拖向左侧。

循环音频源

当声音对象启用 Loop 属性时，其包含的音频源将使用循环点。如果 WAV 文件存在循环点，则默认情况下会使用这些循环点，除非在编辑器中移动这些循环点。一旦在编辑器中修改了这些循环点，文件中的原始循环点将不再起作用。

循环点仅在以下条件下使用：

- 已启用 Override file loop point
- 父声音对象已设置为循环



备注

在 Source Editor 中设置或修改循环点不会修改原始 WAV 文件。这是非破坏性操作。这些循环点存储在工程中。

编辑交叉淡变时长

为了在循环过程中得到更平滑的过渡，避免听到 click 声，可以在音频源中定义交叉淡变时长。



备注

为避免循环过渡时出现爆音，最好将循环点放置在 PCM 数据的零交叉位置。

调整循环点和使用交叉淡变的方法如下：

1. 将 Loop Start（绿色裁剪点标记，位于波形的左上角）拖至右侧。
2. 将 Loop End（红色裁剪点标记，位于波形的右上角）拖至左侧。
3. 定义交叉淡变时长。

添加和编辑文件标记

音频源文件可能包含标记，有时会被用作同步点或提示点。标记具有与文件位置绑定的唯一 ID。Wwise 可在 Source Editor 的坐标图中使用这些标记。另外，还可编辑这些标记或另外添加标记，但不会影响到原始文件。

您可以单击所需标记并将其拖放到要在源文件中标识的位置。比如，以此来显示文件哪里存在问题。或者提示即将发生的重要事件。比方说在对白中，可利用其来区分不同角色说话的时间点。所以，您可以根据具体需要自由地设置标记。

在 Source Editor 的侧面板中，可选择 Marker Input Mode：

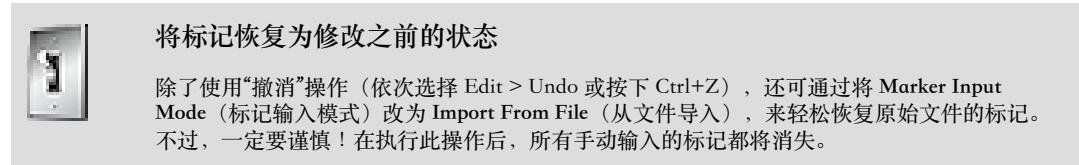
- **Use File Markers**（使用文件标记）：使用源音频文件的标记。
- **Use Markers From Transients**（使用来自瞬态的标记）：通过自动检测瞬态起始点来放置标记。在移动 Marker Detection Sensitivity（标记检测灵敏度）滑杆时，才会显示这些标记。灵敏度越高，产生的标记越多。调整滑杆以使检测到的瞬态起始点尽可能靠近所需位置。然后，手动编辑这些标记。一旦编辑标记，标记模式就切换为 Manual Markers。
- **Manual Markers**（手动标记）：使用用户设置的标记。在选中 Manual Markers 时，并不会改变先前已有的标记位置。无论是文件标记还是来自瞬态的标记，用户都可以随时直接编辑。一旦进行编辑，Markers 选择就会设为 Manual Markers。

在 Source Editor 的坐标图内，可使用四个快捷菜单选项来快速编辑标记。

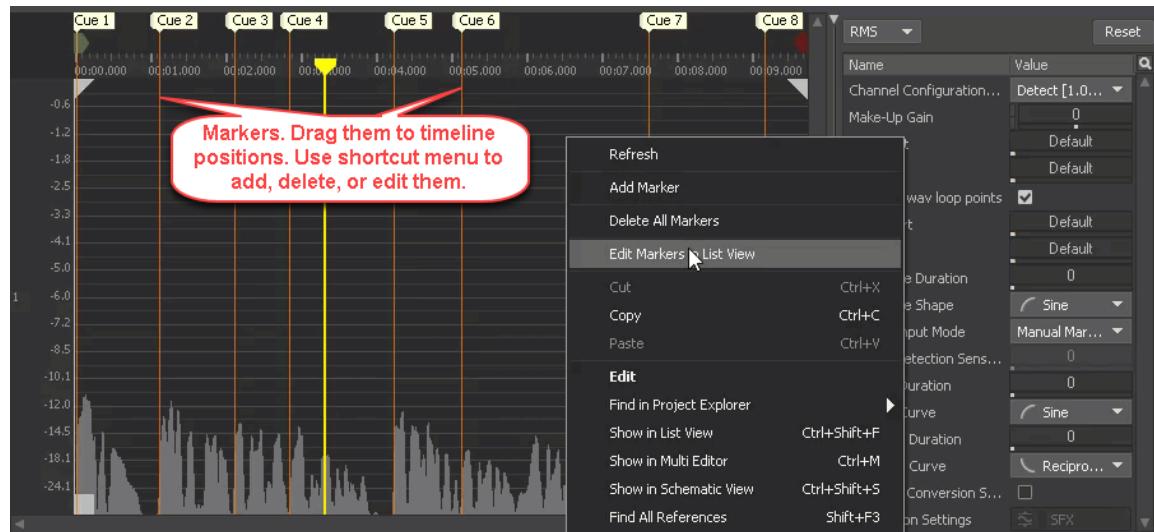
标记专用的快捷菜单选项：

- **Add Marker**（添加标记）：在 Source Editor 坐标图内的现有标记之外打开快捷菜单时，将显示此选项。它用于在坐标图时间线的指定时间点创建新的无标签的标记。
- **Delete All Markers**（删除所有标记）：在 Source Editor 坐标图内任意位置打开快捷菜单时，都会显示此选项。它用于移除来自源文件的所有标记。不过，这并不会影响原始文件标记。
- **Delete Marker**（删除标记）：只有在特定标记图柄位置打开快捷菜单时，才会显示此选项。它用于移除特定标记。它也不会影响原始文件标记。

- **Edit Marker Label...**（编辑标记标签...）：只有在特定标记图柄位置打开快捷菜单时，才会显示此选项。它会显示 Edit Label（编辑标签）对话框，其中显示所选标记的当前标签，便于根据需要更改标签。
- **Edit Markers in List View**（在列表视图中编辑标记）：在 Source Editor 坐标图内任意位置打开快捷菜单时，都会显示此选项。它会显示 List View（列表视图），其中有两个标记专用的列，用来查看和编辑所有源文件中的标记（包括 Wwise 中创建的标记）。



下图简要显示了 Wwise Source Editor 内的一些常用标记选项。



解决音频源完好度问题

使用外部编辑器编辑音频源属性或编辑 WAV 文件后，对音频源已做的设置可能变为不可用，或不完全可用。

完好度问题的情况示例：

1. 在 Wwise 中导入 WAV 文件、创建声音和音频源对象。
2. 在 Source Editor 中，修剪 WAV 文件的结尾。
3. 在外部 WAV 编辑器中，将WAV 文件的结尾区域删除，删除区域比修剪区域更长。

在 Wwise 中，修剪操作将针对不再存在的部分 WAV 文件执行。这种情况就属于音频源完好度问题。

诊断音频源完好度问题

在 Wwise 中的以下地方可以发现音频源完好度出现了问题：

- 在 Source Editor 中，带有黄色三角和叹号
- 在 Integrity Report（完好度报告）中，运行 Audio Source（音频源）测试时
- 生成 SoundBanks（声音包）时音频源转码执行期间

修复音频源完好度问题

要修复音频源完好度问题，请执行以下任一操作：

- 通过处理不可用的操作，手动修复音频源。
- 点击 Audio Source Editor 中的黄色三角。
- （在列有音频源的位置）右键点击 Audio Source 或声音对象，然后选择 Repair Audio Source Integrity Issues。

用流来播放媒体

您可以决定哪些对象加载至内存中播放，哪些对象从硬盘、CD 或 DVD 进行流播放。当从磁盘或硬盘流播放媒体时，您还可以选择避免播放延迟，方法是创建一个小缓冲区来补偿提取剩余文件所需的时间。您可以指定缓冲区的大小，以满足不同媒体源（如硬盘、CD 和 DVD）的要求。与流播放文件余下的部分不同，这个小缓冲区实际上存储在 SoundBank 内，并与 SoundBank 一起在游戏中适当的时刻加载至内存中。



备注

无论是否选择了流播放选项，在 Wwise 中播放音频时总是会使用流播放。因此仅当生成 SoundBank（声音包），且从远程平台或游戏播放时，才会应用流播放设置。

流播放声音的方法如下：

1. 将对象加载到 Property Editor 中。
2. 选择 Stream 选项。

各 Stream 参数变为可用。

3. 选择 Zero Latency 选项，实现从触发对象到实际播放之间零延迟。为实现零延迟，必须将数据文件开头的特定部分存储在内存中，以补偿从媒体提取剩余文件所需的时间。
4. 在 Prefetch length 文本框中，输入毫秒数，即需要存储在内存中的文件数据长度。

相关主题

- [定义对象的播放行为](#)
- [创建循环](#)

定义 Random/Sequency Container 的播放行为

随机容器（Random Container）和序列容器（Sequence Container）为您提供游戏中声音播放的多种组织形式。随机容器可随机播放一组对象，让您可以创建有趣的声景并增强玩家体验；而序列容器用来以指定顺序持续播放对象。

使用随机和序列容器时，您可以执行以下任务：

- [创建随机容器](#)
- [创建序列容器](#)
- [创建播放列表](#)

创建随机容器

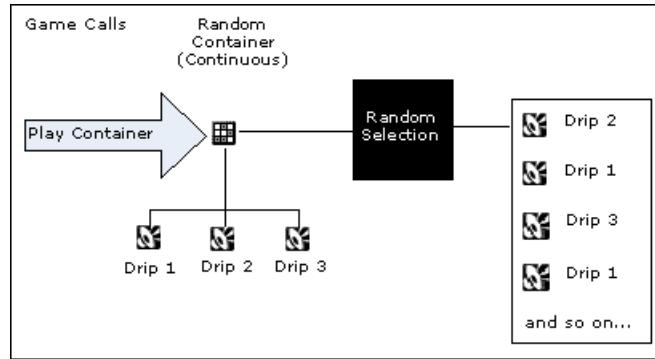
对于所有需要随机播放一组对象的情况，都可以使用随机容器。例如，您可能会需要使用随机容器在格斗游戏中播放所有攻击、击中和反应声音。

在 Wwise 中，随机意味着既可以进行标准随机选择（即在播放时，容器中的各个对象被选中的几率均相等），也可以进行洗牌选择（即被播放完的对象将从选择池中移除）。

您也可以为随机容器内的每个对象指派一个权重值。该权重值可以增加或减小对象被选择播放的概率。为对象设置的权重值是相对于容器内所有对象权重总和来计算的相对值。这就是说，为各个对象指派的权重值代表其被选择播放的几率占总数的比率。例如，如果容器内有两个对象，权重值分别为 1 和 100，则第一个对象将有 1/101 的播放几率，而第二个对象将有 100/101 的几播放率。

使用随机容器——示例

假如游戏的其中一个环境为洞穴。您希望背景中有水滴声，来营造洞穴环境的氛围。在本例中，可以创建一个随机容器，将所有不同的水滴声编组。由于您希望当角色位于洞穴中时，持续播放水滴声，因此可以将容器的播放模式设置为 **Continuous** 并无限循环。通过随机播放有限数量的声音，可以为场景增加真实感。



创建随机容器的方法如下：

1. 在 Project Explorer 中，在 Actor-Mixer Hierarchy 中右键点击以下任一对象：
 - Work unit (工作单元)
 - Virtual Folder (虚拟文件夹)
 - Actor-Mixer (角色混音器)
 - Blend Container (混合容器)
 - Random Container (随机容器)
 - Sequence Container (序列容器)

- Switch Container (切换容器)
2. 从快捷菜单中，选择 New Child > Random Container。
- 此时将创建一个新的随机容器，并会在 Actor-Mixer Hierarchy 中高亮显示。
3. 输入随机容器的名称，然后按下 Enter。
 4. 双击新的随机容器，将打开其相应的属性和内容编辑器。
 5. 将各对象从 Project Explorer 拖至 Contents Editor 中，来填充随机容器。
 6. 在 Random 分组框中，选择以下任一选项：
 - Standard (标准)，始终保持对象池完整。播放某个对象后，该对象不会从可播放的对象列表中移除，因此可重复播放。
 - Shuffle (洗牌)，播放对象后，这些对象会从对象池中移除。该选项可避免重复，直至所有对象均已播放。
 7. 为避免对象被连续重复播放，请选择 Avoid repeating last x played 选项。

您选择的是 Standard 还是 Shuffle 模式将影响该选项的行为。

在 Standard 模式中，完全随机选择播放对象，但最后播放的 x 个对象将从列表中弃用。

在 Shuffle 模式中，重置列表时，最后播放的 x 个对象将从列表中排除。

8. 在 Avoid repeating last x played 文本框中，输入重复播放某个对象前须播放其它对象的数量。
9. 在 Contents Editor 中，为容器内的各个对象指派权重值。

该权重可以让某些对象优先于其它对象播放。为各个对象指派的权重值代表其被选择播放的几率占总数的比率。

相关主题

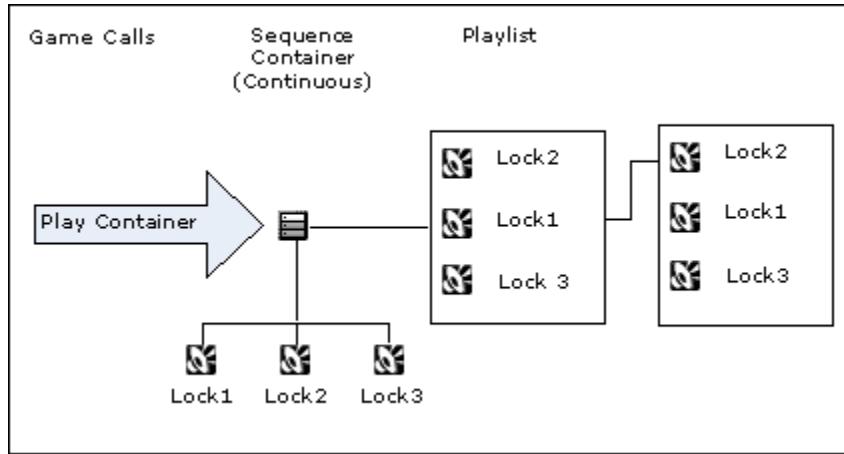
- [定义容器的范围](#)
- [定义容器内对象的播放方式](#)
- [创建序列容器](#)

创建序列容器

对于所有需要按特定顺序播放一组对象的情况，都可以使用序列容器。序列容器会根据特定播放列表来播放容器内的对象。例如，您可以使用序列容器来表现角色的对话。您可以指定对话的顺序，防止角色先说出“再见”后才说出“你好”。

使用序列容器——示例

假设您正在研发第一人称射击游戏。在游戏的某个环节，玩家必须按下按钮才可打开一扇装有很多解锁机制的大铁门。此时，您可以将所有解锁声音编组至一个序列容器。之后可以创建一个播放列表，按逻辑顺序排列这些声音。您可以将容器的播放模式设置为 Continuous，这样在解锁大门时，便会连续播放这些解锁声。



创建序列容器的方法如下：

1. 在 Project Explorer 中，在 Actor-Mixer Hierarchy 中右键点击以下任一对象：
 - Work unit (工作单元)
 - Virtual Folder (虚拟文件夹)
 - Actor-Mixer (角色混音器)
 - Blend Container (混合容器)
 - Random Container (随机容器)
 - Sequence Container (序列容器)
 - Switch Container (切换容器)
2. 从快捷菜单中，选择 New Child > Sequence Container。

此时将创建一个新的序列容器，并会在 Actor-Mixer Hierarchy 中高亮显示。

3. 输入序列容器的名称，然后按下 Enter。
4. 双击新的序列容器，以打开其相应的属性和内容编辑器。
5. 将各对象从 Project Explorer 拖至 Contents Editor 中，来填充序列容器。
6. 要定义 At end of playlist 行为，请选择以下任一选项：
 - Restart 从头开始，播放列表中的最后一个对象播放完后，会再次按其原始顺序从头至尾播放列表。
 - Play in reverse order 倒序播放，播放列表中的最后一个对象播放完后，会按照倒序从最后一个到第一个播放列表。



技巧

通过在 Project Explorer 中选择对象，然后右键点击并选择 New Parent > Sequence Container，您也可以为多个对象和其它容器一起创建父级序列容器。

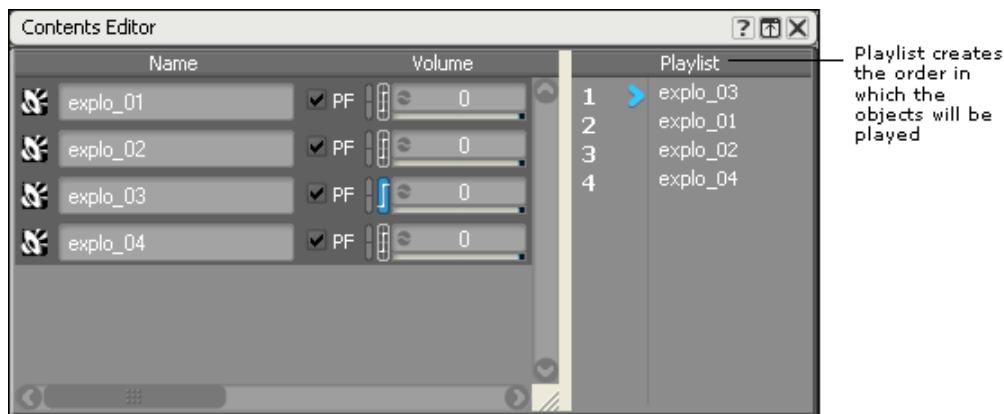
相关主题

- [创建播放列表](#)
- [定义容器内对象的播放方式](#)
- [定义容器的范围](#)

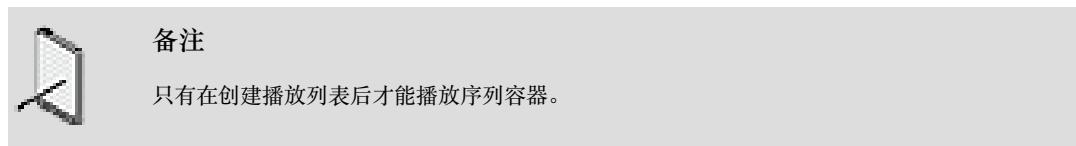
- **创建随机容器**

创建播放列表

创建序列容器时，Contents Editor 中会增加一个 Playlist 面板。您可以创建一个播放列表，来按特定顺序播放容器内的对象。



您也可以测试和微调播放列表，添加、移除和给对象重新排序，直至您满意。可以在创建过程中随时播放该播放列表。有关播放播放列表的详情，请参阅 [播放播放列表](#)。



在播放列表中添加和移除对象

通过在 Contents Editor 的 Playlist 面板中添加和移除对象和其它容器，可以轻松创建播放列表。您也可以选择多个对象，然后同时添加或移除它们，来更快地创建列表。

大多数情况下，您将使用序列容器中的对象创建播放列表。但您也可以从 Project Explorer 的 Audio 选项卡中，将对象直接拖至播放列表。这些对象将被从当前位置移到序列容器中。如果您要创建对象的副本，而不是移动它，则可以按住 Ctrl 键同时将该对象从 Project Explorer 的 Audio 选项卡中拖拽至播放列表。

在播放列表中添加/移除对象的方法如下：

1. 将序列容器加载至 Property Editor。

容器内的对象将显示在 Contents Editor 中。

2. 要向播放列表添加对象，请将该对象从 Contents 面板拖至 Playlist 面板中。

对象即被添加至播放列表。

3. 要从播放列表中移除对象，请在 Playlist 面板中点击要移除的对象。
4. 按 Delete 键。

对象将被从播放列表中移除，但仍保留在 Contents 面板中。

相关主题

- [创建播放列表](#)
- [为播放列表中的对象重新排序](#)
- [播放播放列表](#)

为播放列表中的对象重新排序

创建播放列表时，您可以为这些对象重新排序，以查看顺序不同时它们听起来效果如何。如果您的播放列表内包含容器，则无法为所含容器内的对象重新排序。

为播放列表中的对象重新排序的方法如下：

1. 在 Contents Editor 的 Playlist 面板中，选择要移动的多个对象。
2. 将它们拖拽至播放列表中的新位置。

这些对象将被移至新位置。

相关主题

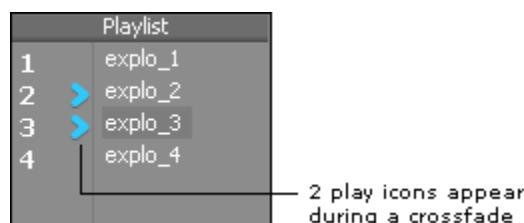
- [创建播放列表](#)
- [在播放列表中添加和移除对象](#)
- [播放播放列表](#)

播放播放列表

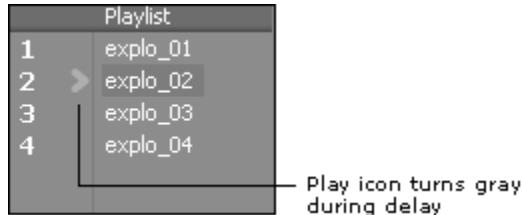
播放序列容器前，您必须创建一个播放列表。蓝色播放图标表明播放列表中的哪个对象正在播放。



如果您在对象过渡时使用了交叉淡变，那么在淡变期间，播放列表中会显示两个播放图标。



如果您在对象过渡时使用了延迟，那么在延迟期间，播放图标会从蓝色变为灰色。



播放播放列表的方法如下：

1. 将序列容器加载至 Property Editor。
2. 在 Transport Control 中，点击 Play 图标。

Wwise 将按指定顺序播放播放列表。

相关主题

- [创建播放列表](#)
- [在播放列表中添加和移除对象](#)
- [为播放列表中的对象重新排序](#)

定义容器内对象的播放方式

由于随机和序列容器都包含多个对象，因此您必须指定播放模式。Wwise 提供以下播放模式：

- Step —— 每次播放容器时，仅播放容器内的一个对象。
- Continuous —— 每次播放容器时，会完整播放容器内的所有对象。

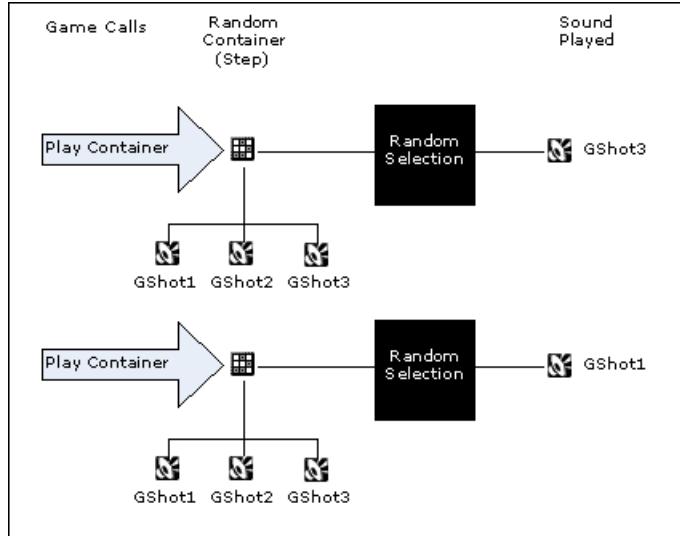
如果将播放模式设置为 Continuous，则您也可以选择循环这些对象，并为容器内的各对象之间创建过渡。



播放容器内的一个对象

在某些情况下，您会希望在每次调用时，仅播放容器内的一个对象。例如，每次开枪时，可能仅需播放一个声音，或每次角色在讲话时，仅希望其讲一句话。这时，您可以将容器的播放模式设置为 Step。

下图显示了手枪的声音以 Step 模式被组织到随机容器内后，开枪时将发生什么。



播放容器内的一个对象：

1. 将随机或序列容器加载至 Property Editor。
2. 在 Play Mode 分组框中，选择 Step 选项。

每次播放容器时，将仅播放一个对象。

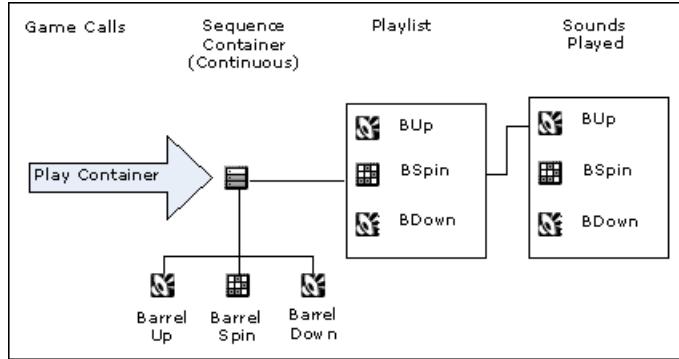
相关主题

- [播放容器内的所有对象](#)
- [定义 Random/Sequency Container 的播放行为](#)
- [创建随机容器](#)
- [创建序列容器](#)
- [创建播放列表](#)
- [定义容器的范围](#)

播放容器内的所有对象

在其它情况下，您会希望每次调用时，会播放容器内的所有对象。例如，当游戏内某些枪支开火时，必须按序播放一系列声音。这时，您可以将容器的播放模式设置为 Continuous。

下图显示了当特定枪支开火时，Wwise 如何使用 Continuous 模式的序列容器来播放声音序列。



在 Continuous 模式下将播放容器内的所有对象，这意味着您可以添加循环，并在对象间应用各种过渡。



备注

当随机容器处于 Continuous 模式时，列表播放完毕前某些对象可能会重复数次。这是由于对容器内的各个对象都应用了权重。

播放容器内的所有对象的方法如下：

1. 将随机或序列容器加载至 Property Editor。
2. 在 Play Mode 分组框中，选择 Continuous 选项。

Continuous 参数变为可用。

3. 对于序列容器，如果要在每次播放容器时将返回至播放列表开头，则可以选择 Always reset playlist 选项。如果您未勾选该选项，则容器将从上次停止处继续播放，更确切地说是从播放列表中下一个对象的起始处播放。
4. 选择 Loop 选项可以循环容器的全部内容。

Loop 参数变为可用。

5. 选择以下选项之一：
 - Infinite，指定容器将无限重复。
 - No. of Loops，指定容器将播放特定次数。
6. 如果您选择 No. of Loops 选项，请输入容器将被播放的次数。
7. 选择 Transitions 选项，可以在播放列表中的对象间应用过渡。

Transition 参数变为可用。

8. 从 Type 列表中，选择以下任一选项：
 - Xfade (amp) 淡变（恒定振幅），在两个对象之间保持恒定振幅进行交叉淡变。



- Xfade (power) 淡变（恒定功率），在两个对象之间保持恒定功率进行交叉淡变。



- **Delay 延迟**, 在两个对象之间添加无声段落。
- **Sample Accurate 精确到采样点**, 对象之间进行零延迟的无缝过渡。请注意, Opus (在从磁盘进行流播放时) 和 XMA 音频格式不支持精确到采样点的过渡。
- **Trigger rate 触发速率**, 使用特定速率来触发容器内的对象。该选项对于模拟快速枪声十分实用。



备注

在使用 Xfade、Sample accurate 和 Trigger rate 时有一些局限性。以下几点值得注意:

- Blend Container 和 Switch Container 不支持交叉淡变 (Xfade)。当子级 Blend/Switch Container 播放时, 其父级 Random/Sequence Container 的 Xfade 设置也无法正常工作。
- 假如为 Random/Sequence Container 应用了精确到采样点的过渡, 若在当前声音末尾使用中断动作退出循环播放, 而没有足够时间预取接下来要播放的声音, 则可能会导致无法精确地过渡到指定的声音。为了避免这一问题, 可针对这些声音使用内存内容 (不进行流播放, 也不使用硬件编解码器)。
- 对于使用了 Xfade 或 Trigger rate 过渡方式的容器, 带有 Fade out 的 Stop Action 将不起作用。为容器调用 StopEvent 时, 如果 Fade out 时间比当前播放的声音更长, 则播放结束时声音会突然切断。要避免这种情况, 可以使用 Set Voice Volume Action 进行淡出, 并延迟 Stop Action。

9. 在 Duration 文本框中, 输入所需的交叉淡变、延迟或触发速率的时长。



备注

Duration 选项不适用于精确到采样点的过渡。

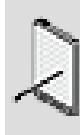
相关主题

- [播放容器内的一个对象](#)
- [定义 Random/Sequence Container 的播放行为](#)
- [创建随机容器](#)
- [创建序列容器](#)
- [创建播放列表](#)
- [定义容器的范围](#)

定义容器的范围

由于可能有多个游戏对象使用同一容器, 因此需要决定游戏中一个容器的所有实例是作为一个对象处理, 还是各个实例单独处理。在 Wwise 中, 这一概念称为容器的范围。您可以将范围设置为以下任一选项:

- **Global 全局** —— 将游戏中使用的所有该容器作为一个对象进行处理, 因此可以针对所有游戏对象避免重复声音或语音对象。
- **Game object 游戏对象** —— 将该容器的各个实例作为单独实体进行处理。即不会在所有游戏对象间共用声音内容。



备注

Scope 选项不适用于 Continuous 播放模式下的序列容器，因为每次事件触发该容器时都将播放全部播放列表。

定义容器的范围 —— 示例

假设您正在研发一个第一人称角色扮演游戏。有 10 名警卫，这些警卫共用 30 条相同的对话。此时，您可以将 30 条语音对象编组至随机容器，并将其设置为 Shuffle 和 Step。这 10 名警卫可以同时使用该容器，将容器的范围设置为 Global，就可以避免不同的警卫重复同一条对话的情况。这一概念可以应用至游戏中供多个对象使用的任何容器。

设置容器的范围的方法如下：

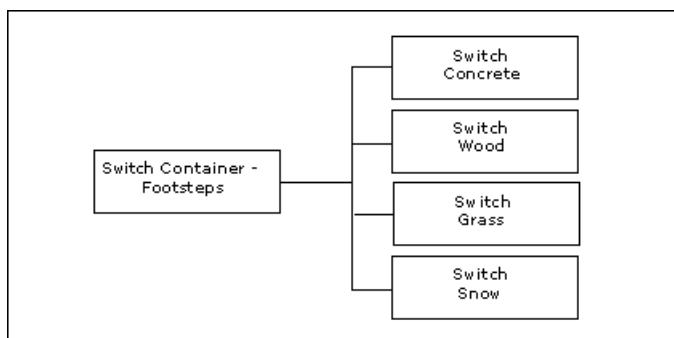
1. 将随机或序列容器加载至 Property Editor。
2. 在 Scope 分组框中，选择以下任一选项：
 - Global 全局，将游戏中使用的所有容器作为一个对象进行处理。因此，可以针对所有游戏对象避免重复声音或语音对象。
 - Game object 游戏对象，将容器的各个实例作为单独实体进行处理。即不会在所有游戏对象间共用声音内容。

相关主题

- [定义 Random/Sequency Container 的播放行为](#)
- [创建随机容器](#)
- [创建序列容器](#)
- [创建播放列表](#)
- [定义容器内对象的播放方式](#)

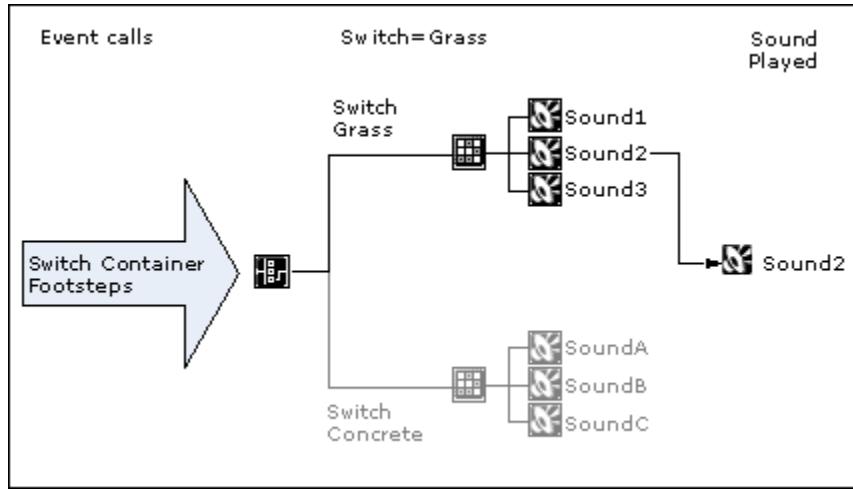
定义 Switch Container 的内容和行为

通过使用 Switch Container（切换容器），您可以根据游戏内的不同条件变化来为对象编组。在 Switch Container（切换容器）中，Switch（切换开关）或State（状态）被用来代表各个不同条件。例如，可以为角色行走时的所有不同表面材质创建一个切换容器。容器中的切换开关可能包含适合用于混凝土、木头、草地、雪和游戏中角色可行走的其它表面材质。



每个切换开关/状态对应于与该条件有关的声音对象。例如，所有混凝土上的脚步声可编组至“Concrete”切换开关，所有木头上的脚步声可编组至“Wood”切换开关，以此类推。当游戏调用切换容器时，Wwise 会验证当前哪个 Switch/State 处于激活状态，以决定播放哪个容器或声音。

下图说明当事件调用“Footsteps”切换容器时发生的情况。该容器根据游戏中角色可行走的不同表面材质将声音编组。在本例中有两个切换开关：Grass 和 Concrete。当该事件调用切换容器时，角色走在草地上（切换开关 = Grass），因此会播放草地上的脚步声。随机容器用于将切换开关内的脚步声编组，因此当角色每次踏到同一表面时，都会播放不同的声音。



定义切换容器的类型

创建切换容器时，您必须定义该容器应该是基于状态、切换开关，还是 RTPC。一方面，您可以在切换容器的属性编辑器中选择切换开关或状态；另一方面，RTPC 可以关联至 Switch Group Property Editor 中的切换开关组。有关 RTPC（实时参数控制）如何与切换开关进行关联的详细信息，请参阅[将 Game Parameter 值映射到 Switch](#)。

决定切换开关容器类型后，您还必须为该容器指派切换开关或状态组。这定义了可供指派对象所用的 Switch/State/RTPC。

在为容器指派状态或切换开关组，或将 RTPC 关联至切换开关之前，您必须先创建它们。有关创建切换开关组、状态组和 RTPC 的信息，请参阅：

- [使用 State](#)
- [使用 Switch](#)
- [创建 Game Parameter](#)

定义切换开关容器类型的方法如下：

1. 将 Switch Container 容器加载至 Property Editor 中。
2. 在 Switch Type 分组框中，选择以下任一操作：
 - Switch，容器将基于游戏切换开关。
 - State，容器将基于游戏状态。

3. 从 Group 列表中，选择要指派至容器的切换开关或状态组。
组内的 Switch/State 将显示在内容编辑器的 Assigned Objects 面板中。
4. 从 Default Switch/State 列表中，选择当游戏无法确定切换开关或状态时，应该播放哪个默认 Switch/State。

相关主题

- 定义 Switch Container 的内容和行为
- 定义切换容器的播放行为
- 管理 Switch/State 的内容
- 定义切换容器内对象的播放行为

定义切换容器的播放行为

由于切换开关或状态可在游戏中随时更改，因此您需要决定对象是立即应用更改，还是下次播放切换容器时再应用更改。Wwise 提供以下播放模式：

- Step
- Continuous

您可针对 one-shot 声音（一次播放一个的声音，如脚步声）使用 Step 选项。而 Continuous 选项对于持续循环的对象（如单板滑雪声）则更为实用。

定义切换容器播放模式的方法如下：

1. 将 Switch Container 容器加载至 Property Editor 中。
2. 在 Play Mode 分组框中，选择以下任一选项：
 - Step，不论播放过程中切换开关是否更改，仅在触发新的播放事件后播放不同的对象。
 - Continuous，只要检测到新的 Switch/State，就播放新对象。当选择 Continuous 时，切换播放对象不需要新的播放事件。

相关主题

- 定义 Switch Container 的内容和行为
- 定义切换容器的类型
- 管理 Switch/State 的内容
- 定义切换容器内对象的播放行为

管理 Switch/State 的内容

您必须将切换容器内的对象指派至特定切换开关或状态。将对象指派至 Switch/State 时，可能需要执行以下操作：

- 为 Switch/State 添加或移除对象
- 为 Switch/State 添加或移除对象
- 在 Switch/State 间移动对象

为 Switch/State 添加或移除对象

通过将对象和其它容器添加至 Contents Editor 中的 Assigned Objects 面板，即可将对象指派至切换开关。您可以按住 Ctrl 或 Shift 键同时点击多个对象，然后同时添加或移除它们，来更快地指派对象。如果您将多个对象指派至一个切换开关，那么在 Wwise 内和游戏运行时，它们会同时播放。

在大多数情况下，您将会指派切换容器中已有的对象，但也可能需要将对象直接从 Project Explorer 中的 Audio 选项卡中拖拽至 Switch/State 上。这些对象将被从当前位置移动到切换容器中。如果您要创建对象的副本，而不是移动它，则可按住 Ctrl 键，同时将该对象从 Project Explorer 的 Audio 选项卡中拖拽至 Assigned Objects 面板中的 Switch/State 上。

为 Switch/State 指派/移除对象的方法如下：

1. 将 Switch Container 容器加载至 Property Editor 中。

容器内的对象将显示在 Contents Editor 中。

2. 要将对象指派至 Switch/State，请将其从 Contents 面板中拖至 Assigned Objects 面板中的 Switch/State 上。

对象将被添加至 Switch/State。



备注

必须将对象直接拖拽至切换开关或状态的名字上。

3. 要从 Switch/State 中移除对象，请在 Assigned Objects 面板中点击要移除的对象。
4. 按 Delete 键。

该对象将从 Switch/State 中移除，但仍保留在 Contents 面板中。

相关主题

- [在 Switch/State 间移动对象](#)
- [定义 Switch Container 的内容和行为](#)
- [定义切换容器的类型](#)
- [定义切换容器的播放行为](#)
- [定义切换容器内对象的播放行为](#)

在 Switch/State 间移动对象

如果首次将对象指派至 Switch/State 时出现错误，或是要移动对象，您则可以随时将它们移动至新位置。



技巧

要同时移动多个对象，则请按住 Ctrl 键并点击各个对象，然后将它们拖至新位置。

在 Switch/State 间移动对象的方法如下：

1. 在 Contents Editor 的 Assigned Objects 面板中，选择要移动的若干个对象。
2. 将对象拖拽至新的 Switch/State。

对象将被指派至新的 Switch/State。

相关主题

- [为 Switch/State 添加或移除对象](#)
- [定义 Switch Container 的内容和行为](#)
- [定义切换容器的类型](#)
- [定义切换容器的播放行为](#)
- [定义切换容器内对象的播放行为](#)

定义切换容器内对象的播放行为

由于在游戏内，切换开关和状态会经常变化，因此您需要决定在发生变化时，切换容器内的各个对象将如何反应。可以选择以下播放行为：

- Play —— 决定是每次切换被触发时都重新播放对象，还是仅当触发的 Switch/State 发生变化时才重新播放对象。
- Across Switches —— 对于被指派至多个切换开关的同一对象，当触发新的 Switch/State 时是否继续播放。
- Fade In —— 决定当触发新的 Switch/State 时，新对象是否应用淡入。
- Fade Out —— 决定当触发新的 Switch/State 时，已有对象是否应用淡出。

定义切换容器内对象播放行为的方法如下：

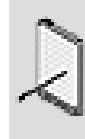
1. 将 Switch Container 容器加载至 Property Editor 中。

容器内的对象将显示在 Contents Editor 中。

2. 在 Play 列中选择 1st only 选项，那么将仅当 Switch/State 有变化时才播放该对象。如果您不勾选 1st only 选项，那么游戏每次触发切换容器时，无论 Switch/State 是否有变化，都将播放该对象。
3. 在 Across Switches 列中选择 Continue to play 选项，那么如果音频对象同时指派至源和目标 Switches/States，则在 Switch/State 变化期间对象将继续播放。如果您不勾选 Continue to play 选项，那么对象将停止并从开头重新播放。



4. 如果您希望在 Switch/State 发生变化时让新对象淡入，则请在 Fade-In 文本框中输入时间长度。



备注

Fade-In 选项仅在 Continuous 播放模式下可用。

5. 如果您希望在 Switch/State 发生变化时让现有对象淡出，则请在 Fade Out 文本框中输入时间长度。

相关主题

- [定义 Switch Container 的内容和行为](#)
- [定义切换容器的类型](#)
- [定义切换容器的播放行为](#)
- [管理 Switch/State 的内容](#)

定义 Blend Container 的内容和行为

您可以使用混合容器（Blend Container）将多个对象编组，就能同时听到这些对象。也可以将对象置于混合容器内的Blend Track（混合轨）中，这样就可以轻松地为它们应用多个 RTPC 曲线。甚至可以在这些对象之间基于 RTPC 设置它们的交叉淡变。这样当游戏中的参数发生变化时，就可以在对象间创建平滑的过渡。

使用混合容器时，可以执行以下操作：

- [创建混合容器](#)
- [使用 Blend Track](#)
- [管理交叉淡变](#)
- [预览混合容器的内容](#)

创建混合容器

在某些游戏情形中，您可能会需要同时播放多个相关对象，来构成复杂的组合。混合容器可以用来将多个对象编组，结构非常灵活。播放容器时，容器内的所有对象会同时播放。

您可以将以下对象添加至混合容器：

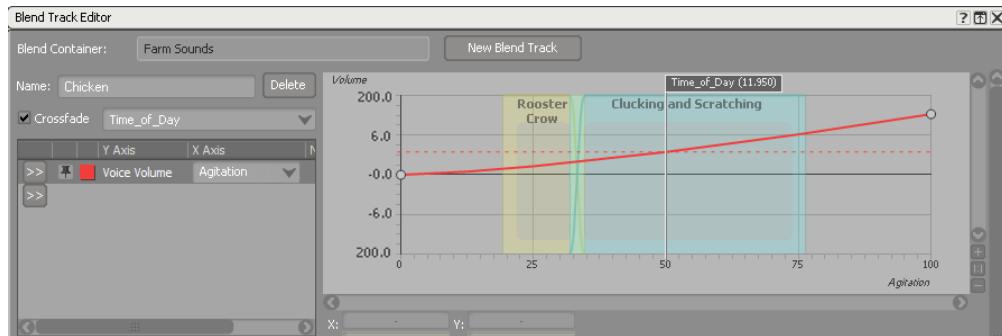
- Blend Container（混合容器）
- Random Container（随机容器）
- Sequence Containers（序列容器）
- 声音
- Switch Container（切换容器）
- Voices（语音）

使用混合容器——示例 1

假设您的游戏发生在农场。您希望玩家能够体验到真实的农场氛围。增强这种体验的一种方式是找到各种农场动物声音（牛哞哞叫声、鸡咯咯叫声、马嘶声），并将这些声音编组至“农场声音”混合容器内，这样玩家就可以同时听到这些声音。

使用混合容器——示例 2

现在来进一步设置您的农场。您希望玩家进入农场时，可以听到一系列不同声音，而不仅仅是动物嘈杂的叫声。解决这个问题的一种方法是在 Blend Track 内组织声音，并根据游戏参数（一天中的时间）让这些声音进行交叉淡变。例如，您可以创建一个“Chicken Blend Track”，这样玩家就能在清晨听到公鸡啼叫，白天听到咯咯叫声和啄食声，而夜晚保持安静。也可以创建 RTPC，以将声音的特征与游戏参数相关联。例如，您可能会需要当您挥舞剑时，鸡会感到紧张。为此您可创建一个“Agitation (躁动)”RTPC，并创建一个 RTPC 曲线，当 Agitation 等级增加时会提高 Chicken Blend Track 中的音量。



使用 Blend Track

Blend Track 用于将混合容器内的对象及其属性值进行编组。每个 Blend Track 可以包含多个对象，它们可以同时被听到，也可以使用 RTPC 由游戏参数决定何时听到它们。

使用 Blend Track 时，您需要执行以下操作：

- [创建 Blend Track](#)
- [为 Blend Track 添加和移除对象](#)

创建 Blend Track

在混合容器内组织对象的第一个步骤是创建并命名 Blend Track。

创建新 Blend Track 的方法如下：

1. 将混合容器加载至属性编辑器。
2. 点击 Edit Blend Tracks。

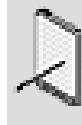
Blend Track Editor 将打开。

3. 点击 New Blend Track。
4. 输入 Blend Track 的名称，然后按下 Enter。

此时新的 Blend Track 将显示在 Blend Track Editor 中。

5. 重复该步骤，来创建所需的新 Blend Track。

您创建的 Blend Track 将显示在 Blend Track Editor，以及 Contents Editor 的 >Blend Tracks> 部分中。



备注

要删除 Blend Track，请选择 Blend Track，并按下 Delete。

相关主题

- [为 Blend Track 添加和移除对象](#)
- [添加 RTPC 曲线至 Blend Track](#)
- [在 Blend Track 中编辑 RTPC 曲线](#)
- [在 Blend Track 中显示 RTPC 曲线](#)

为 Blend Track 添加和移除对象

如果不为创建的 Blend Track 添加对象的话，Blend Track 则将是空白的。混合容器内的每个 Blend Track 最多可以包含 128 个对象。一个对象可以同时存在于多个 Blend Track 中，一个 Blend Track 也可以多次包含同一对象。您可以在 Content Editor 中为 Blend Track 添加或移除对象。



技巧

Blend Track 内的对象顺序十分重要，因为顺序决定了这些对象如何行排列和交叉淡变。有关交叉淡变的详细信息，请参阅 [管理交叉淡变](#)。

将对象添加至混合容器中的 Blend Track 的方法如下：

1. 将混合容器加载至属性编辑器。
2. 将对象从 Contents Editor 或 Project Explorer 中拖至 Blend Tracks 列表中的各个 Blend Track，为其添加对象。



备注

从 Blend Track 中移除对象，选择对象并按下 Delete。该操作会从 Blend Track 列表中移除对象，但不会从 Contents Editor 中移除对象。

相关主题

- [添加 RTPC 曲线至 Blend Track](#)
- [在 Blend Track 中编辑 RTPC 曲线](#)
- [在 Blend Track 中显示 RTPC 曲线](#)

添加 RTPC 曲线至 Blend Track

可以通过两种方式将 RTPC（实时参数控制）曲线添加至 Blend Track：

- 将曲线添加至混合容器本身。
- 将曲线添加至混合容器内的各个 Blend Track。

如果您选择将 RTPC 添加至混合容器本身，该 RTPC 则将同时作用于容器内的所有对象。有关创建和使用 RTPC 的详情，请参阅 [使用 RTPC](#)。

但您也可以将 RTPC 添加至容器内的各个 Blend Track。通过这种方式，您可以仅针对所选对象应用 RTPC。例如，您可以将赛车游戏中的所有碰撞声添加至一个 Blend Track 中。如果您使用冲击力 RTPC 来改变 Blend Track 的音量，则该 RTPC 会影响各个撞击声的音量。

将 RTPC 曲线添加至 Blend Track 的方法如下：

1. 在 Blend Track 属性编辑器中，点击 **Edit Blend Tracks** 来打开 Blend Track Editor。



技巧

您也可以按下 **Ctrl+Shift+T** 以访问 Blend Track Editor。

您之前创建的 Blend Track 将被显示。

2. 在需要添加 RTPC 曲线的 Blend Track 中，点击 **Property Selector** 按钮。

可用声音属性列表将被显示。

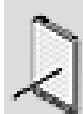
3. 点击您想用 RTPC 影响的声音属性。

RTPC 曲线将被显示在图形视图中。

4. 在 X 轴列表中，点击要指派给 Wwise 属性的游戏参数。

游戏参数被指派给图形视图中的 X 轴。

5. 重复这些步骤，为图形添加更多的 RTPC 曲线。



备注

由于您添加了多条使用不同单位的游戏参数曲线，因此单位将不再显示。但您选中控制点时，仍可看到图形下方 X 轴和 Y 轴框中所显示的单位。

相关主题

- [为 Blend Track 添加和移除对象](#)
- [在 Blend Track 中编辑 RTPC 曲线](#)
- [在 Blend Track 中显示 RTPC 曲线](#)

在 Blend Track 中编辑 RTPC 曲线

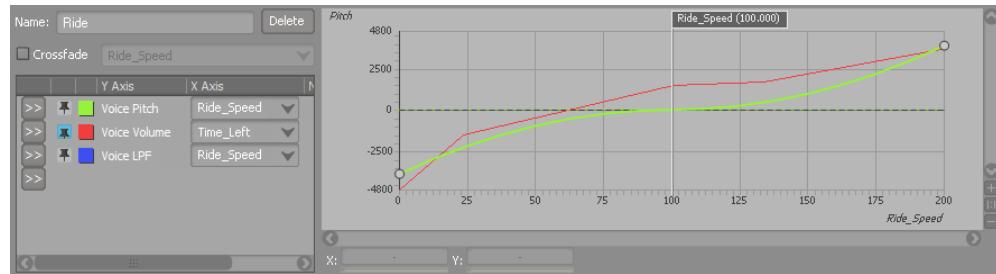
将若干条 RTPC 曲线添加至 Blend Track 后，您可以像定义其它 RTPC 曲线那样定义这些曲线。例如，可以添加点、移动它们，也可以定义曲线段形状。有关这些操作的详细信息，请参阅 [使用 Game Parameter 控制属性值](#)。

相关主题

- [为 Blend Track 添加和移除对象](#)
- [添加 RTPC 曲线至 Blend Track](#)
- [在 Blend Track 中显示 RTPC 曲线](#)

在 Blend Track 中显示 RTPC 曲线

混合容器内的每个 Blend Track 都可能包含多条 RTPC 曲线。使用 Blend Track Editor，您可以同时处理多条 RTPC 曲线。



当在 RTPC 列表中选择某条 RTPC 曲线时，它会在图形视图中高亮显示，并可以进行编辑。有关使用 RTPC 曲线的详情，请参阅 [使用 Game Parameter 控制属性值](#)。

相关主题

- [为 Blend Track 添加和移除对象](#)
- [添加 RTPC 曲线至 Blend Track](#)
- [在 Blend Track 中编辑 RTPC 曲线](#)

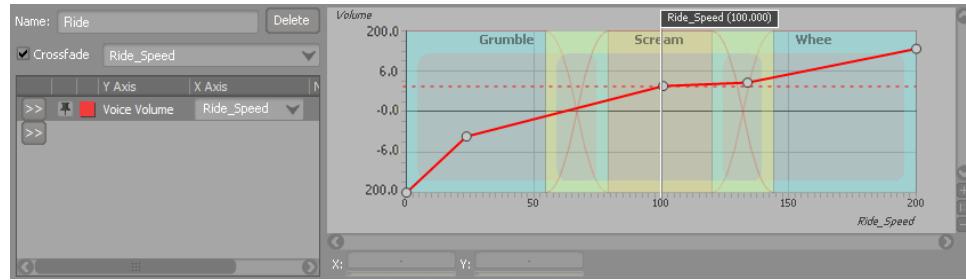
管理交叉淡变

混合容器允许同时播放多个对象。在 Blend Track 中启用交叉淡变会改变这些对象的听感和振动。为了便于管理交叉淡变，Blend Track 内的各个对象都显示为长方块。这些长方块将显示在图中，而 X 轴代表游戏参数的变化。这些长方块在 X 轴上的位置表示游戏参数值，当游戏参数位于相应的 X 轴区间时，将听到对应的声音或感受到振动。

当您重叠方块来创建交叉淡变时，会让对象间的过渡更平滑，增加游戏的真实感。交叉淡变也可以与混合容器的 Blend Track 中的 RTPC 曲线一起使用，来改变对象的属性。

使用交叉淡变——示例

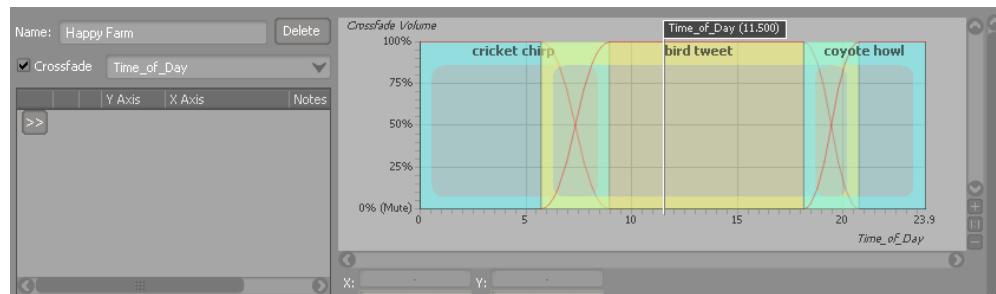
例如，假设您的游戏是过山车模拟游戏。游戏玩家可构建速度为 0 - 200 km/h 不等的过山车。最基本的设计方法，可以使用混合容器将 RTPC 指派给过山车的乘客声音，当过山车加速时，的乘客声的音量会增加。假设您已经收集了各种不同的声音来表现乘客反应，从发牢骚、激动喝彩到惊恐的尖叫。如果使用混合容器来组织这些声音，则可以根据过山车的行驶速度，有选择性地播放这些不同的声音。例如，当过山车的速度达到 40 km/h 时，就能开始听到激动喝彩声。玩家可听到不同乘客的声音，并会听到每种声音的音量在增加。



使用交叉淡变

要在混合容器中使用交叉淡变，您首先需要启用它们。然后可以调整长方块的大小。当游戏参数到达方块所在的 X 轴区间时，就能够听到该对象。当您重叠这些长方块时，就创建了一个交叉淡变。

交叉淡变可以帮助组织播放混合容器内的对象。由于混合容器中的所有声音或振动对象会根据 RTPC 映射同时播放，因此使用交叉淡变可以在对象间实现更平滑的过渡。例如，如果您创建了一个 RTPC 来表现游戏中的时光流逝，则可以让蟋蟀声和鸟鸣声重叠并应用交叉淡变，来营造黎明氛围。



使用交叉淡变的方法如下：

1. 在 Blend Track Editor 中，选择要添加交叉淡变的 Blend Track。

Blend Track 将变为可用。

2. 选择 Crossfading 选项。

Game Parameter 列表变为可用。

3. 选择游戏参数，您将以此参数为基础在该 Blend Track 中应用交叉淡变。

表示对象的长方块将显示在 Blend Track 中，X 轴表示游戏参数。

Blend Track 中的各个对象在图中都表示为彩色块。这些长方块的颜色与内容编辑器的 Blend Track 中列出的对象颜色相同。

4. 选择某个长方块，并向左或向右拖动其边缘以调整其大小和位置。

播放时，将按照 X 轴所示播放相应的对象。

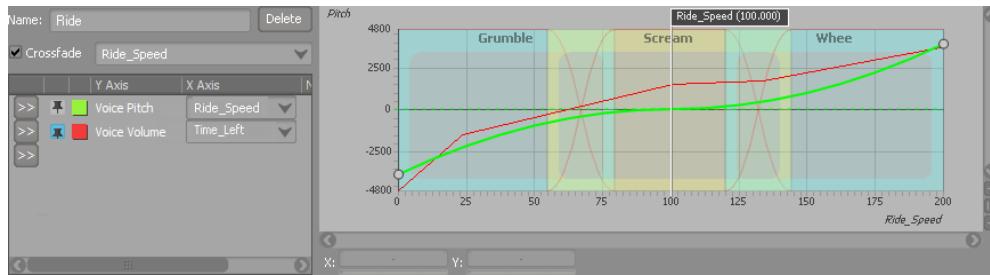
相关主题

- [管理交叉淡变](#)

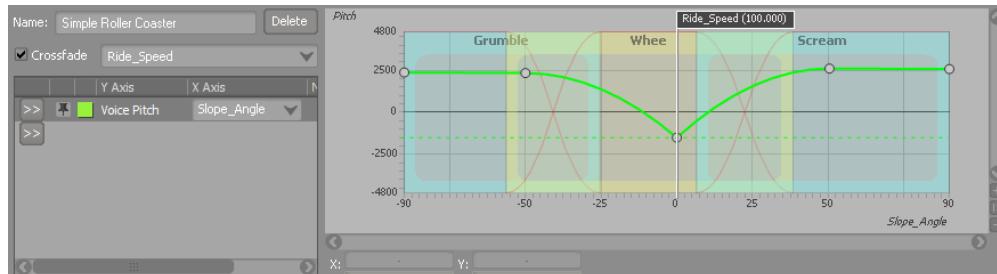
- 与交叉淡变配合使用 RTPC
- 设置交叉淡变模式

与交叉淡变配合使用 RTPC

您可以通过 Blend Track 将 RTPC 曲线的强大功能与交叉淡变的灵活性相结合。在混合容器内，您可以根据游戏参数设置对象之间的交叉淡变，然后为这些相同的对象应用 RTPC 曲线。



记住，在 Blend Track 中的交叉淡变和 RTPC 曲线可以是基于不同参数的。例如，在过山车模拟游戏中，您可以使用 RTPC 曲线在过山车角度变得陡峭时，提升乘客叫声的音高。但您同时可以为该 Blend Track 添加基于速度的交叉淡变，从而在速度改变时听到不同的声音。因此，当过山车提速时，您可以听到声音的内容变化，当过山车角度更陡峭时，这些声音的音调会变得更高。



相关主题

- 使用交叉淡变
- 管理交叉淡变
- 设置交叉淡变模式
- 交叉淡变过渡使用注意事项

设置交叉淡变模式

Wwise 可以用不同的方式处理交叉淡变，这完全取决于您的偏好。可以使用以下三种方法调整 Blend Track 中各对象间的交叉淡变：

- None (无)：当参数值进入方块重叠的部分，RTPC 的属性立即从其最大值过渡到最小值（或由小到大）。
- Automatic (自动)（默认）：淡变范围等于相邻方块的重叠部分宽度。如果没有相邻方块，则没有淡变。

- **Manual (手动)**：您可以移动淡入结束或淡出起始的点。淡入起始或淡出结束的点将总是在重叠块的外部下角处开始。



备注

您也可以修改交叉淡变曲线的形状。有关设置曲线形状的详情，请参阅 [指定控制点之间曲线的形状](#)。

选择交叉淡变模式的方法如下：

1. 右键点击长方块的边缘，并点击选择一个模式。

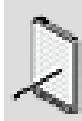
相关主题

- [使用交叉淡变](#)
- [管理交叉淡变](#)
- [与交叉淡变配合使用 RTPC](#)
- [交叉淡变过渡使用注意事项](#)

交叉淡变过渡使用注意事项

在 Random 或 Sequence Container 内使用交叉淡变过渡时，应注意以下注意事项。

- 音频文件长度
 - 音频文件长度必须大于或等于 0.2 秒。
 - 交叉淡变时间最短为 0.1 秒。
- 交叉淡变时间与音频文件长度的关系
 - 从声音 A 交叉淡变至声音 B 时，声音引擎所允许的淡变时间最长为音频文件 A 长度的一半。如果淡变时间大于允许的最大值，那么将被自动调整为淡出文件长度的一半。



备注

如果交叉淡变对于容器内的若干个音频文件过长，则 Wwise 将不会进行限制或提示点。
如果需要对交叉淡变时间进行调整，声音引擎会在运行时进行处理。

- 音高和交叉淡变
 - 如果您使用 RTPC 设置了容器的音高值，或在播放容器时触发了 Set Pitch 事件动作，则对声音应用交叉淡变时可能会产生意外结果。
- 源插件与交叉淡变
 - 对源插件应用交叉淡变时，如果无法确定源的结束时间，那么淡变可能会被忽略。例如，当正弦波生成源的时长基于一个 RTPC 时，就会发生这种情况。在这些情况下，交叉淡变被忽略，并且过渡将会在没有交叉淡变的情况下完成。
- 切换容器与交叉淡变
 - 当 Switch Container 作为 Sequence Container 的子容器时，会根据指派给切换开关的对象数量，区别应用交叉淡变过渡。
- 两个声部
 - 交叉淡变期间，声音引擎会使用两个不同的声部。

- 虚声部和交叉淡变
 - 根据定义，在低于音量阈值或超过播放数量限制时，Play from Beginning 和 Resume 虚声部行为会影响声音持续时间，这不在交叉淡变时间机制的考虑范围之内。
 - 当声部的音量低于阈值时，声部会变为虚声部。对于任何声音，会使用其所有音频通道的实际有效音量与阈值相比较。这个有效音量包括 Actor-Mixer Hierarchy、淡变过渡、互动音乐过渡、RTPC、状态、定位和衰减等所有的音量影响。
 - 当计算声音的有效音量时，淡变过渡的音量影响也计算在内。因此，在 Random、Sequence 或 Blend Container 内的交叉淡变过渡期间，淡变声音在某些时候将有可能低于音量阈值。如果它们从虚声部恢复时的行为是 Play from Beginning 或 Resume，则其真实时长将长于容器淡变的逻辑所预期的时长。这将导致不可预测的行为。更糟糕的是，当声音淡出至低于阈值时，它会停止发声，但仍进行“虚拟”播放，因此永远不会结束。因此，容器将可能停止播放后续声音。

总而言之，应该避免对使用交叉淡变过渡的容器设置这两种类型的虚声部。如果希望对这些容器使用虚声部，则应选择 Play from Elapsed Time 行为。

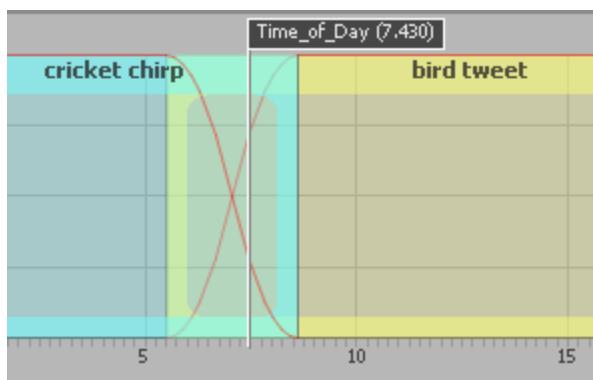
相关主题

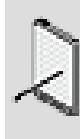
- [使用交叉淡变](#)
- [管理交叉淡变](#)
- [设置交叉淡变模式](#)
- [与交叉淡变配合使用 RTPC](#)

预览混合容器的内容

您可以使用混合容器创建复杂、真实的声景，因此可能会需要在继续下一步工作前先测试一下。使用 Transport Control 和 Blend Track Editor，您可以预览混合容器的内容。也可以随时进行调整，如改变交叉淡变的长短，或添加 RTPC 曲线点。

在 Blend Track Editor 中显示了播放光标，因此可以在播放混合容器的同时，更改游戏参数。由于您已经将这些对象映射至 Wwise 属性值，因此更改游戏参数值时，会自动改变对象属性值。这样就会模拟当游戏参数改变时发生了什么，从而确认属性映射在游戏中的效果。





备注

当您播放混合容器时会播放其中的所有对象，无论它们是否与 Blend Track 相关联。

播放混合容器的内容的方法如下：

1. 将混合容器加载至 Transport Control。
2. 在 Blend Container Property Editor 中点击 Edit，打开 Blend Track Editor。



技巧

您也可以按下 Ctrl+Shift+T 来打开 Blend Track Editor。

此时将显示混合容器中的混合轨。

3. 点击 Play 图标。

播放混合容器时，您可以使用指针更改游戏参数值，以预览或感受对象随参数变化做出的反应。



技巧

微调混合容器内单个对象的 RTPC（实时参数控制）曲线时，您可以将混合容器固定在 Transport Control 中，这样就能一直预览该混合容器。有关在 Transport Control 中固定对象的详情，请参阅 [在 Transport Control 中平移对象](#)。

相关主题

- [创建混合容器](#)
- [使用 Blend Track](#)
- [为 Blend Track 添加和移除对象](#)

对象播放相关的技巧和经验总结

决定工程层级各对象使用何种播放行为之前，您可能会需要阅读以下章节，其中提供了一些技巧和经验总结，可以帮助您获得所需的结果。

对声音采用流播放

一般情况下，您可能会需要将体积过大而无法存储在内存中的声音进行流播放，如音乐、环境声，甚至是语音（如果数量很多）。但决定哪个选项最适用时，需要考虑很多因素。

在项目的初始阶段，应该与开发团队沟通，确定下列事项：

- 音频可以使用多少播放流？
- 每条播放流的带宽是多少？
- 这些播放流是否会共享？
- 可以同时使用多少播放流？

- 从哪里进行流播放？是平台硬盘、光盘，还是从远程服务器？

通过这些信息以及声音的采样率和压缩格式，您可以进行计算，然后就是否使用流播放声音做出决策。

流播放和预取

同时播放的流越多，要求的 Pre-fetch（预取时间）越多。您可以在开始时使用较短的预取时间，但随着项目进展，同时播放的流数量越来越多，可能就需要增加预取时间。

Random Container

将 Random Container（随机容器）范围设置为“Global”（全局）来避免声音重复——使用 Random Container 的 Shuffle 模式进行连续播放，目的是为了避免声音重复。但如果容器包含的声音数量有限，并且同时要播放很多该容器，则可能会出现重复问题。要避免这样的现象，请确保该容器内声音对象的数量至少为播放实例数量的两倍。例如，如果同一 Random Container 有 3 个播放实例，那么您的容器应包含至少 6 个不同的声音对象，以避免重复。

Blend Container

- CPU 占用——请注意，当您播放 Blend Container 时，该容器内的所有对象会同时播放，因此会消耗大量内存。您以可使用虚声部设置来降低 CPU 占用，但如果频繁地进行短暂的交叉淡变，则可能会导致出现错误。有关使用虚声部的详细信息，请参阅 [（音量阈值）和 Virtual Voices（虚声部）](#)。
- 使用 Switch Container（切换容器）还是 Blend Container——在游戏过程中，切换容器和混合容器均可配合 RTPC 使用，并会产生相似的效果。但这两种容器的运作方式有显著不同，您可以根据要求选择使用。
 - 如果您希望在不同时间播放不同的对象，而且不需要交叉淡变，那么请使用切换容器。
 - 如果希望一直播放所有对象，而且需要交叉淡变，则应使用混合容器。

第 11 章 定义定位

定义定位概述	228
理解 Wwise 中的定位	228
使用 Speaker Panning	231
使用 3D 空间化对象	232
Speaker Panning 和 3D Spatialization 交叉淡变	235
为总线应用定位	235
3D 定位图解	236
应用基于距离的衰减	247
使用动画路径定义空间定位	259
将音频信号传送到中置扬声器	268
定位技巧和经验总结	269
理解声道配置	275
声像摆位规则：扬声器对耳机	277
使用 Ambisonics	278

定义定位概述

声音、音乐和振动的定位和传播在吸引玩家，使他们沉浸在在游戏中发挥着关键作用；因此，了解如何处理游戏中各种类型的声音和振动效果非常重要。

典型的游戏将拥有以下各种声音和振动效果的组合：

- 局部环境声——发声体保持在一个位置上。例如，局部环境声音可包括大型机器或喷泉发出的声音。
- 非局部环境声音——发声体可移动，并不依附于某个特定游戏对象。例如，非局部环境声音可包括环境中的鸟鸣声或昆虫声音。
- 移动对象声音——发声体随某个特定游戏对象一起移动。例如，移动对象声音可包括游戏角色、动物等触发的任何声音（例如狗叫声或卫兵的吆喝声）。
- 游戏界面声音——此声音与特定游戏界面要素或者其它保持在屏幕固定位置上的道具相关联。例如，游戏界面声音可包括平视显示器(HUD)的部件、菜单声音（按钮、导航）或第一人称射击游戏中的枪声。

Wwise 拥有一套强大灵活的定位工具，可让您处理上述所有声音类型，创造游戏玩家所期待的体验。

定位——示例（第 1 部分）

假设您正在创建一款第一人称潜行类游戏。在游戏的某个环节，一组特工必须前往一座偏远的火山岛，恐怖分子在岛上劫持了一位特工。此任务非常危险，特工们必须团结合作，紧跟彼此。当特工们潜行通过敌人的丛林基地时，他们将经历以下声音和振动：

- 主人公的脚步声。
- 照亮敌人丛林基地的火把的燃烧声。
- 一群恐怖分子在屋内交谈的声音。
- 蚊子在头顶上飞舞的嗡嗡声。
- 从总部发来的最新消息。
- 执行任务的特工们之间的耳语声。
- 在成功完成任务后用于炸毁基地的炸药的爆炸声。
- 岛上火山持续发出的轰鸣声。
- 火山最后喷发声
- 互动音乐

这些声音中的每一个在定位和传播上都需要接受不同的处理。以下各节讨论在 Wwise 中如何管理每种声音的定位。

理解 Wwise 中的定位

Wwise 拥有各种工具，它们可帮助您逼真地模拟游戏中任何声音对象的定位和传播。以下各节描述 Wwise 中各种对象的定位工作原理：

- 定位声音和音乐对象
- 为单个游戏对象创建多个位置

- 定位振动对象
- 在工程层级结构中设置定位选项
- 处理多声道声源的 3D 定位。

定位声音和音乐对象

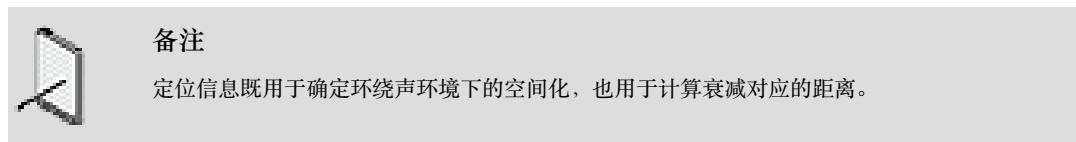
作为最基本的功能，声音和音乐对象可以选择是否使用 3D 空间化。

主要不同之处在于如何将源声道映射至输出扬声器。若不使用空间化，则输出声道将与源声道完全匹配；若使用 3D 空间化，则可将各个输入声道输出至环绕声环境中的任何扬声器。在运行中，可通过 Positioning Type 的 RTPC 更改定位方法。

对于非剧情声音和音乐对象，可使用声像器来平衡各个声道的音量，以便使用不同的扬声器播放声音或音乐对象。

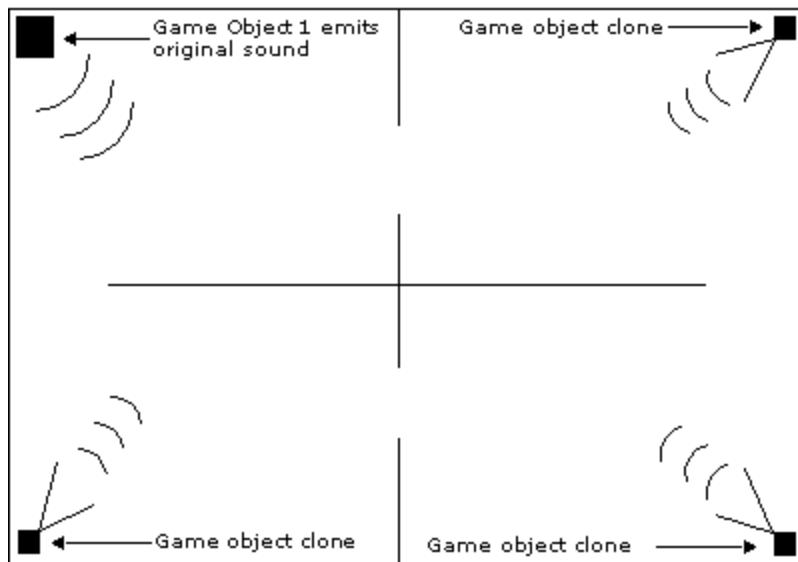
对于剧情声音对象，3D 空间化方便使用 X-Y-Z 坐标位置进行计算（也可使用游戏中发声体顶部和前部朝向矢量的 X-Y-Z 坐标）。不过，可在 Wwise 中使用特定动画路径来预定义听者或发声体的空间化信息。

无论是否使用 3D 空间化，Wwise 都提供衰减功能，即模拟音频信号在发声体远离或背离听者时的自然衰减。



为单个游戏对象创建多个位置

在游戏的某些情景中，一份声音可能不足以创建逼真的声音空间；例如具有大量扬声器的 PA 系统。如果您不想复制声音及相应游戏对象，则可“克隆”发声的游戏对象，然后将克隆声定位于游戏中的不同位置。在 PA 系统例子中，您可以“克隆”发出原声的游戏对象，然后将各个克隆声重新定位于 PA 系统扬声器的位置上，如下图中所示。



各个克隆声是原声的精确复制品，拥有与原声相同的属性、行为和衰减设置。由于您要求原声和“克隆声”同时播放，因此可以同时增加它们的音量，或者指定所有声音的累积音量以原声的最大音量电平播放。

如果使用“克隆声”，则在为原声定义散布和衰减设置时须小心，因为这些声音极可能与声音只来自于一个位置下的情况不同。如果玩家能够同时听到原声和“克隆声”时尤其如此。如果“克隆声”叠加到原声上，合成声音的音量则可能会变得太高而需要剪辑。

需要“克隆”游戏对象的其它例子如下：

- 走廊墙壁上一路布置了许多火把。
- 对湖泊或者瀑布这样很大的异形对象时，使用简单的半径衰减明显是不合适的。
- 墙壁倒塌，墙后传来倒塌声。通过将“克隆的”游戏对象定位到特定位置，您可以高效地模拟部分声笼和声障。

只能使用 Wwise SDK 来“克隆”以及重新定位游戏对象。有关为单个游戏对象设置多个位置的详细信息，请参见 Wwise SDK 文档中的[声音引擎集成纵览 > 将 Wwise 元素集成到游戏中 > 集成 3D 位置 > 集成详情 —— 3D 位置](#)。

定位振动对象

对于不太复杂的设备（例如游戏控制器），它们的马达无法模拟 3D 环境，因此定位无关紧要。然而，可以使用衰减来降低当振动信号远离游戏玩家时振动信号的强度。

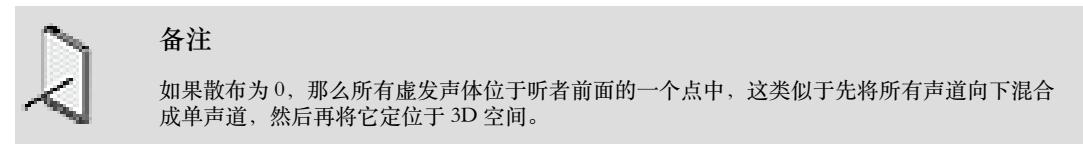
在工程层级结构中设置定位选项

在设置对象的定位时，要记住层级结构中的各个对象都可以有定位设置。定位被视为绝对属性，这意味着顶层对象的定位会自动向下传递给所有子对象。然而，如果需要自定义某个特定对象的定位，则可覆盖这些设置。有关层级结构中的定位工作原理的详细信息，请参阅[工程层级结构中的属性介绍](#)。

处理多声道声源的 3D 定位

多声道声音可像单声道声源一样定位而无需下混为单声道。

为了计算源对各个扬声器的声道贡献大小，“虚发声体”以半圆形状环绕[听者](#)设置。半圆的大小取决于散布（100% 代表整圆，0% 代表听者正前方的点）。然后将半圆划分为与声道数量相同的、大小相等的区域，源的各个原声声道都作用于指定的区域。例如，对于立体声源，圆圈左侧的虚发声体使用源左声道，右侧的所有虚发声体使用源的右声道。一旦完成，将计算各个虚发声体对各个扬声器的实际贡献。因此即使对于 3D 定位，也多少保存了多声道声像。



相关主题

- [使用 3D 空间化对象](#)

- [3D 定位图解](#)

使用 Speaker Panning

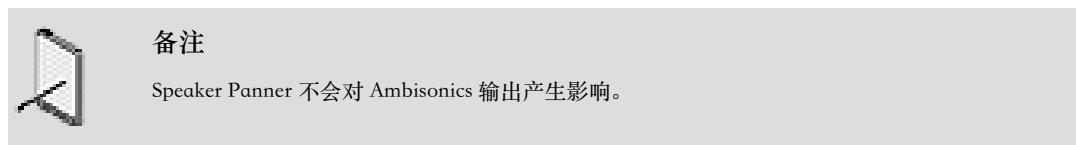
在默认情况下，Speaker Panning（扬声器声像摆位）将设置为 Direct Assignment（直接指派）。此时，无论听者或游戏对象的位置或朝向如何，都会将声音或音乐对象的声音直接指派给输出总线的对应声道。不过，您可以使用 Speaker Panner（扬声器声像器）的 Balance-Fade（平衡淡变）选项，来平衡声音或音乐对象内各个声道的音量，从而以不同的电平在输出总线声道中播放。

Speaker Panner 包含一个二维坐标图，其 X 和 Y 坐标分别模拟汽车声像器的“声像摆位”和“音量淡变”控制。您可以将代表点声源的黑色圆圈拖到此坐标图视图中的任意位置来更改通到输出总线声道的信号功率。

实际功率分布取决于输出总线配置。例如，当输出总线为立体声输出总线时，Y 坐标不起任何作用。请注意，当黑色圆圈位于中间时，7.1 声道配置的侧面扬声器采用其最大电平。

像角色语音之类的声音对游戏玩法至关重要；因此为了确保能够清晰听到，您需要将它们的信号发送至中置扬声器。Center %（中置百分比）控件可用于定义通过中置扬声器的信号量。有关使用 Center % 控件的详细信息，请参阅[将音频信号传送到中置扬声器](#)。

在经过扬声器声像摆位后，可将声音用于游戏中的各种对象，如菜单声音和角色语音。



Attenuation（衰减）

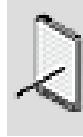
除声像摆位外，还可定义对象的衰减设置。衰减设置用于模拟当信号远离听者时的自然减弱。Wwise 使用一系列曲线来把游戏中的距离值映射到 Wwise 的属性利用这些曲线，可为对象创建基于距离的复杂衰减。为了增加逼真效果，还可使用声锥。声锥根据对象相对于听者的朝向来衰减声音。对于经过摆位的对象，有关如何定义其衰减设置的详细信息，请参阅[应用基于距离的衰减](#)。

振动对象的声像摆位

对于较为简单的振动设备（如游戏控制器），其内部电机无法模拟 3D 环境，因此应将 Speaker Panning 设置为 Direct Assignment。然而，有些情况下可能需要根据声源的距离降低振动信号的强度。在这些情况下，为了使用衰减设置，可启用 Listener Relative Routing（听者相对通路），并将 3D Spatialization（3D 空间化）设为 None（无）。

对于经过摆位的对象，定义其声道的方法如下：

1. 将顶层对象加载到 Property Editor 中。
2. 切换到 Positioning 选项卡。



备注

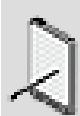
如果该对象不是顶层对象，您则必须选择 **Override parent** 选项后才能设置 Positioning 选项。

3. 从 Speaker Panning 列表选择 Balance-Fade 选项。

Edit... (编辑...) 按钮将激活。

4. 单击 Edit...，打开 Speaker Panner。
5. 将代表点声源的黑色圆圈拖到 2D 平面中的任意位置。

各个声道的音量或强度将调整，以模拟您指定的位置。



备注

可使用 X 和 Y 坐标文本框指定精确值。

相关主题

- [使用 3D 空间化对象](#)
- [将音频信号传送到中置扬声器](#)

使用 3D 空间化对象

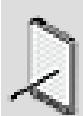
对于 3D 空间化声音和音乐对象，各个输入声道可输出至环绕声环境下的任意扬声器，便于模拟对象相对于听者的移动。为了获得这种增强的环绕声体验，Wwise 同时使用空间定位和衰减。

空间定位

空间化确定游戏 3D 环境中对象的实际位置或定位。使用空间化时，游戏中的声源移动表现为从不同的扬声器中听到声音。

根据正在处理的声音类型，有两种空间化可供选择：

- **Position** (位置)：仅使用游戏对象的三维定位信息 (X-Y-Z 坐标)。
- **Position + Orientation** (位置 + 朝向)：同时使用游戏对象的三维定位和朝向信息 (顶部和底部矢量的 X-Y-Z 坐标)。



备注

为了更好地理解朝向的微妙之处，请参阅“[3D 定位图解](#)”（特别留意“[Spread 的影响](#)”和“[朝向的影响](#)”示例）。

除此之外，还可根据声音类型来选择定位类型（共三种）：

- **Emitter** (发声体)：由实时游戏位置数据来定义“发声体”游戏对象的空间定位。在选择 Emitter 时，Wwise 中所定义的传播属性将直接关联至游戏中听者和发声体的位置或朝向。

- **Emitter with Automation**（发声体自动化）：通过在 Wwise 中创建路径来定义“发声体”游戏对象的空间定位。您可以定义忽略游戏对象位置和衰减的一组固定的传播行为。自动空间化路径对游戏界面声音、菜单声音、非固定位置的环境声及语音来说非常有用。
- **Listener with Automation**（听者自动化）：指定通过在 Wwise 中创建路径来定义“听者”游戏对象的空间定位。您可以定义忽略游戏对象位置和衰减的一组固定的传播行为。自动空间化路径对游戏界面声音、菜单声音、非固定位置的环境声及语音来说非常有用。

游戏中的移动声音和音乐对象大多使用 Emitter 定位，但也可使用 Automation 来完善各种对象，如非固定位置的环境声（鸟叫声或虫鸣声）。

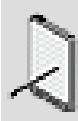
此外，还可选择固定发声体或听者的朝向，或者固定发声体位置（可使用或不使用 Automation）。在启用适用选项后，朝向和位置会随动画路径移动。

Attenuation（衰减）

除空间化外，还可定义对象的衰减设置。衰减设置用于模拟当信号远离听者时的自然减弱。Wwise 使用一系列曲线来把游戏中的距离值映射到 Wwise 的属性利用这些曲线，可为对象创建基于距离的复杂衰减。为了增加逼真效果，还可使用声锥。声锥根据对象相对于听者的朝向来衰减声音。对于空间化对象，有关如何定义其衰减设置的详细信息，请参阅[“应用基于距离的衰减”](#)。

组合使用定位选项

您可以组合使用不同的定位选项为游戏创建丰富多样的音频环境。在特定情况下，可能只需要使用空间化；在其他情况下，则可能需要同时使用空间化和衰减。当然，您也可以只使用衰减（详见[“使用 Speaker Panning”](#)）。



备注

对于那些对游戏玩法很关键的 3D 声音，可能需要将它们的信号传递到中置扬声器，以确保能够清晰听到。Center %（中置百分比）控件可用于定义通过中置扬声器的信号量。有关使用 Center % 控件的详细信息，请参阅[将音频信号传递到中置扬声器](#)。

振动对象的 3D 空间化

并非所有振动设备都完全支持 3D 空间化。例如，游戏控制器内电机很少，移动类型也有限，因此就无法利用 3D 空间定位信息。因此，对于较为简单的振动设备，应将 Speaker Panning（扬声器声像摆位）设置为 Direct Assignment（直接指派）。然而，有些情况下可能需要根据声源的距离降低振动信号的强度。在这些情况下，您可以使衰减设置而不启用空间化。

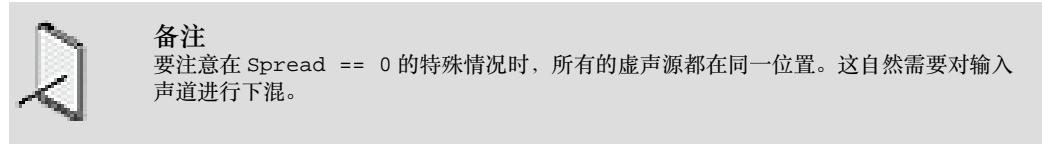
Wwise 3D 详情

请参阅以下列表来详细了解 Wwise 是如何计算 3D 定位的。如果您只想在设计工具中进行设置，而无需知道这些细节，则可以跳转到下面的第二个列表。

3D 定位是如何计算的：

1. 虚声源的位置将根据 Spread 值确定（有关更多详细信息，请参阅[“Spread versus Focus”图示](#)）。使用多个输入声道时，会在散布弧线上进行相应的声道划分（例如对于立体声，其中一半虚声源位置分配给左侧，剩下的一半分配到右侧）

2. 对于每个虚声源位置，都会计算 VBAP 增益。
3. 所有虚声源位置的增益都是这样合并计算的：
 - 对于每个输出声道，会将全部有效输入声道的增益进行平方并加在一起，
 - 再除以输入声道数量，
 - 然后取商的平方根。



定义 3D 对象属性的方法如下：

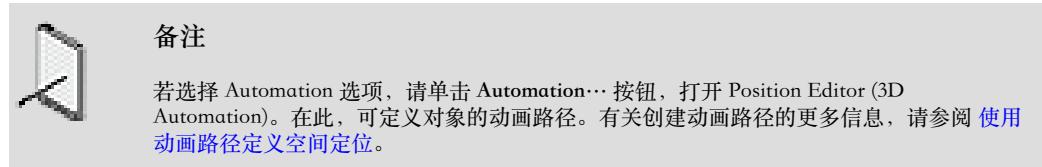
1. 将对象加载到 Property Editor 中。
2. 切换到 Positioning 选项卡。



3. 启用 Listener Relative Routing（听者相对通路）。

分组框将激活。

4. 选择 Position 或 Position + Orientation 选项。
5. 在 3D Position (3D 定位) 分组框中，选择以下选项之一以便确定空间化信息的来源：
 - Emitter：由游戏实时计算空间定位信息。
 - Emitter with Automation 或 Listener with Automation：在 Wwise 中使用动画路径来预定义空间定位信息。



6. 若将 Position 设为使用 Listener with Automation 选项，请启用 Hold Listener Orientation (固定听者朝向) 复选框，以便存储声音开始播放时听者的瞬时朝向，声音播放期间的位置偏移将以此作为基准。在未选择此选项时，将始终通过相同扬声器播放声音，而不管听者朝向如何。在选择此选项时，听者将独立于路径移动。这意味着当听者转身时将通过不同的扬声器听到声音。有关此选项的详细信息，请参阅 [创建顺着听者朝向的动画路径](#)。
7. 若将 Position 设置为 Emitter 选项，请启用 Hold Emitter Position and Orientation (固定发声体位置和朝向) 复选框来保持声源位置不变，直至播放完毕。若禁用此选项，则每一游戏帧都会更新声源位置信息。
8. 若要计算空间定位信息来模拟 3D 游戏空间内的移动，请将 3D Spatialization (3D 空间化) 设置为 Position 或 Position + Orientation。

相关主题

- [应用基于距离的衰减](#)
- [使用动画路径定义空间定位](#)
- [将音频信号传送到中置扬声器](#)

Speaker Panning 和 3D Spatialization 交叉淡变

有时，需要同时使用 Speaker Panning（扬声器声像摆位）和 3D Spatialization（3D 空间化）。其中一种典型情况就是，从需要声像摆位的环境过渡到需要空间化的环境。

例如，在玩家在车内收听音乐时，将使用 Direct Assignment（直接指派）来输出声音。在玩家位于车外的城市街道上时，设计师则可能需要使用 3D Spatialization 选项，使其变成开放世界声音。在玩家打开车门走上街道时，还要避免定位方式之间的转换太过突然。这时可通过平滑过渡来消除这种差异。

在 Wwise 内，若未将 3D Spatialization 设为 None（无），则 Speaker Panning / 3D Spatialization Mix 滑杆将变为可用状态。在设为 0 时，不启用 3D Spatialization；在设为 100 时，不启用 Speaker Panning。在设为这两个极限值之间的值时，则会相应地结合两种定位输出。

若要在示例情况下获得所需结果，则可以使用基于距离的 Game Parameter（游戏参数），并针对音乐对象的 Speaker Panning / 3D Spatialization Mix 设置 RTPC。玩家距离汽车越远，滑杆值越小。距离增大时仍可听到汽车音乐，但它会逐渐融入空间化游戏世界音景，而不会像标准立体声音乐一样输出。



相关主题

- [使用 Speaker Panning](#)
- [使用 3D 空间化对象](#)

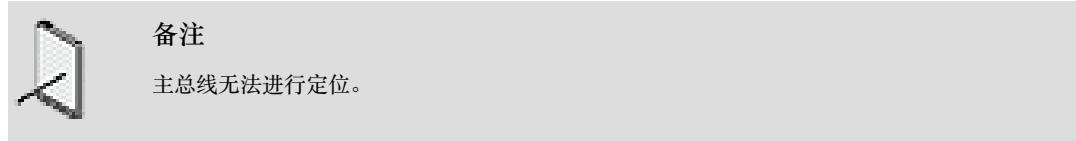
为总线应用定位

您可以像将音频效果器作用于对象一样轻松将音频效果器作用于音频总线和辅助总线。只需将总线加载到 Property Editor（属性编辑器）中，并根据需要将 Speaker Panning（扬声器声像摆位）或 3D Spatialization（3D 空间化）应用于声音对象。有关此过程的详细信息，请参阅[使用 Speaker Panning](#)还有[使用 3D 空间化对象](#)。

由于总线以及声音对象都可以进行定位，声音设计师可以更好地控制声音的传播。虽然并不总是需要这么复杂，但总线定位在某些情况下对增加真实感非常有用，如下所示：

- **虚拟窃听或扩声系统：**游戏中允许玩家使用隐藏式麦克风窥探其他玩家的系统。音频将被捕获并传输至地图另一端的一个虚拟扬声器。在对子混音应用 lo-fi 效果器后，可以对其进行空间化定位，听起来好像从扬声器发出一样。

- Acoustic Portals（声学门户）：您正在制作一款非常倚重空间音频的 VR 游戏。对声学现象进行真实而有说服力的模拟来维持临场感，这是极其重要的任务。您希望玩家能够听到来自隔壁房间的混响声。隔壁房间内的所有声音都会混合并输出到一个混响效果器中，接下来，效果器输出可以经过空间化和定位，就好像它们是从门口传进来一样。
- 早期混响和其他效果：有时候，为每个声部应用效果器显得很麻烦，但使用发送总线上共享的效果器又太笼统了。对于每个可以应用效果器的发声体，您可能都很想为其创建一个总线的实例。空间化可能会在子混音的下游进行，或由您的插件进行处理。这就是 Wwise Reflect 根据每个发声体相对于听者的位置，生成其专属早期反射声的策略。
- 将声音编组来避免额外开销：有一个车辆声音由大量不同的部件声音组成，并由复杂的 RTPC 系统驱动。这些部件处于车辆上的不同物理位置，因此您可以使用不同的游戏对象来作为发声体。但是，为每个发声体应用效果器以及空间化非常消耗性能，因此只有在听者靠得足够近时才需要这样处理。如果听者距离足够远，角度的细微变化并无明显影响，那就应该将所有声音混合在一起，再对整组声音进行处理和空间化。



3D 定位图解

3D 定位会根据游戏中发声体和听者对象的相对位置和朝向，将声源（支持任何[声道配置](#)）在总线（支持任何通道配置）中进行声像摆位。这些对象的位置由游戏通过 Wwise 声音引擎 API 设置。除了游戏驱动的位置和方向属性，Spread 和 Focus 这两个基于距离的属性也协同在摆位算法中发挥作用。在使用 Spread 的情况下，Wwise 中的 3D 定位算法允许在点声源和多声道环绕环境之间进行渐变。

以下是 3D 定位算法的简单说明：

1. 虚声源位于听者的周围，
2. 每个输入通道对应一组虚声源，它们的位置将根据 Focus（仅限标准配置）进行集中。
3. 虚声源会依据发声体相对于听者的朝向进行旋转（仅适用于 3D Spatialization 设为 Position + Orientation 的情况）。
4. 当 Spread 值变小，虚声源将进一步集中在发声体相对于听者的方向上。
5. 每个虚声源都将在输出配置的声道中进行摆位，并进行功率归一化。

关于 Spread 和 Focus 的更多细节，详见[定义各种对象属性的衰减曲线](#)页面。

标准多声道声源的各个声道会通过整个平面等分的形式，分配给虚声源组。声源中的高度声道是不支持的，将被弃用。

声道配置类型

Wwise 原生支持 Standard 及 Ambisonic 两类声道配置。两者主要区别在于上述步骤 5 中涉及的摆位计算，因为两者对于空间音频的表示方式不同。另外，虚声源的布局也

不同，因为 Focus 只是用于控制离散输入声道信号在离散输出声道中的非均匀泄漏，所以该参数只适用于标准配置。但是，虚声源针基于 Spread 的表现和根据发声体-听者关系进行旋转的表现都是一致的，不管声道配置类型如何。

您可以将标准声道配置的音源摆位并输出至 Ambisonic 总线（“编码为 ambisonics”），反之亦然（“从 ambisonics 解码”）。

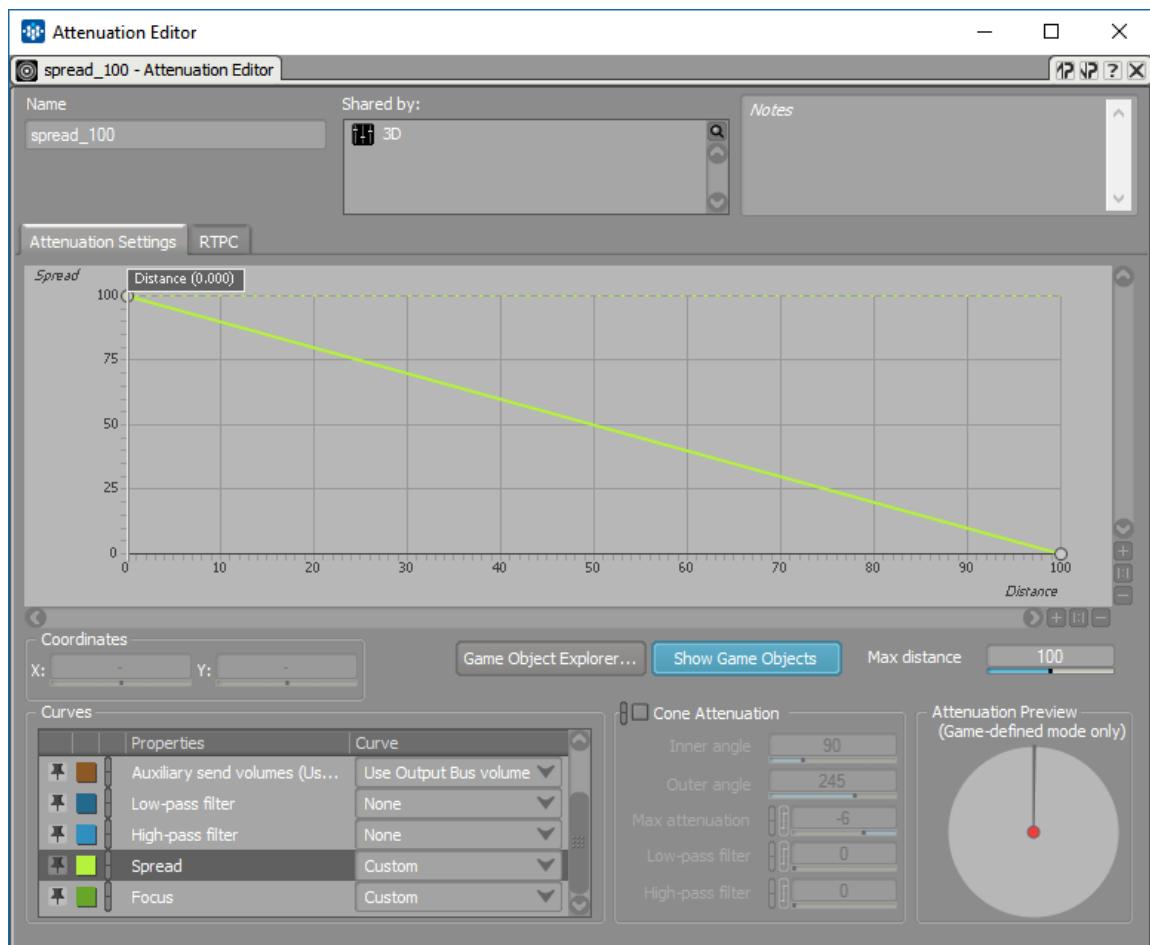
示例

下面几页中的例子对上述算法的不同方面进行了说明：

- Spread 的影响
- 朝向的影响
- Focus
- 斜面声源在平面声道配置中的摆位
- 3D 配置下的声像摆位

Spread 的影响

假设 Spread 的衰减曲线如以下截图所示，在远距离为 0%（点声源），在近距离为 100%（完全环绕）。



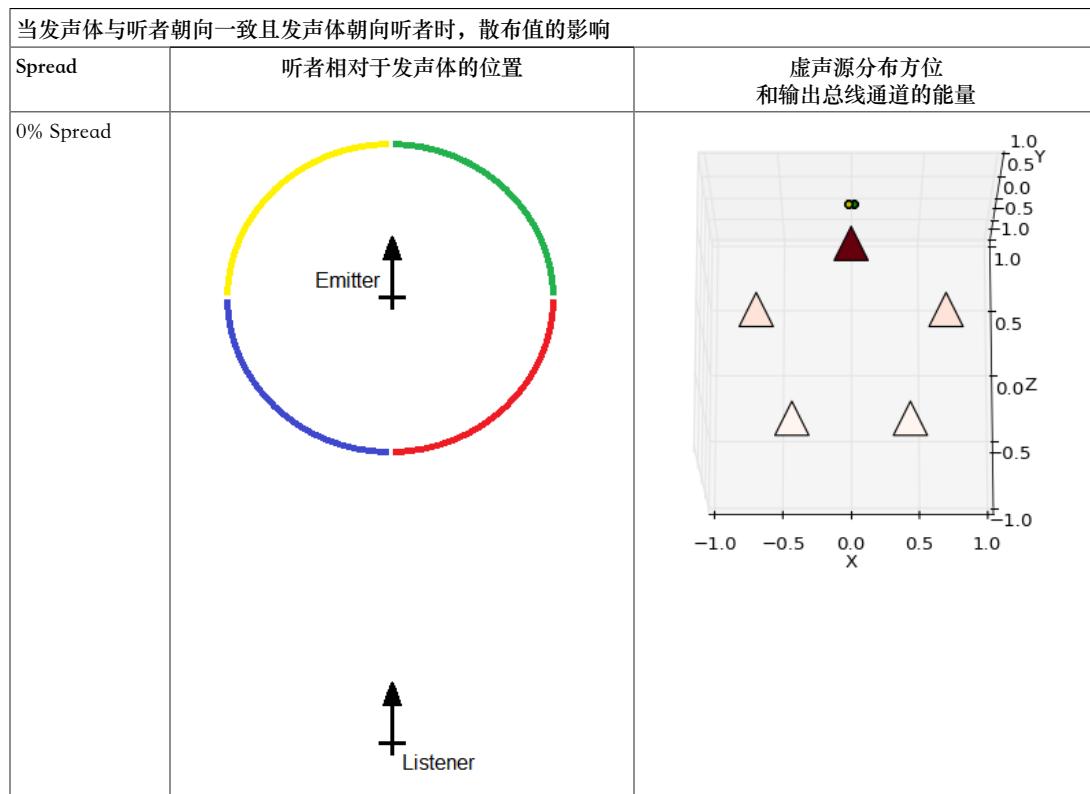
以下各例中使用的 Spread 曲线

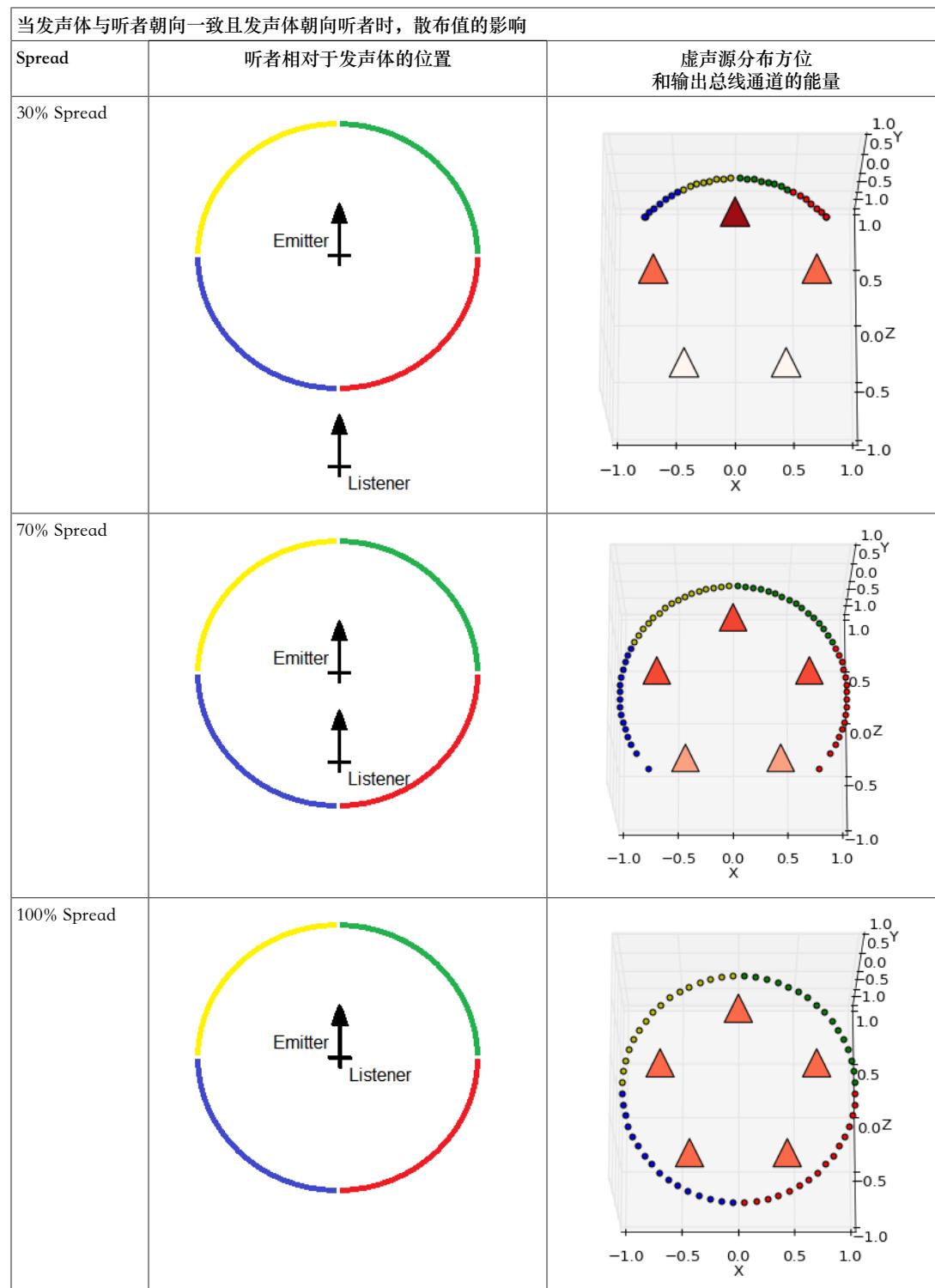
下表中，声源的声道配置是 4.0。这样方便将输入声道的分布与虚声源清楚地关联起来（FL：黄色，FR：绿色，SL：蓝色和 SR：红色）。

在第二列“听者相对于发声体的位置”中，Emitter（发声体）游戏对象表示为指向北方的箭头，即游戏中游戏对象的朝向被设置为向北。Listener（听者）游戏对象也指向北方，同时面向发声体游戏对象。在这四幅图中，分别将听者部署在四个距离不同的位置，对应得出四个不同的 Spread 值：0, 30, 70 和 100 %。

在第三列“虚声源分布方位和输出总线通道能量”中，可以看到虚声源（彩色圆圈）部署方位产生的影响。此外，三角形代表输出总线的声音，中置声道位于顶部。三角形的颜色描述了摆位导致的能量分布，红色越深表示能量越高。

虽然 5.x 扬声器配置的默认角度为 $+/-45^\circ$ 和 $+/-135^\circ$ ，但是在本例中我们采用 5.0 配置并将角度设为 $+/-72^\circ$ 和 $+/-144^\circ$ 。但它其实可以代表任何其他配置，甚至是 Ambisonics 配置，因为虚声源的分布方位完全相同。





朝向的影响



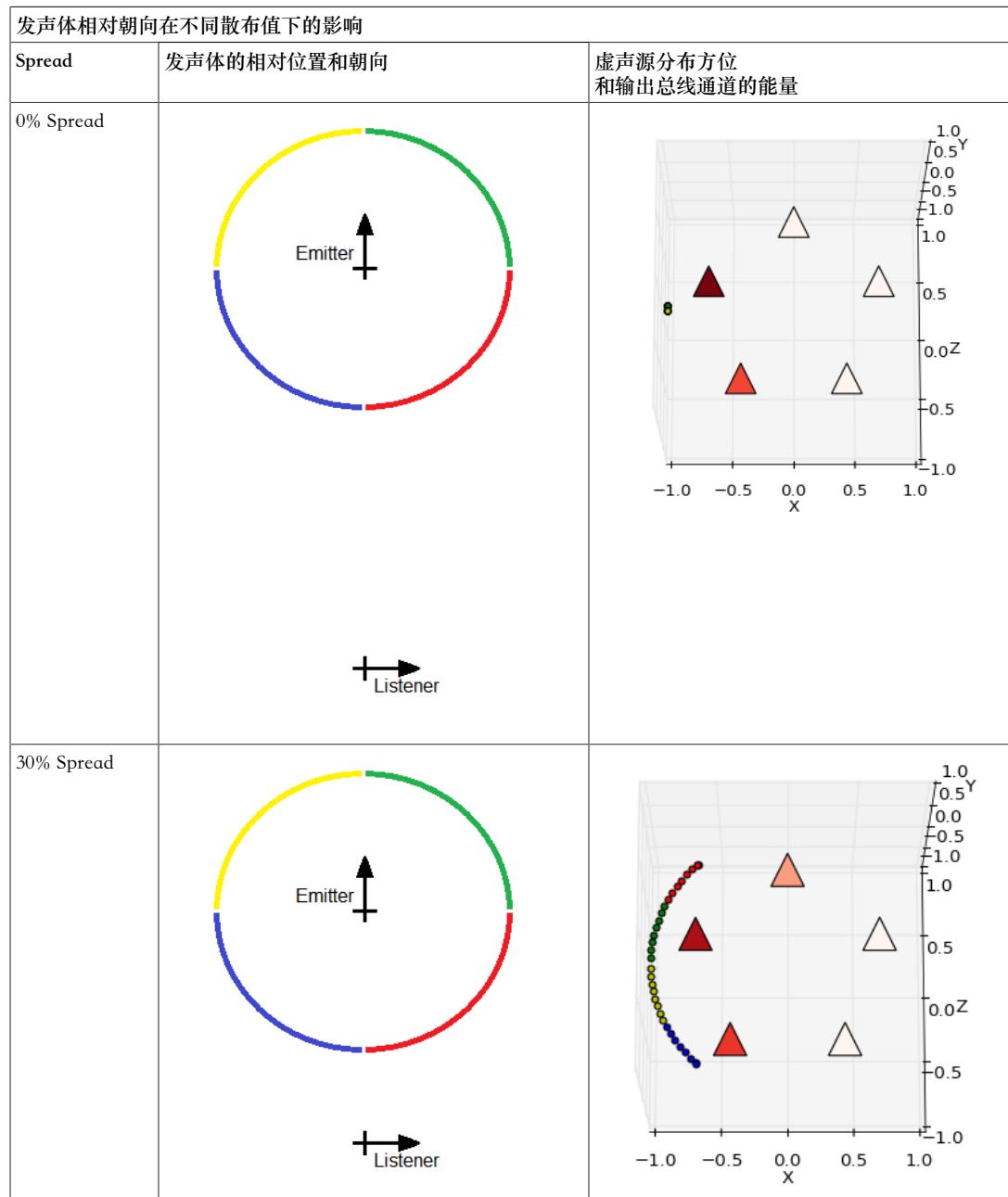
备注

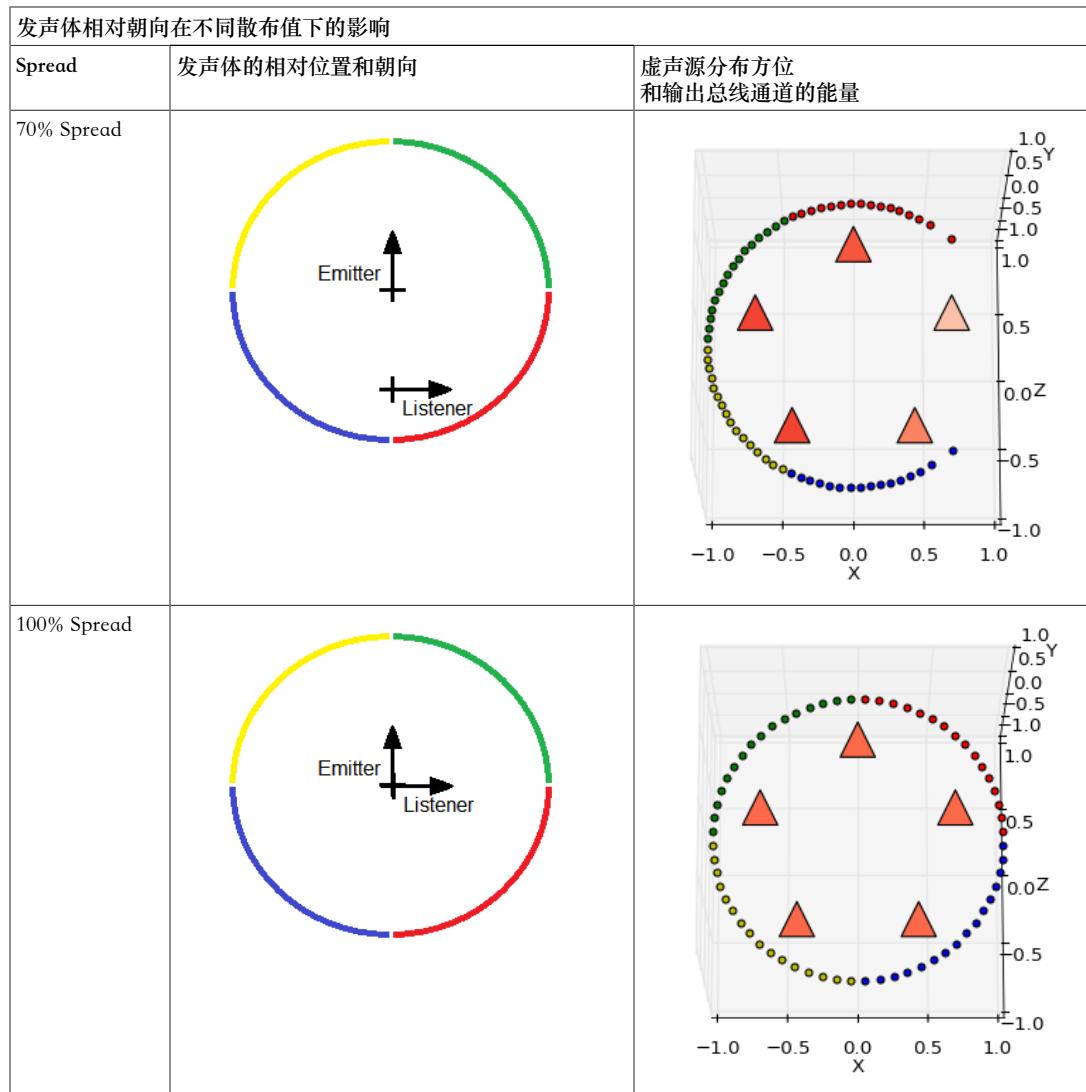
有关背景，请参考例1。

下述发声体朝向的影响仅适用于 3D Spatialization (3D 空间化) 设为 Position + Orientation (方向 + 朝向) 的情况。在此之后，将讨论如何使用 Position + Orientation 和 Position 这两种3D 空间化模式。

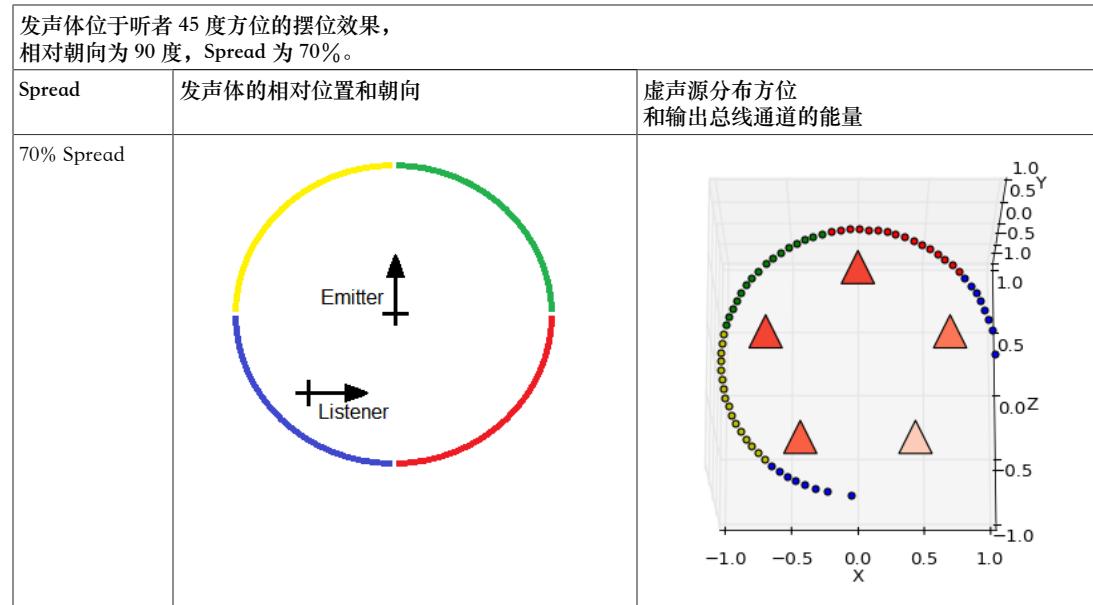
下面的例子类似于上例，只是听者方向朝东。这就同时描述了两种不同朝向因素的影响：

- 发声体相对位置：发声体位于听者的左侧，即根据 Spread 的影响将导致虚声源向左侧集中。在第一张图中 Spread 为 0，效果最明显。
- 发声体相对朝向：发声体朝向相对于听者逆时针旋转了 90 度，这意味着虚声源分布相应地进行了旋转。在第四张图片中 Spread 为 100，效果最为明显，这些颜色表示输入声道已经进行了旋转。当然，如果声源是单声道情况就会不同。





为了确保您了解发声体相对位置和发声体相对朝向之间的差异，下例演示了听者与发声体相对位置为 45 度，相对朝向为 90 度，以及 Spread 为 70% 的结果。这可以看做是听者朝东，位于发声体声场的西南方。绿色和红色发声体声道分别位于听者的左前方和右前方，蓝色声道则环绕听者。



从上面的例子中可知，发声体的朝向仅影响多声道声源。用多声道声源表现“声场”，即用多个声音元素在多声道中摆位来形成“环境声”时，用朝向来驱动旋转效果最佳。在运行时，Wwise 将根据听者和发声体的朝向来旋转多声道素材，将环境声有效地与 3D 世界结合在一起。

但是，有时多声道文件并不一定表现声场，而只是用来传递空间感的简单声音，立体声通常如此。这时，保持离散的原始声道能产生最好的空间感，我们不希望对它们进行旋转。对于这种类型的内容，用户可选择使用 3D Spatialization Position 而不启用发声体朝向。

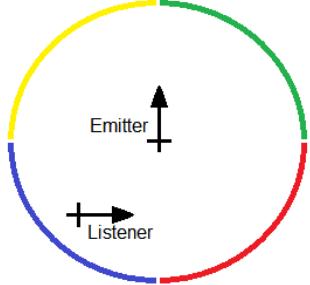
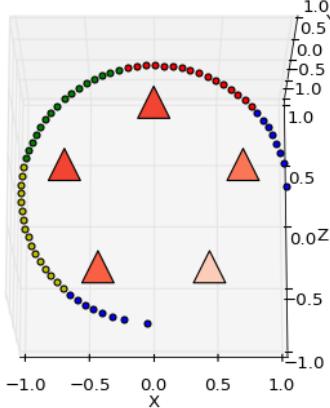
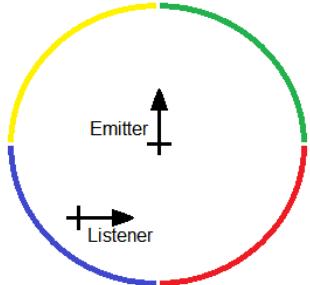
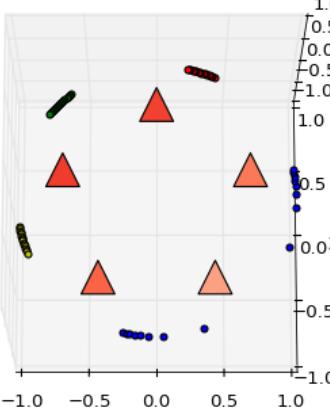
Focus



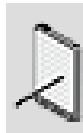
Focus（聚焦）参数能够在应用旋转和扩散之前，将每个输入声道的虚声源集中在一起。在对离散多声道文件进行摆位时，Focus 参数用于减少输入声道信号在其他输出通道中的泄漏。下图显示了我们之前所述 Spread 值为 70% 案例。其中没有使用 Focus，因此我们可以看到，比如红色输入声道最终会泄漏到中置声道、前右声道，甚至左前声道。

在 Focus 值为 80% 时，输入通道会集中，可以看到红色输入声道对中置声道、前右声道还是有影响，但程度比以前少而且不影响左前声道了。<片段0311>尽管如此，还是需要特别注意，就算 Focus 为 100%，在输入声道并非完全对应输出声道位置时，输入声道至少会进入 2 个输出声道。

上述情况并不适用于 Ambisonics 声像摆位，因为其设计即包含输出声道间的泄漏，尽管是均匀泄漏。因此，当声源的声道配置是 Ambisonic 时，将会忽略 Focus 值。

发声体位于听者 45 度方位的摆位效果， 相对朝向为 90 度，Spread 为 70%。			
Spread	Focus	发声体的相对位置和朝向	虚声源分布方位 和输出总线通道的能量
70% Spread	0% Focus		
70% Spread	80% Focus		

斜面声源在平面声道配置中的摆位



备注

有关背景，请参考例1。

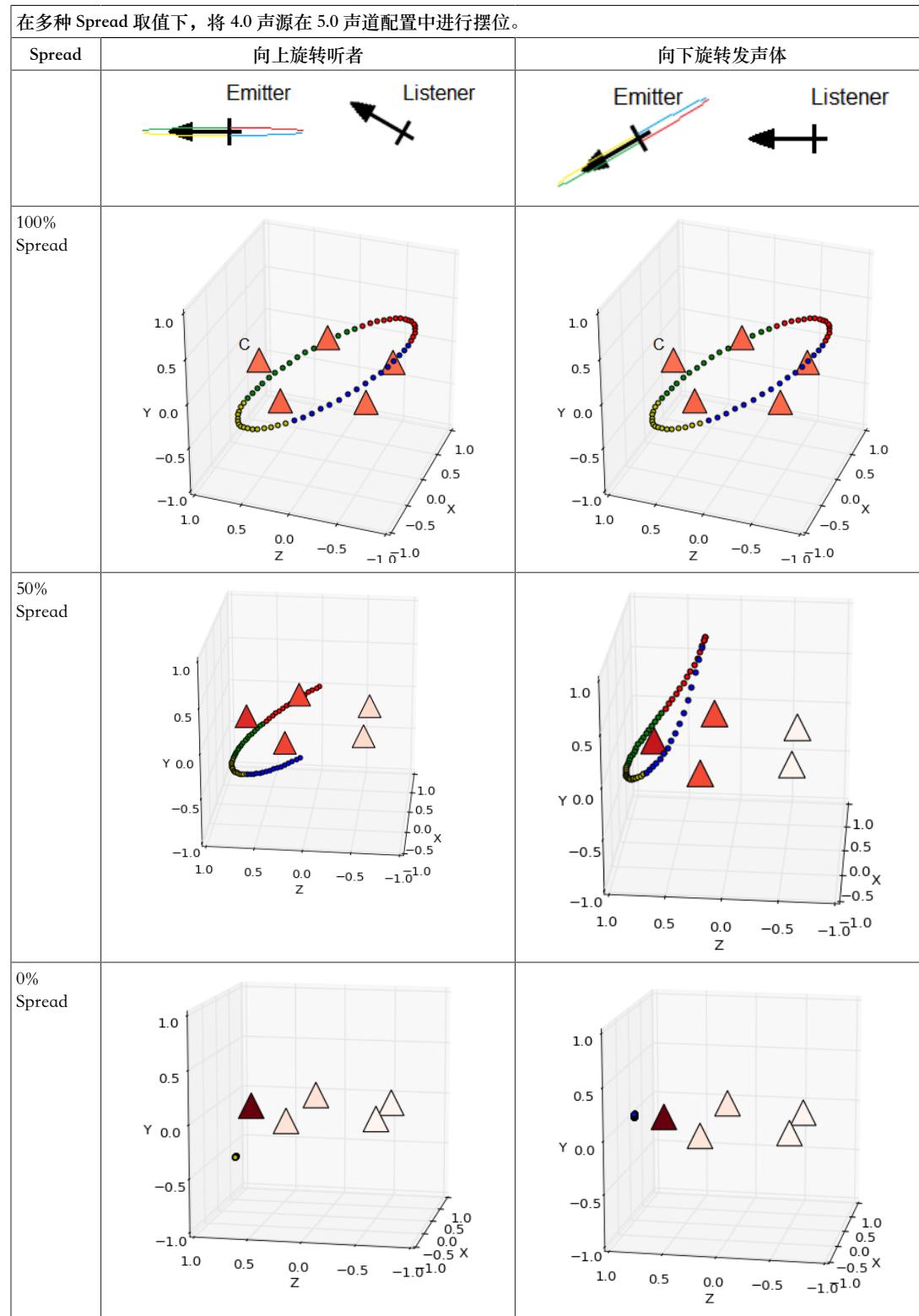
到目前为止，我们仅仅从上方观察到了扁平声源（4.0，没有高度声道）和扁平的输出声道配置（5.0），现在我们要以立体的视角重新审视。请注意，我们讨论的是虚声源坐标在扬声器平面上的投影，用于计算摆位。

在下图的中间一列“向上旋转听者”中，我们展示了当听者向上（朝向天空）30 度时会发生什么。听者的旋转实际上会影响到：

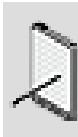
- 发声体声场的相对朝向（如 Spread 值为 100% 时所示）；以及
- 点声源的入射角（如 Spread 值为 0% 时所示）。

这就是为什么点声源（0% Spread）最终定位在中置扬声器下方。

在最右列“向下旋转发声体”中，我们将发声体向下旋转了 30 度。这样只会影响发声体声场的相对方向，而不会影响点声源的入射角。因此，在 Spread 值为 0% 的情况下，点声源位于听者的前方，而不是像中间列一样在其下方。注意从 Spread 100% 到 0% 时，虚声源分布的逐渐变化。



3D 配置下的声像摆位



备注

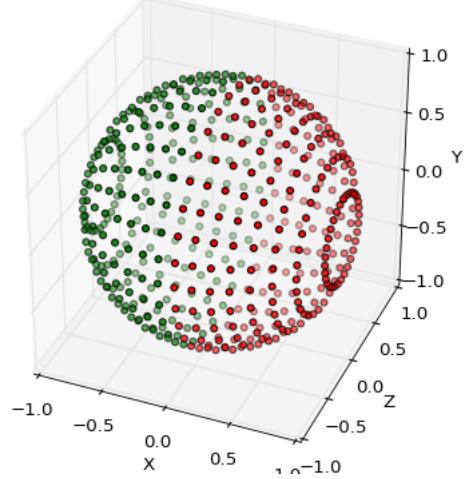
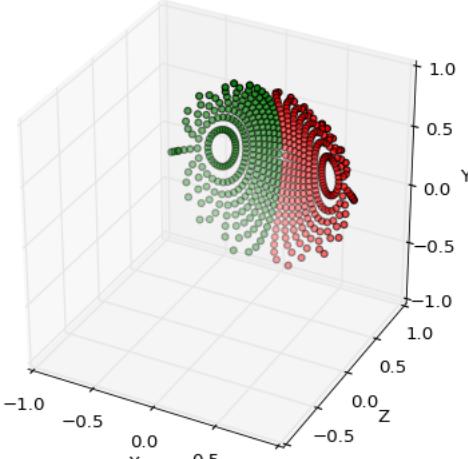
有关背景, 请参考[例1](#)。

3D 声道配置, 即声道配置可以表示来自上方和下方的声音 (例如 7.1.4 或 Ambisonics), 在此配置下进行声像摆位, 虚声源将按照球面形状分布。下图显示了不同 Spread 和 Focus 取值下的立体声以及 4.0 声源。

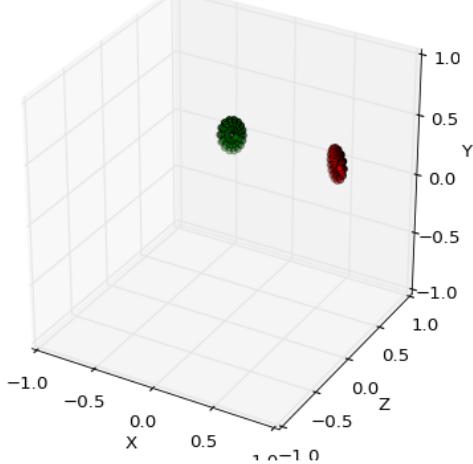
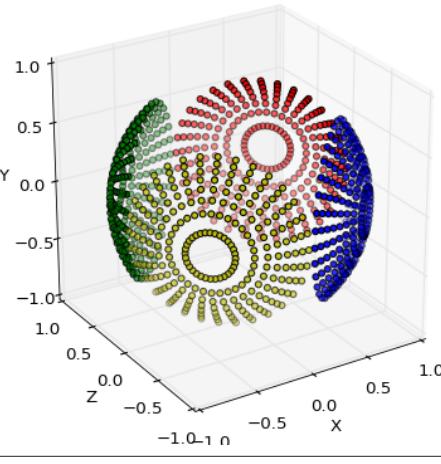
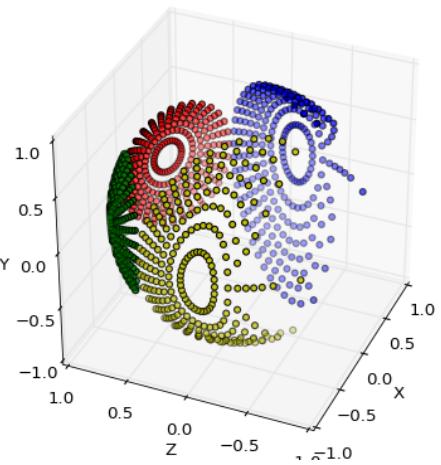
请注意 3D 定位只需 7 个输入声道 (水平面声道) 即可进行摆位, 其他声道将被弃用。

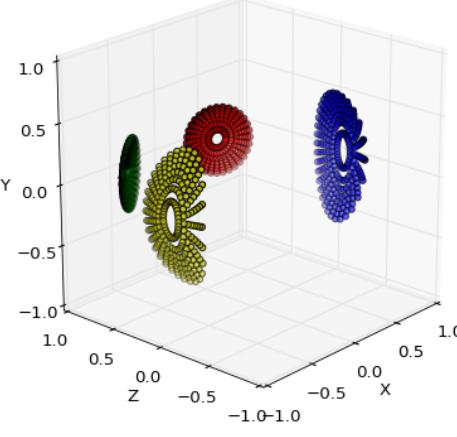
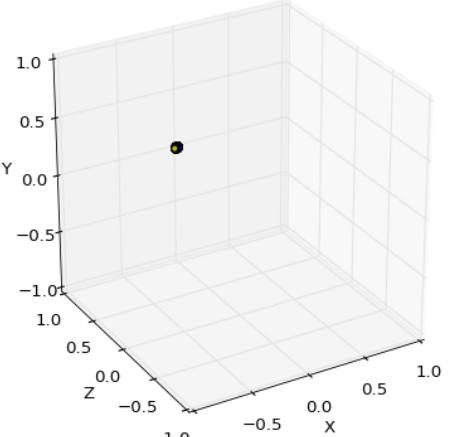
而对于 Ambisonics, 声道与虚声源组并不完全匹配, 但所有的空间信息都被保留了下来。虚声源的转换方式 (旋转, 集中) 与标准配置的转换相同, 因此以下各图中除了颜色不同之外, 全部适用。

在多种 Spread 和 Focus 取值下, 将立体声和 4.0 声源在 3D 配置中 (图中未显示) 进行摆位。

声源配置	Spread	Focus	结果
Stereo	100	0	
Stereo	30	0	

定义定位

在多种 Spread 和 Focus 取值下，将立体声和 4.0 声源在 3D 配置中（图中未显示）进行摆位。			
声源配置	Spread	Focus	结果
Stereo	30	80	
4.0	100	0	
4.0	70	0	

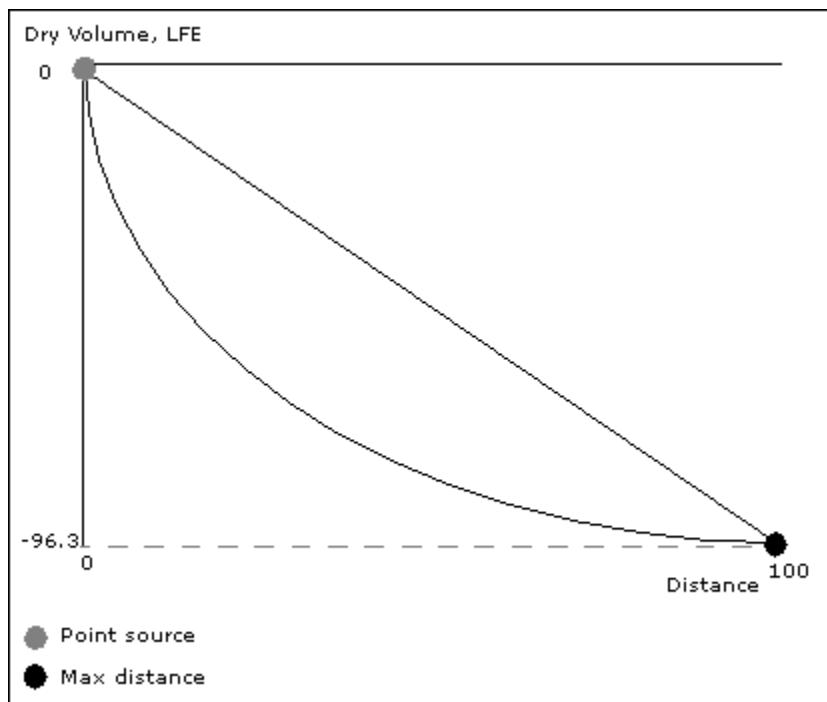
在多种 Spread 和 Focus 取值下，将立体声和 4.0 声源在 3D 配置中（图中未显示）进行摆位。			
声源配置	Spread	Focus	结果
4.0	70	50	
4.0	0	-	

应用基于距离的衰减

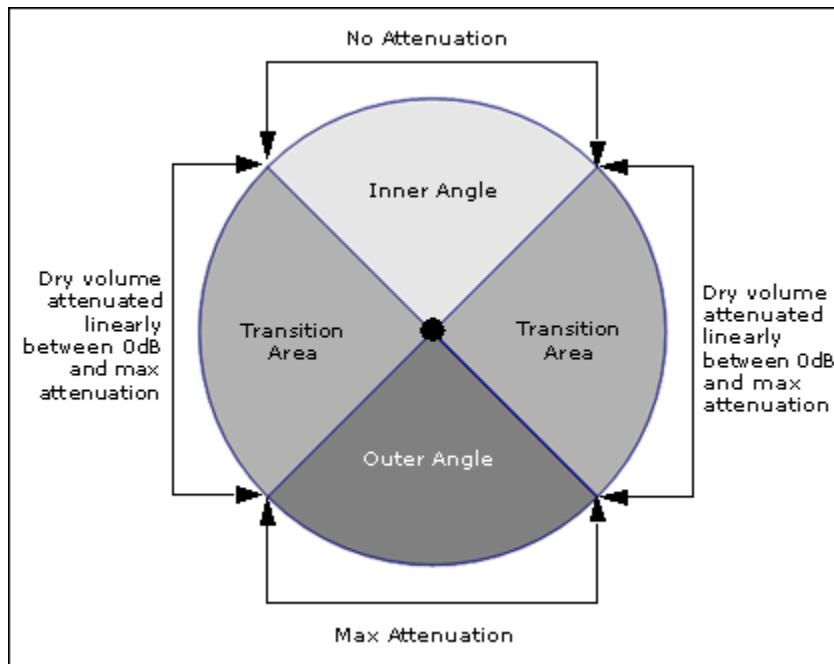
为了模拟当声源远离游戏中的话筒时信号的自然衰减，可使用衰减属性。Wwise 中的衰减基于以下两个属性：

- **Distance attenuation**（距离衰减）：根据发声体和听者之间的距离来影响信号的强度。
- **Cone attenuation**（声锥衰减）：根据发声体相对于听者的朝向来影响信号的强度。

距离衰减使用一系列曲线进行定义。这些曲线将距离值映射到音量等 Wwise 属性值。通过定义曲线沿线各个点的属性，可以在对象远离听者时控制对象的音量衰减。



锥形衰减使用一系列夹角进行定义。这些夹角用于定义声源前方、侧方和背后的区域。通过定义声源周围的这些不同区域，您可以根据游戏中对象的朝向来模拟对象的衰减。



与效果器相似，您可以创建 Attenuation（衰减）属性的实例，然后使用 ShareSets（共享集）在 Wwise 内的大量不同对象之间进行共享。

管理多份衰减

Attenuation 实例是与衰减相关的属性集合。由于游戏中的许多对象拥有相同的衰减属性，因此您可以先创建一份衰减，然后使用 Attenuation ShareSet 在工程中的许多对象之间共享。

ShareSet 是实例属性的集合，多个对象可以采用该集合。在更改 ShareSet 时，采用该 ShareSet 的所有对象都会受到影响。使用 ShareSet 的优势是您不必一个个地更改各个对象的衰减属性，并且在游戏中可以节省宝贵的内存。

然而在某些情况下，可能不应该共享衰减设置。在这些情况下，可创建拥有独特属性值的自定义衰减实例。

您可以执行以下任务来帮助管理工程中的 Attenuation ShareSets：

- [创建 Attenuation ShareSet](#)
- [删除 Attenuation ShareSet](#)

创建 Attenuation ShareSet

在定义对象的衰减属性前，必须先创建 Attenuation ShareSet。Attenuation ShareSet 是衰减属性设置的集合。ShareSets 可供需要同类型衰减属性的所有对象使用。

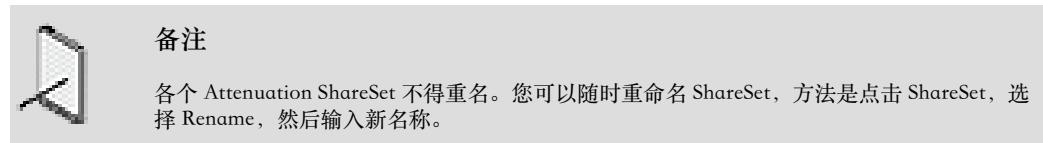
在工程浏览器中创建 Attenuation ShareSet 的方法是：

1. 在 Project Explorer 中，切换到 ShareSets 选项卡。
2. 在 Attenuations（衰减）区域，执行下列其中一项操作：
 - 选择工作单元或虚拟文件夹，然后点击 Project Explorer 工具栏中的 Attenuation 图标。
 - 右键点击工作单元或虚拟文件夹，然后从快捷菜单中选择 New Child > Attenuation（新建子项 > 衰减）。

层级结构中，新的 ShareSet 将作为 Work Unit 或 Virtual Folder 的子级显示。

3. 为新建的 ShareSet 输入名称，然后按 Enter。

新 ShareSet 显示在 Attenuations 层级结构中。



在 Property Editor 中创建 Attenuation ShareSet 的方法是：

1. 将对象加载到 Property Editor 中。
2. 切换到 Positioning 选项卡。
3. 在 Attenuation（衰减）分组框中，单击选择器按钮 (>>)，然后从菜单中选择 New（新建）。



备注

若要激活 Attenuation 分组框，则须启用 Listener Relative Routing。

New Attenuation (新建衰减) 对话框打开。

4. 选择要在其中创建衰减共享集的工作单元。
5. 输入 ShareSet 的名称并点击 OK (确定)。

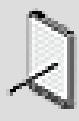
新的 ShareSet 将被创建，并应用于当前对象。

相关主题

- [删除 Attenuation ShareSet](#)
- [将 Attenuation ShareSet 作用于对象](#)
- [将 Attenuation ShareSets 转换为自定义实例](#)

删除 Attenuation ShareSet

如果您不再需要某个 Attenuation ShareSet，则可以删除它。在删除某个共享集之前，您应确保没有对象还在用到它。如果删除 ShareSet，则它会自动从采用它的所有对象中移除。



备注

Attenuation Editor (衰减编辑器) 中的 Shared by (共享对象) 框将显示使用当前 Attenuation ShareSet 的对象的完整列表。

删除 Attenuation ShareSet 的方法是：

1. 在 Project Explorer 中，切换到 ShareSets 选项卡。
2. 在 Attenuation 区域，点击要删除的 ShareSet。
3. 按 Delete 键。

ShareSet 被删除，并从采用它的所有对象中移除。

相关主题

- [创建 Attenuation ShareSet](#)
- [将 Attenuation ShareSet 作用于对象](#)
- [将 Attenuation ShareSets 转换为自定义实例](#)

将衰减应用到对象

在为工程创建了 Attenuation ShareSets 后，便可以将它们应用到对象上了。这些实例既可以专用，也可以共享，具体取决于您希望它们的属性是作用于一个对象还是多个对象。

在应用衰减时，可以执行下列其中一项操作：

- 将 Attenuation ShareSet 作用于对象
- 将 Attenuation ShareSets 转换为自定义实例

将 Attenuation ShareSet 作用于对象

如果需要为对象设置衰减，您则可以对它们应用 Attenuation ShareSet。多个对象可以同时采用同一个 ShareSet，当需要更改时，这可以为您节省大量的时间。

将 Attenuation ShareSet 作用于对象的方法是：

1. 将对象加载到 Property Editor 中。
2. 切换到 Positioning 选项卡。
3. 在 Attenuation 分组框中，点击 选择器 按钮 (>>)。



Attenuation 共享集列表将显示。

4. 选择要应用的 Attenuation 共享集。

ShareSet 的名称出现在相应文本框中，表明 ShareSet 已应用到对象上。



相关主题

- [将设置从一个平台复制到另一个平台](#)
- [创建 Attenuation ShareSet](#)
- [删除 Attenuation ShareSet](#)
- [将 Attenuation ShareSets 转换为自定义实例](#)

将 Attenuation ShareSets 转换为自定义实例

Wwise 中的衰减分为两类：

- 自定义衰减可应用于层级结构中的任意一个对象。在您更改自定义效果器的属性时，只有这个对象会受到影响。
- Attenuation ShareSets 可应用于工程层级结构中的许多对象。在更改 ShareSet 的属性时，使用该 ShareSet 的所有对象都将受到影响。

在默认情况下，您创建的所有衰减初始都为 ShareSets。然而，您可以根据您的喜好将 Attenuation ShareSet 转换成自定义衰减。更改后，对衰减的所有修改将只作用于采用它的对象。

将衰减作用于对象的方法是：

1. 将对象加载到 Property Editor 中。
2. 切换到 Positioning 选项卡。
3. 在 Attenuation (衰减) 分组框的 Mode (模式) 列表中，选择以下选项之一：
 - Use ShareSets (使用 ShareSets)，以将 ShareSet 应用于当前对象。
 - Define custom (自定义)，对当前对象做自定义衰减。



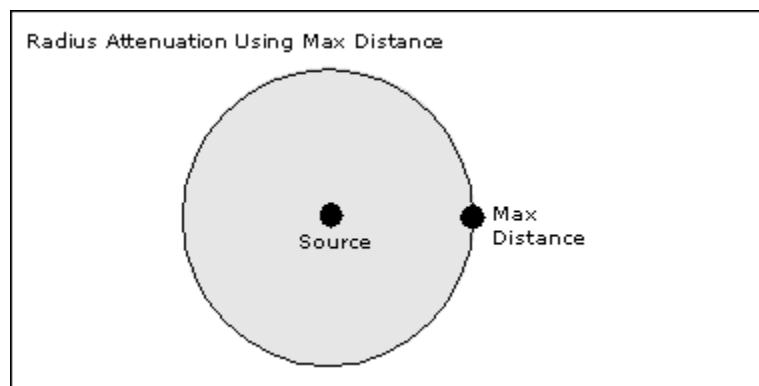
相关主题

- [创建 Attenuation ShareSet](#)
- [删除 Attenuation ShareSet](#)
- [将 Attenuation ShareSet 作用于对象](#)

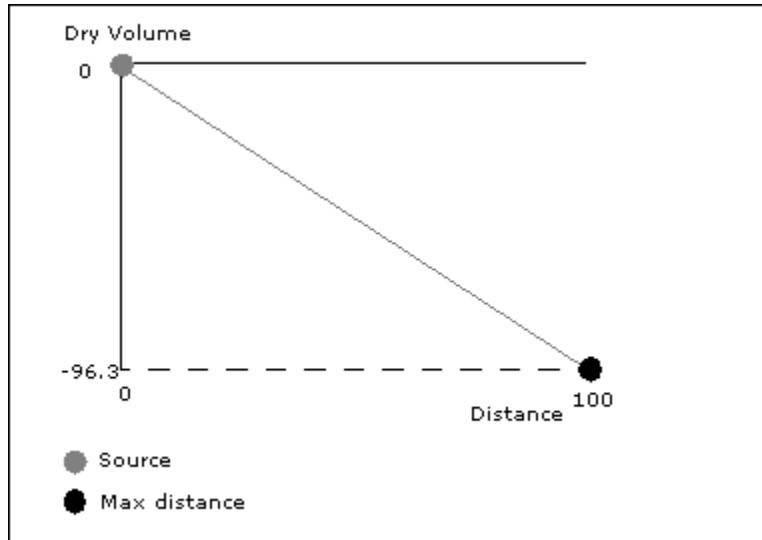
定义各种对象属性的衰减曲线

为了在 Wwise 中创建对象的衰减，您可以创建一系列曲线来定义 Wwise 中特定属性之间的关系（例如音量和低通滤波器），以及游戏中点声源到听者的距离。

各条曲线使用最大距离值来定义对象的最大衰减点。由于声音来自于全向声源，因此以最大距离值为半径可以在每个声源周围形成一个球形衰减范围。

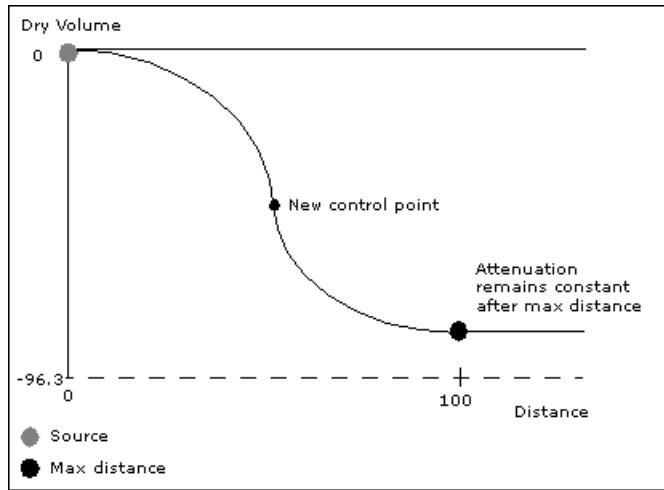


在默认情况下，对象的衰减方法是从声源开始直到最大距离间采用线性插值。



虽然这可能适用于大多数情况，但特定对象可能需要更高级的曲线。为了更好地控制衰减曲线，您可以添加控制点。[<片段0477>](#)这些点可以将衰减曲线分段，以便您更好地控制对象的衰减。

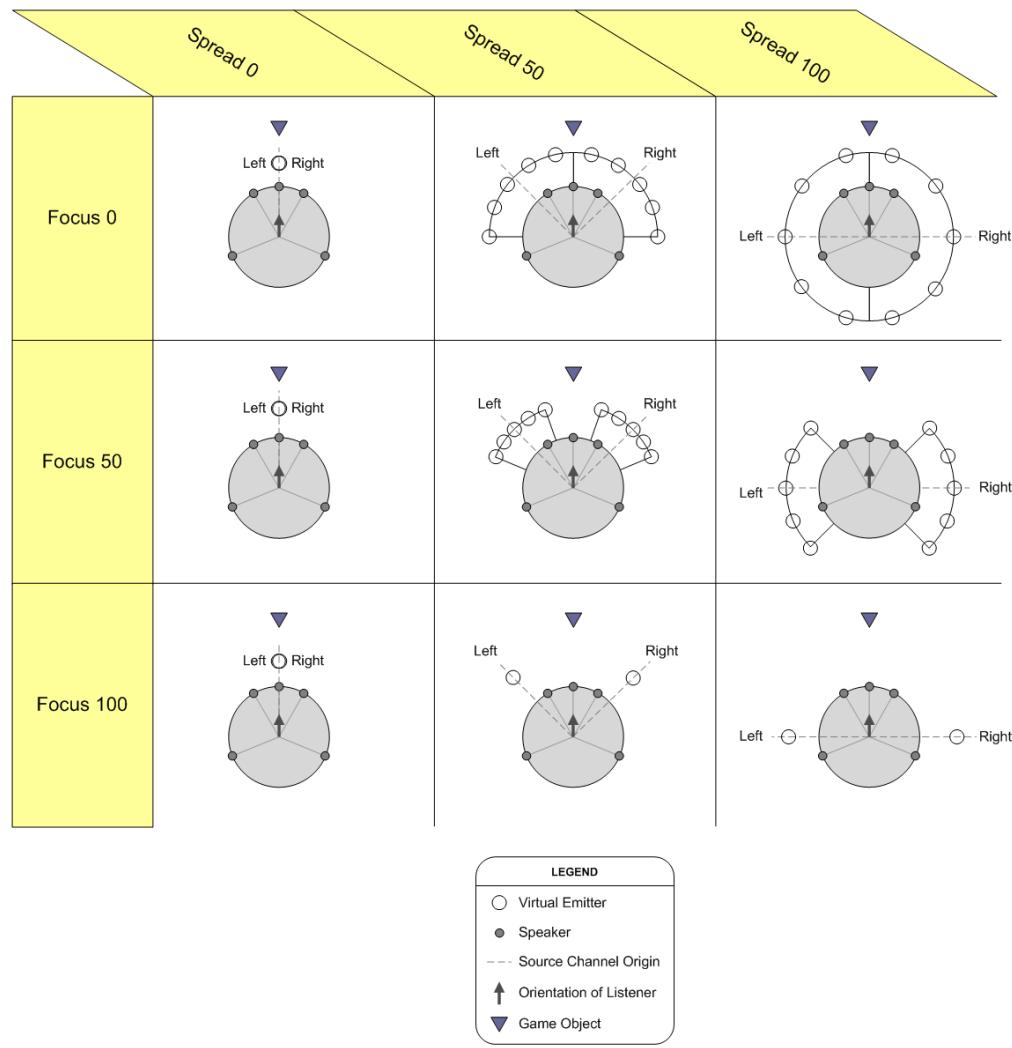
为了创建更加详细和复杂的定位曲线，您还可以定义各条曲线段的形状。曲线段是两个控制点之间的部分。您可选择各种曲线形状，包括线性曲线、恒定曲线、对数曲线、幂数曲线和 S 曲线。要获取指定曲线形状的详细信息，以及使用图形视图的其它信息，请参阅 [第 43 章 了解坐标图视图](#)。



您可为以下 Wwise 属性创建衰减曲线：

- Output Bus Volume -- 输出总线音量。连接到音频输出总线的信号的衰减或振幅。
- Auxiliary Send volumes - 发送到游戏定义和用户定义的 Auxiliary Bus 的信号的衰减或幅度。
- Low-pass filter —— 根据指定值来衰减高频的递归滤波器。低通滤波器的单位代表已经应用的低通滤波比例，0 表示无低通滤波（信号不受影响），100 代表最大衰减。

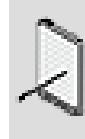
- **High-pass filter** —— 高通滤波器。根据指定值来衰减低频的递归滤波器。高通滤波器的单位代表已应用的高通滤波比例，其中 0 代表无高通滤波（信号不受影响），100 代表最大衰减。
- **Spread** —— 散布。扩散到附近扬声器的音频量或百分比，以使声音能够随着距离的增加从低扩散的点声源变为完全扩散的传播源。值 0 表示某扬声器附近的发声体的所有声道只通过该扬声器播放。值 100 表示将扩散声源的声道，以便通过所有扬声器都能够听到或感觉得到。
- **Focus** —— 聚焦。百分比值，用于收缩由扩散值生成的虚拟发声体。焦距 0% 表示虚发声体保持不变，值越高，各个虚点越靠近源声道的原点。



衰减属性值是相对的，这意味着被衰减影响的对象属性值是衰减值与该对象的现有属性值之和。

定义衰减曲线的方法如下：

1. 将对象加载到 Property Editor 并切换到 Positioning 选项卡。
2. 在 Attenuation (衰减) 分组框中单击选择器，然后选择或创建 Attenuation ShareSet (衰减共享集)。



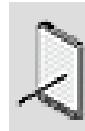
备注

若要激活 Attenuation 分组框，则须启用 Listener Relative Routing。

3. 单击 Edit... (编辑...)。

Attenuation Editor 打开，其中包含选定衰减的属性设置。

4. 在 Max distance (最大距离) 文本框中，指定从源点到声音达到最大衰减时的距离。

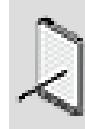


备注

在达到最大距离值后，衰减设置保持不变。

5. Curves (曲线) 分组框中，从列表中选择 Output Bus Volume (输出总线音量) 曲线。

默认的Output Bus Volume曲线于是显示在坐标图视图中了。

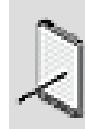


备注

曲线上的一点始终代表点声源，最末一点始终代表最大距离值。

6. 要控制 Output Bus Volume 的 Attenuation 曲线，可以执行以下任一操作：

- 在曲线上添加点。
- 将控制点拖至新位置，也可在 X 和 Y 坐标框中输入特定值。
- 定义曲线段的形状。



备注

有关缩放或平移坐标图视图、同时显示多条曲线、添加、移动或删除点、指定标度法或者更改曲线段形状的信息，请参阅[第 43 章 #####](#)。

7. 对于剩余的曲线，从 Curve 列表中选择以下其中一个选项：

- None - 对相应的属性，不使用 Attenuation 曲线。选择“None”时，相应属性不会衰减，将保持完整长度。
- 采用 Use Output Bus Volume (使用输出总线音量)，则对相应属性使用同一条 Output Bus Volume 曲线。此选项仅可用于 Auxiliary send volumes (辅助发送音量) 曲线。
- Custom - 为相应的属性创建自定义 Attenuation 曲线。

相关主题

- [将设置从一个平台复制到另一个平台](#)
- [使用 3D 空间化对象](#)
- [应用基于距离的衰减](#)
- [定义各种对象属性的衰减曲线](#)

- 使用锥形边界模拟方向性
- 将音频信号传送到中置扬声器
- 缩放和平移坐标图视图
- 添加控制点
- 定义坐标图视图的坐标显示方式
- 指定控制点之间曲线的形状

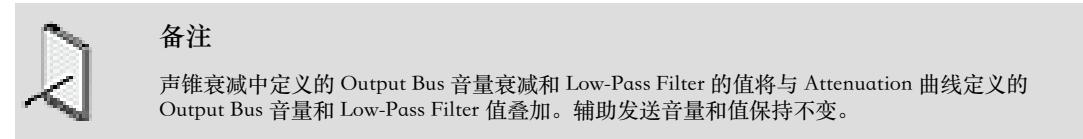
使用锥形边界模拟方向性

在默认情况下，Wwise 中的声音来自于全向声源。然而在现实中，声音通常具有一定的方向性。在 Wwise 中模拟声音的方向可以使用声锥。声锥使用不同的夹角模拟声音在特定方向上的传播。当听者移到这些夹角外面时，Output Bus volume（用作干分量）将衰减。由于 3D Game-defined 定位使用实时游戏数据，因此声锥的方向最终由游戏对象的朝向来控制。

以下角定义声锥的区域：

- **Inner angle:** 内角。此夹角定义的区域中输出总线音量不衰减，也没有低通滤波效果发生。
- **Outer angle:** 外角。此夹角定义的区域中输出总线音量发生衰减，低通滤波效果保持在最高水平。

Output Bus Volume 在过渡区域（也即内外角边界之间的区域）中发生滚降。在无衰减发生的内角边界和达到最大衰减值的外角边界之间，使用线性插值法对音量进行衰减。在外角定义的区域中，音量衰减始终等于最大衰减值。



位于 Cone Attenuation 属性旁边的 Attenuation Preview（衰减预览）显示为您提供声锥内不同区域的视觉表示。当您更改内外角值时，预览视图会自动进行更新。

模拟振动的方向

虽然振动来自于全向振动源，但在游戏中，有时候您需要模拟振动在特定方向上的传播。在这些情况下，可以使用 Cone Attenuation 属性。声锥作用于振动和声音的方式是相同的，例外情况是 Low-Pass Filter 设置不作用于振动对象。声锥根据听者相对于振动源朝向的位置，利用不同的角度来减弱振动。

使用声锥模拟方向性的方法是：

1. 将对象加载到 Property Editor 并切换到 Positioning 选项卡。
2. 在 Attenuation（衰减）分组框中，选择或创建 Attenuation ShareSet（衰减共享集）。



3. 单击 **Edit...**（编辑...）。

Attenuation Editor 打开，其中包含选定衰减的属性设置。

4. 选择 **Cone Attenuation** 选项。

声锥控件变为可用。

5. 在 **Inner angle** 文本框中，指定一个角度，该夹角定义的区域中将不会发生 Output Bus 音量衰减。

6. 在 **Outer angle** 文本框中，指定一个角度，该夹角定义的区域中 Output Bus 音量的衰减和 Low-Pass Filter 效果将保持在最大值。



备注

内外角之间的区域被称为过渡区域。在此区域中，即无衰减发生的内角边界和达到最大衰减角的外角边界之间，使用线性插值法衰减输出总线音量。

7. 在 **Max attenuation** 文本框中，可以设置当物体落在外角范围内时，Output Bus 的衰减量。

8. 要对位于内外角之间的声音应用 Low-Pass Filter，在 **Low-Pass Filter** 文本框中输入一个值。低通滤波器是根据指定值来衰减高频分量的递归滤波器。



备注

低通滤波器的单位代表已经应用的低通滤波比例，0 表示无低通滤波（信号不受影响），100 代表最大衰减。

相关主题

- [使用 3D 空间化对象](#)
- [应用基于距离的衰减](#)
- [定义各种对象属性的衰减曲线](#)
- [将音频信号传送到中置扬声器](#)

预览声音的衰减设置

在设置声音的衰减属性后，您可以直接在 Wwise 中预览设置。这可以让您直观地了解到当玩家在声音衰减半径内转悠时将听到的声音效果。通过直接在 Wwise 中预览声音衰减的功能，您可以在将任何音频集成到游戏中前先优化和微调您的设置。

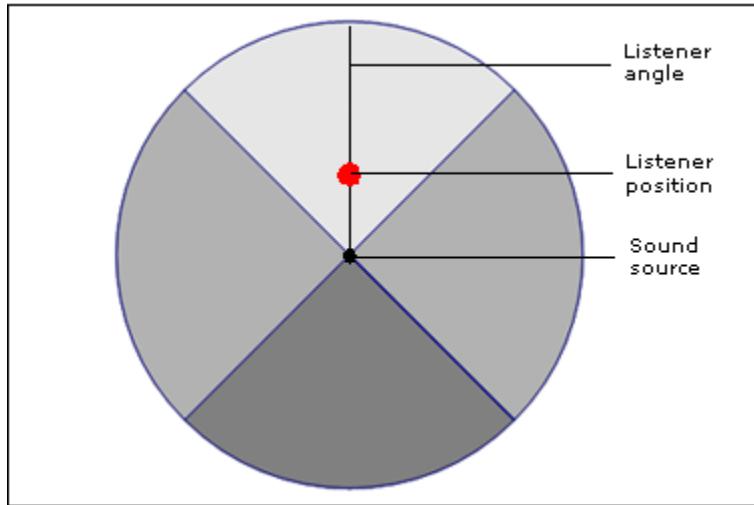


备注

只有使用 Emitter（发声体）定位的声音允许预览衰减设置。

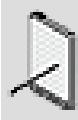
要预览衰减设置，您需要能够改变听者在声音衰减半径内的位置。这是通过使用 **Attenuation Preview**（衰减预览）控件来实现的，这些控件位于 Attenuation Editor（衰减编辑器）中。Attenuation Preview 是声音衰减半径的图形表示。声音的衰减半径由最大距离值决定。声源始终位于圆圈的正中央。使用以下两个控件可修改听者相对于声源的位置：

- 红色小圆圈代表听者相对于声源的实际位置。此红色圆圈可以随意地移到衰减半径中的任何位置。当您将红色圆圈从中心位置移开时，声音的音量将减弱，继续移动直到圆圈的外边缘上时达到最大衰减量。通过指定听者的位置，您可以自动确定听者到声源的距离。
- 黑色细线: 代表听者位置相对于声源的角度。在默认情况下，听者位于声源正前方（0 角度），但您可以通过拖动黑色细线做圆周运动来修改角度。您可以通过将在 360 度 (-180 到 180) 全方位内变动角度来模拟环绕声源的移动。



在播放期间可通过同时修改角度和位置控件来模拟听者相对于声音的移动。

您可以重置听者的位置和角度，方式是在按住 Ctrl 键的同时，点击衰减半径内的任何位置，或者点击 Transport Control（播放控制）中的 Reset Selector (>>)（重置选择器）按钮，然后选择 Reset Position（重置位置）。



备注

您还可以修改听者相对于声源的位置，方式是拖动坐标图视图中的 Distance 光标。

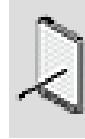
预览声音衰减设置的方法是：

1. 将对象加载到 Property Editor 并切换到 Positioning 选项卡。
2. 在 Attenuation（衰减）下单击选择器，然后选择 Attenuation ShareSet（衰减共享集），接着单击 Edit...（编辑...）。

Attenuation Editor（衰减编辑器）打开了。

3. 在 Transport Control（走带控制）中，单击 Play（播放）图标来播放声音。
4. 要修改听者的位置，将红色小圆圈拖到衰减半径内的任何位置。

声音此时听起来就好像听者正在环绕声源移动。衰减程度将随着您靠近或远离声源而改变。



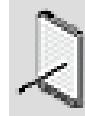
备注

通过指定听者的位置，您可以自动定义听者到声源的距离。您也可以通过拖动坐标图视图中的 Distance 光标移动听者来靠近或远离声源。

5. 若要只更改听者的角度而非距离，则需执行以下操作之一：

- 点击衰减半径中的任何位置。
- 拖动黑色细线做圆周运动。

声音此时听起来就好像听者正在环绕声源移动。如果声锥衰减开启，则衰减将随听者通过声锥的不同区域而变化。更改角度时，距离不会改变。其结果是，声音不会随着距离而衰减。



备注

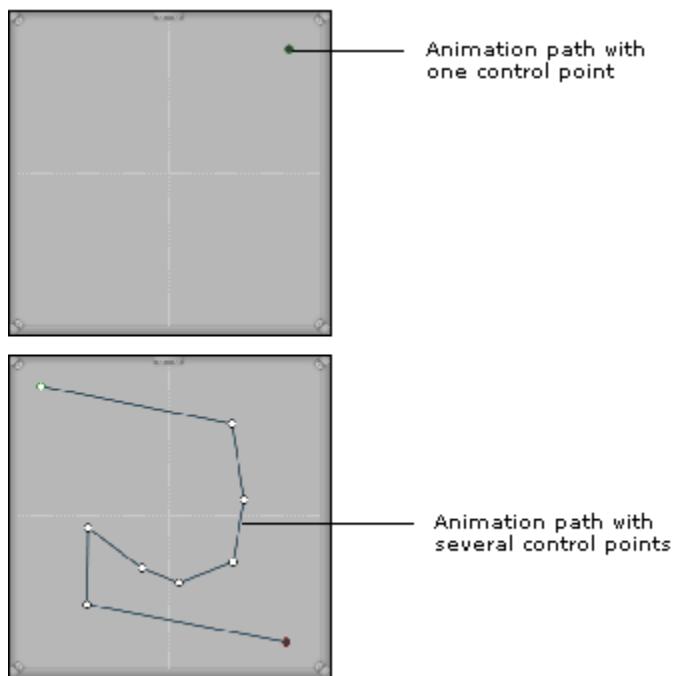
若要重置听者的位置和角度，则要在按住 Ctrl 的同时点击最大衰减半径中的任何位置，或者点击 Transport Control 中的 Reset Selector (>>) 按钮，然后选择 Reset Position。

使用动画路径定义空间定位

若想为玩家打造独特的体验，可使用 Automation（自动化）来定义 3D 对象的空间位置。当您在 Wwise 中预定义位置时，无论听者在游戏中的位置和朝向如何，都会遵循以下情况：

- 声音将总是通过相同的扬声器播放。
- 通过相同的电机感受到振动。

使用动画路径定义定位信息。动画路径由每次定义源位置的若干个控制点构成。如果创建了多个点后，对象则将随着时间沿路径进行运动。



自动化控件定位 - 示例

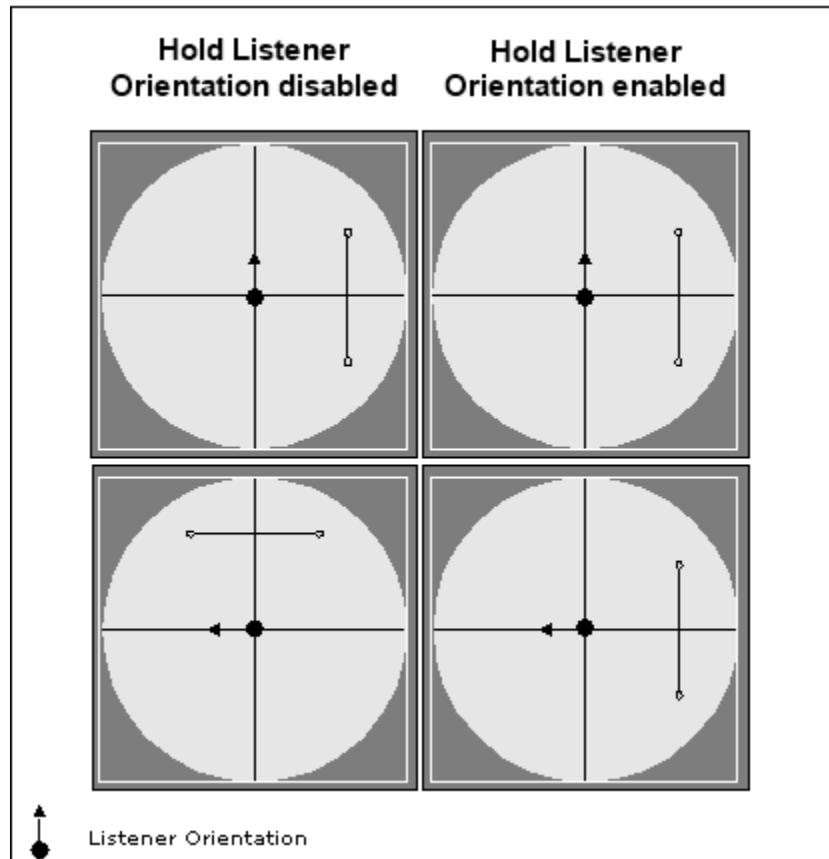
假设您正在创建一款角色扮演游戏，故事发生在船上。您想在环境声中听到海鸥叫声。您希望海鸥叫声听起来好像它们正在船边飞翔。为此，可将定位设为 Emitter with Automation（发声体自动化），然后创建多个不同的声音路径，以便模拟鸟儿围绕船只的飞行路线。您可以让海鸥叫声听起来更加逼真，方法是使用 Attenuation ShareSet 来衰减声音。

Automation 可用于游戏中的各种对象，包括非固定位置的环境声，如丛林环境中的虫鸣声、鸟叫声和猴叫声。

创建顺着听者朝向的动画路径

在使用 3D Automation（3D 自动化）时，可选择将动画路径的位置锁定到游戏中听者的朝向。当两者锁定时，无论听者处于什么朝向，都将始终通过相同的扬声器听到声音。当两者不锁定时，听者将独立于路径移动。这意味着当听者转身时将通过不同的扬声器听到声音。

下图显示了在启用和禁用 Hold Listener Orientation（固定听者朝向）选项的情况下，听者改变朝向时对动画路径产生的影响。



您可以在对象 Property Editor（属性编辑器）的 Positioning（定位）选项卡中设置 Hold Listener Orientation 选项。因为无法在 Wwise 设计工具内更改听者朝向，所以只能在游戏中试听差异。

使用动画路径

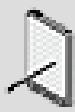
您可以定义对象的定位，方法是使用一个声源点或多个控制点来创建路径。如果创建了路径，则对象将在指定的一段时间上沿路径进行动画显示。为增加功能和提高灵活性，您可以为同一对象创建多条路径，修改各条路径的时长，然后根据指定的顺序或完全随机的顺序来播放它们。

使用坐标图视图和时间线创建动画路径。有关如何移动/删除控制点，以及如何在坐标图和时间线中进行缩放/平移，具体信息请参阅[“第 43 章 了解坐标图视图”](#)。

创建动画路径

在定义对象的定位前，必须创建动画路径。动画路径由若干个控制点构成。如果使用多个点，则这些点会形成一条动画路径，对象会在指定时间内沿着这条路径运动。

这些路径在坐标图视图中创建。坐标图视图以最大距离值的比例显示各个控制点的位置。这意味着各个控制点的值将位于 0 和 +/- 100 之间，具体取决于点在坐标图的哪个象限。



备注

在创建第一条路径时，Wwise 会自动将它命名为“ObjectName_01_Path”。您可以随时重命名此路径，为它取一个更加有意义的名称。然而各条路径不得重名。

创建动画路径的方法是：

1. 在 Position Editor (3D Automation) 中，单击 **New** (新建)。

列表中新增一条动画路径，则坐标图视图中和时间线上也增加一个控制点。

2. 在 Path Name (路径名称) 文本框中，输入描述性的名称。
3. 将路径的第一个点拖到坐标图视图中的任何位置，对其进行定位。
4. 要添加更多的点，则双击坐标图视图。

坐标图视图中的这些点将自动相连，形成一条动画路径。



技巧

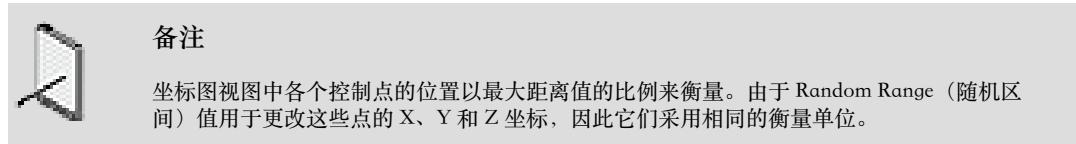
您可以微调任意控制点的位置，方法是在坐标图视图中拖动任意点或直接在 X 和 Y 坐标文本框中输入该点的坐标值。另外，您还可以更改点之间的间隔时间，方法是在时间线中移动点。

相关主题

- [沿路径随机设置各个点的位置](#)
- [更改路径时长](#)
- [在坐标图视图中显示衰减半径和声锥](#)
- [重新排序动画路径列表](#)
- [删除动画路径](#)
- [确定播放动画路径的方式](#)

沿路径随机设置各个点的位置

为提高游戏的逼真度和避免重复，您可以轻松地沿各条路径随机设置各个点的位置。为此，您只需要在水平和垂直平面中为各个点定义一个可能值区间即可。在运行时，Wwise 从这些区间中提取随机值来指定控制点的实际位置。



沿路径随机设置各个点的位置的方法是：

1. 在 Position Editor (3D Automation) 中，从路径列表选择路径。
2. 在 Random Range 分组框中，设置以下两个属性值：
 - Horizontal -- 横向。代表各个点 X 和 Y 坐标值的 +/- 偏移量，用于创建可能值区间。在运行时，Wwise 从此区间选择值来定义控制点的横向位置。
 - Vertical -- 纵向。代表各个点 Z 坐标值的 +/- 偏移量，用于创建可能值区间。在运行时，Wwise 从此区间选择值来定义控制点的纵向位置。

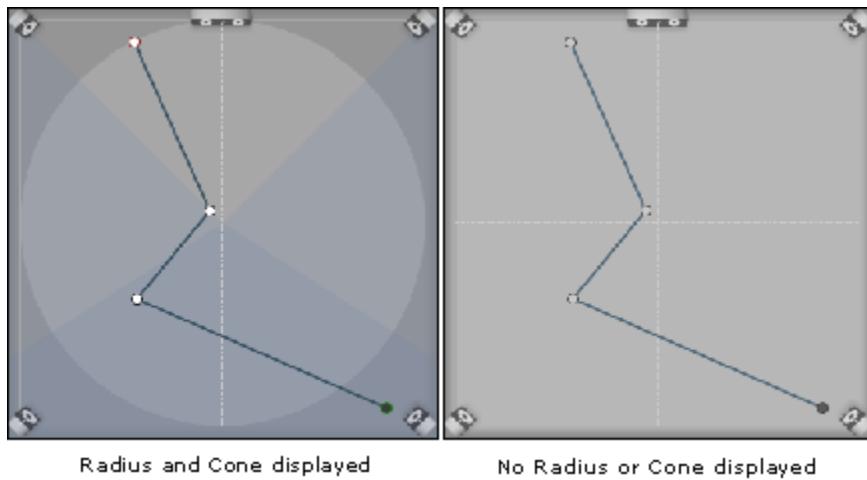


相关主题

- [创建动画路径](#)
- [更改路径时长](#)
- [在坐标图视图中显示衰减半径和声锥](#)
- [重新排序动画路径列表](#)
- [删除动画路径](#)
- [确定播放动画路径的方式](#)

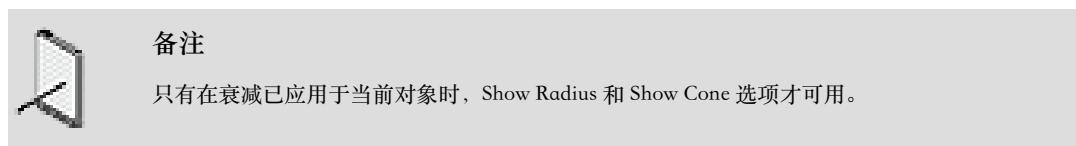
在坐标图视图中显示衰减半径和声锥

如果使用衰减来模拟信号远离听者时的自然衰减，则可能需要使用最大距离半径和声锥作为参考，以便您在坐标图视图中创建路径时能够查看不同的衰减区域。



更改坐标图视图显示的方法是：

1. 在 Display Option (显示选项) 分组框中，选择以下选项之一：
 - Show Radius (显示半径)，以在坐标图视图中显示衰减最大距离半径。
 - Show Cone (显示声锥)，以在坐标图视图中显示声锥角度（内角和外角）。



相关主题

- [创建动画路径](#)
- [沿路径随机设置各个点的位置](#)
- [重新排序动画路径列表](#)
- [删除动画路径](#)
- [确定播放动画路径的方式](#)

更改路径时长

您创建的每条路径可具有不同的时长。要更改路径的时长，必须更改时间线的长度。如果时间线上沿线的现有点间隔不均匀（非线性模式），则必须指定是在时间线上按比例分布这些点还是将它们保留在现有位置上。

更改动画路径时长的方法是：

1. 在 Path (路径) 列表中，点击您要更改其时长的路径。
2. 点击 Configure Timeline (配置时间线) 按钮。

Configure Timeline 对话框于是打开了。

3. 在 Length (长度) 文本框中，为选定路径指定新的时长。
4. 如果时间线中已经包含控制点，并呈非线性模式，则必须选择以下其中一项：

- Stretch proportionally（按比例拉伸），以重新定位现有控制点，并保持它们之间在时间线上的相对位置。
 - Preserve key values（保留关键值），以将控制点保持在它们在时间线上的现有位置上。超出新时间线长度的控制点将被删除。
5. 点击 OK。

动画路径的长度于是更新了。时间线也发生改变，以反映新的时长。

相关主题

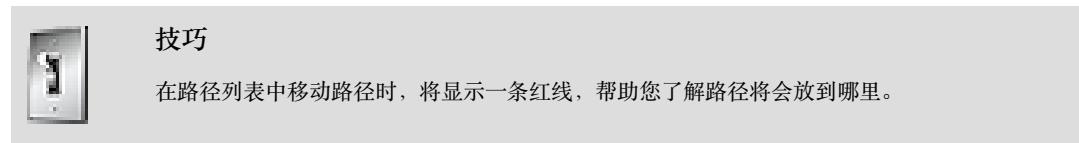
- [创建动画路径](#)
- [沿路径随机设置各个点的位置](#)
- [重新排序动画路径列表](#)
- [删除动画路径](#)
- [确定播放动画路径的方式](#)

重新排序动画路径列表

对于动画路径列表中的路径，如果您想要它们以特定顺序显示或播放，则可以对它们重新排序。

重新排序路径列表的方法是：

1. 在路径列表中，点击您要移动的路径。
2. 将选定路径拖到新位置。



相关主题

- [创建动画路径](#)
- [沿路径随机设置各个点的位置](#)
- [更改路径时长](#)
- [删除动画路径](#)
- [确定播放动画路径的方式](#)

删除动画路径

如果您不再需要路径，则可从列表中移除它。

删除动画路径的方法是：

1. 在 Path 列表中，点击您要删除的路径。
2. 点击 Delete。

确认消息框出现。

3. 点击 OK。

路径于是从列表中移除了。

相关主题

- [创建动画路径](#)
- [沿路径随机设置各个点的位置](#)
- [更改路径时长](#)
- [重新排序动画路径列表](#)
- [确定播放动画路径的方式](#)

确定播放动画路径的方式

在创建路径后，必须决定如何播放它们。与容器一样，您可以决定是随机播放还是按照路径列表中的指定顺序播放创建的路径。您还可以决定当播放对象时是逐一播放所有路径还是每次只播放一条路径。

确定播放动画路径方式的方法是：

1. 在 Position Editor (3D Automation) 中，将播放类型设为以下选项之一：
 - Sequence (序列)，从头到尾按序列播放列表中的路径。
 - Random (随机)，随机播放路径，直至播完所有路径。
2. 在 Play Mode 组框中，选择以下选项之一：
 - Continuous —— 逐一播放路径，直至播完所有路径。
 - Step —— 在每次播放对象时从列表中选择一条路径。
3. 如果选择了 Continuous，则 Loop (循环) 和 Transition time (过渡时间) 将变为可用。

要无限期地播放整个动画路径列表，选择 Loop 选项。

要在一条路径末端和另一条路径开端之间添加一个线性过渡，则选择 Transition Time 选项，并在相应的文本框中指定过渡时间量。

4. 如果选择了 Step，则 Pick new path when sound starts 选项将变为可用。

选择 Pick new path when sound starts 选项将强制 Wwise 在每次播放新声音使用新路径，无论声音是否由播放事件触发。当您希望对连续容器中的各个声音使用不同的路径时，此选项特别有用。

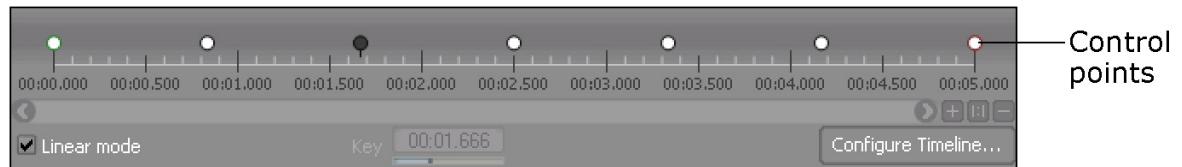
相关主题

- [创建动画路径](#)
- [沿路径随机设置各个点的位置](#)
- [更改路径时长](#)
- [重新排序动画路径列表](#)

- **删除动画路径**

处理时间线中的点

在创建动画路径时，需要定义声音沿路径传播的时间。Position Editor (3D Automation) 时间线方便指定各个控制点所对应的时间。



您可以沿时间线来移动控制点，从而定义对象在点之间沿路径运行的速度。您可以移动和删除点，但不能直接在时间线上添加点。控制点只可在坐标图视图中添加。您还可以同时选择多个点来移动或删除整条线段。



备注

您可以在时间线上缩放和平移控制点，以便更加准确地定位控制点。

更改时间线模式

Position Editor (3D Automation) 中的时间线可设置为以下两种模式之一：

- **Linear:** 线性。控制点沿时间线自动均匀地放置。
- **Non-linear:** 非线性。控制点可放在时间线上任何位置。

当对象沿路径运动时，从一个节点到下一个节点，线性模式将产生匀速的运动。而非线性模式可用于当对象在两点之间沿路径运动时加快和减慢对象的速度。线性模式是默认的时间线模式。

点击 Configure Timeline 按钮可进一步配置时间线的特定行为。有关详细信息，请参阅[配置定位时间线](#)。

更改时间线模式的方法是：

1. 在 Position Editor (3D Automation) 中，执行以下操作之一：
 - 取消选中 Linear 选项以将时间线模式设为非线性。
 - 选择 Linear 选项以将时间线模式设为 Linear。

选择时间线中的控制点

在移动或删除时间线中的控制点前，必须先选中控制点。您每次可选择一个、多个或所有控制点。

选择时间线中一个控制点的方法是：

1. 在时间线上，点击控制点以选中它。

所选控制点将变为黑色。

选择时间线上的多个控制点的方法是：

1. 在时间线上，画矩形来选中您要选择的点。

所选的点将变为黑色。



技巧

在按住 Ctrl 的同时点击各个点也可以选择多个点。

选择时间线中全部控制点的方法是：

1. 点击时间线任何位置来激活它。
2. 按下 Ctrl+A。

时间线中的所有点于是都被选中并变成黑色。

相关主题

- [移动时间线中的控制点](#)
- [删除时间线中的控制点](#)

移动时间线中的控制点

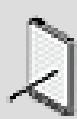
当时间线处于非线性模式时，您可以移动控制点来更改各个点之间的时间。这会让对象在各个控制点之间实现加速或减速移动的效果。一个控制点不可越过它前后的控制点。

移动时间线上控制点的方法是：

1. 在时间线上，选择若干个控制点。

所选控制点将变为黑色。

2. 将选中点拖到时间线上的新位置。



备注

直接在 Key 字段中输入值也可以移动点。如果选中了多个点，您在 Key 字段中输入的正值 (+) 或负值 (-) 则将使这些点偏移到它们初始位置的右侧或左侧。

相关主题

- [选择时间线中的控制点](#)
- [删除时间线中的控制点](#)

删除时间线中的控制点

除第一个控制点外，您可以删除时间线中的任意控制点。

从曲线删中除控制点的方法如下：

1. 在时间线中选择若干个点。

选中的控制点变成黑色了。

2. 按 Delete 键。

控制点于是被从时间线和坐标图视图中移除了。

相关主题

- [选择时间线中的控制点](#)
- [移动时间线中的控制点](#)

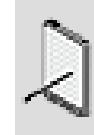
将音频信号传送到中置扬声器

如果有些声音对游戏的玩法至关重要的话，那么您可能要确保玩家始终能够清楚地听到这些声音。为达到这一目的，您可以将 2D 或 3D 对象的任何比例传送到中置扬声器。例如，赛车游戏中赛车手或者解说员的语音可以完全从中置扬声器中发出来，这样不管游戏中其它声音有多吵，这些语音都能听得清清楚楚。

Center % 设置

下表描述在各种 Center % 设置中，多少信号将传送到中置扬声器：

设置	Center %
0%	创建幻像中置图像——没有信号通过中置扬声器。
1-99%	不同的信号量通过中置扬声器。随着 Center % 增加，左前和右前扬声器的音量降低。
100%	创建独立的中置声像——信号被全部传送到中置扬声器。

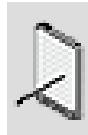


备注

扩散到其它扬声器的信号量多少是基于声源相对于不同扬声器的位置来决定的。

将音频信号传送至中置扬声器的方法是：

1. 将顶层父对象加载至属性编辑器。
2. 切换到 Positioning 选项卡。



备注

如果该对象不是顶层对象，则必须选择 **Override parent** 选项后才能设置 Positioning 选项。

3. 在 Center % 文本框中，可以为信号设置需要通过中置扬声器的音量百分比。

相关主题

- [使用 Speaker Panning](#)

- 使用 3D 空间化对象
- 应用基于距离的衰减
- 使用动画路径定义空间定位

定位技巧和经验总结

在 Wwise 中定义对象的定位之前，您最好仔细阅读以下各节。这些章节提供了一系列的示例、技巧和最佳方案，可帮助您更好地管理游戏中对象的定位。

定位——示例（第 2 部分）

在详细描述 Wwise 中可用的不同定位选项后，现在让我们了解不同的选项可如何用来为我们的第一人称[定位——示例（第 1 部分）](#)中的声音和振动效果定义定位。

- **Footsteps:** 脚步。这是一款第一人称游戏，因此主人公的脚步声应始终伴随着摄像头。因为这些声音没有移动和衰减，所以在这种情况下将基本的 Speaker Panning（扬声器声像摆位）设置为 Direct Assignment（直接指派）较为合适。然而，对于其它的特工，您需要将声音附加到“agent”游戏对象上，使脚步声与他们的动作相匹配。在这种情况下，结合使用 3D Spatialization（3D 空间化）和 Emitter（发声体）定位较为合适；不过，不需要启用衰减。
- 照亮敌人丛林基地的火把的燃烧声 —: 这些声音附加到“torch”游戏对象上。虽然它们固定在一个地方，但发声体的位置及其到话筒的距离将随着玩家的移动而改变。若要模拟这种类型的声音，可结合使用 3D 空间化、Emitter 定位和衰减。
- 一群恐怖分子在屋内交谈的声音 —: 这些声音附加到“terrorist”游戏对象上。这些对象可以在游戏环境内自由移动。若要模拟这种类型的声音，可结合使用 3D 空间化、Emitter 定位和衰减。
- 蚊子在头顶上飞舞的嗡嗡声：可听到蚊子在周围飞舞时发出的嗡嗡声，但看不到蚊子。因为发声体必须在 3D 空间内移动，所以在这种情况下 3D 空间化使用 Emitter with Automation（发声体自动化）或 Listener with Automation（听者自动化）定位较为合适。同时使用空间化和衰减的一系列随机播放声音路径可以创建极其逼真的蚊虫声音。在选择 Listener with Automation 选项时，虽然没有实际游戏对象，但会有嗡嗡声随玩家移动。当然，也可选择 Emitter with Automation 选项，并使用“蚊子”游戏对象来定义能听到嗡嗡声的死水潭区域。
- 从总部接收最新情报的声音：从总部接收情报的通信声既不与任何特定游戏对象关联，也不在周围环境中移动，因此在这种情况下将 Speaker Panning 设置为 Direct Assignment 较为合适。由于最新情报对于完成任务非常关键，因此您还可能需要将某些或全部声音传送到中置扬声器。
- 执行任务的特工们之间的耳语声：队友之间的耳语声将关联至各自的游戏对象，因此这些声音使用 3D 空间化较为合适。特工们相互掩护依次行动，需要某种空间定位，但他们既然必须协同作战，因此他们之间的沟通不需要任何衰减。由于队友之间的沟通对于完成任务非常关键，因此您还可能需要将某些或全部声音连通到中置扬声器。
- 在成功完成任务后用于炸毁基地的炸药的爆炸声 —: 特工将听到和感受到炸药的爆炸声。这些声音和振动对象将被绑定在游戏中会爆炸的对象上。虽然它们固定在一个地方，但发声体/振动体的位置及其到听者的距离将随着玩家的移动而改变。若要模拟这种类型的音效，可结合使用 3D 空间化、Emitter 定位和衰减。
- 岛上火山持续发出的轰鸣声 —: 火山发出的轰鸣声是该偏远岛屿上持续发出的声音和振动效果。声音和振动对象最好附加在“island”游戏对象上。添加一些衰减可在

玩家靠近该岛时，让轰鸣声听起来更加响亮或密集。因为声音和振动没有移动，所以在这种情况下不需要启用空间化。

- 火山最后喷发声：**爆炸导致火山爆发。在最终场景中，玩家坐在运输直升机后排逃离危险之地。火山喷发会产生震耳欲聋的环境声，您可以使用 Listener with Automation 定位选项进行 3D 空间化处理，并应用衰减设置。在 Position Editor (3D Automation) 中，可创建一个或多个路径，来反映直升机和玩家如何上下颠簸、迂回前行，并在烟雾弥漫的风暴中艰难地躲避四溅的碎屑，最终顺利飞越火山口。我们可以启用 Hold Listener Orientation，从而通过不同的扬声器来播放火山喷发声，并反映玩家（听者）所处位置。假如采用多扬声器配置，还可将 3D Spatialization (3D 空间化) 选项设为 Position + Orientation (位置 + 朝向)，从而表现出直升机不断变换朝向和火山喷发声逐渐衰减，增强真实性。
- 互动音乐：**因为音乐不与任何特定游戏对象关联，且无需在环绕声环境中移动，所以使用 Speaker Panning 较为合适。在示例中，我们希望对部分 Music Track (音乐轨) 进行声像摆位，在前后扬声器之间平衡音乐的音量。



请参考下表，了解在本例中可用于创造不同声音的定位选项的完整概述。

声音	Speaker Panning		Attenuation (衰减)	3D Spatialization		
	Direct Assignment	Balance-Fade		Emitter (发声体)	Emitter with Automation	Listener with Automation
特工的脚步	✓					
火把			✓	✓		
恐怖分子交谈			✓	✓		
蚊子嗡嗡声					✓	
总部发来的最新情报	✓					
特工沟通				✓		
爆炸			✓	✓		
火山轰鸣			✓	✓		
火山喷发声			✓			✓
互动音乐		✓				

此示例描述使用 Wwise 中可用的不同选项创建不同定位和传播类型的一种方式。您的最终选择应该取决于音效本身、游戏需求以及希望创造的特定效果。

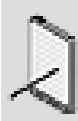
性能优化

- 在不使用 Attenuation 散布曲线时，请使用单声道声音。若不打算使用散布曲线来拓宽音频信号，则应使用单声道声音来优化性能。不使用散布时，立体声的所有输入声道将映射到同一位置，并且不得不进行动态渲染；而使用单声道时，将离线操作，游戏期间不会占用任何 CPU 资源。

- 请复用或减少 Attenuation Editor（衰减编辑器）中的曲线数量，以便提高性能。记住，在 Attenuation Editor 中创建的曲线越多，占用的运算资源和内存就越多。要提高性能，既可以复用（辅助发送音量的）Output Bus Volume（输出总线音量）曲线，也可不使用任何曲线。
- 使用少量的点和线性曲线段来提高性能。记住，沿曲线添加的点越多，曲线形状越复杂，占用的运算资源和内存就越多。在大多数情况下，对于带有两到三个点的曲线，使用线段就足以达到您所需要的衰减结果。
- 使用 ShareSet 共享衰减属性设置。若游戏中多个对象具有相似的衰减属性，则可使用 ShareSet 来共享这些属性设置。通过共享衰减属性设置，在更改衰减属性时可以同时节省内存和时间。
- 使用定位类型 RTPC 可以为类似的用途复用声音。例如，可以使用相同的声音层级结构将玩家的脚步声设置为 2D，敌人的脚步声设置为 3D。在 Default Pool（默认内存池）中，这可以节省大量的内存。

特定定位情景的概述

让我们一起来了解某些特定情景，让您更好地理想 Wwise 中不同定位和衰减设置的工作原理。



备注

在默认情况下，不会通过中置扬声器播放声音。要将信号的任何部分传送到中置扬声器中，使用 Center % 属性滑杆。

情景 1

- Listener Relative Routing: Enabled
- 衰减: 无
- 3D Spatialization: None

结果：这些设置可以有效地为您提供与 2D 声音相同的定位。

情景 2

- Listener Relative Routing: Enabled
- 衰减: 简单线性曲线



- 3D Spatialization: None

结果：这些设置可以在听者远离声源时有效地降低声音音量（衰减），但声音始终处于它在原始声音素材中的准确位置上，没有应用任何定位或旋转（空间化）。

情景 3

- Listener Relative Routing: Enabled
- 衰减: 无

- 3D Spatialization: Position

结果：无论听者距离声源多远（衰减），这些设置可以有效地为您提供一个源自特定位置（空间化）但音量永不衰减的声音。

情景 4

- Listener Relative Routing: Enabled
- 衰减：简单线性曲线



- 散布：简单线型曲线



- 3D Spatialization: Position
- 声源：单声道

结果：这些设置可以有效地达到以下效果：

- 当听者距离声源较远时，声音来自特定位置（空间化），音量较低，主要在一个扬声器中播放（散布）。
- 当听者距离声源较近时，声音来自于特定位置（空间化），接近于最大音量（衰减），并几乎平均分布在两个扬声器之间（散布）。

情景 5

- Listener Relative Routing: Enabled
- 衰减：简单线性曲线



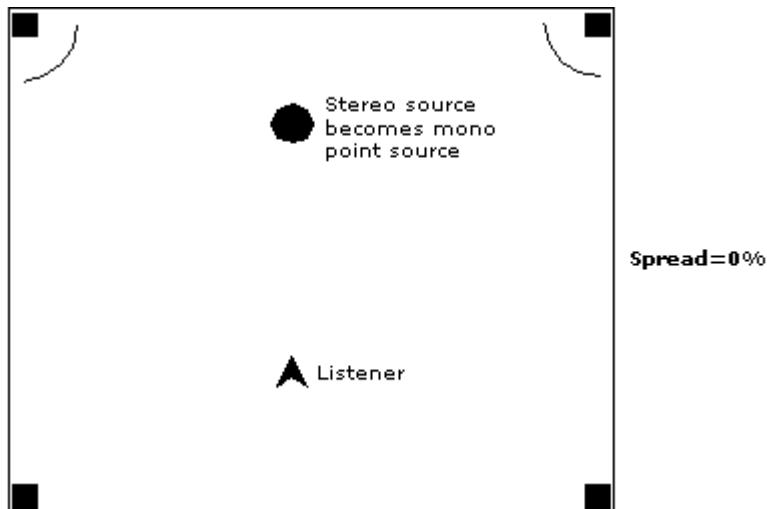
- 散布：简单线型曲线



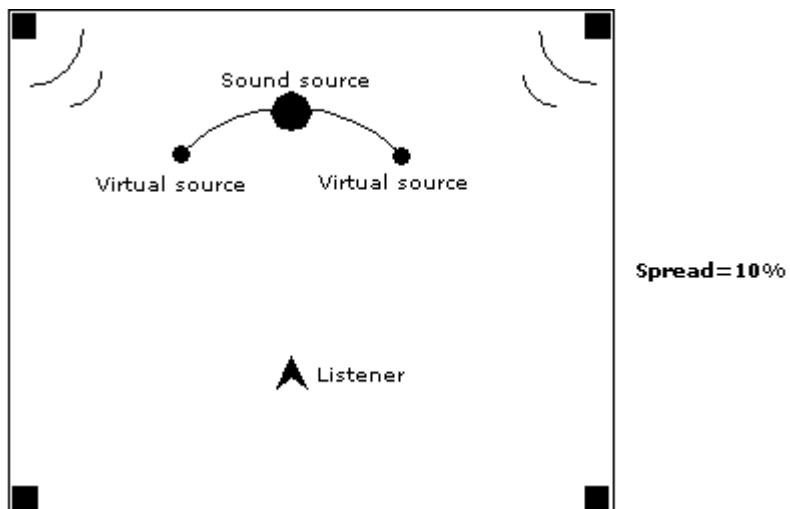
- 3D Spatialization: Position
- 声源：立体声

结果：这些设置可以有效地达到以下效果：

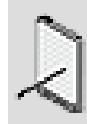
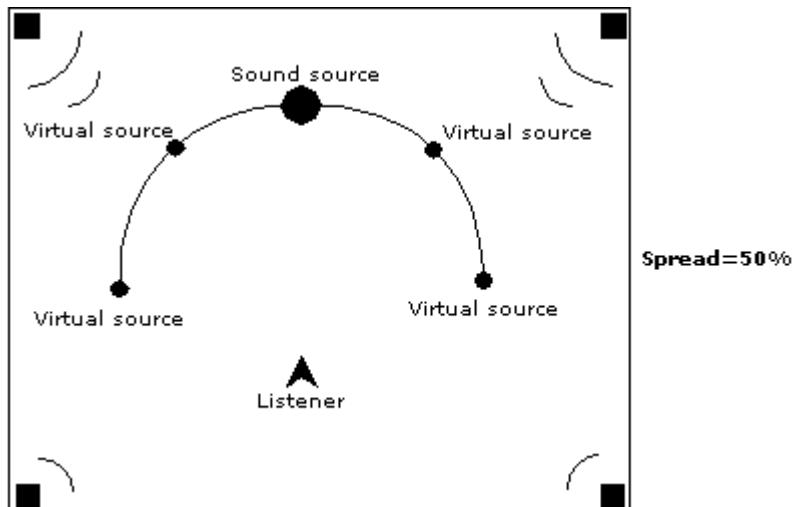
- 听者距离声源较远时，声音来自特定位置（空间化），音量较低（衰减）。对于不带散布的、使用空间化的立体声源，两个声道合并形成一个单声道“点声源”。为此，我们建议在没有散布时使用单声道文件，因为这样做可以更加高效地利用CPU。



- 使用散布时，新的“虚声源”是指偏离原始声源的声源。例如，对于小散布值，计算得到的虚声源将位于实际位置的左侧和右侧，它们的贡献将添加到扬声器中，与正常的无散布声音无异，唯一不同的是位置稍微不同。



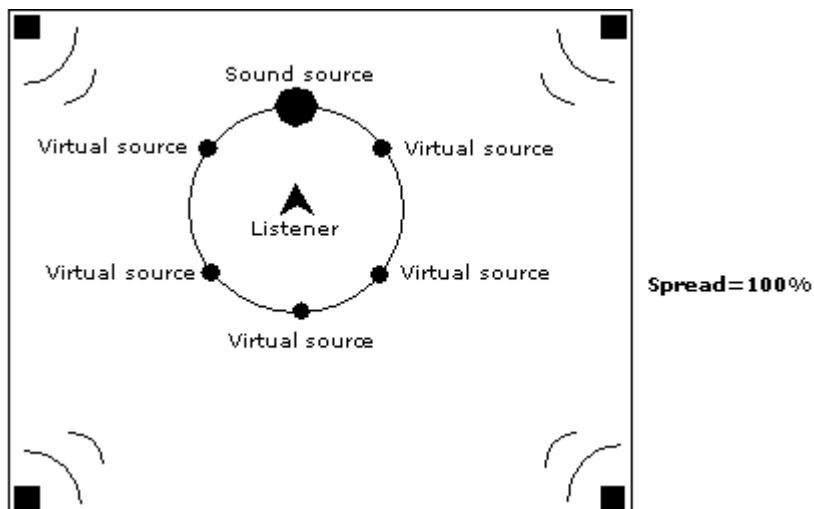
随着散布值增加，用于覆盖听者周围大弧度的虚声源数量将更多。显然，为了保持恒定的总功率，这些声源的功率将低于实际声源。



备注

注意，这些声源仅用于音量计算；实际上并没有播放新的声音。

- 当听者距离声源较近时，声音来自于特定位置（空间化），接近于最大音量（衰减）。使用高散布值时，声音将来自于所有方向。立体声的左右声道将独立散布。



- 现在，需要特别注意距离 = 0 的情形。在 Wwise 中，所有空间化计算（和声锥衰减）都以角度为基础。当距离 = 0 时，Wwise 无法确定听者是朝前、朝左、朝右还是其它朝向。应避免在游戏中发生这种情形。如果在游戏中发生了这种情形，Wwise 则将创建立体声的单声道版本来避免计算越界音量。这一逻辑同样适用于声锥衰减。如果听者的朝向未知，Wwise 则将假定不存在任何衰减。这同样适用于声锥 LPF。

理解声道配置

Wwise 管线支持不同级别的多声道配置，具体取决于源素材、输出设备、总线设置等。不同的平台也存在不同的限制。



下混

为确保在处理不同声道配置时获得最佳性能，理解 Wwise 的下混方法非常重要。有关详细信息，请参阅[附录 A, #####](#)。

以下是 Wwise 中常见的不同声道配置，其中列出了声道名称、建议的扬声器角度和顺序：

0.1 (单声道)	
界面元素	描述
低频效果声道 (LFE)	不适用

1.0 (单声道)	
界面元素	描述
中置 (C)	0 度角

2.0 (立体声)	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角

3.0	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角
中置 (C)	0 度角

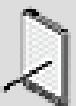
4.0	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角
环绕声左声道 (SL)	90-110 度角
环绕声右声道 (SR)	90-110 度角

5.1	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角
中置 (C)	0 度角
环绕声左声道 (SL)	90-110 度角
环绕声右声道 (SR)	90-110 度角
低频效果声道 (LFE)	不适用

7.1	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角
中置 (C)	0 度角
环绕声左声道 (SL)	90-110 度角
环绕声右声道 (SR)	90-110 度角
左后声道 (BL)	135-150 度角
右后声道 (BR)	135-150 度角
低频效果声道 (LFE)	不适用

Wwise 还支持具有“高度”扬声器的扩展标准声道配置。以下是所有可用声道的定义，以及总线上可用标准配置的描述。

具有高度声道的标准配置扩展	
界面元素	描述
左 (L)	22-30 度角
右 (R)	22-30 度角
中置 (C)	0 度角
环绕声左声道 (SL)	90-110 度角
环绕声右声道 (SR)	90-110 度角
中后 (BC)	180 度
左后声道 (BL)	135-150 度角
右后声道 (BR)	135-150 度角
顶部 (T)	正上方
高度左前 (HFL)	同 L，高出水平线
高度中前 (HFC)	同 C，高出水平线
高度右前 (HFR)	同 R，高出水平线
高度左后 (HBL)	高度左后 (HBL)
高度中后 (HBC)	高度中后 (HBC)
高度右后 (HBR)	同 BR (或 SR)，高出水平线
低频效果声道 (LFE)	不适用

可用的带高度扩展的总线配置	
Dolby™ 总线配置	
	备注 Wwise 支持的 Dolby 配置包括向 5.1 环绕声和 7.1 环绕声配置中添加的高度声道。产生的新标注中增加了第三位数，它表示高度声道的数量。
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

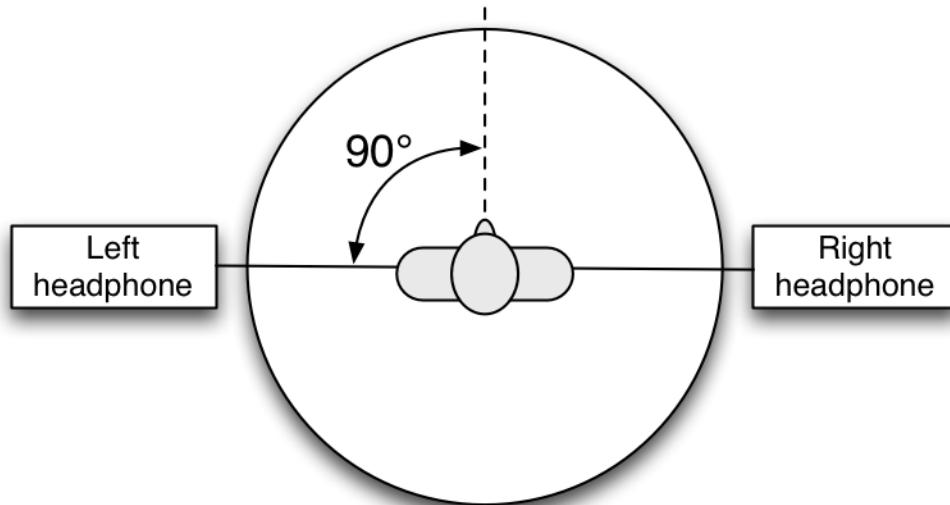
可用的带高度扩展的总线配置	
Auro 总线配置	
	备注
	Wwise 支持的 Auro 配置包括向 5.1 环绕声和 7.1 环绕声配置中添加的高度声道。产生的新标注中声道总数增加了——第一位数分布 9.1 到 11.1 之间，表示增加了 4 到 6 个高度声道。Auro 13.1 表示向 7.1 环绕声配置中添加了 6 个高度声道。
Auro 9.1	L + R + C + SL + SR + HFL + HFR + HBL + HBR + LFE
Auro 10.1	L + R + C + SL + SR + T + HFL + HFR + HBL + HBR + LFE
Auro 11.1	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

可用的 Ambisonics 总线配置	
	
备注	
	Ambisonics 使用独立于扬声器设置的 B-format 声场进行工作。以下描述中的字母标记表示 B-format 各方向声道，而不是扬声器声道。这个表示法基于 ACN 通道排序和 SN3D 归一化，但是——就像使用 MaxN 归一化的 FuMa 排序那样——它使用字母而不是数字。有关详细信息，请参阅 Ambisonics 通道排序 。
First Order Ambisonics	W-Y-Z-X
Second Order Ambisonics	W-Y-Z-X-V-T-R-S-U
Third Order Ambisonics	W-Y-Z-X-V-T-R-S-U-Q-O-M-K-L-N-P

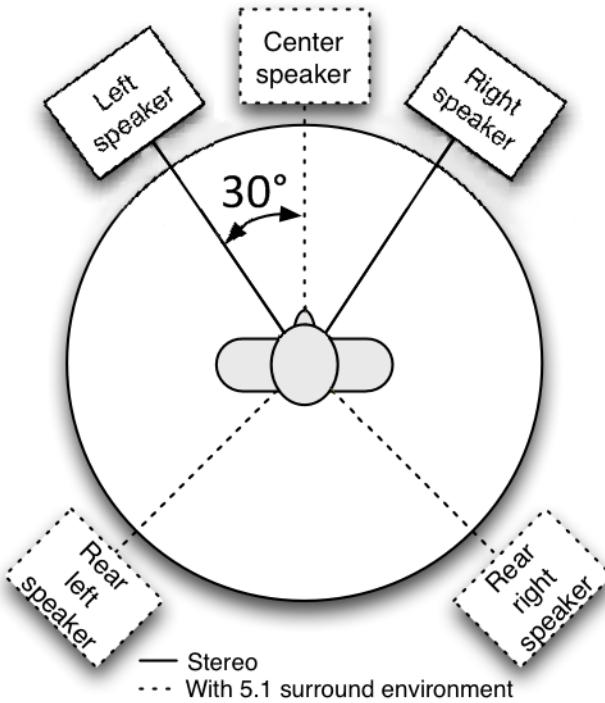
声像摆位规则：扬声器对耳机

在 Wwise 中有两种不同的摆位规则：耳机和扬声器。默认情况下，除了便携设备（例如 Android 和 iOS）使用耳机摆位规则之外，所有平台均使用扬声器摆位规则。两种模式之间的差异很小，但有助于提供逼真、准确的音频体验；具体取决于您的聆听方式。此设置可以在 Wwise 中试听，也可以在游戏中运行时设置。

Headphone panning rule

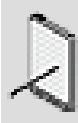


Loudspeaker panning rule



试听这两种模式的方法是：

- 在菜单栏中点击 Audio > Stereo Channel Configuration (Headphones)
- 在菜单栏中点击 Audio > 5.1 Channel Configuration。



备注

要在游戏中设置平移规则，请参阅 SDK 文档中的 [AK::SoundEngine::SetPanningRule](#)。

使用 Ambisonics

Ambisonics 是一种环绕声技术，可以覆盖水平面以及听者上方和下方的区域。B-format 声场通过球谐函数来表示，能够独立于扬声器配置发挥效果。可以想见，这样就能轻松实现声音跟随听者的旋转。对于声音设计师所需的更加全方位的环绕声表现形式，比如在提供环境声或进行 VR 研发时，Ambisonics 都是可行的选择。<片段 1100>



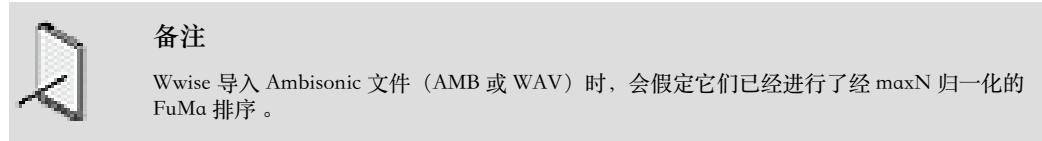
技巧

除了本页和以下页面中的 Ambisonics 帮助材料，您还可以在博客中找到我们发布的文章：[Wwise 中的 Ambisonics：概述](#)。文中深入探讨了 Ambisonics 的理论以及如何在 Wwise 中使用 Ambisonics，用于 VR 声音研发。另外，我们在 Wwise Sample 项目中提供了一些 Ambisonics 的简单使用示例，如 [文档](#) 中所述。

在 Wwise 中可以轻松导入 Ambisonics 文件，将其他声道配置混音为 Ambisonics 格式，并最终在任何支持 Ambisonics 的平台上播放 Ambisonics 音频输出。

使用 Ambisonic 文件的方法如下：

1. 导入录制的 B-format Ambisonic 文件，该格式像其他音频文件一样，可以保存为 WAV 或 AMB。有关详细信息，请参阅[导入媒体文件](#)。



2. 与其他声音对象一样，请指定适当的 Audio Bus 作为 Output Bus。请参阅[指定对象的输出连线](#)了解详细信息。

Wwise 将根据需要为各声道混音。有关详细信息，请参阅[对 Ambisonics 进行子混音](#)。

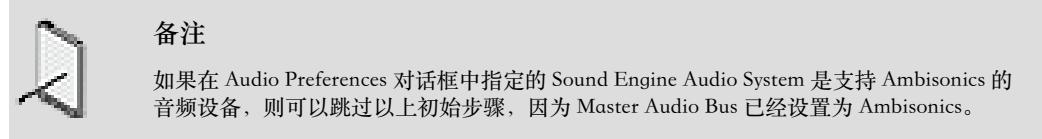
创建 Ambisonics 输出的方法如下：

1. 将 Audio Bus 设置为以下 Ambisonics 声道配置：

- First Order Ambisonics (一阶 Ambisonics)：一阶 4 声道
- Second Order Ambisonics (二阶 Ambisonics)：二阶 9 声道
- Third Order Ambisonics (三阶 Ambisonics)：三阶 16 声道

有关声道配置的更多信息，请参阅[Available Ambisonics Bus Configurations table](#)。

对于通过上述 Audio Bus 输出的源文件，Wwise 会将其混音并适配 Ambisonic 输出。



2. 指定一个支持 Ambisonic 声道配置的最终输出设备，例如 3D 双耳系统。

如果不是支持 Ambisonic 的输出，Wwise 会自动将相应的声道输出进行混音，并适配可用的标准输出。有关详细信息，请参阅[对 Ambisonics 进行子混音](#)。

将 Wwise 引擎的最终输出设置为 HOA 输出

如上所述，当总线的 Channel Configuration 设置为 Ambisonics #-# 时，Wwise 将输出 Ambisonic 和 HOA (higher-order ambisonic，即高阶 Ambisonics) 信号。`<片段1127>` 这些信号将被解码，意味着只要 Ambisonic 总线的父总线声道为标准配置时，它们就会被转换成标准的多声道表示法，例如立体声或 5.1。

Master Audio Bus 会继承最终输出的配置，这取决于平台设置，通常由自动检测得到。目前，大多数平台只允许少数几种输出配置，通常是立体声、5.1 和 7.1。所以一般来说，Master Audio Bus 具有标准的声道配置，这会迫使子 Ambisonic 总线解码为这种配置，也就是说 Wwise 不会直接输出 HOA。

但是，可以编写自定义 Audio Device 插件，来声明 Ambisonics 配置。在这种情况下，主 Audio Bus 将采用 Ambisonics 声道配置；因此，Wwise 将保留 HOA 信号而不将其解码。

另外，如果您只需要 Wwise 将 HOA 信号转存到文件中，则只需在 Ambisonic 总线上插入 [Wwise Recorder](#) 即可。生成的文件将采用 AMB 格式 (FuMa)，其 Ambienonic 阶数等于 Recorder 所在总线的阶数。

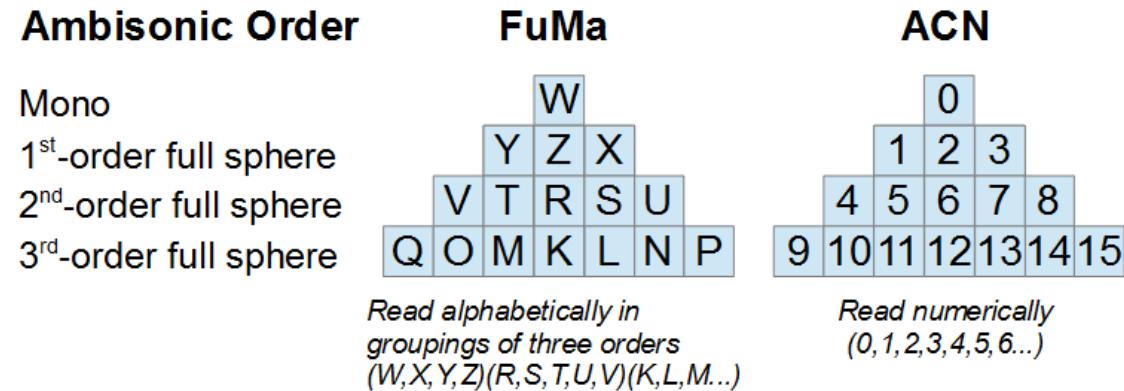
相关主题

- [Ambisonics 通道排序](#)
- [在插件中使用 Ambisonics](#)
- [对 Ambisonics 进行子混音](#)
- [Ambisonics 的局限性](#)

Ambisonics 通道排序

用于 Ambisonics 的通道排序格式主要有两种，包括通常称为 FuMa 的 Furse-Malham，和通常称为 ACN 的 Ambisonics Channel Number。如下图所示，前者即 FuMa 使用字母顺序标记 - 每组中按照字母排序 - 从 W (omni) 声道开始，接着到右下角，然后是左下角，然后是下方中心；然后它移动到下一组并从 R 开始，然后到右边，再到它左边，向右两格，然后再向左两格；然后移动到下一组并遵循相似的模式。相比之下，后者即 ACN 的编号顺序从左到右，更容易跟踪。

从顶部通道到第 1、2、3 阶排序，图中还表明各阶 Ambisonic 可以看作分别具有 4, 9 和 16 声道的金字塔结构。<片段1143>



Wwise 中的 Ambisonic 排序

在 Wwise 中，Ambisonic 文件的排序与初始导入文件或可能导出的文件不同。虽然 Wwise 期望导入的 Ambisonic 文件使用传统的 FuMa 排序，但到了 Wwise 中，它们将转换为经过 SN3D 归一化的新 ACN 排序。但是，它保留了 FuMa 排序字符。也就是说，在前面的图像中，FuMa 排序将从左到右读取，对应于 W Y Z X V T R S U Q O M K L N P。电平表以及设计工具中的 Source Editor 图像都将使用这种遵循 ACN 排序的混合显示法。

这也意味着自定义插件的开发可以使用这种经 Wwise 转换的 ACN 排序。对于在 Ambisonics 总线上用 Wwise Recorder 插件录制的文件，可使用 FuMa 排序（从而生成可导入 Wwise 的文件）或 ACN 排序。这是由 Ambisonic Channel Ordering (Ambisonics 声道排序) 选项决定的。

相关主题

- 在插件中使用 Ambisonics
- 对 Ambisonics 进行子混音
- Ambisonics 的局限性

在插件中使用 Ambisonics

Wwise 插件 API 使用 `AK_ChannelConfigType_Ambisonic` 暴露 Ambisonics 声道配置。因此，在自定义插件研发过程中，开发者可以使用 Ambisonic Bed。

使用 Ambisonics 格式效果器

Wwise 中所有独立处理声道的效果器都能支持 Ambisonics，如 Wwise Compressor，Wwise Guitar Distortion 和 Wwise Parametric EQ 插件。<片段1158>



相反，混响效果器对于声道的处理则不同。它们通常会将输入声道下混，进行混响处理，然后将或多或少的去相关信号混合到输出声道中，从而进行上混和混响输出。Wwise RoomVerb 和 Wwise Convolution Reverb 可以应用于 Ambisonics 声部和总线。它们使用 W（全向）声道，将其进行混响处理，并通过将去相关信号混合到全向和定向声道来进行上混。Wwise Matrix Reverb 如果在 Ambisonic 声部或总线上应用，将无法初始化。

相关主题

- Ambisonics 通道排序
- 对 Ambisonics 进行子混音
- Ambisonics 的局限性

Ambisonics 的局限性

Ambisonics 用来表示从听者周围空间向其传来的音频。因此，它无法正确表示场景中的对象位置。这使得朝向和围绕所述声音对象移动时的处理变得困难，声音事件似乎总是从远处发声。

这对 Wwise 内的 Ambisonic 表现有两个重要的影响：

- 如果设置了 Attenuation 曲线，当您远离相关游戏对象的位置时，整个声场的音量都会减小。当进入和离开声场时，可能会使事情变得复杂。
- 如果您移动到声场的一侧，这一侧的声源听起来并不会比相反方向的声源更响。

Ambisonics 与 VBAP 比较

Wwise 中 3D 声音的默认声像摆位算法基于普遍适用的 VBAP 算法，该算法可以在声音整体能量不变的情况下得到最高的声音准确度。但是，这也最大限度地减少了能

量散布的变化。也就是说，当虚声源位置与扬声器刚好对准时，能量散布最小，当它恰好位于相邻扬声器形成的弧线（标准配置）或三角形（具有高度声道的配置）中央时，能量散布最大。

相比之下，无论声源方向和扬声器布局如何，Ambisonics 都具有恒定的能量散布。Ambisonics 的阶数越低，散布越大。因此，一阶 Ambisonics 非常不精确；而三阶 Ambisonics 较为精确，但仍不如 VBAP 精确。

出于美学考虑，可以将单声道声源进行编码，制作 Ambisonics Bed。

相关主题

- [Ambisonics 通道排序](#)
- [在插件中使用 Ambisonics](#)
- [对 Ambisonics 进行子混音](#)

对 Ambisonics 进行子混音

和其他声道配置一样，Ambisonics 中的声源定位基于混音时游戏对象和听者的位置。因此，Ambisonic 声音对象可以与其他配置进行子混音；但是如果这样做，可能会失去声音对象的空间信息。例如，如果子混音配置是 7.1，那么当 Ambisonics 音频总线与之混音时，将会失去所有的高度信息。同样，对于 7.1.4 子混音，原先位于听者耳朵下方的对象听起来将位于齐耳高度。因此，建议使用 Ambisonics 配置进行子混音。如果空间精度不那么重要，您可以使用较低的阶数。

路由行为

Wwise 会根据不同配置的路由方式，自动将音频转换为 Ambisonics，或从 Ambisonics 转换为其他配置。最终输出取决于使用的不同声道配置，以及定位类型是 2D 还是 3D。根据这些因素，下表提供了路由行为相关的一般信息。

Routing (信号通路)	2D 定位行为	3D 定位行为
标准配置转换为 Ambisonics	<p>该信号被编码为 Ambisonics。</p> <p>单声道信号源将直接编码到 W（全向）声道。</p> <p>对于多声道声源，每个声道都被视为一个单声道源，入射角度由声道位置决定。例如，对于立体声源，会将其左声道编码为一个单声道源，位于左侧 45 度，右声道同样编码为单声道源，位于右侧 45 度。输入声道角度的详细列表，以度为单位（顺时针）表示：</p> <ul style="list-style-type: none">• L: -45°• R: 45°• C: 0°• SL: -90°• SR: 90°• BL: -135°• BR: 135° <p>• 高度声道的高度角为 +45 度。</p>	<p>根据游戏对象相对于听者的位置和朝向，信号将被编码为 Ambisonics。</p> <p>对于多通道文件，Spread 和 Focus 的处理方式与标准配置完全相同：以虚声源代表各输入声道，并且基于游戏对象、听者、聚焦和散布值将虚声源部署在听者周围，然后进行编码。请参阅定义各种对象属性的衰减曲线了解更多有关 Spread 和 Focus 的细节。</p>

Routing (信号通路)	2D 定位行为	3D 定位行为
	使用给定的入射角对声源进行编码非常简单。	
Ambisonics 转换为 Ambisonics	各阶 Ambisonics 之间转换时将直接进行扬声器分配。如果输入的阶数较高，则会丢弃额外的声音。如果输入的阶数较低，则输出的额外通道将保持静音。	Ambisonic Bed 将根据游戏对象和听者的相对方向进行旋转。当输入和输出阶数不同时，旋转是按较低阶数计算的，较高阶数将被静音。
Ambisonics 转换为标准配置	根据游戏指定的扬声器位置，Ambisonics 信号将被解码为输出配置。有关如何设置扬声器位置的信息，请参阅 SetSpeakerAngles() (位于 AK::SoundEngine 中)。	Ambisonic Bed 会根据游戏对象和听者的相对方向进行旋转，然后像 2D 定位一样解码为标准配置输出。



注意

如果将 2D 单声道源编码到 W (全向) 声道，而不是将它们视为正前方的声源，会导致播放异常。当 2D 单声道声源被路由到 Ambisonic 总线，该总线又输出至标准配置总线时，这将导致单声道声源在所有的输出声道中播放。这与 Wwise 中的典型表现不同，通常 2D 单声道源仅发送到 C 或 L 和 R 声道。

相关主题

- [Ambisonics 通道排序](#)
- [在插件中使用 Ambisonics](#)
- [Ambisonics 的局限性](#)

第 12 章 管理优先级

概述	285
了解 Wwise 如何为对象排列优先级	286
限制对象的播放实例	289
定义播放优先级	292
管理音量较低的对象	293
优先级相关技巧和经验总结	295

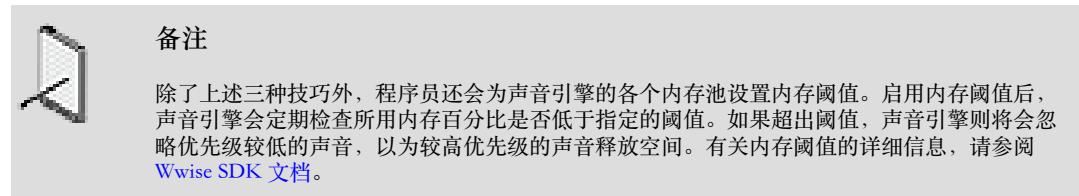
概述

在游戏中，您可以同时播放很多对象，对象的数量甚至有可能超出由项目团队设置的数量上限。为了有效管理播放对象的数量，您必须规定同时最多可以播放多少个对象，以及哪些对象会被优先播放。

在 Wwise 中主要有三个属性可以帮助您确定在游戏中将同时播放哪些对象：

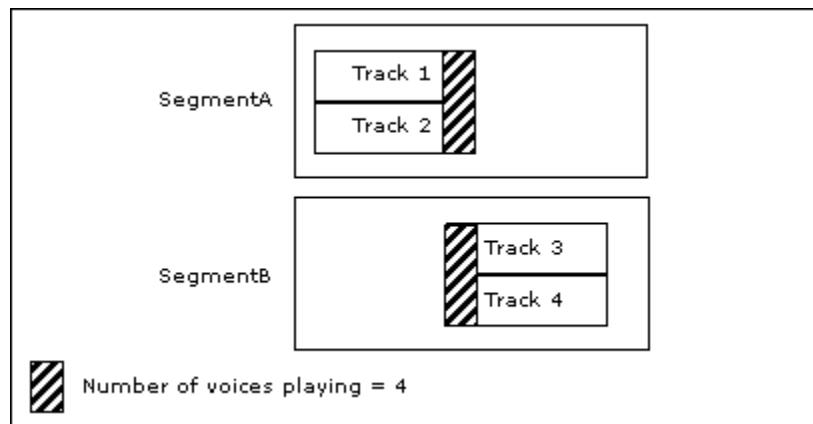
- **Playback limit** —— 限制同时允许播放的对象数量。（不包含虚声部。）
- **Playback priority** —— 一个对象相对于另一个对象的重要性。
- **Volume threshold** —— 低于这一特定音量的对象将不会播放。

通过设置限制、指派优先级和指定最小音量，您可以高效而创新地管理游戏内的多个对象，同时遵循所设定的内存限制。

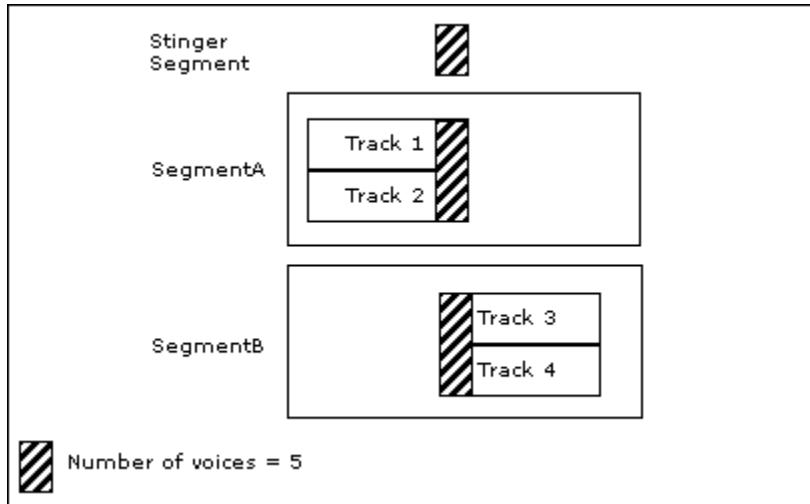


播放数限制和优先级 —— 示例

假设您正在为一个带宽限制非常严格的游戏作曲。同时只能播放不超过 4 个声部。此时您将顶层父对象的播放数限制设置为 4。每个音乐段落最多包含两条轨道，因此在过渡期间播放各条轨道的前导段和后尾段时，仅会占用两个声部。通过这种方式，如果同时播放两个段落，则仍不会超过仅四个声部的播放数限制。

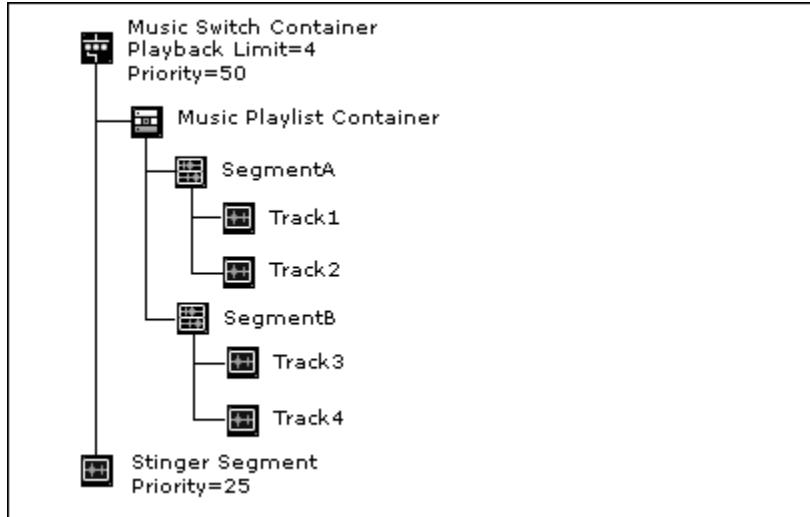


但如果在过渡期间游戏触发了插播乐句，那么将超出四个播放流的上限。



当声部的数量超出播放数限制时，Wwise 会查看各个音乐对象的播放优先级，来决定停止播放哪个音乐对象。此时，您可以将插播乐句段落设置为较低的优先级，以确保“常规”音乐继续播放。但如果该示例中的所有五个对象都具有相同的优先级，那么可以选择终止最后或最先播放的实例。

下图显示了如何设置音乐层级结构，以及在这种情况下如何设置播放数限制和优先级。



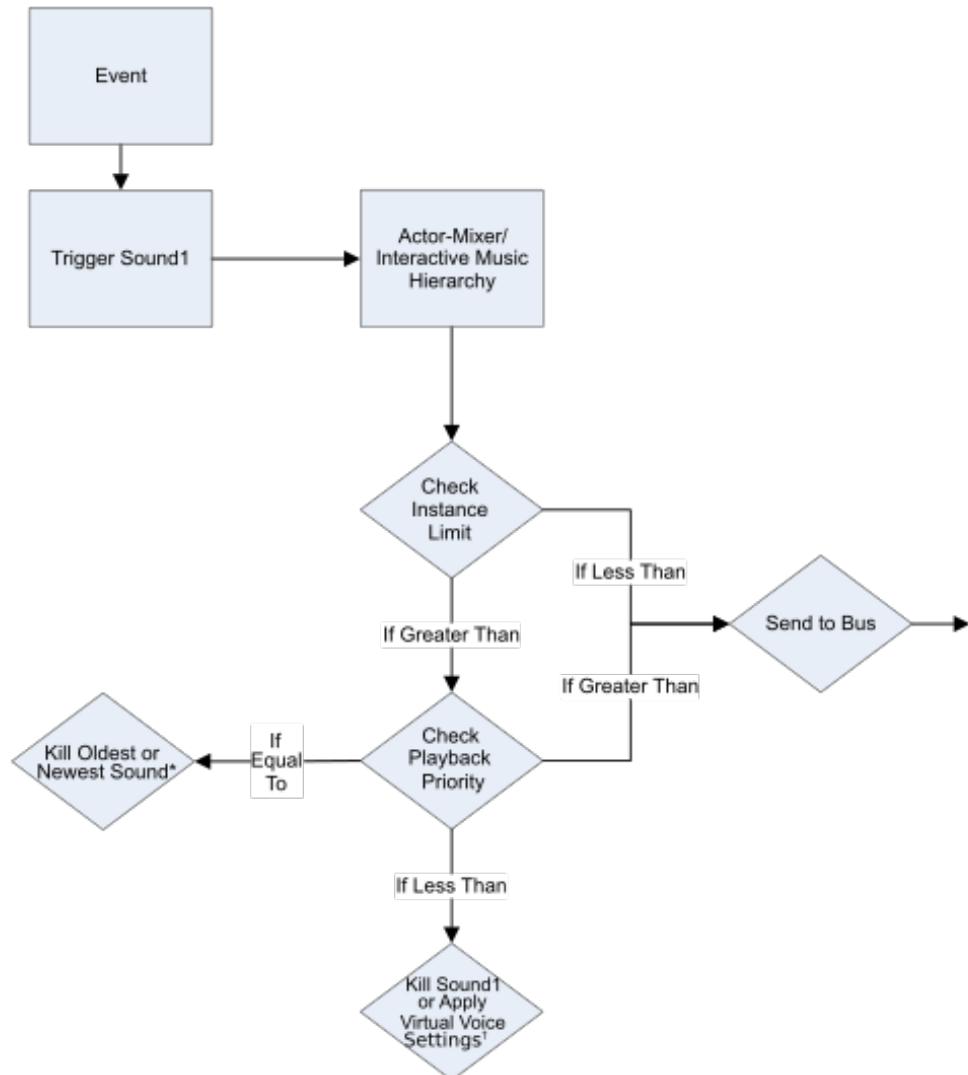
有关音乐对象和构建 Interactive Music 层级结构的详细信息，请参阅 [第 23 章 构建 Interactive Music Hierarchy](#)。

了解 Wwise 如何为对象排列优先级

Advanced Settings（适用于播放）决定游戏中将可以同时播放哪些对象。这些播放设置在 Wwise 中可以在两种不同的层级上进行定义：在 Actor-Mixer Hierarchy 和 Interactive Music Hierarchy 中的对象层级定义，或在 Master-Mixer Hierarchy 中的总线层级定义。因为这些设置在 Wwise 内的不同层级定义，所以对象必须通过两层独立的处理过程才能播放。

第一层处理发生在角色混音器或 Interactive Music 层级。当您在这个层级定义对象的 Advanced Settings 时，所设置的是全局或各游戏对象的限制。如果达到限制，则优先级会决定将哪些对象输出至 Master-Mixer Hierarchy 中的总线层级。

下面的流程图显示了 Wwise 如何决定 Actor-Mixer Hierarchy 内的哪些声音和振动对象，或 Interactive Music Hierarchy 内的哪些音乐对象将被游戏对象播放。事实上，最终决定将哪些对象输出到总线之前，可能需要经过不同父级对象的多次迭代（它们都有各自的限制）。

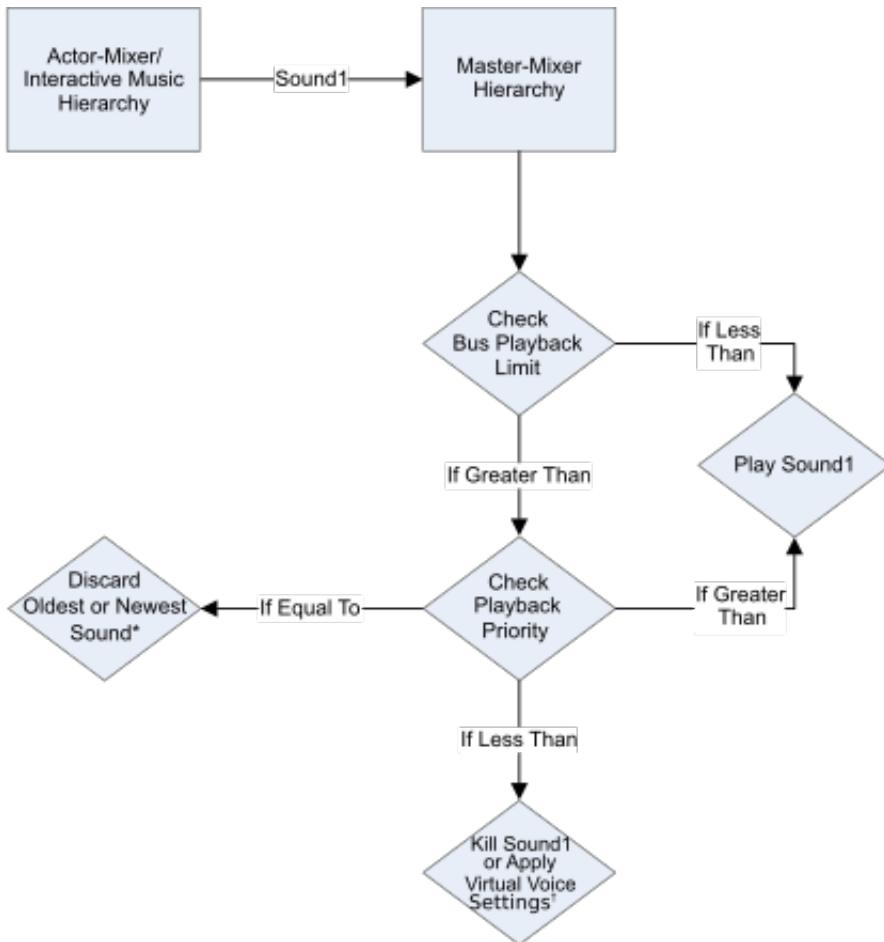


* You decide whether oldest or newest sound is killed.

† You decide whether to kill the sound or to apply the applicable Virtual Voice Settings.

如果新对象没有在角色混音器或 Interactive Music 层级中被被终止或被送至虚声部，那么它就会通过 Master-Mixer 层级进行第二层处理。在这个层级，根据总线结构内的声部限制，对象将经历类似的流程。

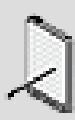
下面的流程图显示了 Wwise 如何决定哪些对象将通过总线输出。



* You decide whether oldest or newest sound is killed.

[†] You decide whether to kill the sound or to apply the applicable Virtual Voice Settings.

此时，Project Settings General 选项卡中的工程全局限制 Max Voice Instances 将决定声音是被终止，还是根据其各自的 Virtual Voice Settings 将其归入虚声部。



备注

Property Editor 的 Advanced Settings 选项卡中的所有属性均为绝对属性。有关绝对属性和不沿用父属性的详细信息，请参阅 [设置绝对属性](#)。

(音量阈值) 和 Virtual Voices (虚声部)

除了 Playback Limit 和 Playback Priority，Wwise 还可以根据音量阈值来决定播放哪些声音、音乐和振动对象。当音量降低至音量阈值，或当声音的数量超出 Playback Limit 上限，会为对象执行以下操作之一：

- 继续播放。
- 终止。
- 移至虚声部列表。

虚声部列表是一种虚拟环境，在这个环境中，声音引擎会监视列表里的声音的特定参数，但不会执行声音处理。如果您选择了该选项，对象则会根据其音量从实声部移至

虚声部（反之亦然）。如果音量恢复至音量阈值以上，或声音的数量低于同时播放声音的限制，那么这些对象会自动恢复为实声部。有关音量阈值设置的详细信息，请参阅 [指定工程的 Volume Threshold](#)。有关定义较音量较低的对象的行为的详细信息，请参阅 [管理音量较低的对象](#)。

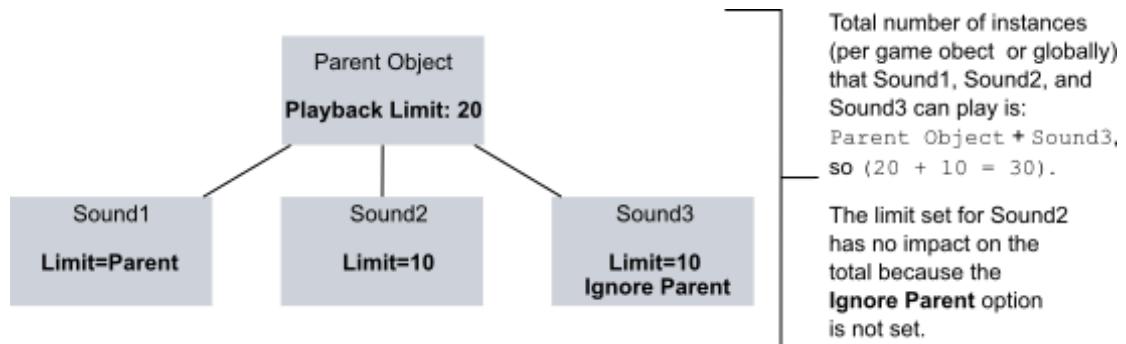
限制对象的播放实例

在游戏资源紧张或受游戏设计所限的情况下，您必须优化游戏内同时播放的对象。您可以通过以下方法，管理工程内同时播放的各种对象：

- 限制每个游戏对象可以播放的实例数量，或者对于若干 Actor-Mixer Hierarchy \Interactive Music Hierarchy 对象及子对象，在全局范围内限制其可以播放的实例数量；
- 限制可通过特定总线的对象总数；或
- 限制整个游戏中的对象总数。

达到限制后，Wwise 将使用对象的优先级设置，根据达到限制时所设置的行为，决定忽略哪个对象。如果对象具有相同的优先级，那么可以选择忽略最新的实例，还是忽略最早实例。

当您在 Actor-Mixer 层级或 Interactive Music 层级设置播放数限制时，您可以针对全局或各个游戏对象来控制同一结构能够被播放的实例数量。如果子对象忽略了父级对象的 Playback Limit，则子对象将不再计入父层级的播放数量限制，可播放的实例总数将是它们的限制数量总和。这意味着，如果父对象的限值为 20，子对象启用 **Ignore parent** 设置且限值为 10，则可播放的实例总数为 30。但是，如果子对象未启用 **Ignore parent** 设置，则可播放的实例总数与父对象相同，均为 20。如下图所示，无论是否忽略父对象都可以为子对象设置限值。这样，就可以针对特定对象及其子对象的最大实例数，实现精准的回放控制。



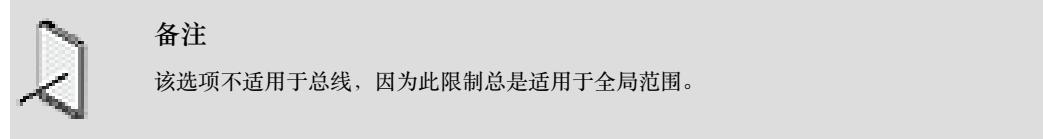
在 Master-Mixer 层级中设置播放数限制时，可以指定同时能够通过该总线的声音、音乐或振动的数量。由于各个对象的优先级已经在角色混音器或 Interactive Music 层级上进行了指定，因此总线上不存在播放优先级设置。

在工程/游戏层面上设置播放数限制时，可指定同时处于激活状态的对象数量。各个声音将采用在其 Advanced Settings 中设置的虚声部行为。

设置 Playback Limit 的方法如下：

1. 将对象加载到 Property Editor 中。

2. 切换至 Advanced Settings 选项卡。
3. 在 Play Limit 分组框中，选择 Limit sound instances to 选项。
各播放数限制参数变为可用。
4. 在对应的文本框中，输入每个游戏对象可同时播放的最大数量。
5. 请选择以下选项之一，来定义 Playback Limit 的范围：
 - Global，针对全局范围应用限制。
 - Per game object，针对每个游戏对象应用限制。



6. 请选择以下选项之一，来决定达到 Playback Limit 之后的操作：
 - Kill voice，停止播放具有最低优先级的实例。
 - Use virtual settings，对声音采用指定的虚声部行为。声音的虚声部行为包括 Continue to play、Kill voice、Send to virtual voice 和 Kill if finite else virtual。
7. 达到 Playback Limit 限值并且有多个对象被赋予最低优先级时，为了明确系统应采取的操作，请从 When priority is equal 列表中选择以下之一：
 - Discard oldest instance，停止播放具有最低优先级的最早实例。
 - Discard newest instance，停止播放具有最低优先级的最新实例。
8. 工程/游戏限制可以在 Wwise Project Settings 中针对各个平台修改。也可以在游戏正确初始化引擎后，使用 Wwise SDK API 进行修改。

使用声部限制系统时，最好了解以下几点：

- 虚声部不会被视为有效声部。
- 如果声音的虚声部设置为 Continue to play，则可能导致超出限值。
- 发送到虚声部的声音在返回实声部时（不再是虚声部），一般可以设置为：Play from beginning、Play from elapsed time 或 Resume。但如果选择了 Kill if finite else virtual 选项，则发送到虚声部的声音在返回实声部时将始终采用 Play from elapsed time 设置。

例如，我们假设：

- 在一条总线上将同时发声上限设置为 4 个，
- 其 When limit is reached 行为设置为 Use virtual voice settings。
- 音量阈值设置为 -60 dB。
- 该总线在一帧中播放了 8 个声部。

声音名称	Priority	Volume	虚声部行为
Sound_1	100	0 dB	Go Virtual
Sound_2	90	-90 dB	Go Virtual
Sound_3	80	-90 dB	Continue to play (继续播放)
Sound_4	70	0 dB	Go Virtual
Sound_5	60	0 dB	Kill voice
Sound_6	50	0 dB	Kill voice

声音名称	Priority	Volume	虚声部行为
Sound_7	40	0 dB	Go Virtual
Sound_8	30	0 dB	Continue to play (继续播放)

结果是：

声音名称	结果	原因
Sound_1	播放 (1/4)	最高优先级且高于音量阈值。
Sound_2	将变为虚拟	音量低于阈值。
Sound_3	播放 (2/4)	即使低于音量阈值也继续播放。
Sound_4	播放 (3/4)	-
Sound_5	播放 (4/4)	-
Sound_6	将被终止	超出限制，已有 4 个优先级较高的声音正在播放
Sound_7	将变为虚拟	超出限制，已有 4 个优先级较高的声音正在播放
Sound_8	播放 (5/4)*	*即使超出限制但仍播放，这是一种特殊的情形，不允许声音变为虚拟或被终止，且不允许替代 4 个正在播放并且优先级较高的声音。

常见问题

问：限制好像不起作用？这是为什么？

答：检查是否未将声音的虚声部行为设置为 **Continue to play**，处于兼容性原因，这是该行为的默认值。超出限制时，使用该设置的任何声音将不会被忽略，也不会被移至虚声部（除非超出限制行为被设置为终止声部）。开始使用 **Use virtual voice settings** 作为超出限制选项时，必须将所有“可去掉”或“可虚拟化”的声音设置为此选项，以开始节省 CPU。

还请牢记，该限值仅适用于选定对象层级内的声音对象。如果子对象启用了 **Ignore Parent** 选项，则将不再计入父层级的播放数量限制且不再受父级限制的约束。

问：该限制在大部分时间都起作用，但有时即使没有任何声音被设置为 **Continue to play**，性能分析显示的值仍超出限制。

答：当声音变为虚声部后，这些声音会在几毫秒内（一个音频帧）保持激活状态，才能淡出并避免出现 clicking。例如，如果限制为 1 且已经播放了一个声音，则另一个声音以较高优先级插入时，有效声音数量暂时变为 2，与此同时第一个声音将淡出，成为虚声部。

问：为什么当超出限制时发送至虚声部不是默认设置？为什么默认情况下所有声音别设置为继续播放？

答：对于虚声部，部分编解码格式可能不精准，可能需要在音频源中添加 **Seek Table**，才能支持虚声部中 **return to physical voice** 时的 **Play from elapsed time** 选项。出于这个原因，也为了确保默认情况下正常运行，该设置不作为默认设置。例如，Vorbis 格式需要 **Seek Table** 才能在 **Play from elapsed time** 设置下支持虚声部行为。

相关主题

- [定义播放优先级](#)

- 管理音量较低的对象

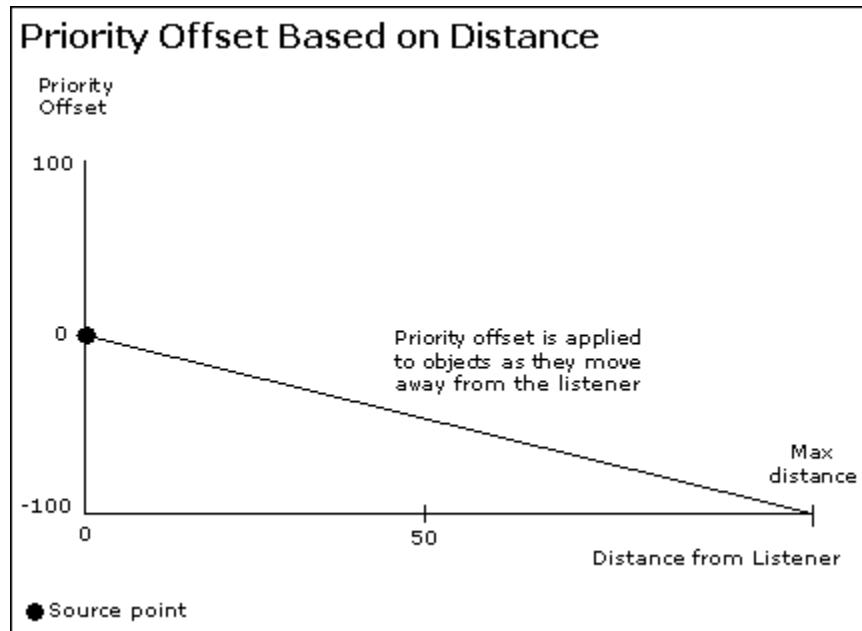
定义播放优先级

达到同时可播放对象数量上限时，无论是在游戏对象层级还是总线层级，Wwise 都将使用各个对象的优先级或相对重要性来确定将播放哪些对象。

您可以使用范围为 1-100 的标准数字分级定义各个对象的优先级，其中 1 为最低优先级，100 为最高优先级。如果对象具有相同的优先级，那么可以选择停止播放最新的实例，还是停止播放最老的实例。

根据距离进行优先级偏置

您也可以根据对象到听者的距离来更改播放优先级。Wwise 使用在 Attenuation Editor 中定义的最大距离来进行优先级偏置。应用的偏置量将取决于对象与听者之间的相对位置。Wwise 在原点不做任何偏置，在衰减最大距离处偏置到最大，而在两点之间偏置值将线性变化。

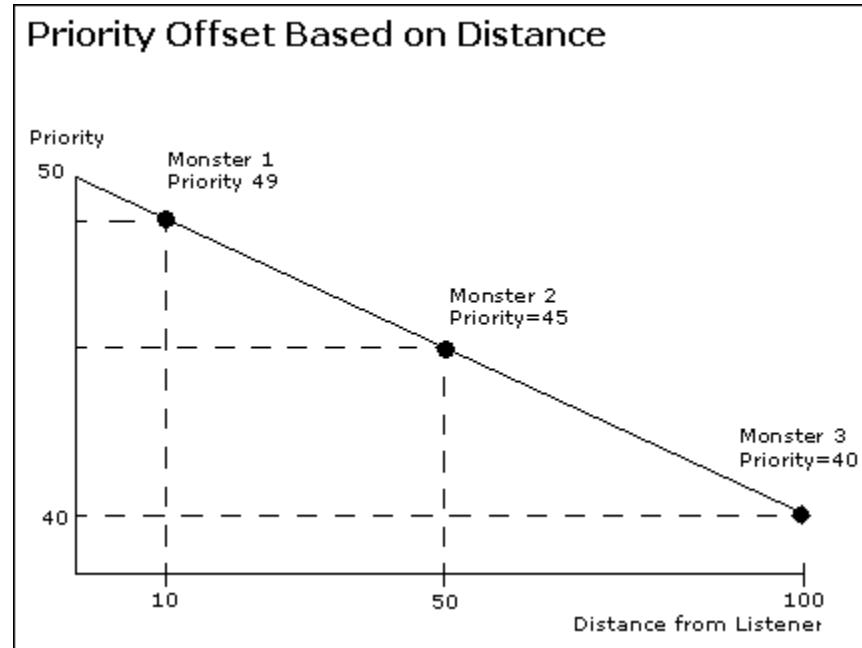


根据距离进行优先级偏置——示例

假设您有三个不同的对象：Monster1、Monster2 和 Monster3。这三种声音对象的优先级均为 50，优先级偏移量为 -10。Attenuation Editor 中的最大距离设置为 100 米。在游戏中，这些怪兽与听者的距离各不相同。但在本例中假设这些怪兽与听者的距离分别为：

- 怪兽 1: 10 米
- 怪兽 2: 50 米
- 怪兽 3: 100 米

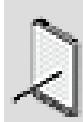
因为各个怪兽与听者的距离不同，并根据距离进行了优先级偏置，因此各个声音的优先级各不相同。下图显示了与听者的距离将如何影响各个声音的优先级。



由于播放优先级与播放数限制配合使用，因此如果发声数超出播放数限制，Wwise 则将使用这些声音的实时优先级决定播放哪些声音。

定义对象的播放优先级的方法是：

1. 将顶层父对象加载至属性编辑器。



备注

如果该对象不是顶层父对象，则必须选择 **Override parent** 选项后才能设置 **Playback Priority** 选项。

2. 切换至 **Advanced Settings** 选项卡。
3. 在 **Priority** 文本框中，输入数字 1 至 100，代表对象的优先级或相对重要性，其中 100 为最高优先级。
4. 要想根据对象与听者之间的距离来进行播放优先级偏置，请选择 **Offset priority by x at max distance** 选项。
5. 在文本框中，指定位于最大距离时优先级的偏移量。

相关主题

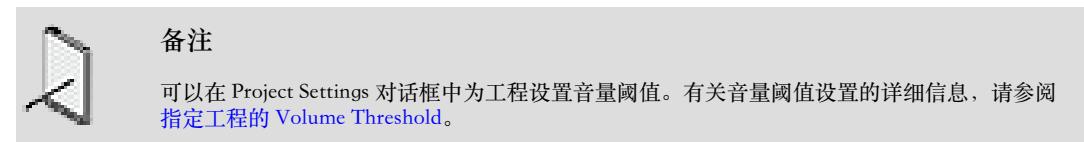
- [限制对象的播放实例](#)
- [管理音量较低的对象](#)

管理音量较低的对象

同时播放多个对象时，您需要维持最佳性能表现。要想实现这点，将低于一定音量的对象标记出来是一种好方法，这样它们就不会再占用宝贵的运算资源和内存。虽然您也可以选择继续播放这些听不到的对象，但可能更希望立刻停止它们，或将其放入虚声部列表中排队。

虚声部列表是一种虚拟环境，在这个环境中，声音引擎会监视列表里的声音的特定参数，但不会执行声音处理。将它们添加至虚声部列表的好处是，可以根据对象的音量在实声部和虚声部之间来回移动它们。当音量低于音量阈值时，这些对象将被添加至虚声部列表并停止处理。当音量增加时，比如对象向最大衰减距离内移动时，将会从虚声部移至实声部，这时声音引擎将再次开始处理音频或振动。尽管对音量较低的对象使用虚声部有诸多好处，但由于某些限制，这并不是一个普适的解决方案，包括对象返回至实声部时会出现较小的延迟，采样准确度也会降低。

为了强化功能和控制，当对象从虚声部返回至实声部时，您也可以定义其播放行为。



管理音量低的声音——示例

假设您正在创建一个第一人称射击游戏，主要角色要在各种走廊中穿梭。角色捡起敌人的对讲机时，能够听到敌人的通信。此时您可以同时播放多个声音，包括角色的脚步声、幽暗的走廊中燃烧的火把声，以及从对讲机中传出的声音。每种声音都有不同的特质和要求，因此，它们在音量改变时需要不同的处理方法。

声音	处理	Wwise 选项
脚步声	脚步声通常很短，是一次性声音，因此当其音量低于音量阈值时，您可能会需要终止这些声音，以节省宝贵的音频运算资源。	Kill voice
对讲机	由于游戏内的角色可调大或调小无线电声音，因此对讲机声音需要持续性、采样精确度和即时反馈。当其音量低于音量阈值时，您可以继续播放它们。这样一来，当玩家调高音量以继续收听敌人的通信时，就能得到即时且精确到采样点的声音反馈。	Continue to play
火把	火把声相对较短，并连续循环。虽然这些声音也需要一定的持续性，但它们不要求与对讲机声音同样的精确度。当这些声音的音量低于音量阈值时，您可以将这些声音发送至虚声部列表。声音引擎可以监测音量电平，但在返回音量阈值以上之前，不会进行任何音频处理。	Send to virtual voice (发送至虚声部)

管理音量较低对象的方法是：

1. 将对象加载到 Property Editor 中。



2. 切换至 Advanced Settings 选项卡。
3. 请从 Virtual Voice 组的 Virtual voice behavior 列表中选择以下选项之一，指定电平低于音量阈值时游戏对象的行为。
 - Continue to play，即使无法听到或感受到对象，但仍继续播放对象。这是能够确保采样点级精确度的唯一选项。
 - Kill voice，停止播放对象。

- Send to virtual voice, 将对象发送至虚声部列表，声音引擎将监视其中的某些参数，但不会进行处理。
4. 如果选择了 Send to virtual voice 选项，必须指定对象从虚声部返回实声部时的行为。因此，请从 On return to physical voice 列表中选择以下之一：
- Play from beginning —— 从头开始播放对象。该选项将重设对象的循环次数。
 - Play from elapsed time —— 继续播放对象，如同从未停止播放一样。
 - Resume —— 对象从实声部移至虚声部列表时，暂停该对象，当它复位至实声部时再恢复播放。



相关主题

- [限制对象的播放实例](#)
- [定义播放优先级](#)

优先级相关技巧和经验总结

在定义对象的高级设置前，您可能会需要查看以下章节，其中提供了一系列技巧和经验总结，可以帮助您获得所需结果。

Playback Priority (播放优先级)

将音乐对象设置为较高优先级——大多数情况下，游戏中音乐对象的优先级应比其它对象要高。触发的对象数量超过播放数限制时，这将确保音乐会连续播放。

Volume Threshold

对于较短的 Sound SFX 对象，其音量低于音量阈值时请将其终止，多数情况下，您应该对低于音量阈值的短小 Sound SFX 对象使用 Kill voice 选项，因为将其发送到虚声部后将导致以下结果：

- 如果选择了 Play from beginning 或 Resume，则返回实声部时，播放的声音很可能将不合时宜。
- 如果选择了 Play from elapsed time，则由于该声音很有可能作为虚声部被终止，所以很可能只会产生无谓的 CPU 消耗。

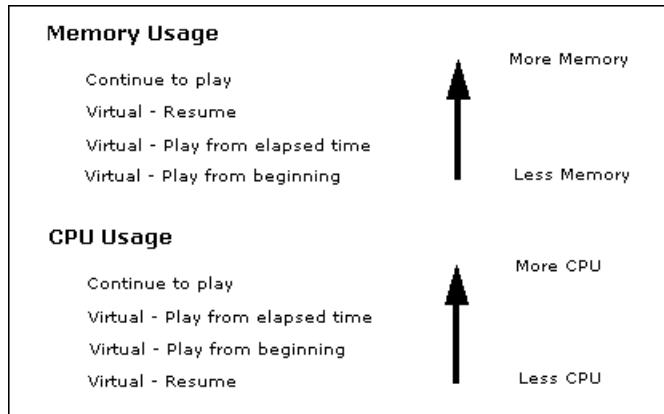
虚声部

决定使用哪个虚声部设置之前，应了解各虚声部设置的内存和 CPU 占用。

- Play from beginning - 该选项仅使用少量内存和 CPU，但如果对声音进行了流播放，则从虚声部返回时可能存在延迟。
- Play from elapsed time —— 该选项可以节省部分 CPU 和内存，但如果流播放声音，则声音从虚声部返回时可能会出现延迟。

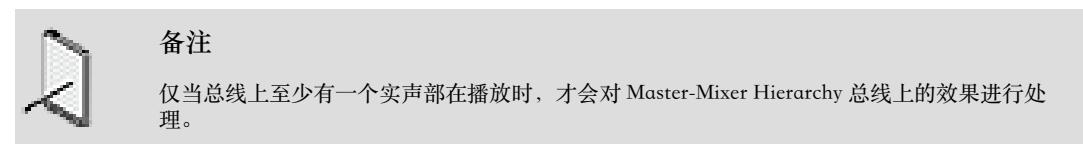
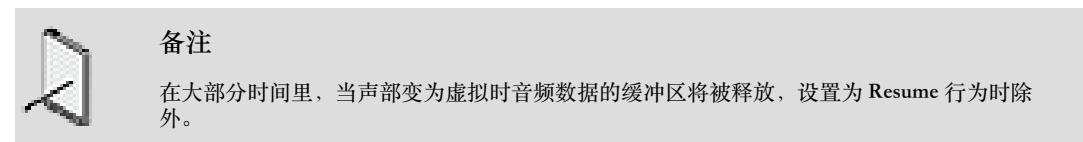
- **Resume** —— 该选项使用较少的 CPU，但会占用较大的内存，因为当声音从虚声部返回时将会保留内存缓冲区。请注意，如果声音永远不会重新进入最大衰减距离，而导致声音永远不会从虚声部返回，则在游戏运行过程中该缓冲区将一直保留在内存中。如果很多声音发生这种现象，则内存缓冲区将随时间累积，最终会在游戏中占用大量内存。

下图显示了各个虚声部选项彼此之间以及相对于 Continue to Play 选项的内存和 CPU 占用：



当然，当您考虑声音的音频格式、采样率、流播放设置等因素时，这些选项之间的差异有些可以忽略不计，但也有些差异巨大。

总是计算	作为虚声部时不计算
<ul style="list-style-type: none"> • 此时仍会进行定位、声部优先级和音量计算，因为需要依据这些计算结果判断声音是否应返回实声部。 • 在 Play from elapsed time 模式下，虚声部播放时仍会进行源插件计算（因为声音引擎无法在源插件中检索播放位置）。 	<ul style="list-style-type: none"> • 编解码解压缩。 • 音频数据移动、复制或混音。 • LPF、采样率转换、音高平移和 Actor-Mixer 层级结构中插入的效果器。 • I/O 读数。



对音乐对象进行数量限制

对音乐对象进行数量限制时，请注意不要将限制设置得过低，因为某些片段可能无法播放，从而导致音乐出现中断。请牢记，在 Music Segment 中淡变或切换时，很容易就能使用很多实例，因为每个段落中都可能有几个片段带有 Pre-entry 和 Post-exit 部分，也可能播放了 Stinger 段落。

第 13 章 管理效果器

概述	298
使用效果器	298
使用效果器实现环境音效	306
效果器技巧和经验总结	308

概述

提升游戏中声音、音乐和振动的一个方式是为工程层级结构中的对象应用若干个效果器。使用 Wwise 的开放式体系结构，您可以轻松创建和定义自己的效果器插件。这些不同的效果器可令游戏声音、音乐和振动更生动和真实。此外，为了帮助您更好地完成此过程，Wwise 自带一些效果器插件和出厂设置共享集。

有关渲染效果器的详细信息，请参阅[渲染效果器](#)。有关用于重建环境音效的效果器的详细信息，请参阅[使用效果器实现环境音效](#)。

使用效果器

效果器的组成是一些属性的集合，这个集合被称为一个实例（instance）。在定义实例时，您可以将其用作共享集，这用来将效果器实例作用于工程内的许多不同对象。使用共享集的优势是您无需为各个对象单独修改效果器属性。但是，如果您愿意，则还可以为一个特定对象定制自定义实例。

在对共享集的效果器属性做出更改后，所有指定了该共享集的对象都会受到影响。

您可以执行以下任务以帮助管理工程中的效果器：

- [创建效果器共享集](#)
- [删除效果器共享集](#)
- [将效果器作用于对象](#)
- [将效果器作用于总线](#)

创建效果器共享集

在您向层级结构中的对象或总线应用效果器时，最常见的做法是将此效果器用作共享集。Wwise 自带多个默认和预定义的效果器共享集，但您可能会需要创建自己的共享集来扩展此列表。各个效果器共享集为您提供效果器的一个不同版本，您可以将其作用于工程中的各种对象。

您可以在 Wwise 中的两个位置创建效果器共享集。

- 在 Project Explorer 中
- 在 Property Editor 中

在 Project Explorer 中创建效果器共享集的方法如下：

1. 在 Project Explorer 中，切换到 ShareSets 选项卡。
2. 在 Effects（效果器）部分中，右键点击工作单元以创建共享集。

此时将会显示快捷菜单。

3. 点击 New Child（新建子项）。

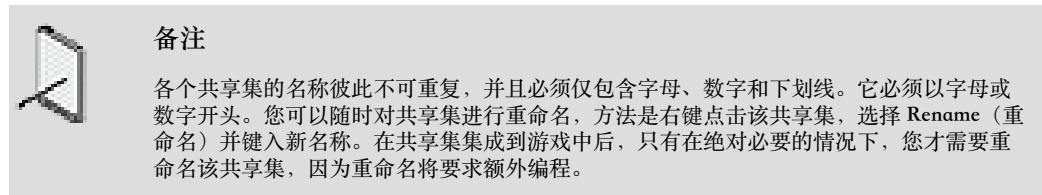
此时将会显示可用效果器插件的列表。

4. 选择您要创建的效果器类型。

新实例会显示为层级结构中工作单元的子项。

- 将默认名称替换为最适合该新共享集的名称，然后按 Enter。

共享集的新名称以及括号中的效果器类型会显示在效果层级结构中。



在 Property Editor 内创建效果器共享集的方法如下：

- 将对象加载到 Property Editor 中。
- 切换至 Effect 选项卡。



- 在 Effects 表中，点击 Selector 按钮 (>>)。

此时将会显示可用效果器列表。

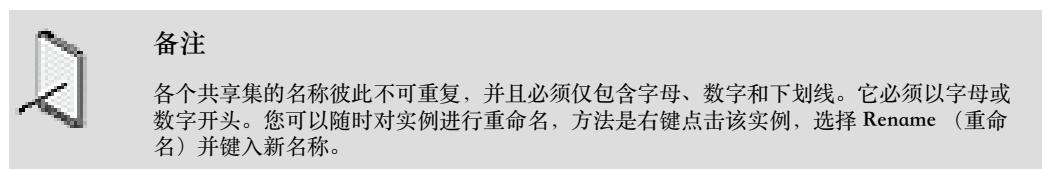
- 将鼠标置于要应用的该类型效果器上。

此时将会显示相应效果器共享集的列表。

- 点击 New（新建）。

此时将会打开 New Effect（新建效果器）对话框。

- 选择要在其中创建效果器共享集的工作单元。
- 为共享集键入名称。



- 点击 OK。

此时会创建新共享集并将其作用于所需对象。

相关主题

- [将设置从一个平台复制到另一个平台](#)
- [删除效果器共享集](#)
- [将效果器作用于对象](#)
- [将效果器共享集转换为多份自定义效果器](#)

删除效果器共享集

如果您不再需要某个共享集，则可以删除该共享集。在删除某个共享集之前，您应确保没有对象还在用到它。如果您删除某个共享集，则采用了该共享集的对象会自动删除该共享集。

删除共享集的方法如下：

1. 在 Project Explorer 中，切换到 ShareSets 选项卡。
2. 在 Effects（效果器）部分中，点击要删除的共享集。
3. 按 Delete 键。

该共享集被删除掉了，并且也从采用它的所有对象中移除了。

相关主题

- [创建效果器共享集](#)
- [将效果器作用于对象](#)

将效果器作用于对象

效果器可为工程中的对象增加真实感和多样性。在您已经创建效果器共享集之后，您可以轻松编辑它们并将其作用于您工程中的对象。您还可以将效果器共享集转换为一份自定义效果器实例，具体取决于您希望其属性是影响一个对象还是多个对象。

您可以将至多四个不同效果器作用于层级结构内的各个对象。如果应用了效果器链，则效果器会以列表中显示的顺序进行处理。应用效果器的顺序至关重要，因为顺序会改变对象的最终结果。某些效果器还会占用很大的 CPU 资源，因此制定高效的效果器使用策略至关重要。关于应用效果器的某些技巧，请参阅[效果器技巧和经验总结](#)。

在应用多份效果器时，您可能需要执行以下操作：

- [将效果器共享集作用于对象](#)
- [将效果器共享集转换为多份自定义效果器](#)
- [旁通效果器](#)
- [渲染效果器](#)
- [编辑效果器属性](#)
- [重新排序效果器](#)

将效果器共享集作用于对象

在已创建某个共享集之后，您可以将其作用于工程层级结构内的任何对象。



技巧

不建议将基于时间的效果器（例如 Wwise RoomVerb 混响器或延迟）作用于音乐对象，因为这会干扰已经指派给这些对象的基于时间的属性和行为。为了避免这种干扰，您可以另择在 master-mixer 层级上应用效果器。有关详细信息，请参阅[将效果器作用于总线](#)。

将效果器共享集作用于对象的方法如下：

1. 将对象加载到 Property Editor 中。
2. 切换至 Effect 选项卡。



3. 在 Effects 表中，点击 Selector 按钮 (>>)。

此时将会显示可作用于对象的效果器列表。

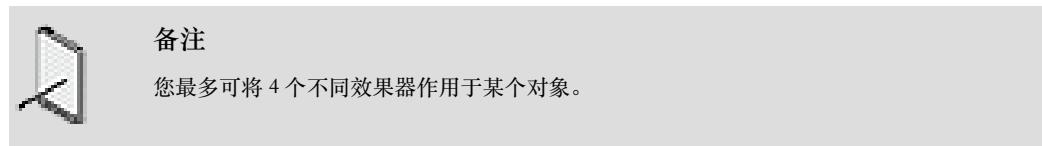
4. 将鼠标置于要应用的该类型效果器上。

此时将会显示相应效果器共享集的列表。

5. 点击要应用的共享集。

该共享集会作用于该对象。

6. 重复步骤 3-5 将其它效果器作用于该对象。



相关主题

- [编辑效果器属性](#)
- [将效果器共享集转换为多份自定义效果器](#)
- [旁通效果器](#)
- [渲染效果器](#)
- [编辑效果器属性](#)
- [重新排序效果器](#)

将效果器共享集转换为多份自定义效果器

Wwise 中的效果器有两个类别：

- 仅作用于单个对象的自定义效果器。在您更改自定义效果器的属性时，只有这个对象会受到影响。
- 可作用于许多对象的共享集。在更改共享集属性时，所有使用该共享集的对象都会受到影响。

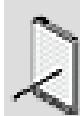
在默认情况下，您创建的所有效果器初始都为 ShareSets。但是，您可能会需要仅为单个对象来微调共享集。为此，您可以将效果器共享集转换为一份自定义效果器。例如，假定您已创建名为 Funky_Reverb 的共享集，但您希望仅针对名为 Speech_1 的声音微调该共享集。您可以将 Funky_Reverb 作用于 Speech_1，将其转换为自定义效果

器，然后进行更改。您所做的更改将仅作用于 Speech_1，而非采用了 Funky_Reverb 共享集的对象。

将共享集转换为自定义效果器的方法如下：

1. 将已应用该效果器的对象加载到 Property Editor。
2. 切换至 Effect 选项卡。
3. 在 Effect 表中，针对您希望转换为自定义效果器，从 Mode（模式）列表中选择 Define custom（定义自定义）。

这份效果器的名称后会跟有单词“Custom”（自定义）。从现在开始，您对实例所做的更改将仅影响使用它的一个对象。



备注

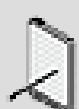
您还可以创建自定义效果器实例，方法是将共享集的默认（自定义）版本作用于对象。原始共享集仍存在，并且对于任何采用了此共享集的对象而言，该共享集保持未变。

相关主题

- [将效果器共享集作用于对象](#)
- [编辑效果器属性](#)

旁通效果器

在您已将某个效果器作用于对象后，您始终还是可以通过旁通效果器来试听原始未处理版本。由于您可以将几个效果器作用于同一个对象，您可以单独跳过各个效果器。您还可以在层级结构中的任何层级上旁通效果器。

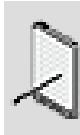


备注

您还可以通过在 Event Manager 中使用跳过动作或使用 RTPC（实时参数控制）跳过游戏中的效果器。

旁通效果器的方法如下：

1. 将顶层对象加载到 Property Editor 中。
2. 切换至 Effect 选项卡。



备注

如果对象不是顶层对象，则必须在选择 Override Parent 选项后才可选择效果器。

3. 在 Effects 表中，为您要跳过的各个效果器选择 Bypass（跳过）选项。

该效果器不再处理。



技巧

要跳过作用于对象的所有效果器，请选择 Bypass All（跳过所有）选项。

相关主题

- [编辑效果器属性](#)
- [将效果器共享集转换为多份自定义效果器](#)
- [渲染效果器](#)
- [编辑效果器属性](#)
- [重新排序效果器](#)

渲染效果器

将对象打包到 SoundBank 前，可以为应用到对象的各效果器选择离线渲染。这将节省游戏期间的处理器功耗，但您将无法使用 Bypass Effect 动作来旁通效果器，也无法在这些效果器中使用 RTPC 曲线。

在效果器链中渲染一个效果时，也将自动渲染它之前所有的效果器。For example, let's say you have a sound 对象 with the following three Effects applied to it:

- 矩阵混响器
- Parametric EQ（参数均衡器）
- 压缩器

如果为 Parametric EQ 选择渲染选项，在打包到 Soundbank 之前还将渲染 Parametric EQ 和 Matrix Reverb 效果器。

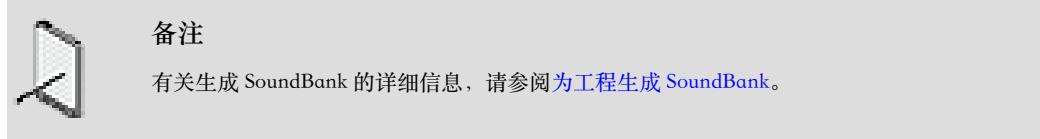
离线渲染效果器的方法如下：

1. 将对象加载到 Property Editor 中。
2. 切换至 Effect 选项卡。



3. 在 Effects 表中，为您要渲染的各个效果器选择 Render（渲染）选项。

在生成 SoundBank 时，所选效果器以及之前所有的效果器都将预渲染至对象中。



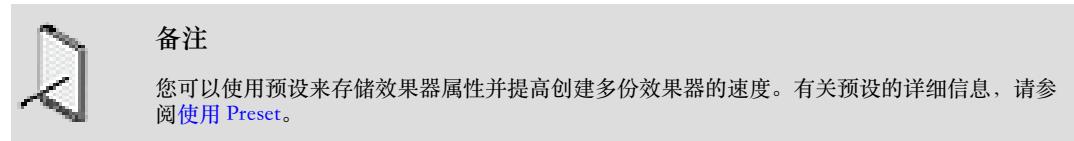
相关主题

- [编辑效果器属性](#)
- [将效果器共享集转换为多份自定义效果器](#)
- [旁通效果器](#)
- [编辑效果器属性](#)
- [重新排序效果器](#)

编辑效果器属性

您可以根据需要编辑任何效果器的属性。特别是，您有可能会需要先编辑新共享集或自定义效果器的属性，然后再应用它们。如果您不进行编辑，则效果器属性会保留其默认设置。通过将 RPTC 指派给某些属性，您可以进一步增强效果器。

在更改共享集的属性值时，所有使用该效果器的对象都将受到影响。如果编辑某自定义效果器实例，则只有应用了该效果器实例的对象会被更改。

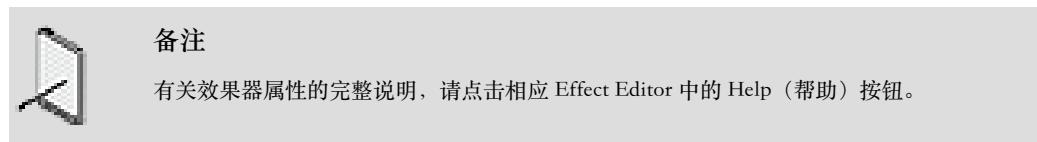


编辑共享集的方法如下：

1. 在 Project Explorer 的 ShareSet 选项卡中，双击要编辑的共享集。或者，点击 Property Editor 内 Effect 框中的 Edit...按钮。

此时将会为所选效果器打开 Effect Editor。

2. 您可以按照项目中实际需要达到的听觉效果来定义效果器属性。

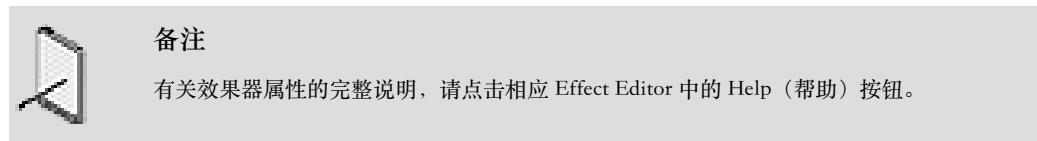


编辑自定义效果器的方法如下：

1. 将对象或总线加载到 Property Editor。
2. 切换至 Effect 选项卡。
3. 在 Effect 表中，点击要编辑的自定义效果器的 Edit 图标（...）。

此时将会为所选效果器打开 Effect Editor。

4. 您可以按照项目中实际需要达到的听觉效果来定义效果器属性。



5. 要将自定义效果器实例的属性值返回到该实例基于的共享集的效果器，请点击 **Reset Effect Settings**（重置效果器设置）。

相关主题

- [将效果器共享集转换为多份自定义效果器](#)
- [将效果器共享集作用于对象](#)
- [旁通效果器](#)
- [渲染效果器](#)

- **重新排序效果器**

重新排序效果器

效果器处理顺序可对对象的最终结果产生显著影响。因此，以正确顺序应用效果器至关重要。Wwise 以效果器在表中显示的顺序处理效果器。如果需要，您可以轻松改变效果器在列表中的顺序，方法是将某个效果器拖到列表中的其它位置。

重新排序效果器的方法如下：

1. 将对象或总线加载到 Property Editor。
2. 切换至 Effect 选项卡。
3. 在 Effect 表中，选择要在列表中上移或下移的效果器。

整个行现在处于突出显示状态了。

4. 将该效果器拖至列表中的新位置。

红线表明效果器将在列表中放置的位置。

相关主题

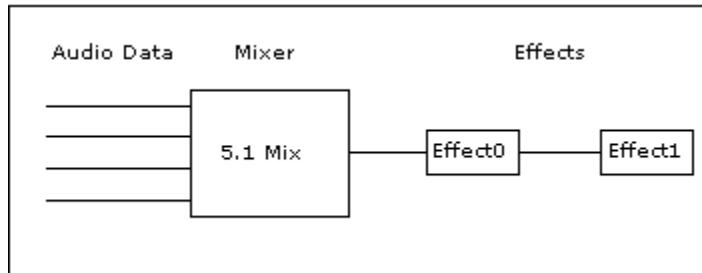
- [将效果器共享集转换为多份自定义效果器](#)
- [将效果器共享集作用于对象](#)
- [旁通效果器](#)
- [渲染效果器](#)
- [编辑效果器属性](#)

将效果器作用于总线

您可以像将音频效果器作用于对象一样轻松将音频效果器作用于总线。您只需将总线加载到 Property Editor，然后对声音对象应用所需效果器即可。有关此过程的详细信息，请参阅[将效果器作用于对象](#)。



在效果器作用于总线时，所有传入音频数据都会在应用效果器之前进行子混音（sub-mixing）。如果应用了效果器链，则效果器会按照列表中显示的顺序生效。



使用效果器实现环境音效

Wwise 中有两种不同类型的总线，分别是：音频总线和辅助总线。辅助总线与发送系统可以用于实现环境音效。在辅助总线上应用效果器可以重新构建游戏环境对播放声音的影响方式，例如添加混响。辅助总线上的效果器还可以根据游戏对象的位置数据动态发挥作用。

以下各节将向您介绍在工程中如何在辅助总线上使用效果器。

- [理解发送](#)
- [将游戏定义的辅助发送集成到工作流程中](#)

理解发送

游戏内的环境声学效果可以被模拟出来，方法是在辅助总线上添加一系列效果器并将信号发送给该总线。您可以根据需要创建任意数量的辅助总线；不同总线对应于游戏中的不同环境。您还可以在游戏中使用 SDK 函数 `SetBusEffect()` 动态地向辅助总线上添加效果器。在您为辅助总线定义了效果器后，您需要定义向这些总线发送信号的对象。

您可以选择使用用户定义的辅助发送来静态定义对象发送，也可以选择使用游戏定义的辅助发送，从而让游戏定义这些发送。

创建辅助总线的方法如下：

1. 在 Master-Mixer 层级结构中，创建辅助总线对象。
2. 察看辅助总线。
3. 在辅助总线的 Effect（效果器）选项卡中，添加实现相应环境音效所需的效果器。

使用用户定义的辅助发送将环境音效施加于对象的方法如下：

1. 察看 Actor-Mixer Hierarchy 对象。
2. 将辅助总线拖放至用户定义的辅助发送列表中。
3. 使用 Send volume（发送音量）滑杆定义要发送给辅助总线的信号量，即定义对象参与环境音效的程度。

使用游戏定义的辅助发送将环境音效施加于对象的方法如下：

1. 察看 Actor-Mixer Hierarchy 对象。
2. 启用选项 Use game-define auxiliary sends（使用游戏定义辅助发送）。
3. 在游戏中，使用函数 `AK::SoundEngine::SetGameObjectAuxSendValues()` 为各个游戏对象指派一个辅助发送。



备注

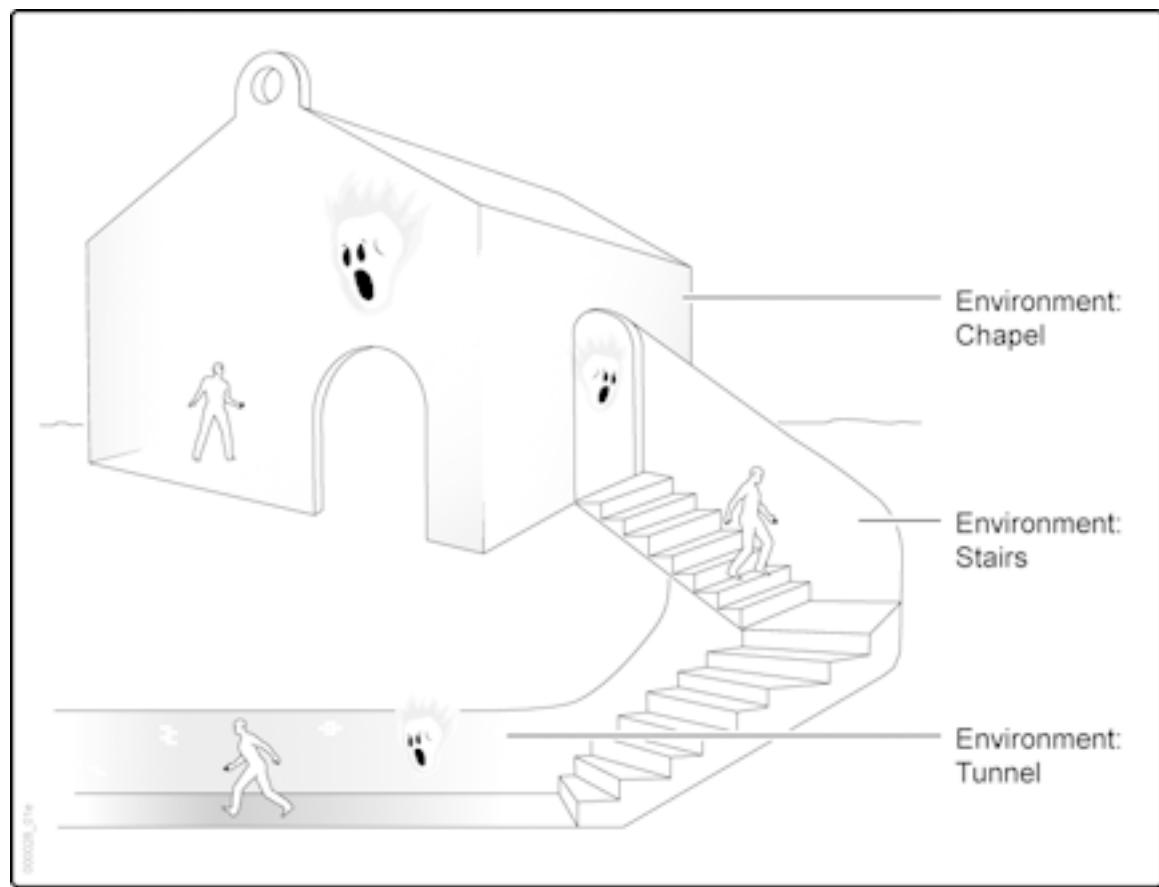
有关在游戏中映射环境的详细信息，请参阅 Wwise SDK - Windows > 声音引擎集成纵览 > 将 Wwise 元素集成到游戏中 > 集成环境和游戏定义的辅助发送，该章节位于 [Wwise SDK 文档](#) 中。您和音频程序员还可观看集成演示，该演示将向您展示如何设置辅助发送的示例。

使用辅助发送重建环境音效——示例

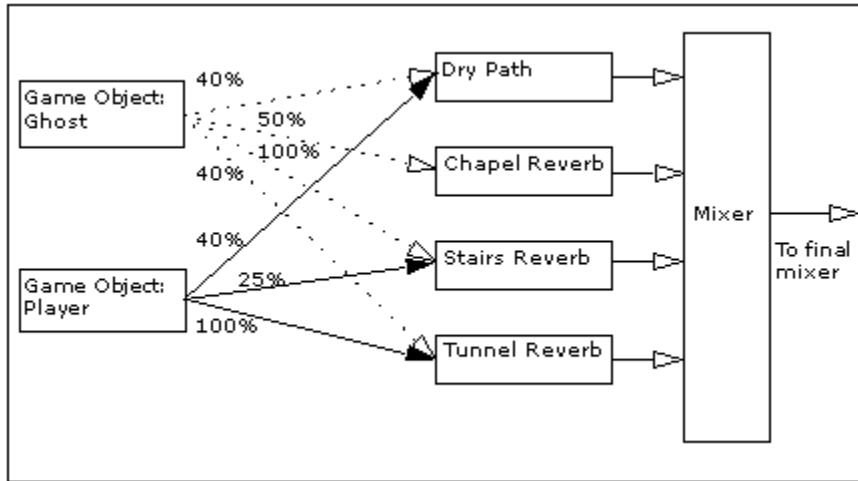
假定您正在制作一款游戏，故事发生在幽灵出没的墓地内和附近。游戏中魅影重重，您希望这些幽灵在不同的环境中听起来不一样。在游戏的一个环节，玩家可以探索教堂、隧道和连接两者的楼梯。对于游戏的这一环节，您定义了以下辅助总线：

- 教堂
- 楼梯
- 隧道

您决定为这三个环境中的各个环境赋予一个独特的混响效果。例如，隧道的空间比教堂小，周围是石墙，形似洞穴，因此它的混响比在教堂中明显。在 Wwise 中，您可以创建一个具有更高混响各个层级和更短衰减时间的混响实例来匹配隧道中的音效，并可创建其它混响来匹配教堂和楼梯中的音效。然后，程序员可以将您创建的辅助总线映射到游戏几何空间结构上。产生的效果是，当幽灵进入隧道中时，幽灵发出的声音所产生的回音远远大于在教堂中的回音。



Wwise 辅助发送还可以通过动态计算所有并行发送的发送音量来模拟在环境之间的移动。下图说明了此游戏示例的内置动态辅助总线通路和混音。



对特定声音使用的各个发送的比例取决于各个游戏对象在游戏几何空间中的位置。在本例中，勇敢的玩家正从教堂沿着楼梯下到隧道，幽灵在后面紧追不舍。在穿过隧道的途中，可以将玩家和幽灵定义为 100% 处于楼梯环境，但还要加上 50% 处于教堂环境，40% 处于隧道环境。然后使用各个混响实例以适当的比例处理幽灵的声音。

将游戏定义的辅助发送集成到工作流程中

在 Wwise 中，您可以使用效果器创建和编辑辅助总线，以及定义哪个对象发送信号给它们。根据游戏对象的位置以编程方式为声音定义辅助发送。下面的步骤描述开发团队如何能够实现游戏定义的辅助发送：

- 音效设计师根据游戏中的环境定义辅助总线对象，例如小屋、教堂和洞穴。
- 音效设计师需确保应受环境效果影响的所有声音都启用了 **Use game-defined auxiliary sends** 选项。
- 当环境出现在游戏几何空间中时，游戏程序员将辅助总线映射到环境中。
- 游戏程序员要创建一个机制来实时计算出环境比例值，并将比例值报告给声音引擎。
- 声音引擎会计算哪一条辅助总线将运用于各个游戏对象所触发的声音。



备注

要试听和微调游戏定义的辅助发送，必须连接到游戏中对效果器进行实时微调，或者暂时将该辅助发送当作用户定义的发送使用。推荐您试试 SoundFrame 测试工具 (SFTest.exe)。此工具是 Wwise SDK 安装内容的一部分，可用于远程控制 Wwise 和从工程中获取数据。您必须将事件附加到游戏对象上，播放事件，然后按 **Edit Sends** (编辑发送) 按钮来编辑游戏定义的辅助发送。通过这种办法，您可以控制各个层级的游戏定义的发送。

效果器技巧和经验总结

在 Wwise 中使用效果器时有许多选项可用。在实现效果器时使用某些策略可以为您提供更好的声音效果并节省资源。下面是在工程中使用效果器时您可能需要考虑的一些策略。

CPU 占用

效果器始终会消耗 CPU 功率，了解此功耗可以帮助您更高效地使用各种效果器。一般来说，在主混音器各个层级使用效果器时占用 CPU 较少。例如，如果在游戏的主

音频总线上使用 Wwise 峰值限幅效果器，则运行时将只动用一份效果器来处理。而如果您在对象级使用此效果器，一次则可能需要动用数百份相同的效果器来处理。

将效果器渲染出来也可以帮助您节省 CPU 资源，因为这样做可以避免实时处理这些效果。当然，您无法对已经渲染的效果器应用 RTPC，因为渲染后，这些效果器的属性是无法更改的。

就单个效果器而言，延迟和 Wwise 参数均衡器往往使用很少的 CPU 资源。Wwise Compressor（压缩器）、Peak Limiter（峰值限幅器）和Expander（扩展器）效果使用得多一些。对于混响，您可以选择 Wwise RoomVerb 混响器，它对资源的需求更大，但品质更高；也可以选择 Wwise 矩阵混响器，您可以通过调整它来满足您的品质和性能需求。

总之，最好的策略是使用游戏性能分析工具来测试您的工程。这样，您可以实时地观察效果器的 CPU 占用，然后决定应该如何使用效果器。有关性能分析的更多信息，请参阅[理解 Wwise 中不同类型的性能分析](#)。

效果器与音乐

建议不要在对象层级对音乐对象使用基于时间的效果器（例如 Wwise 矩阵混响器或延迟），这可能干扰基于时间的属性和已经指定给它们的行为。在主混音器各个层级（即对音频总线）上使用基于时间的效果器可避免发生此干扰。

第 14 章 管理 Motion

概述	311
为游戏创建振动	313
为振动来构建输出结构	321
振动相关技巧和经验总结	322

概述

使游戏更具代入感的另一种有效方法是添加振动（Motion）反馈。

Wwise 为在游戏中创建和集成振动提供了一套完整的管线。Wwise 为振动实现了全面的、与音频创作相似的管线解决方案，您可以实现以下功能：

- 只需稍加学习，就可以创建复杂、真实的振动效果。
- 将振动轻松集成至游戏内，不会明显影响游戏或声音引擎的表现。
- 使用与音频相同的功能来构建和集成振动。
- 无需额外工作，为各种平台上的同类设备创建振动效果。
- 根据游戏要求，轻松添加或移除振动元素。



了解 Wwise 中振动的工作方式

为了尽可能简单地从音频过渡至振动，Wwise 使用相同的工作流程，很多原理和功能都同时适用于两者。与音频类似，振动对象可以在层级结构中进行组织、可输出至总线，并可使用事件在游戏中触发。同时，还可在 3D 空间化环境中定位，由工程中的各种游戏同步器控制，并打包到 SoundBank（音频包）中。播放方面的主要区别在于，振动是通过游戏控制器而非扬声器播放的。

相关主题

- [振动设备的类型](#)
- [生成振动的方法](#)

振动设备的类型

Wwise 目前支持以下类型的振动设备：

- 游戏控制器（所有平台和通用类型）
- PS4 Move®

通过添加第三方插件，可以使用其他类型的设备。

生成振动的方法

振动数据在 Wwise 中基本上是由一个源产生的。这个源可以是已有的媒体文件，新的媒体文件，或是信号发生器插件。

在决定了游戏将支持哪些振动设备以后，就要决定使用哪种方法来生成振动。在 Wwise 中，有两种不同的方法可以用于创建一个振动源：

- 使用现有的音频信号。
- 使用振动专用源。

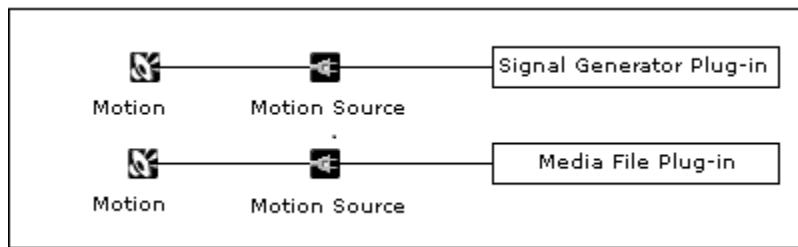
使用现有音频信号生成振动信号

生成振动的简单方法是使用现有声音并将其发送到 Motion bus（振动总线）下的 Auxiliary Bus（辅助总线）。Motion bus 是一种常规音频总线，只是其 Audio Device 被设置为振动设备。振动设备插件会将音频信号转换为适合硬件的低频信号。原始声音将不受影响，就像其他使用发送的情况一样。

由于振动源是由现有的音频源生成的，因此在游戏中，该振动也与该音频的播放关联在一起。这意味着在游戏中不需要再用单独的事件触发该振动源。这也意味着所有影响该音频对象的属性、行为、游戏同步器等，也会同样影响这个振动源。

使用振动专用源生成振动

另一种生成振动的方式是创建振动专用的 Wwise 对象。这些振动对象与其他声音对象类似，但使用专门生成振动的源插件，而不是依靠转换音频数据。

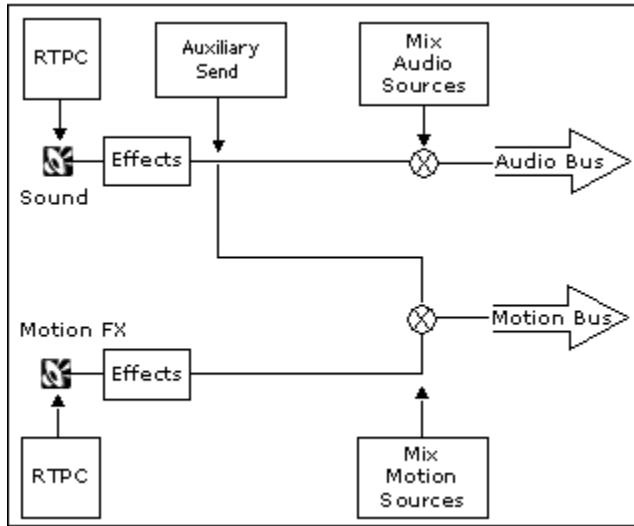


在其他方面，这些振动对象行为就和普通的声音一样。您可以使用 Container 和 Actor-Mixer 创建复杂的振动结构，并设置振动效果的属性和行为。由于这些振动效果并不需要与游戏音频关联在一起，因此可以用它们自己的事件，在游戏中的任何时刻触发它们。

当然，在有些情况下，某一种方法可能会比另一种更适用。关于上述方法的进一步讨论，请参阅 [为游戏创建振动](#)。

振动处理管线

振动处理管线会获取两种源所生成的振动信号，将它们混合后输出至特殊的振动总线。下图显示了 Wwise 如何处理两种类型的振动源，以及振动管线如何融入音频处理管线。



为游戏创建振动

在 Wwise 中，您可使用以下方法生成振动：

- **通过现有声音生成振动**
- **创建专用对象来生成振动**

具体选择使用哪种方法，要依据各个使用情形的要求和限制来决定。花些时间了解每种方法的优势和劣势，有助于正确选择方法。

代码设置：集成 Wwise Motion

在声音引擎端，音频程序员必须确保一切设置就绪，以便 Wwise 实现振动效果。请参阅 SDK 文档的[集成 Wwise Motion](#) 页面。

通过现有声音生成振动

由于很多振动效果将关联至音频事件，如枪声、路面振动、爆炸等，因此您可以直接通过工程中的现有声音和音乐对象生成振动数据。在本例中，Wwise 只需取得音频信号，提取出低频成分，然后对这部分信号进行重采样，就能创建令人信服振动效果。虽然该方法给音频设计师和程序员带来的工作量很少，并且完全适用于与音频同步的振动，但该方法有以下劣势：

- 对创建的振动类型不能进行更多控制。
- 更多的 CPU 资源消耗，因为音频信号必须在运行时进行滤波和重采样，以创建振动数据。
- 无法处理与音频没有关联的振动。

尽管有上述劣势，但在某些情形下该方法也十分实用。例如，该方法极适用于创建游戏中振动的原型。当原型完成且效果得到认可后，您可以决定是否继续创建专用振动对象来实现进一步控制。

通过现有声音生成振动涉及以下任务：

- **指定振动的输出连线**

- 改变由音频生成的振动信号

通过现有声音生成振动前，您应该留意以下几点：

- 从原始音频源中移除直流偏置将会改变振动输出。
- 由于电机功率、最高速度与重量的物理差异，每个平台控制器的振动输出都会有所不同。

指定振动的输出连线

如果您计划从现有音频源或振动源生成振动数据，则需要先决定输出至哪条振动总线。振动数据与音频数据不同，要求单独的振动总线层级结构，可通过其中的总线将数据进行输出。您可以在 Master-Mixer Hierarchy 中创建单独的层级结构，只需创建一个新的 Master Bus 并为其分配 Wwise 振动设备即可。有关创建振动总线层级结构的详细信息，请参阅 [为振动来构建输出结构](#)。

为振动专用源指定输出总线的方法如下：

1. 将顶层对象加载到 Property Editor 中。
2. 在 Output Bus 分组框中，点击 Browse 按钮 (...)。

此时将会打开 Project Explorer —— Browser。

3. 选择一个启用了振动的 Master Bus 的子总线。
4. 点击 OK。

当前对象及其子对象生成的振动数据即可通过所选的总线进行输出。

为音频源生成的振动指定输出总线的方法如下：

1. 将顶层对象加载到 Property Editor 中。
2. 在 User-defined Auxiliary Sends 分组框中，点击 Browse 按钮 (...)。

此时将会打开 Project Explorer —— Browser。

3. 选择一个启用了振动的 Master Bus 下的 Auxiliary Bus。
4. 点击 OK。

当前对象及其子对象生成的振动数据即可通过所选的总线进行输出。

相关主题

- 改变由音频生成的振动信号

改变由音频生成的振动信号

振动设备仅可生成低频信号，因此可能需要减少或过滤音频源中的高频信号。得到的信号也可能需要放大才能达到合适的强度。鉴于音频信号是经由辅助总线发送到振动总线，您可以使用发送功能提供的常规属性。



请勿将 HPF 应用于振动对象

因为振动设备只会生成低频信号，所以基本上所有高通滤波器都会抑制振动效果。

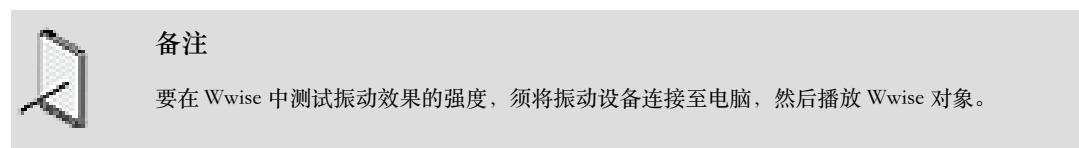
滤除高频的方法如下：

1. 将启用振动的 Auxiliary Bus 加载到 Property Editor 中。
2. 在 Output Bus 组框中，修改 Low Pass filter 值来设置将用于生成振动信号的音频截止频率。

增强或减弱生成振动信号的方法如下：

1. 将音频对象加载到 Property Editor 中。
2. 在 User-defined Auxiliary Sends 组框中，修改 Volume 值来放大或衰减振动信号。

此外，您也可以在 Auxiliary Bus 上使用效果器，例如 EQ，来提升或减弱某些频段。



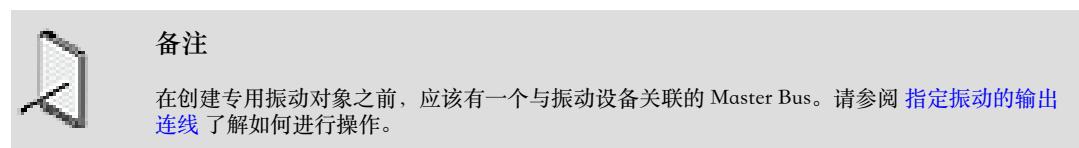
相关主题

- [指定振动的输出连线](#)

创建专用对象来生成振动

为了更好地控制振动强度，工程中，最好通过独立的对象来生成振动数据。可以用 Container 和 Actor-Mixer 来将它们分组。与 Sound SFX（音效）对象类似，它们也具有相应的源（通常为信号发生器插件）。与常规 Sound SFX 对象的唯一区别是，这些对象会直接输出至启用了振动的总线。另外，使用直接生成振动数据的专用插件源通常效果会更好。目前，Wwise 提供两种专门用于生成振动效果的源插件：已弃用的 Wwise Motion Generator 和全新的多用途 Wwise Motion Source。但是，使用其他插件也是可以的。

具体选用哪个源将取决于实现振动的设备和所需的振动类型。每个源均包含一组属性参数，您可以使用这些参数微调振动效果。

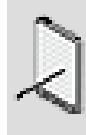


创建振动专用对象的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，右键单击要在其中创建声音对象的工作单元。
2. 从快捷菜单中选择 New Child > Sound。

新的声音对象将被添加到 Actor-Mixer Hierarchy 中。

3. 将默认名称替换为最适合该对象的名称。



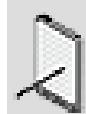
备注

在 Wwise 中命名对象时，不得使用以下字符：':<>%*?/\|.'

4. 在 Contents Editor 中，单击 Add Source > Wwise Motion Source。

Wwise Motion Source 将作为源添加。

5. 双击源插件，在属性编辑器中将显示完整属性列表。
6. 根据需要，编辑属性。



备注

要获得源插件属性的完整说明，请点击属性编辑器中的 Help 按钮。

相关主题

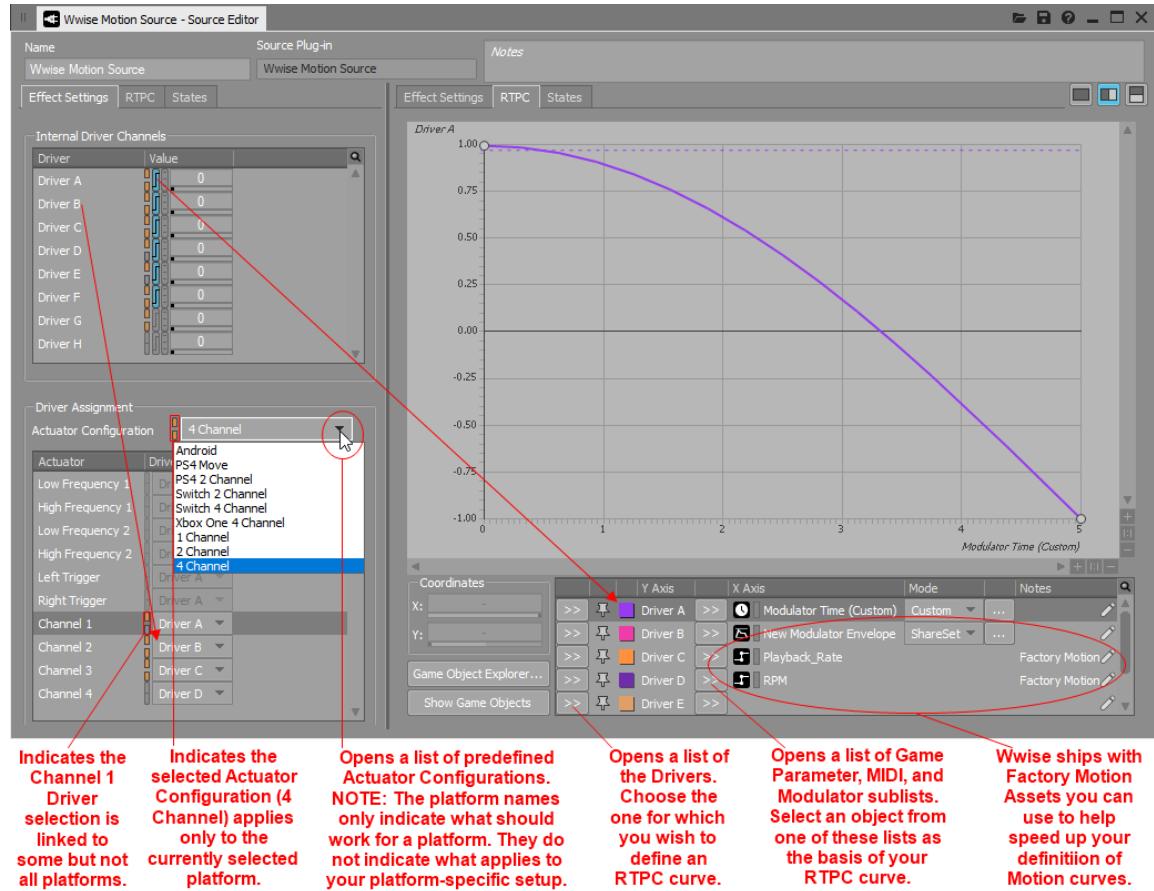
- [针对游戏控制器生成振动](#)
- [通过现有声音生成振动](#)

针对游戏控制器生成振动

由于受电机速度限制，游戏控制器只能重现有限的振动效果。为此，我们设计了专用源插件 [Wwise Motion Source \(Wwise 振动源\)](#)（新增）和（不建议使用）[Wwise Motion Generator](#)，以便为这些类型的设备生成振动效果。

Wwise Motion Source (Wwise 振动源)

在结合此插件使用 [RTPC](#) 时，最多可利用八个不同的 Driver（驱动器）来指定振动强度和时长。然后，可以将这些 Driver 应用于不同的促动器（电机），从而实现不同的配置。利用标准的 Wwise 平台链接功能，可根据平台分别设置 Driver、Actuator Configuration（促动器配置）和 Actuator（促动器）。



Wwise Motion Source 的基本设置

为了确保振动设备（游戏控制器）的促动器与 Wwise 配合良好，您可以快速选择 Actuator Configuration，并将偏置 Value（值）应用于适用的 Driver。虽然从技术上来说在游戏中直接使用偏置 Value 也未尝不可，但实际应用时最好还是使用 RTPC 及关联参数（如时间）来创建和控制振动曲线。

利用 Wwise Motion Source 插件的偏置 Value 来创建振动效果：

1. 将[振动专用对象](#)加载到 Property Editor（属性编辑器）中。
 2. 在 Contents Editor（内容编辑器）中，双击所列 Wwise Motion Source 将其加载到 Source Editor（源编辑器）中。
 3. 在 Internal Driver Channels（内部驱动器声道）分组框中，向右移动 Driver 对应的 Value 滑杆，使其值大于 0。

若使用此 Driver，则将在对象播放期间持续生成恒定强度的振动效果，直至 Event（事件）停止播放对象。

4. 在 Driver Assignment (驱动器指派) 分组框中, 从 Actuator Configuration 列表选择 1 Channel (1 声道)。

这时声道 1 将成为唯一激活的促动器。

5. 在 Channel 1 (声道 1) 列表中，选择刚刚调整的 Driver。

若使用[正确设置的振动输出](#)播放对象，则在对象播放期间将在单个控制器声道中生成恒定强度的振动效果。

这表示设备和 Wwise Motion 配合良好。

尝试不同的 Actuator Configuration

在确认设备配合良好的情况下，可按照类似方式尝试使用不同的 Actuator Configuration，并确定适合不同[选定平台](#)的配置。比如，可将某一 Driver 的偏置 Value 设为 0.1，并将另一 Driver 的偏置 Value 设为 1.0。这样方便测试并辨别哪个控制器电机对应于默认 Sink 的哪个 Actuator。

Actuator Configuration 的给定平台名称指示了已知与 Wwise 默认 Sink 配合良好的设备。不过，这并不意味着这些配置仅适用于所述平台。事实上，有时需要为特定平台选择不同的配置。比如，有些 Android 平板电脑配有两个促动器。这时 2 Channel (2 声道) 配置要比默认 Sink 的匿名单声道 Android 配置效果更好。在声道数不匹配时，并不会阻止设备产生振动效果。若声道数量太少，则多个促动器将使用相同的 Driver 声道；若声道数量太多，则将忽略部分声道。

结合使用 RTPC 和 Wwise Motion Source

大部分情况下，仅仅使用偏置 Value 是不够的。您需要根据游戏参数或预定义 MIDI 事件的时长来定义具体的曲线。[RTPC](#) 刚好可以帮您实现这一操作。

结合使用 RTPC 和 Wwise Motion Source 插件来调整振动效果：

1. 将[振动专用对象](#)加载到 Property Editor (属性编辑器) 中。
2. 在 Contents Editor (内容编辑器) 中，双击所列 Wwise Motion Source 将其加载到 Source Editor (源编辑器) 中。
3. 在 RTPC 选项卡中，单击选择器按钮。

这时将列出所有 Driver。

4. 选择 Driver。

这时将在 Y Axis (Y 轴) 列下显示该 Driver，并在其右侧列中显示新的选择器按钮。

5. 点击选择器按钮。

这时将显示 Game Parameter (游戏参数)、MIDI 和调制器列表。

6. 选择选项。为简单起见，假定选择 Time (时间)。

这时将显示子列表，以供选择现有 Time Modulator (时间调制器) 或创建新的 Time Modulator。

7. 选择 New...。

这时将显示 New Modulator Time (新建调制器时间) 对话框。

8. 输入新的 Time Modulator 的名称。然后，单击 OK (确定)。

这时将在 X Axis (X 轴) 下显示新创建的 Time Modulator，且坐标图中显示平直曲线。

9. 在 Time Modulator (X) 轴上约 0.2 s 处，双击坐标图曲线。

这时将新建一个控制点。

10. 将新控制点向上拖至 Driver (Y) 轴上 1.00 处。

若使用[正确设置的振动输出](#)播放对象，则对于要使用指定 Driver 的控制器，其所生成振动效果的强度将会跃升；然后，缓慢降低，直至完全消失。

结合使用 State 和 Wwise Motion Source

通常，在[使用 RTPC](#)的同时，可定义 State（状态）专用偏置 Value，以便根据当前激活的 State 来增强或减弱振动效果。

结合使用 Wwise Motion Source 插件和 State 来创建振动效果：

1. 将[振动专用对象](#)加载到 Property Editor（属性编辑器）中。
2. 在 Contents Editor（内容编辑器）中，双击所列 Wwise Motion Source 将其加载到 Source Editor（源编辑器）中。
3. 选择 States（状态）选项卡。
4. 单击 Add State Group（添加状态组）选择器按钮，以便选择所需 State Group（状态组）。

这时将在选项卡的表格中一并显示 Driver 列以及 State Group 和所含 State。



技巧

若未显示 Driver 列，请右键单击表格标题区，并选择 Configure Columns...（配置列...）。这时将显示 State Properties（状态属性）对话框。在此，可选择要显示的 Driver 列。

5. 向右或向左移动 State 对应的 Driver Value 滑杆，使其值不为 0。

若使用[正确设置的振动输出](#)播放对象，则将在对应 Actuator 中使用相应 Driver，并激活相应 State，然后，定义的振动效果将按照相应值来进行偏置。



备注

有多种 Motion Preset（振动预设）可作为 Factory Asset（出厂素材）导入。若在创建工程时未添加这些预设，请从 Project（工程）菜单选择 Import Factory Assets...（导入出厂素材...），以便打开对话框并将预设及其他素材一并添加到工程中。这些预设包含各种带 Motion Source 插件的 Sound SFX 对象，并经过预定义，输出至 Actor-Mixer Hierarchy（角色混音器层级结构）内的 Motion Factory Bus（振动出厂总线）。

相关主题

- [跨平台制作](#)
- [使用 RTPC](#)

(不建议使用) Wwise Motion Generator

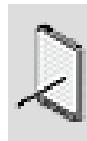
您可以通过这个特殊插件为各个平台的控制器创建和定制若干条振动曲线。各条振动曲线定义了特定时间段内振动效果的强度，还可以使用 ADSR 包络控制的选项来进一步定义曲线的形状。

默认情况下，每条振动曲线在整个时间段内的强度值为 1（即 100%），会生成完整的振动。您可以移动现有控制点，也可以添加新的点，以创建所需的效果。

还可定义各条曲线段的形状，来创建更详细、更复杂的振动曲线。曲线段是两个控制点之间的部分。您可选择各种曲线形状，包括线性曲线、恒定曲线、对数曲线、幂数曲线和 S 曲线。要获取指定曲线形状的详细信息，以及使用图形视图的其它信息，请参阅 [第 43 章 了解坐标图视图](#)。

您可创建以下任一曲线：

- Default Small —— 默认曲线，用于若干个振动发生器设备的小型马达。
- Default Large —— 默认曲线，用于若干个振动发生器设备的大型马达。
- Default Left Trigger —— 默认曲线，用于一个或多个振动发生器设备的左侧 Trigger 马达。
- Default Left Trigger —— 默认曲线，用于一个或多个振动发生器设备的右侧 Trigger 马达。
- Xbox One™ Small —— 指派给 Xbox One 控制器中小型马达的曲线。
- Xbox One™ Small —— 指派给 Xbox One 控制器中小型马达的曲线。
- Xbox One™ Left Trigger —— 指派给 Xbox One 控制器左侧 Trigger 马达的曲线。
- Xbox One™ Right Trigger —— 指派给 Xbox One 控制器右侧 Trigger 马达的曲线。



备注

Trigger 马达（Default 和 Xbox One）无法在 Wwise 创作工具中激活。只有 PC UWP 应用能使用它们。

- Trigger 马达（Default 和 Xbox One）无法在 Wwise 创作工具中激活。
- PlayStation®4 Large —— 指派给 PlayStation 4 控制器中大型马达的曲线。
- Switch™ Small —— 指派给 Switch 控制器中小型马达的曲线。
- Switch™ Large —— 指派给 Switch 控制器中大型马达的曲线。

但为了节省时间，您可能会需要复用部分曲线。您也可以让一个马达单独振动（solo），来更轻松地微调其振动曲线。让一个马达单独振动时，播放过程中其它所有马达将被静止。

(不建议使用) 使用 Wwise Motion Generator 插件创建振动效果的方法如下：

1. 将 Motion 加载至属性编辑器。
2. 在 Contents Editor（内容编辑器）中，单击标题栏中的 Add Source（添加源）按钮。
3. 从列表中，点击 Wwise Motion Generator。

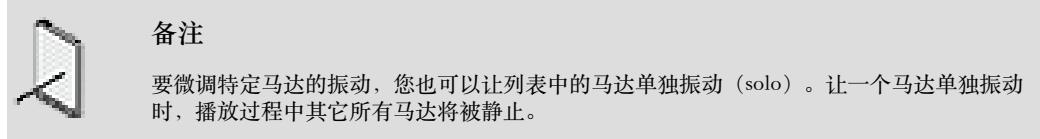
振动源被添加至 Controller 标题栏下。

4. 双击源，将其加载至 Property Editor 中。
5. 从 Curve 列表中，选择 Default Small 曲线。

曲线将显示在坐标图视图中。

6. 执行以下任一操作，控制 Default Small 振动曲线：

- 在曲线上添加点。
- 将控制点拖至新位置，也可在 X 和 Y 坐标框中输入特定值。
- 定义曲线段的形状。



7. 对于其它曲线，执行以下任一操作：

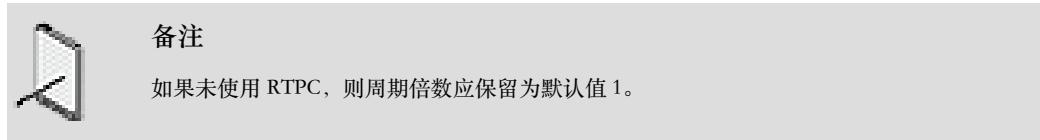
从 Usage 列表中选择 Custom，然后创建新的振动曲线。

从 Usage 列表中选择现有曲线名称，重用该曲线。

8. 在 Period 文本框中，指定完成一个曲线周期所用的时间量。

该值定义了 X 轴的长度。

9. 如果您要使用 RTPC 实时修改振动效果的周期长度，则必须在 Period Multiplier（周期倍数）文本框中输入一个值。更改周期倍数值时，曲线将被拉伸或压缩，以适应新的周期。



10. 要指定振动效果的时长，请选择以下任一选项：

- One period —— 使用一个周期作为振动效果的时长，该期间定义为周期乘以周期倍数。
- Fixed duration —— 为振动效果指定固定时长。根据“Fixed duration”的指定值，振动效果器可完成任意数量的曲线周期。如果固定时长值不是周期与周期倍数乘积的倍数，则最后一个曲线周期将是不完整的。
- Envelope —— 使用 ADSR 包络来控制曲线随时间变化的形状。

相关主题

- [通过现有声音生成振动](#)

为振动来构建输出结构

振动数据类似于音频数据，但混音方式通常与音频不同。要输出到振动设备，需要将振动数据输出至单独的总线层级结构中。您可以在 Master-Mixer Hierarchy 中创建单独的层级结构，只需创建一个新的 Master Bus 并为其分配 Wwise 振动设备即可。

创建启用振动的 Master Bus 的方法如下：

1. 打开 Project Explorer。
2. 在 Audio 选项卡的 Master-Mixer Hierarchy 中，右键单击 Default Work Unit。

3. 从菜单中选择 New Child，然后选择 Audio Bus。

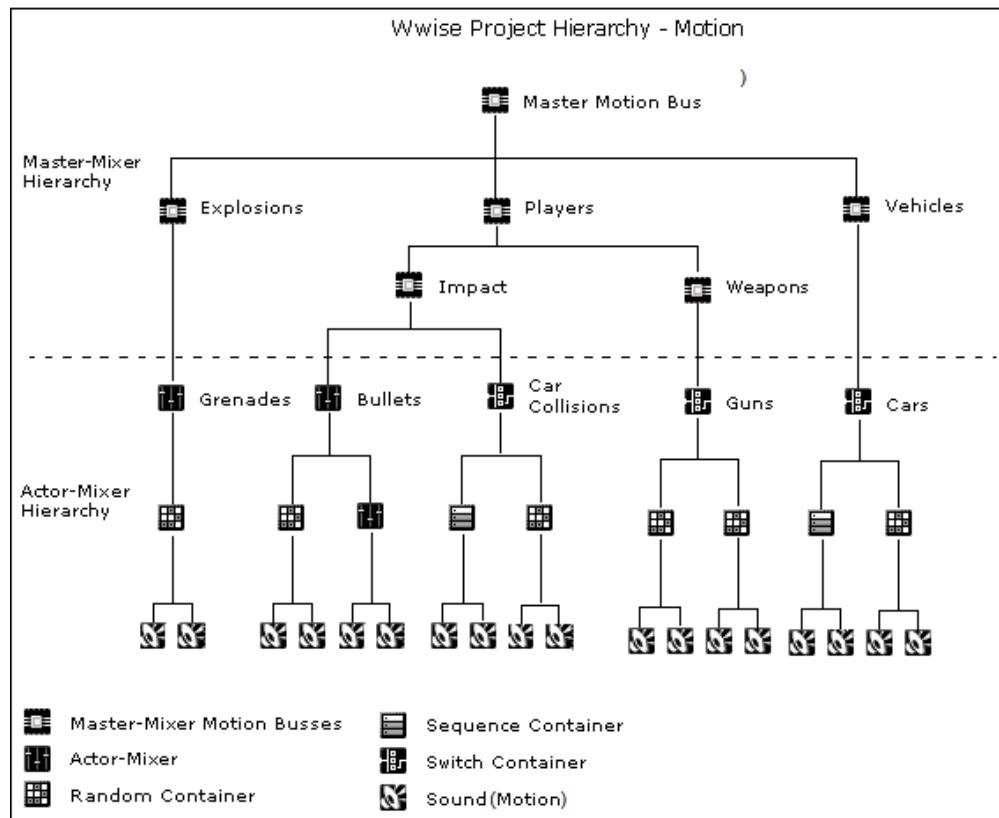
总线将被创建，其属性页面将打开。

4. 在 Audio Bus 的 Property Editor 中，找到 Audio Device 属性并单击选择按钮 (">>")。

可用的音频设备插件列表将出现。

5. 从列表中选择 Wwise_Motion。

您可以在工程中创建无限数量的 Master Bus。在构建总线结构时，振动总线和音频总线具有相同的功能、规则和条件。您可以将任意振动源或音频源路由到启用振动的总线。两种情况下，信号都将通过振动总线关联的 Audio Device 插件转换为振动数据。



有关如何为工程构建输出结构的详细信息，请参阅[构建输出总线结构](#)。

振动相关技巧和经验总结

创建游戏振动之前，您可能需要查看 SDK 文档。SDK 文档中提供了一系列可以帮您更好地管理 Wwise 中振动的示例、技巧和最佳实践。您也应该查看下面的[振动设备的限制](#)。

振动相关的 SDK 文档

- [集成 Secondary Outputs](#)
- [集成 Wwise Motion](#)

振动设备的限制

- 游戏控制器的播放能力有限。游戏控制器只能播放有限范围的振动效果，游戏玩家很难区分不同类型的振动效果，特别是当同时播放多个效果的时候。因此，最好能够限制为每次仅播放一个振动源。
- 将游戏控制器连接至大功率的 USB 端口。如果振动对象未能在游戏控制器中播放，则请确保该游戏控制器已连接至大功率的 USB 端口。如果 USB 端口没有足够的功率运行振动设备，则系统会卸载设备以保护操作系统或设备本身。计算机前方的 USB 端口通常不具有足够的功率，无法运行振动设备，因此您需要将振动设备连接至计算机背面的 USB 端口。

部分 IV. 与游戏互动



15. 管理 Event	326
概述	327
创建 Event	331
处理 Event	337
事件技巧和经验总结	340
16. 管理动态对话	341
概述	342
理解 Dynamic Dialogue 系统	342
创建 Dialogue Event	345
处理 Dialogue Event	354
Dialogue Event 技巧和经验总结	357
17. 使用 State	360
概述	361
使用 State	363
为 State Group 内的 State 之间的切换设置过渡	366
将 State 指派给对象和总线	367
State 技巧和经验总结	371
18. 使用 Switch	372
概述	373
使用 Switch	374
将 Game Parameter 值映射到 Switch	376
Switch 技巧和经验总结	377
19. 使用 RTPC	379
概述	380
管理 RTPC 中使用的 Game Parameter	382
使用 Game Parameter 控制属性值	386
使用 LFO	393
使用包络	394
使用时间	397
查看 Game Object	398
使用旁链压缩	398
RTPC 技巧与最佳实践	404
20. 使用 Trigger	406
概述	407
使用 Trigger	408
21. 将 State 和 State Group 用于动态对话	410
概述	411
使用 State Group	411

第 15 章 管理 Event

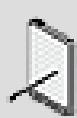
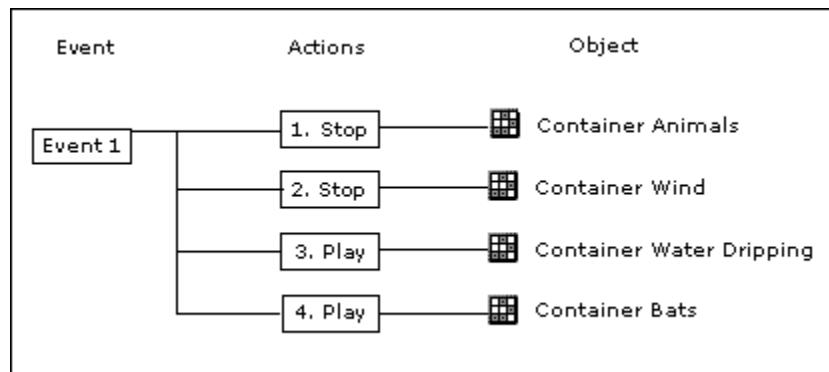
概述	327
创建 Event	331
处理 Event	337
事件技巧和经验总结	340

概述

Wwise 使用 Event（事件）来驱动游戏中的声音、音乐、对话和振动。事件可确定在游戏中的任意时刻上播放何种声音、音乐、振动或对话。为尽可能多地满足各种情况的要求，提供了两种 Event：

- 动作 Event
- Dialogue events（对白事件）

“动作”事件包含若干个动作（Action），这些动作会被用于工程层级结构内不同声音、音乐或振动结构。您选择的动作将指定 Wwise 对象是否播放、暂停、停止等。例如，以下事件可能说明一款游戏中的某处，角色离开多风的田野进入神秘洞穴。



备注

在大多数情况下，Play 事件与 Stop 动作事件甚至 Pause 或 Resume 事件配对。如果不想为这些类型的动作分别创建单独的事件，则作为替代，您可以使用 Wwise SDK 对声音执行这些动作。有关详细信息，请参阅[以编程方式停止、暂停和恢复声音](#)。

Dialogue Events, on the other hand, use a type of decision tree containing 状态 Groups to dynamically determine what 对象 is played. 有关对白事件的详细信息，请参阅[第 16 章 管理动态对话](#)。

在 Wwise 中创建“动作”事件后，这些事件可集成到游戏引擎中，以便在游戏中的适当时间调用这些事件。由于游戏引擎使用事件名称或 ID，因此您可以创建事件，将这些事件集成到游戏中去，之后建立并微调事件的内容，而不必将事件重新集成到游戏中去。只要事件名称或 ID 未更改，就无需额外编程。这为您试用不同对象、添加或删除某些对象以及微调对象之间的过渡提供了极大的灵活性。

为了帮助您轻松识别界面中的事件，事件由以下图标来代表。

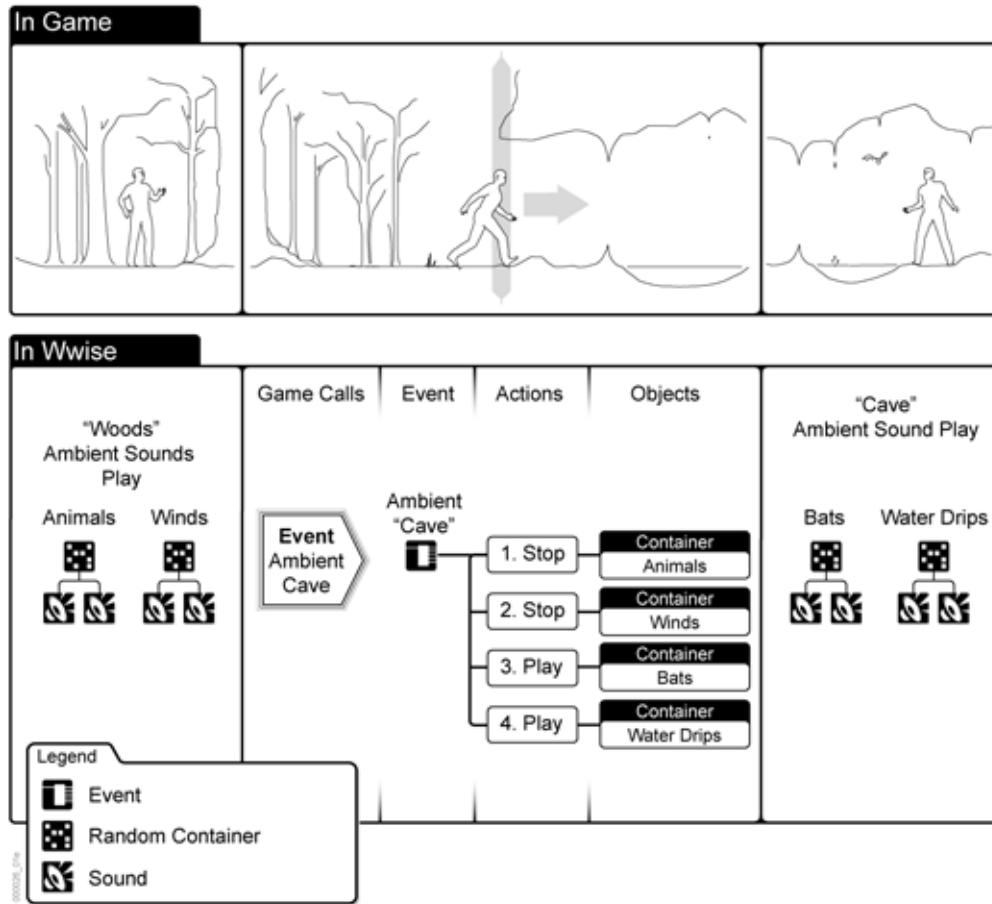
图标	代表
	Event

使用 Event —— 示例

假定您正在创建一款奇幻角色扮演游戏。您知道在游戏的一关中，角色将从森林进入洞穴。您希望在角色进入洞穴时环境音效发生改变。在工程开始阶段，您可以在

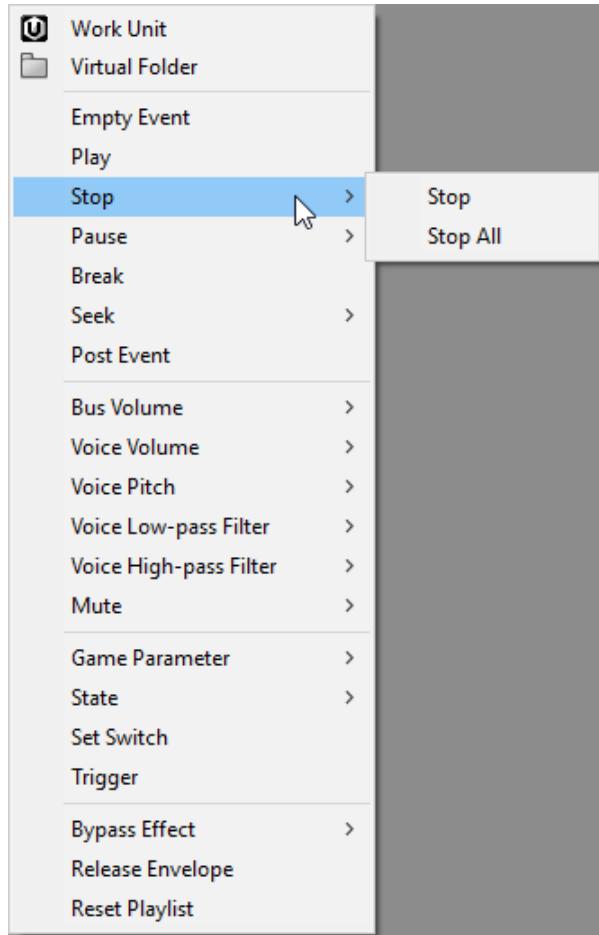
Wwise 中使用临时或占位符声音创建一个事件（Event）。该事件将包含一系列动作，这些动作将停止环境“森林”声音并播放环境“洞穴”声音。在创建该事件后，该事件可集成到游戏中，接下来就可以在适当时间触发该事件了。由于在初始集成之后无需额外编程，所以您可以试用不同的声音、添加和删除动作，以及更改动作属性，直到声音听上去满意为止。

下图说明了游戏中动作、事件以及播放的声音之间的关系。

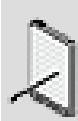


事件 Action 的类型

Wwise 自带多种动作（Action），这些动作可用于驱动游戏中声音、音乐和振动。动作按类别进行分组，每个类别都包含一系列供您选择的动作。



每个动作还有一组属性（例如延迟和淡变），可用于更好地管理输入和输出的对象。下表介绍了所有的事件动作。



备注

以下为一般描述，没有考虑每个 Action 的 Scope 设置。Scope 可以是 Game Object 或 Global，用于指定存在多个游戏对象时，动作的作用范围。

事件动作类型	描述
Empty Event	不包含动作或对象。
Play	播放关联对象。
Stop > Stop	停止播放关联对象。
Stop > Stop All	停止所有对象的播放，但可以添加例外。
Pause > Pause	暂停关联对象的播放。
Pause > Pause All	停止所有对象的播放，但可以添加例外。
Resume > Resume	恢复播放之前暂停了的关联对象。
Resume > Resume All	恢复所有暂停对象的播放，但可以添加例外。
Break	停止播放循环声音或连续容器，同时允许当前对象完成播放。
Seek > Seek	更改关联 Wwise 对象的播放位置。此操作不会影响当前未播放的对象。

事件动作类型	描述
Seek > Seek All	更改所有 Wwise 对象的播放位置，但可以添加例外。此操作不会影响当前未播放的对象。
Post Event	从事件内触发别的事件。
Bus Volume > Set Voice Volume	更改关联总线的音量电平。
Bus Volume > Reset Volume	将关联总线的音量复位至其原始电平。
Bus Volume > Reset Volume All	将所有总线的音量复位至其原始值，但可以添加例外。
Voice Volume > Set Voice Volume	更改关联对象的音量电平。
Voice Volume > Reset Volume	将关联对象的音量恢复至其原始电平。
Voice Volume > Reset Volume All	将所有对象的音量复位至其原始值，但可以添加例外。
Voice Pitch > Set Voice Pitch	更改关联对象的音高。
Voice Pitch > Reset Pitch	重置音高，将关联对象的音高复位至其原始值。
Voice Pitch > Reset Voice Pitch All	将所有对象的音高复位至其原始值，但可添加例外。
Voice Low-pass Filter > Set Voice Low-pass Filter	更改作用于关联 Wwise 对象的低通滤波器效果量。
Voice Low-pass Filter > Reset Voice Low-pass Filter	将作用于关联 Wwise 对象的低通滤波器效果量复位至其原始值。
Voice Low-pass Filter > Reset Voice Low-pass Filter All	将所有 Wwise 对象的 Low-Pass Filter 复位至其原始值，但可以添加例外。
Voice High-pass Filter > Set Voice High-pass Filter	更改应用于关联 Wwise 对象的高通滤波器数量。
Voice High-pass Filter > Reset Voice High-pass Filter	将应用于关联 Wwise 对象的高通滤波器恢复至其原始值。
Voice High-pass Filter > Reset Voice High-pass Filter All	将所有 Wwise 对象的 High-Pass Filter 复位至其原始值，但可以添加例外。
Mute > Mute	将关联对象静音。
Mute > Unmute	将关联对象恢复为其原始“静音前”音量电平。
Mute > Unmute All	将所有对象复位至静音前的原始电平，但可以添加例外。
Game Parameter > Set Game Parameter	更改游戏参数值。
Game Parameter > Reset Game Parameter	恢复游戏参数的原始值。
States > Set State	激活特定状态。
States > Enable State	在应用 Disable State Action 后，为相关 Wwise 对象重新启用 State。
States > Disable State	为关联 Wwise 对象禁用状态。
Set Switch	激活特定切换开关。
Trigger (触发器)	调用 Trigger，以启动 Stinger。
Bypass Effect > Enable Bypass	旁通作用于关联 Wwise 对象的效果器。
Bypass Effect > Disable Bypass	删除效果器旁通会将效果器重新作用于关联 Wwise 对象。
Bypass Effect > Reset Bypass Effect	重置旁通效果器。将关联对象的旁通效果器选项复位至其原始设置。
Bypass Effect > Reset Bypass Effect All)	将所有 Wwise 对象的旁通效果选项复位至其原始值，但可以添加例外。
Release Envelope	释放与 Wwise 对象关联的包络。

事件动作类型	描述
Reset Playlist	将指定随机容器/序列容器的播放列表重置为初始状态。这不会影响连续模式播放，也不会影响当前播放的声音。

创建 Event

游戏中的所有对象都是由 Event 驱动的。事件创建过程涉及以下步骤：

- [创建新 Event](#)
- [将 Action 添加到 Event](#)
- [将目标指派给 Event Action](#)
- [定义事件 Action 的作用域](#)
- [设置事件 Action 的属性](#)

为了给您提供额外的可控性和灵活性，事件可执行一个动作或一系列动作。事件的管理可使用事件编辑器来完成。

在跨平台创作时，您可能会需要从特定平台弃用某些动作。在默认情况下，所有 Action 都包含在 Event 中，但是您可以根据平台进行自定义。有关跨平台创作的详细信息，请参阅[从平台中弃用工程元素](#)。

如果您作为团队的一份子参与同一工程，则可以将事件指派给不同的工作单元，以便各个团队成员都可同时处理不同事件。有关处理工作单元的详细信息，请参阅[将工程分成 Work Units](#)。

创建新 Event

创建新 Event 时，可以执行以下操作之一：

- 创建不包含动作或目标的空 Event。
- 创建包含特定动作的事件。
- 创建不包含动作或目标的空 Event。



技巧

您还可以在 Event Viewer（事件查看器）中通过右键点击来创建事件。在通过 Event Viewer 添加事件时，您还必须将事件指派给特定工作单元。

创建空 Event 的方法如下：

1. 在 Project Explorer 中，切换至 Events 选项卡。
2. 执行以下操作之一：
 - 选择工作单元或虚拟文件夹，然后点击 Project Explorer 工具栏中的 Event 图标。
 - 右键点击工作单元或虚拟文件夹，然后从快捷菜单中选择 New Child > Empty Event（空事件）。

新事件已在您在 Project Explorer 中选择的工作单元或虚拟文件夹内创建成功。

3. 将默认名称替换为最适合事件的名称。



备注

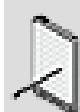
事件名称中只能包含不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

创建包含动作的事件的方法如下：

1. 在 Project Explorer 中，切换至 Events 选项卡。
2. 右键点击要添加事件的工作单元或虚拟文件夹。
3. 从快捷菜单中，选择 New Child 以显示事件动作列表。
4. 在 Action 列表中选择 Action 类别或 Action。（选择前者时，将显示包含一系列 Action 的子菜单；请从中选择一个 Action。）

包含所需动作的新事件已在您在 Project Explorer 中选择的工作单元内创建成功。

5. 将默认名称替换为最适合事件的名称。



备注

事件名称中只能包含不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

创建包含 Action 和目标的事件方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，选择事件中要包含的一个或多个目标，然后右键单击选项。
此时将会显示快捷菜单。
2. 选择以下选项之一：
 - New Event（新建事件），创建包含所需对象的事件。
 - New Events (One event per object)（新建事件（每个对象一个事件）），为每个所选对象创建一个事件。
 - New Event (Single event for all objects)（新建事件（所有对象一个事件）），创建一个包含所有所选对象的事件。
3. 在 Action 列表中选择 Action 类别或 Action。（选择前者时，将显示包含一系列 Action 的子菜单；请从中选择一个 Action。）

于是我们在事件编辑器中创建成功了若干个事件，里面包含所选的动作和对象。

4. 在 Name（名称）字段中，将默认名称替换为最适合事件的名称。



备注

事件名称中只能包含不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

相关主题

- [将 Action 添加到 Event](#)

- 将目标指派给 Event Action
- 定义事件 Action 的作用域
- 设置事件 Action 的属性

将 Action 添加到 Event

必须定义事件中要包含的 Action。每个事件都可包含若干个动作。

将动作添加到事件的方法如下：

1. 在 Event Editor 中，单击 Add >> 按钮。

此时将会显示动作列表。

2. 从 Action 列表中选择 Action 或 Action 类别。（选择前者时，将显示包含一系列 Action 的子菜单；请从中选择一个 Action。）

所选的 Action 将被添加到事件。

现在，可以给事件中的 Action [指派目标](#)或者继续为事件添加 Action。

可通过几种其它方式将 Action 添加到事件，如：

- 将 Project Explorer 中的一个或多个元素拖动到 Event Editor 的 Event Action 窗格空白处。随即将生成适用于元素的 Action，但是可以自由更改它。
- 在 Event Editor 的 Event Action 窗格空白处打开快捷菜单后，从其中选择 New Action 选项。显示的选项与单击 Add >> 按钮相同，后者请参阅上面的“将 Action 添加到 Event”列表。
- 可以使用标准 Copy 和 Paste 快捷方式或快捷菜单选项，同时复制和粘贴若干个 Action。

相关主题

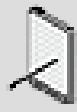
- [事件 Action 的类型](#)
- [创建新 Event](#)
- [将目标指派给 Event Action](#)
- [定义事件 Action 的作用域](#)
- [设置事件 Action 的属性](#)
- [播放 Event](#)
- [处理 Event](#)
- [重命名 Event](#)
- [删除 Event 中的 Action](#)

将目标指派给 Event Action

大多数事件动作必须指派给特定对象、结构或游戏同步器。包含一个或多个 Action 但未关联目标的事件称为 Orphaned Event（落单事件）。这些落单事件将显示在 Event Viewer 的 Orphans（落单）选项卡中。这些落单事件也会在您为工程生成完好度报告（Integrity Report）时显示。

为了帮助您识别 Event 中对象的状态，对象名称将以以下一种颜色显示：

- 白色 —— 表示包含的对象。（在当前平台中。）
- 灰色 —— 表示未包含的对象。（在当前平台中。）
- 红色 —— 表示缺失关联对象的事件动作或当前工程缺失的对象。
- 黄色 —— 表示目前从当前工程中卸载的对象。



备注

将音乐对象指派给特定 Action（例如 Trigger、Set Switch 以及 Set State）时，请注意这些 Action 可能会延迟，这是因为在音乐对象中可能已经预先定义了这些特定 Action 发生的时刻。

将目标指派给 Event Action 的方法如下：

1. 在 Event Editor 中，选择要为其指派目标的 Action。
2. 点击 **Browse**（浏览）。

此时将会显示 Project Explorer - Browser。

3. 浏览层级结构并选择要指派给动作的对象。
4. 点击 **OK**。

对象于是指派给 Action 了。



技巧

您还可以通过将某对象从 Project Explorer 拖至 Event Action 列表中的动作，来将该对象指派给相应事件动作。

相关主题

- [创建新 Event](#)
- [将 Action 添加到 Event](#)
- [定义事件 Action 的作用域](#)
- [设置事件 Action 的属性](#)
- [播放 Event](#)

定义事件 Action 的作用域

创建事件时必须定义每个 Action 的 Scope。Scope 会指定 Action 在游戏对象中作用的范围。Scope 会指定 Action 在游戏对象中作用的范围。对于某些 Action，您可以选择 Scope；对于其它 Action，Scope 是预先定义好的。

例如，假定您为玩家离开游戏进入菜单这种情况创建了一个事件。此事件将播放“Enter_Menu”声音、暂停玩家相关的所有声音，并且将 State 设置为“Menu”。接下来，“Menu”State 的设置会让 Master Audio Bus 的音量降低 20 dB，但会将“Music”Audio Bus 的音量增加 20 dB。

这些事件中各个事件的范围如下所示：

事件动作类型	Scope (作用域)	备注
Play > Menu_Enter	Game Object	由于 Play Event 总是由一个游戏对象触发，作用域会设置为 Game Object。
Pause All	Global 或 Game Object	在该场景中，玩家角色即为调用 Event 的 Game Object。暂停玩家相关的声音，可以让您在阅读菜单时不受干扰。因此，虽然可以将 Scope 设置为 Global，但在这种情况下，使用 Game Object 范畴将允许玩家仍然听到游戏中的其它声音。
Set State > Menu	Global	<p>Set State 的作用域始终为 Global，原因是工程中所有使用该 State 的地方都会响应状态变化。在该场景中，我们在 Master Audio Bus 和“Music”Audio Bus 中为“Menu”State 进行了设置，让前者音量降低，后者音量增加。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <p>备注</p> <p>本示例旨在对 Scope 进行说明，并非最佳音频设计选择。实际上，如果两条总线都是 Mixing Bus，在音量更改时会出现非常短的延时，可能会发出类似 click 或 buzz 的噪声。此外，如果“Music”Audio Bus 上存在线性效果器，则不一定能够保持该总线的音量恒定。不过，在该情况下，我们还可以使用 Bypass Effect Action。</p> </div>

定义事件的作用域的方法如下：

- 从 Scope 列表中，选择以下选项之一：
 - Game object，将事件动作作用于触发事件的游戏对象。
 - Global，将事件动作作用于所有游戏对象。

相关主题

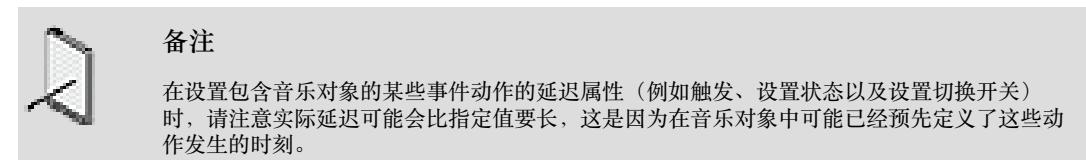
- [创建新 Event](#)
- [将 Action 添加到 Event](#)
- [将目标指派给 Event Action](#)
- [设置事件 Action 的属性](#)
- [播放 Event](#)

设置事件 Action 的属性

各个事件动作都有一组可用于进一步美化游戏声音、音乐和振动的相关属性。

各个动作都包含不同的属性，但这些属性分别属于以下类别之一：

- 延迟
- Transitions (过渡)
- 旁通效果器属性
- 音量、音高、LPF、游戏参数、跳转、状态或切换开关设置。



设置事件动作属性的方法如下：

1. 在 Event Editor 中，从 Event Action 窗格中选择 Action。

与所选 Action 关联的属性将显示在 Action Property 窗格中（右侧）。

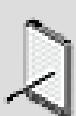
2. 根据需要为关联的属性指定值。

相关主题

- [创建新 Event](#)
- [将 Action 添加到 Event](#)
- [将目标指派给 Event Action](#)
- [定义事件 Action 的作用域](#)
- [播放 Event](#)

播放 Event

在创建过程中，您随时都可以试听事件。



备注

为了试听振动，相应振动设备必须连接到您的电脑。

播放事件的方法如下：

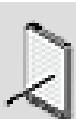
1. 执行以下操作之一：

- 在 Event Viewer 中选择一个事件。
- 将事件加载到 Event Editor（事件编辑器）中。

事件于是加载到 Transport Control 中来了。

2. 点击 Transport Control 中的 Play（播放）图标。

事件已播放。



备注

您还可以使用 Soundcaster（声音选角器）播放事件。有关使用 Soundcaster 的详细信息，请参阅[在 Soundcaster 中试听](#)。

相关主题

- [创建新 Event](#)
- [将 Action 添加到 Event](#)
- [将目标指派给 Event Action](#)
- [定义事件 Action 的作用域](#)
- [设置事件 Action 的属性](#)

处理 Event

由于在初始集成事件之后无需额外编程，所以您可以试用不同对象、更改现有对象属性、添加和删除动作以及更改动作属性，直到一切如您所愿为止。

管理事件时，您可以执行以下任务：

- [重命名 Event](#)
- [删除 Event 中的 Action](#)
- [替换指派给 Event Action 的目标](#)
- [删除 Event](#)

重命名 Event

在创建事件后 Wwise 会自动为该事件提供一个名称。这时最佳的做法是用更具描述性的名称对这个事件重新命名。各个事件不得重名，并且名称只能包含字母、数字和下划线。第一个字符必须是字母或下划线。

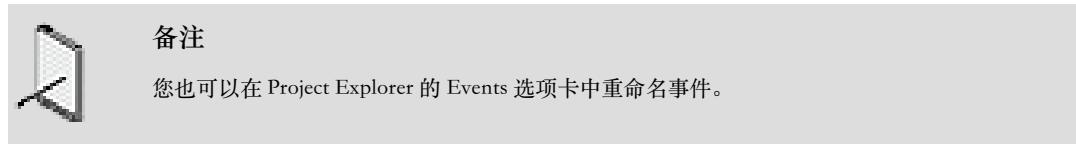
只有在绝对必要的情况下，您才需要在事件集成到游戏中后重命名该事件，因为重命名后将需要额外编程。

重命名事件的方法如下：

1. 在 Event Editor（事件编辑器）中，在 Name（名称）字段内点击鼠标。

该事件的名称处于突出显示状态。

2. 为事件键入新名称。



相关主题

- [创建新 Event](#)
- [删除 Event 中的 Action](#)
- [替换指派给 Event Action 的目标](#)
- [删除 Event](#)
- [事件技巧和经验总结](#)

删除 Event 中的 Action

在您试用和建立事件（Event）时，您可能需要从事件中删除若干个动作（Action）。只要事件名称不更改，您可以删除动作而无需额外编程。

删除 Event 中的 Action

1. 在 Event Editor 中，选择要从 Event 中删除的 Action。

2. 点击 Remove (移除) 按钮。

该动作于是就从事件中删除了。



技巧

您也可以删除 Event 中的 Action，方法是选择 Action 并按 Delete 键，或者选择 Action 快捷菜单中的 Delete 选项。

相关主题

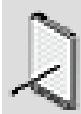
- [将 Action 添加到 Event](#)
- [重命名 Event](#)
- [替换指派给 Event Action 的目标](#)
- [删除 Event](#)

替换指派给 Event Action 的目标

您可能需要用其它对象替换特定对象，以判断它们在游戏中是否合适。即使在事件已集成到游戏中后，您仍可自由试用不同声音、动作、振动以及动作（Action）等。

替换指派给 Action 的目标对象的方法如下：

1. 在事件编辑器中，选择要为其替换对象的动作。
2. 点击 Browse... (浏览) 按钮，



备注

Browse... 按钮位于 Event Editor 的底部；但是从 Target 列的快捷菜单选择 Set Target 时，也会显示 Browse... 选项。

或打开所选 Action 目标的快捷菜单。

此时将会显示 Project Explorer - Browser。

3. 浏览层级结构并选择要指派给 Action 的新对象。
4. 点击 OK。

Action 的新目标对象将显示在 Target 列中。



技巧

将适用于 Action 类型的对象从 Project Explorer 拖至 Event Editor 中的 Event Action，替换 Action 的新目标对象将显示在 Target 列中。

如果对象不适合 Action，则无法拖动它。

相关主题

- [将目标指派给 Event Action](#)
- [创建新 Event](#)
- [删除 Event](#)

在 Schematic View 中显示事件的对象

如果要查看 Event 中包含的特定对象的工程结构或管线，可通过 Schematic 视图进行快速显示。

在 Schematic 视图中显示 Event 目标对象管线的方法如下：

1. 在 Event Editor 中，右键单击要查看其管线的目标对象。

此时将会显示快捷菜单。

2. 点击 Show in Schematic View（显示在对象网络图中）。

The sound or music 管线 is displayed in the Schematic view.

相关主题

- 替换指派给 Event Action 的目标
- 将目标指派给 Event Action
- 创建新 Event
- 删除 Event
- 播放 Event

删除 Event

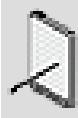
如果您不再需要某事件，则可以将其删除。在删除某个事件之前，您可能会需要验证它是否被团队中其它成员用在工程中的其它部分，以及它是否已包含在其中一个 SoundBank 中了。如果您或团队中的其他人删除了某个包含在 SoundBank 中的事件，那么这样会产生一个无效的事件。Wwise 不会自动从 SoundBank 中删除事件或其它无效工程元素，因此您将需要手动删除它们。为了帮助您在 SoundBank 内找到这些类型的事件和对象结构，Wwise 会通过 SoundBank 编辑器的 Add（添加）选项卡在名称之后添加单词“Missing”（缺失）。有关无效事件的详细信息，请参阅[从 SoundBank 中移除工程元素](#)。

删除事件的方法如下：

1. 执行以下操作之一：

- 在 Project Explorer 的 Events 选项卡中，右键点击要删除的事件，然后从快捷菜单中选择 >Delete Selection。
- 在 Event Viewer 中，点击要删除的事件，然后按 Delete 键。

于是事件删除掉了。



备注

如果您误删除了某事件，则可以撤消删除，方法是按 Ctrl+Z 或点击 Edit > Undo。

相关主题

- [创建新 Event](#)

- [删除 Event 中的 Action](#)
- [重命名 Event](#)
- [搜索 SoundBank 中的元素](#)

事件技巧和经验总结

以下各段将介绍一些可帮助您更好管理工程中事件的技巧和经验总结。

重命名事件

更改 Event 名称前, 请检查 Event 是如何集成到游戏中的。如果事件已使用包含事件名称的字符串或 Wwise_IDs.h 头文件中的 ID 进行了集成, 则更改名称将需要额外编程方能使用相应事件。

组织事件

您可以使用工作单元和虚拟文件夹来组织工程中的事件。在工程中有多人同时工作时, 将事件划分成多个工作单元可能会很有用处。例如, 各个团队成员可在其自己的个人事件工作单元中创建事件。将您的事件划分到不同的工作单元不会影响您的工作效率或工作流程, 而且这样做还能避免令人沮丧且费时的修复合并冲突。

将游戏事件与 Wwise 中的事件进行匹配

您可以使用 SoundBank 定义文件 (SoundBank definition file) 来跟踪哪些事件已经集成到游戏中、缺失哪些事件以及还需要在 Wwise 中创建哪些事件。音频程序员可生成游戏的事件列表, 然后您可以将定义文件导入 Wwise。您可以使用日志文件中的信息来对游戏中的事件与 Wwise 中创建或缺失的事件进行匹配。有关 SoundBank 定义文件的详细信息, 请参阅[通过导入定义文件填充 SoundBank](#)。

以编程方式停止、暂停和恢复声音

事件信息存储在声音引擎的默认内存池 (Default Memory Pool) 中。为避免占用默认内存池中的过多空间, 您可以使用 Wwise SDK 中的 [ExecuteActionOnEvent\(\)](#) 功能, 以编程方式停止、暂停和恢复声音。除了创建成对的播放/停止事件以外, 您可以创建单个 Play Event, 再配套调用 ExecuteActionOnEvent 函数来停止播放这个声音。如果您需要暂停和恢复声音, 则无需创建两个单独的事件, 只需使用上述相同的函数来执行暂停和 Resume 动作即可。这可以显著减少工程中的事件数量并且可释放默认内存池中的宝贵空间。为了提供额外的可控性, 在以编程方式停止、暂停和恢复声音时, 您还可以指定淡出时间。有关 ExecuteActionOnEvent 函数的详细信息, 请参阅 Wwise SDK 文档。

使用 Set Voice Pitch 事件创建音高包络

对于最自然的声音, 音高实际上会在一段时间内波动或改变, 直到其到达其稳定状态 (steady state) 或“保持” (sustain) 电平。要创建此效果器, 您可以使用音高包络。Wwise 不支持预编的音高包络, 但您可以通过创建包含多个连续 Set Voice Pitch 动作的事件来靠近该行为。事件中各个 Set Voice Pitch 动作都可在动作属性中的其中一个预定义曲线之后调整一段时间内的音高。当然, 序列中最后一个动作可能需要为 Reset Voice Pitch, 以便使游戏对象回到其原始状态。

第 16 章 管理动态对话

概述	342
理解 Dynamic Dialogue 系统	342
创建 Dialogue Event	345
处理 Dialogue Event	354
Dialogue Event 技巧和经验总结	357

概述

当今许多游戏（包括即使战略、体育和冒险游戏）都具有动态的或由游戏中发生动作驱动的音频组件。对于游戏中可存在的所有不同变量和结果，此类音频的确成为脚本作者和音频设计者的严峻挑战。要涵盖每种条件或结果的所有不同对话组合，您可能需成千上万的素材和复杂的切换容器层级结构。这可能导致内存使用成本非常高昂。为了简化开发过程并降低总内存占用，Wwise 推出了 Dynamic Dialogue（动态对话）系统。

Dynamic Dialogue 系统是一种在 Wwise 中建立和管理动态音频的高效轻量级方法。它使用决策树结构内的一套规则来确定要在游戏的任何特定时刻播放哪段对话。

这个系统已经很强大了，而 Wwise SDK 中还有其它功能使您能够在此基础上更进一步，创建运行时的动态序列，这非常适合体育游戏中的现场解说评论。在 Wwise 中，您可以将对话分解为个别单词或小短语，然后将这些对话指派给决策树结构内的特定路径。当游戏播放时，特定单词会提交给声音引擎，这些单词会在该引擎中动态连接在一起以形成流畅而精确到样本点的句子。因为单词和小短语是单独存在的，所以它们还可以多种方式进行组合并复用。有关将动态序列集成到游戏中的详细信息，请参阅 [Wwise SDK 文档](#)。



备注

虽然创建对白事件的初衷是处理游戏对话，但对白事件不只可以用于对白，在游戏中还可以有多种用途，包括处理各种地表类型上的脚步声。

理解 Dynamic Dialogue 系统

Wwise 的动态对白系统的核心是对白事件，它是一套规则或条件，用于决定播放哪一条对白。Dialogue Event 可用于表现游戏中存在的各种场景、条件或结果。为了确保涵盖所有情形，Wwise 还可以创建默认或后备条件。

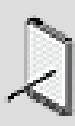
所有这些条件是通过使用一系列状态组和状态来定义的。这些状态组和状态可进行组合以创建路径，这些路径可定义游戏中的特定条件或结果。各条路径于是可以与 Wwise 中的特定声音对象进行关联。在游戏进行过程中，随着各个对白事件被调用，游戏会用对白事件中定义的路径来验证当前的条件。与游戏中当前情况以及各条路径的模式、概率和权重匹配的路径可确定播放哪段对话（如果有的话）。

例如，以下对白事件包含与体育游戏中各个玩家名称相关联的状态组。各个状态组里的各个值可组合起来以创建可能存在的不同路径或条件。在本例中，解说员可使用玩家的姓氏或全名。

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	

当没有状态与游戏中的当前情形相匹配时，用户可以创建一条由默认或后备状态构成的路径。这些“后备”路径包含若干个状态组而非状态，并且通常与一个更一般化的声音对象相关联。在前例中，后备路径被关联到声音对象“he（他）”上，而不是某个运动员的名字上。

在 Dialogue Event 中重现了所有条件后，即可将其集成到游戏引擎。当对白事件被游戏调用时，声音引擎返回与匹配路径相对应的音频对象，从而解决这个对白事件。声音引擎随后可以决定是否把这个音频对象插入到一个动态序列中用于播放。“返回一个音频对象”与“将其插入到一个动态序列中”的比例关系不必为1:1。这意味着对于各个被解决的对白事件（Resolved Dialogue Event），被返回的音频对象可以根据需要被多次加入到动态序列中。



备注

在创建动态对话时，您与音频程序员紧密合作至关重要，因为许多功能都仅在 Wwise SDK 中可用，包括将单词或短语连接在一起形成句子的能力。

由于游戏引擎使用的是对白事件的名称，因此用户可以先创建事件并把它们整合到游戏中，然后再对各个事件的内容进行构建和精雕细琢，此时就无需再把它们重新整合到游戏中了。这给了您极大的灵活性，您可以添加或移除状态，并且能够对不同的声音进行试验，所有这些都无需额外的编程。



注意

添加、删除和移动对白事件内的状态组将自动修改路径。虽然此类更改可在 Wwise 中轻松完成，但还是需要程序员介入，因为代码将需要相应更新。

为了帮助您轻松识别界面中的对白事件，对白事件由以下图标来代表。

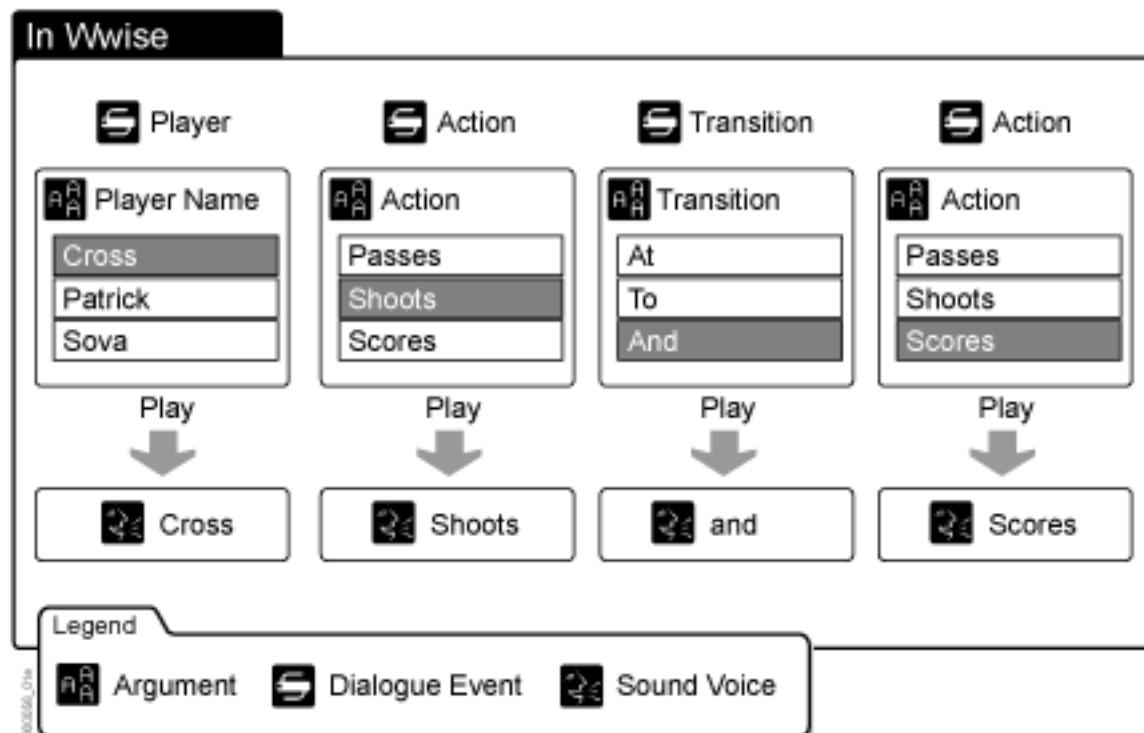
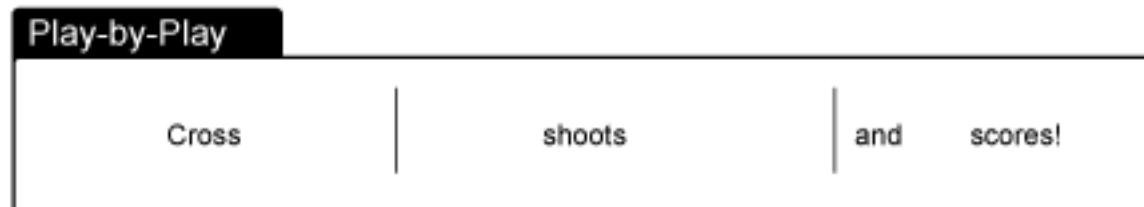
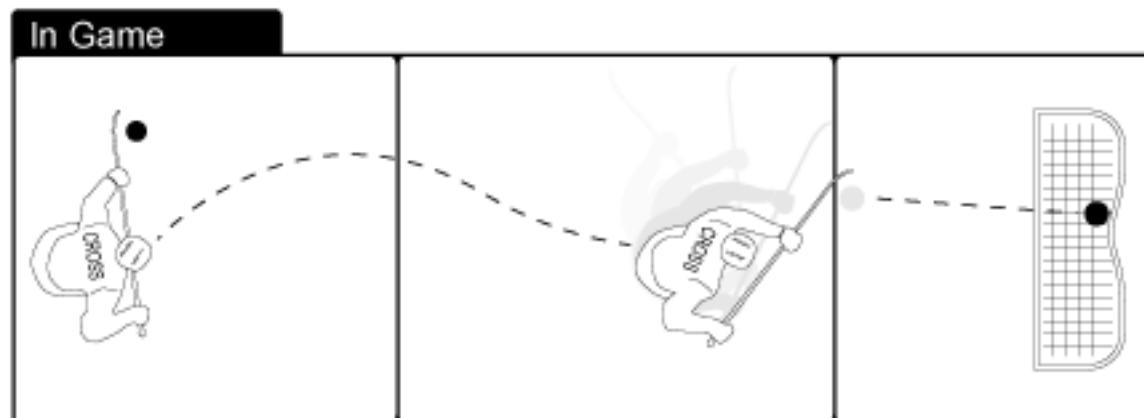
图标	代表
	对白事件

使用对白事件——示例

假定您正在创建一款有现场解说评论的冰球游戏。在玩家射门得分时，您希望现场解说对应游戏中的此动作。

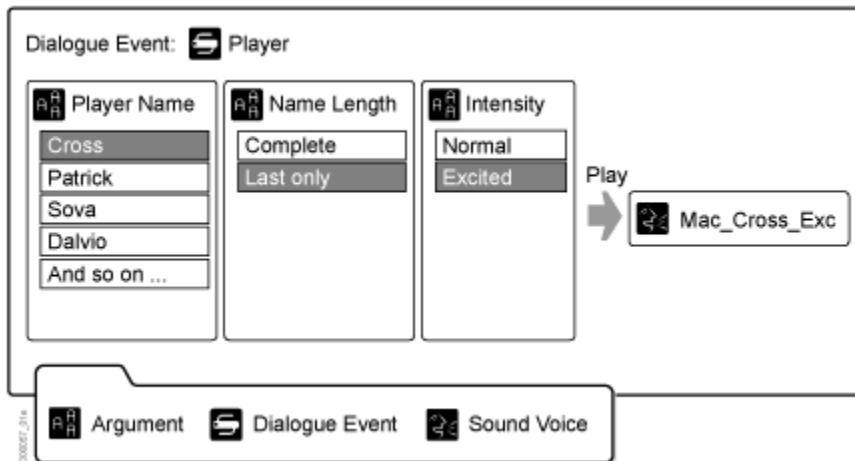
为了设置 Wwise 中不同的可能性和结果，您将需要为玩家、动作、过渡等创建对白事件。每个事件都包含一组您已创建好的相应状态组和状态。您必须为每个条件和结果创建一条路径，然后将适当的语音对象指派给各条路径。玩游戏期间，游戏将依据您在 Wwise 中定义的路径匹配当前状态，以确定要播放哪个语音对象。

下图说明了在 Wwise 中创建的对白事件如何能够生成现场解说“克劳斯（Cross，人名）射门，球进了”。



上图经过简化，以说明状态组和对白事件如何能够用于您的游戏。但是，在大多数情况下，游戏将需要更多包含多种不同状态组和状态的复杂对白事件。Wwise 使您能够建立各种更复杂的情景，可使游戏中的动态对话更真实。

下面展示玩家对白事件的一个复杂一点的例子。注意这个对白事件是如何包含多个状态组的。在您有多个状态组时，选中的状态会创建该路径。您可以为状态组和状态的每个组合创建一条路径，然后为每条路径指派一个对象。在游戏中符合特定路径内的每个状态时，指派给该路径的语音对象将会播放。

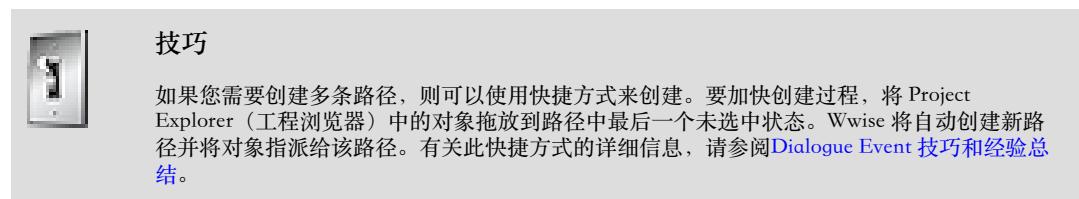


创建 Dialogue Event

游戏中动态对话的规则要在 Dialogue Event（对白事件）内定义。这些事件的创建涉及以下步骤：

- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 Path 重现游戏中的条件](#)
- [将对象指派给路径](#)

您可以在 Dialogue Event Editor（对白事件编辑器）中管理对白事件。



创建新对白事件

在项目的开始阶段，最好花些时间定义动态音频总体需求，方法是分析游戏并将其分解为可管理的组分。在您手头有所有必要的信息时，您可以创建游戏所需的所有对白事件的列表。在创建了这个列表之后，您在 Wwise 中处理和创建事件就会容易多了。



备注

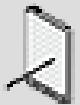
如果您的团队合作处理同一个工程，则可以将对白事件指派给不同工作单元，以便各个团队成员可同时处理不同的对白事件。有关处理工作单元的详细信息，请参阅[将工程分成 Work Units](#)。

创建新 Dialogue Event 的方法如下：

1. 在 Project Explorer 中，切换至 Events 选项卡。
2. 在 Dynamic Dialogue（动态对话）部分中，执行以下操作之一：
 - 选择工作单元或虚拟文件夹，然后点击 Project Explorer 工具栏中的 Dialogue Event（对白事件）图标。
 - 右键点击工作单元或虚拟文件夹，然后从快捷菜单中选择 New Child > Dialogue Event（新建子项 > 对白事件）。

新对白事件于是在您选择的工作单元中创建成功了。

3. 将默认名称替换为最适合对白事件的名称。



备注

Dialogue Event 名称只能包括不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

相关主题

- [将 State Group 添加到 Dialogue Event](#)
- [定义 Dialogue Event 的设置](#)
- [使用 Path 重现游戏中的条件](#)
- [使用 State 创建路径](#)
- [创建后备路径](#)
- [将对象指派给路径](#)
- [为路径设置概率和权重](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

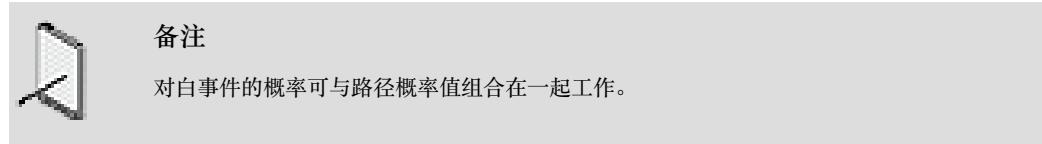
定义 Dialogue Event 的设置

对于各个对白事件（Dialogue Event），您需要定义概率设置和模式设置。概率设置有助于您用一定概率来控制对白事件具体会不会将音频对象提交到声音引擎来播放。一旦提交，声音引擎随后可确定是否将音频对象插入动态序列。值 100% 表示每当游戏调用对白事件时一定会将音频对象提交到声音引擎。而值为 0 表示绝不提交音频对象。如果您不希望每次触发对白事件时播放音频，则此选项会特别有用。例如，在格斗比赛中，您可能不希望解说员在格斗选手之间每次发生接触时都进行解说。为了避免此类情形中过度触发音频，您可以把对白事件的概率设置调低。

利用模式设置，您可以进一步控制几条路径同时匹配游戏当前条件这类情形中对白事件的行为。您可以选择两个选项中的一个：Best Match（最佳匹配）和 Weighted（加权）。

定义对白事件设置的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 为对白事件指定 0-100 中的一个概率值。概率值确定对白事件将音频对象提交给声音对象以进行播放的概率。



3. 当有多个预定义路径都符合运行时触发的 State 时，为了指定声音引擎将以何种方式选择路径，请选择 Mode 列表中选项之一：
 - **Best Match**: 最佳匹配。选择与运行时触发的 State 最匹配的路径。如果不能完全匹配，则将选择具有最少通配符 (*) 的路径。
 - **Weighted**: 加权。基于各路径的权重值，随机选择匹配路径之一。

相关主题

- [创建新对白事件](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 Path 重现游戏中的条件](#)
- [使用 State 创建路径](#)
- [创建后备路径](#)
- [将对象指派给路径](#)
- [为路径设置概率和权重](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

将 State Group 添加到 Dialogue Event

各个对白事件（Dialogue Event）包含一组反映游戏内存在的不同条件和结果的规则。这些规则是用 State Group（状态组）和 State（状态）来定义的。在将状态组添加到对白事件中去时，相应状态也会被添加进去。

将状态组添加到对白事件中的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
3. 在 State Groups（状态组）部分，选择您要添加到对白事件中的若干个状态组。
4. 将状态组拖至 Dialogue Event Editor 的 State Groups 窗格。

这些状态组和相应的状态就显示在 Dialogue Event Editor 中了。

相关主题

- [创建新对白事件](#)

- 定义 Dialogue Event 的设置
- 使用 Path 重现游戏中的条件
- 使用 State 创建路径
- 创建后备路径
- 将对象指派给路径
- 为路径设置概率和权重
- 删除 Dialogue Event
- 播放 Dialogue Event 内的对象

使用 Path 重现游戏中的条件

在 Dialogue Event 中添加了 State Group 和 State 后，您需要设置规则，用来表现游戏中的每个条件和结果。做到这点，需要将不同状态组和状态组合成路径（Path）。

由于涵盖每种情况或结果可能很困难，所以您最好创建每当声音引擎遇到尚未定义特定路径这种情形时可使用的后备或默认路径。这些后备路径使用状态组而非状态。

两种不同类型的路径如下所示：

- 使用 State 创建路径
- 创建后备路径

使用 State 创建路径

通过从对白事件内的各个不同状态组（State Group）中选择一种状态（State），您可以重建游戏中的不同条件和结果。选中的各个状态构成了一条路径。每条路径可与一段对话相关联，以便在游戏中出现特定的条件或结果时，播放适当的语音对象。



技巧

如果您需要创建多条路径，则可以使用快捷方式来创建。要加快创建过程，将 Project Explorer 中的对象拖放到路径中最后一个未选中状态上去。Wwise 将自动创建新路径并将对象指派给该路径。有关此快捷方式的详细信息，请参阅[Dialogue Event 技巧和经验总结](#)。

创建路径的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 为对白事件中各个状态组点击一个状态。

Add Path（添加路径）变成激活状态。

3. 点击 Add Path 创建一条路径。

路径于是就添加到路径列表了。

4. 通过选择不同的状态组合来继续添加新的路径。



备注

要删除某条路径，请选择该状态系列或该路径本身，然后点击 Remove Path（删除路径）。

相关主题

- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [创建后备路径](#)
- [将对象指派给路径](#)
- [为路径设置概率和权重](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

创建后备路径

由于重建游戏中存在的各个条件和结果可能非常困难、非常耗时并且非常昂贵，所以您会希望有某种后备机制能涵盖那些您没有想到或没有时间创建的情况。在 Wwise 中，可通过创建后备路径来实现这种机制。这些后备路径包含若干个状态组而非状态，并且通常与一个更常规的声音对象相关联。例如，在即时战略游戏中，某些情形下某个操作的结果可能无法确定，例如您军团的最终目的地。要涵盖此情形，您可以创建一条路径，其中使用至少一个状态组而非状态。此路径之后将指派给一个声音对象，该对象用来代表行动的开放式结局。在您的军团前往最终目的地的情况下，声音对象可能是“Your army is advancing”（您的军团正在向前推进）。

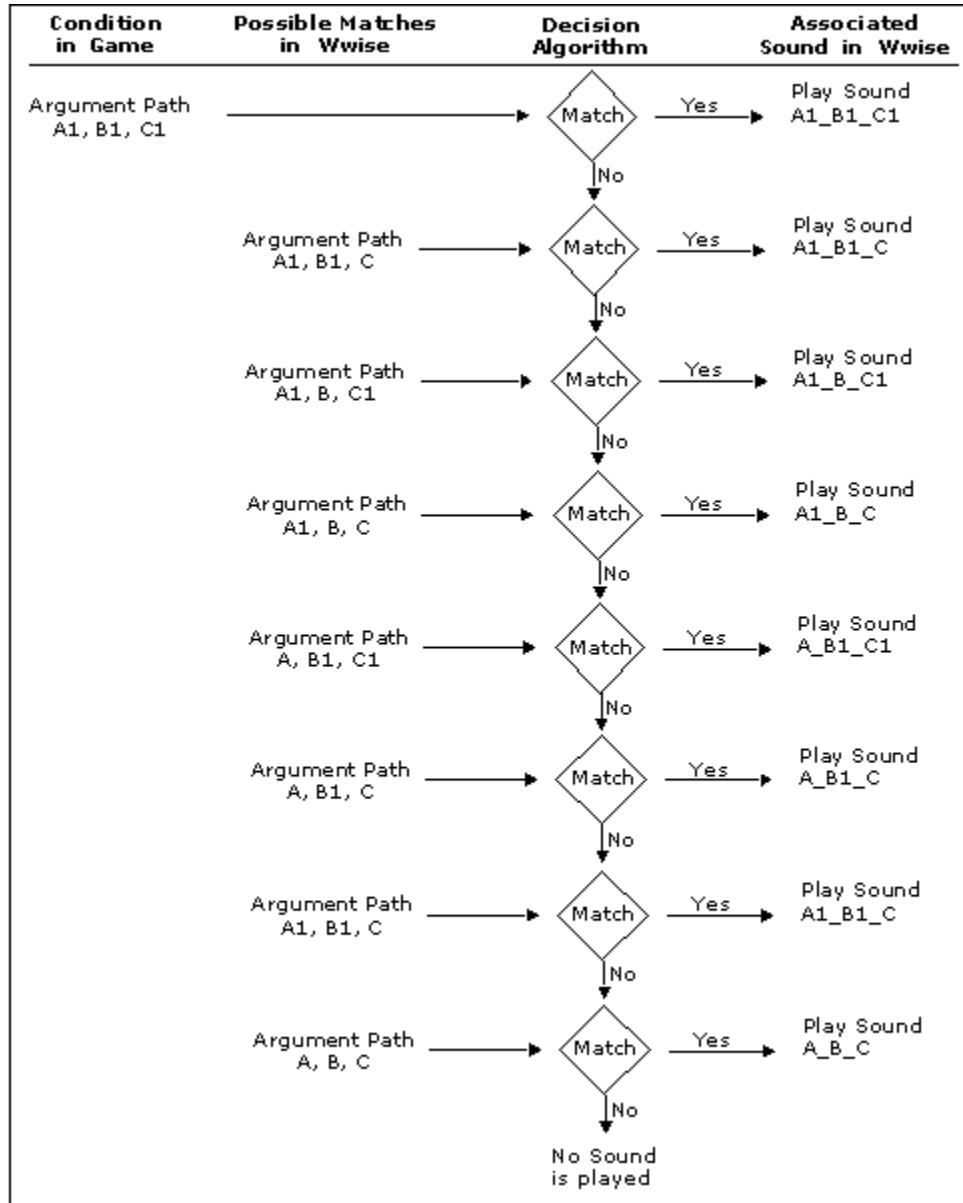
在为游戏创建对话之前，脚本作者了解 Wwise 中后备算法的工作方法至关重要。假定您有一个包含以下状态组和状态的对白事件：

Arguments	A	B	C
Argument paths	A1	B1	C1
	A2	B2	C2
	A3	B3	C3

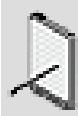
在游戏调用该对白事件时，该算法将尝试将游戏中的当前条件（例如（A1, B1, C1））与对白事件中的其中一条路径相匹配，以确定要播放的声音。在不只一条路径与游戏中当前条件匹配时，对白事件的模式将最终确定 Wwise 挑选路径和相应声音的方式。在对白事件处于“Best Match”模式时，Wwise 将根据对白事件中创建的路径验证游戏中的当前条件，来查找完全的匹配项。如果未找到完全匹配项或找到了匹配路径但没有关联的对白，则算法将在后备路径中查找最近似匹配项（如果存在的话）。

在对白事件处于“Weighted”模式时，Wwise 会查找所有匹配项，包括所有后备路径。从此匹配项列表中，该算法将查看每条路径的权重并依此确定选择哪条路径以及要播放哪段对话。

下图说明了“最佳匹配”决策算法如何验证所有可能匹配项以确定要播放哪个对象。



请注意决策算法如何从右到左查找后备。在我们的示例中，它会从 C 开始，然后查看 B，最后查看 A。如果所有其它可能组合都检查过了，则仅考虑最左侧的后备。



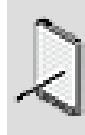
备注

您还可为对白事件创建通用路径。通用路径涵盖所有情况并且无论游戏触发何种状态，都可确保播放对白。

创建后备路径的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 点击 State Group 或 Dialogue Event 中每个 State Group 的 State。

Add Path 按钮变成激活状态。

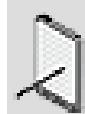


备注

在路径中需要选择至少一个状态组，以便创建后备路径。

3. 点击 Add Path 创建一条路径。

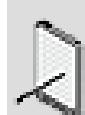
路径于是就添加到路径列表了。



备注

如果某个状态组已用于创建后备路径，则该状态组在路径名称中会用星号 (*) 加以表示。

4. 通过选择状态组和状态的不同组合，继续添加新路径。



备注

要删除某条路径，请选择该状态系列或该路径本身，然后点击 Remove Path（删除路径）。

相关主题

- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 State 创建路径](#)
- [将对象指派给路径](#)
- [为路径设置概率和权重](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

将对象指派给路径

在定义对白事件中的路径后，您需要将每条路径与一个对象相关联。通过将对象指派给路径，您可指定当路径内的所有状态都在游戏中成立的时候要播放的对象。要为每种情况添加多样性，您也可以将容器指派给路径。

您随时都可以更改指派给特定对象的路径，方法是只需选择列表中该对象，接着在 State Groups 窗格中选择一个新的路径，然后点击 Update Path（更新路径）按钮。此按钮将仅在 State Groups 窗格中新选择的路径尚未在列表中有相应的条目时才可用。

为了帮助您识别指派给路径的对象的状态，对象名称将以以下颜色之一显示：

- 白色 —— 表示包含的对象。（在当前平台中）
- 灰色 —— 表示未包含的对象。（在当前平台中）
- 红色 —— 表示当前工程中缺失的对象。
- 黄色 —— 表示目前从当前工程中卸载的对象。



技巧

您可在一步骤中完成创建一条路径并将对象指派给该路径。在创建路径时，只需将某对象从 Project Explorer 拖放至路径中最后一个未选中的状态即可。Wwise 将自动创建新路径并将对象指派给该路径。有关此快捷方式的详细信息，请参阅 [Dialogue Event 技巧和经验总结](#)。

将对象指派给路径的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 在路径列表中，点击与要为其指派对象的路径对应的 Browse (浏览) 按钮 (...)。
此时将会打开 Project Explorer —— Browser。
3. 选择要指派给路径的容器或对象。
4. 点击 OK。

对象已指派给该路径并显示在 Object (对象) 列中。



技巧

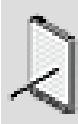
要从路径中删除某个对象，请右键点击该路径并从菜单中选择 Set to none (设置为无)。

相关主题

- [将设置从一个平台复制到另一个平台](#)
- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 Path 重现游戏中的条件](#)
- [使用 State 创建路径](#)
- [创建后备路径](#)
- [为路径设置概率和权重](#)
- [筛选路径列表](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

筛选路径列表

如果同一对白事件内的状态组和状态数量很大，则路径列表可能会变得非常满。如果列表变得无法管理，则您往往需要使用筛选器来进行筛选。



备注

您还可以通过点击 Path (路径) 标题栏按字母顺序对路径列表进行排序。

筛选路径列表的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。

2. 从 Path Filter (路径筛选器) 列表中，选择以下选项之一：
 - All -- 所有。显示所有创建的路径。
 - Current Selection -- 当前选中项。仅显示那些包含所选状态组或状态的路径。
3. 从 Filter 列表中，选择以下选择之一：
 - All —— 显示所有创建的路径。
 - Assigned object -- 指派对象。仅显示与对象关联的路径。
 - Missing -- 缺失。仅显示与已从工程中删除的对象相关联的路径。
 - None -- 无。仅显示与对象无关联的路径。

相关主题

- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 Path 重现游戏中的条件](#)
- [使用 State 创建路径](#)
- [创建后备路径](#)
- [将对象指派给路径](#)
- [为路径设置概率和权重](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

为路径设置概率和权重

Wwise 为您提供了额外的工具来管理游戏中路径的选择和播放。对于对白事件内的每条路径，您都可以指派概率和权重值。在有多条匹配路径时，这些属性定义了各条路径将被选择的概率以及将要播放相应音频对象的概率。

将概率和权重指派给路径的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 使用 Probability (概率) 滑杆将概率指派给路径：

概率值决定了某特定路径被选择进而使音频得以播放的概率。最终的播放概率是所选路径概率和对白事件概率的组合。

3. 使用 Weight (权重) 滑杆将权重指派给路径。

权重值有助于将某些路径的优先级设定为高于其它路径。在存在其它匹配路径的情况下，权重较高的路径更有可能被选择。在一条匹配路径的权重为 100 时，其它权重低于 100 的匹配路径会被自动弃用。在一条配路径的权重为 0 时，除非其它所有匹配路径的权重也都为 0，否则该路径会被弃用。



备注

Weight 选项仅在对白事件处于“Weighted”（加权）模式时可用。

相关主题

- [创建新对白事件](#)
- [定义 Dialogue Event 的设置](#)
- [将 State Group 添加到 Dialogue Event](#)
- [使用 Path 重现游戏中的条件](#)
- [使用 State 创建路径](#)
- [创建后备路径](#)
- [筛选路径列表](#)
- [将对象指派给路径](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

处理 Dialogue Event

如果您打算使用对白事件（Dialogue Event）来驱动游戏中的音频，则应该在工程开始之前与开发团队会面定义游戏中会有的所有不同条件和结果，这点至关重要。

由于路径是一组在游戏调用对白事件时需要验证的规则，所以对路径做的任何更改（包括状态组的数量和顺序）都将导致额外编程。通过提前花些时间定义所有不同规则、变量和条件，您可以避免问题并最大限度地减少额外编程需求。

管理对白事件时，您可以执行以下任务：

- [重命名 Dialogue Event](#)
- [重新排序对白事件中的 State Group](#)
- [从对白事件中删除 State Group](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

重命名 Dialogue Event

Wwise 自动在对白事件创建后为该对白事件提供一个名称。最佳做法是用更具描述性的名称来重新命名对白事件。各个事件不得重名，并且名称只能包含字母、数字和下划线。第一个字符必须是字母或下划线。



注意

只有在绝对必要的情况下，您才需要在对白事件集成到游戏中后重命名该对白事件，因为它将要求额外编程以重新集成名称更改。

重命名对白事件的方法如下：

1. 在 Project Explorer 的 Events 选项卡中，点击要重命名的对白事件。

该事件的名称处于突出显示状态。

2. 为事件键入新名称。



技巧

您还可以在 Dialogue Event Editor 中重命名事件。

相关主题

- [创建新对白事件](#)
- [将 State Group 添加到 Dialogue Event](#)
- [将对象指派给路径](#)
- [重新排序对白事件中的 State Group](#)
- [从对白事件中删除 State Group](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

重新排序对白事件中的 State Group

对白事件内的状态组顺序非常重要，因为它定义了每条路径上的值序列。如果状态组的顺序有错误，则您还可以随时为这些组重新排序。



注意

只有在绝对必要的情况下，您才需要在对白事件集成到游戏中后重新排序对白事件内的状态组，因为它将要求额外编程。如果您必须重新排序状态组，则请确保告知音频程序员以便他可以在代码中进行相应更改。

重新排序对白事件中状态组的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 在 Dialogue Event Editor 的 State Groups 窗格中，将某个状态组拖动到新位置。红线将会显示以帮助您定位该状态组。

现有路径会自动更新以反映状态组的新顺序。

相关主题

- [创建新对白事件](#)
- [将 State Group 添加到 Dialogue Event](#)
- [将对象指派给路径](#)
- [重命名 Dialogue Event](#)
- [从对白事件中删除 State Group](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

从对白事件中删除 State Group

在项目开发期间，您可以针对对白事件添加或删除若干个状态组。



注意

若非绝对必要，请勿在 Dialogue Event 集成到游戏后再为其添加或删除 State Group，因为这将要求额外编程。如果您必须添加或删除状态组，则请一定告知音频程序员以便他可以在代码中做相应更改。

从对白事件中删除状态组的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 右键点击您要删除的状态组，然后从快捷菜单中选择 Remove Column（删除列）。

状态组及其状态于是从对白事件中删除掉了。

相关主题

- [创建新对白事件](#)
- [将 State Group 添加到 Dialogue Event](#)
- [将对象指派给路径](#)
- [重命名 Dialogue Event](#)
- [重新排序对白事件中的 State Group](#)
- [删除 Dialogue Event](#)
- [播放 Dialogue Event 内的对象](#)

删除 Dialogue Event

如果您不再需要某个对白事件（Dialogue Event），则可以将其删除。在删除某个对白事件之前，您可能会需要验证团队中是否有别人在工程中的其它部分用到了它，以及是否某个 SoundBank（声音包）已经包含了它。如果您或团队中的其他人删除了某个包含在 SoundBank 中的对白事件，则它将创建一个无效的对白事件。Wwise 不会自动从 SoundBank 中删除对白事件，您将需要手动删除它们。为了帮助您跟踪和管理这些类型的对白事件，Wwise 会在 SoundBank Manager（SoundBank管理器）中显示事件的完整列表。有关删除无效事件的详细信息，请参阅[搜索 SoundBank 中的元素](#)。您还可以在工程完好度报告中查看无效对白事件的完整列表。有关使用完好度报告的详细信息，请参阅[排查工程的问题](#)。

删除事件的方法如下：

1. 在 Project Explorer 的 Events 选项卡中，右键点击您要删除的对白事件，然后从快捷菜单中选择 Delete Selection。

对白事件已删除。



备注

如果您错误删除了某对白事件，则您可以撤消删除，方法是按 Ctrl+Z 或点击 Edit > Undo（编辑 > 撤消）。

相关主题

- [创建新对白事件](#)

- [将 State Group 添加到 Dialogue Event](#)
- [将对象指派给路径](#)
- [重命名 Dialogue Event](#)
- [重新排序对白事件中的 State Group](#)
- [从对白事件中删除 State Group](#)
- [播放 Dialogue Event 内的对象](#)

播放 Dialogue Event 内的对象

创建过程中，您可以随时试听指派给对白事件（Dialogue Event）内任何一条路径的对象。

播放对白事件的方法如下：

1. 将对白事件加载到 Dialogue Event Editor。
2. 选择路径。

指派的声音对象会自动加载到 Transport Control（播放控制）。

3. 点击 Transport Control 中的 Play（播放）图标。

声音对象现在播放了。

相关主题

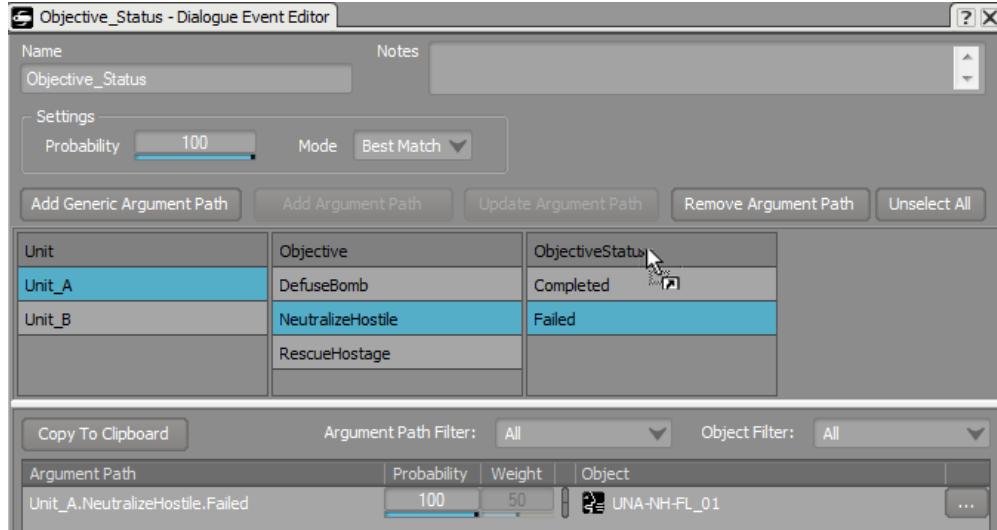
- [创建新对白事件](#)
- [将 State Group 添加到 Dialogue Event](#)
- [将对象指派给路径](#)
- [重命名 Dialogue Event](#)
- [重新排序对白事件中的 State Group](#)
- [从对白事件中删除 State Group](#)
- [删除 Dialogue Event](#)

Dialogue Event 技巧和经验总结

在创建对白事件（Dialogue Event）之前，您最好查看以下几节，这几节会为您提供一系列技巧和经验总结，帮助您高效创建和管理工程中的对白事件。

创建路径的快捷方式

如果您需要创建多条路径，则可以使用快捷方式来创建。此快捷方式用来确定路径，同时关联声音对象。在确定路径的内容之前，将相应用对象从 Project Explorer 中拖放至最后一个未选中状态组或状态上去。



Wwise 将自动创建新路径并将该对象指派给它。如果路径已经存在，那么当前指派给该路径的对象会被这个新对象所替代。

建立路径的策略

- 首先创建后备路径 - 作为一般规则，最好首先为对白事件创建所有后备路径，以确保您已覆盖所有条件和结果。在创建这些更多的一般后备路径之后，您可以开始建立更多的特定路径。如果您没有时间重建各个条件或开发周期后期添加了新条件，则已经覆盖了所有条件。
- 对状态组排序以最大限度提高算法效率 - 不太可能使用后备的状态组应放置在多个状态组中的首位。这会减少解析路径中回溯算法的运算量，并可提高动态对话系统的整体效率。

在下例中，名叫 Unit (单位) 的这个状态组被放在了首位，这是因为它不会有后备。此状态组的后备没有什么意义，因为与 Unit A 和 Unit B 相关联的是游戏中的特定角色，所以这种情况下不可能创建某个通用的“个人”状态组。由于为 Objective (目标) 和 ObjectiveStatus (目标状态) 状态组创建后备是有意义的，所以这些状态组应放在 Unit 状态组的后面。

管理动态对话

Dialogue Event Editor

Name	Notes
Objective_Status	

Settings

Probability: 100 Mode: Best Match

Add Generic Argument Path Add Argument Path Update Argument Path Remove Argument

Unit	Objective	ObjectiveStatus
Unit_A	DefuseBomb	Completed
Unit_B	NeutralizeHostile	Failed
	RescueHostage	

No fallback for this argument Fallback for this argument Fallback for this argument

第 17 章 使用 State

概述	361
使用 State	363
为 State Group 内的 State 之间的切换设置过渡	366
将 State 指派给对象和总线	367
State 技巧和经验总结	371

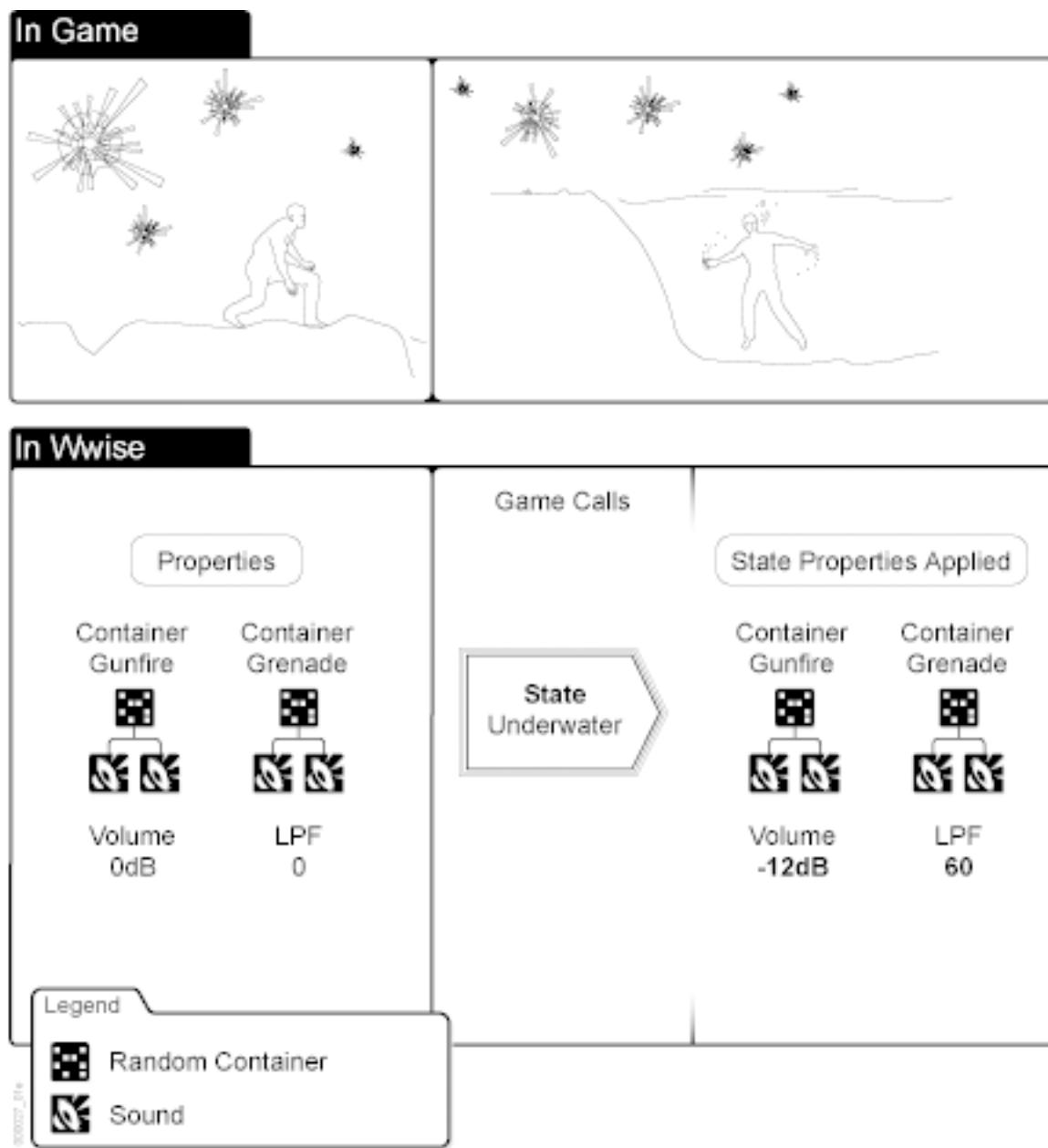
概述

利用最少的内存、CPU、素材和磁盘空间创建最动听音频，这是设计师日常面临的挑战。一种创新而高效的应对方案就是 State（状态）。使用 State 可以优化声音和音乐素材，允许您为同样的声音灵活创建不同的“Mixing Snapshot”（混音器快照），响应游戏中的变化并改变全局属性。通过改变声音或音乐对象的属性，无需添加新素材就可以创造性地匹配各种游戏情景。在规划工程时，您可以确定 State 在什么时候和什么地方效率最高、最具创意。

使用 State —— 示例

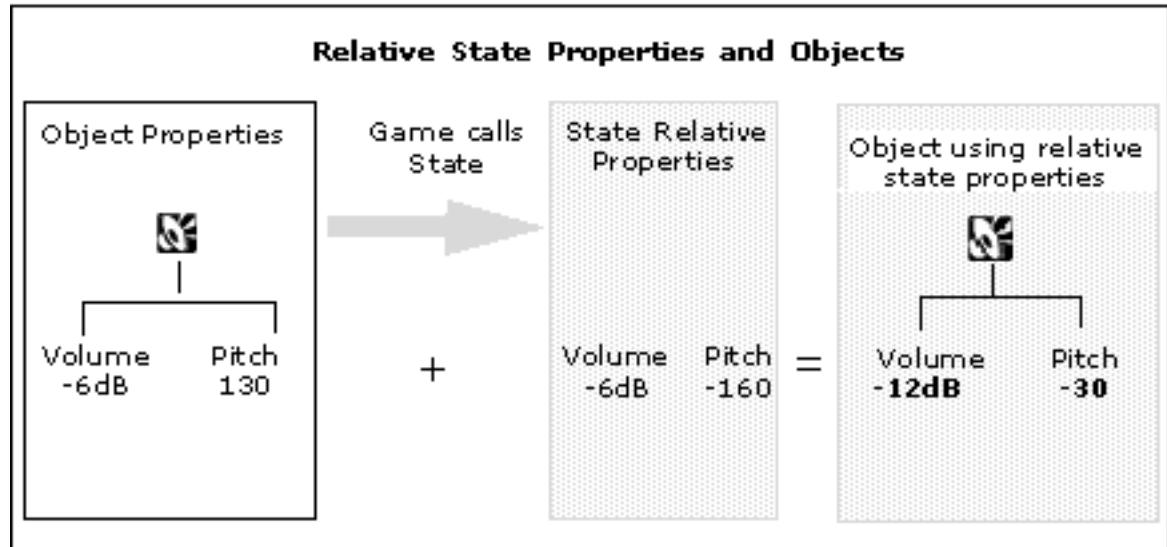
假设您要模拟角色进入水下时的声音处理，对此时正在播放的声音，可以用 State 来更改 Volume 和 Low-Pass Filter 值。这样的声音变化就可以模仿角色在水下听到的的枪声或手榴弹爆炸声。

下图演示了游戏调用 underwater（水下）State 时，枪声和手榴弹对象的 Volume 和 Low-Pass Filter 将如何改变。



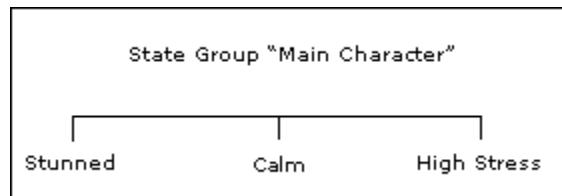
理解 State 属性值影响对象的方式

State（状态）属性值始终是相对的，即使用 State 时，对象属性受到的影响是累加的。



使用 State

State（状态）需要隶属于 State Group（状态组），才能供 Wwise 对象使用。可以按逻辑将各各种 State 划分成 State Group 来简化管理。例如，您可能发现将游戏主角相关的各 State 组合在一起会很实用。那么可以创建一个名为 Main Character（主角）的 State Group，为其添加 State，再应用到主角相关对象的属性上。游戏中，您知道主人公将可能经历以下状态：Stunned（吃惊）、Calm（镇定）和 High Stress（高度紧张），将它们组合在一起，然后设置每个 State 对应的属性变化会非常有用。



设置 State 结构后，您可以在 Property Editor（属性编辑器）中为对象指定 State Group，并按需设置 State 属性。

为帮您在界面中轻松找到 State Group 和 State，Wwise 用特殊图标来表示它们。

图标	代表
	State Group
	State

使用 State 时涉及以下任务：

- [创建 State Group](#)
- [创建 State](#)
- [为 State Group 内的 State 之间的切换设置过渡](#)

- [删除 State/State Group](#)

创建 State Group

在 Wwise 中，您可以在两个位置创建所有需要的 State Group（状态组）：

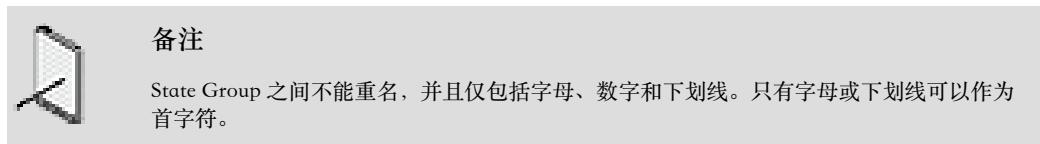
- [Project Explorer（工程浏览器）](#)
- [Property Editor（属性编辑器）的 States（状态）选项卡](#)

在 Project Explorer 中为工程创建新 State Group 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，执行以下操作之一：
 - 选择 Work Unit 或 Virtual Folder，并点击 Project Explorer 工具栏中的 State Group 图标。
 - 右键点击 Work Unit 或 Virtual Folder，并在快捷菜单中选择 <New Child> State Group。

新 State Group 将添加到列表中。

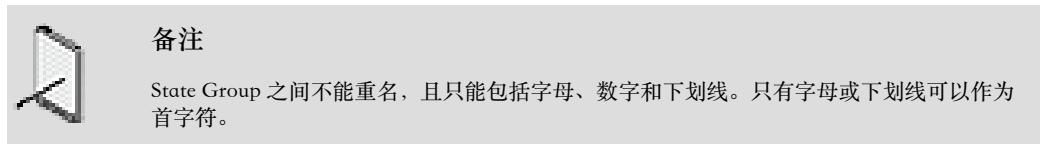
3. 将默认名称替换为更合适的名称。



4. 继续按需添加 State Group。

在 Property Editor 的 States 选项卡中为工程创建新 State Group 的方法如下：

1. 在 Property Editor 的 States 选项卡中，点击 Add >>。
2. State 选择器菜单将打开。
3. 选择 New...。
4. 在 Name 字段中，将默认名称替换成更合适的名称。



5. 点击 OK 即可创建并采用新 State Group。

相关主题

- [创建 State](#)
- [为 State Group 内的 State 之间的切换设置过渡](#)

- [删除 State/State Group](#)

创建 State

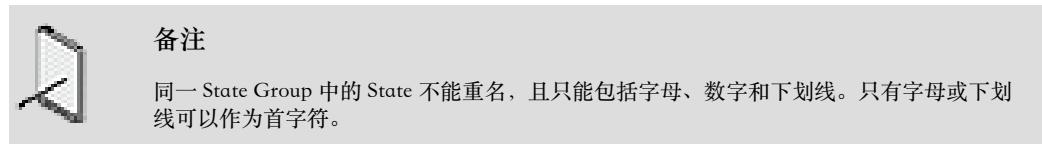
游戏调用 State（状态）时，就能据游戏条件相应地改变对象属性。State 间的过渡可以在 State Group Editor（状态组编辑器）中设置。

创建新 State 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，执行以下操作之一：
 - 选择 State Group，并点击 Project Explorer 工具栏中的 State 图标。
 - 右键点击 State Group，并在快捷菜单中选择 New Child > State。

新 State 将被添加到 State Group 中。

3. 将默认名称替换为更合适的命名。



4. 继续按需创建 State。

相关主题

- [创建 State Group](#)
- [为 State Group 内的 State 之间的切换设置过渡](#)
- [删除 State/State Group](#)

删除 State/State Group

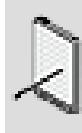
您可能想要删除工程中不再需要的 State（状态）或 State Group（状态组）。记住，删除 State Group 也会删除其中所有的 State。State 删除后将不能再用于之前的对象和预设。



删除 State Group 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，右键点击要删除的 State Group 或 State，并选择 Delete Selection。

选定的 State 或 State Group 将被删除。



备注

如果您误删了 State 或 State Group，可以按 Ctrl+Z 或点击 Edit > Undo 来撤消删除。

相关主题

- [创建 State Group](#)
- [创建 State](#)

为 State Group 内的 State 之间的切换设置过渡

为了同一 State Group（状态组）内各 State（状态）之间能平滑过渡，您可以定义状态过渡的时长。设置过渡时长时，有两个选项：

- [为 State Group 中的所有 State 设置过渡](#)为 State Group 中所有 State 切换设置相同的过渡时长。
- [自定义 State Group 内 State 之间的切换过渡](#)为 State Group 内的 State 切换设置不同的过渡时长。

为 State Group 中的所有 State 设置过渡

您可以为 State Group（状态组）内的所有 State（状态）切换设置统一的过渡时长。

为选定 State Group 中所有 State 切换设置过渡时长的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，双击 State Group 将其加载到 Property Editor 中。
3. 在 Default transition time 中，为该 State Group 中所有 State 切换设置默认过渡时长。



备注

如果未定义 Custom Transition Time，将使用默认过渡时间。

相关主题

- [创建 State](#)
- [删除 State/State Group](#)
- [自定义 State Group 内 State 之间的切换过渡](#)

自定义 State Group 内 State 之间的切换过渡

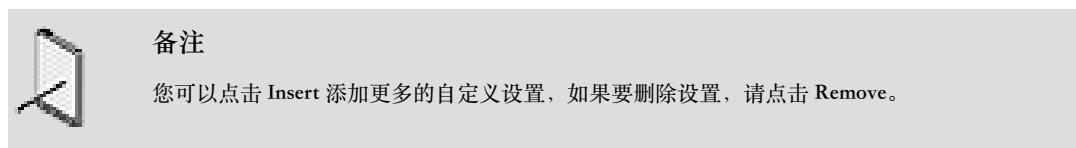
为让游戏中的 State 改变更逼真，您可能不希望 State Group 内所有 State 切换时过渡时间都相同。在 Custom Transition Time（自定义过渡时间）列表中，可以为任意两个 State 间的切换定义特殊的过渡时长，反向切换时，也可以采用相同的自定义过渡时长。例如，在 Rain State 切换至 Snow State 和反向切换时，可能希望其过渡时间相同。

为选定 State Group 内 State 之间的切换设置自定义过渡时间的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，双击 State Group 将其加载到 Property Editor 中。
3. 点击 Insert。

Transition Time 列表中将新增一行。

4. 在 From 列中，选择源 State。
5. 在 Time 列中，设置两个 State 之间的过渡时间。
6. 在 To 列中，选定目标 State。
7. 要为两个切换方向设置相同的过渡时间，请勾选双向复选框（标记为：<->）。



相关主题

- [创建 State](#)
- [删除 State/State Group](#)
- [为 State Group 中的所有 State 设置过渡](#)

将 State 指派给对象和总线

为工程创建并设置 State 后，您可以将 State Group（状态组）和 State（状态）指派给对象和总线，用来让声音、音乐和振动与游戏中的场景相匹配。然后即可有针对性地调节各场景的声音、音乐或振动属性，以进一步区分。

对象须先采用 State Group，然后其中的所有 State 将自动指派到该对象。每个对象都可以采用多个 State Group。您还可以在 Master-Mixer（主混音器层级）、Actor-Mixer（混音器层级）或 Interactive Music Hierarchy（交互音乐层级）指定 State Group。

将 State Group 指派到对象的方法如下：

1. 将对象加载到 Property Editor 中。
2. 切换到 States 选项卡。
3. 点击 Group Selector (>>) 按钮，并选择要指派给对象的 State Group。

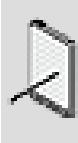
相关 State 将出现在列表中，对象将采用选定 State Group 中的 State。

相关主题

- [自定义对象的 State 属性](#)
- [为音乐对象设置 State 切换点](#)

为音乐对象设置 State 切换点

在互动音乐中，为 State（状态）指定精确的切换时间点非常有用，这样就可以按照正在播放的音乐节奏，确保过渡平滑。在音乐对象属性级别和总线级别都可以设置切换时间点。在音乐对象或总线的 Property Editor（属性编辑器）中，您可以为每个 State Group（状态组）设置最优的切换时间点。时间点可以是 Immediate（立即）、Next Cue（下一个提示点）或 Next Beat（下一拍）等。不过，当 State 切换涉及多个音乐对象且切换设置各不相同时，State 切换将发生在段落中遇到的下一个时间点。



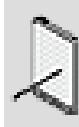
备注

如果音频总线只包含 Actor-Mixer Hierarchy（混音器层级）的声音对象，时间点设置将被忽略，状态会立即切换。但如果音频总线同时包含音乐和声音对象，State 切换将基于音乐对象的时间点设置。

为音乐对象和总线定义 State 切换点的方法如下：

1. 将音乐对象或总线加载到 Property Editor。
2. 切换到 States 选项卡。
3. 对于采用的 State Group，在 Change occurs at 列中选择下列之一：
 - Immediate —— 立即切换状态。如果音轨设置了 Look-ahead time（预读时间），那么经过预读时间之后 State 才会切换。
 - Next Grid —— 在下一网格线处切换状态。网格线允许您按照任意间隔长度来分割音乐对象。
 - Next Bar —— 在下一小节切换状态。
 - Next Beat —— 在下一拍切换状态。
 - Next Cue —— 在下一提示点处切换状态。下一提示点可以是 Entry cue（入口提示点）、Exit cue（出口提示点），也可以是 Custom Cue（自定义提示点）。
 - Next Custom Cue —— 在下一自定义提示点处切换状态。
 - Entry Cue —— 在 Entry cue 处切换状态。
 - Exit Cue —— 在 Exit cue 处切换状态。

当前 State Group 中的所有 State 切换都将发生在指定时间点。



备注

如果多个音乐对象采用了同一个 State Group，State 切换将对所有对象生效。在遇到的下一个时间点，会为所有对象切换 State。

相关主题

- [自定义对象的 State 属性](#)

自定义对象的 State 属性

将 State Group 指派给对象后，您可以自定义该对象在各 State 下的属性。对于每个 State，可以修改以下属性值：

- Pitch (音高)
- Low-Pass Filter (低通滤波器)
- High-Pass Filter (高通滤波器)
- Volume (音量)
- Make-up Gain (补偿增益)



备注

对于有些对象和平台，只会用到部分属性。例如，Pitch 属性不影响音乐对象，Make-up Gain 不影响 Master-Mixer 对象。

为对象自定义 State 属性值的方法如下：

1. 将对象加载到 Property Editor 中。
2. 切换到 States 选项卡。
3. 设置以下属性值：
 - Pitch —— 将加快或放慢对象的播放速度。
 - Low-Pass Filter —— 将采用衰减高频的递归滤波器。
 - High-Pass Filter —— 将采用衰减低频的递归滤波器。
 - Volume —— 将调整输出电平或振幅。

相关主题

- [将 State 指派给对象和总线](#)
- [在 State 之间复制属性值](#)
- [示例：制造暂时性听力损失效果](#)

在 State 之间复制属性值

如果多个 State (状态) 要使用相似的属性值设置，可以一次性设置好，然后复制到同一 State Group (状态组) 中的其它 State。

您可以将 State 属性值复制到同一 State Group 中的现有或新增 State，甚至可以只让部分对象受到源 State 属性值的影响。

复制 State 属性值的方法如下：

1. 在以下视图中点击 Copy State Values... 按钮：
 - State Property Editor。
 - Mixing Desk。

Copy State Values 对话框将打开。

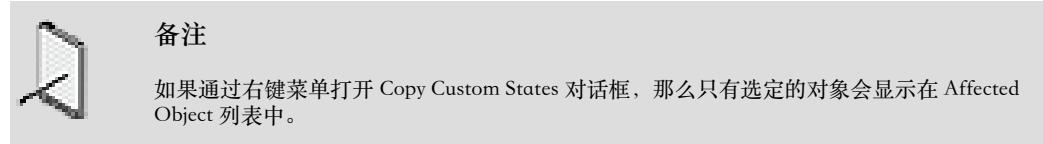


技巧

也可以右键点击某个 State，然后在 Property Editor 的 States 选项卡或 Mixing Desk 中选择 Copy State values...。

2. 点击 **State Group** 选择器 (>>)，并选择要复制自定义属性值的 State 所在的 State Group。
3. 点击 **From** 选择器 (>>)，并选择要复制自定义属性值的 State。

采用该 State Group 的所有对象都将显示在 Affected Object 列表中。



4. 点击 **To** 选择器 (>>)，并执行下列操作之一：
 - 要将自定义属性值复制到新 State，请选择 New，为新 State 命名并点击 OK。
 - 要将自定义属性值复制到现有 State，则只需从列表中选择该 State 即可。

Wwise 将分别处理 Affected Object 列表中的对象。

5. 在 Use 列中，勾选要使用新自定义 State 设置的对象。
6. 点击 OK 为选定的对象应用自定义 State 设置。

相关主题

- [将 State 指派给对象和总线](#)
- [自定义对象的 State 属性](#)
- [示例：制造暂时性听力损失效果](#)

示例：制造暂时性听力损失效果

假设您正在研发一款第一人称射击游戏。在游戏中，一颗闪爆弹在附近爆炸，让玩家出现了暂时性听力损失。为了模拟这种效果，需要在短时间内让所有声音受到影响。

步骤

使用 State 是实现“暂时性听力损失”的最佳方式。例如，您可以创建名为“GrenadeFX”的 State Group，其中包括两种 State：“Stunned”（振晕）和“Normal”（正常）。对于“震晕”State，可以修改 Pitch、Volume、Low-pass Filter 和 High-pass Filter 属性值来制造“暂时性听力损失”效果。然后可以在多个控制总线上用这个 State Group 来影响音效、环境声、音乐等。

要增强真实感，可以在采用“GrenadeFX”State Group 的控制总线上同时用 RTPC，来影响距离效果的强度。State 属性值（音量、音高、HPF 和 LPF）将与 RTPC 叠加，这样即可根据手榴弹与主角/话筒之间的距离来减小或增大属性值。要实现这样的效果，可以执行以下操作：

1. 在 Game Sync 选项卡中创建新游戏参数，例如“Grenade Proximity”，并根据游戏情况设置合适的最小和最大距离。
2. 在使用 State Group "GrenadeFX" 的控制总线上，转至 RTPC 选项卡，并为游戏参数“Grenade Proximity”指派一个或多个属性值。
3. 建立属性值映射，榴弹冲击离角色越近，影响越强烈。

在 Soundcaster 中进行游戏情景模拟时，可以直接优化 State 和 RTPC 值。例如，播放音乐、环境声、语音和音效，然后将 State Group 从“Normal”切换到“Stunned”。别忘了优化不同 State 之间的 Transition Time。

为了更真实，还可以在角色进入“震晕”State 时让混响尾音变得混浊。通过对混响的“LPF Cutoff Frequency”属性使用 RTPC 并与 State 同步更改，您可以让混响与对象的 LPF 值一致。

相关主题

- [将 State 指派给对象和总线](#)
- [自定义对象的 State 属性](#)

State 技巧和经验总结

使用 State（状态）时，最好仔细阅读以下各节，其中的一系列技巧和经验总结可以帮您更好地管理 State 在研发中的使用。

删除 State/State Group

State 可以通过两种机制集成到游戏中。一种集成机制是调用含 Set State 动作的事件，另一种是调用 State Group 和 State 本身。前一种机制下，删除 State 或 State Group 后，被调用的 State 不再可用时，在 Wwise 中会发生问题。后一种情况下，声音设计师要删除 State 或 State Group 时，应该告诉音频程序员。

使用 Find all Reference 命令可以找出工程中哪里使用了该 State。

State 属性以及 CPU 和内存占用

在 Wwise 中，有些相对属性（例如音高）可能影响不同平台上的性能。Wwise 中管理音高的机制基于采样率，调整对象音高时须对文件重新采样，因此会提高 CPU 占用。

第 18 章 使用 Switch

概述	373
使用 Switch	374
将 Game Parameter 值映射到 Switch	376
Switch 技巧和经验总结	377

概述

在使用层级结构组织对象之外，Wwise 中的 Switch（切换开关）也能帮您简化声音、音乐和振动对象的组织。Switch 代表游戏中特定元素的不同条件，可以用来管理这些条件下的相应用对象。这些元素的条件种类非常丰富，可以是天气状况，游戏主角使用的武器等。您可以将特定条件下的对象指派到特定 Switch，这样播放该游戏元素时，将播放当前 Switch 对应的对象。

Switch 可以在运行时简化各种条件的管理，对于众多游戏场景或元素都适用，下面列出少数几个：

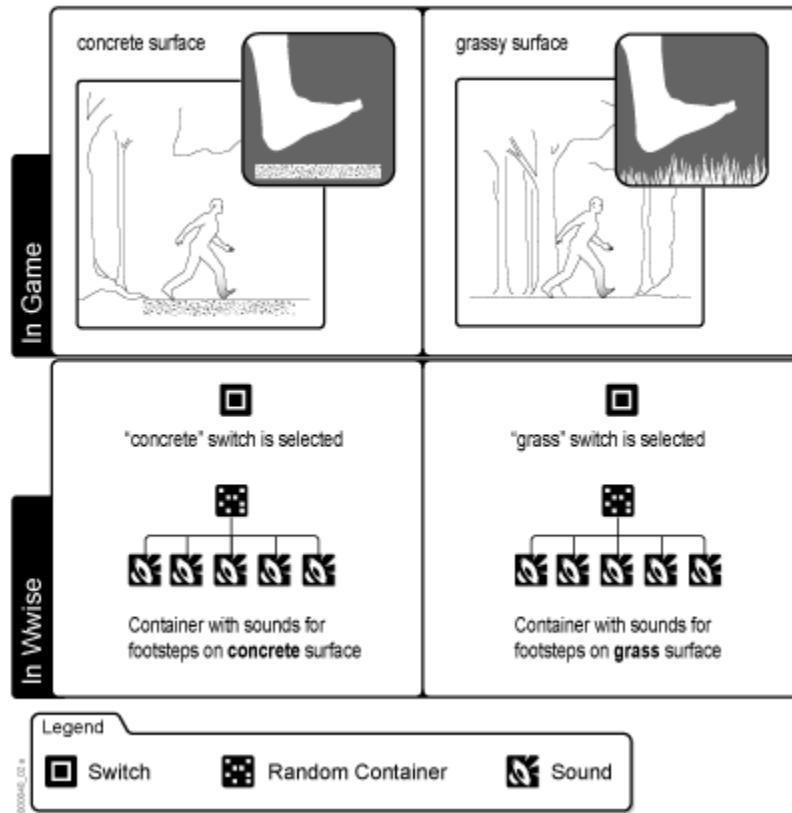
- 房间、地面、室内/室外的游戏设置——可以为不同的地面材质（例如木板、草地、碎石等）创建 Switch。
- 游戏角色——可以在男性或女性角色讲话时，为对白创建 Switch。
- 天气状态——可以为暴风雨、暴风雪、小雨或晴天创建 Switch。
- 魔界或仙界的游戏氛围——可以为不同世界的特有声音创建 Switch。
- 武器——可以为游戏中火器的不同发射方式、激光和剑创建 Switch。

上述各例中，您都可以先创建 Switch，然后指派相应的对象。指派到 Switch 的对象需要用 Switch Container（切换开关容器）进行编组。当 Event（事件）或 Game Parameter（游戏参数）值发生改变时，Switch Container 将检查 Switch，然后播放正确的对象。

使用 Switch —— 示例

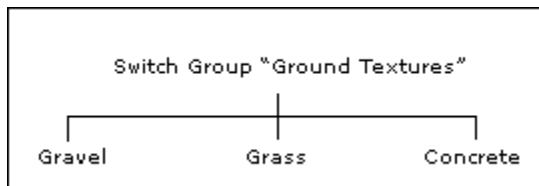
假设您正在研发一款第一人称射击游戏，其中角色可以在各种环境中走动。每种物理环境中都有不同的地面材质，例如混凝土、草地和泥地，每种表面都需要不同的脚步声。这种情况下，您可以为各地面材质创建 Switch，然后将不同脚步声指派到相应的 Switch。当角色行走在混凝土地面上，“Concrete” Switch 将激活，并播放相应声音。从混凝土走到到草地上时，“Grass” Switch 将激活，并播放相应声音。

下图演示了激活的 Switch 如何影响播脚步声的播放。



使用 Switch

在 Wwise 中，需要先将 Switch（切换开关）置于 Switch Group（切换开关组）中才能使用。通过用 Switch Group 对 Switch 进行编组，您可以高效管理游戏中不同条件下的声音、音乐和振动对象。例如，为了管理角色的各种脚步声，可以创建名为“Ground Textures”的 Switch Group。然后在 Wwise 中为游戏中每种地面材质创建对应的 Switch。根据游戏条件，您可以创建针对碎石地、草地和混凝土的 Switch。



有了 Switch Group 和 Switch，就可创建 Switch Container 并将对象指派给 Switch 了。关于使用 Switch 和 Switch Container 的详细信息，请参阅[定义 Switch Container 的内容和行为](#)。

为了帮您在界面中轻松识别 Switch 或 Switch Group，Wwise 采用独特的图标来标识它们。

图标	代表
	Switch Group

图标	代表
	Switch

使用 Switch 涉及以下任务：

- [创建 Switch Group](#)
- [创建 Switch](#)
- [删除 Switch Group/Switch](#)
- [将 Game Parameter 值映射到 Switch](#)

创建 Switch Group

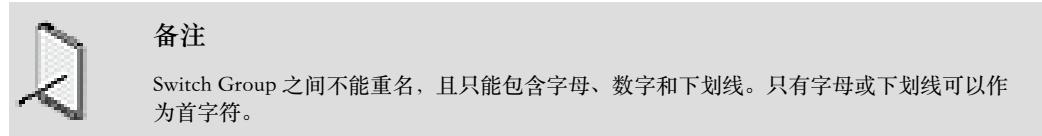
按逻辑将 Switch（切换开关）编组成 Switch Group（切换开关组）非常有用，用起来也更方便。您可以在 Project Explorer（属性编辑器）的 Game Syncs（游戏同步器）选项卡中创建所有需要的 Switch Group。

为工程创建新 Switch Group 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Switches 层级，执行以下操作之一：
 - 选择 Virtual Folder 或 Work Unit，然后点击 Project Explorer 工具栏中的 Switch Group 图标。
 - 右键点击 Virtual Folder 或 Work Unit，从快捷菜单中选择 New Child > Switch Group。

新的 Switch Group 将被添加到列表中。

3. 将默认名称替换成更合适的命名。



4. 继续按需添加 Switch Group。

相关主题

- [删除 Switch Group/Switch](#)
- [创建 Switch](#)

创建 Switch

游戏元素（例如地面）具有的不同条件在 Wwise 中要有相应的 Switch。您可以在 Project Explorer（属性编辑器）的 Game Syncs（游戏同步器）选项卡中创建 Switch。

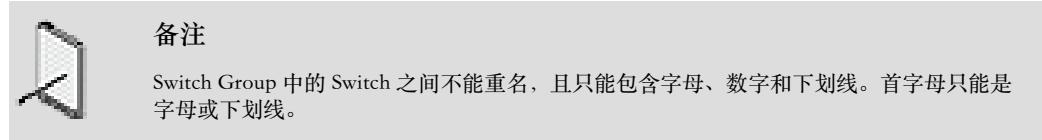
创建新 Switch 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Switch Groups 列表中，执行下列操作之一：

- 选择 Switch Group，并点击 Project Explorer 工具栏中的 Switch 图标。
- 右键点击 Switch Group，从快捷菜单选择 New Child > Switch。

新的 Switch 将被添加到 Switch Group 中。

3. 将默认名称替换成更合适的命名。



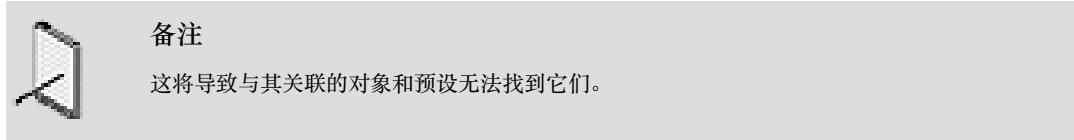
4. 继续按需创建 Switch。

相关主题

- [创建 Switch Group](#)
- [删除 Switch Group/Switch](#)

删除 Switch Group/Switch

您可能想删除不再需要的 Switch（切换开关）或 Switch Group（切换开关组）。记住，删除 Switch Group 时也会删除其中的所有 Switch，



删除 Switch 或 Switch Group 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Switch Group 列表中，右键点击要删除的 Switch Group 或 Switch，然后选择 Delete Selection。

所选 Switch 或 Switch Group 将被删除。



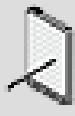
相关主题

- [创建 Switch](#)
- [创建 Switch Group](#)

将 Game Parameter 值映射到 Switch

在 Wwise 中，您还可以使用 Game Parameter（游戏参数）值来驱动 Switch 的切换。创建 Switch Group 和 Game Parameter 后，就可以将参数值映射到 Switch。例如汽车碰撞声，如果您使用 RTCP 来驱动 Switch 切换，那么撞击声和振动效果可以根据冲

击力大小而有所不同。通过用冲击力值来触发 Switch 切换，可以轻松确保在发生碰撞时播放正确的对象。



备注

将 Game Parameter 值映射到 Switch 之前，需要创建并定义游戏参数。关于创建和定义游戏参数的详细信息，请参阅[创建 Game Parameter](#)。

将 Game Parameter 值映射到 Switch 的方法如下：

1. 在 Project Explorer 的 Game Syncs 选项卡中，双击想要通过游戏参数值来驱动的 Switch Group。

Switch Group 将加载到 Switch Group Property Editor 中。

2. 勾选 Use Game Parameter 复选框。

坐标图视图将启用，Y 轴将显示 Switch Group 中的 Switch 列表。

3. 在 Game Parameter 列表中，选择用于驱动 Switch 切换的游戏参数。

X 轴将显示所选 Game Parameter 的值域。

4. 在坐标图视图中，双击 X 轴方向的 Game Parameter 曲线来添加点，然后拖拽该点至目标 Switch。

Switch 的变化将映射到指定的游戏参数值。

5. 双击曲线来继续添加点，并映射到所需的 Switch。

相关主题

- [创建 Switch Group](#)
- [创建 Switch](#)
- [删除 Switch Group/Switch](#)
- [第 19 章 使用 RTPC](#)

Switch 技巧和经验总结

使用 Switch（切换开关）时最好能仔细阅读以下各节，其中的一系列技巧和经验总结可以帮助您在整个音频开发过程中更好地管理 Switch。

重命名 Switch

更改 Switch 名称前，请检查 Switch 是如何集成到游戏中的。如果是用名称字符串集成的，则更名后将需要额外的编程才能使用。

删除 Switch / Switch Group

删除 Switch 或 Switch Group 前，请检查 Switch 是如何集成到游戏中的。Switch 的集成方式有两种机制：

- 通过调用带有 Set Switch 动作的事件——删除 Switch 或 Switch Group 后，在 Wwise 中会造成问题，因为事件调用的 Switch 将不存在。

- 通过调用 Switch Group 或 Switch 本身——要删除 Switch 或 Switch Group 时，声音设计师应告知程序员。

为对象指定多个 Switch Group

通常，对象仅可指定到一个 Switch Group。但如果游戏要求将对象指定到不同 Switch Group 中的 Switch，您可以在 Master-Mixer、Actor-Mixer 或 Interactive Music 层级结构中的更高层级指定第二个 Switch Group。

第 19 章 使用 RTPC

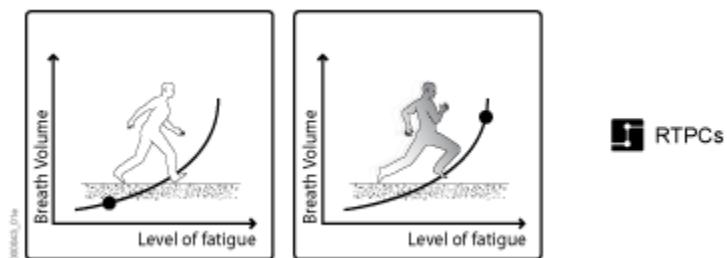
概述	380
管理 RTPC 中使用的 Game Parameter	382
使用 Game Parameter 控制属性值	386
使用 LFO	393
使用包络	394
使用时间	397
查看 Game Object	398
使用旁链压缩	398
RTPC 技巧与最佳实践	404

概述

为了在游戏中实现更多动态效果，您可能需要将特定的对象属性与游戏中的某些参数值绑定。在 Wwise 中，您可以使用实时参数控制（RTPC）来实现这一点。您可以使用曲线沿线上的一系列点创建 RTPC。该曲线表示了 Game Parameter（游戏参数）和 Wwise 中音频属性之间的关系。当游戏中的 Game Parameter 发生变化时，Wwise 使用 RTPC 曲线来确定相应的属性值。

使用 RTPC —— 示例

假设您正在创建一款第一人称射击游戏，您希望根据游戏中角色的疲惫级别来确定主人公呼吸声的音量。当角色不太疲惫时，您希望呼吸声听起来非常柔和；非常疲惫时，呼吸声听起来很沉重。这种情况下，您可以使用 RTPC 将 Game Parameter（疲惫程度）指派给 Wwise 属性（音量）。然后您可以使用坐标图视图，将呼吸声音的音量电平映射到游戏中主人公的疲惫级别。



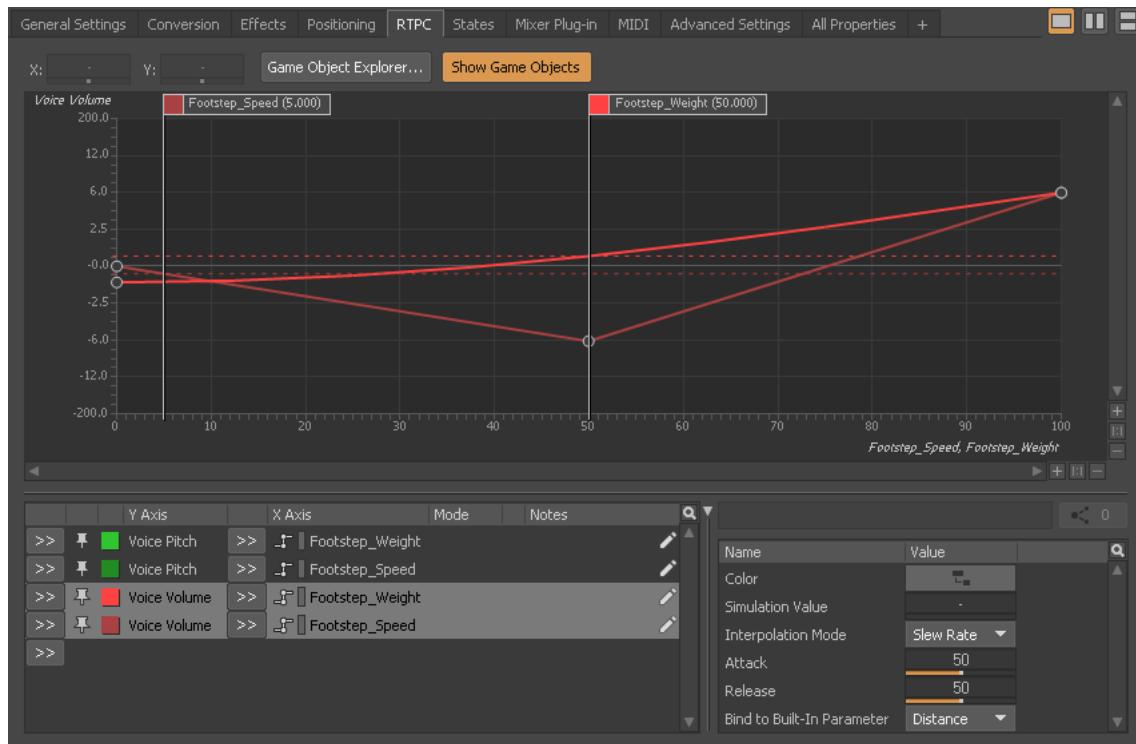
RTPC 还可用于获取游戏中的其它效果，例如将水深映射到低通滤波值，将爆炸力映射到低频效果值（LFE）等等。

理解 Wwise 中的 RTPC

在 Wwise 创建 RTPC 涉及以下操作：

- [创建 Game Parameter、MIDI Parameter、LFO、Envelope 或 Time Modulator](#)。
- [将 Wwise 属性指派给游戏参数](#)
- [映射 RTPC 坐标图中的值](#)

您可以在 Project Explorer（工程资源管理器）的 Game Syncs（游戏同步器）选项卡中或者 RTPC 选项卡内定义 Game Parameter。属性值和 Game Parameter 之间的关系可以在 Property Editor、Effect Editor 或 Attenuation Editor 中的 RTPC 选项卡中定义。另外，还可直接在 RTPC 选项卡的侧面板内调节所选参数的属性。



通过将 X 轴所示的 Game Parameter 映射到 Y 轴所示的 Wwise 属性值，即可让 Game Parameter 按照期望的方式影响对象属性。引擎创建完整 RTPC 曲线的方法是在您创建的控制点之间插入新的控制点。

每个对象、总线、Attenuation（衰减）或 Effect（效果器）实例都可具有多条曲线，用来表示对象属性与参数或调制器之间的不同关系。用于比较时一次可显示多条曲线，也可以一次只显示一条曲线。

使用 RTPC 侧面板

无论是 Game Parameter、MIDI Parameter（MIDI 参数）还是 Wwise 中的三种调制器，RTPC 的 X Axis（X 轴）对象都可设置多种属性值。而且，每个对象都可在对应的 Property Editor（属性编辑器）中单独打开，以便检查并根据需要调节各项属性。不过，一般直接在 RTPC 选项卡内调节属性值会更为便捷。侧面板会始终显示所选 RTPC X Axis 对象的属性，这些属性通常都可直接进行编辑。

若选择多个 RTPC，则侧面板将合并显示其属性及数值。若多个 RTPC 共用同一属性，但其值各不相同，则将显示破折号。



如何查找引用 RTPC 的对象

主要可以通过两种方式直接从 RTPC 选项卡来查找引用 RTPC 的对象。

- 在 RTPC 快捷菜单中选择 Find All References 命令，以此打开 References 视图并查找使用该 RTPC 的对象。
- 直接单击侧面板右上角带有编号的 References 按钮。此操作同样可以打开加载有对象的 References 视图。



在视图中加载所有 RTPC 信息

RTPC 选项卡中有两条分割线，可通过拖动来调节三个面板的大小。

- 横线：将所 RTPC 列表与坐标图隔开。
- 纵线：将所列 RTPC 与侧面板隔开（后者列有关联属性）。

在 Blend Track 内使用 RTPC

要在对象属性和游戏参数之间创建更加复杂的关系，您可在混合容器的混合轨中使用 RTPC。有关在混合容器中使用 RTPC 的详细信息，请参阅[定义 Blend Container 的内容和行为](#)。



注意

虽然可为工程中的所有对象、总线、衰减和效果创建 RTPC，但选择性地使用它们非常重要，因为它们会消耗大量的内存和 CPU 资源。

管理 RTPC 中使用的 Game Parameter

将游戏中的参数映射到 Wwise 属性之前，您需要先创建 Game Parameter。您可以管理多个游戏参数，并在 Project Explorer 中 Game Syncs 选项卡中定义其最小值和最大值。

为了帮助您轻松识别界面中的 Game Parameter，它由以下图标表示。

图标	代表
	Game Parameter (游戏参数)

管理 Game Parameter 可能涉及以下任务：

- [创建 Game Parameter](#)
- [定义游戏参数的取值区间](#)
- [删除游戏参数](#)
- [将游戏参数绑定到内置参数](#)

创建 Game Parameter

若要使用 Game Parameter（游戏参数）来驱动对象的属性值，必须先创建游戏参数。您可以在 Wwise 中以下两个位置来创建所需的 Game Parameter：

- Project Explorer 的 Game Sync 选项卡
- Property、Attenuation 或 Effect Editor 的 RTPC 选项卡



注意

在 Wwise 中命名 Game Parameter 时，只能使用字母、数字和下划线，并确保每个 Game Parameter 名称都是唯一的。

在 Project Explorer 中创建新的游戏参数的方法是：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Game Parameter 部分，执行以下操作之一：
 - 选择工作单元或虚拟文件夹，然后点击 Project Explorer 工具栏中的 Game Parameter（对白事件）图标。
 - 右键点击工作单元或虚拟文件夹，然后从快捷菜单中选择 New Child > Game Parameter（新建子项 > 对白事件）。

一个新的 Game Parameter 被添加到 Game Parameter 列表中。

3. 将默认名称替换为最适合该游戏参的名称。



4. 根据需要继续添加游戏参数。

在 RTPC 选项卡中新建 Game Parameter 的方法如下：

1. 在 Property Editor、Attenuation Editor 或 Effect Editor 的 RTPC 选项卡中，选择 RTPC 列表中的一个条目。

如果还没有为该条目选择属性，则通过选择按钮选择一个。>>

2. 选择 Game Parameter 菜单项，然后选择 New。

此时将打开 New Game Parameter 对话框。

3. 选择您要在其中创建游戏参数的工作单元。
4. 在 Name 字段中，将默认名称替换成最适合游戏参数的名称。
5. 点击 OK 来创建新的 Game Parameter。

相关主题

- [定义游戏参数的取值区间](#)
- [删除游戏参数](#)
- [将游戏参数绑定到内置参数](#)

定义游戏参数的取值区间

创建 Game Parameter 后，您必须设置其最小值和最大值。以赛车为例，最小和最大速度可为 0 和 300 km/h。

您也可以指定 Game Parameter 的默认值。通过游戏参数，您可以为没有明确指定特殊值的所有参数对象设置一个全局值。在以下情况下，Wwise 中指定的 Game Parameter 默认值将被忽略：

- 游戏对象明确地指定了一个特殊值。
- 游戏程序员在 SDK 中定义了一个全局 RTPC 值。

定义游戏参数的取值区间：

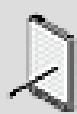
1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
 2. 在 Game Parameter 部分中，双击要定义其值的 Game Parameter。
- 此时游戏参数加载到 Property Editor 中。
3. 通过指定最小值和最大值来定义参数取值区别。
 4. 如果没有明确地指定特殊值，则在 Default 文本框中，指定一个您希望游戏对象使用的全局值。

相关主题

- [创建 Game Parameter](#)
- [删除游戏参数](#)
- [将游戏参数绑定到内置参数](#)

删除游戏参数

您可以删除工程中不再需要的 Game Parameter。删除 Game Parameter 后，对于之前使用它的对象、事件和预设，该参数将变为不可用。



备注

删除 Game Parameter 之前，请务必与您的音频程序员沟通，以确保不会影响游戏代码。

删除游戏参数：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Game Parameters 部分中，右键点击您要删除的该游戏参数，然后选择 Delete Selection（删除选中项）。

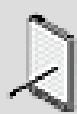
Game Parameter 将从 Wwise 中删除。

相关主题

- [创建 Game Parameter](#)
- [定义游戏参数的取值区间](#)

将游戏参数绑定到内置参数

声音引擎计算会对声音设计师有用的一组标准值来创建动态音频和振动。这些“内置”参数可通过绑定到内置参数机制来访问，并在内部每帧更新。使用绑定到内置参数的 Game Parameter 将不需要额外的编程。



备注

当 RTPC 绑定到内置参数时，声音引擎更新各个游戏对象的值。如果 RTPC 用于没有游戏对象关联的全局 Wwise 对象，例如总线或总线效果，则使用默认值。

可用的内置参数：

- **Distance**

距离。游戏对象和听者之间的距离。当多个听者或位置指定到同一个游戏对象时，将使用所有听者与声音位置组合之间的最短距离值。

- **Azimuth**

方位角。投射在水平面上的听者与游戏对象之间的角，单位为度。0 度值表示声音来自听者正前方，-90 度表示声音来自左侧，90 度表示声音来自右侧，+/- 180 度表示声音位于听者正后方。

当为游戏对象指定了多个听者或声音位置时，则使用听者和声音之间位置最靠近的角度值。

- **Elevation**

仰角。听者与游戏对象之间相对于水平线的顶角，单位为度。0 度值表示声音位于听者的同一水平面上；90 度值表示声音位于正上方；-90 度表示声音位于正下方。

当为游戏对象指定了多个听者或声音位置时，则使用听者和声音之间位置最靠近的角度值。

- **Emitter Cone**

Emitter Cone（发声体锥）表示发声体朝向与发声体和听者间连线所形成的向量之间的 3D 角度。0 度表示发声体直接朝向听者，180 度表示发声体完全背朝听者。

当为游戏对象指定了多个听者或声音位置时，则使用听者和声音之间位置最靠近的角度值。

- **声障 (Obstruction)**

声障。通过 Obstruction，可访问通过 [SetObjectObstructionAndOcclusion API](#) 为游戏对象设置的声障值。

如果为游戏对象指定了多个听者，则声障取值为给距离当前声音位置最近的听者指定的值。

- **Occlusion**

声笼。通过 Occlusion，可访问通过 [SetObjectObstructionAndOcclusion API](#) 为游戏对象设置的声笼值。

当为游戏对象指定了多个听者时，则声笼取值为给距离当前声音位置最近的听者指定的值。

- **Listener Cone**

Listener Cone（听者锥）表示听者朝向与发声体和听者间连线形成的向量之间的 3D 夹角。0 度表示听者直接朝向发声体，180 度表示听者完全背朝发声体。

当为游戏对象指定了多个听者或声音位置时，则使用听者和声音之间位置最靠近的角度值。

- **Diffraction**

当使用房间和门户表现声音传播时，可以通过 Diffraction（衍射角度）访问由 Wwise Spatial Audio 计算的衍射角度。

为了接收这个内置参数，游戏对象必须已经在 Spatial Audio 中注册，并且发声体和听者必须各自位于独立的房间中，由一个或多个门户连接。

发声体游戏对象可以接收与其“干声”相关的衍射值：一个以度为单位的偏离角，表示传播路径与发声体-听者连线的偏差角度。由 Spatial Audio 内部注册的房间游戏对象也会接收到一个衍射值，但与其“湿声”衍射相关——即房间内声音扩散后，声场的衍射保留区。湿声衍射角度是传播路径与从门户开口法线之间的发散角，以度为单位。

存在多个门户的情况下，声音可以通过多条路径抵达听者，这种情况下，衍射角度将采用各路径中的最小衍射角。

相关主题

- [创建 Game Parameter](#)
- [定义游戏参数的取值区间](#)
- [删除游戏参数](#)

使用 Game Parameter 控制属性值

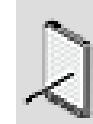
创建 Game Parameter 后，即可开始为工程中的对象、总线、效果器、衰减和切换开关组创建 RTPC。RTPC 在 Property、Effect 和 Attenuation Editor 的 RTPC 选项卡上或者混合容器的混合轨中定义。创建 RTPC 涉及以下步骤：

- [将 Wwise 属性指派给游戏参数](#)
- [映射 RTPC 坐标图中的值](#)

在曲线视图中，您可以创建将属性值映射到 Game Parameter 值的 RTPC 曲线。由于在坐标图视图中可以同时显示许多曲线，因此 Wwise 对每条曲线使用不同的颜色。三个相对属性（Volume、Pitch 和 LPF/HPF）始终采用相同基色的色调表示，而工程中的其他属性则可随意地指派颜色。这些随意指定的颜色在不同的工作会话之间可能不相同。

下表显示了都有哪些基色被指派给了相对属性。

Wwise 属性	Color
Volume	 (红色)
Pitch	 (绿色)
LPF/HPF	 (蓝色)



备注

对于有些对象和平台，只有部分属性可用；例如，音高属性将无法影响音乐对象。

即使配有颜色编码，您也会发现隐藏特定曲线有助于集中关注一条曲线。有关隐藏曲线的详细信息，请参阅[在坐标图视图中显示曲线](#)。

在工程中使用 RTPC 前，了解 RTPC 值如何与现有属性值互动非常重要。在将 RTPC 作用于现有属性值时，最终的属性值采用以下两种方式中的一种确定：

- **Absolute**: 将使用 RTPC 确定的值，忽略对象现有的属性值。
- **Relative**: RTPC 确定的值将与对象的现有属性值相加。

用于确定最终属性值的方法取决于 RTPC 属性属于绝对属性还是相对属性。如果是绝对属性，则原始属性控制会被禁用。如果是相对属性，则原始属性控制仍保持可用。



将 Wwise 属性指派给游戏参数

创建 Game Parameter 并定义它们的取值范围后，可以考虑将哪些游戏参数指派给哪些属性。

在将属性指派给 Game Parameter、Modulator 或 MIDI 对象后，将发生以下情况：

- 坐标图视图中显示 X 轴和 Y 轴。
- 坐标图视图中创建默认 RTPC 曲线。
- 属性值旁边的 RTPC 图标变为主题指定颜色（Classic 中为蓝色，Dark 中为橙色）。



将对象属性指派给游戏参数的方法是：

1. 将对象、总线、Effect（效果器）或 Attenuation（衰减）实例加载到编辑器中。
2. 切换到 RTPC 选项卡。
3. 点击选择器按钮 (>>)，然后从列表中选择属性。

新的 RTPC 曲线于是被创建，并被指定唯一的颜色。此时 Wwise 属性也被指派给坐标图视图中的 Y 轴。

4. 从 X Axis (X 轴) 列表中选择您要指派给 Wwise 属性的游戏参数。

游戏参数被指派给图形视图中的 X 轴。

快速添加 RTPC

您可以将 RTPC 添加到任何具有活跃 RTPC 图标的属性。

快速添加 RTPC:

1. 找到具有活跃 RTPC 图标的属性，并右键单击文本框、滑杆或旁边的[标志图标](#)。

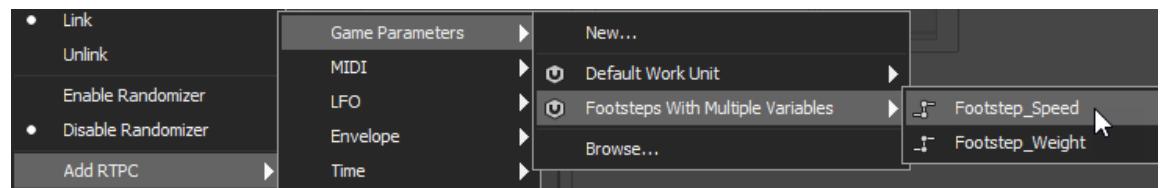
此时将会显示快捷菜单。

2. 在菜单中，选择 Add RTPC（添加 RTPC）。

此时将打开子菜单并显示五个不同的 RTPC 对象选项（Game Parameters、MIDI、LFO、Envelope 和 Time）。

3. 选择符合 RTPC 需要的选项。

此时将显示下一级子菜单。在此可选择新建 RTPC 对象、选择现有对象或浏览并查找特定对象。



4. 选择或创建 RTPC 对象。

此时将转到 RTPC 选项卡，并在视图中高亮显示刚刚添加的 RTPC 对象。您可以根据需要进行编辑。

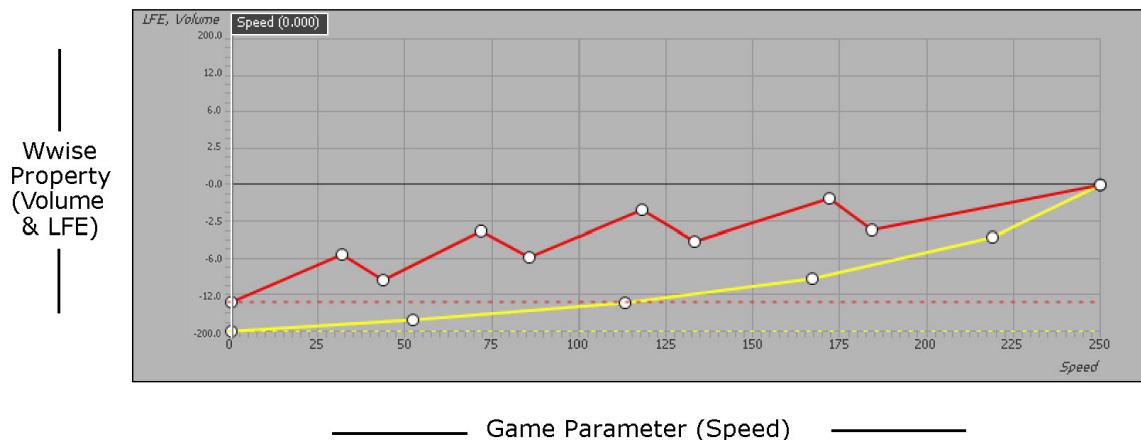
相关主题

- [使用 Game Parameter 控制属性值](#)
- [映射 RTPC 坐标图中的值](#)
- [从列表中移除 RTPC](#)

映射 RTPC 坐标图中的值

在将 Wwise 属性指派给游戏参数后，便可以开始在坐标图视图中定义它们的关系。您可以使用控制点将属性值映射到游戏参数值。例如，如果您希望汽车以 250 km/h 行驶时音量为 0 dB，则可以在 250 km/h 和 0 dB 的交点处添加一个控制点。创建 RTPC 曲线的方法是在两个控制点之间插值。

下面的 RTPC 坐标图显示如何将游戏参数值（在本例中为“Speed”）映射到 Wwise 中的属性值（在本例中为“Volume”）。X 轴上的 Speed 值在游戏中定义，Y 轴上的 Volume 值由用户控制。



有关缩放和平移坐标图视图，添加、移动和删除控制点，更改点间曲线的形状，使用线性和分贝标度的信息，以及有关坐标图视图的通用信息，请参阅[第 43 章 了解坐标图视图](#)。



备注

您可以复制工程中对象内或对象之间的 RTPC 曲线。有关 RTPC 的详细信息，请参阅[复制 RTPC 曲线](#)。

映射坐标图中值的方法是：

1. 在坐标图视图中，双击曲线上任意位置以创建控制点。
2. 执行以下操作之一：
 - 将控制点拖到相应的 X 和 Y 坐标。
 - 直接在 X 和 Y 坐标文本框中输入值，以指定控制点的准确位置。
3. 继续添加点，直至曲线完全代表了属性和游戏参数之间的关系。



技巧

如果创建控制点后继续按住鼠标键，则可以立即将点拖到适当的位置。

相关主题

- [添加控制点](#)
- [选择控制点](#)
- [移动控制点](#)
- [指定控制点之间曲线的形状](#)
- [使用 Game Parameter 控制属性值](#)
- [将 Wwise 属性指派给游戏参数](#)
- [复制 RTPC 曲线](#)
- [从列表中移除 RTPC](#)

创建智能音高曲线

当您使用 RTPC 将声音的音高与 Game Parameter 相匹配时，复杂的曲线是创造逼真效果的关键。通过智能音高曲线，您可以轻松地创建自然声音的 RTPC 音高曲线。

智能音高曲线基于以下两个变量：

- **Native value:** 原始音高对应的 Game Parameter 值。
- **Subdivision level :** 曲线的精度，从 1 到 10 分为十个等级。

例如，在赛车游戏中，可以将智能音高曲线应用于引擎声音。如果录制了发动机转速为 2000 RPM 时的发动机噪声，则可以采用原声值为 2000 RPM 的方式创建 RTPC 音高曲线。然后 Wwise 据此推断出其它点，创建出自然声音的曲线。

在使用智能音高曲线创建音高 RTPC 曲线时，Wwise 使用下面的等式创建线性节段：

$$\text{Pitch (in cents)} = 1200 * \log_2 \left(\frac{x}{\text{reference } x} \right)$$

曲线的节段越多，曲线的精度越高。不过，在运行时，其它节段需要进一步处理。为了节省 CPU 和内存，应选择能够达到您所需效果的最低细分等级。



技巧

有关确保曲线准确性的详细信息，请参阅[智能音高曲线的有效范围](#)。

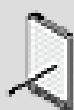
创建智能音高曲线的方法是：

1. 在 RTPC 坐标图视图中创建音高曲线。有关如何操作的详细信息，请参阅[将 Wwise 属性指派给游戏参数](#)。
2. 右键点击音高曲线，然后从菜单中选择 Build Smart Pitch Curve。

此时将打开 Build Smart Pitch Curve 对话框。

3. 输入智能音高曲线的原声值。原声值是指录制声音所对应的属性值。
4. 输入智能音高曲线的细分等级。曲线细分等级越高，曲线节段越多，运行时评估时间越长。
5. 点击 OK。

此时将创建智能音商曲线。



备注

只可为 4800 和 -4800 音分之间的音高值创建智能音高曲线。在此范围之外，曲线将在一个极端值处“锁定”。

相关主题

- 添加控制点
- 选择控制点
- 移动控制点
- 指定控制点之间曲线的形状
- 使用 Game Parameter 控制属性值
- 将 Wwise 属性指派给游戏参数
- 映射 RTPC 坐标图中的值
- 复制 RTPC 曲线
- 从列表中移除 RTPC

复制 RTPC 曲线

在工程中，有时候您会希望跨属性甚至跨对象使用 RTPC 曲线。与其分别创建各条曲线，您可以先定义曲线形状，然后将它复制粘贴到同一对象的另一属性中，甚至工程中其它对象的属性中。

RTPC 曲线可以复制到 Wwise 的以下区域：

- Master-Mixer、Actor-Mixer 和 Interactive Music 层级结构中的对象和总线的 RTPC 选项卡。
- 效果的 RTPC 选项卡。
- 源插件的 RTPC 选项卡。
- Blend Track Editor。

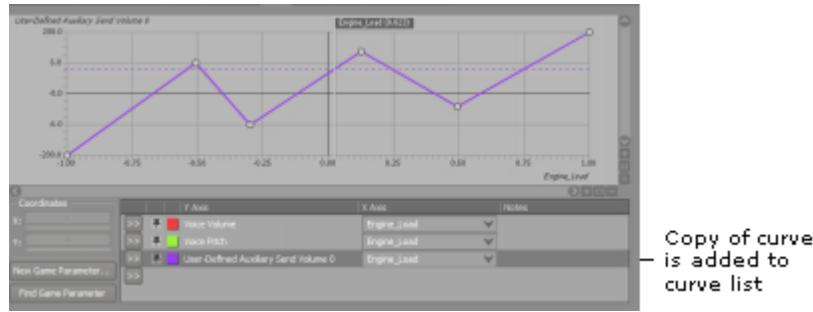
复制对象内和对象之间的 RTPC 曲线的方法是：

- 从曲线列表中选择您要复制的曲线。



- 执行以下操作之一：
 - 右键点击选定的曲线，然后从菜单中选择 Copy。
 - 按 Ctrl+C。
- 执行以下操作之一：
 - 右键点击并从菜单中选择 Paste。
 - 按 Ctrl+V。

此时列表中被添加了一条完全一致的 RTPC 曲线。

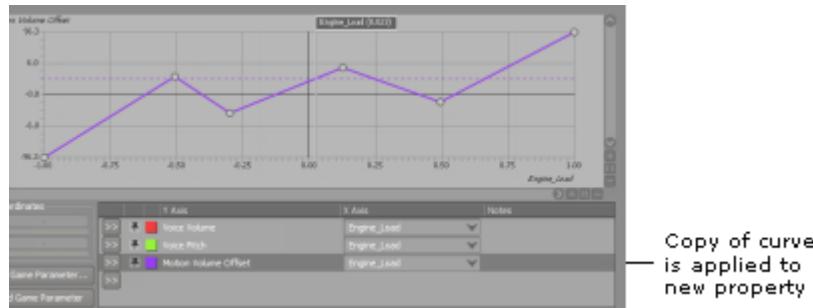


技巧

要将若干条曲线粘贴到工程中的另一对象中，只需加载新对象到 Property Editor，切换到 RTPC 选项卡，然后继续执行步骤 3。

4. 点击选择器开关 (>>)，并从列表中选择任意属性。

此时选定的属性就应用到曲线形状上了。



相关主题

- 添加控制点
- 选择控制点
- 移动控制点
- 指定控制点之间曲线的形状
- 使用 Game Parameter 控制属性值
- 将 Wwise 属性指派给游戏参数
- 映射 RTPC 坐标图中的值
- 从列表中移除 RTPC

从列表中移除 RTPC

当您不再需要 Game Parameter 来驱动 Wwise 中某个特定属性的值时，可以将其从 RTPC 列表中删除。

从列表中删除 RTPC 的方法是：

1. 从 RTPC 列表中选择您要删除 RTPC。

2. 按 Delete 键。

RTPC 于是从列表中被移除了。

相关主题

- [使用 Game Parameter 控制属性值](#)
- [将 Wwise 属性指派给游戏参数](#)
- [映射 RTPC 坐标图中的值](#)

使用 LFO

LFO（低频振荡器）用于为属性值带来随时间变化的调制。LFO 的属性如下：

界面元素	描述
Depth	深度。振荡器的幅度变化。最大幅度为 1.0。 单位: % Default value: 100 Range: 0 to 100
Frequency	频率。每秒钟的周期数。 单位: Hz Default value: 1 Range: 0 to 20000 Units: Frequency
Waveform	调制器的波形包含以下选项： <ul style="list-style-type: none"> Sine Triangle Square Saw up Saw down Random:Random（随机）：选择 Random 将在调制器每次运行时随机应用电平。 Default value: Sine
Smoothing	平滑。对波形进行低通滤波，从而平滑尖锐的边缘。 单位: % Default value: 0 Range: 0 to 100
PWM	脉冲宽度调制。脉冲波的宽度；仅用于Square（方波）波形。 单位: % Default value: 50 Range: 0 to 100
Attack	起音。振荡器达到满幅度所用的时间。 单位: s Default value: 0 Range: 0 to 100000
Initial Phase Offset	初相。振荡器波形的初始相位。 单位: °

界面元素	描述
	Default value: 0 Range: -180 to 180
Scope	<p>作用域。定义如何创建 LFO 实例：</p> <ul style="list-style-type: none"> • Voice: 声部。为每个声音/对象播放创建的 LFO 实例。 • Note/Event: 音符/事件。为每个播放实例或在 MIDI 环境中使用时的音符创建一个 LFO 实例。 • Game Object: 游戏对象。为每个游戏对象创建一个 LFO 实例。 • Global: 全局。为整个工程创建单个 LFO。 <p>Default value: Note or Event</p>

在 Wwise 中，有些属性是可加的（例如 Voice Volume、Voice Pitch），有些是不可加的。当把一个LFO 添加到加性属性上时，LFO 产生的调制将被叠加到该属性的当前值上。在对不可加属性添加 LFO 时，LFO 调制值代替属性的当前值。

使用 LFO 调制 Voice Volume 的方法是：

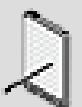
1. 在 Project Explorer 中，选择要添加 LFO 的对象。
2. 在 Property Editor 中，转至 RTPC 选项卡。
3. 在 RTPC 列表中，点击 [>>] 按钮。
4. 从选择器菜单中，选择 Voice Volume。
5. 点击 X 轴选择器按钮。
6. 从选择器菜单中选择 LFO > Default (Custom)。
7. 点击 [...] 按钮以编辑 LFO 属性。
8. 编辑曲线以设置调制范围。

LFO 对象可以被创建成 Custom 或 ShareSet。Custom 对象是原地保存的，即直接保存在拥有该对象的对象内部。ShareSet 被保存在一个单独的工作单元中，并且可以在多个对象之间进行重用。



注意

LFO 和 Envelope（包络）调制器的处理时间取决于其 RTPC 用法。对于大部分属性，将在每个音频控制帧对调制器进行估值。不过，对于 Voice Volume（声部音量）属性，将在每个音频采样帧对这些调制器进行估值。请有选择地使用这些调制器，因为它们会占用相当大一部分平台内存和 CPU 资源。



备注

LFO 和 Envelope 调制器的 RTPC 光标并不能指示其所在时间点的具体值，因为该值是由调制器的内部属性决定的。

使用包络

包络用于将属性值控制在预定义的形状内。包络具有 [ADSR](#) 形状：

界面元素	描述
Attack Time	<p>起音时间。定义了当琴键第一次被按下时，电平从零值上升至峰值所用的时间。</p> <p>单位： s</p> <p>Default value: 0.2 Range: 0 to 10000</p>

界面元素	描述
Attack Curve	<p>起音曲线。把起音曲线的斜率从线性的默认斜率（50%）调整到其它形式：</p> <ul style="list-style-type: none"> 指数风格的包络（0%），其变化速率在最开始较慢，随后逐渐变快 对数包络（100%），其变化速率在最开始较快，随后逐渐变慢 <p>单位： %</p> <p>Default value: 50 Range: 0 to 100</p>
Decay Time	<p>衰减时间。指随后从起音电平下降到指定延音电平所用的时间。</p> <p>单位： s</p> <p>Default value: 0.2 Range: 0 to 10000</p>
Sustain Level	<p>延音电平。指在释放按键前，声音持续期间主序列的电平。</p> <p>单位： %</p> <p>Default value: 100 Range: 0 to 100</p>
Release Time	<p>释音时间。指释放按键后电平从延音电平衰减到零值所用的时间。</p> <p>单位： s</p> <p>Default value: 0.5 Range: 0 to 10000</p>
Scope	<p>作用域。定义如何创建包络实例：</p> <ul style="list-style-type: none"> Voice:Voice（声部）：为每个声音/对象播放创建的包络实例。 Note/Event:Note/Event（音符/事件）：为每个播放实例或在 MIDI 环境中使用时的音符创建一个包络实例。 <p>Default value: Note or Event</p>
Trigger On	<p>可能触发包络（即进入起音段）的 Action/MIDI Event：</p> <ul style="list-style-type: none"> Play:Play Action 或 MIDI Note 事件 Note-Off:仅 MIDI Note-Off 事件 <p>Default value: Play</p>
Auto Release	<p>决定是否需要 Action/MIDI Event 才能让包络退出延音段，并进入释音段。如果进行了设置，则该包络将自动退出延音段：</p> <p>游戏可通过 Release Envelope 事件让包络进入释音段。</p> <p>如果包络是由 MIDI Note-On Event 触发，还可以通过 MIDI Note-Off Event 让其进入释音段。</p> <p>Default value: false</p>
Maximum Sustain Time	<p>延音时间。定义了该包络在进入释音段之前，在延音段中维持的时间。如果已设置 Auto Release，则该包络将自动退出延音段：</p> <p>单位： s</p> <p>Default value: 0 Range: 0 to 10000</p>
Stop playback after release	<p>在释音后停止播放。如果进行了设置，则被关联声音的播放将在释音段完成以后终止。</p> <p>Default value: true</p>

包络既可用在 MIDI 环境中，也可用在正常播放环境中。

用在 MIDI 环境中时，包络配置成在 Note-On 或 Note-Off（音符停止）的方式下播放声音。如果以 Note-On 的方式播放声音：

- 包络配置成在 Note-On 时触发（Trigger On 参数）。
- 包络一直持续到第一次出现以下情况：
 - 收到 Release Envelope 事件，
 - 收到 MIDI Note-Off 事件，
 - 出现延音段的最大时长（设置了 Auto Release）。

如果以 Note-Off 的方式播放声音：

- 包络配置成在 Note-Off 时触发（Trigger On 参数）。
- 包络一直持续到第一次出现以下情况：
 - 收到 Release Envelope 事件，
 - 出现延音段的最大时长（设置了 Auto Release）。

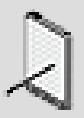
当用于播放声音的一般环境中时：

- 包络配置成遇到播放动作时触发（Trigger On 参数）。
- 包络一直持续到第一次出现以下情况：
 - 收到 Release Envelope 事件，
 - 出现延音段的最大时长（设置了 Auto Release）。



注意

LFO 和 Envelope（包络）调制器的处理时间取决于其 RTPC 用法。对于大部分属性，将在每个音频控制帧对调制器进行估值。不过，对于 Voice Volume（声部音量）属性，将在每个音频采样帧对这些调制器进行估值。请有选择地使用这些调制器，因为它们会占用相当大一部分平台内存和 CPU 资源。



备注

LFO 和 Envelope 调制器的 RTPC 光标并不能指示其所在时间点的具体值，因为该值是由调制器的内部属性决定的。

结合 MIDI 对象使用包络

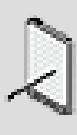
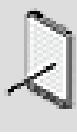
包络可用于控制属性值和声音的时长。Release 完成时，包络可以停止声音播放。

将 Voice Volume 包络添加到 Actor-Mixer Hierarchy 对象（乐器）的方法是：

1. 在 Project Explorer 中，从 Actor-Mixer Hierarchy 中选择一个对象。
2. 在 Property Editor 中，转至 RTPC 选项卡。
3. 点击 RTPC 列表中的 [>>] 按钮。
4. 从选择器菜单中，选择 Voice Volume。
5. 点击 X 轴的 [>>] 按钮
6. 在选择器菜单中，选择 Envelope > Default (Custom)
7. 点击 [...] 按钮以编辑包络
8. 编辑曲线以设置调制范围。

使用时间

时间可用来控制属性值，其方式与 [Distance](#) 和 [Azimuth](#) 内置参数相同，即使用 Time Modulator（时间调制器），其中 RTPC 的 X 轴为时间元素。

界面元素	描述
Initial Delay	<p>初始延迟。等待指定时长后，再开始基于时间的调制。</p>  <p>备注 在此延迟期间，属性值在 0 时间点处保持为 RTPC 坐标图所决定的值。</p> <p>单位: s Default value: 0 Range: 0 to 4</p>
Duration	<p>时长。在正常播放速率下，基于时间的调制进行一次迭代的时长。</p>  <p>备注 实际总时间受 Initial Delay、Playback Rate 和 Loop Count 影响。</p> <p>单位: s Default value: 1 Range: 0.1 to 100</p>
Loop Count	<p>循环次数。调制的重复次数。播放容器的次数。</p> <p>Default value: 1 Range: 0 to 100</p>
Playback Rate	<p>播放速率。调整关联声音的播放速率：</p> <ul style="list-style-type: none"> 若值等于 1，则正常播放 若值小于 1，则减速播放 若值大于 1，则加速播放 <p>Default value: 1 Range: 0.25 to 4</p>
Scope	<p>范围。定义如何创建时变调制器实例：</p> <ul style="list-style-type: none"> Voice:为每个声音/对象实例播放创建一个时间调制器实例。 Note/Event:为每个播放实例或在 MIDI 环境下使用时的音符创建一个时间调制器实例。 <p>Default value: Note or Event</p>
Trigger On	<p>触发时机。可能触发时变调制器的 Action/MIDI Event：</p> <ul style="list-style-type: none"> Play:Play Action 或 MIDI Note 事件 Note-Off:仅 MIDI Note-Off 事件 <p>Default value: Play</p>
Stop playback at end	<p>若勾选，达到总时长时（初始延迟、循环次数和播放速率都会计入）将停止播放关联声音。若未</p> <p>Default value: true</p>

从某些方面来说，Time Modulator 与 [LFO Modulator](#) 和 [Envelope Modulator](#) 存在很大差异：

- 它的曲线图可以包含多个控制点，而其他调制器最多只有两个控制点。这方便设计师在特定时间点处精确调整属性值。
- 它可以使用所有可用的[坐标图曲线](#)，而其他调制器只能使用线性和分贝标度的曲线。
- 它只包含两项可启用 RTPC 的属性：Initial Delay（初始延迟）和 Playback Rate（播放速率）。

Time Modulator 可创建为 Custom（自定义）或 ShareSet（共享集）。Custom 对象是原地保存的，即直接保存在拥有该对象的对象内部。ShareSet 存储在单独的 Work Unit（工作单元）中，并可被多个对象复用。

查看 Game Object

[游戏对象](#)（Game Object）有时候以复杂的方式与 RTPC 交互。为了跟踪这些交互，您可以将 RTPC 坐标图设置为显示您感兴趣的游戏对象。

查看 RTPC 坐标图中的游戏对象的方法是：

1. 打开 Game Object Explorer，该浏览器可通过 RTPC 选项卡视图底部面板左侧上的 Game Object Explorer... 按钮轻松打开。

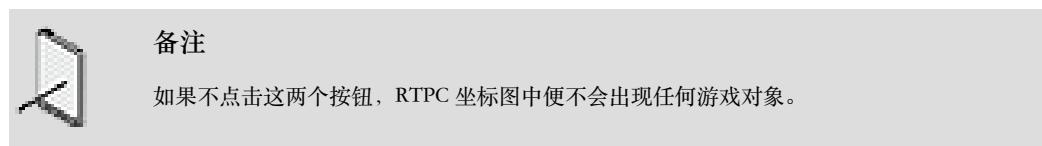
该浏览器打开后将显示音频程序员创建的所有活动游戏对象。

2. 将若干个游戏对象监视器添加到 Game Object Explorer，方法是执行[为游戏对象指派 Watch](#) 中所述的步骤。
3. 返回 RTPC 选项卡视图的底部面板，选择列表中带有相关游戏参数的若干个 RTPC。

曲线图中将显示相应的 Game Parameter 光标标识。

4. 点击底部面板中的 Show Game Object 和 Wwise 工具栏中的 Start to Capture。

此时您监视的所有游戏对象都会出现在 RTPC 坐标图中。



相关主题

- 通过 Watch 监视对象和听者
- 理解游戏对象列表

使用旁链压缩

动态混音在电子游戏中十分常见。我们经常会使用实时参数控制、混音事件和混音器快照来控制音频焦点。不过，还有一种强大的工具可用来简化游戏架构，那就是旁链压缩。

旁链压缩会监控音频信号的电平，并用来操控另一音频信号。比如，在电台广播中，DJ 说话时会自动降低音乐的音量。在音乐制作中，经常使用旁链压缩来控制低频的能量。比如，在底鼓演奏时快速降低贝斯的音量。

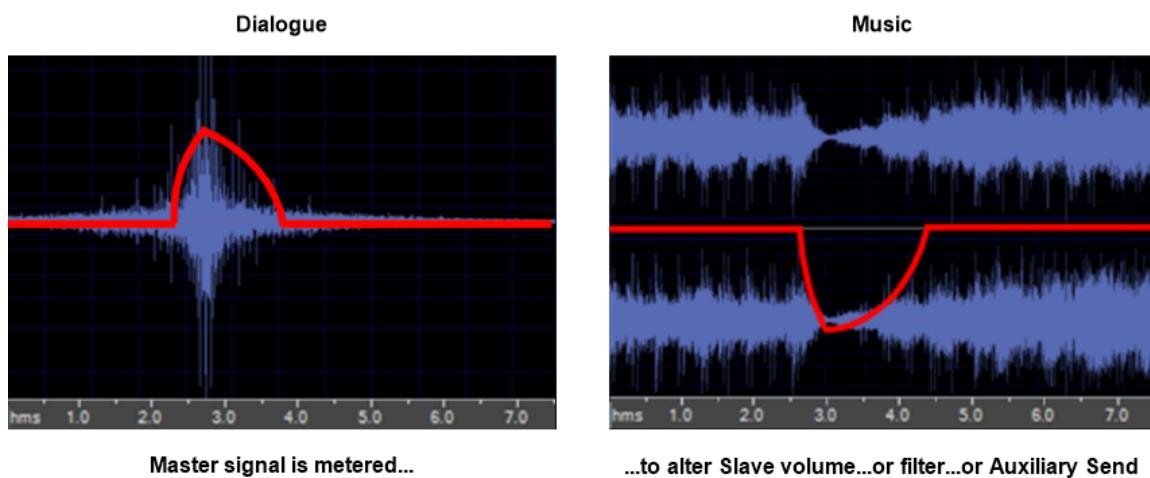
以下页面阐述了如何在游戏中使用旁链压缩，以及具体如何在 Wwise 中应用。

- [游戏应用](#)
- [在 Wwise 中设置旁链压缩](#)
- [其他应用](#)

游戏应用

旁链压缩对游戏来说是非常强大的工具，它方便控制玩家关注焦点，并减少喧闹环境下的嘈杂声。另外对于同类对象，它还有助于安排优先级并获得清晰的混音。

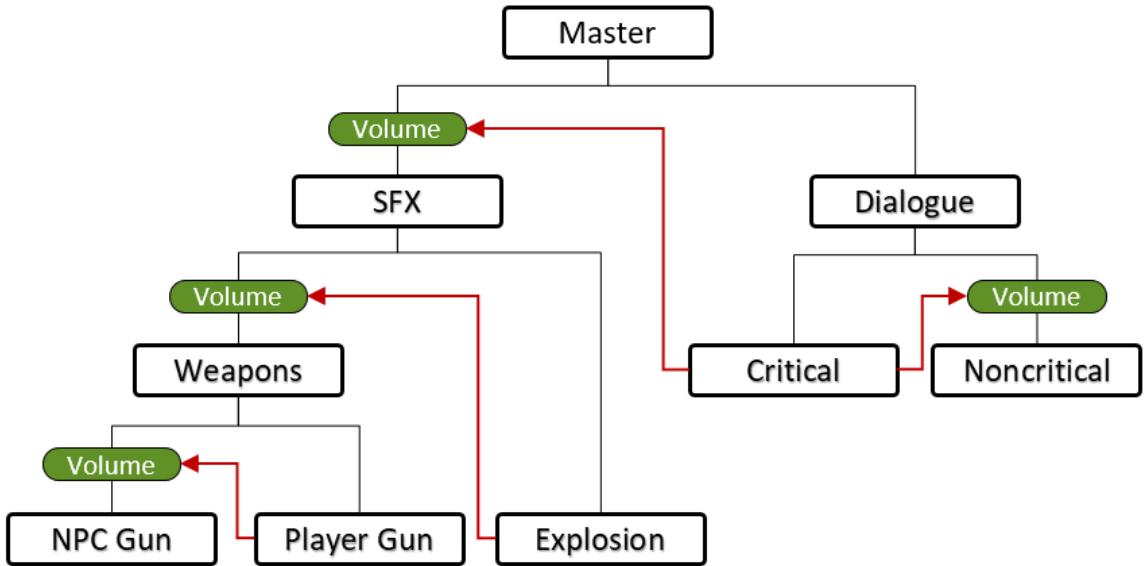
当播放的重要声音中存在瞬态信号时，旁链压缩会按照瞬态形状来降低次要声音的音量。



根据从一些游戏制作中得出的经验，一般会先在同类对象之间设置优先级系统，再在不同类别之间设置优先级系统。例如，按照这一规则，游戏可判定玩家角色 (PC) 的武器声比非玩家角色 (NPC) 的武器声更重要。然后，旁链压缩就可设置为 PC 武器声播放时降低 NPC 武器声的音量。PC 和 NPC 之间的武器声非常相似；但是，在这种情况下，系统需要确保始终将 PC 声音作为玩家的主要关注焦点。

按照这一理念推断，游戏可判定附近爆炸声应作为主要关注焦点，优先于 PC 和 NPC 武器声。同样地，关键对白应优先于音效声（包括武器声和爆炸声）。

以下图表显示了这一系统的层级结构。



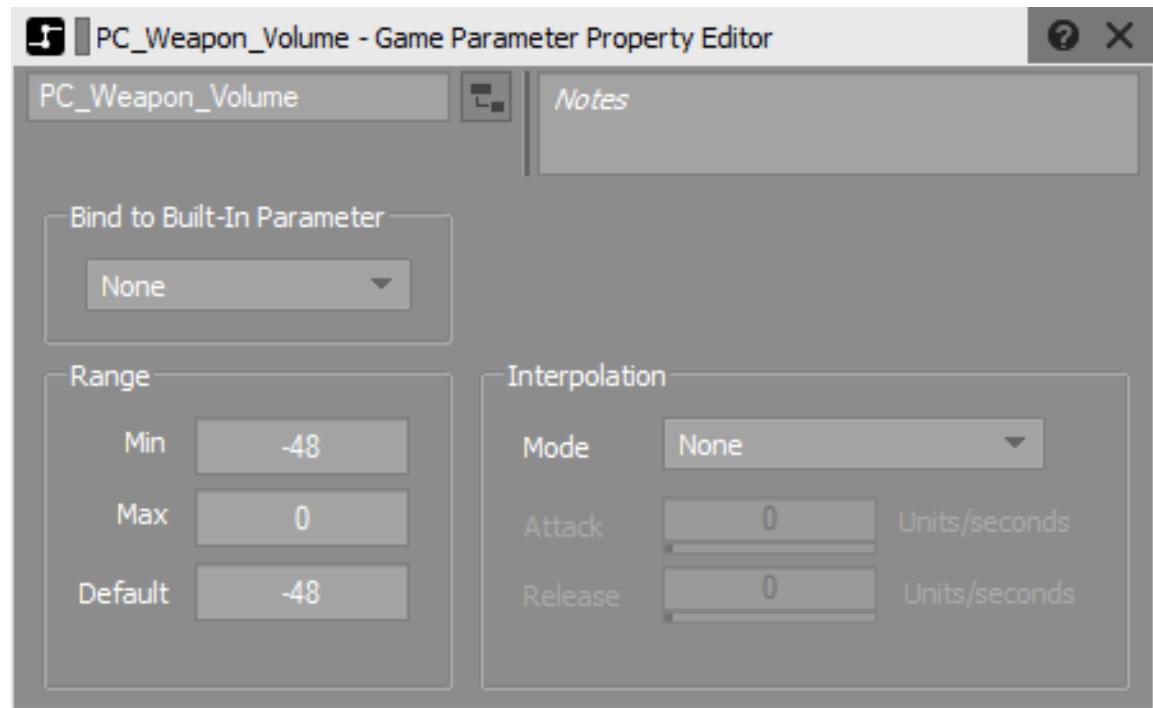
相关主题

- [闪避信号](#)
- [使用旁链压缩](#)
- [在 Wwise 中设置旁链压缩](#)
- [其他应用](#)

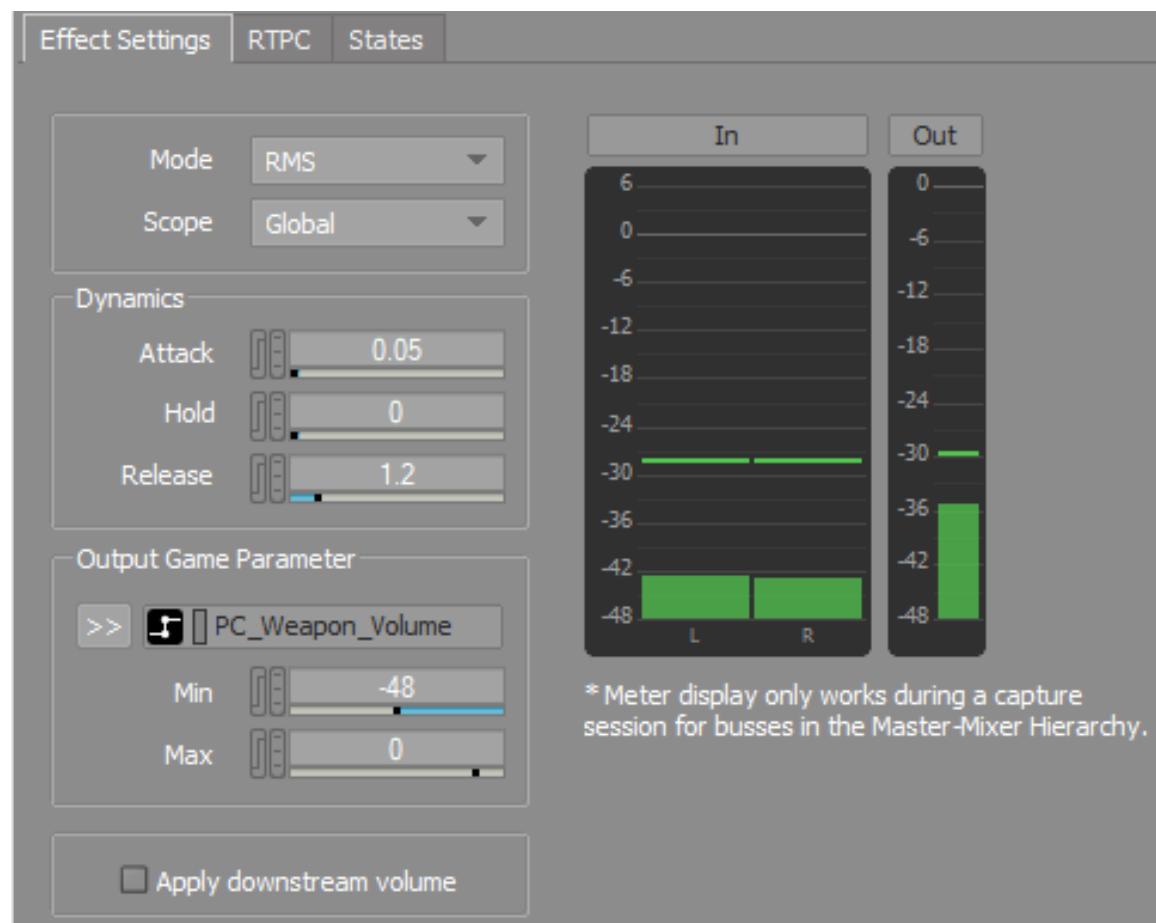
在 Wwise 中设置旁链压缩

在 Wwise 中，可利用实时参数控制 (RTPC) 和 Wwise Meter 效果器轻松设置旁链压缩。以下示例分三步介绍了如何让 PC Weapon 音频总线的信号自动压缩 NPC Weapon 音频总线的信号音量。

1. **创建 Game Parameter:** 首先，创建一个游戏参数（即 PC_Weapon_Volume），并将取值区间设为 -48 ~ 0，近似代表游戏的动态范围。该游戏参数将被用作总线之间的沟通渠道。

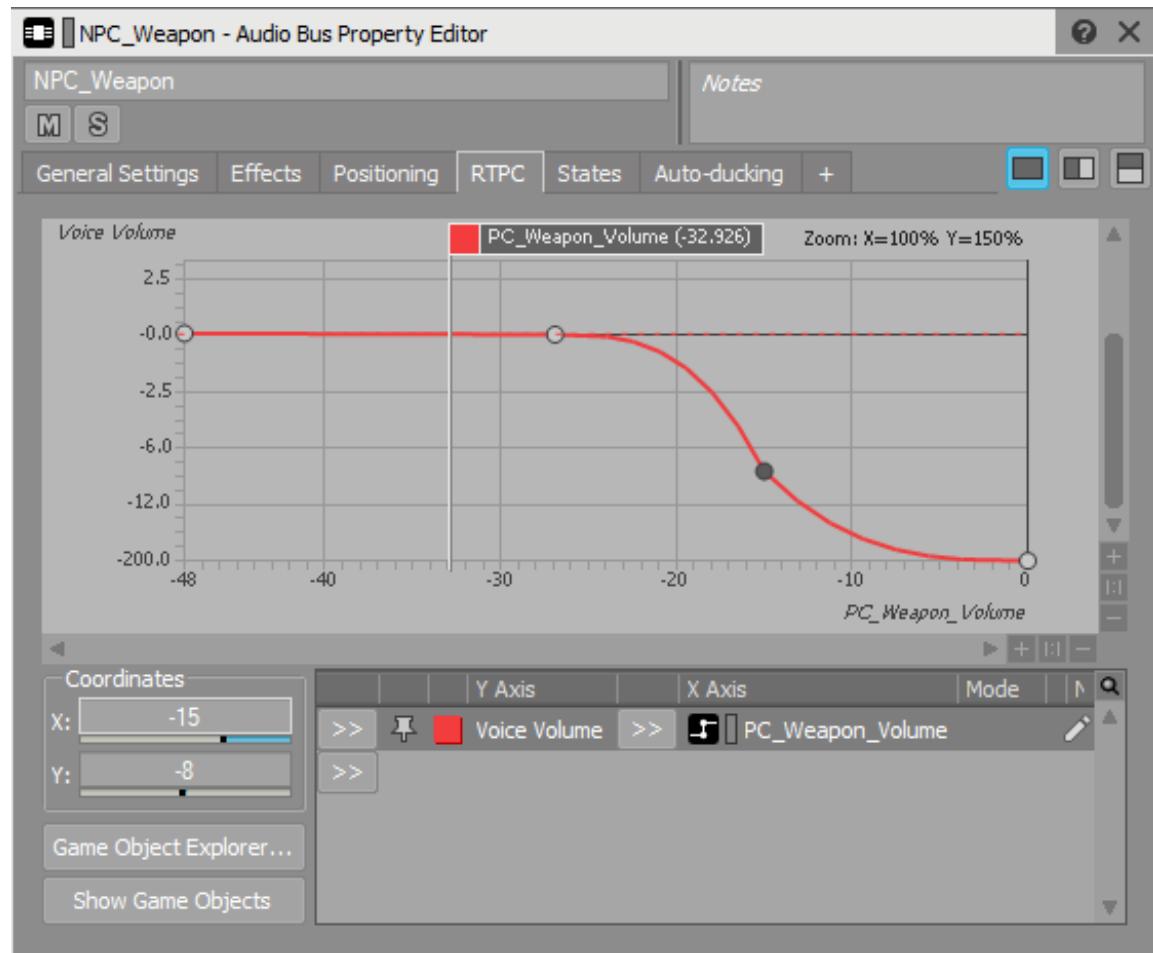


2. 在总线上插入 Wwise Meter 效果器：然后，对于需要确保信号清晰可闻的总线（本例中为 PC_Weapon 总线），为其插入 Wwise Meter 效果器，并选择 Output Game Parameter（输出游戏参数）。



Wwise Meter 效果器会监控输入的音频信号。在将效果器用于旁链压缩时，将使用 Mode（模式）、Attack（起音）、Release（释音）和 Hold（保持）参数来降低输出信号的响应速度，并将平滑处理后的值发送给游戏参数（本例中为 PC_Weapon_Volume）。

3. 创建 RTPC 曲线：最后，为需要衰减的总线创建 RTPC 曲线。在本例中，先将总线音量关联至 PC_Weapon_Volume 游戏参数，然后创建衰减曲线。X 轴代表 Wwise Meter 效果器计算得出的 RMS 信号，Y 轴代表 NPC Weapon 音频总线音量的衰减量。



若要使用旁链压缩构建总线层级结构，只需重复上述三个简单步骤即可。

相关主题

- [使用旁链压缩](#)
- [游戏应用](#)
- [其他应用](#)

其他应用

因为 Wwise Meter 效果器输出的是通用游戏参数，所以任何可关联至 RTPC 的属性都可由旁链压缩驱动。例如，可以用旁链压缩控制 EQ 频段的增益，从而对特定频段进行陷波处理。再如，可以驱动 Compressor 效果器的阈值、修改 Flanger 效果器的 LFO 频率或者增大 FuzzBox Lo-Fi 效果器的失真强度。

Wwise 旁链压缩可灵活用于各种应用。利用这种强大的工具，您将可轻松控制玩家在各种情况下的关注焦点。

相关主题

- [使用旁链压缩](#)

- 游戏应用
- 在 Wwise 中设置旁链压缩

RTPC 技巧与最佳实践

在使用 RTPC 前，您最好仔细阅读以下章节，它们为您提供了一系列的技巧和经验总结，可帮助您在游戏中充分运用您的声音和振动。



RTPC 命名

在更改 Game Parameter 名称之前，一定要查证程序员是如何将其整合到游戏引擎中的。如果整合时使用了游戏参数的名称，则应该尽量避免在 Wwise 中自行更名，因为需要您的程序员进行相应处理。

性能

虽然可为工程中的所有对象、总线、效果器、衰减、切换开关组和混合轨创建 RTPC，但选择性地使用它们非常重要，因为它们会消耗大量的平台内存和 CPU 资源。

智能音高曲线的有效范围

在原声值附近，智能音高曲线可产生良好的效果；但在极值附近就一定动听。例如，在 2000 RPM 时测得的发动机声音在 2000 RPM 处听起来很完美，并且在 500 到 3,500 RPM 之间听起来可能也非常不错。然而，超出这个范围，听起来可能就不太自然。在这种情况下，一种解决办法是使用多个测量值和原声值，然后将测量值汇集到混合容器中。有关混合容器的详细信息，请参阅[定义 Blend Container 的内容和行为](#)。

使用 RTPC 创建多普勒式效果器

Wwise 声音引擎管线中使用的音高属性针对重采样进行高度优化，可以实时地加快或放慢回放。在 Wwise 中创建多普勒式效果器的“经验总结”方法是让游戏引擎跟踪听者与声源之间的位置增量，该增量基本上等于速度值。然后可以使用 RTPC 将“Speed”游戏参数映射到声音的音高属性。当听者和声音在游戏中相互靠近或走远时，音高将升高或降低。这是目前最节省 CPU 资源创建多普勒效果器的方法。

当 Wwise 中使用了多个听者（包括分屏）时，使用音高的多普勒效果器需要特殊的设计考虑。在现实生活中，根据声速以及对象与各个听者之间的距离，声音对各个听者可能具有不同的音高值。在 Wwise 中，由于游戏对象对各个播放音使用一份设定，因此一个声音不可能具有两个不同的音高值。所以，需要通过编程的方式确定一个音高值。

相关主题

- [使用旁链压缩](#)

- 游戏应用
- 在 Wwise 中设置旁链压缩
- 其他应用

第 20 章 使用 Trigger

概述	407
使用 Trigger	408

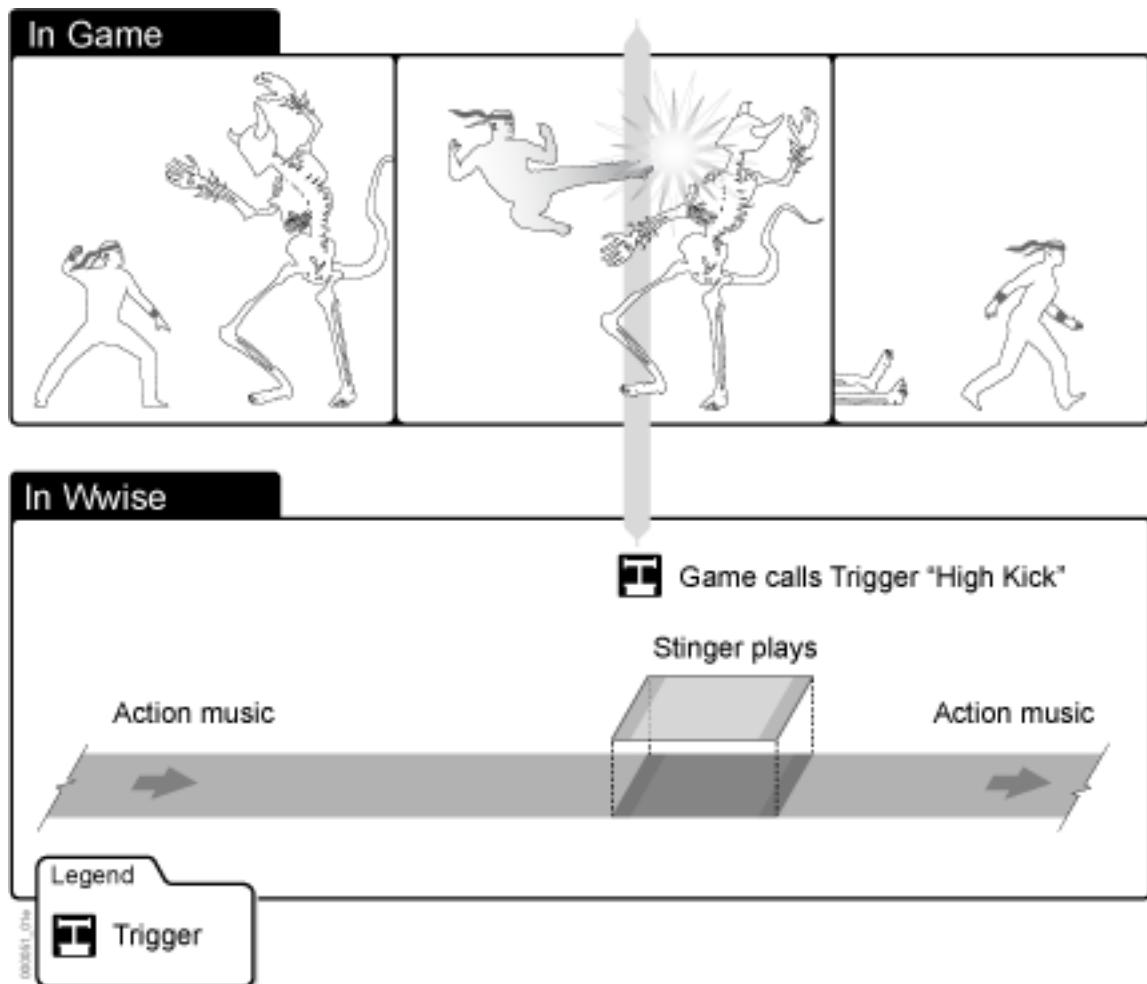
概述

Trigger（触发器）是一种 Wwise 元素，跟其他 Game Sync（游戏同步器）一样会被游戏调用。触发器会定义 Wwise 将做出怎样的特定响应来反映游戏中的情节变化。具体到互动音乐中，Trigger 在响应游戏突发事件时，将播放 Stinger（插播乐句）。Stinger 是一种短乐句，它会与当前音乐叠加并混合播放，以音乐的形式来对游戏做出响应。例如，当忍者拔出武器时，您可能希望播放动作音乐的同时插入强调的音乐效果，增强情节感染力。游戏调用 Trigger，继而播放 Stinger，强调音乐片段将随当前配乐一起播放。关于 Trigger 和 Stinger 如何协同工作的详细信息，请参阅[第 29 章 使用 Stinger](#)。

使用 Trigger —— 示例

假设您创作了一款格斗游戏，主人公是一位忍者战士，经常会进入与敌人格斗的打斗模式。当角色踢出一记强力飞腿时，您希望播放一个音乐片段，增强听觉感染力。要为这样的动作配乐，您需要创建 Trigger，可将它命名为“High kick”，在游戏需要的时候调用即可。另外需要定义简短的 Music Segment（音乐段落），可以是一段快节奏的铜管乐，增强“飞腿”效果。

下图展示了游戏关键时刻用来播放 Stinger 的 Trigger 机制。



使用 Trigger

在游戏的关键时刻，Trigger（触发器）将调用 Stinger（插播乐句），让一段简短的 Music Segment（音乐片段）与当前音乐叠加播放。创建 Stinger 时，要将各个 Trigger 映射到 Music Segment，并进行播放设置。您可以在 Project Explorer（工程浏览器）的 Game Syncs（游戏同步器）选项卡中管理 Triggers 列表。

为了帮您在界面中轻松识别 Trigger，Wwise 使用独特的图标来标识它。

图标	代表
	Trigger（触发器）

管理 Trigger 涉及以下任务：

- [创建 Trigger](#)
- [删除 Trigger](#)

创建 Trigger

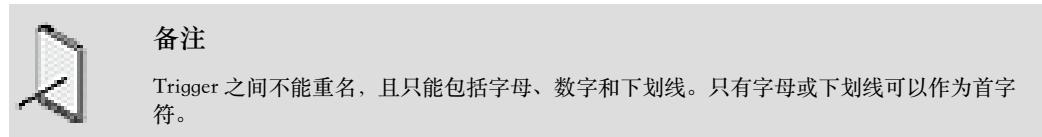
您可以在 Project Explorer 的 Game Sync 选项卡中为工程创建所有需要的 Trigger。

在 Project Explorer 中新建 Trigger 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Triggers 层级，执行以下操作之一：
 - 选择 Work Unit 或 Virtual Folder，然后点击 Project Explorer 工具栏中的 Trigger 图标。
 - 右键点击 Work Unit 或 Virtual Folder，然后从快捷菜单中选择 <New Child> Trigger。

新的 Trigger 将添加到触发器列表中。

3. 将默认名称替换成更合适的命名。



4. 继续按需添加 Trigger。

相关主题

- [删除 Trigger](#)
- [第 29 章 使用 Stinger](#)

删除 Trigger

您可能想删除不再需要的 Trigger（触发器）。记住，如果 Trigger 已经关联了 Stinger（插播乐句），则删除时，Stinger 与 Music Segment（音乐片段）之间的关联也不复存在。

删除 Trigger 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 Triggers 层级，右键点击要删除的触发器，并选择 Delete Selection。

选定的 Trigger 将被删除。

相关主题

- [创建 Trigger](#)
- [第 29 章 使用 Stinger](#)

第 21 章 将 State 和 State Group 用于动态对话

概述	411
使用 State Group	411

概述

如今，许多游戏中都含有 Dynamic Dialogue（动态对白），或是由游戏情节驱动的音频内容，在体育游戏中尤为显著。如果用传统方法模拟真实对白，可能要创建成千上万的素材，再构建复杂的 Switch Container（切换容器）层级结构，才能满足所有可能的方案。由于内存成本昂贵，所以您需要通过有效的方式来管理工程中的素材。

为了应对这些挑战，Wwise 引入了一种独特的动态对话管理方式，即采用 State（状态）和 State Group（状态组）来预先定义游戏中所有可能的情况或结果。State Group 可以表示游戏中的各种元素类别，例如足球游戏中，State Group 可能是球队、球员和动作类型等。每个 State Group 类别都需要一组对应的 State（状态）值，本例中的球队 State Group 可能包括多个 State 值，如 Dallas、Pittsburgh、New England 等。

在对白事件中，State Group 和 State 将被排列组合，以涵盖游戏中所有可能的情况。这些情况称为 Path（路径），它们会被指派给特定的 Voice（语音）对象。游戏过程中的实时状态会与 Wwise 内对白事件中的状态进行对照匹配，从而确定播放哪个对白片段。

将 State 用于动态对话——示例

假定您正在开发一款带有现场解说的高尔夫游戏，需要先为其中各个不同类别分别创建 State Group，每个 State Group 包含该类别所有的 State。高尔夫游戏需要多种 State Group，包括选手（Player）、俱乐部（Club）、球场（Course）、击球（Shot）、位置（Location）、反应（Reaction）等。

下图说明了高尔夫游戏中，如何将部分类别与 State Group 和其中的 State 进行对应。

Arguments	Players	Clubs	Shots	Locations
Argument Values	Woods	Driver	Slice	Fairway
	Ames	Iron	Hook	Tee
	Cabrera	Wedge	Shank	Rough
	Garcia	Putter	Chip	Green

定义了 State Group 和 State 之后，即可将它们添加到游戏所需的 Dialogue Events（对白事件）中。关于创建 Dialogue Event 的更多详细信息，请参阅[创建 Dialogue Event](#)。

使用 State Group

State Group（状态组）代表游戏中的不同类别，例如体育游戏中球队和球员，或者动作冒险游戏中的队友、敌人和武器。在 Project Explorer（工程浏览器）的 Game Syncs（游戏同步器）选项卡中，可以管理 State Group 列表。

为了让您在界面中轻松识别 State Group 和 State，Wwise 将它们用特定图标表示。

将 State 和 State Group 用于动态对话

图标	代表
	State Group
	State (即 State Group 值)

管理 State Group 涉及以下任务：

- [创建 State Group](#)
- [创建 State](#)
- [删除 State Group 或 State](#)

创建 State Group

在 Project Explorer (工程浏览器) 的 Game Sync (游戏同步器) 选项卡中，可以创建工程所需的所有 State Group (状态组)。

在 Project Explorer 中新建 State Group 的方法如下：

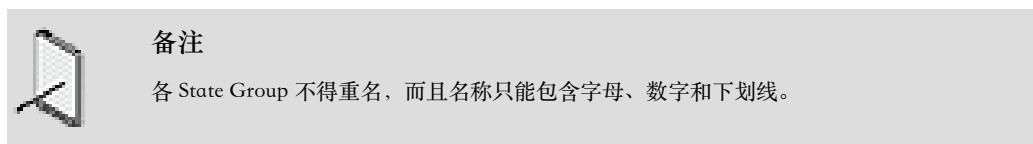
1. 在 Project Explorer 中，切换至 Game Syncs (游戏同步器) 选项卡。
2. 在 State (状态) 层级中，执行以下操作之一：

选中 Virtual Folder (虚拟文件夹) 或 Work Unit (工作单元)，并点击 Project Explorer 工具栏上的 State Group 图标。

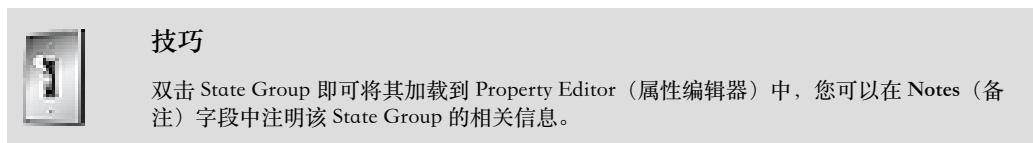
右键点击 Virtual Folder 或 Work Unit，从快捷菜单中选择 New Child > State Group (新建子项 > 状态组)。

新 State Group 将被添加至 State Group 列表。

3. 将默认名称替换为更合适的命名。



4. 根据需要，继续添加 State Group。



相关主题

- [创建 State](#)
- [删除 State Group 或 State](#)

创建 State

您创建的各个 State Group (状态组) 可以具有多个不同的值，分别代表每个类别内的不同选项。例如，Player Name (玩家名字) 这个 State Group 中，各 State (状

态) 就是游戏中各个玩家的名字。可以在 Project Explorer (工程浏览器) 的 Game Syncs (游戏同步器) 选项卡中创建工程所需的所有状态。

在 Project Explorer 中新建 State 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 State (状态) 层级中，执行以下操作之一：

选中 State Group，并点击 Project Explorer 工具栏中的 State 图标。

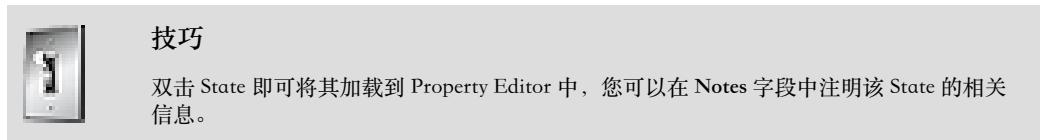
右键点击 State Group，并从快捷菜单中选择 New Child > State (新建子项 > 状态)。

新状态将被添加到 State Group 中。

3. 将默认名称替换为更合适的命名。



4. 根据需要，继续添加状态。



相关主题

- [创建 State Group](#)
- [删除 State Group 或 State](#)

删除 State Group 或 State

您可能需要删除不再使用的 State Group (状态组) 或 State (状态)。删除 State Group 时，其中的所有 State 也将被删除。请记住，如果删除了某个 State Group，则该 State Group 将会从所有用到它的对白事件中删除。

删除 State Group 的方法如下：

1. 在 Project Explorer 中，切换至 Game Syncs 选项卡。
2. 在 States 层级中，右键点击要删除的 State Group 或 State 并选择 Delete Selection (删除选中项)。

所选 State Group 或 State 将被删除。



相关主题

- [创建 State Group](#)
- [创建 State](#)



22. 理解互动音乐	418
概述	419
理解互动音乐	419
互动音乐技巧和经验总结	420
23. 构建 Interactive Music Hierarchy	423
概述	424
什么是 Music Segment?	424
容器类型	426
关于 Interactive Music Hierarchy 中的属性	427
将对象添加到 Interactive Music Hierarchy	428
添加父对象	429
管理 Interactive Music Hierarchy 中的音乐对象	430
构建 Interactive Music Hierarchy 的技巧和经验总结	431
24. 定义音乐对象播放行为	432
概述	433
定义音乐对象的 Time Settings	433
定义 Music Playlist Container 的播放行为	434
定义 Music Switch Container 的内容和行为	437
互动音乐播放的技巧和经验总结	440
25. 使用 Music Track 和 Music Segment	442
概述	443
将 Music Track 添加至 Music Segment	444
定义 Music Track 的播放行为	445
将子音轨添加至音轨	447
将 Sub-Track 关联至 Switch/State	448
添加音轨	449
从段落中移除音轨和子音轨	449
在 Music Editor 中按间距对齐	450
使用片段	450
预览 Music Segment	453
使用 Cue	454
26. 处理 MIDI	460
创建 MIDI 内容	461
导入MIDI 文件	461
理解 MIDI 内容和 MIDI 目标	461
混合 MIDI 和音频内容	462
理解 MIDI 速度	463
更改 MIDI 的播放速度	463
27. 创建 MIDI 乐器	465
设计 Synth One 乐器	466
设计简单的采样 MIDI 乐器	466
理解 MIDI 音符跟踪	466
理解 MIDI 筛选器	467
了解 MIDI 事件	467
在 MIDI 事件上添加淡入和淡出	468
使用 MIDI 数据控制对象属性值	468
使用 MIDI Keymap Editor	469
使用 MIDI 键盘测试乐器	469
将 MIDI 从 DAW 连通到 Wwise	470

28. 使用 Transition	472
概述	473
理解 Transition	473
添加 Transition	475
复制和粘贴 Transition	476
移除 Transition	476
设置源和目标属性	477
使用 Transition Segment	479
互动音乐中 Transition 的技巧和经验总结	481
29. 使用 Stinger	483
概述	484
添加 Stinger	485
定义 Stinger 的播放设置	486
移除 Stinger	487
预览 Stinger	488

第 22 章 理解互动音乐

概述	419
理解互动音乐	419
互动音乐技巧和经验总结	420

概述

将配乐集成到游戏的过程中存在诸多特殊挑战。电子游戏配乐不仅应具有互动性，而且实现起来应该尽量直观，还要充分利用已有资源。Wwise 的互动音乐功能为您提供了解决这些问题的工具。

Wwise 允许将正在播放的音乐与游戏情节绑定，增强游戏配乐的互动性。其图形界面便于您安排和编辑音乐片段，并决定它们在游戏中如何播放，何时播放。Wwise 的互动音乐功能还可以从多方面帮您节省资源。在长达数小时的游戏过程中，仅使用有限数量的音乐片段可以帮助项目节约资金。互动音乐功能还可以节约人力，作曲家或其它音乐专业人士以前需要为日常编曲任务花费大量时间，现在可以加速完成。对于最终成品，互动音乐功能还能帮您管理游戏机资源，可以在播放音乐的同时控制音乐的时长、播放时机和属性。

理解互动音乐

成功的互动音乐可以让游戏配乐富于变化和意义，更适合游戏情景，从而提升游戏体验。例如战斗时可以加快节奏，变得更加有力，而在一片太平之下，可以较为平静和轻快。还可以在当前音乐上叠加播放 Stinger（插播乐句）或片段来烘托特殊事件。

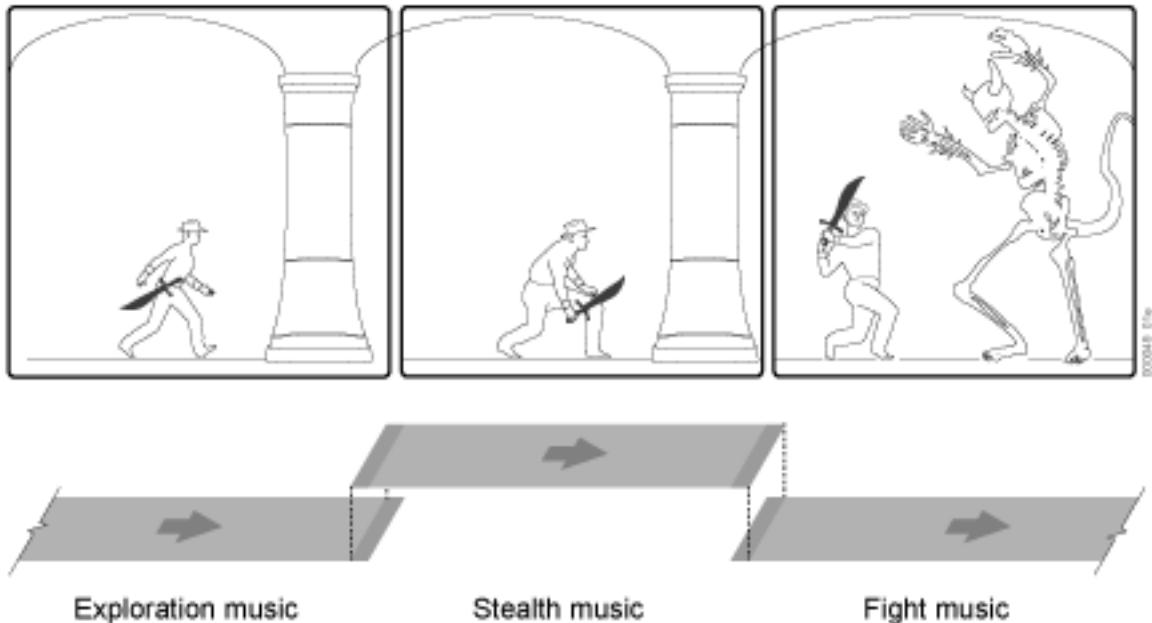
成功的互动音乐工程还能充分利用为数不多的资源。源音乐时长与游戏中的配乐总时长之间的比例通常会很低：一小时的作曲覆盖三十个小时的游戏时长并不鲜见。您可以用 Wwise 来组织播放，满足效率需求的同时兼顾多样性。可以顺序或随机播放音乐片段，甚至单独播放构成这些片段的音轨。这样听起来就好像您的游戏配备了时间持久、不断变化的音乐，而原始素材的长度可能只占总时长的一个零头。

使用互动音乐——示例

假设您正在创建一款探险/益智游戏，玩家扮演一位勇敢的考古学家，在古庙中探险，搜寻宝物。

游戏中，玩家有三个主要行为：探索寺庙、躲避陷阱以及与寺庙护卫格斗。您可以用 Wwise 为这些行为分别定义游戏状态：探索（Exploration）、躲避（Stealth）和格斗（Fight）。这些状态会影响游戏中不同情况下 Wwise 将播放哪些音乐，确保音轨与游戏中的行为匹配。

下图展示了使用互动音乐可以达到的效果：



您还可以通过互动音乐让游戏音乐变得更持久、更有趣。毕竟，如果游戏环境优美新奇，玩家可能愿意花大量时间进行探索。要是考古学家探险时配乐机械地重复，玩家则可能会失望甚至失落。通过互动音乐可以利用和扩展有限数量的源素材。

总之，互动音乐是功能多样的工具，可以将您的游戏从简单的休闲娱乐提升为一种参与体验。玩家会感受到考古学家的兴奋和紧张，甚至会花时间来调查精心设计的游戏陷阱和谜题。

互动音乐工程结构

构建互动音乐工程时，Wwise 具有很强的灵活性，有无数种方法可以将互动音乐对象编排组合成游戏配乐。不过，遵守基本统一的结构可以让您的工作流程变得更加高效。互动音乐工程的两种基本结构如下：

- **纵向结构：**改变 Music Segment（音乐段落）中包含的音轨序列，对游戏配乐重新编排。这类似于在音乐制作中使用的音轨混音，可以帮您利用多轨的长段落让配乐多样化。
- **横向工程结构：**通过随时改变播放的音乐段落，让游戏配乐富于变化。可以在 Interactive Music Hierarchy（互动音乐层级）中安排短小的独立段落，就像在 Actor-Mixer Hierarchy 中安排对象一样。这样不仅可以基于一系列短小的音乐片段制作引人入胜的配乐，而且还可以将游戏机需求降到最低。

一般来说，可以同时使用这两种结构来高效地利用工程资源。良好的结构可以让您充分展示音乐，并将游戏机资源运用到极致。

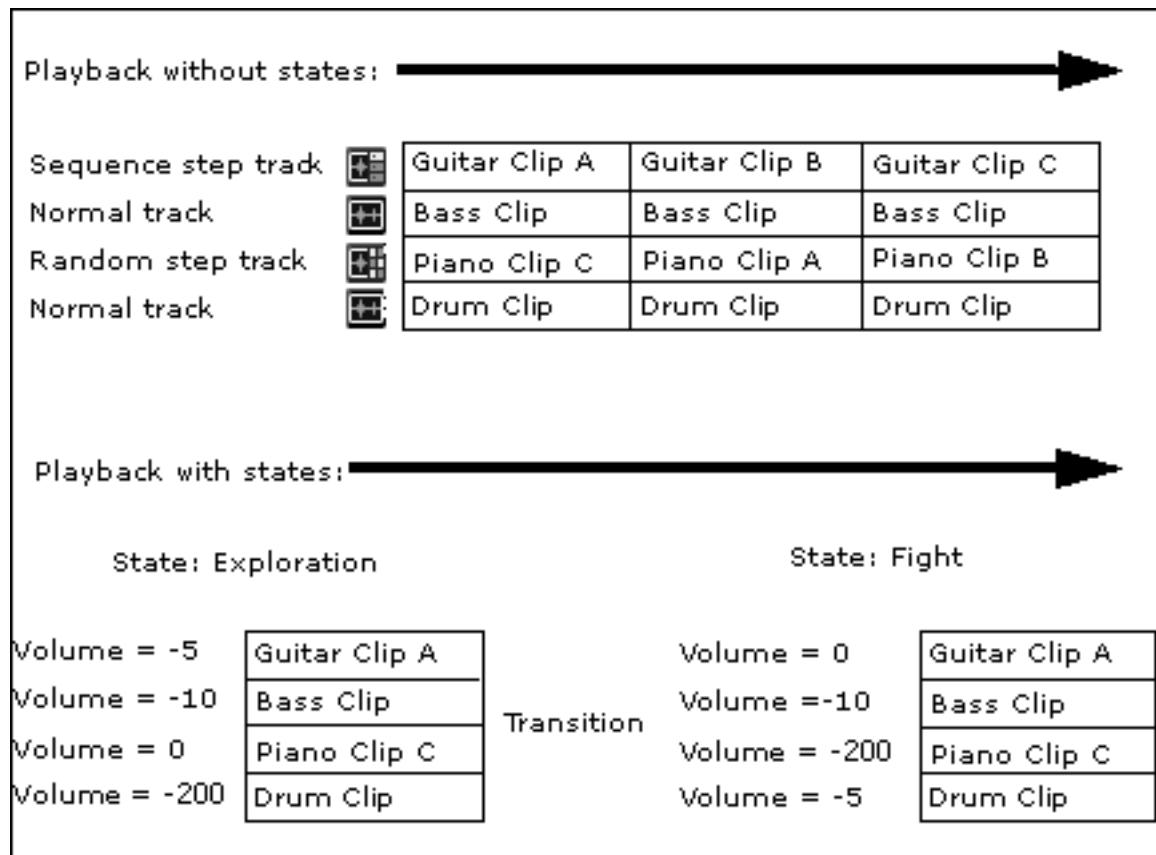
关于如何组织工程的详细信息，请参阅[互动音乐技巧和经验总结](#)。

互动音乐技巧和经验总结

互动音乐是具有多种功能的复杂工具，工程伊始就为其制定一致的策略可节省后期的时间和精力。当然，处理互动音乐工程的方法很多，您可以在 Wwise 中使用任何方法来为游戏创造最佳的结果。以下是有关如何实现互动音乐的建议。

实现纵向结构

纵向结构的基础是少量重复和变化的 Music Segment（音乐段落），每个段落含有多少条 Music Track（音乐轨）。基本的纵向结构可能如下图所示：



此例中的互动音乐基于一条四轨的 Music Segment。上半部分中，段落正在播放，但没有指定状态。贝斯和鼓的片段位于 Normal (正常) 音轨中，每次重复时会播放相同的片段。而吉它则使用序列步进 Sequence Step (序列步进) 音轨，其中的片段将按照指定的序列播放。钢琴则是 Random Step (随机步进) 音轨，其中的片段将随机播放。对于 Sequence Step 和 Random Step 音轨，制造多样性的关键在于创建 Sub-track (子音轨)，让其中的片段替代原始音轨中的片段。

下半部分中，音轨开始响应状态切换。当游戏处于相对平静的探索状态时，鼓是听不到的，只会播放钢琴音轨（由多个子音轨构成）。一旦过渡到激烈的格斗状态，钢琴轨的音量将被拉到底，而鼓的音量将增大，吉它音量也会增大。总体效果是音乐将随着游戏情节而改变。

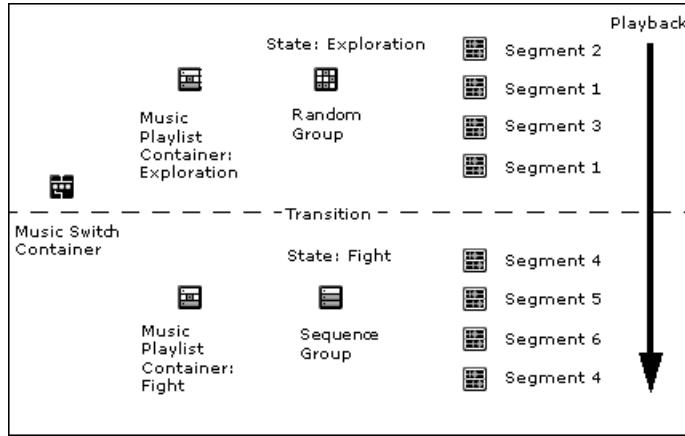
在纵向结构中，可以使用以下方式来改变将被播放的音轨：

- State 变换
- Random step 子音轨
- Sequence step 子音轨。

如果工程只有少量复杂音乐片段，并且状态切换频率较低，就非常适合纵向法。

实现横向工程结构

横向工程结构的基础是多个简短段落的层级编排。基本的横向工程结构可能如下图所示：



此例中，两个 Music Playlist Container（音乐播放列表容器）中包含对应于两种游戏状态的 Music Segment：Exploration（探索）和 Fight（格斗）。第一个容器中，段落随机组合排列，因此将随机播放。第二个容器中，段落按序列组合排列，因此将按预定序列播放。游戏从 Exploration 状态过渡到 Fight 状态时，Exploration 容器中播放的最后一个段落将过渡到 Fight 容器中的第一个段落。得到的效果是从一种音乐整体切换到了另一种音乐。

横向工程结构中，可以使用以下方式改变将要播放的段落：

- State 变换
- Random Container（随机容器）
- Sequence Container（序列容器）

如果工程包含大量简短而相对简单的音乐片段，特别是还会频繁切换状态的话，建议使用横向法。

实现组合式工程结构

通过 Wwise 创建的工程大多同时使用横向和纵向元素，您可以决定每种结构占多大比例最适合您的工程。例如，可以为每种状态创建不同的段落（横向结构），同时用随机子音轨改变各个段落中的音轨播放（纵向结构）。

使用多级 Music Switch Container

Music Switch Container 支持绑定到多个 State Group（状态组）或 Switch Group（切换开关组）上。如果音乐依赖于多种游戏条件，也最好能避免创建层叠式的 Music Switch Container 结构，而是将所有条件应用至同一个 Music Switch Container。Music Switch Container 中的内容可以使用虚文件夹来编组，关联时也允许复用来自多个 State 或 Switch 的内容。

请注意，使用多级 Switch Container 会造成有些过渡无法设置重要性顺序，因为过渡是在多个不同子对象中设置，它们各自内部都有普通过渡规则，而且子对象间不存在优先级关系。因此音乐过渡可能会出乎预料，并且很难跟踪所有的可能性。

第 23 章 构建 Interactive Music Hierarchy

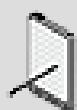
概述	424
什么是 Music Segment?	424
容器类型	426
关于 Interactive Music Hierarchy 中的属性	427
将对象添加到 Interactive Music Hierarchy	428
添加父对象	429
管理 Interactive Music Hierarchy 中的音乐对象	430
构建 Interactive Music Hierarchy 的技巧和经验总结	431

概述

提供游戏音乐时，您需要管理许多音乐素材，将它们分组整理是简化管理的一种有效方式。就像在 Actor-Mixer Hierarchy（角色混音器层级结构）中允许把对象组织在一起并创建父/子关系，同样，在 Interactive Music Hierarchy（互动音乐层级结构）中，您可以将工程的所有音乐素材有效地组织起来。除了组织素材，还可以为各音乐分组设置属性和行为。

通过将以下音乐对象类型结合使用，您可以对音乐素材进行分组，并构建互动音乐结构：

- **什么是 Music Segment？** —— 由若干 Music Track（音乐轨）组成，每个 Music Track 都包含 Music Clip（音乐片段）。这些片段可以在 Music Segment（音乐段落）内进行排列对齐。
- **容器类型** —— 包含一组 Music Segment 或其它音乐容器，并遵循特定的播放行为。音乐容器共有两种：
 - Music Switch Container（音乐切换容器）根据调用的 Switch（切换开关）或 State（状态）来播放音乐对象。
 - Music Playlist Container（音乐播放列表容器）会按随机或顺序方式播放音乐。



备注

也可以用 Virtual Folder（虚拟文件夹）将音乐对象分组，但与其它音乐对象不同，Virtual Folder 不具有任何属性或行为。

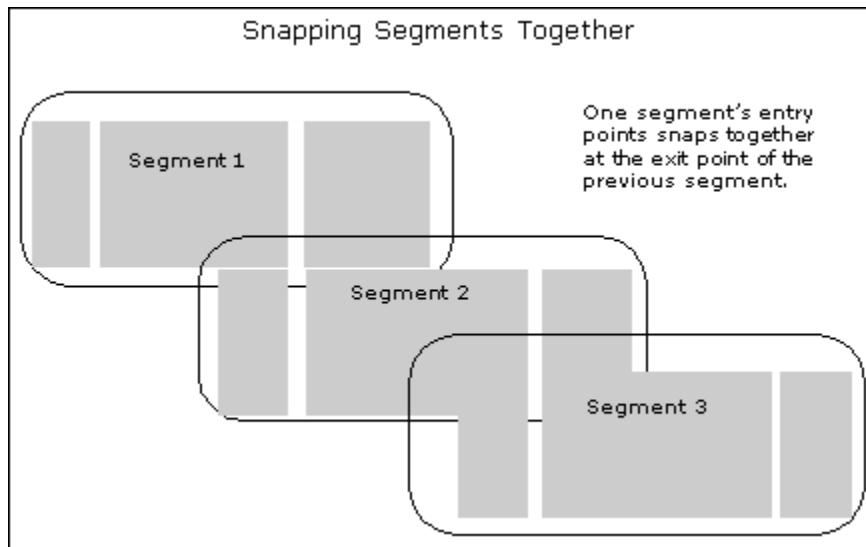
什么是 Music Segment？

Wwise 的 Music Segments（音乐段落）作为 Interactive Music Hierarchy（互动音乐层级结构）的基础，是一种核心音乐对象。它包含音乐素材，与 Actor-Mixer Hierarchy（角色混音器层级结构）中的声音对象相似。但是，声音对象和 Music Segment 有几个重要不同点：

- Music Segment 可以与其它段落对齐。
- Music Segment 与音乐源之间还有另一个层次，即 Music Track（音乐轨）。

对齐 Music Segment

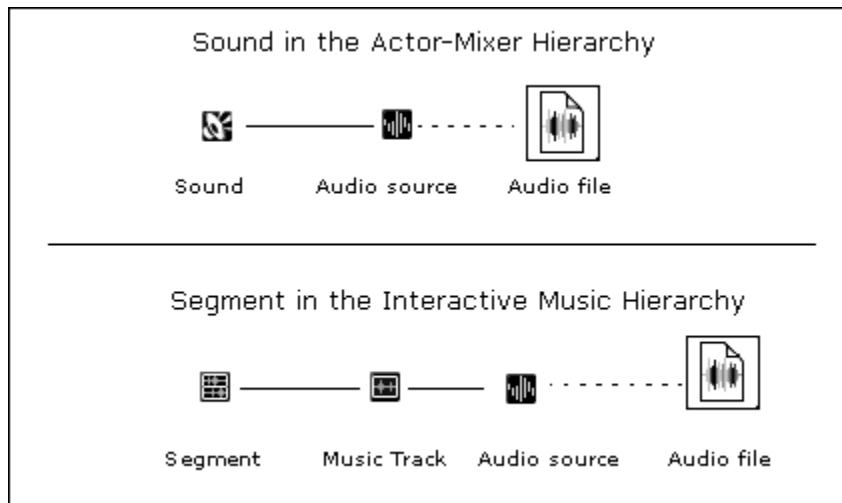
Music Segment 可在指定点严格对齐，让配乐编曲无缝衔接。下图说明了如何使用 Sync Point（同步点）或 Cue（提示点）来排列和对齐段落。Cue 是附加在 Music Segment 上的标记，用来指示关键时间点，例如 Entry（入点）和 Exit（出点）。将 Cue 结合曲速拍号设置一起使用，就可以按需对齐播放列表中的各段落。



Music Segment 和 Music Track

通常，Music Segment 中至少包含一个 Music Track，您需要用 Music Track 作为中间层来处理 Music Clip（音乐片段）。每个 Music Clip 代表一个音频源，它包含针对不同游戏平台的转码设置，用于转换相应的音频文件。转码过程中，会为各平台创建新文件，而原始音乐文件则保持不变。这样您的互动音乐就可以针对不同目标平台进行优化了。关于针对平台进行音频文件转码，更多信息请参阅[跨平台制作](#)。

下图说明了音频文件和声音对象、音频文件和 Music Segment 之间的关系。



同一声音对象中，可以添加多个音频源，各自链接不同的音频文件，并允许选择其一来播放。不过，若在 Music Track 中添加新的音频源，则会作为另一 Music Clip 添加到该 Music Track，并显示在 Music Segment Editor（音乐段落编辑器）中。

除此之外，还可将 Event 放在 Music Segment Editor 音轨内的特定时间点。

为了便于在界面中识别 Music Segment 和 Music Track，Wwise 分别为其使用了不同的图标。

图标	代表
	Music Track (音乐轨)
	Music Segment (音乐段落)

相关主题

- [处理 Event](#)
- [概述](#)

容器类型

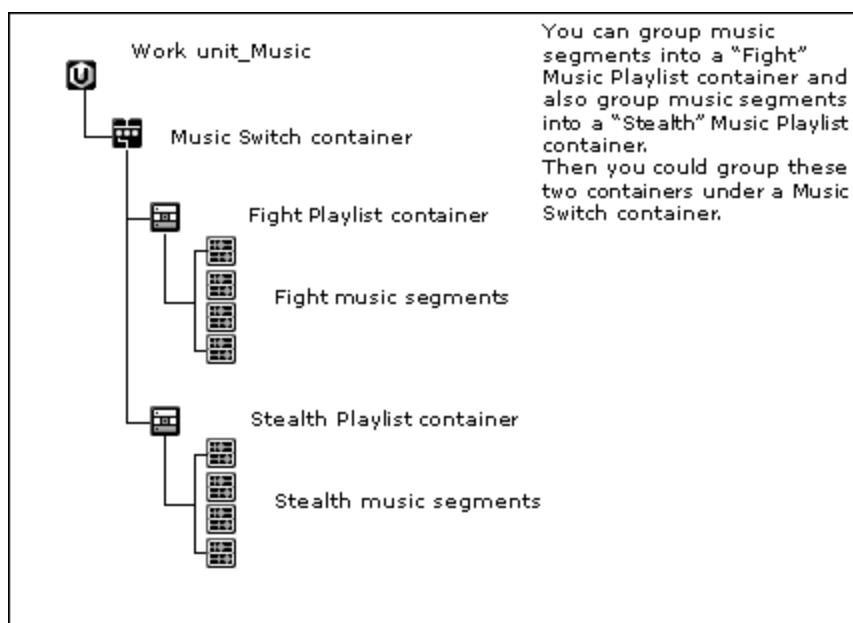
为了让游戏配乐充分发挥作用，您需要设置具有不同播放行为的容器。在 Interactive Music Hierarchy (互动音乐层级结构) 中，可以设置两类容器：Music Switch Container (音乐切换容器) 和 Music Playlist Container (音乐播放列表容器)。

下表列出了 Interactive Music Hierarchy 中允许使用的容器种类。

图标	代表
	Music Playlist Container
	Music Switch Container (音乐切换容器)

组织音乐素材 —— 示例

假设您正在为一款第一人称动作冒险游戏创建互动音乐，并且想要不同的潜行和战斗音乐。可以把这些音乐对象组织到 Music Switch Container 中，并管理其子对象的属性。在 Music Switch Container 中，您可以添加若干包含音乐对象的 Music Playlist Container，并将它们分别指派给 Music Switch Container 中的 Switch 或者 State。



在工程的早期阶段，就可以基于游戏设计来构建音乐层级结构了。同时还需考虑工程中的其它元素，例如 Work Unit（工作单元）、Routing（信号通路）以及 Game Syncs（游戏同步器）等。从工程整体来考虑问题可以帮您有效地为对象分组。

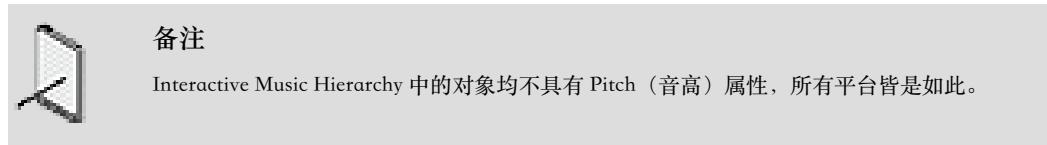
关于如何建立通路结构、Work Unit 以及创建 Game Sync 的详细信息，请参阅以下几节：

- [第 6 章 管理 Workgroup](#)
- [概述](#)
- [第 17 章 使用 State](#)
- [第 20 章 使用 Trigger](#)
- [第 18 章 使用 Switch](#)
- [第 19 章 使用 RTPC](#)

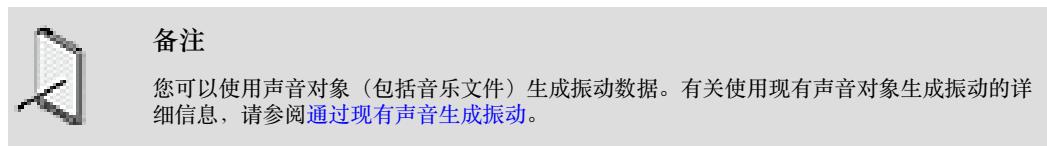
关于 Interactive Music Hierarchy 中的属性

Actor-Mixer（角色混音器）和 Interactive Music（互动音乐）层级结构以相同方式管理属性。在 Wwise 中，音乐对象的属性分为两类：

- 相对属性——在层级结构中累积的属性（例如音量和低通滤波器），在各层级上定义，其总和决定最终属性值。了解各相对属性的最大和最小值限制至关重要，各属性的值域如下：
 - Volume（音量）：(-200, +200) 单位为 dB
 - Low-pass filter（低通滤波器）：(0, 100) 以百分比表示
 - High-pass filter（高通滤波器）：(0, 100) 以百分比表示



- 绝对属性——在特定层级（通常为最高层级）上定义的属性，可以传递给其下所有子音乐对象，例如输出通路。但在各层级上，都可以选择 Override（不沿用）父对象的绝对属性。



与声音对象类似，音乐对象也带有属性标记，说明该属性值是否与其它平台 Link（链接），是否与 RTPC（实时参数控制）中的游戏参数关联，以及是否启用了 Randomizer（随机化器）。

关于为属性值启用/取消 Link、使用 RTPC 以及随机化的详细信息，请参阅以下几节：

- [根据平台自定义对象属性](#)
- [使用 Game Parameter 控制属性值](#)
- [通过随机化属性值来改善播放](#)

将对象添加到 Interactive Music Hierarchy

Interactive Music Hierarchy（互动音乐层级结构）允许您灵活组织工程中的音乐对象，实现完整的配乐。创建层级结构从 Work Unit（工作单元）开始，如果您一个人工作，可以直接从 Default Work Unit（默认工作单元）开始创建层级结构；多人协作时，则需首先创建 Work Unit，再将音乐对象添加到相应的 Work Unit 中。关于 Work Unit 和工作组的详细信息，请参阅[第 6 章 管理 Workgroup](#)。

要构建音乐结构，可以执行以下操作之一：

- 设置好工程结构，然后将音频文件导入其中。
- 先导入音频文件，再将其组织到工程结构中。

关于如何导入音频文件，以及如何在 Interactive Music Hierarchy 中创建新对象的详细信息，请参阅[第 7 章 管理工程中的媒体文件](#)。

使用 Project Explorer 工具栏来新建子对象的方法如下：

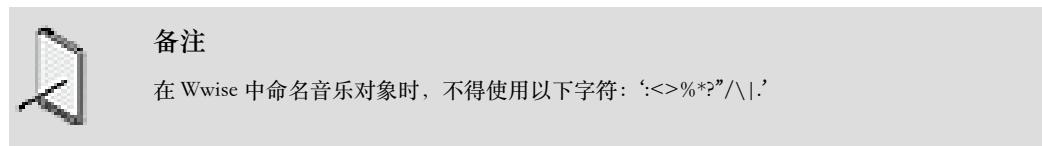
1. 在 Project Explorer 的 Audio（音频）选项卡中，选择新音乐对象所在的工作单元。

Project Explorer 工具栏中的一系列图标将被激活。

2. 在图标列表中，点击所需对象的图标。

该对象将被添加到 Interactive Music Hierarchy 中所需的工作单元下。

3. 将默认名称替换为更合适的名称。



然后可以将其它音乐对象添加到层级结构中。请提前花些时间理解一下对象之间的关系，以便根据关系相应地组织这些对象。在工程后续进程中，这样会帮您节省大量的时间。

在 Interactive Music Hierarchy 中新建子对象的方法如下：

1. 在 Project Explorer 中 Audio 选项卡的 Interactive Music Hierarchy 部分，右键点击新音乐对象所在的工作单元。
2. 在快捷菜单中，选择 New Child（新建子项）。

子菜单中会显示可添加的对象列表。

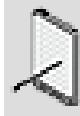
在此层级中，允许添加以下对象：

- Virtual Folder（虚拟文件夹）
- Music Switch Container（音乐切换容器）
- Music Playlist Container（音乐播放列表容器）
- Music Segment（音乐段落）

3. 从列表中选择要添加的对象。

该对象将被添加到 Interactive Music Hierarchy 中。

4. 将默认名称替换为最适合该音乐对象的名称。



备注

在 Wwise 中命名对象时，不得使用以下字符：‘:<>*?%/\|.’

接下来可以继续添加其它对象。请提前花些时间熟悉理解对象之间的关系，以便有依据地组织它们，从而帮您在后续工作中节省大量时间。



快速创建父对象或子对象

在 Project Explorer 内，可使用预定义的键盘快捷方式快速创建父对象或子对象。比如，在默认情况下，按下 Shift+Alt+F 即可创建父文件夹。若要打开 Keyboard Shortcuts 视图并查找工程所用快捷方式，请在 Project 菜单中单击 Keyboard Shortcuts...。

相关主题

- [添加父对象](#)
- [管理 Interactive Music Hierarchy 中的音乐对象](#)

添加父对象

在 Work Unit（工作单元）中添加第一个对象后，可以将更多的对象添加到 Interactive Music Hierarchy（互动音乐层级结构），并建立父子关系。父对象可以包含其它对象，新建父对象后，您可以将现有对象移至其下。创建父子关系的好处是，更改父对象属性和行为时，可以影响其子对象。关于 Interactive Music Hierarchy 中各属性的详细信息，请参阅[关于 Interactive Music Hierarchy 中的属性](#)。

使用 Project Explorer 工具栏新建父对象的方法如下：

1. 在 Project Explorer 中 Audio 选项卡的 Interactive Music Hierarchy 部分，选中需要添加父对象的音乐对象。
2. 按 Shift 键以显示备选音乐对象的图标，这些对象可以作为所选对象的父级。
3. 在列表中，点击要添加的音乐对象图标。

此对象将作为所选对象的父级添加到 Interactive Music Hierarchy 中。

4. 将默认名称替换为更合适的名称。



备注

在 Wwise 中命名对象时，不得使用以下字符：‘:<>*?%/\|.’

新建父对象的方法如下：

1. 在 Project Explorer（工程浏览器）中 Audio（音频）选项卡的 Interactive Music Hierarchy 部分，右键点击您要创建父级的对象。

2. 在快捷菜单中，选择 New Parent（新建父项）。

子菜单将打开，并显示允许添加的对象列表。

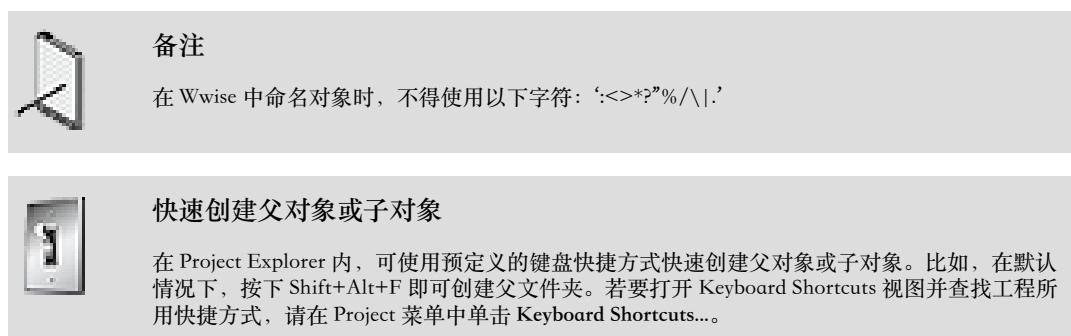
根据所选对象的结构层级不同，可以添加的父对象包括：

- Virtual Folder（虚拟文件夹）
- Music Switch Container（音乐切换容器）
- Music Playlist Container（音乐播放列表容器）
- Music Segment（音乐段落）

3. 在对象列表中，点击要添加的对象。

新的父对象将被添加到 Interactive Music Hierarchy 中，并按对象类型默认命名。

4. 键入更适合该音乐对象的名称。



在 Wwise 中命名对象时，不得使用以下字符：‘:<>*?%/\|.’

- 管理 Interactive Music Hierarchy 中的音乐对象
- 将对象添加到 Interactive Music Hierarchy

管理 Interactive Music Hierarchy 中的音乐对象

在 Interactive Music Hierarchy（互动音乐层级结构）中，您可以用快捷菜单和标准 Windows 快捷方式来重命名、剪切、复制和粘贴对象。在更改层级结构时，请牢记以下几点：

移动音乐对象

操作	Results（结果）
移动音乐对象	<p>如果将对象移动到层级结构中的新位置，该对象的属性和行为将受其新父对象影响。</p> <p>如果您移动一个与某事件相关联的对象，则该对象仍与该事件有关联。</p>

复制和粘贴音乐对象

操作	Results（结果）
复制音乐对象	如果将音乐对象复制并粘贴到新位置，该音乐对象的属性和行为将受其新父对象影响，其子对象也会被复制到新位置。

操作	Results (结果)
	如果被复制的音乐对象与某事件相关联，复制后，新对象将与该事件无关。

剪切/删除音乐对象

操作	结果
剪切或删除音乐对象	<p>如果剪切或删除某音乐对象，则其子对象也将被删除。</p> <p>但是，与其关联的已转码音频文件不会被删除，不再与对象关联的已转码音频文件称为 Orphan File（落单文件）。要删除 Orphan File，您需要清除音频缓存。关于如何删除落单音频文件的详细信息，请参阅清除缓存。</p> <p>如果您删除或剪切的对象与某事件相关联，删除后，将会导致事件中缺失对象。</p>

相关主题

- [将对象添加到 Interactive Music Hierarchy](#)
- [定义音乐对象的 Time Settings](#)
- [定义 Music Playlist Container 的播放行为](#)
- [定义 Music Switch Container 的内容和行为](#)

构建 Interactive Music Hierarchy 的技巧和经验总结

互动音乐是一种复杂工具，具有多种选项。自工程伊始就采取统一策略来构建 Interactive Music Hierarchy，可以在后期节省时间精力。当然，处理互动音乐的方式有很多，在 Wwise 中可以用任何方法来为游戏创造最佳效果。以下是一些建议，告诉您如何为对象分组能够充分发挥互动音乐的优势。

在 Interactive Music Hierarchy 中为对象分组

在构建音乐层级结构之前，您需要构思组织对象的最佳方式，既要节省创作时间，又要更好地管理工程内存占用。

要优化内存使用率，可以考虑在层级较高的对象上设置属性，供整组对象共享。以下属性可以共享：

- Positioning (定位)
- RTPC (实时参数控制)
- State (状态)
- Randomizer (随机化器)

第 24 章 定义音乐对象播放行为

概述	433
定义音乐对象的 Time Settings	433
定义 Music Playlist Container 的播放行为	434
定义 Music Switch Container 的内容和行为	437
互动音乐播放的技巧和经验总结	440

概述

游戏中各种情境下需要播放不同种类的音乐，为满足需求，Wwise 允许独立定义各 Music Segment（音乐段落）和 Music Track（音乐轨）的播放行为。播放音乐时，为了更加灵活可控，Wwise 还允许用两种不同的容器来为对象分组：Music Playlist Container（音乐播放列表容器）和 Music Switch Container（音乐切换容器）。

容器类型决定了其中对象的播放方式，例如，Music Playlist Container 中的内容会根据特定的播放列表来播放，其中可能有一系列随机或顺序播放的对象组。而 Music Switch Container 中的内容则会基于游戏当前的 Switch（切换开关）或 State（状态）来播放。

您可以组合使用不同容器和播放行为，创建多样的情景从而减少重复，让游戏音乐得到极大提升。

定义音乐对象的 Time Settings

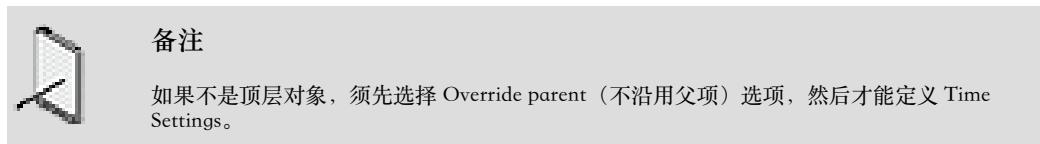
为游戏创作互动音乐时，一个很有效方式是让音乐改变的时机与当前音乐的曲速和节拍相符。原始音乐导入 Wwise 后，可以用 Time Settings（时间设置）来指定音乐对象速度和节拍。这些设置作为 Wwise 中音乐对象的标记，可以让您轻松地设置音乐的同步点和过渡点。

除了曲速和拍号，还可使用 Grid（网格）设置来指定如何划分时间网格，从而将 Music Segment（音乐段落）划分成特定长度的虚拟段落。这将为音乐段落添加另一个划分精度，让您在为音乐过渡、状态更改和插播乐句定义同步点时，有更大的灵活性。

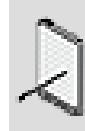
您可以为 Music Segment、Music Playlist Container（音乐播放列表容器）和 Music Switch Container（音乐切换容器）设置 Time Settings。

为音乐对象设置 Time Settings 的方法如下：

1. 将音乐对象加载到 Property Editor（属性编辑器）。



2. 在 Tempo（速度）文本框中，键入原始音乐的速度（单位为每分钟节拍数，即 bpm）。
3. 在 Time Signature（拍号）文本框中，键入原始音乐每小节的拍数和长度。
4. 在 Frequency（网格频率）下拉菜单中，选择用多长的网格来划分 Music Segment，例如 4 bars（每 4 小节）、beat（每拍）、whole note（每个全音符）等。
5. 若要为频率值设定偏置，请从列表中选择预定义选项，或者通过选择 Custom（自定义）并在对应 Offset（偏置）文本框中键入数值（单位为毫秒）来创建自定义偏置。



备注

Grid 设置中的 Frequency 和 Offset 让音乐对象多了一种划分精度，在给音乐过渡、状态更改和插播乐句设置同步点时，将会更加灵活。

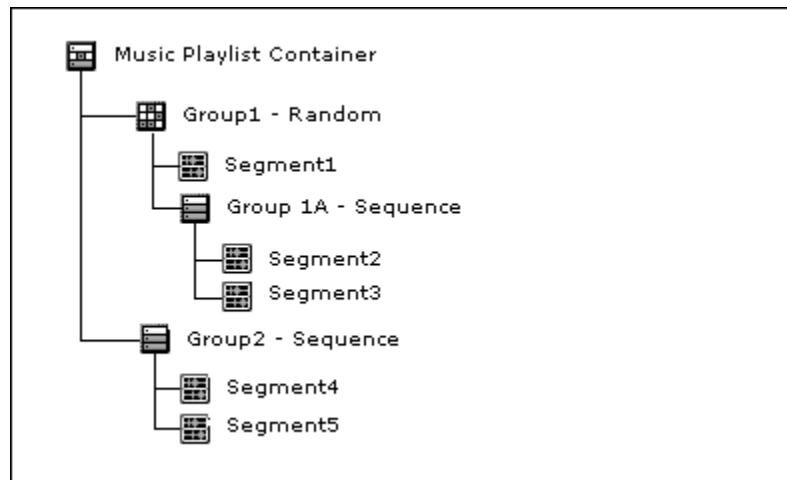
相关主题

- [定义 Music Track 的播放行为](#)
- [定义 Music Playlist Container 的播放行为](#)
- [定义 Music Switch Container 的内容和行为](#)

定义 Music Playlist Container 的播放行为

为了让工程中数量有限的音乐得到充分利用，您可以用 Music Playlist Container（音乐播放列表容器）来为 Music Segment（音乐段落）编组。Music Playlist Container 用途广泛，允许以多种方式将段落编组，各 Group（编组）都可以有不同的播放行为，播放模式和类型。播放类型决定了该编组将随机播放还是顺序播放，而播放模式决定了该编组每次播放的段落数量。

下图说明了怎样使用 Music Playlist Container 对工程中的 Music Segment 进行编组。



使用 Music Playlist Container 包括以下任务：

- [在 Music Playlist Container 内新建编组](#)
- [定义 Music Playlist Container 内各编组的行为](#)
- [为 Music Playlist 添加内容](#)

在 Music Playlist Container 内新建编组

通过用 Music Playlist Container（音乐播放列表容器）将 Music Segment（音乐段落）编组，您可以创建复杂的音乐结构。可以建立多个 Group（编组），将其重新排列、删除、剪切、复制和粘贴，再将 Music Segment 添加到编组中。关于为 Music Playlist Container 添加内容的详细信息，请参阅[为 Music Playlist 添加内容](#)。

在 Music Playlist Container 中新建编组的方法如下：

1. 将音乐播放列表容器加载到 Property Editor。

顶层组会自动在音乐播放列表编辑器中加以创建。

2. 要在播放列表层级结构中添加新编组，请点击 New Group（新编组）。

层级结构中所选父对象下面将添加新的编组。



备注

要从播放列表中删除某个编组，请右键点击并选择 Delete（删除），或选择编组并按 Delete 键。整个编组的全部内容都会从播放列表中删除。

3. 继续向播放列表中按需添加编组。

您可以随时为播放列表内的各编组改变其行为。

相关主题

- 定义 Music Playlist Container 内各编组的行为
- 为 Music Playlist 添加内容
- 定义 Music Playlist Container 的播放行为

定义 Music Playlist Container 内各编组的行为

对于 Music Playlist Container（音乐播放列表容器）内的各 Group（编组），可以分别设置两种播放模式和播放类型。播放类型决定组是以随机方式还是以顺序方式播放。播放模式决定每次播放组时将会播放组的段落数量。



备注

您选择的播放模式和类型将决定各编组有哪些行为选项可用。

在 Music Playlist Container 中定义 Random Group（随机编组）的方法如下：

1. 将 Music Playlist Container 加载到 Property Editor（属性编辑器）。

在 Music Playlist Editor 中，会自动创建顶层编组。

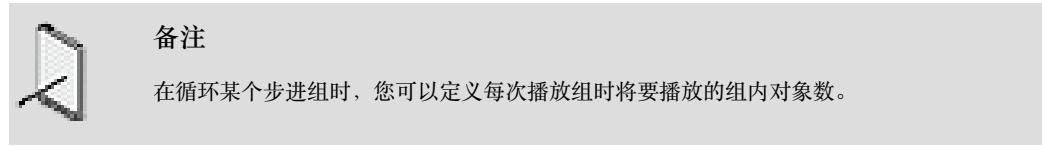
2. 在 Group/Segment（编组/段落）列表中，为顶层编组选择下列播放行为之一：

- Random Continuous -- 随机连续。每次播放编组时，将以随机方式逐个播放其中所有的音乐对象。
- Random Step -- 随机步进。每次播放编组时，仅随机播放其中的一个音乐对象。

3. 从 Random Type（随机类型）列表中，选择以下选项之一：

- Standard -- 标准模式，始终保持对象池完整。一个音乐对象播放后，该对象将不会从对象列表中删除，因此可能重复播放。

- Shuffle -- 洗牌模式，音乐对象播放后，会从对象池中删除。一个音乐对象播放后，会等到编组中所有其他音乐对象都已播放完毕，才可能重复播放该音乐对象。
4. 可以在 Avoid Repeat (避免重复) 文本框中键入数量，这样一个音乐对象播放后，将先播放该数量的其他音乐对象，之后才可能重复播放该音乐对象。
 5. 在 Loop Count (循环次数) 文本框中，指定整个组将播放的次数或将要播放的步进数。



在 Music Playlist Container 中定义 Sequence Group (序列编组) 的方法如下：

1. 将音乐播放列表容器加载到 Property Editor。

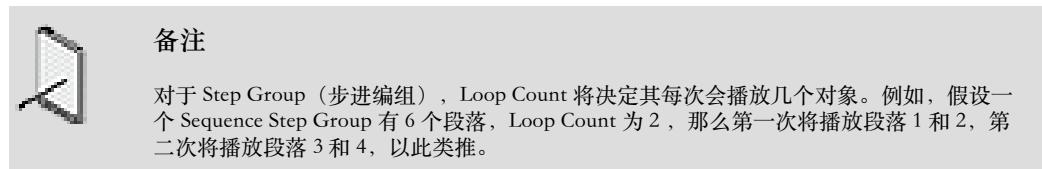
在 Music Playlist Editor 中，会自动创建顶层编组。

2. 从 Group/Segment 列表中，选择以下选项之一来定义顶层组的播放行为：

Sequence Continuous -- 顺序连续。每次播放编组时，将顺序播放其中的所有音乐对象。

Sequence Step -- 顺序步进。每次播放编组时，仅播放其中的一个音乐对象，而下次将播放下一个音乐对象。

3. 在 Loop Count 文本框中，指定整个编组的循环播放的次数（或步进数）。



相关主题

- [在 Music Playlist Container 内新建编组](#)
- [为 Music Playlist 添加内容](#)
- [定义 Music Playlist Container 的播放行为](#)

为 Music Playlist 添加内容

您需要指定 Music Playlist Container (音乐播放列表容器) 中哪些段落将被添加到播放列表，以及按什么顺序播放，就像 Actor-Mixer Hierarchy (角色混音器层级) 中的 Sequence Container (序列容器) 一样。每个段落必须属于一个 Group (编组)，它定义了其中各段落的播放行为。关于定义编组播放行为的详细信息，请参阅[定义 Music Playlist Container 内各编组的行为](#)。

为 Music Playlist Container 添加内容的方法如下：

1. 将音乐播放列表容器加载到 Property Editor。
2. 在 Content Editor (内容编辑器) 中，将若干段落拖拽到音乐播放列表内的编组中。

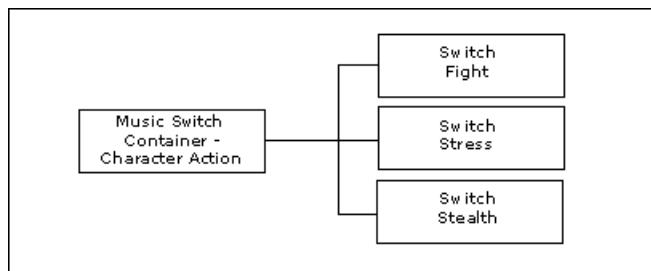
所选段落将被添加到编组中。
3. 对于 Random Step (随机步进) 或 Random Continuous (随机连续) 编组，可以在 Weight (权重) 文本框中为各对象设置权重。权重可以让某些音乐对象的优先级高于其它对象。
4. 在 Loop Count (循环次数) 文本框中，指定各段落将播放的次数。
5. 继续将段落拖拽到 Music Playlist Container 内的各编组中。

相关主题

- [在 Music Playlist Container 内新建编组](#)
- [定义 Music Playlist Container 内各编组的行为](#)
- [定义 Music Playlist Container 的播放行为](#)

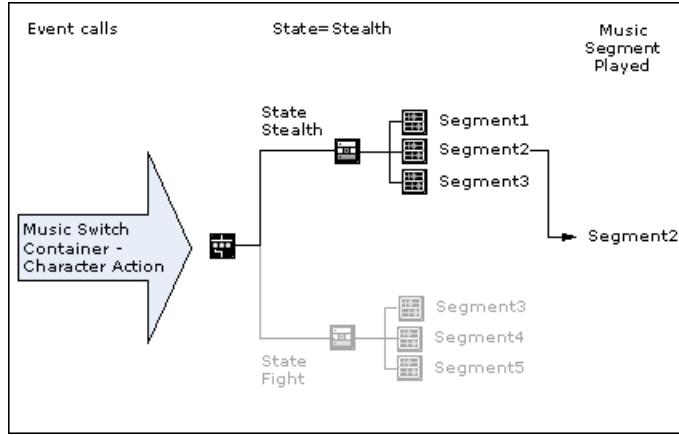
定义 Music Switch Container 的内容和行为

当游戏中的特定条件具有多种不同选项，可以用 Music Switch Container (音乐切换容器) 根据这些条件在几组音乐之间切换。在 Switch Container (切换容器) 中，Switch (切换开关) 或 State (状态) 代表着各种不同条件。例如，游戏中主角采用不同的行动模式时，可以针对性地创建 Music Switch Container，其中包含战斗场景、紧张情况以及潜行模式的 Switch。



各 Switch/State 对应着特定条件选项下的音乐对象。本例中，所有游戏战斗相关的 Music Segment 将对应 Switch Fight，与紧张情况相关的 Music Segment 将对应 Switch Stress，以此类推。游戏调用切换容器时，Wwise 会验证哪个 Switch/State 处于激活状态，进而确定播放哪个 Container 或 Music Segment。

下图表示当事件调用了“Character Action”(角色行动模式)这个 Music Switch Container 时，会发生什么。该容器中，已根据游戏中可能的角色动作类型将音乐分成了几组，有两种状态：潜行 (Stealth) 和战斗 (Fight)。事件调用 Music Switch Container 时，角色正处于潜行模式 (State=Stealth)，因此会播放潜行模式音乐。潜行状态内的音乐使用 Music Playlist Container 编组，这样每次都可以听到潜行主题的不同变奏。



将 Music Switch Container 关联至 Game Sync

Music Switch Container（音乐切换容器）可以基于 State（状态）、Switch（切换开关）或 Game Parameter（游戏参数）。要将 Music Switch Container 关联到 Game Parameter，需要先在 Switch Group Property Editor（切换开关组属性编辑器）中将 Switch Group（切换开关组）关联到所需的 Game Parameter。关于 RTPC（实时参数控制）如何与 Switch 进行关联的详细信息，请参阅[将 Game Parameter 值映射到 Switch](#)。

设置 Music Switch Container 时，首先需要将 Switch Group 或 State Group（状态组）指派给容器，这决定了音乐将对哪些 Switch、State 或 RTPC 做出反应。

为容器指派 Switch\State Group，或将 RTPC 关联至 Switch之前，必须先创建它们。关于创建 Switch Group、State Group 和 RTPC 的详细信息，请参阅：

- [第 17 章 使用 State](#)
- [第 18 章 使用 Switch](#)
- [创建 Game Parameter](#)

将 Game Sync 与 Music Switch Container 关联的方法如下：

1. 将音乐切换容器加载到音乐编辑器。
2. 执行以下操作之一：
 - 点击编辑器顶部的 [>>] 按钮，并选择 Game Sync（游戏同步器）。
 - 将 Switch Group 或 State Group 对象从 Project Explorer（工程浏览器）拖拽至编辑器顶部。

相关主题

- [定义 Music Switch Container 的内容和行为](#)
- [将音乐对象与 State 或 Switch 相关联](#)
- [定义音乐对象的 Time Settings](#)

定义 Music Switch Container 内音乐对象的行为

如果您复用了部分音乐，那么 Switch 或 State 就可能包含几个相同的音乐对象。这种情况下，需要确定 State 改变时这些共用音乐对象如何播放。视具体情况而异，可能需要音乐对象继续播放，或在下一个同步点停止并从头播放。

下图中，Stealth（潜行）和 Stress（紧张）两个 Switch 关联了同一播放列表编组（S2_Playlist）。游戏从潜行切换至紧张模式时（反之亦可），可以选择在下一同步点继续播放 S2_Playlist，或停止当前音乐段落并从播放列表开头播放。

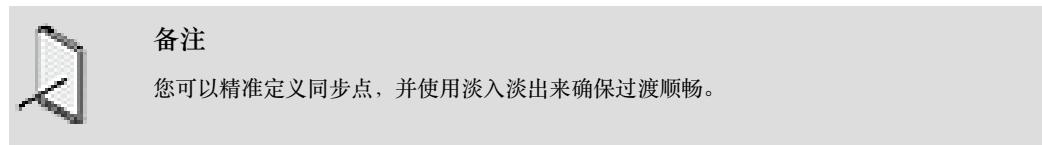
Switch/State	Child
<input type="checkbox"/> None	
<input type="checkbox"/> Fight	F_Playlist
<input type="checkbox"/> Stealth	S2_Playlist
<input type="checkbox"/> Stress	S2_Playlist

定义 Music Switch Container 内音乐对象播放行为的方法如下：

1. 将 Music Switch Container（音乐切换容器）加载到 Property Editor（属性编辑器）。
2. 在 Play Options（播放选项）部分，执行以下操作之一：

如果希望 Switch/State 改变时，源和目标共用的音乐对象继续播放，请选择 Continue to play on Switch change（在 Switch 更改时继续播放）选项。

如果希望 Switch/State 改变时，源和目标共用的音乐对象在下一个同步点停止并从头播放，则不应勾选 Continue to play on Switch change 选项。



相关主题

- [定义 Music Switch Container 的内容和行为](#)
- [将 Music Switch Container 关联至 Game Sync](#)
- [将音乐对象与 State 或 Switch 相关联](#)
- [定义音乐对象的 Time Settings](#)

将音乐对象与 State 或 Switch 相关联

将 State Group（状态组）或 Switch Group（切换开关组）指派给 Music Switch Container（音乐切换容器）后，即可将不同音乐对象关联至相应的 State（状态）或 Switch（切换开关）组合。

每个特定的 State 或 Switch 组合称为一个 Path（路径），Path 中还可能包括通配符。Music Switch Container 内的音乐对象将会与 Path 进行关联。

指派音乐对象的方法如下：

1. 将音乐切换容器加载到音乐编辑器。

此容器的状态组和切换开关组会显示在音乐切换容器的顶部区域中。

2. 在顶部每个 State Group 或 Switch Group 中选择一个状态。
3. 点击 Add Path（添加路径）按钮，在列表中创建条目。

条目将显示在列表中，但尚未指派对象。

4. 点击条目上的 [...] 按钮，为条目指派音乐对象。

对象浏览器将弹出，并显示 Music Switch Container 的子对象。

5. 选中一个对象，点击 OK。

使用拖拽方式指派音乐对象的方法如下：

1. 将 Music Switch Container 加载到 Music Editor。

此容器内的 State Group 和 Switch Group 会显示在 Music Switch Container 的顶部。

2. 将 Music Switch Container 的子对象拖拽到顶部的 Switch 或 State 上，即可关联该对象。

相关主题

- [定义 Music Switch Container 的内容和行为](#)
- [将 Music Switch Container 关联至 Game Sync](#)
- [定义 Music Switch Container 内音乐对象的行为](#)
- [定义音乐对象的 Time Settings](#)

互动音乐播放的技巧和经验总结

互动音乐是一套复杂工具，具有多种选项，项目开始时就为其制定统一的策略可以节省后期的时间和精力。当然，处理互动音乐的方法有很多，您可以在 Wwise 中使用任何方法来为游戏创造最佳音乐效果。关于如何充分发挥互动音乐的优势，以下是一些建议，包括针对可能出现的播放问题如何进行故障排查。

排查播放故障

连接到游戏并试听时，如果遇到不同步或 Voice Starvation 问题，最好在 Track Property Editor 中为音乐轨重新设置 Look-ahead Time（预读时间）。Wwise 创作工具中，如果未采用流播放的音轨遇到了上述问题，可以在 Audio Preferences 中调整 Look-ahead Time，即 Wwise 中所有非流播放对象的默认预读时间。关于如何在 User Preferences 中设置 Look-ahead Time 的详细信息，请参阅[设置音乐轨预读时间](#)。

一般来说，同时播放的音轨越多，所需预读时间也应越长，这样才能避免 Voice Starvation 并保持音轨同步。Look-ahead Time 具体需要多长取决于多种因素，包括请求预读时的压缩格式和总带宽占用。

确保 Sample Accurate 转码

对于以下频率的声音源，Wwise 中使用的采样率转换器会略微增加声音长度，每分钟大约增加 12 个采样。

- 44,100 Hz
- 22,050 Hz
- 11,025 Hz

虽然对于声音对象不成问题，但音乐对象可以在任何 Exit（出口）提示点停止，不一定播放到文件结束，从而可能导致连续性问题。因此，音乐对象最好使用 12,000、24,000、36,000 或 48000Hz 音频文件，这样不会引入错误。

第 25 章 使用 Music Track 和 Music Segment

概述	443
将 Music Track 添加至 Music Segment	444
定义 Music Track 的播放行为	445
将子音轨添加至音轨	447
将 Sub-Track 关联至 Switch/State	448
添加音轨	449
从段落中移除音轨和子音轨	449
在 Music Editor 中按间距对齐	450
使用片段	450
预览 Music Segment	453
使用 Cue	454

概述

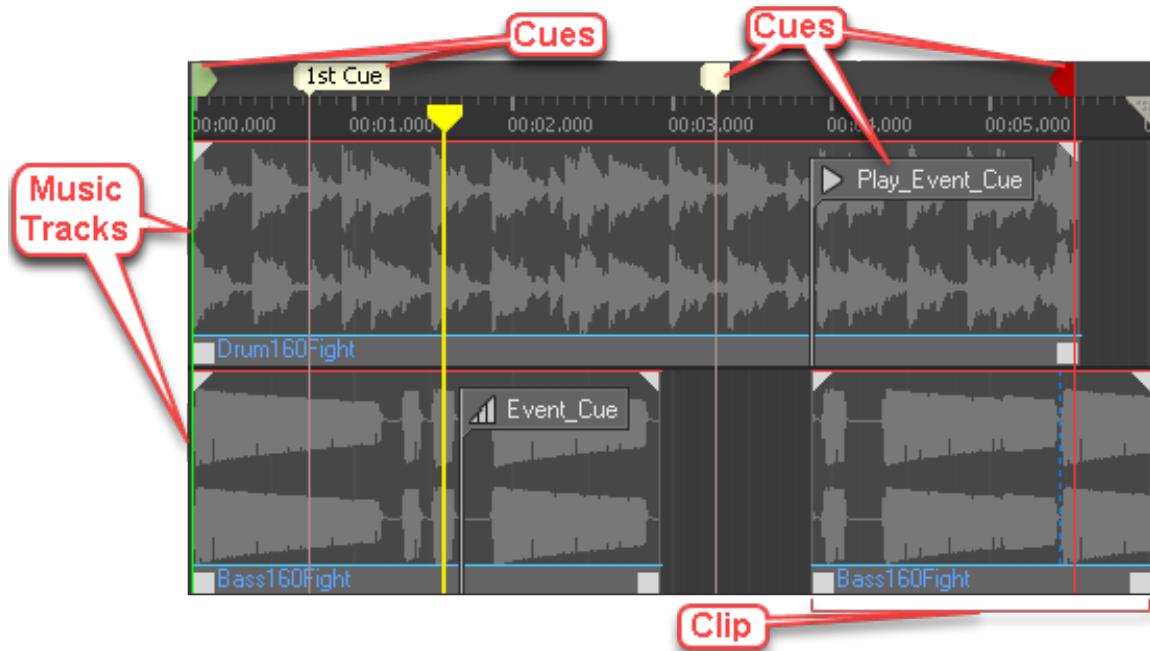
音乐段落（Music Segment）是 Interactive Music 层级结构的基本单位。它与多轨音序器类似，用于排列音乐轨和子音乐轨，为游戏创建多声部配乐。音乐轨（Music Track）可包含无限数量的子音乐轨，各条子音乐轨可包含不同的音乐片段。通过为各条音乐轨指定不同的播放行为，可以为游戏中的音乐增加变化。

为了帮助您轻松识别界面中音乐段落和音乐轨，Wwise 使用特定的图标来标识它们。

图标	代表
	Music Track —— 音乐轨。每次播放其父段落时播放该音乐轨。
	Random Music Track —— 随机音乐轨。每次在播放其父段落时，按随机顺序播放其子音乐轨。
	Sequence Music Track —— 序列音乐轨。每次在播放其父段落时，按序播放其子音乐轨。
	Switch Music Track —— 切换开关音乐轨。根据相关联的切换开关组播放其子音乐轨。
	Music Segment —— 音乐段落。音乐轨的父对象。

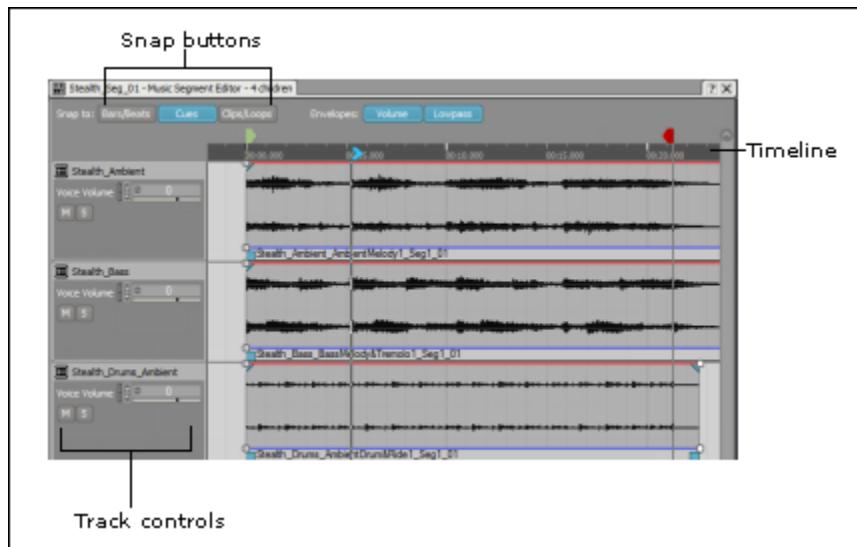
指定段落中的所有音乐轨都显示在 Music Segment Editor 中。音乐轨包含音乐片段，可以对这些音乐片段进行查看和编辑。这些片段代表音频源，并以波形方式在 Music Segment Editor 时间线上显示。您可以编辑这些这些片段，并在时间线上的特定点给这些片段添加 cue（提示点）。您还可以创建自定义提示点（Custom Cue），以指示何时改变属性变化或进行过渡，或何时播放插播乐句（Stinger）。

其中一种最好用的提示便是 Event（事件）提示。利用这种提示，可将 Event 快速添加到 Music Segment 时间线中的特定时间点。



Music Segment Editor 包含三个区域：

- Snap 按钮部分，用于将片段和提示点对齐至所选的元素。
- 音轨控制部分，用于编辑音轨的属性设置，以及单条轨道的静音和强制播放。
- 段落时间线部分，用于查看片段和提示点。



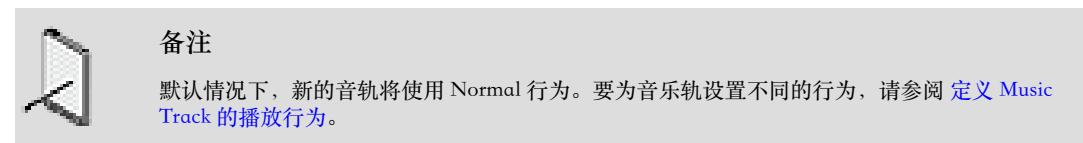
使用段落包含以下任务：

- 将 Music Track 添加至 Muisc Segment
- 定义 Music Track 的播放行为
- 将子音轨添加至音轨
- 添加音轨
- 从段落中移除音轨和子音轨
- 预览 Music Segment
- 分割片段
- 使用 Cue

将 Music Track 添加至 Muisc Segment

您可以从 Wwise 中的以下位置将音乐轨（Music Track）添加至段落（Muisc Segment）：

- Project Explorer 的 Audio 选项卡
- Music Segment Editor（音乐段落编辑器）



从 Audio 选项卡添加新音乐轨的方法如下：

1. 在 Interactive Music 层级结构中，右键点击要添加音乐轨的段落，然后选择 New Child > Music Track。

新音乐轨被添加到段落中，并显示在 Music Segment Editor 里。

- 将默认名称替换为更适合音乐轨的名称。

在 Music Segment Editor 中添加新音乐轨的方法如下：

- 将段落加载至 Music Segment Editor。
- 在 Music Segment Editor 中，右键点击以打开快捷菜单，然后选择 New Track。

New Music Track 对话框将打开。

- 将默认名称替换为更适合音乐轨的名称，然后点击 OK。

新音乐轨将被添加至 Music Segment Editor。

相关主题

- [添加音轨](#)
- [定义音轨播放类型](#)
- [从段落中移除音轨和子音轨](#)

定义 Music Track 的播放行为

工程内的每条音乐轨都包含一组特定行为。这些行为决定了音乐轨及其子音乐轨的播放方式，还决定音轨中的音乐是存储在内存中，还是直接从媒体或硬盘中进行流播放。

您可为音乐轨定义以下行为：

- [流播放音乐](#)
- [定义音轨播放类型](#)

流播放音乐

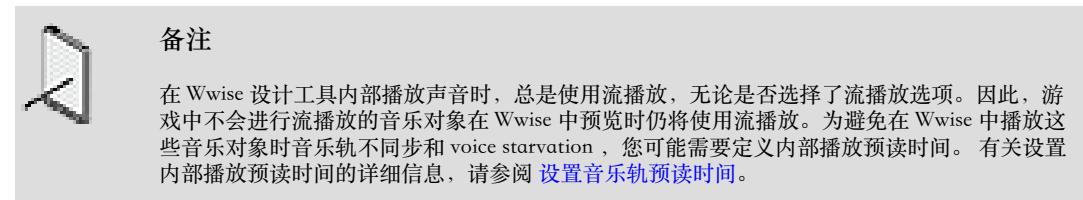
您可以决定从内存播放哪些音乐，以及从 DVD、CD 或硬盘流播放哪些音乐。当从磁盘或硬盘流播放音乐时，还可以结合使用以下选项，来避免出现播放延迟和同步问题。

- Look-ahead time -- 预读时间。保留特定时间量，使声音引擎能够查找流播放数据。此时间确定该轨的延迟。
- Prefetch -- 预取。创建较小的音频缓冲区来预先播放音频文件的开头，同时读取文件的剩余部分，以补偿直接缓冲播放整个文件会造成的延迟。

您可以指定预读时间量和预取量的大小，以满足不同媒体源（如硬盘、CD 和 DVD）的要求。预取过小将导致延迟，过大则会占用过多内存。预读时间过小将导致段落内的音乐不能严格同步，而预读时间过大将导致整体播放延迟。您需要对预读大小和预读时间进行试验，以在音乐片段的整体延迟和同步之间找到平衡。

预读时间和预取设置可以结合使用，以保证在游戏中使用流播放音乐时，降低不同步和 voice starvation 的几率。例如，当一段音乐的同步点不在段落的开始处，或者没有

使用段落的 Pre-entry（前导段）部分，那么加载至内存中的预取数据就无法使用，因此 Wwise 将使用预读时间。

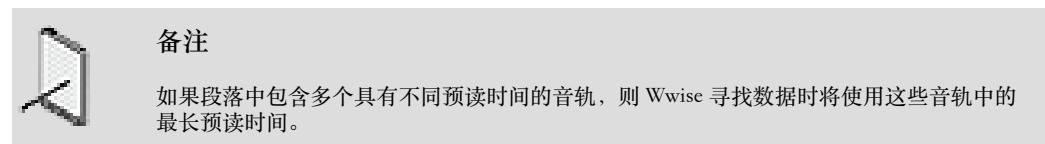


流播放音乐轨的方法如下：

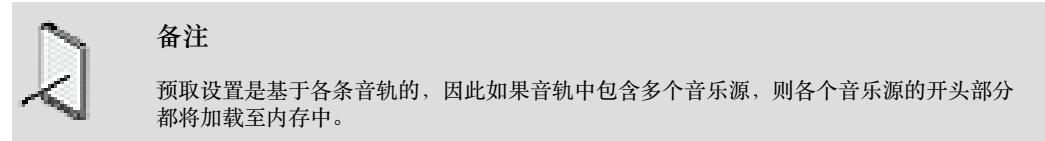
1. 双击音乐轨，将其加载至属性编辑器中。
2. 选择 Stream 选项。

流播放设置将变为可用。

3. 在预读时间文本框内，输入要预留的毫秒数，来允许声音引擎查找流播放数据。该时间定义了音轨的延迟量，因此如果您将其设置为 100 毫秒，则会在按下 Play 按钮 100 毫秒后才听到音乐。



4. 要实现从触发音乐到实际播放之间没有延迟，可以选择 Zero Latency 选项。
5. 要达到零延迟，必须将声音开头的特定部分存储在内存中，同时不断读取文件的剩余部分，以补偿直接缓冲播放整个文件会造成的延迟。在 Prefetch length 文本框中，输入要存储在内存中的音乐的毫秒数。



相关主题

- [定义音轨播放类型](#)
- [添加音轨](#)

定义音轨播放类型

要为游戏中的音乐增加变化，您可以为段落内的各条音轨定义播放类型。您可使用 normal（常规）音轨，即每次调用段落时只播放同一音轨，也可创建 random（随机）或 sequence（序列）音轨。Random Step（随机步进）和 Sequence Step（序列步进）音轨可包含无限数量的子音轨，用来在每次播放该音轨时播放不同的音乐。

下图显示了三种音轨类型之间播放行为的差别。

Track Type	Segment A	Game calls Segment A (1st time)	Game calls Segment A (2nd time)	Game calls Segment A (3rd time)
Normal	Track1	Wwise Plays ↓	Wwise Plays ↓	Wwise Plays ↓
Random Step	Track1 Sub-track1 Sub-track2 Sub-track3	Sub-track3	Sub-track1	Sub-track2
Sequence Step	Track1 Sub-track1 Sub-track2 Sub-track3	Sub-track1	Sub-track2	Sub-track3

结合使用这三类音轨，可以让游戏每次调用段落时都感觉像是在播放全新的音乐。例如，可以创建包含四个随机步进音轨的音乐段落，各条音轨包含四个子音轨。当游戏调用该音乐段落时，这段音乐就可以拥有 256 种不同的变化。

定义音轨播放类型的方法如下：

- 双击音乐轨，将其加载至属性编辑器中。
- 从 Track Type 下拉菜单中，选择以下任一选项：
 - Normal，每次播放父段落时即播放当前音轨。处于 Normal 模式时，无法添加子声轨。
 - Random Step，每次播放父段落时，随机播放其中一个子音轨。
 - Sequence Step，每次播放父段落时，按顺序播放其中一个子音轨。

Switch，选择与切换开关相对应的子音轨进行播放。

相关主题

- [定义 Music Track 的播放行为](#)
- [流播放音乐](#)
- [将 Sub-Track 关联至 Switch/State](#)

将子音轨添加至音轨

要为配乐增加多样性，您可以向音轨添加分层或子音轨（Sub-Track）。由于可以为音轨和子音轨定义随机或序列行为，因此播放时可以实现近乎无限数量的排列组合。

添加子音轨这一功能仅适用于被定义为 Random Step、Sequence Step 或 Switch 的音轨。有关如何使用这些上述行为和音轨的详细信息，请参阅 [定义 Music Track 的播放行为](#)。创建子音轨后，您可以单独预览部分子音轨。当仅需在 Wwise 设计工具中进行测试时，可以使用 Force Usage 功能来取消沿用音轨的随机或序列行为。有关预览单个子音轨的详细信息，请参阅 [强制播放子音轨](#)。

将子音轨添加至音轨的方法如下：

1. 右键点击要添加子音轨的音轨。
2. 在快捷菜单中，选择 Add Sub-Track。

新的子音轨将被添加至音轨。

相关主题

- [添加音轨](#)
- [定义音轨播放类型](#)
- [将 Music Track 添加至 Muisic Segment](#)
- [将 Sub-Track 关联至 Switch/State](#)

将 Sub-Track 关联至 Switch/State

为了增加配乐的多样性，可以通过切换开关组（Switch Group）/状态组（State Group）来决定播放哪些子音轨。为此，必须将 Track Type 设置为 Switch。有关如何使用这些上述行为和音轨的详细信息，请参阅 [定义 Music Track 的播放行为](#)。

要将子音轨关联至切换开关/状态，必须先为音轨选择切换开关组/状态组。通过在音轨的 General Settings 选项卡的 Switch Type 框中的控件，可以选择音轨的切换开关组/状态组。

通过 Transitions（过渡）选项卡设置音轨随切换开关/状态改变而发生的变化行为。有关如何使用过渡的详细信息，请参阅 [第 28 章 使用 Transition](#)。

将子音轨关联至切换开关/状态的方法如下：

1. 右键点击子音轨的标题栏（黄色栏）。
2. 在快捷菜单中选择 Association。
3. 选择现有切换开关/状态，或新建一个切换开关/状态。

子音轨将被关联至该切换开关/状态。

切换开关/状态名称将显示在子音轨顶部。

相关主题

- [添加音轨](#)
- [定义音轨播放类型](#)
- [将 Music Track 添加至 Muisic Segment](#)

添加音轨

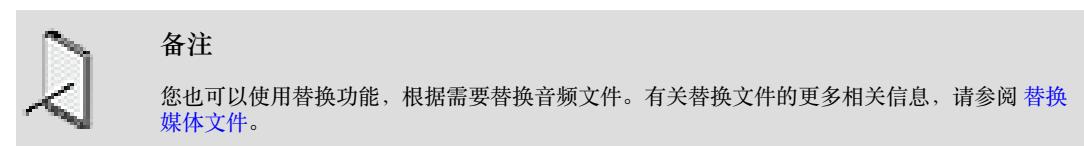
您在互动音乐工程中所做的大部分工作都是基于音轨层级，可以添加音乐片段至音轨，然后根据需要对这些片段进行排列、编辑和混音。

要为音乐轨添加子音轨，您可以使用 Audio File Importer 将音乐文件导入至音乐轨中，跟在 Wwise 中导入音频文件一样。有关导入的更多相关信息，请参阅 [导入媒体文件](#)。

快速导入

下表列出了将音频文件导入至音乐轨的快捷方式。

操作	使用此快捷方式
导入音乐文件。	将文件拖至 Project Explorer 中的 Music Track 内。



相关主题

- [从段落中移除音轨和子音轨](#)
- [定义音轨播放类型](#)
- [将 Music Track 添加至 Music Segment](#)

从段落中移除音轨和子音轨

当您不再需要音轨或子音轨时，可以从段落中移除它们。删除音轨会同时删除其所有子音轨。

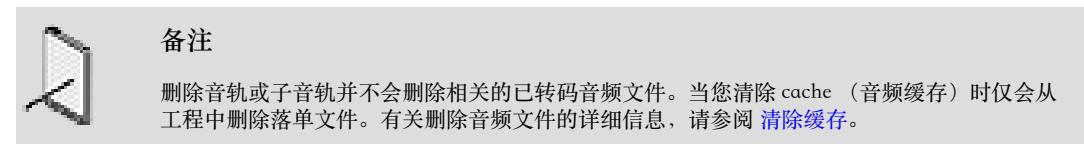
从音乐段落移除音轨或子音轨的方法如下：

1. 在 Music Segment Editor 中，右键点击要移除的音轨或子音轨。



2. 在快捷菜单中，选择 **Delete**。

所选音轨及其子音轨将从段落中删除。



相关主题

- [添加音轨](#)
- [定义音轨播放类型](#)
- [将 Music Track 添加至 Muisic Segment](#)

在 Music Editor 中按间距对齐

为了更准确地沿时间线放置片段、提示点，或播放指针，您可以选择按定义的间距对齐它们。

沿时间线对齐的方法如下：

1. 要将播放指针、片段或提示点沿时间线与以下项对齐，请点击相应的按钮：

Bars/Beats (栏/节奏)

Cues (提示点)

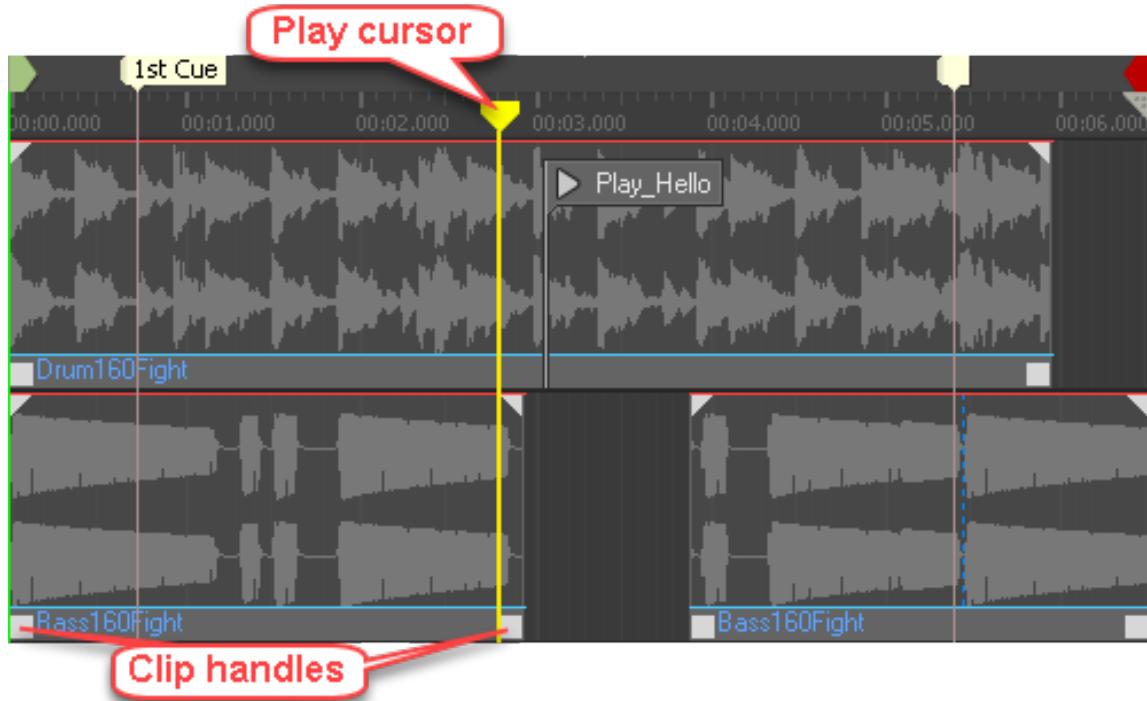
Clips/Loops (片段/循环)

相关主题

- [使用 Cue](#)
- [使用片段](#)

使用片段

通过在 Music Segment Editor 中使用针对片段（Clip）的各种工具，您可以轻松地在时间线上编辑片段。通过上下拖动片段，您可将其从一条音轨移至另一条音轨，也可以将一个片段与另一个片段重叠。向内拖动片段的其中一个手柄可缩短片段，向外拖动手柄可延展片段。在延展片段时，该片段将重复其自身。每次重复被称作一次循环。



下表列出了在 Music Segment Editor 中使用片段时的各种片段工具和标志。

图标	名称	描述
	片段裁剪点	拖动裁剪点可缩短或延展片段。延展片段时，您将创建循环。
	播放指针	播放光标。播放或暂停段落时，指示段落中的当前播放位置。 当段落停止时，指点开始播放的点。 备注 如果启用了 Initial Delay，则不会遵循 Source Editor 的播放光标起始位置。有关详细信息，请参阅 Source Editor 文档。
	结束指针	拖动它可以缩短或延展 Music Segment Editor 中段落的所有音乐轨。缩短或延展音乐轨不会影响这些声轨内的片段长度或播放。
	片段边缘标志	循环播放片段时，标识片段的开始。

循环片段

在 Music Segment Editor 中拖动片段裁剪点可轻松循环片段。

在 Music Segment Editor 中循环片段的方法如下：

1. 点击片段裁剪点，并拖动以延展片段。
2. 延展片段，直至片段边缘标志显示在时间线上。

一个片段循环就创建好了。您可以继续延展该片段，以根据需要添加多个循环。

移动片段

为帮助分层和排列音乐，您可以在音轨内移动片段。可以在 Music Segment Editor 中，以任意方式排列和重叠片段，对音乐进行编排混合。使用 Snap 功能，可以准确沿时间线间隔对齐片段。

移动片段的方法如下：

1. 在音轨中选择一个片段，并将其拖至新位置。



技巧

您也可以使用标准 Windows 剪切、复制和粘贴命令移动片段和整条音轨。

分割片段

可以在 Music Segment Editor 中分割片段。例如，当要切分单个低音鼓点时该功能十分实用，这样就可以在每小节中重复它，也可将其插入另一条音轨。您可以将切分出来的低音鼓片段添加至音轨的任意位置，以重新排列甚至重新创作您的音乐。

分割片段的方法如下：

1. 将播放指针拖至要分割片段的位置。
2. 在音轨中，右键点击要分割的片段，然后从快捷菜单中选择 Split on Play Cursor。

片段将被分割，分割点位于播放指针位置。现在可以根据需要编辑这些片段。

堆叠片段

重叠片段时，更改片段的堆叠方式十分实用，特别是当要查看已重叠片段的同步点时。可以通过快捷菜单中的 Bring to Front 和 Send to Back 命令实现。

更改片段的堆叠顺序的方法如下：

1. 在音乐轨中，右键点击上方的重叠片段，然后选择 Send to Back。

这些片段的顺序会颠倒，原来处于下方的片段现在处于上方。



备注

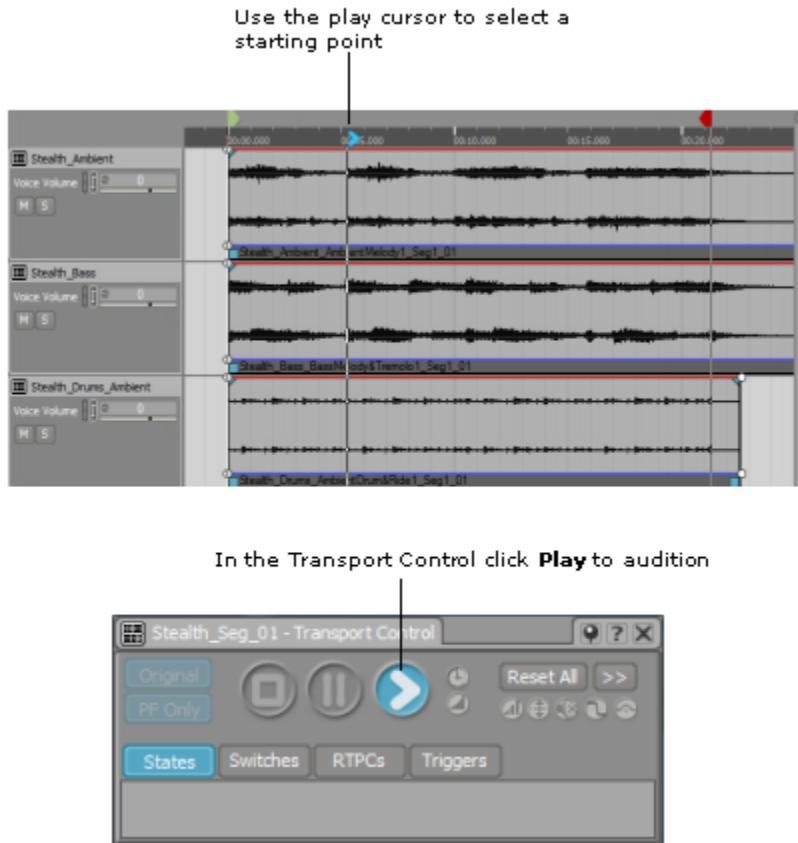
您也可以右键点击下方片段的可见部分，然后选择 Send to Front 以更改片段的堆叠顺序。

相关主题

- [使用片段](#)
- [分割片段](#)
- [移动片段](#)
- [循环片段](#)

预览 Music Segment

在段落 (Music Segment) 中创建好子音轨后，可以随时在 Transport Control 试听段落音轨。要播放片段中的特定点，您可使用播放指针，以指出要从哪里开始播放。



预览段落时，您可以将所有音轨作为一个整体来预览，也可静音或 Solo 单条音轨，以根据需要预览各条音轨的内容。您也可以仅强制播放一个子音轨。

在 Transport Control 中试听音轨的方法如下：

1. 将音乐段落加载至 Music Segment Editor。

音乐段落中的所有音轨和子音轨都将被加载至 Segment Editor 中，音乐段落也将被加载至 Transport Control 中。



2. 如果要预览整个段落，则请点击 Transport Control 中的 Play。
3. 要静音或 Solo 某条音乐轨，请点击音轨控制区域中的 Mute 或 Solo 按钮。
4. 如果要从音轨时间线上的某个位置开始播放，则请将播放指针拖至要开始播放的位置。

相关主题

- [使用片段](#)
- [将 Music Track 添加至 Muisic Segment](#)
- [将子音轨添加至音轨](#)
- [强制播放子音轨](#)

强制播放子音轨

当播放随机或序列音轨时，将以随机或序列顺序播放子音轨。如果您要对特定子音轨中的片段进行实验，则对于预览变化十分有用的方法是不沿用音轨的随机或序列行为，并强制播放该子音轨。

强制播放子音轨的方法如下：

1. 

在 Music Segment Editor 中，点击 Force Usage 图标，强制在 Transport Control 中播放该子音轨。

图标会变为蓝色，所选的子音轨将在 Transport Control 中播放。

2. 要关闭该选项，请再次点击 Force Usage 图标。



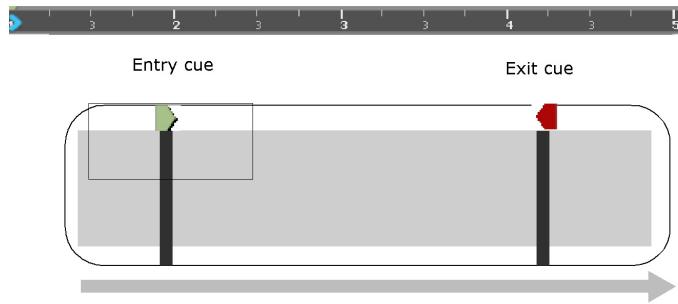
相关主题

- [使用片段](#)
- [将 Music Track 添加至 Muisic Segment](#)
- [将子音轨添加至音轨](#)

使用 Cue

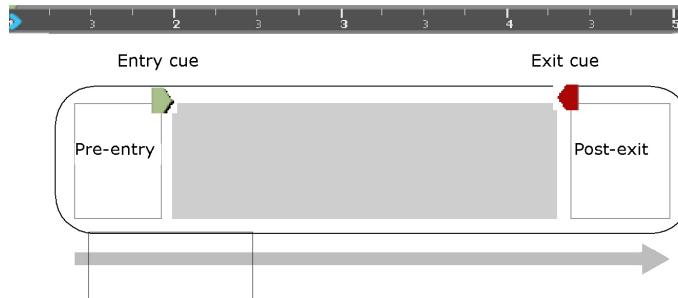
提示点 (Cue) 代表时间线上的特定点。提示点主要用作同步点，用于针对在互动音乐中发生各种变化（如状态改变、播放插播乐句和过渡）时对齐音乐段落。这些编排必须考虑曲速和精确到采样点的对齐问题，以及前后衔接相关的注意事项。所有段落均包含入口提示和出口提示，不过也可能包含自定义提示或 Event 提示。

使用 Music Track 和 Music Segment



入口提示点定义了第一个小节的第一个节拍从哪里开始，并根据上一个段落的出口提示点决定从哪里对齐。入口和出口提示点无法从片段中移除，但可以进行编辑。

入口提示点之前的段落区域被称为 Pre-entry（前导段），超出出口提示点的区域被称为 Post-exit（后尾段）。段落的这些部分都可以被播放，但这取决于过渡设置。有关过渡的详细信息，请参阅 [理解 Transition](#)。



除入口和出口提示点外，您还可以添加任意数量的自定义提示点，用来将音乐与游戏内容进行同步。甚至，还可直接将 Event 提示添加到 Music Segment（音乐段落），以便灵活调节游戏内容与音乐之间的同步方式。

使用提示点包含以下任务：

- [添加 Custom Cue](#)
- [移除 Cue](#)
- [命名提示](#)
- [使用 Entry 和 Exit 提示点](#)
- [使用 Event 提示](#)

添加 Custom Cue

您可以在时间线上的任意位置添加自定义提示点（Custom Cue），用于状态更改、过渡或插播乐句插入。

将提示点添加至音轨的方法如下：

1. 在 Music Segment Editor 中，右键点击要添加自定义提示点的时间线标尺刻度。
2. 从快捷菜单中，选择 Add Custom Cue。

新的自定义提示点将被添加到时间线上所选的位置。

3. 将提示点拖至所需位置。
4. 根据需要继续添加自定义提示点。

相关主题

- [移除 Cue](#)
- [命名提示](#)
- [使用 Entry 和 Exit 提示点](#)
- [使用 Event 提示](#)

移除 Cue

您可随时移除自定义提示点。但无法删除入口和出口提示点。

移除自定义提示点的方法如下：

1. 沿音轨时间线，右键点击要移除的自定义提示点。
2. 从快捷菜单中，选择 Delete。

提示点于是从音轨中移除掉了。

相关主题

- [添加 Custom Cue](#)
- [命名提示](#)
- [使用 Entry 和 Exit 提示点](#)
- [使用 Event 提示](#)

命名提示

您可以为提示添加描述性名称。例如当您需要寻找为插播乐句创建的特定提示点时，该功能十分实用。

描述自定义提示点的方法如下：

1. 在音轨的时间线上，右键点击要描述的自定义提示点。
2. 在快捷菜单中，选择 Rename…。

此时将打开 Rename Cue 对话框。

3. 输入最能代表提示的名称。
4. 点击 OK。

相关主题

- [添加 Custom Cue](#)

- 移除 Cue
- 使用 Entry 和 Exit 提示点
- 使用 Event 提示

使用 Entry 和 Exit 提示点

您可以将入口（Entry）和出口（Exit）提示点移至时间线上不同的位置，从而延长或缩短段落的前导段（pre-entry）和后尾段（post-exit）区域。

针对所选段落内的片段来移动入口和出口提示点的方法如下：

1. 选择提示点并沿段落左右拖动。

将入口和出口提示点分别移至所选片段首尾的方法如下：

1. 右键点击要定义入口和出口提示点位置的片段。
2. 从快捷菜单中，选择 Move Entry/Exit Cues to Selection（将入口和出口提示点与所选片段对齐）。

入口和出口提示点将被放置在所选片段的开头和结尾。



技巧

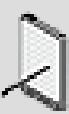
当移动入口提示点时，整个段落将随入口提示点（包括片段和其它提示点）一起移动。当按住 CTRL 键并拖动入口提示点时，只有入口提示点将移动。

相关主题

- 移除 Cue
- 命名提示
- 添加 Custom Cue
- 使用 Event 提示

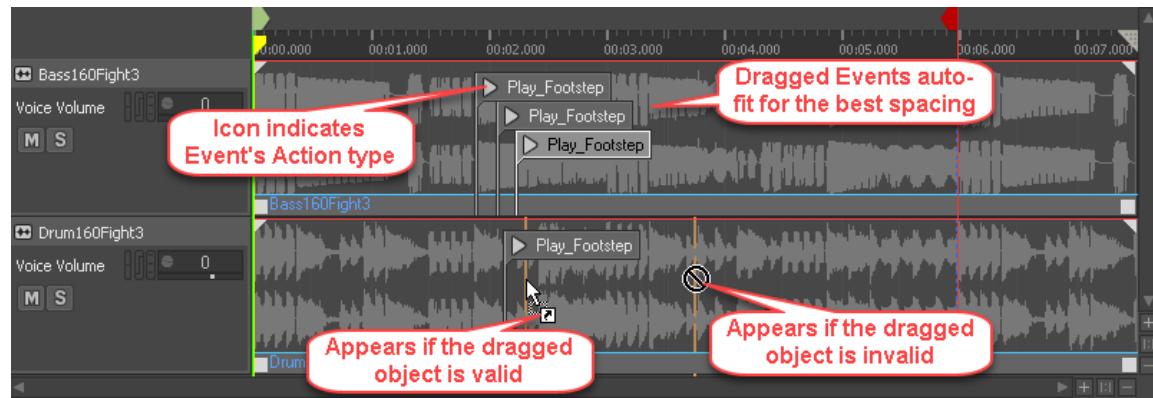
使用 Event 提示

您可以在 Music Segment（音乐段落）时间线的任意位置添加 Action Event（Dialogue Event 不行）。在播放 Music Segment 时，一旦 Play Cursor（播放光标）到达相应 Event 所在位置，便会自动触发该 Event。为此，可从 Project Explorer 的 Events 选项卡拖动 Event，也可[使用快捷菜单](#)。



备注

在触发 Event 后，即便 Music Segment 结束，也可继续播放 Event 中的声音。



将 Event 拖到 Music Segment 中：

1. 转到 Interactive Music (互动音乐) 布局 (依次选择 Layouts > Interactive Music 或按下 F10)。
2. 在 Project Explorer (工程资源管理器) 的 Interactive Music Hierarchy (互动音乐层级结构) 中, 选择或创建 Music Segment 并确保其至少包含一个子级 Music Track (音乐轨), 然后双击该 Music Segment。

此时, 将在 Music Segment Editor (音乐段落编辑器) 中加载该 Music Segment。

3. 在 Project Explorer 的 Events (事件) 选项卡中, 选择或创建 Event, 然后单击该 Event 并将其拖到 Music Segment Editor 时间线中。

若该 Event 有效, 则时间线中将显示箭头标志。若其无效 (比如拖入一个会调用该 Music Segment 的 Event), 则时间线中将显示禁止符标志。

4. 在 Event 有效的情况下, 在相应时间线位置松开鼠标左键。

Event 提示在 Music Track 内看上去像个小旗子。

使用快捷菜单将 Event 提示添加到 Music Segment:

1. 转到 Interactive Music (互动音乐) 布局 (依次选择 Layouts > Interactive Music 或按下 F10)。
2. 在 Project Explorer (工程资源管理器) 的 Interactive Music Hierarchy (互动音乐层级结构) 中, 选择或创建 Music Segment 并确保其至少包含一个子级 Music Track (音乐轨), 然后双击该 Music Segment。

此时, 将在 Music Segment Editor (音乐段落编辑器) 中加载该 Music Segment。

3. 在 Music Segment Editor 中, 在要添加 Event 提示的位置右键单击 Music Track。

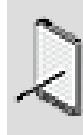
此时将显示快捷菜单。

4. 在快捷菜单中, 选择 Add Music Event Cue... (添加音乐事件提示...)。

此时, Project Explorer Browser (工程资源管理器浏览器) 将显示 Events 选项卡内容。

5. 选择所需 Post Event (发送事件)。

此时将把新的 Event 提示添加到之前在时间线中选择的位置。



更改 Event 提示的 Post Event Target

若 Browser 中没有所需类型的 Post Event，则可另外创建一个。为此，无需返回 Events 选项卡，可通过右键单击 Event 提示来打开快捷菜单，然后单击 Set Post Event Target。此时将弹出子菜单。在此可选择另一现有 Event 或另外新建一个。之后，系统将更新 Event 提示，并将其与新的 Post Event Target 关联。



技巧

在播放 Music Segment 的过程中无法通过快捷菜单添加 Event，即便已经暂停也不行。为此，请务必先停止播放 Music Segment。

Event 提示跟其他 Music Segment 提示一样可以自由拖放到时间线中所需位置。若其刚好与另一 Event 提示重叠，则 Wwise 将自动调节间距以将其插入到适当位置。

相关主题

- [添加 Custom Cue](#)
- [移除 Cue](#)
- [命名提示](#)
- [使用 Entry 和 Exit 提示点](#)

第 26 章 处理 MIDI

创建 MIDI 内容	461
导入 MIDI 文件	461
理解 MIDI 内容和 MIDI 目标	461
混合 MIDI 和音频内容	462
理解 MIDI 速度	463
更改 MIDI 的播放速度	463

创建 MIDI 内容

除 AMB 和 WAV 文件外，MIDI 文件也可用于 Interactive Music Hierarchy 中。这些文件类型一旦导入，可用于创建包含在一个或多个音乐段落轨中的音乐片段。就 WAV 片段而言，各个片段都是独立的声音源。然而，MIDI 片段仅包含要播放的音符；必须使用 MIDI 乐器来解释这些音符。该乐器将解释由指向乐器的若干 MIDI 片段所生成的所有 MIDI 事件。

导入 MIDI 文件

MIDI 文件可导入到 Interactive Music Hierarchy 中的音乐片段内。MIDI 文件不可导入到 Actor-Mixer Hierarchy。

创建 MIDI 片段的方式与 WAV 片段完全相同。有关如何使用文件创建片段的详细信息，请参阅[添加音轨](#)。

使用拖放方式导入 MIDI 文件的方法是：

1. 从 Windows 资源管理器或 Mac Finder 将 MIDI 文件拖到 Interactive Music Hierarchy 中的任何位置。

Audio File Importer 将打开。

2. 点击 Import 以完成导入操作。



使用 Audio File Importer 导入 MIDI 文件的方法是：

1. 在 Project Explorer 中选择要导入 .mid 的位置。
2. 使用键盘快捷键 (Shift + I) 打开 Audio File Importer。
3. 点击 Add Files... 按钮
4. 找到 MIDI 文件，然后点击 OK。
5. 再次点击 OK，以完成导入操作。

理解 MIDI 内容和 MIDI 目标

MIDI 文件引用存储在音乐轨和段落中包含的音乐片段内。MIDI 片段只包含源 MIDI 信息：音符、弯音、CC。它们并不定义用于播放的乐器。用于播放的乐器由音乐段落和其它上层音乐对象中的 MIDI 目标（MIDI target）引用定义。

- 需要在 Actor-Mixer Hierarchy 中创建并存储可用 MIDI 播放的乐器。

- MIDI 可播放乐器使用声音常用的标准对象结构。

例如：

- Actor-Mixer Hierarchy (角色混音器层级结构)
 - Instrument Work Unit
 - Drum Kit
 - Piano
 - Interactive Music Hierarchy
 - Default Work Unit
 - Song Segment
 - Piano Track (MIDI 片段)
 - Drum Track (MIDI 片段)
 - Guitar (音频片段)

在前例中：

- Piano Track 有一个 MIDI 目标引用了 Actor-Mixer Hierarchy 中的 Piano 对象。
- Drum Track 有一个 MIDI 目标引用了 Actor-Mixer Hierarchy 中的 Drum Kit 对象。
- >Guitar Track 没有 MIDI 目标。

在使用 MIDI 片段播放音乐段落时，MIDI 片段将把 MIDI 数据发送到指定的 MIDI 目标。位于 >Actor-Mixer Hierarchy 的实际 MIDI 目标（或乐器）将收到 MIDI 数据，并根据其 MIDI 设置播放。

为音乐对象指定 MIDI 目标的方法是：

1. 选择音乐对象（音乐轨、音乐段落或任何父项）。
2. 在 Property Editor 中，转至 MIDI 选项卡。
3. 在 MIDI Target 组中，点击 Override Parent（如果适用）。
4. 点击 [...] 按钮以浏览 MIDI 目标乐器。

有关详细信息，请参阅[第 27 章 创建 MIDI 乐器](#)。

混合 MIDI 和音频内容

在 Music Hierarchy 中播放 MIDI 内容时，音乐对象将 MIDI 目标（乐器）发送到位于 Actor-Mixer Hierarchy 中的 MIDI 目标（乐器）。播放到 MIDI 目标中的实际音频内容将在 Actor-Mixer Hierarchy 环境而不是 MIDI 源位置环境中播放。这意味着音乐对象的声部属性不会影响正在播放的乐器。

例如，Music Segment 或 Music Track 中的 Voice Volume 不会影响位于 Actor-Mixer Hierarchy 中的钢琴乐器。要修改钢琴的声部音量，则需要直接在钢琴对象上而不是 MIDI 源上修改声部音量。

为了帮助混音，可创建一个总线层级结构，将乐器内容与音乐音频内容组合起来。将乐器和音乐轨连到同一条总线将提供单一的音频混音点。

理解 MIDI 速度

MIDI 片段和 WAV 片段之间有一个重要区别。WAV 片段的时长完全由源文件的内容决定。然而，MIDI 文件的时长取决于速度（tempo）。速度是在创建片段时通过 Tempo Source 属性决定的。

MIDI 文件通常包含速度信息，以保证播放是以正确的速度来进行的。此速度信息可能会与互动音乐对象（段落和父项）中的速度信息相冲突。您可以选择使用源 MIDI 文件的速度或使用 Interactive Music Hierarchy 的速度。

下表列出了可能的速度源：

速度源	定义
文件	使用 MIDI 文件中包含的速度。
层级结构	使用 MIDI 片段的 Interactive Music 上层对象指定的速度。

设置 MIDI 速度源的方法是：

1. 选择音乐对象。
2. 在 Property Editor 中，转至 MIDI 选项卡。
3. 在 MIDI Clip Tempo 组中，选择要使用的 Source。

MIDI 片段的速度在创建后可更改，但不会影响时长。时长可通过片段裁剪点进行更改。有关详细信息，请参阅[使用片段](#)。

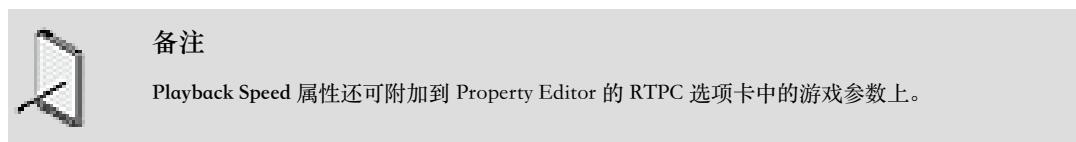
更改 MIDI 的播放速度

在 Interactive Music 层级结构中不可设置 Voice Pitch 属性。但是，可更改音乐层级结构中的播放速度。

更改音乐对象的 Playback Speed 将：

- 影响音频片段的音高。
- 影响 MIDI 片段的播放速率。

播放速度 1 将以原始速度播放。播放速度 2 将以 2 倍的原始速度播放。播放速度 0.5 将以一半的速度播放。



更改播放速度的方法是：

1. 察看音乐对象（音乐轨除外）。
2. 在 Property Editor 中，转至 General Settings（常规设置）选项卡。

3. 设置 Playback Speed。

第 27 章 创建 MIDI 乐器

设计 Synth One 乐器	466
设计简单的采样 MIDI 乐器	466
理解 MIDI 音符跟踪	466
理解 MIDI 筛选器	467
了解 MIDI 事件	467
在 MIDI 事件上添加淡入和淡出	468
使用 MIDI 数据控制对象属性值	468
使用 MIDI Keymap Editor	469
使用 MIDI 键盘测试乐器	469
将 MIDI 从 DAW 连通到 Wwise	470

设计 Synth One 乐器

如果源插件理解 MIDI 消息，则可以通过源插件来创建 MIDI 乐器。

创建源插件乐器（例如：合成器）的方法如下：

1. 在 Actor-Mixer Hierarchy 中创建一个空的声音对象。
2. 在 Project Explorer（工程浏览器）中，选择该声音。
3. 在 Contents Editor 中，点击 Add Source>>。
4. 从选择器菜单中，选择 Synth One。
5. 从 Views（视图）菜单中，选择 Source Editor（源编辑器）（Shift+X）。
6. 点击 Actor-Mixer Hierarchy 中的声音对象，即可在 Source Editor 中看到它。
7. 在 Source Editor 中，将 Frequency Mode（频率模式）设置为 MIDI Note（MIDI 音符）。

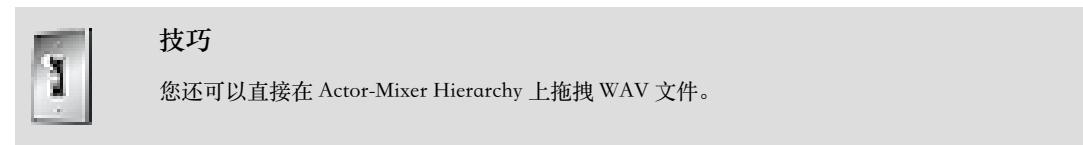
现在，乐器可作为音乐对象内的 MIDI 目标引用（例如：音乐段落）。

设计简单的采样 MIDI 乐器

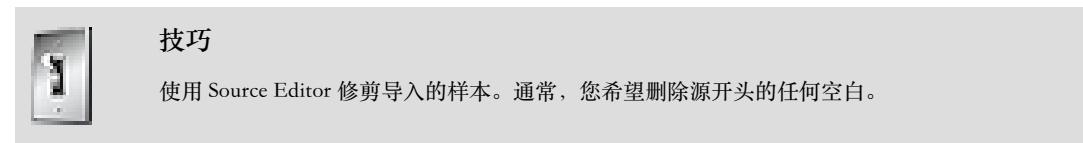
采样乐器的设计中可以使用所有 Actor-Mixer Hierarchy 容器（混合容器、切换容器、随机容器、序列容器和声音）。实现的复杂性将随着乐器设计复杂性而增长。最简单的采样乐器应为单采样点乐器。

创建单采样点乐器的方法如下：

1. 在 Project Explorer 中，选择要创建乐器的位置。
2. 从 Views 菜单中，选择 Audio File Importer（音频文件导入器）（Shift+I）。
3. 点击 Add Files...（添加文件...）
4. 浏览找到 .wav 文件，然后点击 OK（确定）。
5. 再次点击 OK，以完成导入操作。



现在，乐器可作为音乐对象内的 MIDI 目标引用（例如：音乐段落）。



理解 MIDI 音符跟踪

MIDI 音符跟踪参数可以在 Actor-Mixer Hierarchy 对象的 Property Editor 的 MIDI 选项卡>中找到。这些参数决定在播放期间是否在收到 MIDI 消息时改变声音对象的音

高。如果播放的声音要改变音高，则它根据的是 MIDI 消息的音符，以及由角色混音器对象的源代表的音符；它是根音。

音符跟踪参数可在任何乐器的 Actor-Mixer Hierarchy 对象中进行指定或选择不沿用父项。有关 Actor-Mixer Hierarchy 中属性的详细信息，请参阅[工程层级结构中的属性介绍](#)。MIDI 音符跟踪参数包括：

- **Override parent**（不沿用父项）：如果勾选，则对象将忽略其上级对象的音符跟踪参数。
- **Enable**（启用）：如果设置该选项，则对象的声音在播放时将会改变音高。音高平移量由接收到的MIDI事件的音符与 Root Note 值决定。
- **Root note**：由对象的源所代表的音符。

理解 MIDI 筛选器

MIDI 筛选器可以在 Actor-Mixer Hierarchy 对象的 Property Editor 的 MIDI 选项卡中找到。在收到有关复杂对象结构的 MIDI 消息时，您可以使用 MIDI 筛选器来选择要播放的子对象。

MIDI 筛选器指定将基于以下项来播放相应的子对象：

- MIDI 音符音高。
- MIDI 音符力度。
- MIDI 声道。

因为在声音音高逐渐偏离根音时做重采样可能会产生副作用，所以您可能会需要使用不同根音上的多个录制样本来覆盖乐器的整个音域。在内存受限时有一种技术通常能实现好的效果：在每个八度音阶范围内只能使用几个不同样本根音，并且将这些音符的音高上调或下调以覆盖八度音阶的全部 12 个半音。

以下是乐器两个八度音阶将根音音高下调二度和上调小二度的示例。

- 乐器（混合容器）
 - 根音：C3 - 范围：Bb2 至 C#3
 - 根音：E3 - 范围：D3 至 F3
 - 根音：G#3 - 范围：F#3 至 A3
 - 根音：C4 - 范围：Bb3 至 C#4
 - 根音：E4 - 范围：D4 至 F4
 - 根音：G#4 - 范围：F#4 至 A4

在播放容器时，混合容器同时播放各个子对象。然而，筛选器会阻止与筛选器规则不匹配的子对象。

了解 MIDI 事件

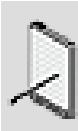
MIDI 事件属性可在 Actor-Mixer Hierarchy 对象的 Property Editor 的 MIDI 选项卡中找到。在收到 MIDI 消息时，MIDI 事件属性用于确定对象是否必须要播放。对象可在

发生 note-on (音符开) 事件或 note-off (音符关) 事件时播放。请注意，这些属性仅用于启动对象的播放。要停止播放对象，必须为属性指派包络（请参阅[使用包络](#)）。

典型的情形是在 Note-On 时 Play (播放)。

创建循环乐器的方法如下：

- 在 Project Explorer 中，选择要循环的声音。
- 在 Property Editor 中，转至 General Settings (常规设置) 选项卡。
- 对声音启用 Loop (循环)。
- 在 Property Editor 中，转至 MIDI 选项卡。
- 将 Play On (播放) 属性设置为 Note-On。
- 从 Views 菜单中，打开 Source Editor (Shift+X)。
- 再次察看声音。
- 在 Source Editor 中，移动 Loop Start> (循环开始) 和 >Loop End> (循环结束) 光标以排除 WAV 文件的起音区段和释放区段。
- 调整 Crossfade duration (交叉淡化时长)，直到无法听到循环点。



备注

乐器的循环可通过 Break on Note-Off (音符关时中断) 属性进行停止。如果该属性已设置，则音符关事件会在允许当前对象完成播放的同时停止循环声音的播放。请记得在源编辑器中设置 Loop Start 和 Loop End 光标，以便为最终循环播放所需的时长。

另一个情形是在释放乐器（即音符关）时播放特定声音。例如，它可用于触发吉他音符结束时的弦闷音。您应创建以下对象和设置：

- Blend Container (混合容器)
 - 起音+循环声音：播放 = Note-On
 - 释放声音：播放 = Note-Off

在 MIDI 事件上添加淡入和淡出

在您希望向乐器添加动态时，您可以使用连接到乐器的 Voice Volume (声部音量) 的 Envelope (包络)。

要向 Voice Volume 添加 Envelope，请参阅[使用包络](#)。

使用 MIDI 数据控制对象属性值

可以使用以下 MIDI 消息来控制对象属性值：

- MIDI 音符力度。
- MIDI 音符音高 (音符号)。
- MIDI 音符频率。
- MIDI 音符触后。
- MIDI CC 值 (0-127，包括调制轮)。
- MIDI 弯音。

典型情形是使用 MIDI Note Velocity (MIDI 音符力度) 控制乐器的 Voice Volume。

使用 MIDI Note Velocity 控制 Voice Volume 的方法如下：

1. 在 Project Explorer 中，从 Actor-Mixer Hierarchy 中选择一个对象。
2. 在 Property Editor 中，转至 RTPC 选项卡。
3. 点击 RTPC 列表中的 [>>] 按钮以添加新的条目。
4. 从选择器菜单中，选择 Voice Volume。
5. 为 X 轴点击 [>>] 选择器。
6. 从选择器菜单中，选择 MIDI > MIDI Note Velocity。
7. 调整 RTPC 图中的声部音量曲线。

使用 MIDI Keymap Editor

MIDI Keymap Editor 视图可用于编辑 Actor-Mixer Hierarchy 对象的所有 MIDI 属性。

打开该视图的方法如下：

1. 在 Project Explorer 中，从 Actor-Mixer Hierarchy 中选择一个对象。
2. 在 Property Editor 中，转至 MIDI 选项卡。
3. 按 Keymap Editor 按钮。

为多个对象设置相同属性值的方法如下：

1. 在 MIDI Keymap Editor 中选择要编辑的对象。
2. 为其中一个所选对象设置属性值。

所选对象现在已设置为相同值。

偏移多个对象的属性值的方法如下：

1. 在 MIDI Keymap Editor 中选择要编辑的对象。
2. 按住 ALT 键并移动其中一个所选对象的属性滑杆。

所选对象的属性值于是就被偏置了。

在 MIDI Keymap Editor 中添加属性的方法如下：

1. 打开 MIDI Keymap Editor 视图设置 (Ctrl+Alt+V) 。
2. 选择要添加的属性。
3. 按 OK。

新列现在添加上了。

使用 MIDI 键盘测试乐器

在设计 MIDI 乐器时，您可以使用外部 MIDI 键盘设备测试乐器。

将设备连接到 Wwise 的方法如下：

1. 在 Project 菜单中，选择 Control Surface Devices（控制设备）。
2. 点击 Add（添加）按钮。
3. 为设备命名。
4. 点击 OK。

设备现在添加到列表里了。

5. 在 Receive From（输入端）列中，选择 MIDI IN（MIDI 输入）设备。

此时将会显示 Connected（已连接）消息。

6. 在 Send To（输出端）列中，选择 MIDI OUT（MIDI 输出）设备。

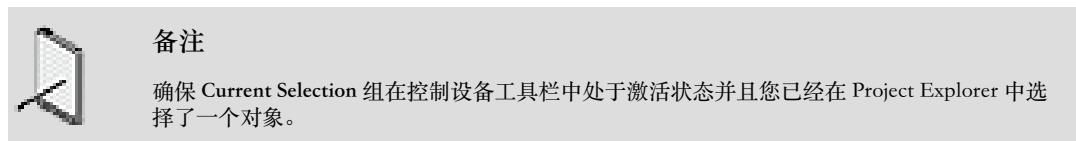
此时将会显示 Connected（已连接）消息。

7. 点击 Close（关闭）。

设备现在就可以用了。

将键盘键绑定到当前选中项的方法如下：

1. 从 Views 菜单中，选择 Control Surface Bindings（Ctrl+Shift+Q）。
2. 通过点击视图左上方的 [>>] 按钮，创建新的控制设备会话。
3. 点击 Current Selection（当前选中项）组（文件夹）。
4. 点击 Add Binding（添加绑定）按钮。
5. 点击 Property/Command（属性/命令）选择器按钮以打开菜单。
6. 选择 Object Command > Pass MIDI Note（对象命令 > 传递 MIDI 音符）。
7. 保存工程。



在创建会话和绑定后，在 Project Explorer 中选择一个对象将自动为 MIDI 乐器加载该对象，并随时可以播放。

将 MIDI 从 DAW 连通到 Wwise

可以使用虚拟 MIDI 连接器将来自外部应用程序（DAW 或数字音频电脑）的 MIDI 消息连通到 Wwise。在使用 Wwise 内置的乐器时在 DAW 中制作 MIDI 音乐非常有用。

通过创建包含所有乐器的混合容器并使用 MIDI Channel 筛选器区分乐器，您可以同时创建多个乐器。

- Mac: 您将要使用的是内置 IAC Driver。从 Applications/Utilities 下启动 Audio MIDI Setup 来开始操作。如果不显示 MIDI Studio，请在 Window 中选择 Show MIDI Window。单击 MIDI Device，双击 IAC Driver 图标，并选中 Device is online 选项将

其激活。添加任意数量的端口后，它们将显示在您的 DAW 中。端口可用于发送或接收 MIDI Note、MIDI Control Change 和 MIDI Sync 消息。

- **Windows:** 由于 Windows 不自带虚拟 MIDI 驱动程序，您必须下载该程序。在 Windows 上，Mac 的 IAC Bus 替代工具中最知名的是 MIDI Yoke，它是 [MIDI OX Utility](#) 的组件。此驱动程序的另一个替代工具是 Tobias Erichsen 的 [loopMIDI](#)，可以用于将 MIDI 消息从 DAW 输出到 Wwise。您需要将应用程序创建的虚拟 MIDI 端口添加到 Wwise 的控制设备中去。

第 28 章 使用 Transition

概述	473
理解 Transition	473
添加 Transition	475
复制和粘贴 Transition	476
移除 Transition	476
设置源和目标属性	477
使用 Transition Segment	479
互动音乐中 Transition 的技巧和经验总结	481

概述

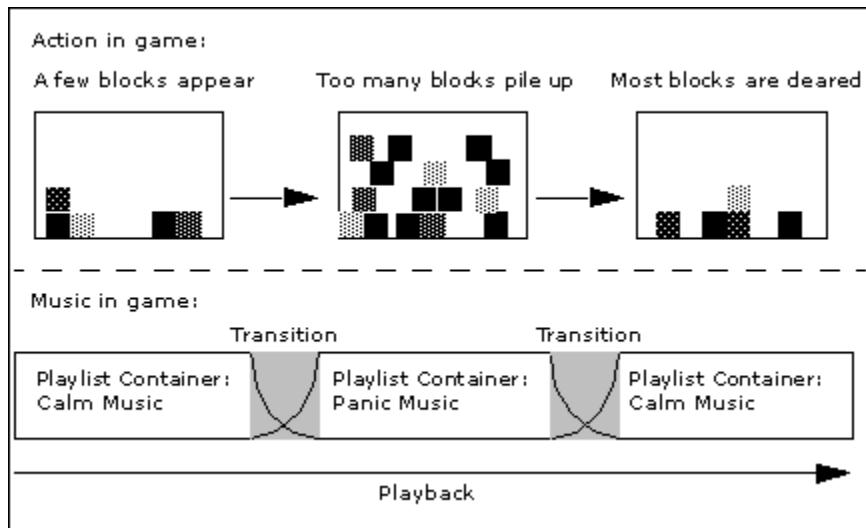
想要交互音乐听起来自然，关键在于音乐的 Transition（过渡）。没有平滑的过渡，游戏配乐就无法保持连续无缝的变化。将各段音乐粗暴地组合在一起会让人感到极不和谐，并会削弱游戏的真实感。要避免这些问题，可以通过 Wwise 来订制音乐对象之间的过渡，尽可能做到无缝衔接。

过渡好比是源段落与目标段落之间的平滑桥梁，Wwise 为您提供工具来构建这样的桥梁。可定义各个段落间的过渡，以及段落所在容器之间的过渡。这样就能避免音乐段落之间的不和谐和突变。

使用 Transition —— 示例

假设您正在制作一款方块消除游戏。在游戏过程中，玩家时刻处于以下状态之一：游戏进行得很好（方块被快速清除），或游戏进行得很差（方块都堆在了一起）。您为每种情景都制作了很酷的音乐来最大限度地展示它们。为此，还得确保这两段音乐的过渡听起来自然、和谐。

为满足游戏需要，您可以创建名为 Puzzle Sounds 的 Music Switch Container（音乐切换容器），其中包含两个 Music Playlist Container（音乐播放列表容器），分别命名为 Calm 和 Panic。它们分别关联至两种游戏状态，反映游戏情况。在这两个 Playlist Container 之间实现良好过渡可以让游戏配乐引人入胜。下图显示了如何在游戏中实现过渡：

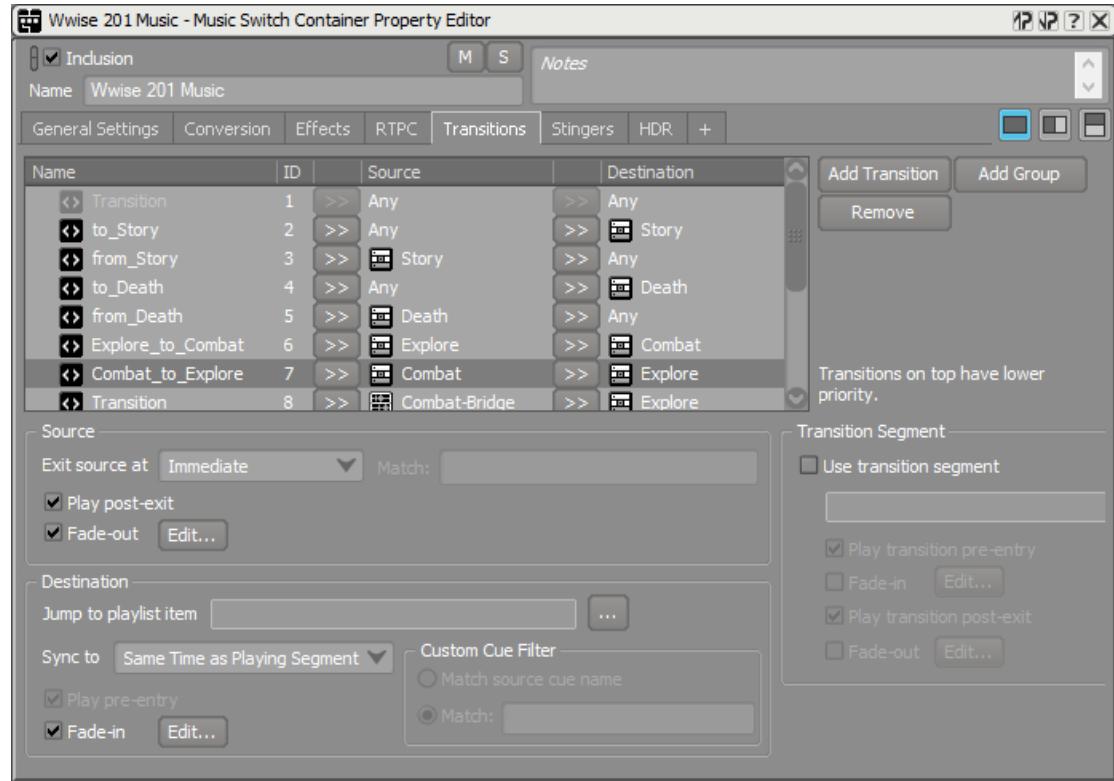


在本例中，使用淡入和淡出曲线可以让 Playlist Container 之间的过渡变得平滑。两个容器间的衔接将更为平缓、自然，还能避免产生噪声缺陷。

理解 Transition

互动音乐的过渡是一种音乐行为，每当前一个音乐对象（称为 Source，即源对象）停止播放，后一个音乐对象（称为 Destination，即目标对象）开始播放时，过渡就开始了。根据 Property Editor（属性编辑器）的 Transitions（过渡）选项卡中所定义的属性，Wwise 会自动执行各个过渡。

使用 Transition



每当游戏音乐从一个 Music Segment（音乐段落）切换到另一段落，或从一个包含段落的容器更改为另一容器时，Wwise 都会执行过渡。您可以指定过渡何时开始，以及是否进行淡变。也可以指定一段音乐作为 Transition Segment（过渡段落），在衔接时播放。总之，您可以在 Wwise 中自定义过渡段落，来构建无缝衔接的互动音乐。

理解 Transition 规则矩阵

Transitions（过渡）选项卡的中间是过渡矩阵（Transition Matrix），里面的一系列规则定义了 Music Switch Container（音乐切换容器）或 Music Playlist Container（音乐播放列表容器）内的各个对象如何过渡至容器内的其它对象。设置过渡时，可以为容器内的各对象创建显性规则，也可以创建适用多个对象的通用规则。以下元素均可作为过渡规则的 Source（源）或 Destination（目标）：

- **Music object:** 音乐对象，可以是 Music Segment（音乐段落）、Music Playlist Container，也可以是 Music Switch Container。
- **Virtual Folder:** 虚拟文件夹，Music Switch Container 内的音乐对象可组织到虚拟文件夹中。将 Virtual Folder 选为音乐过渡的 Source 或 Destination 时，过渡规则将对该文件夹内的所有音乐对象生效。
- **Any:** 任意对象，表示容器中的任何音乐对象都可用作 Source 或 Destination。
- **Nothing:** 无对象，表示 Source 或 Destination 为空，即不播放任何音乐对象。

过渡时，Wwise 将从规则列表底部开始向上搜索，直至找到与当前情形匹配的过渡规则。如果没有找到，Wwise 则将使用默认的“Any to Any”过渡。

添加 Transition

Music Playlist Container（音乐播放列表容器）或 Music Switch Container（音乐切换容器）的过渡矩阵中至少包含一个 Transition（过渡）规则：Any to Any。如果想创建更特殊的过渡规则，须将其添加至过渡规则矩阵。

为 Music Switch Container 或 Playlist Container 添加自定义过渡的方法如下：

1. 将 Music Switch Container 或 Music Playlist Container 加载至 Property Editor 中。
2. 切换至 Transitions 选项卡。
3. 点击 Add。

新的 Any to Any 过渡规则将被添加至过渡矩阵。

4. 在 Source（过渡源）列中，点击选择按钮(>>)，并选择下列选项之一：
 - Any任意对象。过渡将适用于任意源音乐对象。
 - Nothing无对象。创建没有源对象（即不播放任何音乐对象）的过渡。
 - Browse，浏览并选择特定音乐对象或含有对象的虚拟文件夹，过渡规则会将其作为源对象。Project Explorer - Browser 将打开，可以从中选择过渡源。



技巧

也可以从 Project Explorer 中将音乐对象直接拖拽到过渡矩阵中。但要注意，对于加载至 Property Editor 中的当前容器，只有其子音乐对象才能添加至矩阵中。

5. 在 Destination（过渡目标）列中，点击选择按钮(>>)，并选择下列选项之一：
 - Any任意对象。过渡将适用于任意目标音乐对象。
 - Nothing无对象。创建没有目标对象（即不播放任何音乐对象）的过渡。
 - Browse，浏览并选择特定音乐对象或含有对象的虚拟文件夹，过渡规则会将其作为目标对象。Project Explorer - Browser 将打开，可以从中选择过渡目标。

新的自定义过渡规则将被添加至过渡矩阵。



技巧

在过渡矩阵中选择规则并将上下拖动，即可更改它们的顺序。红线会显示过渡规则的最终位置。“Any to Any”将是 Wwise 检查的最后一个过渡规则，因此无法从矩阵顶部下移。

相关主题

- [移除 Transition](#)
- [设置源和目标属性](#)
- [复制和粘贴 Transition](#)
- [使用 Transition Segment](#)

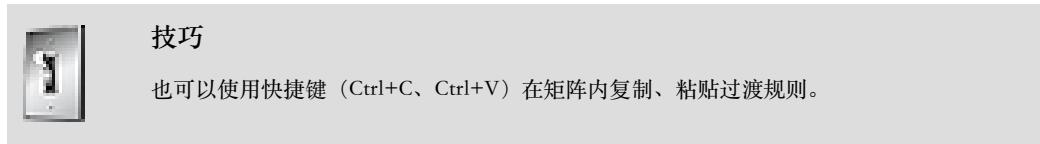
复制和粘贴 Transition

为了让创建 Transition（过渡）的流程更快捷，过渡矩阵内的规则可以轻松复制、粘贴。

从 Music Playlist Container（音乐播放列表容器）或 Music Switch Container（音乐切换容器）中复制自定义过渡规则的方法如下：

1. 将 Music Switch Container 或 Music Playlist Container 加载至 Property Editor 中。
2. 切换至 Transitions 选项卡。
3. 在过渡矩阵中，右键点击要复制的过渡规则，并在菜单中选择 Copy。
4. 在列表中，右键点击要粘贴过渡规则的位置，并在菜单中选择 Paste。

过渡规则的副本将添加至过渡矩阵中所选位置。



相关主题

- [添加 Transition](#)
- [设置源和目标属性](#)
- [使用 Transition Segment](#)

移除 Transition

如果您不再需要过渡矩阵中已有的过渡规则，可以轻松将其移除。移除过渡规则时，不会删除 Source（源）和 Destination（目标）音乐对象。

从 Music Playlist Container（音乐播放列表容器）或 Music Switch Container（音乐切换容器）中移除自定义过渡规则的方法如下：

1. 将 Music Switch Container 或 Music Playlist Container 加载至 Property Editor 中。
2. 切换至 Transitions 选项卡。
3. 在过渡矩阵中，选择要移除的过渡规则。
4. 点击 Remove 或按 Delete 键。

所选过渡规则将从过渡矩阵中移除。

相关主题

- [添加 Transition](#)
- [设置源和目标属性](#)
- [复制和粘贴 Transition](#)

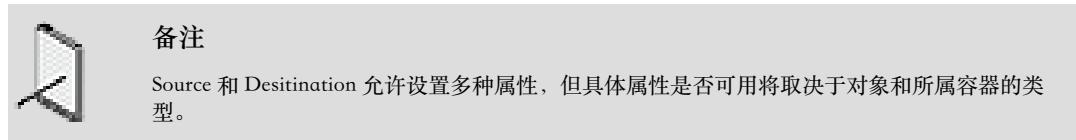
- 使用 Transition Segment

设置源和目标属性

默认情况下，过渡规则将使一个音乐对象简单切换至另一个音乐对象。过渡规则的强大之处在于您可以自定义 Source（源）和 Desitination（目标）对象，并创建独特的音乐过渡。通过设置 Source 和 Desitination 属性，可以使对象间的过渡听起来平滑、流畅。

为了让对象间的过渡更加灵活可控，可以为 Source 和 Desitination 分别设置多个不同的 Exit（切出）点和 Entry（切入）点。您也可以在 Desitination 对象中选择随机时刻切入。这样按照过渡规则播放时每次切入点都不同，可以减少过渡的重复听感。

设置 Source 和 Desitination 属性时，请注意这些属性仅对该过渡规则有效。例如，假设您的过渡矩阵中存在从 Happy Music（愉快情绪）到 Sad Music（悲伤情绪）的过渡规则，以及从 Happy Music 到 Scary Music（恐惧情绪）的过渡规则。如果希望两个规则中 Happy Music 源都在第一个 Custom Cue（自定义提示点）切出，就需要在两个规则中分别设置。



设置 Source 属性的方法如下：

1. 将 Music Switch Container 或 Music Playlist Container 加载至 Property Editor 中。
2. 切换至 Transitions 选项卡。
3. 在过渡矩阵里选中过渡规则。

现在可以为该过渡规则编辑 Source 属性了。

4. 如果 Source 对象位于切换容器内，请从 Exit source at（切出点）列表中选择下列之一：

Immediate（立刻）：立即停止播放 Source。

Next Grid（下一网格线）：在下一个网格线停止播放 Source。通过 Grid（网格线），可以按任意间隔长度来划分音乐对象。

Next Bar（下一小节）：在下一小节停止播放 Source。

Next Beat（下一拍）：在一拍停止播放 Source。

Next Cue（下一提示点）：在下一个提示点停止播放 Source，可以是 Custom Cue，也可以是 Exit Cue（出口提示点）。

Next Custom Cue（下一自定义提示点）：在下一个 Custom Cue 停止播放 Source。如果当前 Music Segment（音乐段落）不包含 Custom Cue，则 Wwise 将继续播放下一段落，直到遇到 Custom Cue 为止。

Exit Cue（出口提示点）：在出口提示点处停止播放 Source。

5. 如果您选择 Next Cue 或 Next Custom Cue，就可以在 Match 文本框中输入提示点名称，精确筛选对该过渡规则有效的提示点。
6. 如果过渡时希望播放 Source 的 Post-exit（后尾段），请选择 Play post-exit。



7. 如果 Source 结束时需要淡出，请选择 Fade-out。

设置 Destination 属性的方法如下：

1. 将 Music Switch Container 或 Music Playlist Container 加载至 Property Editor 中。
2. 切换至 Transitions 选项卡。
3. 在过渡矩阵里选中过渡规则。

现在可以为该过渡规则编辑 Destination 属性了。

4. 如果目标对象是 Music Playlist Container，还可以选择其中的特定对象来播放。点击 Browse 按钮 (...) 并为 Jump to playlist item（跳转至播放列表项）指定对象。
5. 如果是 Music Switch Container 的过渡，请从 Sync to 列表中选择以下选项之一：

Entry Cue（入口提示点）：将从入口提示点处开始播放 Destination。

Same Time as Playing Segment（与现有音乐时间相同）：Destination 开始播放的位置将与 Source 段落切出的时间点相同。例如，如果 Source 段落已经播了 10 秒，则 Destination 也将从 10 秒处开始播放。

Random Cue（随机提示点）：从随机选择的提示点处开始播放 Destination。勾选该选项即选择了所有提示点，包括 Entry Cue（入口提示点）和所有 Custom Cue。

Random Custom Cue（随机自定义提示点）：将从随机选择的 Custom Cue 处开始播放 Destination。如果段落中没有 Custom Cue，则将使用 Entry Cue。

6. 如果选择了 Random Cue 或 Random Custom Cue，还可以使用 Custom Cue Filter 精确筛选有效的提示点。

Match（匹配名称）：只有用了该名称的提示点才会被选为开始位置。

Match source cue name（匹配源提示点名称）：仅当提示点与 Source 段落所用提示点名称相同时，才会作为开始位置。

7. 过渡时如果要播放 Destination 的 Pre-entry（前导段），请选择 Play pre-entry（播放前导段）。
8. 如果开始播放 Destination 时需要淡入，请选择 Fade-in。

相关主题

- [添加 Transition](#)
- [移除 Transition](#)

- [复制和粘贴 Transition](#)
- [使用 Transition Segment](#)

编辑淡变

Fade-in（淡入）和 Fade-out（淡出）是两个特殊属性，分别针对 Destination（目标）和 Source（源）音乐对象，也可以用于 Transition Segment（过渡段落）的开头和结尾。音乐对象切入和切出时，您可以用淡变让过渡变得更为平滑，很能设置各淡变的长度，时间偏置及曲线形状，进一步修正淡变听感。

编辑 Fade-out 的方法如下：

1. 在 Property Editor（属性编辑器）的 Transitions 选项卡中，选中 **Fade-out** 并点击 **Edit**。
2. 在 **Music Fade Editor**（音乐淡变编辑器）将打开。
3. 在 **Time** 文本框中，键入淡出的持续时间。
4. 在 **Offset** 文本框中，可以指定 Exit Cue（出口提示点）到淡出结束的时长。
5. 在 **Curve** 列表中，选择淡出的曲线形状。

音乐将按照指定的方式淡出。

编辑 Fade-in 的方法如下：

1. 在 Property Editor 的 Transitions 选项卡中，选中 **Fade-in** 并后点击 **Edit>**。
2. 在 **Music Fade Editor**（音乐淡变编辑器）将打开。
3. 在 **Time** 文本框中，键入淡入的持续时间。
4. 在 **Offset** 文本框中，可以指定从淡入开始到 Entry Cue（入口提示点）的时长。
5. 从 **Curve** 列表中，选择淡入的曲线形状。

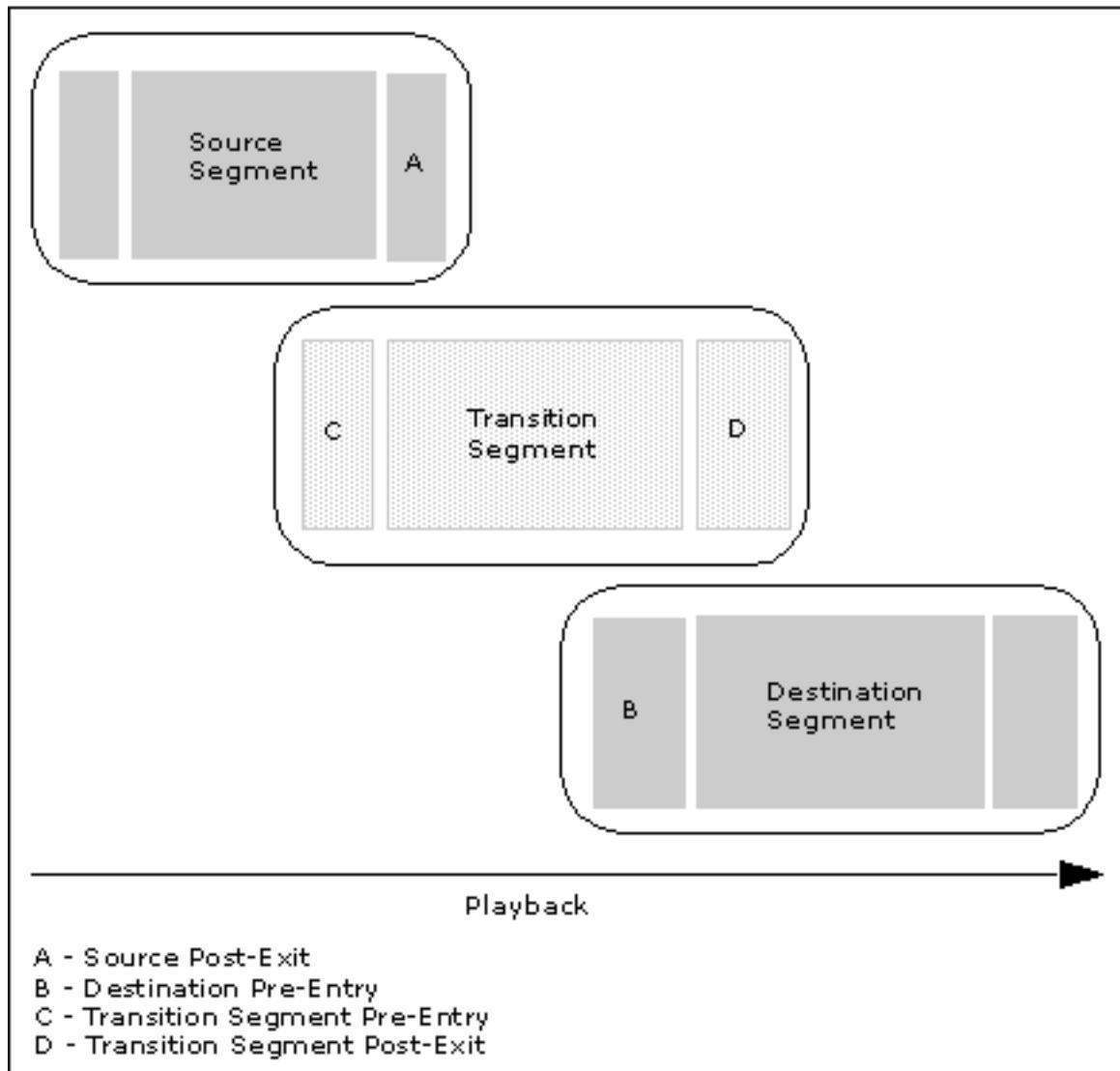
音乐将按照指定的方式淡入。

相关主题

- [添加 Transition](#)
- [移除 Transition](#)
- [设置源和目标属性](#)
- [复制和粘贴 Transition](#)
- [使用 Transition Segment](#)

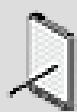
使用 Transition Segment

有时为了让过渡效果更好，需要另选一段音乐，让它在 Source（源）音乐尚未结束就开始播放，并持续至 Destination（目标）音乐开始后。连接所用的这段音乐叫做 Transition Segment（过渡段），在 Wwise 中的所有过渡中都可以使用。下图说明了从 Source 对象到 Destination 对象的过渡中，Transition Segment 将如何播放。



用户也可以用 Source 对象、Destination 对象及 Transition Segment 三者的 Pre-entry（前导段）与 Post-exit（后尾段）进行组合，让过渡更加平滑无缝。

任何 Music Segment（音乐段落）都可以作为过渡段，关于如何创建段落，请参阅[第 25 章 使用 Music Track 和 Music Segment](#)。



备注

如果过渡段所在的 Work Unit（工作单元）已从工程中卸载，该过渡段将显示为黄色。

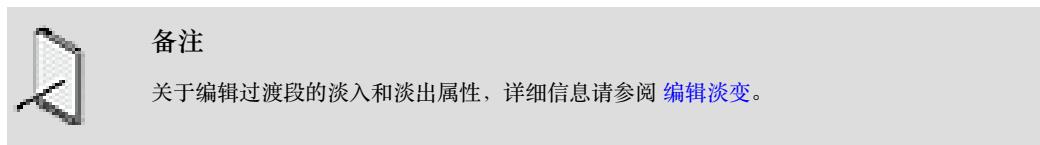
使用 Transition Segment 的方法如下：

1. 将 Music Switch Container（音乐切换容器）或 Music Playlist Container（音乐切换容器）加载至 Property Editor（属性编辑器）中。
2. 切换至 Transitions 选项卡。

3. 在过渡矩阵中选择过渡规则。
4. 勾选 **Use transition segment**。
5. 执行以下操作之一：
 - 将段落从 Project Explorer 拖至 Transition Segment 中。



- 点击 **Browse** 按钮 (...), 在 Project Explorer - Browser 中选择段落。
- 6. 如果要播放过渡段的 Pre-entry, 请勾选 **Play pre-entry**。
- 7. 如果播放过渡段时需要淡入, 请勾选 **Fade-in**。
- 8. 如果要播放过渡段的 Post-exit, 请勾选 **Play transition post-exit**。
- 9. 如果停止过渡段时需要淡出, 请选择 **Fade-out**。



相关主题

- [添加 Transition](#)
- [移除 Transition](#)
- [复制和粘贴 Transition](#)
- [设置源和目标属性](#)

互动音乐中 Transition 的技巧和经验总结

互动音乐是一套复杂工具, 提供了多种选项。自工程伊始就为互动音乐制定统一的策略, 可节省后期的时间和精力。当然, 处理互动音乐的方法有很多, 在 Wwise 中, 您可以使用任何方法来为游戏创造最佳的结果。就如何更好地管理互动音乐中的 Transition (过渡), 以下是一些建议。

为 Transition 规则排序

进行过渡时, Wwise 会在过渡矩阵中自下至上检索, 寻找与当前情形匹配的过渡规则。一旦找到匹配的过渡规则就会停止检索, 无论找到的过渡规则是否最为合适。为确保使用最佳过渡规则, 过渡矩阵中的规则可以按从通用到特定的顺序排列, 如下例所示。

Source	Destination
>> Any	>> Any
>> Any	>> Exciting Music
>> Exciting Music	>> Any
>> Casual Music	>> Creepy Music
>> Creepy Music	>> Casual Music

与 Source（源）或 Destination（目标）使用特定音乐对象的过渡规则相比，使用 **Any** 或 **Nothing** 的过渡规则应该排在列表前面，这样 Wwise 在检索通用规则前就已经找到了特定规则。关于创建和排列过渡规则的详细信息，请参阅 [添加 Transition](#)。

第 29 章 使用 Stinger

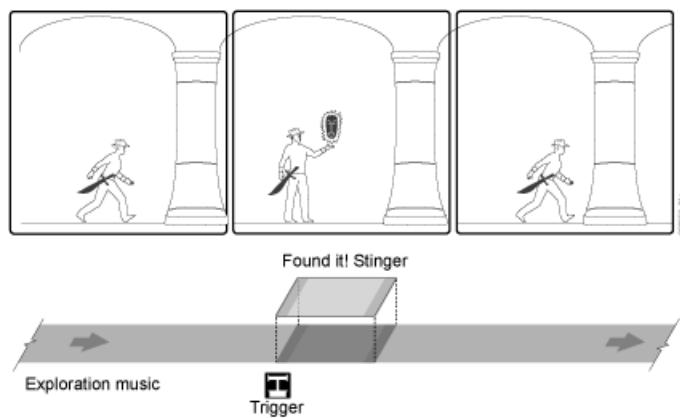
概述	484
添加 Stinger	485
定义 Stinger 的播放设置	486
移除 Stinger	487
预览 Stinger	488

概述

为了让互动音乐给游戏提供更多反馈，您可以在游戏关键时刻播放 Stinger（插播乐句）。Stinger 是一种简短乐句，可以与当前播放的音乐叠加并混合播放。游戏会调用与 Stinger Music Segment（插播乐句音乐段落）相关联的 Trigger（触发器），从而播放 Stinger。关于使用 Trigger 的详细信息，请参阅 [第 20 章 使用 Trigger](#)。

使用 Stinger —— 示例

假设上例中勇敢的考古学家正在探索神庙，展开寻找遗迹的冒险之旅。在他探索周边事物时会播放探索音乐，而找到宝藏时将播放一段小音乐，强调这一振奋人心的发现。但探索音乐不会停止，游戏会调用 Trigger，让一个名为“Found it”的 Stinger 与当前音乐叠加播放，来表示这是重要时刻。Stinger 播完后，探索音乐将继续播放。



要让 Stinger 与当前音乐叠加播放，需进行以下操作：

- 将音乐对象与 Trigger 关联。
- 将 Music Segment 映射至 Trigger，从而创建 Stinger。
- 定义如何播放 Stinger。

在 Interactive Music Hierarchy 中使用 Stinger

Stinger 允许在不同层级中创建，因此您可以将同一 Stinger 关联至不同的 Music Segment。比如顶层音乐对象使用了名为“Headshot”的 Trigger，那么其子对象也可以使用这个 Trigger，并且子对象可以将“Headshot”Trigger 关联到其他段落。这会自动覆盖父级对象的 Trigger/Stinger 关联，使得游戏重要时刻可以播放更多不同的 Stinger。对于同一 Trigger，每次只能播放一个 Stinger，因此只有当前子对象所关联的 Stinger 会播放。

使用 Stinger 涉及以下任务：

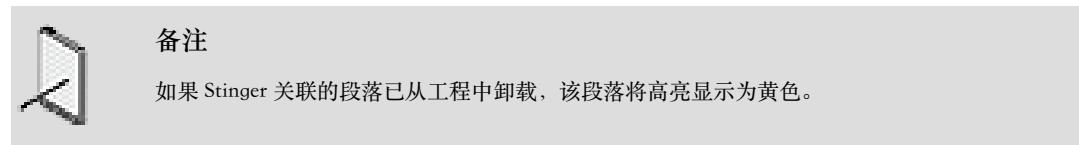
- [添加 Stinger](#)
- [定义 Stinger 的播放设置](#)
- [预览 Stinger](#)
- [移除 Stinger](#)

添加 Stinger

您可以在 Music Object Property Editor（音乐对象属性编辑器）的 Stingers（插播乐句）选项卡中为音乐对象创建 Stinger。Wwise 允许通过两种方式添加 Stinger：

- 在 Project Explorer（工程资源管理器）中进行拖放。
- 使用 Property Editor（属性编辑器）中的按钮。

您可以任选其一，也可结合使用。



从 Project Explorer 中拖拽添加 Stinger 的方法如下：

1. 将音乐对象加载到 Property Editor（属性编辑器）中，并切换到 Stingers 选项卡。
2. 从 Project Explorer 的 Game Syncs（游戏同步器）选项卡中，将 Trigger（触发器）拖至 Stinger 列表。

Stinger 将自动创建，并由刚刚拖拽的 Trigger 触发。默认情况下，Trigger 不会关联至任何 Music Segment（音乐段落），即显示 Nothing（无段落）。

3. 要将 Music Segment 与 Trigger 关联，请切换至 Project Explorer 的 Audio（音频）选项卡，然后将 Music Segment 拖至 Segment to Play（要播放的段落）一栏。

该 Music Segment 将替换默认“Nothing”选项，并与该 Trigger 关联。

4. 在 Play At（播放位置）列中选择以下选项之一，来定义何时播放段落：
 - Immediate —— 立即切换状态。如果为 Music Track（音乐轨）设置了 Look-ahead Time（预读时间），则须等到这段时间过后才能播放该 Stinger。
 - Next Grid -- 下一网格线。在下一网格线处播放 Stinger。通过 Grid，可以按想要的频率将音乐对象进行虚拟划分。
 - Next Bar -- 下一小节。在下一小节处播放 Stinger。
 - Next Beat -- 下一拍。在下一拍播放 Stinger。
 - Next Cue -- 下一提示点。在下一提示点处播放 Stinger。下一提示点可以是 Entry Cue（入口提示点）、Exit Cue（出口提示点）或者 Custom Cue（自定义提示点）。
 - Next Custom Cue -- 下一自定义提示点。在下一个自定义提示点处播放 Stinger。
 - Entry Cue -- 入口提示点。在入口提示点处播放 Stinger。
 - Exit Cue -- 出口提示点。在出口提示点处播放插播乐句。
5. 重复步骤 2-4，继续按需添加 Stinger。

在 Property Editor 中通过按钮插入 Stinger 的方法如下：

1. 在 Property Editor 的 Stingers 选项卡中，点击 Add。

Project Explorer - Browser（浏览）对话框将打开。

2. 选择要指派给音乐对象的 Trigger，并单击 OK（确定）。



3. 选择 Trigger 并单击 OK。

Trigger 将添加至 Stinger。

4. 在 Segment to Play 列中，点击 "Browse" 按钮 (...)。

Project Explorer-Browser 将打开。

5. 转到要使用的段落，并单击 OK。

该段落将添加至 Stinger。

6. • Immediate —— 立即切换状态。如果为 Music Track (音乐轨) 设置了 Look-ahead Time (预读时间)，则须等到这段时间过后才能播放该 Stinger。
- Next Grid -- 下一网格线。在下一网格线处播放 Stinger。通过 Grid，可以按想要的频率将音乐对象进行虚拟划分。
- Next Bar -- 下一小节。在下一小节处播放 Stinger。
- Next Beat -- 下一拍。在下一拍播放 Stinger。
- Next Cue -- 下一提示点。在下一提示点处播放 Stinger。下一提示点可以是 Entry Cue (入口提示点)、Exit Cue (出口提示点) 或者 Custom Cue (自定义提示点)。
- Next Custom Cue -- 下一自定义提示点。在下一个自定义提示点处播放 Stinger。
- Entry Cue -- 入口提示点。在入口提示点处播放 Stinger。
- Exit Cue -- 出口提示点。在出口提示点处播放 Stinger。

7. 继续按需添加 Stinger。

相关主题

- [定义 Stinger 的播放设置](#)
- [移除 Stinger](#)
- [预览 Stinger](#)

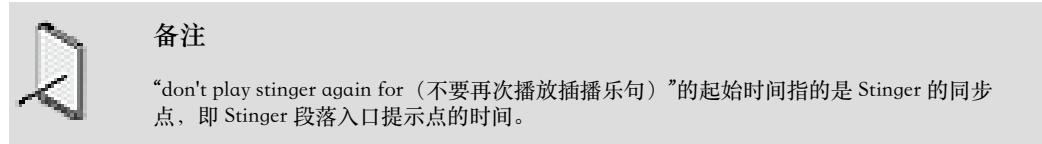
定义 Stinger 的播放设置

为音乐对象创建 Stinger (插播乐句) 后，您可以进行设置，管理 Stinger 在游戏中如何播放。要使 Stinger 达到最佳效果，需要考虑以下两个问题：

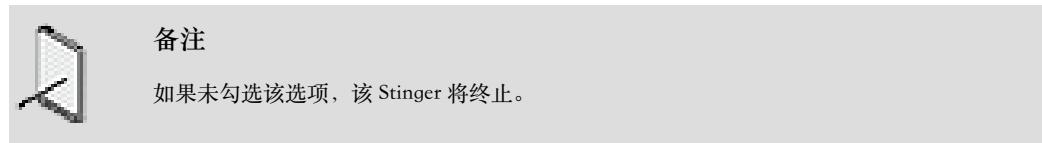
- 播放 Stinger 后，要等多长时间才能再次播放。Stinger 的作用是增加音乐编曲的表现力，因此播放时一定不要过多、过频繁，以免弱化其作用。
- 段落所剩时间不足以播放 Stinger 时怎么办。如果 Trigger 调用了 Stringer，但在播放时机到来前当前段落就会结束，则不会播放该 Stinger。不过，可选择在播放列表的下一段落中首次遇到符合条件的机会时播放该段落。如果未勾选该选项，该 Stinger 将终止。

定义 Stinger 播放设置的方法如下：

1. 在 Property Editor（属性编辑器）的 Stingers（插播乐句）选项卡内，在 **Don't play this Stinger again for x seconds**（在 x 秒内不再播放该插播乐句）字段中键入须经过几秒才能再次播放该 Stinger。若 Trigger（触发器）在指定时间内调用该 Stinger，则该 Trigger 将被忽略。



2. 若当前段落剩余时间不够而需在下一段落中播放 Stinger，请选中 **Allow playing the Stinger in next segment**（允许在下一段落中播放插播乐句）选项。



相关主题

- [添加 Stinger](#)
- [移除 Stinger](#)
- [预览 Stinger](#)

移除 Stinger

您可以删除不需要的 Stinger（插播乐句）。

从 Stinger 列表中移除 Stinger 的方法如下：

1. 在 Stingers 选项卡中，选择要移除的 Stinger。
2. 点击 Remove（删除）。

所选 Stinger 将从列表中移除。



相关主题

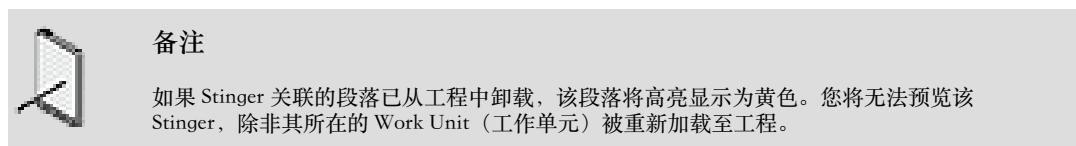
- [定义 Stinger 的播放设置](#)
- [添加 Stinger](#)
- [预览 Stinger](#)

预览 Stinger

创建 Stinger（插播乐句）段落后，可以在 Transport Control（播放控制）中单独预览该段落或与其它段落一起试听，从而验证叠加播放效果。Stinger 允许在不同层级中创建，因此您可以为同一 Stinger 进行不同的段落设置。但对于音乐对象层级结构中的每个 Trigger，由于同时只能播放一个 Stinger，因此只有当前子对象所关联的 Stinger 会播放。例如，如果您使用名为“Headshot”的 Trigger 在 Music Switch Container（音乐切换容器）层级创建了 Stinger，其子对象 Music Playlist Container（音乐播放列表容器）中也使用同一 Trigger 创建了 Stinger，那么在 Transport Control（播放控制）中加载该 Music Switch Container 并调用 Trigger，结果将只播放 Playlist Container 的 Stinger。

此外，还以用 Soundcaster（声音选角器）进行模拟，并通过 Profiler（调试窗口）进行性能分析，监控性能问题。关于模拟和性能分析的详细信息，请参阅：

- [创建模拟](#)
- [使用 Performance Monitor 进行监控和故障排查](#)



在 Transport Control 中预览 Stinger 的方法如下：

1. 将 Stinger 段落加载至 Transport Control。



2. 点击 Play 图标。

Stinger 段落将播放。

预览 Stinger 与当前音乐叠加播放效果的方法如下：

1. 将音乐对象加载至 Transport Control。
2. 点击 Play 图标。

Transport Control 中的音乐对象将播放。

3. 在 Game Syncs 区域中，点击 Triggers 按钮来显示列表。



4. 点击 Call Trigger（调用触发器）图标。

相应的 Stinger 将与当前音乐对象叠加播放。可以继续选择其它 Trigger，播放相应的 Stinger 来试听游戏中的效果。

相关主题

- 定义 Stinger 的播放设置
- 添加 Stinger
- 移除 Stinger



部分 VI. 完善工程



30. 管理输出	493
概述	494
指定对象的输出连线	494
使用用户定义的辅助发送	495
使用游戏定义的辅助发送	497
使用响度归一化或补偿增益来调节音量	497
理解声部管线	498
内置音频设备	500
理解 Secondary Output	501
理解 HDR	506
使用 HDR	509
深入了解 HDR	517
完成终混	521
31. 管理平台和语言版本	536
概述	537
跨平台制作	537
对工程进行本地化	555
版本技巧和经验总结	559
32. 创建模拟	562
概述	563
创建模拟	564
管理模拟的播放	567
使用 Game Sync 模拟	569
在模拟中细调属性	572
创建模拟的技巧和经验总结	574
33. 在 Wwise 中管理内存	576
概述	577
理解内存管理器的组件	577
设置内存池的大小	578
排查内存问题	579
优化内存池	580
内存管理技巧和经验总结	584
34. 性能分析	586
概述	587
连接到本地/远程游戏系统	595
从声音引擎捕获数据	597
使用 Performance Monitor 进行监控和故障排查	606
使用 Voice Inspector 分析声部	608
使用 Game Object Explorer 跟踪对象和听者	614
通过 Game Object 3D Viewer 检查对象	618
通过 Game Sync Monitor 评估游戏同步器	623
性能分析技巧和最佳实践	624
35. 管理 SoundBank	626
概述	627
理解 SoundBank 如何加载到游戏中	629
构建 SoundBank	632
管理 SoundBank	647
定义 SoundBank 的自定义属性	649
为工程生成 SoundBank	656

使用 CopyStreamedFile 工具	659
SoundBank 的管理策略	660
SoundBank 技巧和经验总结	672
36. 管理 File Package	676
概述	677
使用 File Packager 工程	677
管理工程内的 File Package	679
可下载内容 (DLC) 概述	683
生成 File Package	684
在命令行中使用 File Packager 参数	684
File Packager 技巧和最佳实践	687

第 30 章 管理输出

概述	494
指定对象的输出连线	494
使用用户定义的辅助发送	495
使用游戏定义的辅助发送	497
使用响度归一化或补偿增益来调节音量	497
理解声部管线	498
内置音频设备	500
理解 Secondary Output	501
理解 HDR	506
使用 HDR	509
深入了解 HDR	517
完成终混	521

概述

在包含所有对象的工程层级的上方，您可以找到 Master-Mixer Hierarchy。通过这一独立的总线层级结构（之前在 [第 9 章 建立输出总线的结构](#) 中已进行说明），可以将工程层级内很多不同的结构分组。Master Audio Bus 将用于主输出的音频混音。您应为游戏的二路输出（如聊天耳机和游戏手柄振动）创建单独的总线结构。

这些主总线为工程内的结构创建了最终的控制层级。总线位于工程层级的顶层，因此您可以使用这些总线创建游戏的最终混音。

对于不同平台，您还可以在主总线上应用特定效果器，让游戏的沉浸式体验更加完善。

相关主题

- [概述](#)
- [定义总线的属性](#)
- [构建总线层级结构 —— 示例](#)

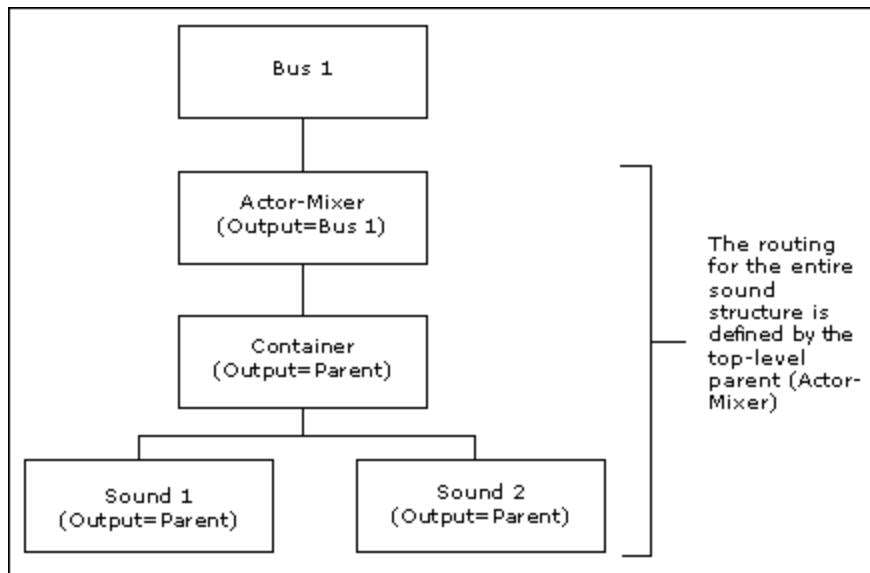
音频输出格式

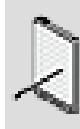
Wwise 将最终混音结果作为离散的 LPCM 声道进行输出。需要注意的是，Wwise 将不会执行任何多声道编码；而只是简单地将 LPCM 数据传送至游戏机或系统。游戏机或系统接收到 LPCM 数据后，可以用其支持的几乎所有格式（包括 Dolby、DTS 或 DPL2）输出。

指定对象的输出连线

层级结构内的各个对象必须连线至特定总线。只需为顶层父对象进行连线设置，就可以定义整个结构的连线。由于输出连线是一个绝对属性，因此这些设置将自动传递至其下的子对象。但如果需要，您则可以取消沿用父级设置。

以下图示说明父对象的输出连线设置将如何自动传递至其子对象。



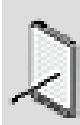


备注

默认情况下，对象会自动连线至在 Default User Setting（默认用户设置）中指定的总线。

指定音频对象的输出连线的方法如下：

1. 将顶层对象加载到 Property Editor 中。



备注

如果该对象不是顶层对象，您必须选择 **Override parent** 选项后才能设置 Output 选项。

2. 在 Output Bus 分组框中，点击 "Browse" 按钮 (...)。

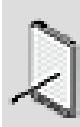
此时将会打开 Project Explorer —— Browser。

3. 选择要将对象进行连线的音频总线。
4. 点击 OK。

音频总线将被应用至当前对象及其下任何子对象。

解除当前平台的输出连线的方法如下：

1. 将顶层对象加载到 Property Editor 中。



备注

如果该对象不是顶层对象，您则必须选择 **Override parent** 后才能设置 Output 选项。

2. 在 Output Bus 分组框中，右键点击当前总线名称以打开快捷菜单。
3. 从快捷菜单中选择 Unlink。

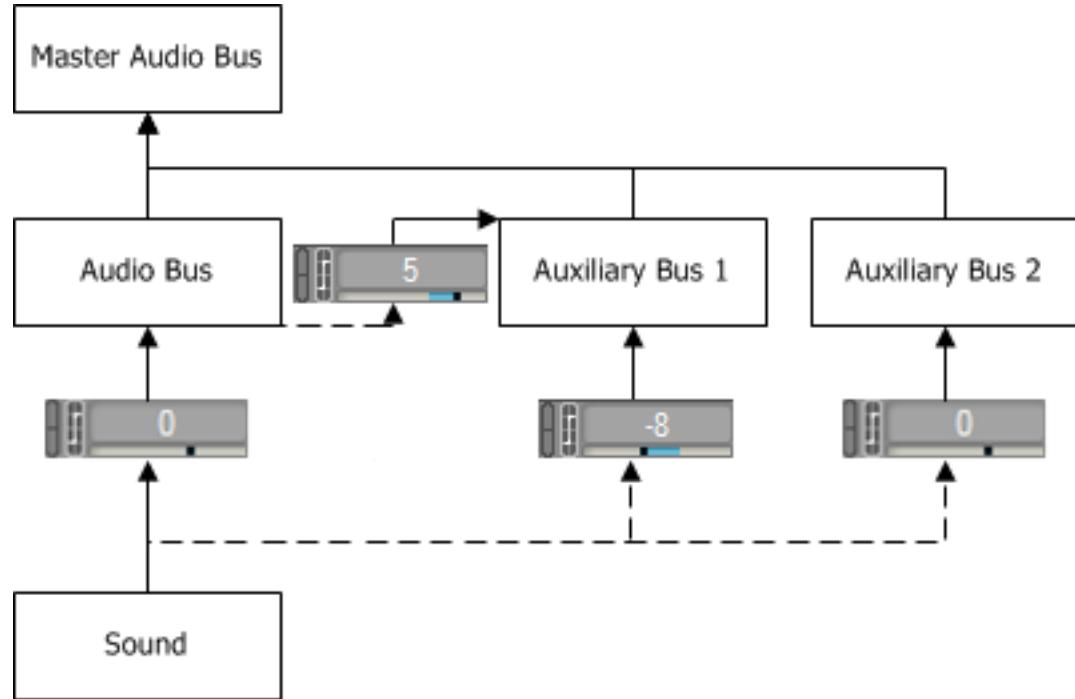
现在总线连线已取消链接，您可以为该平台指定其它总线。

使用用户定义的辅助发送

除了可以输出音频到 Audio Bus，每个 Interactive Music 或 Actor-Mixer Hierarchy 对象还可以指定最多四个 User-Defined Auxiliary Send。同样，Audio Bus 本身也可以具有最多四个 User-Defined Auxiliary Sends。辅助发送允许将音频信号的一部分发送到其他总线（即 Auxiliary Bus）中，进行并行处理。

用户定义的辅助总线用于在创作和编辑时直接定义静态辅助发送，游戏定义的辅助发送则与它不同，大多是通过在游戏中进行动态定义和控制。

下图显示了一个输出至 Audio Bus 的声音，总线上的辅助发送音量为 5dB，并且该声音还另有两个辅助发送。各个辅助发送都包含音量设置，并被连线至一条辅助发送总线。



添加用户定义的辅助发送

1. 在 Project Explorer 中选择对象，察看该对象的属性。
2. 将一条辅助总线从工程浏览器中拖拽至用户定义的辅助发送列表，或
点击浏览按钮 [...] 从 Project Explorer 列表中选择一个辅助总线。
3. 为所选的辅助总线设置发送音量



避免反馈循环

如果将总线发送至其自身的下级 Auxiliary Bus，则信号将被一系列相同的总线进行循环处理。这可能会导致延迟、音量异常巨大或产生其他不和谐音频效果。

使用用户定义的辅助总线来控制游戏环境效果

辅助发送可以用于控制游戏中简单场景的环境效果。环境效果通常由密闭空间中的混响和早期反射声来决定。对于各个发声体，游戏可控制以下元素：

- **Send volume**（发送音量），对应湿声部分或反射声大小。
- **Output bus volume**（输出总线音量），对应干声部分或直达声大小。
- **Output bus low-pass filter**（输出总线低通滤波器），对应阻挡或封闭效果的多少，影响直达声或干声部分的频率响应。

通过将游戏参数关联至前两个值，您可以控制各个游戏对象干声信号和湿声信号的大小。之后游戏会计算听者和发声体之间的距离，并将该值指派至游戏参数。在游戏参数指派中，RTPC 曲线将定义信号的湿声和干声部分如何随距离而变化。

通过将游戏参数关联至输出总线的 Low-Pass Filter，可以根据游戏参数来控制声音中多少高频会被阻挡或削弱。

使用游戏定义的辅助发送

除了 4 个用户定义的辅助发送外，最多还可以设置 4 个游戏定义的辅助发送。在游戏中使用以下 Wwise SDK 函数，即可控制游戏定义的辅助发送：

- `AK::SoundEngine::SetGameObjectAuxSendValues()`

使用该函数可以针对特定游戏对象来定义输出至辅助总线的发送音量。通常也被称为湿声音量。

- `AK::SoundEngine::SetGameObjectOutputBusVolume()`

使用该函数可以定义输出总线音量。通常也被称为干声音量。

但要启用某个对象的游戏定义辅助发送功能，必须激活 Use game-defined auxiliary sends 选项。

启用游戏定义的辅助发送

1. 在 Project Explorer 中双击某个对象可察看该对象
2. 在 Property Editor 中，转至 General Settings（常规设置）选项卡。
3. 在分组框 Game-defined Auxiliary Sends 中：
 - 如果允许，则点击 Override parent。
 - 点击 Use game-defined auxiliary sends。

通过为特定对象启用游戏定义的辅助发送，您可以控制哪个对象将会受其影响。在使用游戏定义的辅助发送来控制环境效果的场景中，您将能够控制哪个对象受环境效果的影响，哪个对象不受影响。

当游戏定义的辅助发送被激活时，您可以直接在 Wwise 中更改游戏定义的辅助发送音量，从而对游戏发送音量值（该值将与游戏定义的值叠加）进行调整。

将游戏定义的辅助发送与衰减结合使用

游戏定义的辅助发送可以与对象定位属性中所定义的衰减设置结合使用。衰减设置用来根据听者与游戏对象之间的距离来控制以下游戏定义的属性：

- Game-defined send volume：用来基于距离来控制衰减多少湿声信号。
- Output bus volume：用来基于距离来控制衰减多少干声信号。

使用响度归一化或补偿增益来调节音量

您可以使用 Make-up Gain（补偿增益）和 Loudness Normalization（响度归一化）来调节各个音频源的音量。与 Wwise 中的其它音量控制不同，这些音量控制对所有基于音量的逻辑行为都是不起作用的。例如，无论声部是否被认为低于阈值（虚拟），改变它们都不会产生影响，另外在进行 HDR 信号衰减时也会忽略它们。它们也不会显示在 Voice Monitor 中。

这些针对声音源的增益总和会显示在 Wwise Profiler 中 Voices 选项卡的 Normalization / Make-Up Gain 列中。

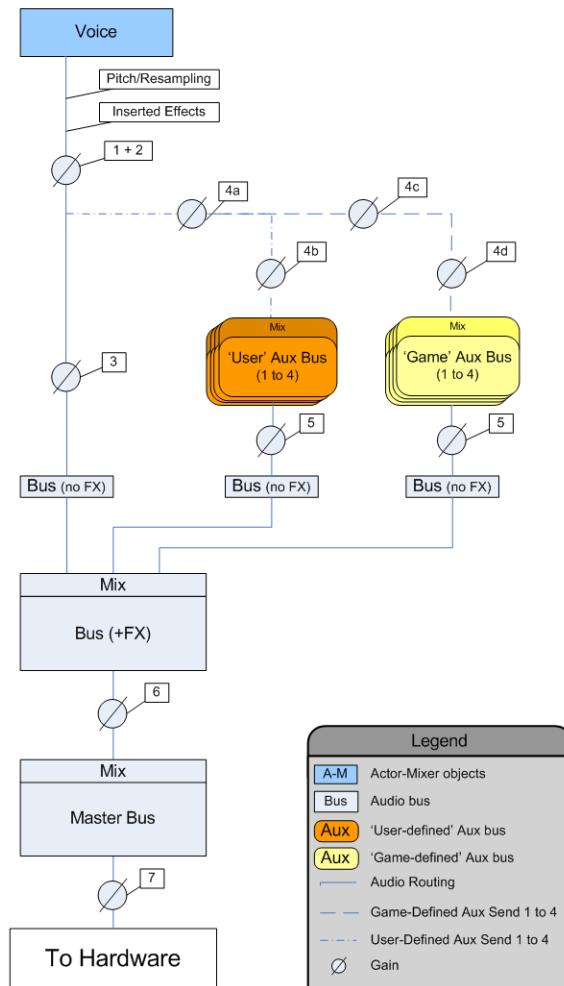
源归一化

Wwise 会使用在分析原始文件阶段所收集的响度数据，在运行时根据预估响度自动归一化您的素材。响度估算会大致遵循 ITU-R BS 1770 响度测量建议：采用 K 权重滤波，绝对和相对门限分别为 -70 dB 和 -10 dB，时间窗为 400 ms，75% 交叠。

源归一化是非破坏性的：声音经分析后的响度值由 Wwise 单独存储，并会在运行时应用正确的归一化增益。该增益是响度的函数：增益 = -响度 - 23 [dB]。例如，如果声音经分析后的响度为 -37 dB 并启用了源归一化，则 Wwise 在运行时将应用 +14 dB (+37 - 23) 的归一化增益。如果声音经分析后的响度为 -12 dB，则归一化增益将为 -11 dB (+12 - 23)。换言之，较弱的声音会加强，而较强的声音会衰减。如果您对基于响度计算的归一化增益不满意，则还可以使用 Make-up Gain（补偿增益）做进一步调节。

理解声部管线

以下表格和图示说明如何处理声部（voice）、如何为声部设置通路，以及在哪些环节应用不同的音量和效果。



编号	描述
1	<p>音量应用至所有声道</p> <ul style="list-style-type: none"> 声部音量*（总线上的 Voice Volume 参数以及自动闪避都将作用于这个层级） 归一化（Normalization）和补偿增益（Makeup Gain） HDR 信号衰减 声笼（Occlusion） <ul style="list-style-type: none"> Project Settings 中的“声笼音量曲线” API: SetObjectObstructionAndOcclusion() API: SetScalingFactor() <p>LPF 属性和声笼 LPF 曲线（工程设置）</p>
2	<p>各声道独立音量</p> <ul style="list-style-type: none"> 定位 <ul style="list-style-type: none"> 2D 声像摆位和 3D 定位 中心% Distance Attenuation Curve（距离衰减曲线） API: SetGameObjectOutputBusVolume() API: SetPosition() API: SetMultiplePositions() API: SetListenerSpatialization()
3	<p>“干声路径”</p> <ul style="list-style-type: none"> 输出总线音量 (+ RTPC) 距离衰减曲线中的“Output Bus Volume”（输出总线音量） API: SetGameObjectOutputBusVolume() API: AkSpeakerVolumeMatrixCallbackInfo() 声障（Obstruction） <ul style="list-style-type: none"> Project Settings 中的“声障音量曲线” API: SetObjectObstructionAndOcclusion() 未插入效果的父级总线：音量* <p>声障 LPF 曲线（Project Settings）</p>
4	<p>“湿声路径”</p> <ul style="list-style-type: none"> 4a - 距离衰减曲线中的“Aux Send Volumes”（辅助发送音量） <ul style="list-style-type: none"> API: AkSpeakerVolumeMatrixCallbackInfo() 4b - 用户定义的辅助发送音量 (+ RTPC) 4c - 游戏定义的辅助发送音量 (+ RTPC) 4d - API: SetGameObjectAuxSendValues()
5	<ul style="list-style-type: none"> 总线音量** 定位 <ul style="list-style-type: none"> 2D 声像摆位和 3D 定位 Center % Distance Attenuation Curve（距离衰减曲线） API: AkSpeakerVolumeMatrixCallbackInfo() 未插入效果的父级总线：总线音量**
6	<ul style="list-style-type: none"> 总线音量** 定位 <ul style="list-style-type: none"> 2D 声像摆位和 3D 定位 Center % Distance Attenuation Curve（距离衰减曲线） 未插入效果的父级总线：总线音量**

编号	描述
7	• 总线音量**

* 声部音量 = 音量滑杆 + RTPC + 状态 + Set Voice Volume Action (动作)

** 总线音量 = 滑杆 + RTPC + 状态 + Set Voice Volume Action (动作)

相关主题

- [了解声部性能分析视图](#)
- [使用 Voice Inspector 分析声部](#)

内置音频设备

对于每个支持的平台，Audio Device（音频设备）代表平台提供的所有实际输出和虚拟输出。默认情况下，Wwise 支持多种音频设备，有些设备仅适用于部分平台。如果您的游戏针对功能不同的多个平台，则可能需要使用 [Link/Unlink](#) 功能将音频分别定向到正确的输出。该操作可以通过更改 Master-Mixer Hierarchy 中任何主总线属性页内的 Audio Device 属性来完成。

要查看默认音频设备的平台特定详情，请参阅 SDK 文档中的 [Default Wwise Audio Devices](#)。

您也可以安装第三方插件来添加其他虚拟输出。请参阅 [Audio Device 插件](#)。

有关支持的输出，请参阅以下列表。

System

系统提供的默认音频输出。这是每个平台上播放所有音频的输出。

Communication

专为聊天或远程通信而设计的输出。可以是耳机或虚拟设备。以下平台均支持该输出：

- Windows
- Xbox One
- PS4

DVR Bypass

有些平台具有 DVR 功能，允许玩家录制游戏过程并发布。这带来了一些法律问题，涉及到游戏音频中有受版权保护的音乐，和用户可替换音乐中受版权保护的音乐。虽然游戏工作室有权在其游戏中使用该音乐，但终端用户无权以任何形式发布该音乐。因此，平台要求中通常规定不允许录制用户背景音乐。此虚拟音频设备将进行单独混音，而不发送到 DVR，但之后会在主输出中与 DVR 进行混音。以下平台均支持该输出：

- Xbox One
- PS4

Game controller speaker

有些平台的游戏手柄上有扬声器，每位玩家都可以有单独的输出。以下平台均支持该输出：

- PS4

Game controller headphones

某些平台可以将耳机连接到游戏控制器，每位玩家都可以有单独的输出。以下平台均支持该输出：

- Xbox One
- PS4

Auxiliary (辅助)

仅支持 PlayStation 4 平台。请参阅制造商的文档和 [SDK documentation](#) 以了解更多信息。

无输出

该虚拟设备不进行输出，可以用于测试，并支持所有平台。

相关主题

- [Audio Device 插件](#)
- [第 9 章 建立输出总线的结构](#)
- [Secondary Bus Hierarchy](#)
- [集成 Secondary Outputs](#)

理解 Secondary Output

术语“Secondary Output”（二路输出）指的是主电视或主扬声器之外的任何音频输出。对于这些输出，必须根据不同情况独立完成音频混音。对于游戏中的各个可用输出，您都可以按需创建足够多的主二路输出总线。最常见的二路输出是游戏控制器上的扬声器或耳机。也存在其它独立的输出（聊天、背景音乐、耳机等）。本章的下面部分将围绕游戏控制器扬声器进行讨论，但也可以应用至其它所有类型的输出。

要针对音频使用二路输出，需要采用以下两种方法之一将声音连线至 Master Secondary Bus 层级：

- 将声音的 **Output Bus** 属性直接设置为二路输出层级中的总线。这与其他声音路由的工作方式相同。对于通常只与一个二路输出相关联的声音，这是首选方法。例如，玩家发起的枪声，网球拍，PDA 声音和玩法反馈等。

- 将声音输出至 Master Audio Bus 层级中的总线，并将 Game\User-Defined Auxiliary Send 发送到二路输出总线层级中。如果希望相同的声音在多个输出和电视中都能同时听到，则推荐使用此方法。例如间谍相机和广播等。

总线层级只是混音时使用的层级结构。要将该混音结构与特定输出相关联，请在对应的 Master Bus 上选择相应的 [Audio Device ShareSet](#)。

必须要注意的是，对于游戏中多个玩家可能同时使用的输出端（如游戏控制器、耳机等），相关混音层级结构会针对每个玩家进行复制。对于特定类型的输出，工程定义的仅仅是特定的混音“处方”。实际信号输出到该结构的哪个副本则取决于听者和游戏对象的关系，是由程序员设置的。以下示例对此进行了说明。

- [二路输出示例](#)
- [Background Music 与 DVR 混音示例](#)
- [第三方和自定义 Audio Device](#)

二路输出示例

Secondary Bus 层级的所有示例都将使用以下所示的总线结构。Main Audio Bus 层级将被省略。对于每路输出，该结构都将被复制，因此最终的混音是独立的。请注意，只有所需的总线将被实例化。

图 30.1. Master Secondary Bus 结构

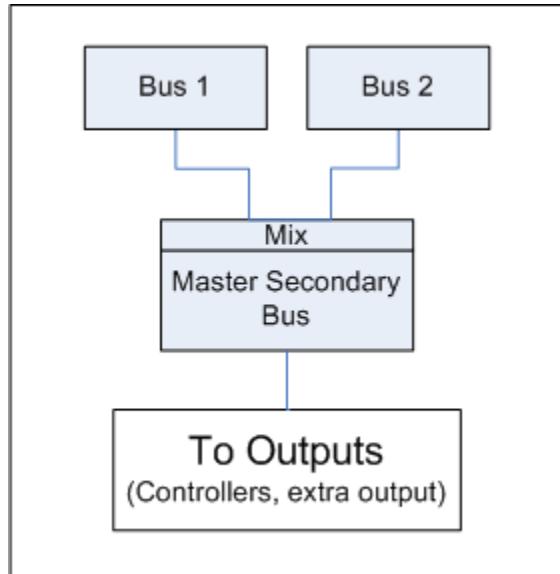
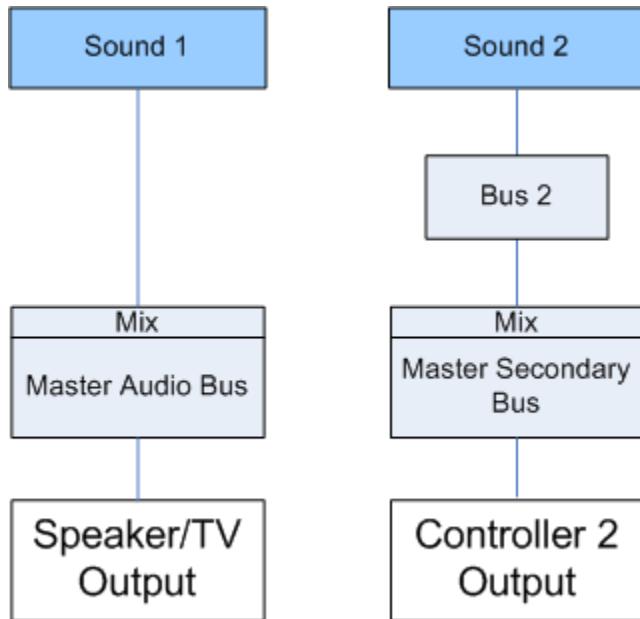


图 30.2. 声音进入主输出的简单示例



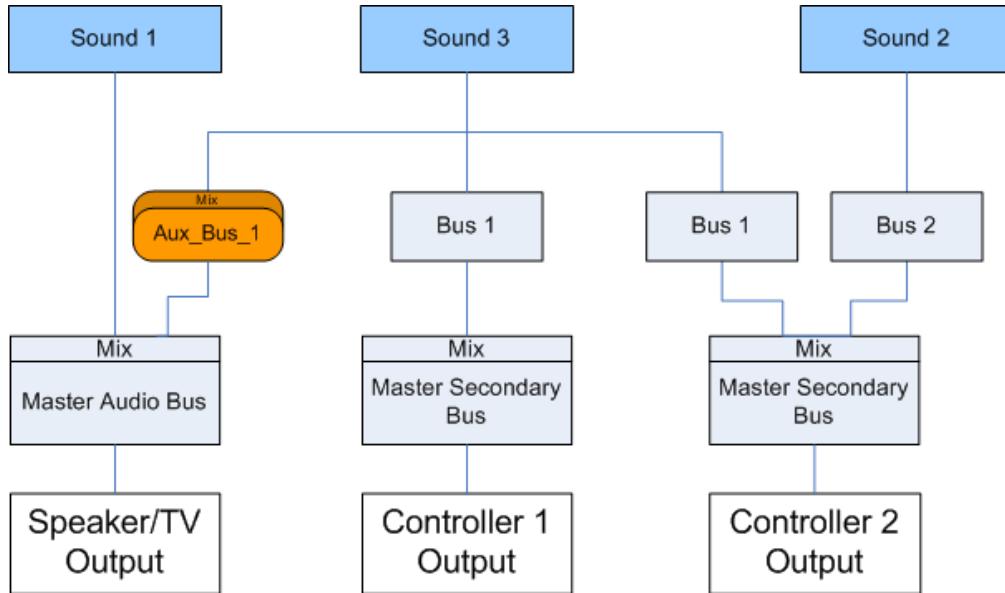
这是进入主输出的一个声音的简单示例。在常规游戏中，很多声音将被连接至多个子总线。但为简单起见，让我们假设电视输出中仅播放一个声音。

图 30.3. 声音进入第二个玩家控制器输出的简单示例



这次让我们在第二个玩家的游戏控制器中播放另一个声音。这次以菜单声为例，假设该玩家在使用菜单，而其它玩家没有。在这种情况下，声音应该独立输出至其总线中，而不进入电视混音。事实上，通过设置游戏听者与游戏对象的关联，该声音将仅在玩家的 2 号控制器中播放。在这里，程序员播放该声音的游戏对象将仅会被控制器 2 的听者所听到。

图 30.4. 声音传输至玩家 1、玩家 2 和电视。



该示例表明两点：声音可以连线至多个二路输出，并可以连线至电视混音。在本示例中，这可以是玩家 1 发出的预先录制好的无线电求助呼叫，该呼叫可以被玩家 2 听到，因为他们属于同一团队，呼叫声也可以出现在电视中，因为电视角度靠近玩家 2 发出的声音。请注意，该声音的输出总线被设置为总线 1。该总线为玩家 1 和玩家 2 按 Master Secondary Bus 原样进行复制。很明显，玩家 2 听到混音必须包含声音 2（只有玩家 2 可以听到）以及声音 1。Master Secondary Bus 的两个副本将应用相同的效果（如果有的话），但并非在同一音频信号上应用。

您还应注意到，声音 3 发出的 Send 进入电视混音。可以使用所有类型的 Send，以发送至其它输出。在本例中，由于呼叫也可被“摄影机”听到（根据距离和衰减），因此也需要进入主电视混音。该呼叫将从设定的辅助总线“进入”该混音。

Background Music 与 DVR 混音示例

通过 PS4 和 Xbox One 的 DVR 功能录制受版权保护的音乐通常是不允许的。两台主机都提供了特殊的虚拟设备专门管理音乐，来避免用户录制。在 Wwise 中，这也意味着需要在单独的层级中为音乐混音。我们在 Integration Demo 工程中实现了这个功能。您可以查看 NonRecordable_Bus，以及 RecordableMusic 和 NonRecordableMusic 声音的设置。代码方面可以查看 DemoBGM.cpp。

NonRecordableMusic（不可录制音乐）被输出到 Non Recordable Bus（不可录制总线），该总线将其信号输出到 DVR_Bypass 输出中，如下所示：

图 30.5. 路由至 NonRecordable_Bus 的声音

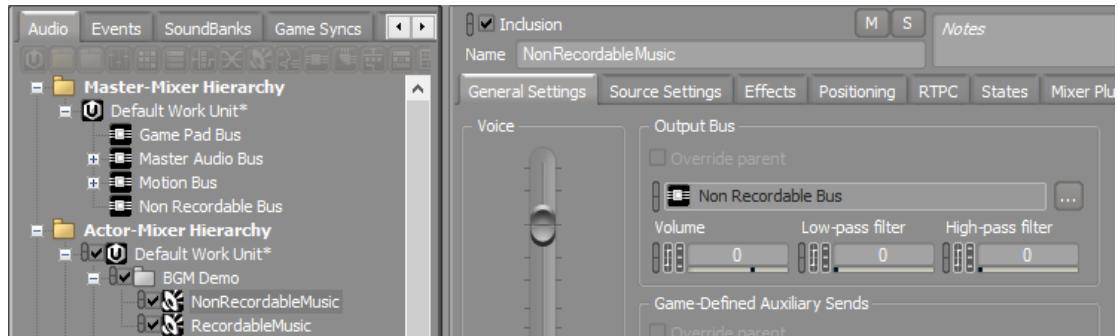
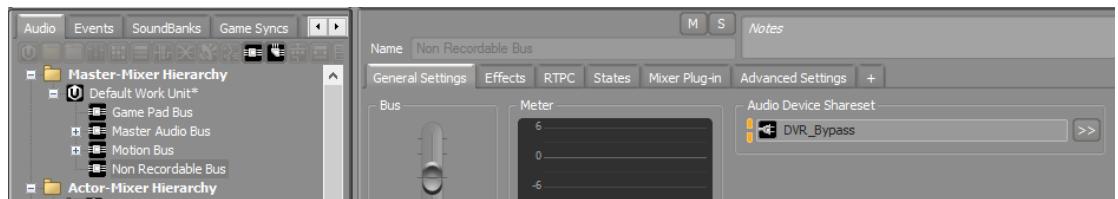


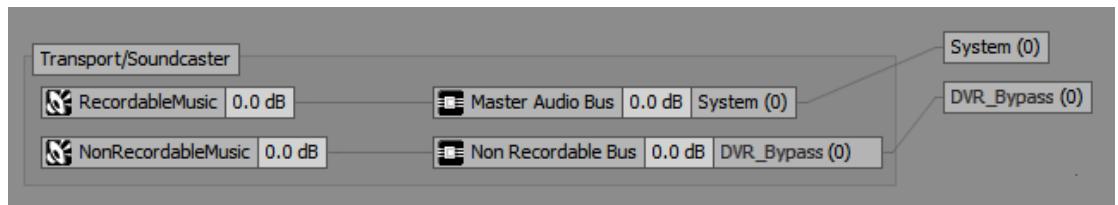
图 30.6. 总线 Audio Device 设置为 DVR_Bypass



在代码方面，为实现该效果所需的处理非常简单。游戏必须使用 Audio Device ShareSet 名称调用 `AK::SoundEngine::AddOutput()` 来初始化输出。

在 Wwise 的 Advanced Profiler 中，我们可以看到每个声音都输出至单独的总线和输出。这是 Xbox One 和 PS4 上的常见情况。（注意，中间的音乐总线被省略了。）

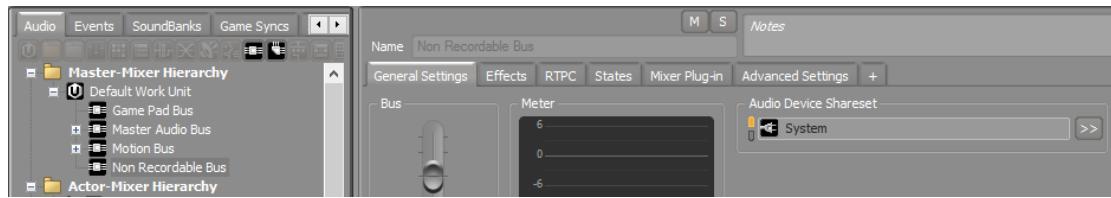
图 30.7.



管理缺失的硬件输出

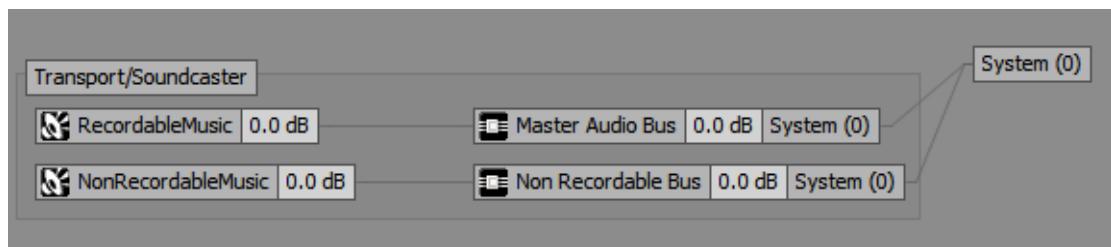
在 Wwise 支持的其他平台上均没有 DVR 功能，所以我们要做一些不同处理。在这种情况下，如果不希望游戏音乐缺失，您必须正确地将其输出。您仅需正确地将音频设备 Unlink（取消链接），即可解决该问题。在 Master Bus 的 Property Editor 中，也允许将 Audio Device 属性取消链接。在下面的示例中，我们将 Non Recordable Bus 输出至普通系统输出。在 Xbox One 和 PS4 中，它仍然输出至 DVR_Bypass，因此它的链接图标有一半是橙色。这就是我们在 Integration Demo 工程中的处理方式。这种情况下，代码不需要调用 `AK::SoundEngine::AddOutput` 来初始化输出，因为系统输出总是在启动时初始化。

图 30.8. 音频设备被设置为系统输出



播放这两种声音时，我们看到两个音频源信号分别通过独立的总线，但是它们最终混合在同一个系统输出中。

图 30.9. 两个 Master Bus 在设备中混合



理解 HDR

高动态范围音频（HDR 音频）是一种利用自然声音所跨越的大动态范围电平值来进行混音设计的技术。HDR 也是一个实时系统，可以将宽泛的电平范围动态地映射至更适合于您的声音系统数字输出的范围。

HDR 也是一个实时系统，可以将宽泛的电平范围动态地映射至更适合于您的声音系统数字输出的范围。HDR 系统的任务是把现实中的整个动态范围从约 190 dB 折叠或“压缩”到 96 dB（数字设备允许的动态范围），而且由于基底噪声的存在，这个范围实际会更低。

在 HDR 图像中，色调映射会独立应用至图像的不同区域，以增强各区域内的对比度。HDR 音频采用相同的工作方式；它进行声音电平映射，而非色调映射，并且在本地实时完成。因此，系统随时都会根据构成音频场景的声音电平来自动采用映射。

HDR 词汇表：

术语	定义
分贝 (dB)	某声音电平与另一声音电平或任意基准值相比较的对数表示方法。处理振幅时，1 分贝等于 $20 * \log_{10}(A/AR)$ 。 $+20$ dB 的差异表示该声音的振幅比参考声要大 10 倍。
满量程分贝 (dBFS)	信号振幅与出现削波前设备可处理的最大振幅相比较的对数表示方法。0 dBFS 值是数字音频输出所能产生的最响的声音。16 位数字音频输出设备范围从 0 dBFS 下至 -96 dBFS。因此 Wwise 中 Master Audio Bus 输出的音频信号的电平将在此范围内。

相关主题

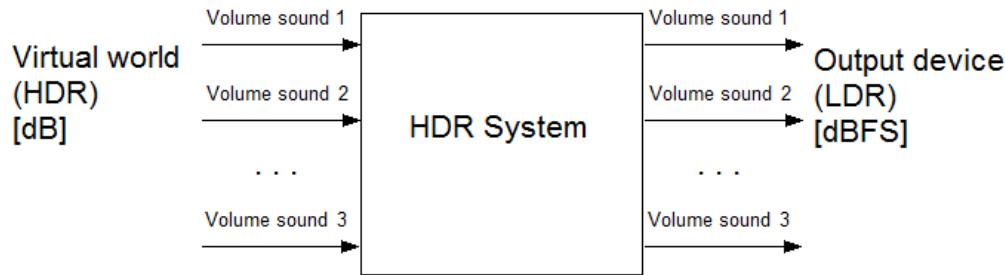
- [HDR 示例](#)

- 使用 HDR
- 使用旁链压缩

HDR 示例

在 HDR 音频中，您可以为游戏虚拟世界的声音指定音量值，这些声音涵盖的动态范围要远大于标准 96 dB 16 位输出设备，更像在真实世界的声音。HDR 系统的任务是将这些值转化为 dBFS，如下图所示。

图 30.10. HDR 系统输入与输出



输入是虚拟世界的声音电平，相对于任意参考值用分贝 (dB) 表示。这些值可以指定任意大小，因此具有高动态范围。而输出电平则是这些声音在 dBFS 中相应的电平。输出电平的范围视输出设备而异，动态范围通常要比输入低。

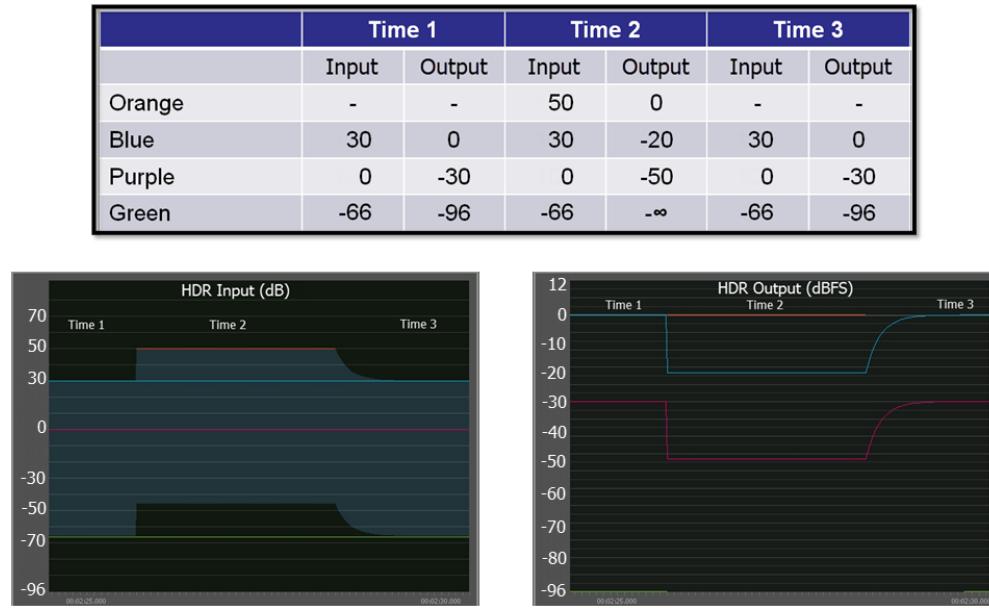
HDR 系统最简单的形式如下：系统在各个时刻会选择当前虚拟世界中被设置为最高音量的声音，将其自动映射至 0 dBFS 输出值，然后按比例映射其它所有声音。

让我们用一个例子来说明。假设在给定时刻（“时间 1”），虚拟世界中的声音“blue（蓝）”以 +30 dB 播放，如图 30.11 “HDR 窗口”中左侧所示。参考值 (0 dB) 为任意设定。因为“blue”是在时间 1 出现的最响的声音，所以它在 HDR 系统的输出端以 0 dBFS 播放。另一个声音“purple（紫）”在虚拟世界中以 0 dB 播放，即比声音“blue”低 30 dB。因此，它在 HDR 系统的输出端将以 -30 dBFS 播放。第三个声音“green（绿）”在虚拟世界中以 -66 dB 播放，因此在 HDR 系统的输出端为 -96 dBFS。由于系统的输出动态范围限制在 96 dB，因此“green”的电平与响度声音的下界相对应。在时间 1，低于“green”电平的任何声音都无法听到。

在上例中，虚拟世界（输入端）的动态范围 [-66, +30] dB 称为 HDR 窗口。它由图 30.11 “HDR 窗口”左侧的蓝色区域表示。HDR 窗口具有固定宽度，该宽度由输出设备的动态范围决定。对于 16 位设备，最多为 96 dB，但实际上通常要小一些。在时间 1，+30 dB 的声音是虚拟世界中最响的声音，任何低于 -66 dB 的声音都会听不见，因为该声音低于 HDR 窗口下限。

假设之后在时间 2，另一个声音“orange（橙）”在虚拟世界中以 +50 dB 开始播放。为对应这个更响的声音，HDR 系统将窗口向上滑动 20 dB，因此该声音在输入端的范围现在为 [-44, +50] dB。所有声音随后被映射至新的电平值。+50 dB 的声音以 0 dBFS 播放，+30 dB 的声音以 -20 dBFS 播放，-66 dB 的声音现在低于该窗口下限，因此变得完全听不见了。当“orange”声音在时间 3 停止播放时，窗口会缓慢恢复为原始大小，其它声音也恢复之前的音量。

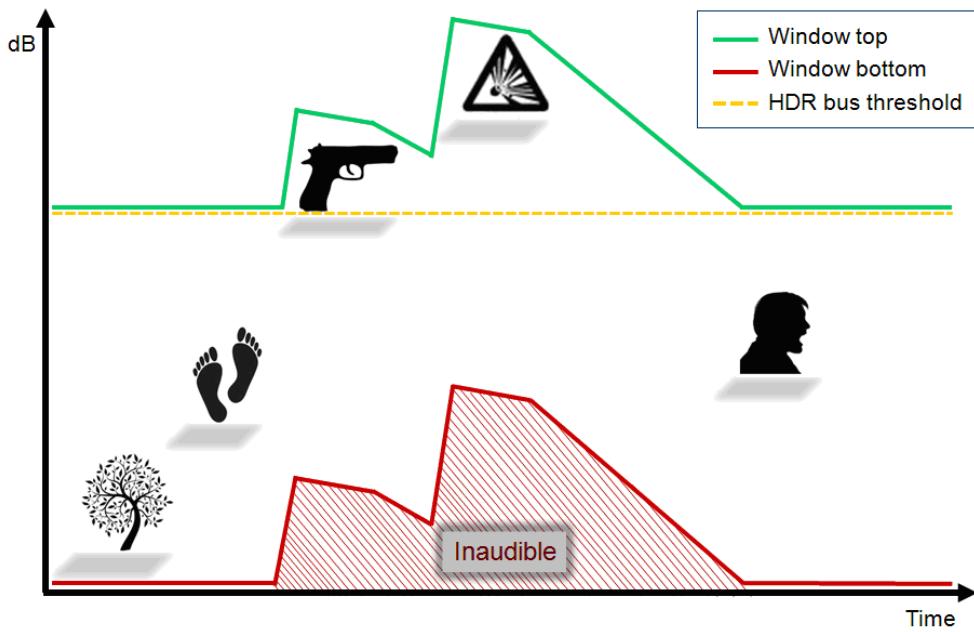
图 30.11. HDR 窗口



在左侧，声音电平以系统的输入来表示，分贝 (dB) 为任意参考值。HDR 窗口由蓝色区域表示。在时间 1，窗口顶部与当前最响的声音（蓝色）齐平。在时间 2，另一个声音（橙色）以 +50 dB 开始播放，窗口立即向上滑动以满足需求。之后 -66 dB 的声音（绿色）明显低于该窗口，因此将无法听到。系统输出得到的 dBFS 电平显示在右侧。当 HDR 窗口由于橙色声音向上滑动 +20 dB 时，其它声音的音量下降 -20 dB。在此期间，输出信号中将完全排除绿色声音。请注意，橙色声音在时间 1 期间的输出电平与蓝色声音相同。当橙色声音停止播放时，窗口会缓慢向下滑动，恢复至之前的电平，而其它声音的音量则相应增加。

HDR 系统的工作方式与动态范围限幅器/压缩器类似。当播放较响的声音时，较弱的声音会变得听不见，从而影响您的混音，当单独播放时这些声音就又能听见了。HDR 中声音之间的相对电平会保留，以产生更大动态范围的幻觉，但实际上它们会被压缩至输出设备的较低动态范围内。此外，由于在播放较响声音时系统会进行自动音量闪避，因此您的混音将变得更清晰，并重点更明确。下图对该原则进行了说明。

图 30.12. HDR 概述，如系统输入端所见



仅当播放较响的声音时窗口向上滑动当输入端窗口向上滑动时，输出端音量则会下降。当播放枪声时，之前响度的声音（如树叶声）变得完全听不到。系统输出端声音的实际音量取决于当前时刻声音与窗口顶部之间的距离。此时，如果单独播放，则枪声和爆炸声将以相同的电平从系统中输出，但由于爆炸声被认为要比枪声响并会有效降低枪声的音量，因此听者会留下爆炸声更响的印象。

使用 HDR

在Wwise中，用户需要选择总线，以将其用作HDR电平与满量程（设备）音量之间的转换器。该HDR总线使用的输入电平是您在Wwise中设置的逻辑电平。因此，连线至HDR总线的声音可将其音量设置为远超过0 dB。而唯一被考虑的因素是其在HDR窗口中的位置，该位置由HDR总线根据正在播放的内容动态放置。因此，HDR总线通过提供与音频限幅器/压缩器类似的控制，起到了逻辑限幅器/压缩器的作用。它具有起止响应时间曲线（极小的启动时间、用户定义的释放时间）以控制HDR窗口如何及时滑动。它也拥有一个阈值，该阈值可被视为HDR窗口可以滑至的最低位置。



dB SPL 的使用

在过去的文献中，HDR音频系统通常将输入端音量以声压级单位(dB SPL)来表达。dB SPL以分贝为单位，其参考值(0 dB SPL)与人类的听觉阈值相对应。Wwise中没有SPL的概念，因为它添加了不必要的复杂性、让界面凌乱，且不会让系统变得更为实用。输入端参考值为任意值，由您来对其进行定义。如果您要使用dB SPL，则可以将声音结构音量直接设置为正数dB SPL值。另一方面，Wwise中的音量推子的默认范围最高仅为+12 dB，因此更为实际的做法是选择另一个参考值，然后进行减法运算，为所需的SPL值找到对应的分贝值。例如，您可能决定将100 dB SPL作为0 dB参考值。则80 dB SPL声音的音量推子应该被置为-20 dB，而130 dB SPL声音的音量推子应该被置为+30 dB，以此类推。相应地，您还需要设置HDR总线阈值。

- 在工程中启用 HDR
- 将声音连通到 HDR 总线

- 监视窗口
- 设置 HDR 的动态

在工程中启用 HDR

想要在工程中启用 HDR，您只需定义哪些 Audio Bus 将作为 HDR 子系统。

为 Audio Bus 启用 HDR 的方法如下：

1. 加载 Audio Bus。
2. 点击 Property Editor 中的 HDR 选项卡。
3. 点击 Enable HDR。

所有连线至 HDR 总线（或其任一子总线）的声音结构，其音量都将受到 HDR 总线影响。没有连线至 HDR 总线的声音将不受影响，但它们仍可以在您设计的系统中与连线至 HDR 总线的声音共存。工程可以存在多条 HDR 总线，只要这些总线完全独立即可：HDR 总线不能以其它 HDR 总线作为上级或下级。

如上所述，HDR 窗口顶部的声音将以 0 dBFS 输出。使用 HDR 总线的总线音量推子按比例来缩小 HDR 系统的输出后，再将其与工程中的其它非 HDR 声音进行混音。

将声音连通到 HDR 总线

要让 Actor-Mixer Hierarchy 或音乐层级中的声音对象进入 HDR 系统，您需要将这些对象连线至 HDR 总线。

将声音连线至 HDR 总线的方法如下：

1. 选择声音对象以察看其内容。
2. 在 Property Editor 中，转至 General Settings（常规设置）选项卡。
3. 选择 HDR 总线作为 Output bus。

监视窗口

Voice Monitor 视图显示声部的音量及其包络（如果可见）。

理解并调试 HDR 系统的方法如下：

1. 打开 Voice Monitor 视图。
2. 将 HDR 总线拖入视图中。
3. 将 Mode 设置为 Bus input 或 Bus output。Bus input 模式以分贝来显示在进行 HDR 压缩前，HDR 总线输入端的声部电平；而 Bus output 模式则显示在 HDR 压缩和总线输出增益之后，HDR 总线的输出电平。本文档中的大部分图示都是 Voice Monitor 视图的“两端”的屏幕截图。

设置 HDR 的动态

每条 HDR 总线均有独立的 HDR 窗口，其位置由当前最响声音的电平决定，而窗口宽度则由和工程的音量阈值决定。HDR 窗口的行为可在各条 HDR 总线的 HDR 选项

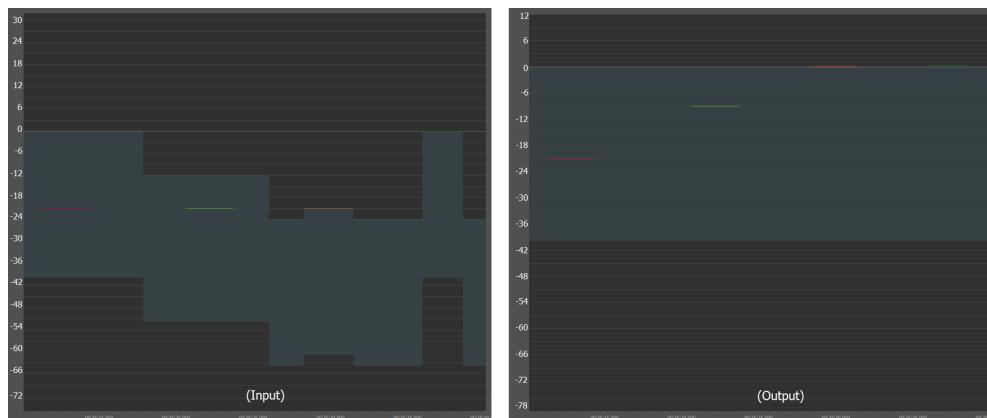
卡中进行编辑。由于 HDR 总线的作用与音频限幅器/压缩器类似，因此其控制参数也都相似。

- 使用 HDR 阈值
- 使用 HDR 起止响应时间
- 使用 HDR 压缩比率

使用 HDR 阈值

与典型的限幅器/压缩器的阈值类似，HDR 阈值定义了最小输入电平，超出该输入电平则可能导致 HDR 窗口滑动（参考 [图 30.12 “HDR 概述，如系统输入端所见”](#) 中的黄线），换言之，超出该电平后压缩器将会“启动”。仅播放较弱的声音时，这会直接影响输出时的电平：距离窗口顶部越远，这些声音的输出电平越低。当比阈值高的声音被播放时，较弱的声音会自动向下闪避。理解这一点很重要，即采用无限压缩率时（我们将在稍后讨论这一比率），任何两个输入电平高于该 HDR 阈值的声音在单独播放时都将以相同的输出音量播放，而不管其在 [图 30.13 “HDR 阈值的效果”](#) 中显示的输入音量是多少。

图 30.13. HDR 阈值的效果

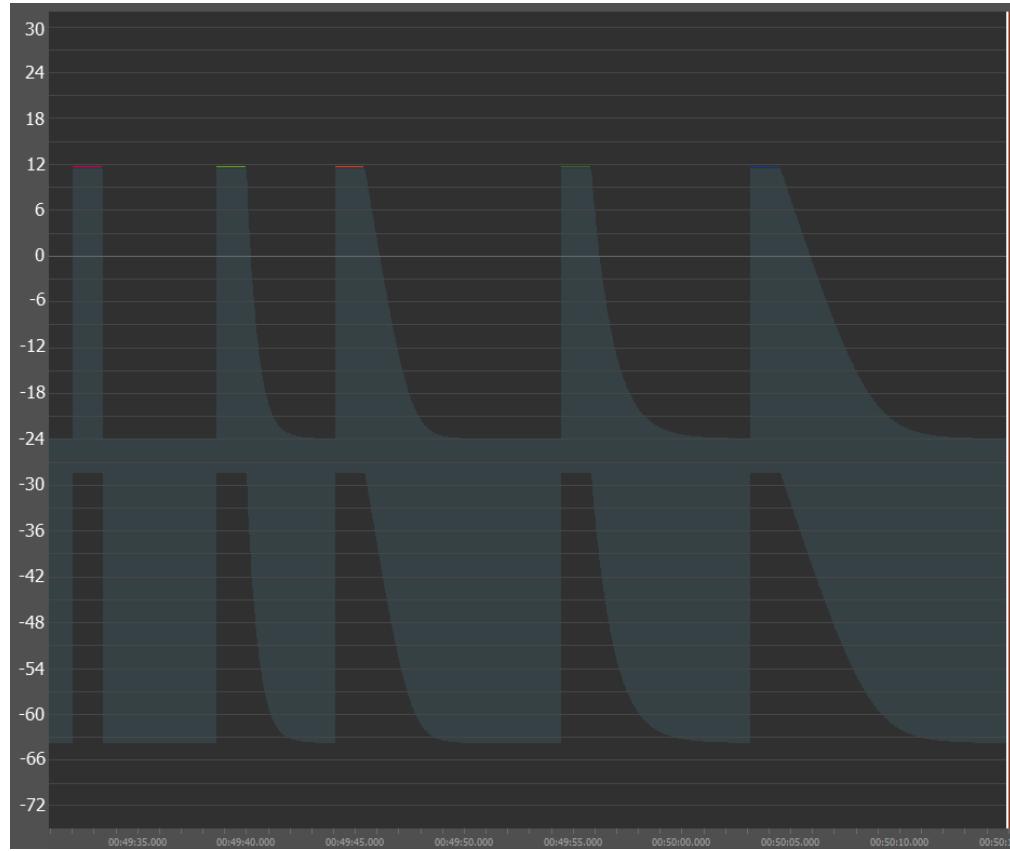


在上图中，左侧显示输入；右侧显示输出。第一次，声音设置为 -21 dB，阈值为 0 dB，以 -21 dBFS 输出。第二次，阈值为 -12 dB，声音以 -9 dBFS 输出。第三次，阈值为 -24 dB，声音高于阈值，因此促使窗口滑动并以 0 dBFS 输出。当声音以 0 dB 重播，输入音量高出阈值，将仍以 0 dBFS 输出。

使用 HDR 起止响应时间

标准音频压缩器采用起止响应时间（启动/释放时间），用于定义压缩的时间特性。同样地，HDR 总线可以由用户控制释放时间和形状，由您定义 HDR 窗口如何恢复至较低值。[图 30.14 “HDR 释放：0、0.5 指数、0.5 线性、1 指数、1 线性”](#) 显示了各种释放时间和模式。对其进行调整，以最小化副作用，如 pumping（突变声）。请注意，没有启动时间的控制器，因为 HDR 系统的启动必须是瞬间的，就像音频限幅器一样。

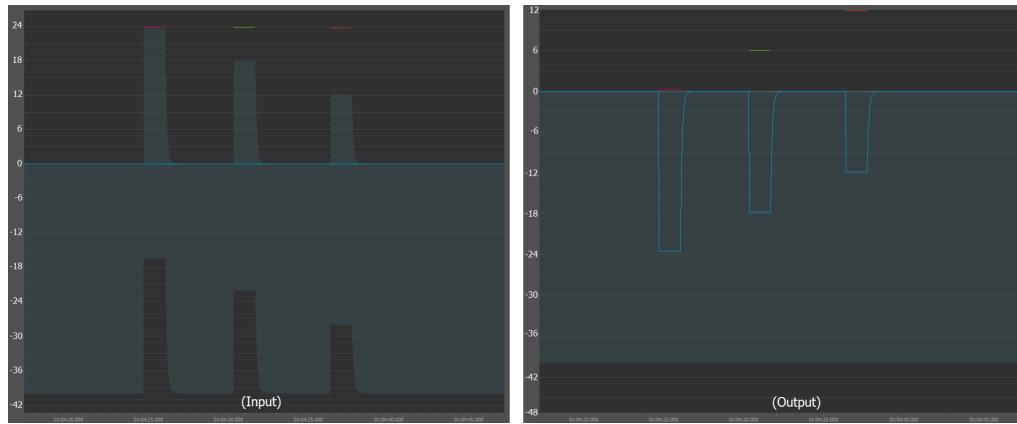
图 30.14. HDR 释放：0、0.5 指数、0.5 线性、1 指数、1 线性



使用 HDR 压缩比率

到目前为止，我们一直在用音频限幅器来比拟HDR系统。HDR窗口顶部严格等同于最响声音的电平，各个高于HDR阈值的声音在单独播放时，都将以相同的输出电平播放，如图30.13“[HDR 阈值的效果](#)”中所示。如上所述，HDR会增加各声音之间的主观动态范围，而这会使“绝对”动态范围（输出端能够呈现的动态范围）变得更小。通过将HDR限幅器转换为HDR压缩器，可以重新获得部分“绝对”动态范围。您可以使用有限的、更小的压缩比率来实现（默认情况下是无限的）。压缩比率定义了HDR窗口是否严格跟踪最响声音电平，它是关于声音电平与HDR阈值差值的函数。[图30.15“\[HDR 压缩比率的效果\]\(#\)”说明该压缩比率带来的效果](#)。请注意，采用非无限压缩比率时，较响的声音可能会超出HDR窗口，因此以高于0dBFS的值从HDR系统输出。可以通过降低HDR总线或信号路径中其它下游总线的音量，给自己留出足够的动态余量。

图 30.15. HDR 压缩比率的效果



声音的输入电平为 +24 dB（左侧面板），在 HDR 总线中分别以 100:1、4:1、2:1 的比率播放。右侧显示相应输出电平为 0 dBFS、+ 6 dBFS 和 +12 dBFS。另一个声音输入电平为 0 dB，它将以 -24 dB、-18 dB 和 -12 dB 向下闪避。比率为 2:1 时，阈值以上的能量只有一半用于向上推动 HDR 窗口，因此其它声音将以 -12 dB 闪避。请注意，两个声音自始至终保留 24 dB 的电平差。同时，在压缩比率较小时，声音可以超出窗口顶部，因此会高于 0 dBFS，您可以通过降低总线或其父总线的音量保留足够的动态余量。

使用振幅包络

由于 HDR 系统采用 Wwise 设置的逻辑音量，因此它并不知道输入声音的实际振幅，因此会在整个声音的持续期间将其视为音量恒定的黑盒。在绝大部分时间，声音的振幅都不同。设想一下具有瞬态峰和衰减过程的撞击声。如果该声音足够响，可以导致 HDR 窗口的位置滑动，那么在整个声音持续期间窗口将很不自然地恒定保持在该位置。在 [图 30.16 “带衰减的撞击声的 HDR 窗口，无包络”](#) 中查看对窗口位置的影响。

在 Wwise 中，通过启用包络跟踪，可使 HDR 系统看到黑盒内的部分。在所需声音属性的 HDR 选项卡中，选择 Enable Envelope 复选框：随后会对音频文件的幅度包络进行分析，并将其附加至声音的元数据。在运行时，HDR 系统使用该数据相应地移动窗口，如 [图 30.17 “开启包络的带衰减的撞击声的 HDR 窗口，在较弱而稳定的背景音上播放。”](#) 中所示。

在下图中，输入位于左侧，输出位于右侧，相应的输出波形位于下方。

图 30.16. 带衰减的撞击声的 HDR 窗口，无包络

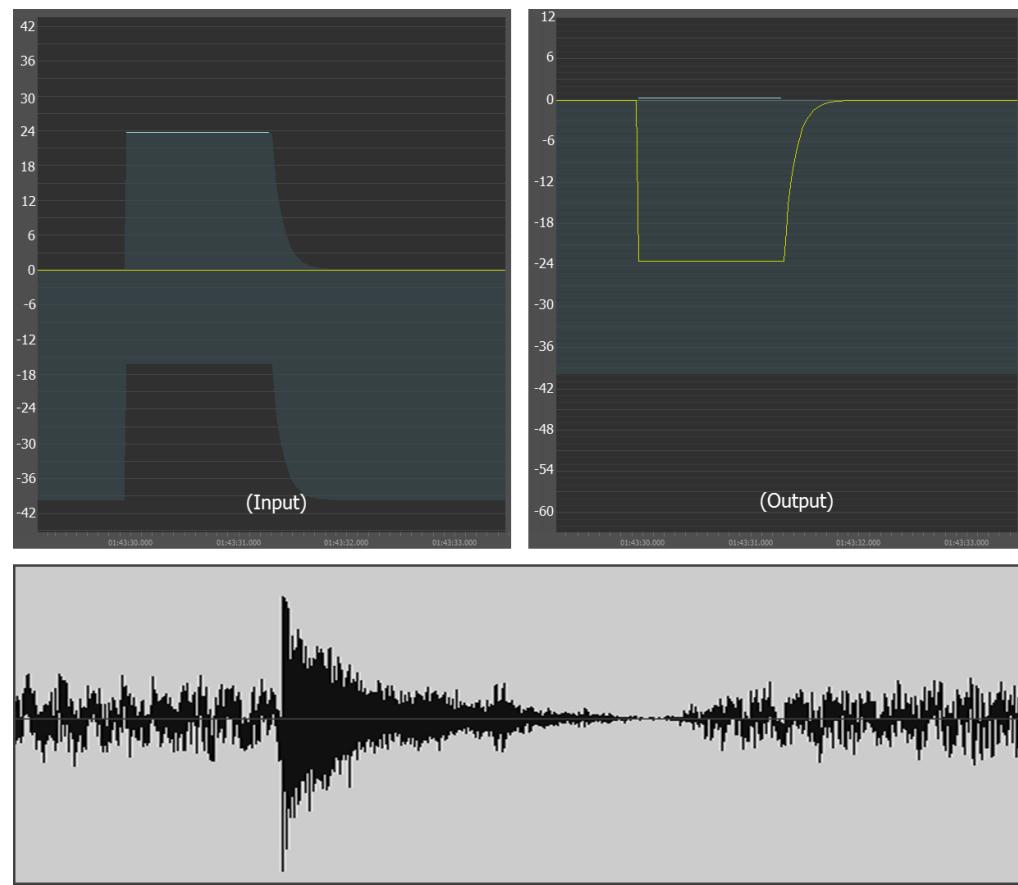
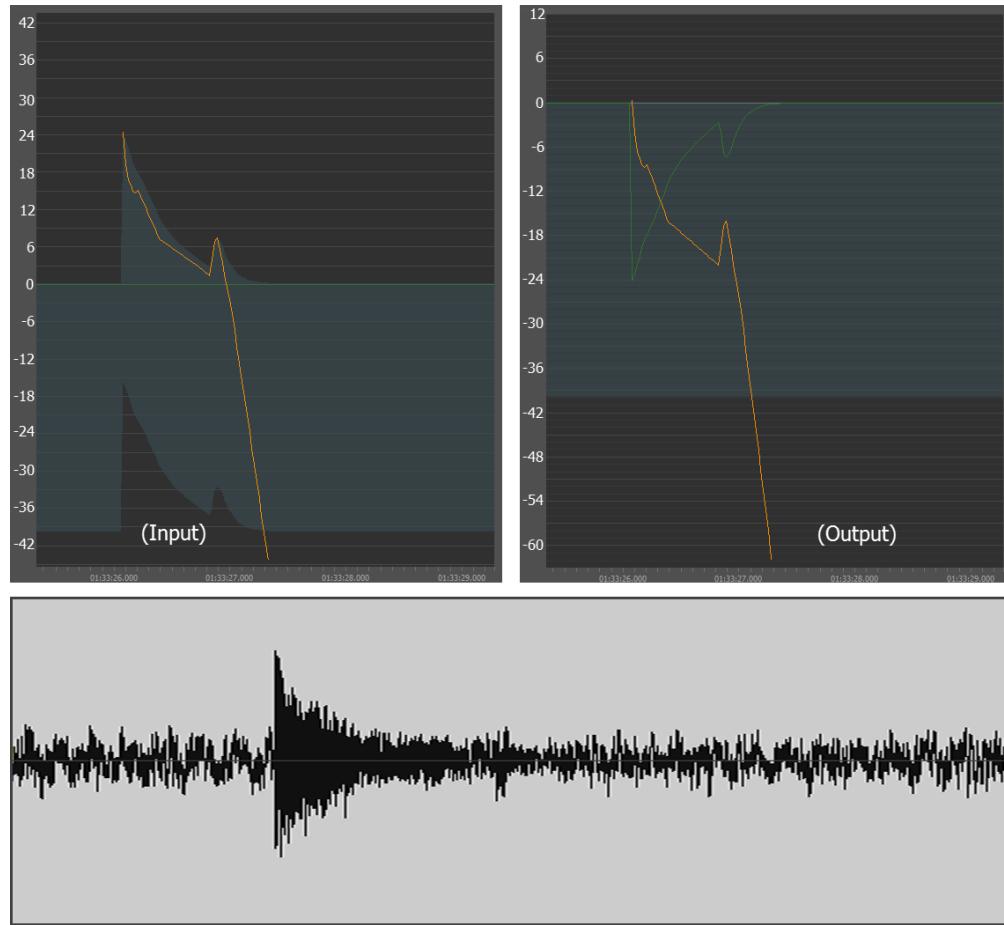


图 30.17. 开启包络的带衰减的撞击声的 HDR 窗口，在较弱而稳定的背景音上播放。



请注意上方第一张图中撞击声后有一段间歇，这是由 HDR 系统将其音量视为恒定所导致的。

包络灵敏度和手动编辑

启用包络后，可通过打开源编辑器（下图）来预览结果。如果您对精确度或点的数量不满意，则可以更改 Envelope Tracking（包络跟踪）分组框中 sensitivity（灵敏度）滑杆的值（请参阅 [图 30.19 “Property Editor 中的 HDR 选项卡”](#)）。可以在角色混音器这样的父级声音结构中，为一组声音启用包络和设置灵敏度设置。请注意，灵敏度的效果根据原始音频数据而异，因此对于相似声音的变体，结果也有可能不同。您也可以手动编辑各个声音的包络：设置灵敏度参数以便获得更接近所需结果的曲线，并可移动、移除已有点或添加新的点。

图 30.18. 源编辑器中开启RMS 模式，并显示包络

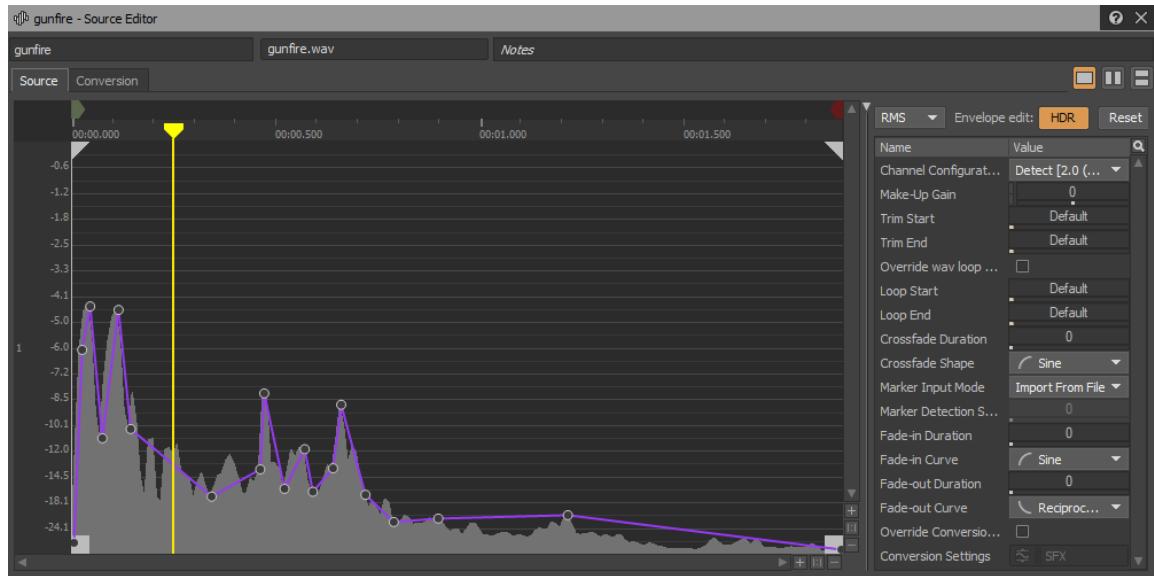
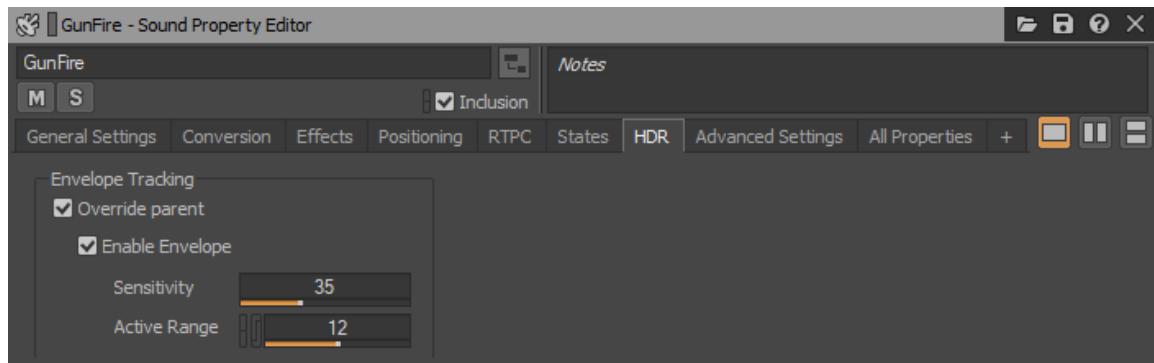


图 30.19. Property Editor 中的 HDR 选项卡

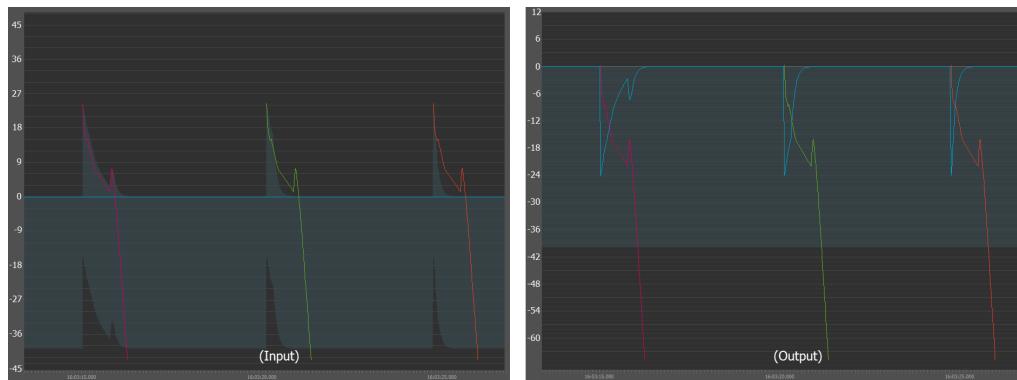


有效区域：活跃范围

在典型的情况下，具有较大电平差的声音（如 0 dB 至 -96 dB 范围内的撞击声）在其整个持续期间内不应全程驱动 HDR 窗口，而仅在有限时间段内驱动 HDR 窗口。这一基本原理基于以下悖论：HDR 是一种基于响度的优先级系统，但较响声音中稍弱的部分不应受到强调。例如：手雷声的衰减拖尾与瞬态峰相比重要性很低，因此不应掩盖霰弹枪声的瞬态峰，即使前者从绝对规模上讲更响亮。在本例中，手雷声的有效区域应限制在其第一个瞬态峰。它将使 HDR 窗口向上移动，因此其它声音会闪避。而一旦该声音进入尾部，比如低于峰值 12 dB 时，我们通常不希望其它声音继续闪避，即使手雷的峰值音量减去 12 dB 但仍高于音频场景中其它声音的音量。在 HDR 图像中，将 HDR 窗口限制在有效区域内是一种优雅地融合各种“色调区域”的方法。

在 Wwise 中，定义有效区域的方法是以分贝为单位来指定从声音峰值开始的一定范围，即活跃范围。当声音的包络降至活跃范围以下时，该声音被认为在有效区域以外，HDR 系统将认为该声音不再能够驱动 HDR 窗口。[图 30.20 “活跃范围”](#) 使用三种具有不同活跃范围的相似声音说明了这一点。

图 30.20. 活跃范围



具有衰减过程的同一声音与稳定的背景声一起播放三次，活跃范围分别设置为 96 dB、12 dB 和 6 dB。当声音从其峰值降低相应分贝值时，窗口将停止跟随该声音，释放并恢复至空闲状态。窗口移动会明显影响背景声的电平。而窗口和活跃范围不会影响前景声本身。

深入了解 HDR

声音的包络只会影响其它声音，而永远不会影响其本身，了解这一点十分重要。这是项特点是 HDR 系统与音频压缩器的最大不同之处。在任何时刻，都会测量最响声音的峰值，但自身不会受其包络影响，这会让 HDR 系统的声音清晰可辨。因此在使用包络时，输入声音之间的电平差将不会被严格保留。这样做的好处是在声景中将各种“自身响度”不断变化的声音混合在一起，正如在之前章节中讨论的一样。但当 HDR 窗口大幅移动时，可能会发现声景有一些失真。在 HDR 图像中，当使用大量范围较小的色调区域时，也会出现类似情形。网络中可找到很多失真的 HDR 照片示例。因此我们建议您小心设置高于 HDR 阈值的电平：不要只是盲目地拉拽推子，而是要仔细设置电平和最响声音的包络，以便在混音时为它们留出空间。

避免启用空白包络，因为这需要占用额外的内存。包络对于较弱的声音并不起作用，因为它们永远不会高于 HDR 阈值，因此请确保禁用它们。

请注意在活跃范围值上下摆动的包络，这可能会导致意外行为。在下图中，一个长的爆炸声包络不断出入活跃范围，促使 HDR 窗口发生快速变动。这种情况下，您可能会需要将包络近似粗化，或手动进行编辑。您甚至可能会需要仅在声音的初始峰值期间压低其它声音，然后在后面的隆隆声期间更快速地恢复，从而对包络进行艺术处理并设计出不完全符合现实的包络。

含有较少点的简单的包络会减少声景中的不稳定振幅变化，通常更为可取。

图 30.21. 编辑包络 (a)

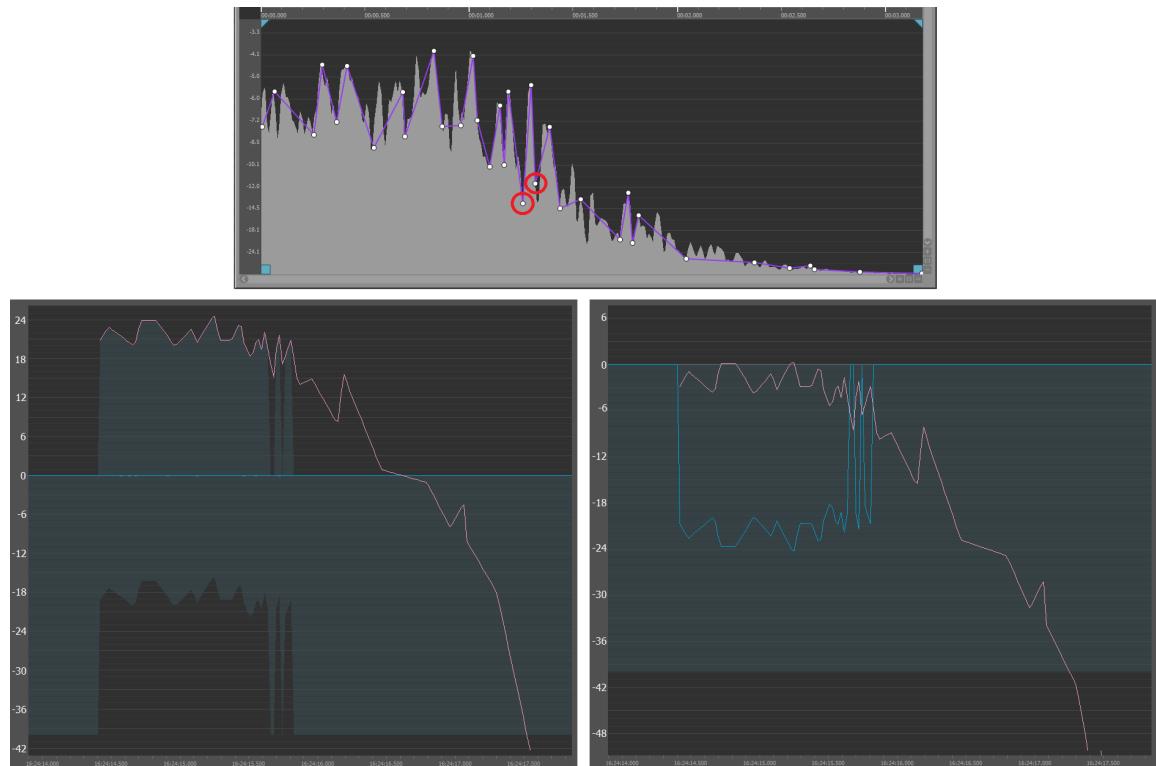


图 30.22. 编辑包络 (b)

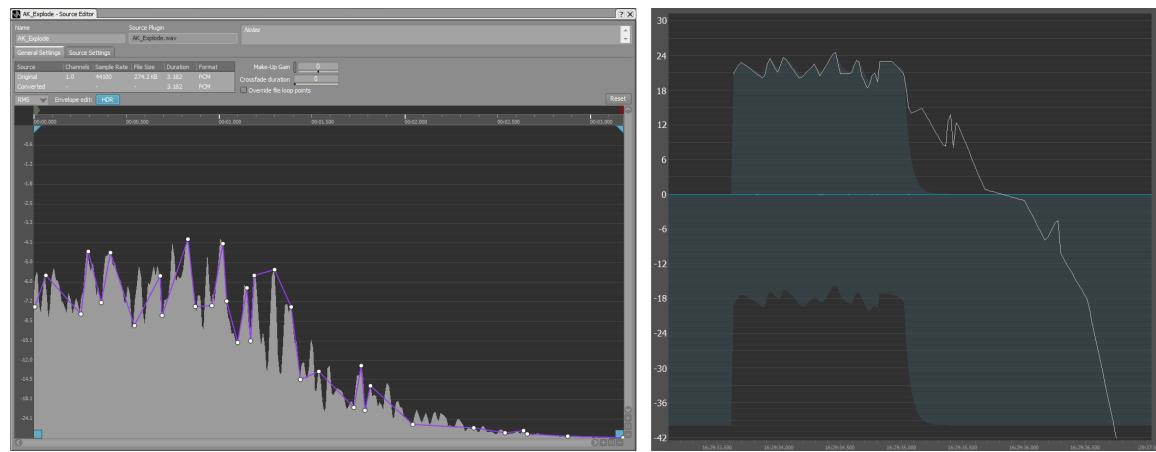
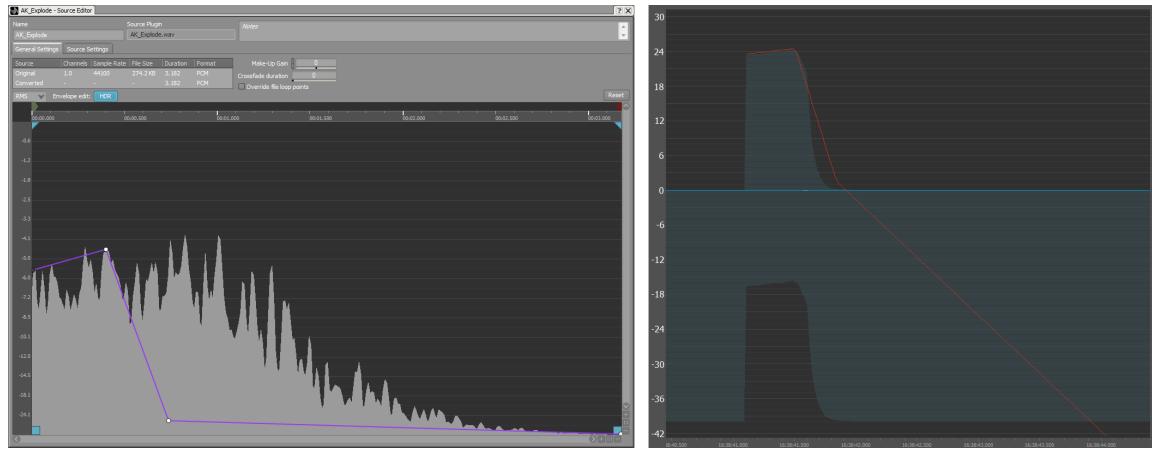


图 30.23. 编辑包络 (c)



在(a)中，源编辑器中显示了爆炸声的包络。红色圈中的点低于活跃范围。播放过程中，包络进出有效区域，导致 HDR 窗口快速移动。输出结果中很明确地显示出背景声受到的严重影响。要修复这个问题，可通过在(b)中手动编辑包络，移除两个突出的点。请记住，编辑爆炸声包络只会影响 HDR 窗口，也即影响其它声音，但不会影响爆炸声本身。出于审美原因，设计师可能还会决定仅在爆炸的初始峰值期间压低其它声音，但通过编辑类似于(c)的包络，在接下来隆隆声衰减时更早地让其它声音恢复正常音量。有趣的是，三个示例中播放的是完全相同的爆炸声。

HDR 和 Wwise 声部管线

如上所述，HDR 系统使用 Wwise 设计工具中所设置的逻辑音量，并会忽略音频数据的实际振幅。HDR 系统逻辑将认为声音的电平等同于 HDR 总线输入端的电平，即以 Input bus 模式察看 HDR 总线时，在 Voice Monitor 中所看到的电平。这些电平取决于声部音量，包括以下环节的音量总和：Actor-Mixer 层级、Master-Mixer 层级的总线、动作、RTPC 和距离衰减，以及信号流中 HDR 总线上游的各条混音总线的增益。当多个信号路径通往 HDR 总线，例如当使用辅助发送时，HDR 系统逻辑将使用这些路径中增益最大的一条。

在每个音频帧中，声音引擎将计算 HDR 总线输入端的所有声部音量。随后会执行 HDR 系统逻辑，它会根据 HDR 窗口的位置来计算全局 HDR 增益/衰减，并应用至各个声部。该步骤完成后，将按照正常流程处理声部：首先根据音量阈值对声部进行评估，决定是否为虚声部，然后处理数据并应用音量。

补偿增益和源归一化

HDR 系统的逻辑会忽略由 Wwise 中的两个音量属性：源归一化和补偿增益。这些音量控制独立于逻辑音量，主要用于归一化音频素材，您也可以在将音频导入至 Wwise 之前，在波形编辑器中执行这些操作。例如，您可以使用本地化补偿增益校正两种语言之间的音量差异，但 HDR 系统可以跨语言实现相同的表现。此外，具有可以绕过 HDR 逻辑的音量控制十分重要，源归一化和补偿增益便发挥着这一作用。

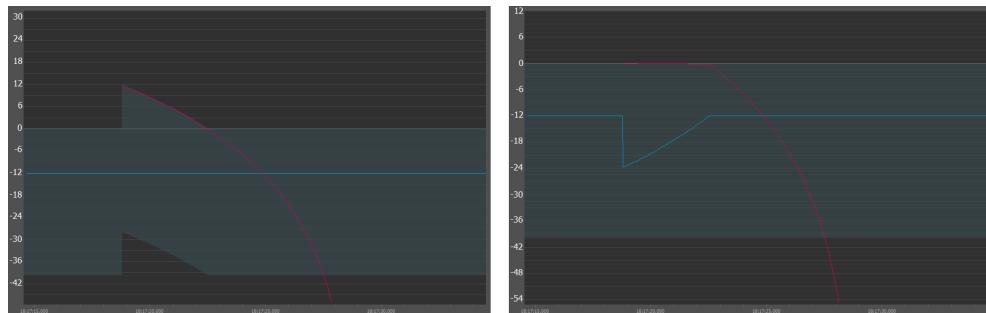
请注意，虚声部系统以及 Voice Monitor 计算也不考虑补偿增益和源归一化。但虚声部会将 HDR 信号衰减考虑在内。因此，当声部低于 HDR 窗口时会变为虚声部。

可以在使用 HDR 时配合使用属性编辑器 Conversion 选项卡中的补偿增益推子，以满足审美需要。由于从 HDR 系统角度而言它是不起作用的，因此可将其视为 post-HDR 音量（就信号流而言），您可以通过它来绕过 HDR 窗口更改音量。通常，以高于 HDR 阈值的音量单独播放任何声音，通过系统后都是以同一电平输出。使用补偿增益，您可以有效地以更大音量进行播放。例如，您可能会需要第一人称射击游戏中玩家的枪声远大于其余枪声，这就可以通过 HDR 系统的动态混音功能实现。在这里，可使用补偿增益加强音量。

HDR 和距离衰减

可以按常规设计距离衰减曲线。但请注意，HDR 系统可能导致意外表现。当您播放高于 HDR 阈值并且是声景中最响的声音时，它会以 0 dBFS 从系统输出。如果该声音播放时距离为 50 米，但仍高于阈值并且仍为场景中最响的声音，那么它将仍然以 0 dBFS 从系统输出。因此会给人留下衰减曲线在近距离时未起作用的印象。但并非如此。在较近距离时，增加输入音量实际用于压低其它声音，而非增加输出音量。如果没有播放其它声音，则不会引起注意，但您应在为较响的声音设计衰减曲线时记住这点。

图 30.24. 距离对较响声音的影响



本例中，听者将稳步远离一个较响的声音（红色）。当较响声音保持高于 HDR 阈值，输入端（左侧）的音量看上去是正确的，输出（右侧）出现停滞，并未随着距离增加而减弱，这样便会留下距离衰减未起作用这一错误印象。而实际上，在距离较近时，超出阈值的音量偏移会影响其它声音（蓝色），而不是较响的声音。

将 HDR 窗口用作输入变量

通过查看总线 HDR 选项卡中的窗口电平计，可以监视 HDR 窗口位置。您也可以为它的值指派一个游戏参数。然后就可以通过 RTPC 将该游戏参数映射至任何对象的任何属性。例如，您可以使用窗口位置来改变声音实例的数量上限。

将游戏参数关联至 HDR 窗口位置的方法如下：

1. 选中 HDR 总线，在 Property Editor 中察看其内容。
2. 点击 HDR 选项卡。
3. 在 Window Top Output Game Parameter 中，点击 [...] 浏览并选择 Game Parameter。
4. 在 HDR 总线输入中设置游戏参数的范围（单位为分贝）。

完成终混

在工程中设置总线结构后，就可以对游戏中声音、音乐和振动的混音进行精确调整和故障排查。您可以通过 Wwise 中的两个主要工具对游戏音频进行混音：

- [Master-Mixer Console](#) - 主混音控制台，它可以将各种总线属性集中到一个视图中。
- [Mixing Desk](#) - 调音台，使用灵活、功能强大的调音台，将各种总线和对象属性集中到一个视图中。

这两种视图都可用于对游戏进行实时音频终混。但 Mixing Desk 而已提供更多控制，因为提供了更多的属性供您编辑，并且对象或总线可以包含在同一个混音会话中。



混音与非混音总线

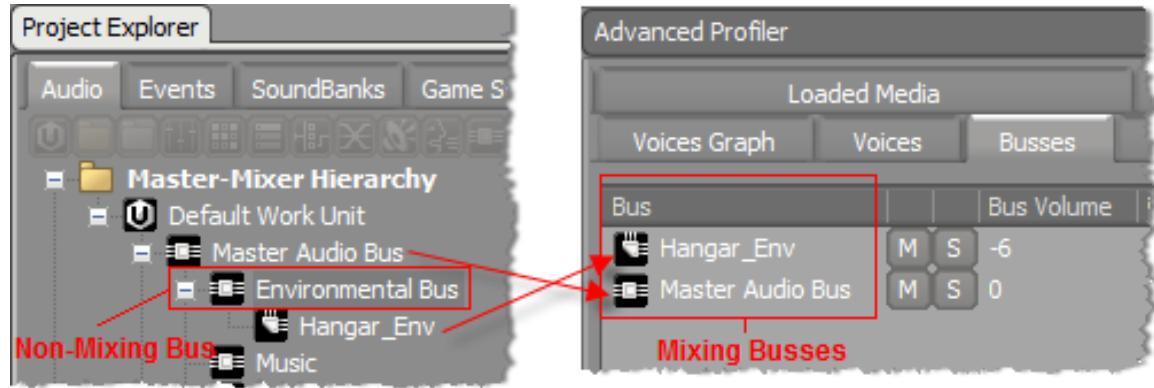
在 Wwise 中，Master-Mixer 层级结构对象（即总线）将作为非混音总线在后台处理，除非需要将其转换为混音总线。非混音总线将其总线属性传递至其输入，因为这样能更高效地使用 CPU 和内存。出于以下原因导致此类传递不起作用时，非混音总线将被升级为混音总线：

- 该总线是主音频总线；
- 该总线是辅助总线；
- 该总线是 HDR 总线；
- 至少包含一个插入效果；
- 启用了混音器插件；
- 其声道配置未设置为 Parent；
- 已启用定位；或者
- Bus Volume（总线音量）上或 Output Bus（输出总线）的 Volume（音量）、LPF、HPF 上设有 RTPC。

Advanced Profiler

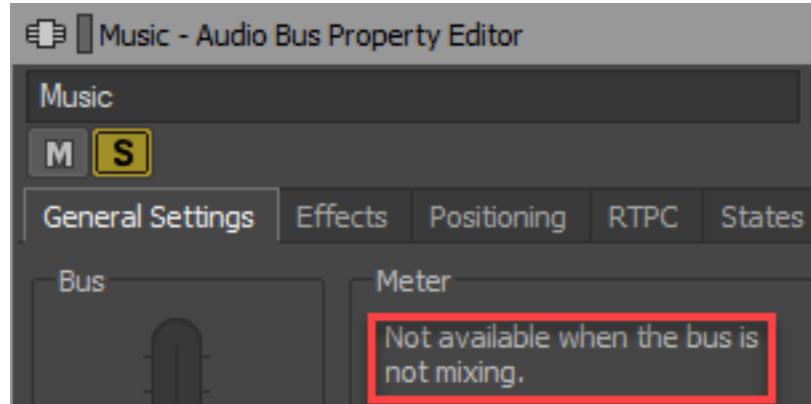
验证总线状态的一种方法是在捕获时查看 Advanced Profiler 的 Voices Graph 选项卡或 Busses 选项卡。此处只显示混音总线；而不显示非混音总线。

如以下 Wwise Integration 工程截图所示，Environmental Bus 是非混音总线，因为它未显示在 Advanced Profiler 的 Busses 选项卡中，而它的子总线 Hangar_Env 辅助总线则会被显示。此外，作为非混音总线，Environmental Bus 的音量包含在 Hangar_Env 辅助总线 Bus Volume 列中。



检查总线对应的 Meter

通过查看 Property Editor (属性编辑器) 的 General Settings (常规设置) 选项卡，可以快速了解总线是否正在混音。若正在混音，则 Meter (电平表) 可用；若未在混音，则 Meter 不可用，同时显示效果器消息。



监视信号电平

在 Wwise 内，可通过不同的方式测量信号。比如响度级。

响度概览

ITU-R BS.1770 和 EBU R128 标准为测量特定音频工程的响度级和响度范围建议和规范，单位是 loudness unit (LU 或 LUFS)。它还为 True Peak (真实峰值) 电平提供了规范和建议。Wwise 的 Loudness Meter 视图和 Meter 视图采用了上述规范和建议。

测量方法

Wwise 提供不同的 PCM 信号测量方法：

测量方法	描述
响度级	测量数字信号感知响度的方法。响度级可以使用不同的时间窗口大小来计算。 响度级可以在 Loudness Meter 视图中找到。 单位：LUFS 或 LU

测量方法	描述
响度范围	对一段时间内响度值的统计分析，是动态范围的单位。 响度范围可以在 Loudness Meter 视图中找到。 单位: LU
True-Peak 电平	真实峰值电平。对模拟世界中最大持续信号电平的估算。它非常实用，可确保数模转换器或外部编解码器不会出现削波失真。 真实峰值电平可以在 Meter 视图中找到。 单位: dBTP
峰值电平	测量极短的时间段内 PCM 信号的最大或最小值。 Wwise 在以下位置显示峰值： <ul style="list-style-type: none"> • Wwise Toolbar • Master-Mixer Console • Mixing Desk • Meter view 单位: dBFS
Root mean square (RMS)	均方根。RMS 电平以峰值表值测量，并取极短时间段内的平均值。 RMS 电平可以在 Meter 视图中找到。 单位: dBFS

Loudness Unit (响度单位)

EBU R128 和 ITU-R BS.1770 规定了响度的测量单位。响度不是以 dBFS 度量的。

单位	描述
Loudness Unit (LU)	响度单位。0 LU 等于参考电平（默认为 -23 LUFS）。 请注意：可在 Loudness Meter 设置中更改该默认值 23 LUFS。
Loudness Unit Full Scale (LUFS)	满量程响度单位。0 LUFS/LKFS is equal to 0 dBFs @ 1 kHz according to ITU-R BS.1770-3 and EBU R128
LKFS	K加权满量程响度单位。LKFS 与 LUFS 相同。 ITU-R BS.1770-3 使用术语 LKFS。 EBU R128 使用 LUFS。

响度测量范围

可使用三种不同的时间范围测量响度。在 Integrated (积分) 模式中，可以在较长的时间段内测量游戏的响度级。

术语	描述
Momentary	瞬时。使用长度为 0.4 秒的滑动矩形时间窗计算响度电平。测量不使用门限 (not gated)。
Short-term	短时。使用长度为 3 秒的滑动矩形时间窗计算响度电平。测量不使用门限 (not gated)。
Integrated	使用用户捕获的数据计算响度级。积分响度使用门限（忽略所有 -70 LUFS 以下的值），并根据瞬时响度执行长期测量（基于整个捕获时间积分）。当点击 Capture 按钮时测量开始，再次点击该按钮时停止。点击 Reset 按钮重新开始捕获。

积分响度的详细信息

EBU R128 建议积分响度使用 -23 LUFS 的目标电平。针对游戏选择 -23 LUFS 作为目标响度级有助于在游戏玩家使用的输出设备（电视、移动设备等）上获得与其它媒体更为一致输出电平。为整个游戏选择特定目标响度级还可以确保在游戏的不同层级或不同部分，声音的响度都一致。由于游戏内容是动态的，基于 -23.0 LUFS 的目标电平实现精确的归一化几乎不可能实现，因此 ± 1.0 LU 的偏差是可以接受的。游戏特定段落的响度级通常是不可预测的。当某些段落仅包含背景元素时（例如背景音乐），或某些段落故意降低声音音量时，响度级可能超出公差。

Sony Worldwide Studios Audio Standards Working Group (ASWG) 在 ASWG-R001 中提出以下建议：

- 面向家用平台所开发的游戏内音频内容，平均响度级应归一化至 -23 LUFS 目标电平，公差为 ± 2 LU，考虑到交互式娱乐音频内容的非线性本质，该公差是可以接受的。
- 面向便携式平台所开发的游戏内音频内容，平均响度级应归一化至 -18 LUFS 目标电平，公差为 ± 2 LU。
- 音频内容应作为一个整体进行测量，而不应单独针对语音、音效或音乐。
- 音频内容应根据实际尽可能长时间地进行测量，最短为 30 分钟，所测量的游戏段落应具有代表性，应根据游戏玩法涵盖游戏的所有不同部分；

更多关于响度范围 (LRA) 的信息

根据游戏类型、目标听众和体验环境不同，推荐的最大 LRA 也不同。最大 LRA 值也可能根据发布平台而异，如移动游戏。这里是根据聆听环境采用不同响度范围的示例：

环境	LRA
家庭影院	20
客厅	18
厨房	15
客厅（深夜）	9
公共交通（手机）	6

通则：

- 嘈杂的环境要求 LRA 较低，以便听者可正确听到所有内容。
- 较响的扬声器可以使用较大的 LRA

使用响度范围还有助于决定是否需要进行动态压缩。

EBU 提出，通过响度范围测量，可以选择恰当的方式进一步对工程进行动态压缩，以适应听众或发布平台的动态范围公差窗口。实际上，柔和的整体压缩可能会带来令人满意的结果，并可确保对信号的整体动态范围执行统一压缩：

- 低阈值 (< -40 dBFS)
- 中等压缩比 (1:1.2 – 1:1.5)
- 长释放时间 (> 1 s)

EBU 还提出，根据原始响度级大小，可以同时通过调节压缩器的补偿增益，以达到 -23 LUFS 目标电平。

真实峰值的详细信息

真实峰值测量可检测模拟削波失真。甚至当波峰位于相邻样本之间时，也可预测并避免在后面的数模转换器、采样率转换器，或常用编解码器中出现的失真。

EBU R 128 中推荐的 Maximum Permitted True Peak Level（最大允许真实峰值）为 -1 dBTP。

监视主音频总线中的真实峰值的方法如下：

1. 使用菜单：Views > Meter > Meter - Sync Group 1...4 来打开四个 Meter 视图之一。
2. 点击 [...] 并选择 Master Audio Bus。
3. 从列表中，选择 True Peak 以监视真实峰值。

在 Wwise 中测量响度

监视主音频总线中响度的方法如下：

1. 使用菜单：Views > Loudness Meter > Loudness Meter - Sync Group 1...4 来打开四个 Loudness Meter 视图之一。
2. 点击 [...] 并选择 Master Audio Bus。
3. 点击 Capture，开始计算积分响度。

捕获响度级

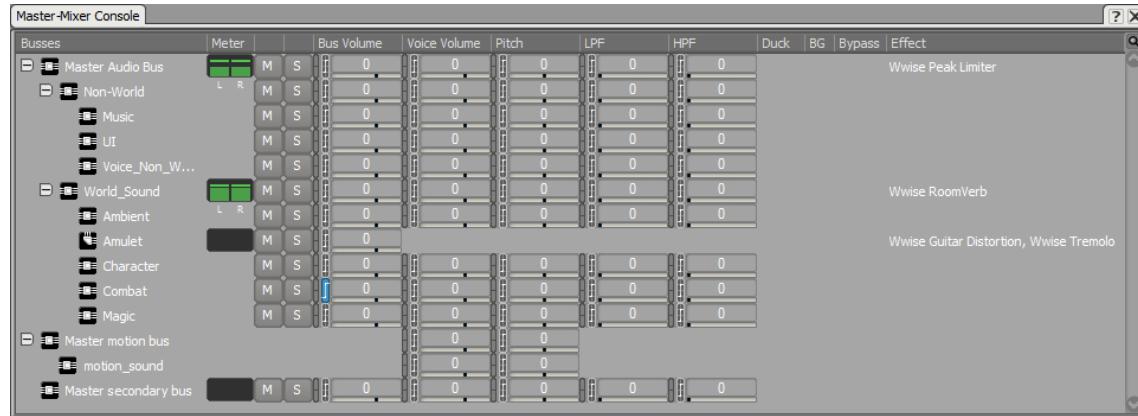
1. 打开 Loudness Meter 视图。
2. 在该视图的标题栏中，从同步组选择图标中选择一个同步组。
3. 点击 [...] 选择一条总线来测量响度。
4. 在 Wwise Toolbar 中，点击 Capture。
注意：请勿在 Loudness Meter View 中点击 Capture。
5. 打开 Performance Monitor（或按下 F6 打开性能分析布局，或按 Shift-N）
6. 在 Performance Monitor 工具栏中，点击视图设置图标。
7. 在列表中，为所选择的同步组启用所需响度值（例如，Momentary Loudness Level - Sync Group 1）。
8. 点击 OK。

了解 Master-Mixer Console 视图

Master-Mixer Console 是一个调音台，它将各种总线属性集中到一个视图中，以便您对游戏音频进行终混。连接至游戏时，您可以一边试听，一边在游戏音频实时播放过程中调节音频和振动的属性。。

您也可以监视与连线至各总线的对象相关的信息，其中包括：

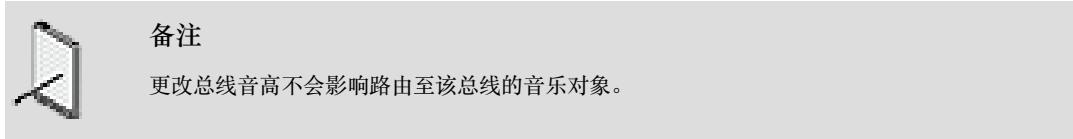
- Meter - 显示所有声道的峰值电平。
- Duck - 表示总线正在闪避。
- BG - 表示总线已绑定背景音乐选项。
- Bypass - 表示为该总线选择了 Bypass All 选项。
- Effect - 表示该总线应用了效果插件，或一系列效果插件。



使用 Master Mixer Console 进行混音微调

工程中的总线连线完成后，您就可以进行终混，方法是连接至游戏，并在游戏过程中实时微调相关属性。可以使用 Master-Mixer Console 轻松修改工程中每条总线的相关属性。属性是累积的，这意味着您定义的这些值会与对象属性值相加。

微调混音时，可以静音或单独播放若干条总线，以便专注于游戏声音或振动的某个特定方面，或者排查特定问题。

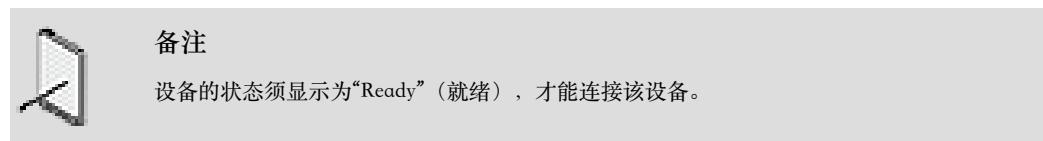


使用 Master-Mixer console 进行终混的方法如下：

1. 点击工具栏中的 **Remote** 按钮，连接至游戏。

Remote Connection 对话框将打开，其中显示当前网络中运行游戏的设备列表。

2. 从该列表中，选择要连接的设备。



3. 点击 **Connect**（连接）。

Wwise 将连接至远程游戏设备。游戏设备的名称显示在工具栏中。

4. 从菜单栏中，点击 **View > Master-Mixer Console**。

此时将显示 Master-Mixer Console。

5. 游戏运行过程中可以试听音频或振动。
6. 可以通过输入数值或拖动相应滑杆来调整以下属性：

Volume - 输出信号的电平或振幅。

Pitch - 输出信号的播放速度。



技巧

要静音或单独播放特定总线，请点击相应的 M 或 S 按钮。

相关主题

- [了解 Master-Mixer Console 视图](#)
- [将效果器作用于对象](#)
- [定义总线的相对属性](#)
- [闪避信号](#)

了解 Mixing Desk 视图

Mixing Desk 是一个灵活强大的调音台，它能把各种各样的属性放到同一个视图中，以供实时细调游戏的混音。用户可以在 Mixing Desk 中添加任意对象或总线，然后定义总线连线、应用效果器和衰减共享集、编辑状态属性，并修改各个独立对象和总线的各种属性。

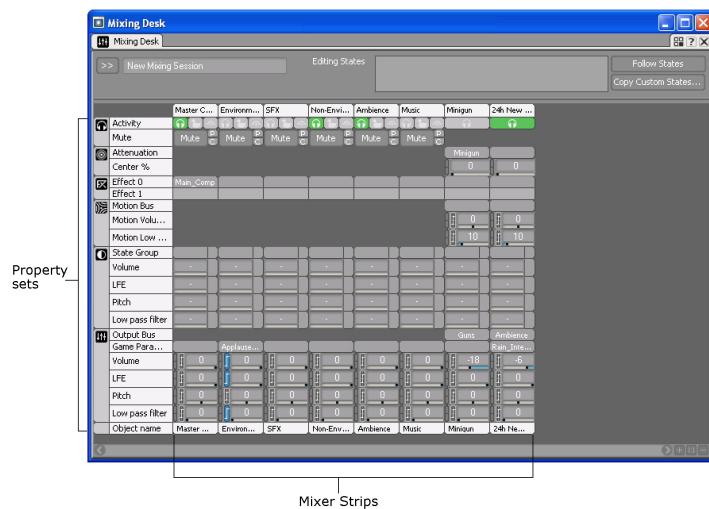


技巧

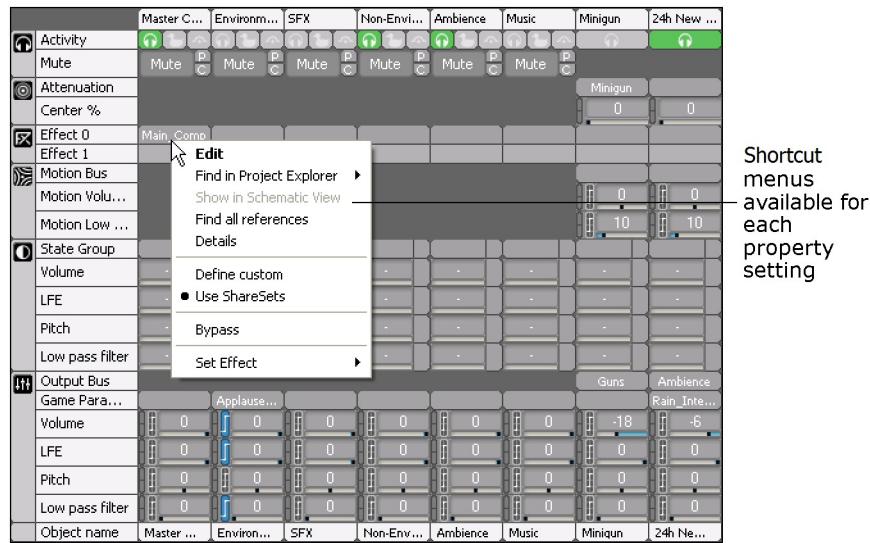
在 Wwise 中，默认情况下您可以按 F8 来切换到 Mixer 布局。

如果您启动捕获会话，则还可在 Mixing Desk 中查看各个对象的活动，包括某声部何时播放、是否闪避总线以及是否旁通了效果器。各个音频对象也可以被静音或单独播放，方便为各个对象进行混音微调。

Mixing Desk 基本上就是一个网格，每列是一个混音器工具栏，每行对应于 Wwise 中的一组常用属性。每个混音器工具栏绑定至一个对象，对象的名称显示在各个工具栏的顶部和底部。



通道条中的各个属性设置都有一个快捷菜单。这些菜单包含与所选属性相关的一组命令。例如，当右键点击 Effect (效果器) 插槽 (0-3) 时，就可以编辑插入效果器的各种属性，可以设置一个新的效果器，或者旁通当前效果器等等。为了访问快捷菜单，只需要在调音台通道条上用右键点击各个属性设置。



Mixing Desk 内的所有元素都会被保存在一个 Mixing Session 中。这用来为游戏的不同元素创建不同的混音会话。这也意味着您可以建立一个混音会话，并在日后其它时间继续对该游戏的音频混音进行精细调整。

建立 Mixing Session

使用 Mixing Desk 对游戏音频的混音进行微调前，您需要创建一个混音会话并添加对象。混音会话与预设或 Soundcaster 会话类似，会存储 Mixing Desk 中的工程内容，这样您就可在日后继续使用该会话。各个混音会话都将保存在 Project Explorer 的 Sessions 选项卡内相应的工作单元中。

为帮助您轻松识别界面中的混音会话，Wwise 使用独特的图标来表示它。

图标	代表
	Mixing Session (混音会话)

创建 Mixing Session

使用 Mixing Desk 前，您需要创建一个混音会话（Mixing Session）。混音会话包含一组总线或对象，并呈现在调音台视图中。混音会话中每条总线和对象都会显示一组滑杆和属性值，以便您微调游戏的混音。

您可以在 Wwise 中的以下位置创建混音会话：

- Project Explorer 的 Sessions 选项卡。
- 在 Mixing Desk 内。

通过 Project Explorer 创建混音会话的方法如下：

1. 在 Project Explorer 中，切换至 Sessions 选项卡。

2. 在 Mixing Sessions 标题下方，执行以下任一操作：

选择工作单元或虚拟文件夹，然后在 Project Explorer 工具栏中点击 Mixing Session 图标。

右键点击工作单元或虚拟文件夹，然后在快捷菜单中点击 New Child > Mixing Session。

新的混音会话将被添加至虚拟文件夹或工作单元。

3. 使用最适合该混音会话的名称来替代默认名称。

4. 双击新混音会话，将其加载至 Mixing Desk。

您现在即可将总线和对象添加至混音会话。

通过 Mixing Desk 创建新的混音会话的方法如下：

1. 在 Mixing Desk 中，点击 Mixing Session Selector 按钮 (>>)，然后从列表中选择 New。

此时 New Mixing Session 对话框将打开。

2. 为新混音会话选择工作单元。

3. 在 Name 栏中，使用最适合该混音会话的名称来替代默认名称。

4. 点击 OK。

混音会话已创建，您可以将总线和对象添加至其中。

相关主题

- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [将对象/总线添加至 Mixing Session](#)
- [在 Mixing Session 内排列混音通道条](#)
- [在 Mixing Session 内调整通道条的宽度](#)
- [从 Mixing Session 中移除通道条](#)
- [使用 Mixing Desk 混音](#)

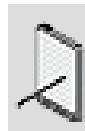
将对象/总线添加至 Mixing Session

创建混音会话（Mixing Session）后，您便可以开始将总线和工程内的各个对象添加至该混音会话。将总线或对象添加至 Mixing Desk 时，将会创建一个混音通道条。通过混音通道条，您可以总览对象或总线的属性设置。它还包含一组控件，让您可以监视各种活动、修改属性、设置效果和状态，以及设置定位属性。



您可以向 Mixing Desk 中添加以下对象：

- 总线
- Actor-Mixer (角色混音器)
- 容器
- 声音
- 音乐容器
- Music Segments (音乐段落)
- Music Tracks (音乐轨)



备注

如果混音会话包含已从工程中卸载的对象或总线相关的混音通道条，则该通道条将为空。

将对象/总线添加至混音会话的方法如下：

1. 执行以下任一操作，将混音会话加载至 Mixing Desk：
 - 点击 Mixing Desk Selector (>>), 然后从列表中选择混音会话。
 - 在 Project Explorer 的 Sessions 选项卡中双击混音会话。
2. 从 Project Explorer 中，将总线或对象拖至 Mixing Desk。

此时在 Mixing Desk 中将创建一个混音通道条，代表该对象或总线。

3. 继续添加对象和总线至 Mixing Desk 中。



备注

Mixing Desk 中包含多个通道条时，一条蓝色插入线会标明通道条的目标位置。

相关主题

- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [创建 Mixing Session](#)
- [在 Mixing Session 内排列混音通道条](#)
- [在 Mixing Session 内调整通道条的宽度](#)
- [从 Mixing Session 中移除通道条](#)
- [使用 Mixing Desk 混音](#)

在 Mixing Session 内排列混音通道条

将总线和对象添加至混音会话（Mixing Session）内后，您可将这些调音台通道条从一个位置拖至另一位置，按任意顺序排列它们。

在混音会话内重新排列通道条顺序的方法如下：

1. 将混音会话加载至 Mixing Desk。
2. 点击混音通道条的标题栏，然后将其拖至 Mixing Desk 中的新位置。

蓝色插入线将显示混音通道条的目标位置。

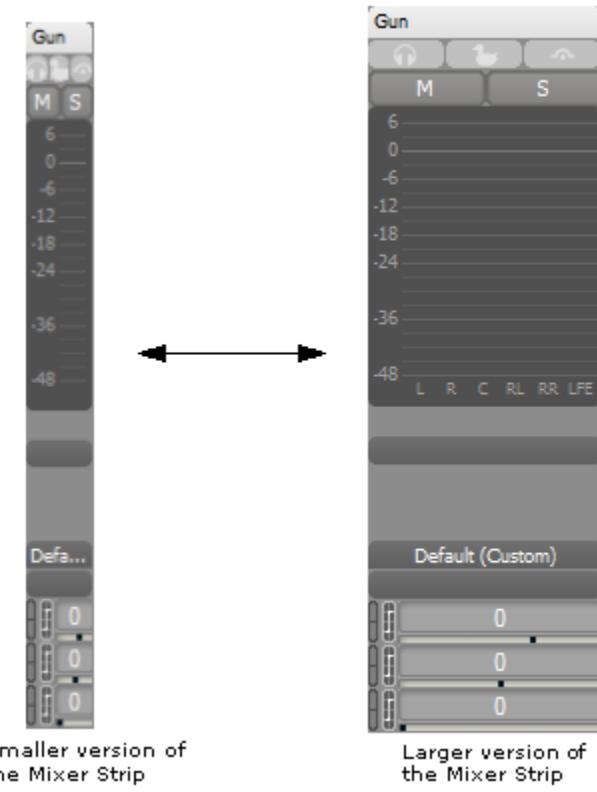
3. 根据需要继续重新排列混音器通道条。

相关主题

- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [创建 Mixing Session](#)
- [将对象/总线添加至 Mixing Session](#)
- [在 Mixing Session 内调整通道条的宽度](#)
- [从 Mixing Session 中移除通道条](#)
- [使用 Mixing Desk 混音](#)

在 Mixing Session 内调整通道条的宽度

如果您发现混音通道条过宽或过窄，则可以调整它们的宽度。



调整混音通道条宽度的方法如下：

1. 将混音会话加载至 Mixing Desk。
2. 点击 Mixing Desk 的网格部分。
3. 执行以下操作之一：

要将混音通道条调宽，请按下 Ctrl 键，同时向上滚动鼠标滚轮，或点击滚动条旁边的 + 符号。

要将混音通道条调窄，请按下 Ctrl 键，同时向下滚动鼠标滚轮，或点击滚动条旁边的 - 符号。

相关主题

- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [创建 Mixing Session](#)
- [将对象/总线添加至 Mixing Session](#)
- [在 Mixing Session 内排列混音通道条](#)
- [从 Mixing Session 中移除通道条](#)
- [使用 Mixing Desk 混音](#)

从 Mixing Session 中移除通道条

在混音会话（Mixing Session）中，您可以随时从 Mixing Desk 中移除若干个调音台通道条。

从混音会话中移除通道条的方法如下：

1. 执行以下操作之一：

- 右键点击通道条的标题栏，然后从菜单中选择 Remove。
- 点击要移除的通道条的标题栏，然后按下 Delete 键。

通道条此时将从调音台中移除。

相关主题

- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [创建 Mixing Session](#)
- [将对象/总线添加至 Mixing Session](#)
- [在 Mixing Session 内排列混音通道条](#)
- [在 Mixing Session 内调整通道条的宽度](#)
- [使用 Mixing Desk 混音](#)

定制 Mixing Desk 中显示的信息

Mixing Desk 可以进行定制，您可以决定将显示哪些类型的信息。使用 Mixing Desk Settings 对话框，可以添加或移除不同类型的信息、选择使用横向的滑杆还是竖直的推子，还可以指定推子的高度。

定制 Mixing Desk 中显示信息的方法如下：

1. 执行以下任一操作，打开 Mixing Desk：

- 从菜单栏中，点击 Views > Mixing Desk。
- 按下 Ctrl+Shift+M。

2. 在 Mixing Desk 标题栏中，点击 View Settings 图标。



此时将打开 Mixing Desk Settings 对话框。

3. 对于要在 Mixing Desk 中显示的每类信息，勾选其旁边的复选框。



4. 如果您选择了显示竖直推子，则请在 Vertical fader height 文本框中指定高度。
5. 点击 OK 应用设置。

相关主题

- [了解 Master-Mixer Console 视图](#)

- 了解 Mixing Desk 视图
- 创建 Mixing Session
- 将对象/总线添加至 Mixing Session
- 在 Mixing Session 内排列混音通道条
- 在 Mixing Session 内调整通道条的宽度
- 从 Mixing Session 中移除通道条
- 使用 Mixing Desk 混音

使用 Mixing Desk 混音

Mixing Desk 是您对游戏中不同音频对象进行混音的完美工具。您可以连接至游戏，启动捕获会话，然后监视音频活动，同时实时微调混音会话内各个对象和总线的各个属性。



使用 Mixing Desk 混音的方法如下：

1. 将混音会话加载至 Mixing Desk。
2. 点击工具栏中的 Remote 按钮，连接至游戏。

Remote Connection 对话框将打开，其中显示当前网络中运行游戏的设备列表。

3. 从该列表中，选择要连接的设备。
4. 点击 Connect（连接）。

Wwise 将连接至远程游戏主机。游戏主机的名称将显示在工具栏中。

5. 点击 Start Capture 按钮（如尚未选择），开启捕获会话。

现在您可以监视游戏中音频的以下活动：

图标	Activity	描述
	声部播放	表示正在播放对象。对于总线，它指示正在播放的声部信号正在通过总线。
	闪避	表示总线是否正在闪避。
	效果器旁通	表示正在旁通插入的效果。

6. 游戏运行过程中可以试听音频。
7. 使用下列工具微调游戏中的混音：
 - **Property sliders/faders** - 修改混音会话中对象和总线的各个属性，包括音量、音高和低通滤波器。
 - **shortcut menus** — 访问与所选属性相关的专用命令。例如，当右键点击 Effect（效果器）插槽（0-3）时，就可以编辑插入效果器的各种属性，可以设置一个新的效果器，或者旁通当前效果器等等。

- Follow states —— 遵循游戏中的状态更改。



技巧

您可以随时双击调音台通道条中对象、总线、状态或游戏参数的名称，来将相应的元素加载至 Property Editor 中。

相关主题

- [了解 Master-Mixer Console 视图](#)
- [了解 Mixing Desk 视图](#)
- [定制 Mixing Desk 中显示的信息](#)
- [创建 Mixing Session](#)
- [将对象/总线添加至 Mixing Session](#)
- [在 Mixing Session 内排列混音通道条](#)
- [在 Mixing Session 内调整通道条的宽度](#)
- [从 Mixing Session 中移除通道条](#)

第 31 章 管理平台和语言版本

概述	537
跨平台制作	537
对工程进行本地化	555
版本技巧和经验总结	559

概述

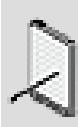
由于游戏制作始终是在优化品质和将内存占用最小化之间追求平衡的一门艺术，因此您需要有用于调整和试验属性及素材的手段，以在各个平台的约束条件下提供最佳的游戏体验。在 Wwise 中，您可以为同一工程中的所有平台分别自定义素材、属性和转码设置。另外，您可以为工程的本地化自定义一些语言属性和转码设置。通过使用模拟和性能分析工具，您可以验证您的作品，确保您成功创建了语言和平台版本。

在为工程中的所有语言和平台完成音频制作后，您可以同时为它们中的每一个生成单独的 SoundBank。有关使用 SoundBank 的详细信息，请参阅[第 35 章 管理 SoundBank](#)。

跨平台制作

对于音频设计师，处理不同的平台通常意味着为各个平台创建不同的工程。在 Wwise 中，只用一个工程就可以全部做到。在创建工程时，您可以自动进行跨平台开发，如果您愿意，则甚至可以同时开发。在默认情况下，工程中的所有可用平台都是活跃的，但是您可以从工程中移除平台，以提高工作流程的效率。有关从工程中移除平台的信息，请参阅[移除平台](#)。

在创建工程并导入素材和音乐后，设置一系列的转码设置 ShareSet（可以针对各个平台进行自定义），另外根据各个平台的优点和限制定义如何使用素材。



备注

在 Wwise 中，跨平台制作的前提是您有权针对这些平台进行开发。如果您后来得到了更多的平台权限，则需要更新安装包，以包含这些额外的平台。

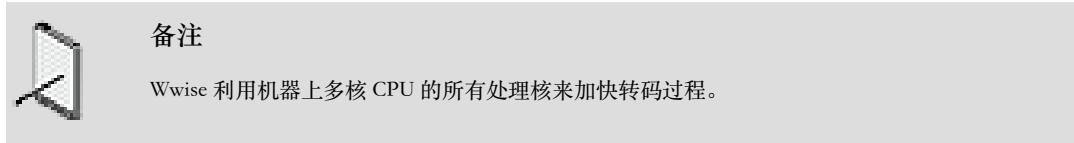
Wwise 可以让您按照以下方式为各个激活的平台自定义工程：

- 对音频文件做转码
- 根据平台自定义对象属性
- 从平台中弃用工程元素
- 切换到不同的平台
- 将设置从一个平台复制到另一个平台

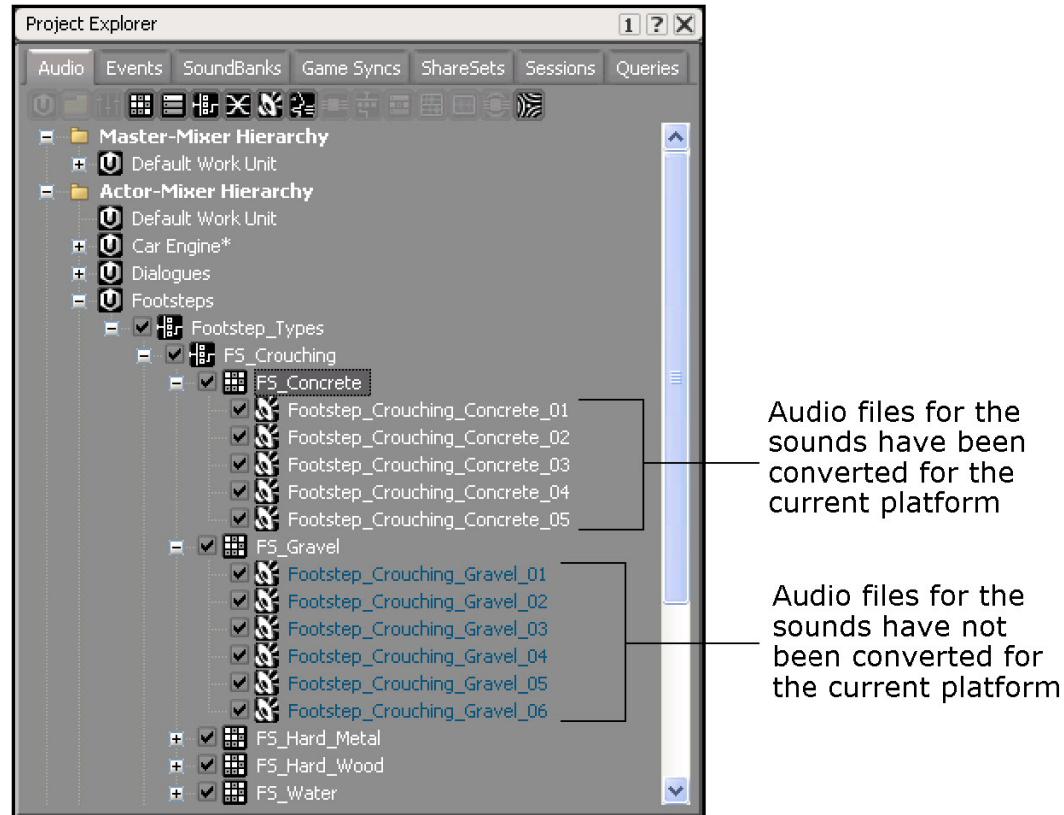
对音频文件做转码

在将媒体文件导入 Wwise 中后，就可以针对不同的平台对媒体文件进行转码了。由于许多媒体文件可能使用相同的转码设置，因此您可以创建一系列转码设置 ShareSet，然后将这些 ShareSet 分配给工程中的各个对象。与工程层级结构中的其它属性一样，子项可从父项继承转码设置 ShareSet。当然，您可以在层级结构中的任何层级选择不沿用这些设置，以便对层级结构中的特定对象应用不同的转码设置 ShareSet。

虽然定义属性和行为的时候您无需对媒体文件做转码，但在生成 SoundBank 之前为各个对象指定转码设置 ShareSet 是个好主意，这样做可以充分利用这些自定义设置。如果您不为对象指定 ShareSet，Wwise 将使用默认的转码设置 ShareSet。有关默认转码设置 ShareSet 的详细信息，请参阅[指定默认转码设置](#)。



与源相关联但未针对当前平台转码的声音对象在 Project Explorer 的 Audio 选项卡中显示为蓝色。

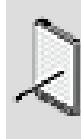


转码过程包括三步：

- 创建音频转码设置 ShareSet
- 将转码设置 ShareSet 指派到对象
- 对音频文件做转码

创建音频转码设置 ShareSet

您应当基于您项目和各个活跃平台的需求来创建用于管理转码设置的 ShareSet。您在此处的许多选择都可能对音频工程的性能和品质产生重大影响。在对工程中的对象应用 ShareSets 后，您可以随时回来调整转码设置 ShareSet，以在平台和游戏的约束下取得最佳的品质。在您导入音频文件时，您还可以复用在此为多语言和源版本定义的 ShareSet 来加速这一过程。



备注

您可以将其中一个 ShareSet 用作工程的默认转码设置。有关为工程设置默认转码设置的详细信息，请参阅[指定默认转码设置](#)。

Conversion Settings Editor 分为两大区域：

- Setting:** Audio Source 上方的区域。在此区域中您可以针对每个平台设置转码设置，包括采样率、音频格式和声道数量。
- Result:** 此区域中列出您的所有音频源。在此区域中您可以比较原始源和转码源，包括声道数量、采样率和文件大小。

Platform	Channels	L-R mix	Sample Rate	Min Sample Rate	Max Sample Rate	Format	Quality	Adv.
Android	Mono drop	<C>	Auto High	None	24000	ADPCM	9	Edit...
iOS	Stereo	<C>	Auto High	None	22050	AAC	9	Edit...
Linux	As Input	<C>	As Input	32000	32000	PCM		
Mac	As Input	<C>	As Input	None	None	Vorbis	4	Edit...
PS4	As Input	<C>	As Input	24000	None	ATRAC9		Edit...
Switch	As Input	L26	As Input	None	None	OpusNX	9	Edit...

Audio Source	Language	Orig. Chan.	Conv. Chan.	Original SR	Converted SR	Original Size	Converted Size	Size Ratio	Duration	Bandwidth
\Interactive Music Hie...	SFX	2.0	1.0	44100	24000	1.0 MB	79.2 KB	13.1:1	6.000	13.2 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	48000	16000	6.0 MB	290.2 KB	21.3:1	33.000	8.8 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	44100	24000	1.0 MB	79.2 KB	13.1:1	6.000	13.2 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	48000	16000	2.2 MB	105.6 KB	21.3:1	12.000	8.8 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	44100	24000	1.0 MB	81.7 KB	13.0:1	6.188	13.2 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	48000	24000	8.6 MB	618.1 KB	14.2:1	46.875	13.2 KB/s
\Interactive Music Hie...	SFX	2.0	1.0	44100	24000	1.0 MB	81.7 KB	13.0:1	6.188	13.2 KB/s

音频转码过程保留了原始文件中的相同音高和时长；不过您可以为您的转码定义以下属性：

- 声道数量（请参阅下文的[关于音频声道](#)。）
- 左右混音
- 采样率（请参阅下文的[关于采样率](#)。）
- 音频格式（请参阅下文的[关于音频格式](#)。）
- 音频格式品质
- 采样率转码品质

您还可以指定您是否要：

- 为对口型或字幕Insert a filename marker（插入文件名标记）；
- Remove DC offset（移除直流偏置）；



注意

对于循环声音，我们建议不要消除直流偏置。消除机制采用高通滤波器，因此不保证将以相同方法修改循环的第一个采样和最后一个采样，因为并不知道这两个样本将连续播放。这可能产生信号中断，导致听到爆音。

- Apply dither (应用抖动)；或者
- Allow channel upmix (允许声道上混)，这意味着当声道被标为 Stereo 或 Stereo drop 时单声道源文件将转换成立体声。



关于音频声道

在转换多声道音频源时，必须决定保留哪些声道。有关详细信息，请参阅[Channel Configuration](#)。

在 Wwise 中，您可以从以下选项中做出选择：

声道选项	描述	备注
As Input	保留与原始媒体文件相同的音频声道数量。	部分平台可能不支持某些音频格式，这时会将多声道文件下混为立体声文件。
Mono	所有声道全部混音为一个声道。	L-R Mix 只用于从立体声到单声道的转换。其它声道配置根据 附录 A, 下混行为 下混。 LFE 声道总是要丢弃。
Mono drop	除第一个声道外，丢弃所有声道。	根据原始文件的声音配置，第一个声道可以是左声道，也可以是中置声道。
Stereo	所有声道混音为左前和右前声道。	L-R Mix 只用于从单声道到立体声的转换。其它声道配置根据 附录 A, 下混行为 下混。 LFE 声道总是要丢弃。
Stereo drop	除定义为左声道或右声道外，丢弃所有其它声道。	如果没有定义左声道或右声道，但定义了中置声道（单声道），则会发生以下转换： 左 = 0.707C 右 = 0.707C 转码结果文件将是原始文件的两倍大。

记住 Wwise 不执行任何多声道编码，这一点很重要；它只将 LPCM 数据馈送到采用立体声、5.1 或 7.1 环绕声配置的游戏机或系统。游戏机或系统一旦收到 LPCM 数据，就可以使用这些游戏机或系统支持的几乎任何格式输出这些数据，包括 Dolby、DTS 或 DPL2 格式。不过的确存在一些限制，包括：

- 在 Android 和 iOS 平台上仅支持立体声。
- Switch™ 平台仅支持立体声和 5.1 环绕声输出。
- Mac 和 tvOS 平台仅支持立体声及 5.1 和 7.1 环绕声输出。
- 其他平台（例如 Windows、PlayStation 4 和 Xbox One）在本机的输出端支持最高 7.1 声道。Wwise 可以支持所有标准声道配置（最高 13.1 声道）和匿名配置（最多 256 个声道）。注意，这些配置需要使用特殊的 sink 插件，这些插件可以解释这些配置或将它们传递到专用硬件。

关于采样率

采样率决定数字音频信号每秒的采样次数。在决定选择什么采样率时，需要考虑许多因素。和其它品质/性能问题一样，设置采样率也是一项保持平衡的艺术。为了给您尽量多的控制，Wwise 为您提供了大量不同的采样率转码选项：

- **As Input** —— 使用与原始音频文件相同的采样率对文件做转码。如果特定平台或音频格式不支持该采样率，则会改用最接近的可用采样率。
- **Auto (Low/Medium/High)** —— 在对文件执行 FFT 分析后，使用 Wwise 选定的采样率对文件做转码。低、中、高品质设置之间的区别在于算法使用的截止阈值不同。您可以通过在 Project Setting (工程设置) 对话框中定义频率阈值，来微调各个品质设置的品质高低。有关 Wwise 执行的自动采样率检测的详细信息，请参阅[定义 Sample Rate Automatic Detection \(采样率自动检测\) 设置](#)。
- **300 to 48,000** -- 300 至 48,000。使用特定采样率对文件做转码。各个平台的采样率范围各不相同，最高 48,000 Hz。

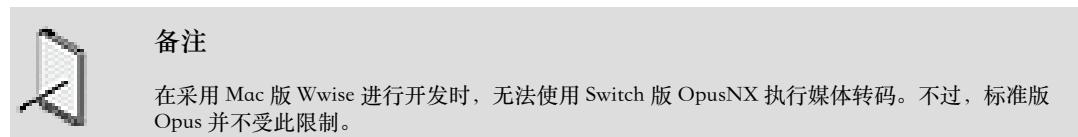


关于音频格式

在对音频文件做转码之前，需要决定它们的目标转码格式。Wwise 支持多种不同的音频格式，您可以更加灵活和可控地应对各个平台的限制。

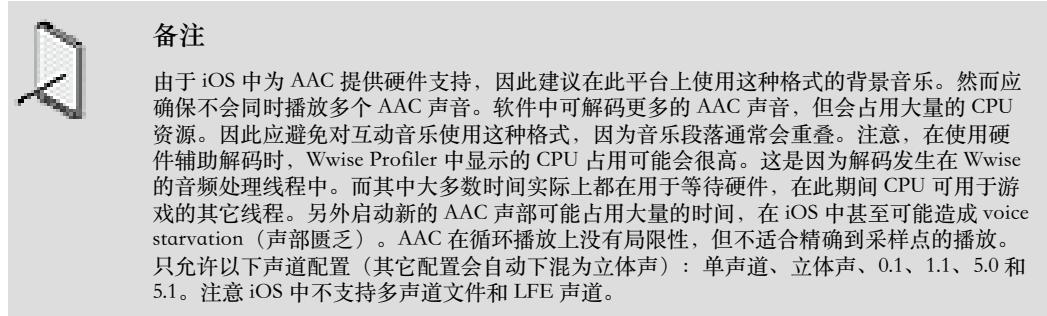
由于各个平台具有不同的规格，因此不可能所有格式都适用于各个平台。下表按平台列出了可用的声音格式。

Platform	AAC	ADPCM	ATRAC9	OPUSNX	OPUS	PCM	Vorbis	XMA
Android		✓			✓	✓	✓	
iOS	✓	✓			✓	✓	✓	
Linux		✓			✓	✓	✓	
Mac	✓	✓			✓	✓	✓	
Switch (切换开关)		✓		✓	✓	✓	✓	
PS4		✓	✓		✓	✓	✓	
Windows		✓			✓	✓	✓	
Windows Phone		✓			✓	✓	✓	
Xbox One		✓			✓	✓	✓	✓

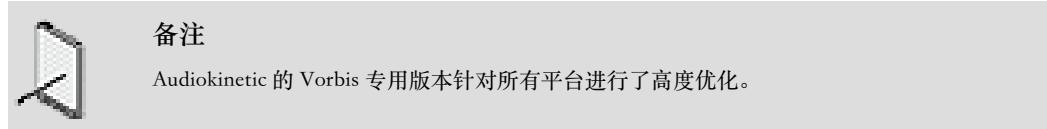


下面简要说明了 Wwise 支持的不同音频格式：

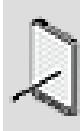
- **AAC** - 针对 Mac 和 iOS 的感知编码压缩方法。据说在比特率相似的情况下，AAC 声音品质优于 MP3。压缩是可变的，取决于内容，品质设置可通过“品质”滑杆进行控制。在 iOS 中，AAC 由硬件辅助的编解码器（如果可用）解码。注意，硬件一次只能解码一个 AAC 声音。



- **ADPCM** —— 一种音频文件编码方法，可对声音信号和由对该声音信号做的预测之间的差异进行量化。ADPCM 量化步长是自适应的，与直接量化信号的 PCM 编码不同。基本上，ADPCM 以音质为代价，显著缩减了容量和 CPU 占用量。因此，它通常用于移动平台。
- **ATRAC9** —— 一种感知编码方法，支持以各种比特率编码音频文件，同时保持非常好的感知声音品质。
- **Opus** - 一个低延迟的音频编解码器，针对语音和通用音频进行了优化，在不影响音质进行压缩方面优于其他编解码器。利用 Bitrate（比特率）设置可以平衡数据压缩效率和感知声音品质。比特率越大，声音品质越好。此版本为 Opus 规格的标准应用，所有平台均提供支持。Opus 解码器需要大量前滚缓冲才能开始解码，所以会占用大量 CPU 和流播放资源。因此，它不适合需要大量寻址或循环的情况，比如复杂的互动音乐、精确到采样点的循环或按照指定触发速率播放的容器。
- **OpusNX** - 一个低延迟的音频编解码器，针对语音和通用音频进行了优化，在不影响音质进行压缩方面优于其他编解码器。利用 Bitrate（比特率）设置可以平衡数据压缩效率和感知声音品质。比特率越大，声音品质越好。此版本针对 Nintendo Switch，并启用了硬件加速。
- **PCM** —— 采用独特二进制表示法即脉冲编码的音频文件编码方法。通过测量两个编码点之间的值进行量化；选择与最近点相关的值。
- **Vorbis** —— 一种感知编码方法，支持以固定和可变比特率编码音频文件，同时保持极好的感知声音品质。通过使用 Quality Factor 设置或指定每声道的最大、最小和平均比特率来控制数据压缩效率和感知声音品质之间的平衡。Vorbis 编码器可能需要使用寻址表。有关详细信息，请参阅[结合 Vorbis 编码器使用 Seek Table](#)。



- **XMA** —— 针对 Xbox One 的硬件支持的感知编码音频压缩方法。XMA 是 Windows Media Audio Pro 的游戏机优化版本。压缩是可变的，取决于内容，可通过 Compression Quality 滑杆来控制品质设置。最新的版本 XMA 2.0 包含新的、用于创建寻址表（seek table）的块大小参数。寻址表可以方便查找 XMA 数据。XMA 音频格式支持单声道文件和立体声文件。Xbox One 同样支持最高 7.1 的声道配置。



XMA 的局限性

XMA 可以处理无缝循环，但无法处理精确到采样点的循环。素材和原始循环点位置会对处理结果产生很大影响。XMA 文件在文件开头和结尾处填有空白。如果这些文件很小，则特别是接连播放时，由此造成的延迟可能会比较明显。

另外，虽然 XMA 的速度是最快的，但在相同比特率下其他编解码器的品质要高很多。

每种格式各有长短，您选择的格式取决于特定游戏的 CPU 和内存限制。有关何时使用哪种音频格式的进一步探讨，请参阅[音频格式](#)。

关于直流偏置

使用直流偏置滤波器是移除直流偏置的好办法，因为直流偏置可影响 Wwise 中的音量并产生副作用。不过，有的情况下不得移除直流偏置，例如精确到采样点的容器。在其它情况下，例如在声音归一化为 0 dB 的情况下，您可能需要消除直流偏置，也可能不需要。在转码过程中，默认移除直流偏置。但是，如果需要，您则可以在 Conversion Setting 对话框中禁用这一设置。



注意

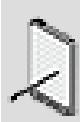
如果您直接从音频源中生成振动，则应注意移除直流偏置可能会更改控制器的振动输出。

创建音频转码设置 ShareSet：

1. 在 Project Explorer 中，切换到 ShareSets 选项卡。
2. 在 Conversion Settings 部分，选择您要创建新 ShareSet 的工作单元。
3. 在 Project Explorer 工具栏中，单击 Conversion Settings 图标

选定的工作单元中创建一个新的转码设置 ShareSet。

4. 为 ShareSet 取一个恰当的名称，然后按 Enter。
5. 双击新建的 ShareSet 以将它加载到 Conversion Settings Editor 中。
6. 通过选择以下其中一个选项为各个平台指定声道：
 - As Input —— 保留与原始媒体文件相同的音频声道数。
 - Mono —— 将所有声道混音到一个单声道中。
 - Mono drop —— 丢弃除第一个声道外的所有声道。
 - Stereo —— 将所有声道混音到左前和右前声道。
 - Stereo drop —— 丢弃除左声道和右声道之外的所有声道。



备注

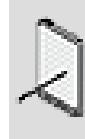
在默认情况下，声道设置跨所有平台链接。要为特定平台设定唯一的声道设置，首先取消链接属性，然后定义设置。



备注

如果您不想增加单声道文件的声道数，则确保禁用 Allow channel upmix 选项。

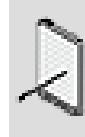
7. 如果您将立体声源转换为单声道或者相反，则可以使用 L-R Mix 设置来指定分配给左声道和右声道的信号功率电平。



备注

在默认情况下，所有平台之间的 L-R Mix 设置是断开链接的。要为特定平台指定通用的 L-R Mix 设置，首先要链接属性，然后定义设置。

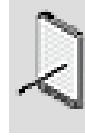
8. 在 Sample Rate 列表中选择转码期间每秒采样音频文件的频率。根据游戏的具体情形，您可以选择以下其中一个选项：
- **As Input** —— 使用与原始文件相同的采样率来对文件转码。如果特定平台或音频格式不支持该采样率，则会改用最接近的可用采样率。
 - **Auto (Low/Medium/High)** —— 在对文件执行 FFT 分析后，使用 Wwise 所选的采样率对文件做转码。低、中、高品质设置之间的区别是截止阈值，该值可识别的频率可用于确定文件转码的最佳采样率。您可以通过在 Project Setting (工程设置) 对话框中定义频率阈值，来微调各个品质设置的品质高低。有关 Wwise 执行的自动采样率检测的详细信息，请参阅[定义Sample Rate Automatic Detection \(采样率自动检测\) 设置](#)。
 - **300 to 48,000** —— 使用特定采样率对文件做转码。各个平台的采样率范围各不相同，最高 48,000 Hz。
9. 如果将 Sample Rate 设为 As Input 或 Auto，则使用 Min Sample Rate 和 Max Sample Rate 条目来限制转码采样率。
10. 请选择以下选项之一来指定转码音频格式：[AAC](#)、[ADPCM](#)、[ATRAC9](#)、[OPUS](#)、[OPUSNX](#)、[PCM](#)、[Vorbis](#) 和 [XMA](#)。



备注

点击 Edit 按钮以修改 AAC、ADPCM、Opus、Vorbis 和 XMA 音频格式的编码参数。有关这些音频格式的编码参数的完整描述，请点击相对应对话框中的 Help 按钮。有关为 Vorbis 音频格式选择参数的最佳做法，请参阅参考文档中的 Vorbis Encoder Parameter 页面。

11. 在 Sample rate conversion quality 列表中选择用于转码文件采样率的方法。您可以选择以下其中一种选项：
- **Normal (Faster)** —— 以高出 Best 选项三到六倍的速度进行高品质转码。
 - **High (Slower)** —— 产生最高品质的转码。



备注

如果您希望内容中包含高频分量并且您正在转码到低于 24 kHz 的采样率，则建议使用 High 选项。

12. 如果您想在各个转码结果文件的开头创建一个标记，则从 Insert Filename Marker 列表中选择 Yes。

此标记只包含文件名，不包含文件的路径和扩展名。在您将动作绑定到声音引擎中播放的声音上时（例如对口型或制作字幕），让名称能看得见会非常有用。

13. 如果您不想在转码过程中移除直流偏置，则取消勾选 Remove DC Offset 复选框。

在默认情况下此选项为勾选状态。在大多数情况下，应该选择移除所有直流偏置。但是在有些情况下，可能不需要消除直流偏置，这些情况包括：

- 将添加到精确到采样点的容器的声音。
- 归一化为 0 dB 的声音。

有关直流偏置如何影响 Wwise 中的音频信号的详细信息，请参阅[移除直流偏置](#)。



注意

如果您直接从音频源中生成振动，则应注意移除直流偏置将改变振动输出。

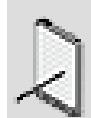
14. 如果您不想在比特率转码期间应用抖动，则应取消选择 **Apply Dither** 复选框。

抖动是在量化之前添加到信号中的噪声，目的是减少量化过程导致的失真和噪声调制。只有在分辨率变化时（例如从 24 位变为 16 位）才应用抖动。

15. 关闭 Conversion Settings Editor。

您指定的设置将自动保存，现在可将 ShareSet 指派到工程层级结构中的若干个对象上。

16. 根据工程中所需的转码设置 ShareSet 数量，重复执行步骤 1-14 若干次。



备注

在填充 Audio Source 表格前，必须将转码设置 ShareSet 指派到对象，然后使用这些设置对音频文件转码。有关将转码设置 ShareSet 指派到对象的详细信息，请参阅[将转码设置 ShareSet 指派到对象](#)。

相关主题

- [将设置从一个平台复制到另一个平台](#)
- [将转码设置 ShareSet 指派到对象](#)
- [对音频文件做转码](#)
- [音频格式](#)
- [根据平台自定义对象属性](#)
- [根据平台选择源](#)
- [从平台中弃用工程元素](#)
- [切换到不同的平台](#)

移除直流偏置

[DC offset](#) 代表信号距离零振幅的百分比。DC 偏置可能会对声音造成不良影响，可选择将其移除。如果您要保留偏置，则应注意在 Wwise 中出现的一些不良影响：

- 具有直流偏置的归一化声音在音频处理期间会达不到最高音量，因为偏置会消耗[裕量空间](#)。此问题可能会蔓延到整体混音，因为对两个声音（一个带有直流偏置，另一个没有）进行混音将造成这两个声音均获得直流偏置。
- 应用了递归 DSP 算法的任何信号可能对带有直流偏置的音频文件非常敏感，这将造成副作用。[reverb](#) 插件和环境效果插件尤其如此。



注意

对于循环声音，我们建议不要消除直流偏置。消除机制采用高通滤波器，因此不保证将以相同方法修改循环的第一个采样和最后一个采样，因为并不知道这两个样本将连续播放。这可能产生信号中断，导致听到爆音。

为避免性能问题，Wwise 在运行时不做直流偏置的移除。因此，建议在将文件导入 Wwise 之前移除直流偏置。可使用直流偏置滤波器（大多数音频设计工具都提供）来做到，也可以在 Wwise 中的转码过程期间执行（位于 Conversion Setting 窗口）。

将转码设置 ShareSet 指派到对象

在为您的工程创建转码设置 ShareSet 后，您可以开始将它们指派给工程层级结构中的对象。转码设置 ShareSet 与其他对象属性一样，子项可从父项继承。这意味着如果您将转码设置 ShareSet 指定给一个角色混音器，其下的所有容器和对象将自动使用该 ShareSet。当然，您可以不沿用父对象使用的 ShareSet，以便对层级结构中的特定对象或源应用不同的转码设置。

在默认情况下，您创建的所有 Conversion Settings 初始都为 ShareSets。但是，您可以针对一个特定对象来调整 ShareSet。如果您不想在对象之间共享转码设置，则可以创建自定义转码设置。在使用自定义转码设置时，这些设置仅应用于当前对象。如果更改转码设置，则只有此对象将受到影响。

将转码设置 ShareSet 指派到对象

1. 将顶层对象加载到 Property Editor 中。



备注

如果对象不是顶层对象，则在设置转码设置选项前必须选择 Override Parent 选项。

2. 切换至 Conversion 选项卡。
3. 在 Conversion Settings 分组框中，点击 Selector 按钮 (>>).

此时将显示工程中当前可用的转码设置 ShareSet 的列表。

4. 在列表中选择您要应用到对象的 ShareSet。
5. 如果您要为对象采用自定义转码设置，则从 Mode 列表中选择 Define custom。

这份 Conversion Settings 的名称后会跟有单词“Custom”（自定义）。从现在开始，您对实例所做的更改将仅影响使用它的一个对象。

6. 点击 Edit 按钮以将 ShareSet 或自定义转码设置加载到 Conversion Settings Editor 中，您可以在其中优化各种转码设置。
7. 在 Audio Source 表中，检查一下将应用这些转码设置的音频源列表以确保列表正确。

要完成此表中的信息，必须对音频源做转码。有关转换音频源的详细信息，请参阅[对音频文件做转码](#)。

相关主题

- [创建音频转码设置 ShareSet](#)

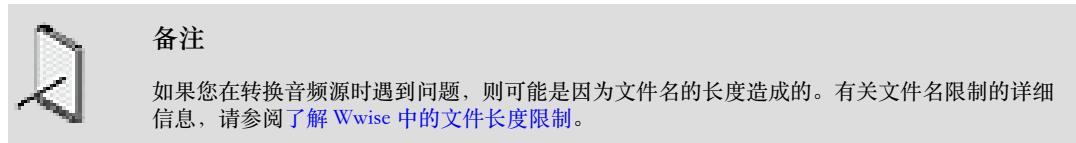
- 对音频文件做转码
- 根据平台自定义对象属性
- 根据平台选择源
- 从平台中弃用工程元素
- 切换到不同的平台

对音频文件做转码

在创建转码设置 ShareSet 并将它们应用到工程中的各种对象后，就可以对音频文件做转码了。您可以决定是对层级结构中选定对象的文件还是对所有未转码的音频文件做转码。您还可以根据以下条件指定转码的范围：

- Platform —— 工程中的当前或所有平台。
- Language —— 当前语言或为工程创建的所有语言。这种转码只针对 sound voice (语音声) 文件。
- Sources —— Contents Editor 中正在使用的声音版本或声音 (源) 的所有版本。

对音频文件做转码是一个非常耗时的过程，特别是当工程中有成千上万的素材时尤其如此。为了帮助加速这一过程，Wwise 会自动使用多核机器中的所有处理核心。



为特定对象对音频文件做转码的方法是：

1. 要打开 Audio File Conversion 对话框，执行以下操作之一：
 - 在 Project Explorer 的 Audio 选项卡或 Contents Editor 中，右键点击您要转换的对象，然后从菜单中选择 Convert。
 - 选择对象或音频源，然后在菜单栏中点击 Edit > Convert。
 - 在 SoundBank Editor 的 Edit 选项卡中，选择并右键点击若干个对象，然后从菜单中选择 Convert。
 - 在 Conversion Settings Editor 中，选择音频源表格中的若干个音频源，然后击 Convert。

此时将打开 Audio File Conversion 对话框。

2. 在 Platform 分组框中选择若干个所列平台。
3. 在 Language 分组框中，选择以下其中一个选项：
 - Current language (当前语言)
 - All languages (所有语言)
4. 在 Source 分组框中，选择以下其中一个选项：
 - In Use version (正在使用的版本)
 - All version (所有版本)
5. 点击 OK。

Conversion - In Progress 对话框于是打开了。第一个进度条显示音频源转码过程的总体进度。若干个次要进度条显示各个文件的转码进度。次要进度条的数量将取决于设备中的处理器内核数。

当文件转码完毕时，对话框将关闭。



备注

如果在转换音频源时 Wwise 遇到任何问题，它则将在对话框底部的转码日志中显示有关它们的信息。您可以随时点击“Stop”来停止转码过程，以进一步调查这些错误消息。Conversion Log（转码日志）条目会同时显示在 Logs（日志）视图的 Conversion（转码）选项卡中。

6. 在 Conversion Settings Editor 中，您可以仔细阅读音频源表格中的信息，以确保各个转码源声道数正确，格式正确，采样率恰当，并且大小合理。



备注

要查看不同平台的转码结果信息，只需从 Wwise 工具栏的 Platform Selector 列表中选择对应平台即可。

对 Audio 选项卡中的所有音频文件做转码的方法是：

1. 从 Project 菜单中，选择 Convert All Audio Files>（清除音频文件缓存）。
- 此时将打开 Audio File Conversion 对话框。
2. 在 Platforms 分组框中，选择若干个已定义的平台。
3. 在 Language 分组框中，选择以下其中一个选项：
 - Current language（当前语言）
 - All languages（所有语言）
4. 在 Source 分组框中，选择以下其中一个选项：
 - In Use version（正在使用的版本）
 - All version（所有版本）
5. 点击 OK。

Conversion - In Progress 对话框于是打开了。第一个进度条显示音频源转码过程的总体进度。若干个次要进度条显示各个文件的转码进度。次要进度条的数量将取决于设备中的处理器内核数。

当文件转码完毕时，对话框将关闭。



备注

如果在转码音频源时 Wwise 遇到任何问题，它则将在对话框底部的转码日志中显示有关它们的信息。您可以随时点击 Stop 来停止转码过程，进一步调查这些错误消息。

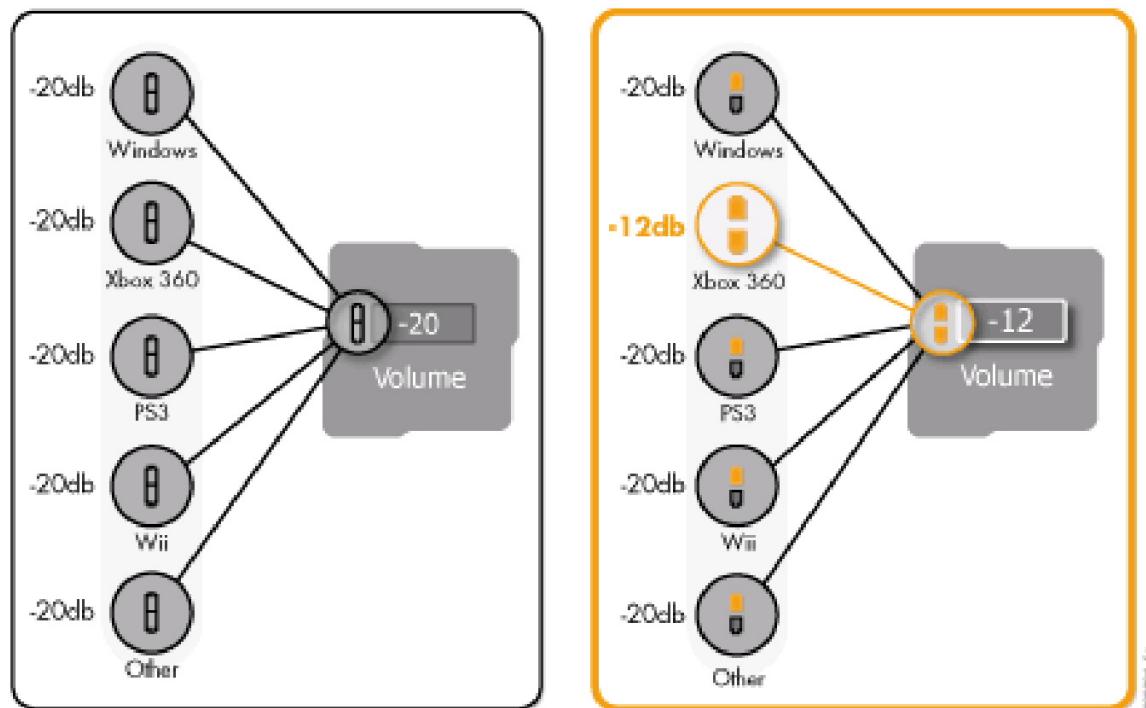
相关主题

- [创建音频转码设置 ShareSet](#)

- 将转码设置 ShareSet 指派到对象
- 根据平台自定义对象属性
- 根据平台选择源
- 从平台中弃用工程元素
- 切换到不同的平台

根据平台自定义对象属性

在 Wwise 中为工程中的特定平台定义声音属性时，默认情况下这些属性设置会覆盖所有活跃的平台。这些属性被称为受到了跨平台链接（linked）。这可以简化跨平台创建工程的过程。但是，如果您想自定义特定平台的属性，则可以对该平台的属性取消链接，然后定义您想要的属性。在取消链接属性时，链接标志会变成橙色。



Properties are linked across all active platforms.

Volume is unlinked and customized for "XBox 360" platform, and the remaining platforms are partially unlinked.

在为平台定义属性时，您可以轻松地判断这些属性在另一平台上是否已经取消链接。链接/取消链接标志将部分变为橙色，表示属性已经部分地取消链接。

对某个平台的某个属性取消链接的方法是：

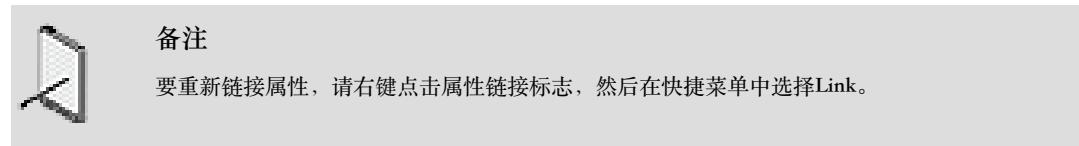
- 右键点击您要取消链接的属性旁边的链接标志。

此时将会显示快捷菜单。

- 点击 Unlink。

标志变成橙色，您定义的值将仅适用于当前平台。

当您切换到该属性仍然链接的平台时，链接标志将显示此属性在另一个平台上未链接。

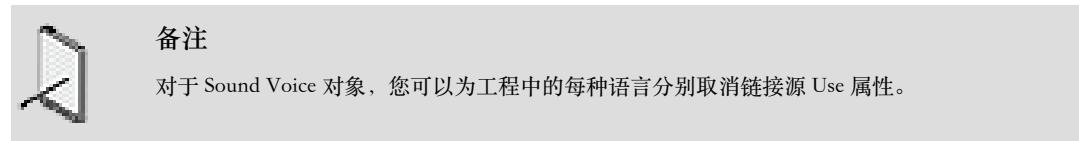
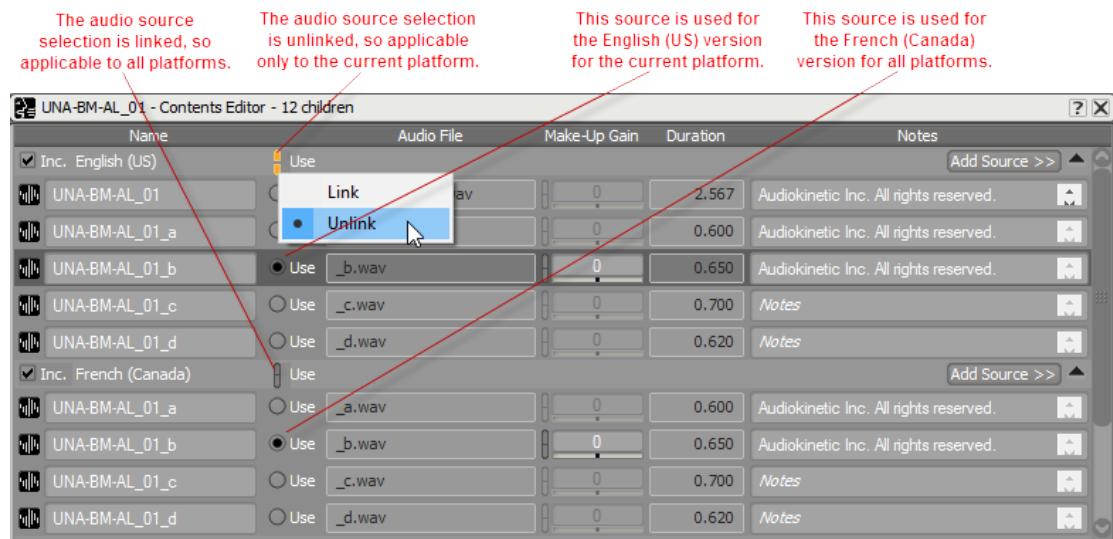


相关主题

- 将设置从一个平台复制到另一个平台
- 创建音频转码设置 ShareSet
- 对音频文件做转码
- 根据平台选择源
- 从平台中弃用工程元素
- 切换到不同的平台

根据平台选择源

在Wwise中，对象可以拥有多个源。在默认情况下，当跨平台播放时，对象将使用同一个源。然而，您可以通过在Contents Editor中取消链接这些源来为各个平台指定不同的源。在为该平台生成SoundBank时，将使用您选择的源。



例如，假设格斗游戏中的角色根据游戏的平台拥有不同的名称。游戏主角最初取名为Max，但在Windows改名为Arthur。特定 Sound Voice 对象因此将包含两种不同的音频源，各个音频源包含的每行对几乎完全相同，但使用的是不同的名称。您可以在

Windows 版本中取消链接声音对象 Use 属性，以便该版本中的声音对象使用 Arthur 的对白而不是 Max 的对白。

在定义用于您平台的源时，您可以使用链接/取消链接标志来检查各个源的状态。当某个源在当前平台上被取消链接时，其链接/取消链接标志将完全变为橙色。然而，当该源在另一个平台上被取消链接时，其标志将部分变为橙色。

为平台选择源的方法是：

1. 在 Contents Editor 中，通过点击相应的 Use 选项选择您要使用的源。

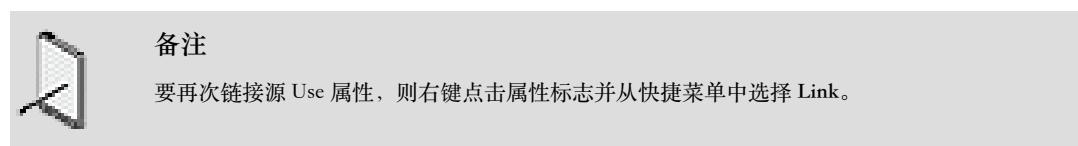
此时选定的源将应用于所有平台。

2. 如果您只想在当前平台上使用此音频源，请右键单击语言栏中的链接标志。

此时将会显示快捷菜单。

3. 点击 Unlink。

标志将变为橙色，您决定使用的音源只适用于当前平台和所选语言。

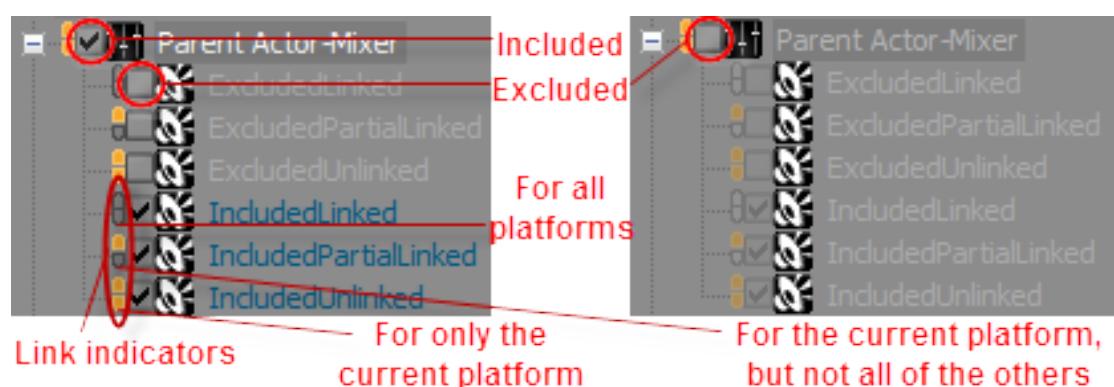


相关主题

- [将设置从一个平台复制到另一个平台](#)
- [创建音频转码设置 ShareSet](#)
- [对音频文件做转码](#)
- [根据平台自定义对象属性](#)
- [从平台中弃用工程元素](#)
- [切换到不同的平台](#)

从平台中弃用工程元素

进行音频研发时，需要牢记游戏和游戏平台的内存需求。您还需要考虑哪些平台能够恰当地处理特定素材。为了优化各个平台，您可能需要启用或弃用特定对象。在弃用某个对象后，为该平台生成的 SoundBank 中不会包含该对象。



Inclusion 复选框可以用来在平台中包含或排除特定对象、文件夹、事件或语言，在 Wwise 的多个视图中都可以看到：

- Project Explorer (对象、效果器、音频设备以及虚拟文件夹和工作单元左侧的无标题复选框)
- Property Editor (对象图标左侧的无标题复选框)
- Contents Editor (内容编辑器中的 Inc. "Language" 复选框，其中“Language”对应您工程中的任何语言)
- Event Editor (Inclusion 复选框)
- List View (Inclusion 复选框)
- Multi Editor (Inclusion 复选框)
- Query Editor (Inclusion 复选框)

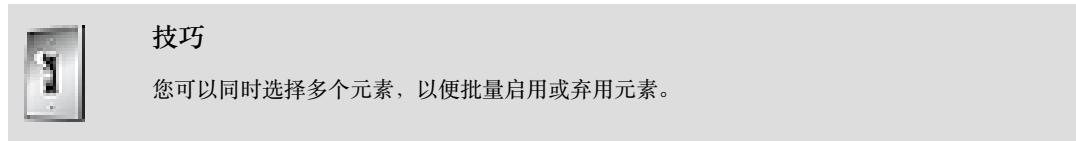
在决定启用或弃用对象后，您可以在 Transport Control 中试听平台专有的音频。还可以使用 Soundcaster 和 Game Profiler 来帮助决定平台中要启用或弃用的对象。有关性能分析和模拟的详细信息，请参阅以下各节：

- [第 34 章 性能分析](#)
- [第 32 章 创建模拟](#)

要在 Wwise 视图中弃用工程元素，请执行以下操作：

1. 在任何含有 Inclusion 复选框的 Wwise 视图中直接清除 Inclusion 复选框，或使用快捷菜单的 Inclusion > Exclude Selection 选项。

该工程元素将不再包含于当前平台中。



从所有平台中弃用工程元素的方法如下：

1. 确保要弃用的工程元素的链接标志为灰色，即该元素已链接。
 - 如果链接标志为橙色，则选择 Inclusion > Link。
 - 如果链接标志只有顶部为橙色，请[切换当前平台](#)来查找未链接的平台（全橙）并将其全部改为链接。



2. 重复要在 Wwise 视图中弃用工程元素，请执行以下操作：中的步骤。

相关主题

- [将设置从一个平台复制到另一个平台](#)

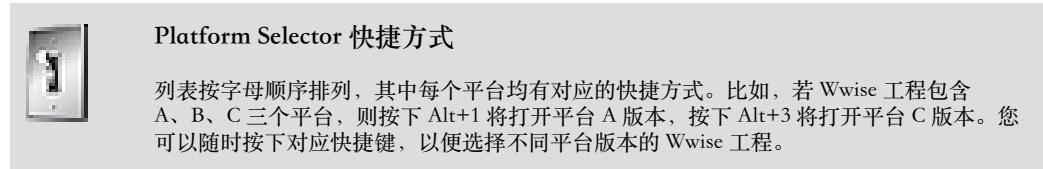
- [创建音频转码设置 ShareSet](#)
- [对音频文件做转码](#)
- [根据平台自定义对象属性](#)
- [根据平台选择源](#)
- [切换到不同的平台](#)

切换到不同的平台

由于 Wwise 工程包含所有活跃的平台版本，因此在开发周期中，您可以随时轻松地从一个平台切换到另一个平台。例如，您可以同时针对多个平台进行开发，而且还可以系统地为各个平台定义特定的对象属性。在您切换平台后，将显示当前的平台版本。

在有效平台之间切换的方法是：

1. 在工具栏左侧，单击箭头以显示 Platform Selector（平台选择器）列表。
2. 从 Platform Selector 列表中选择新的平台。



此时将显示您选择的平台版本。



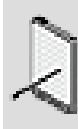
相关主题

- [创建音频转码设置 ShareSet](#)
- [对音频文件做转码](#)
- [根据平台自定义对象属性](#)
- [根据平台选择源](#)
- [从平台中弃用工程元素](#)

将设置从一个平台复制到另一个平台

由于 Wwise 支持在工程针对平台进行编辑，因此有时您可能希望将这些设置从一个平台复制到另一个平台。例如，假设您开发两个平台已经有了一段时间，现在准备添加第三个平台，您可以复制您已经自定义的平台之一。

Copy Platform Settings 对话框（通过 [Platform Manager](#) 访问）可以帮助您做到这一点。您只需指定源和目标平台，Wwise 就会将所有平台相关设置从源平台复制到目标平台。

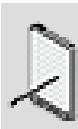


备注

平台设置不同于 **Base Platform**。后者一旦选定用于某个平台之后便不会改变了。如果需要新的 **Base Platform**，则必须创建新平台。

从源平台复制到目标平台的平台相关设置：

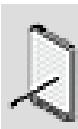
- 取消链接的对象参数



备注

有一些例外情况：Project Settings 对话框 SoundBanks 和 External Sources 选项卡中的平台相关设置不会复制到目标平台。

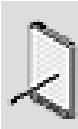
- Inclusion/exclusion（启用/弃用）
- 取消链接的活跃音频源（Contents Editor 中为声音指定的源）
- 转码设置



备注

如果目标平台上不支持源平台上使用的转码格式，则使用目标平台的默认格式代替，并且日志中将显示警告。另外，如果目标平台上的格式不支持源平台上指定的采样率，则它将被修复，并且日志中将显示警告。

- 取消链接的 3D 衰减（在 Property Editor 的 Positioning 选项卡中指定）
- 取消链接的 3D 衰减曲线（在 Attenuation Editor 中定义）
- 取消链接的效果器（在 Property Editor 的 Effects 选项卡中指定）



备注

如果目标平台上不支持源平台上使用的效果器，则目标平台上不会为相应的效果 ID 设置任何效果器。

- 取消链接的对白事件路径目标对象

当设置在源平台上已经取消链接时，该设置在目标平台上也会取消链接，并复制到目标平台。当某设置在目标平台上被链接但在目标平台上已经取消链接时，则目标平台上将重新链接它。

相关主题

- [概述](#)
- [指定工程的 Volume Threshold](#)
- [根据平台自定义对象属性](#)
- [从平台中弃用工程元素](#)
- [根据平台选择源](#)
- [创建音频转码设置 ShareSet](#)
- [将 Attenuation ShareSet 作用于对象](#)
- [定义各种对象属性的衰减曲线](#)

- 使用效果器
- 将对象指派给路径

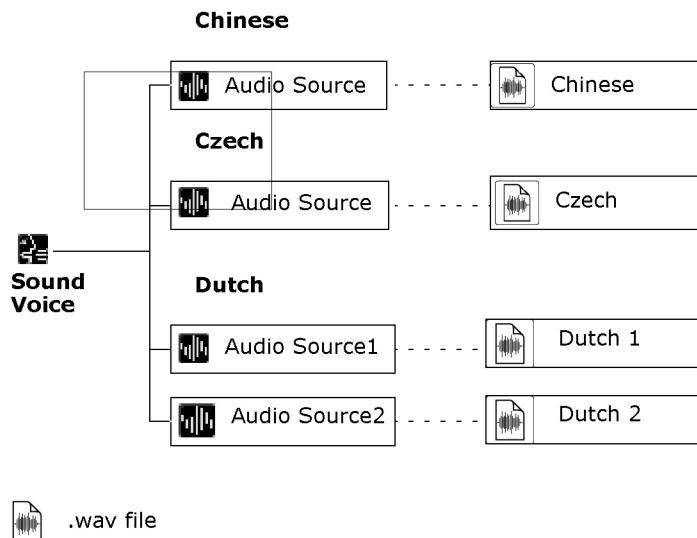
对工程进行本地化

一个良好的本地化工程不仅包括将 Sound Voice (Sound Voice) 对象替换成经过翻译的版本。本地化版本必须能够成功地重建原始语言中创建的同一种音频体验。在技术上，这包括调整和同步不同语言版本的属性，以使本地化工程的品质可与原始版本媲美。根据游戏的发行计划，可以在开发期间同时进行本地化，也可以在游戏完成后再进行本地化。在 Wwise 中，您可以在开发周期中随时进行本地化。与您在一个工程中同时针对多个平台进行制作一样，您也可以在 Wwise 为一个工程中同时创建多个语言版本，并编辑各个语言版本。

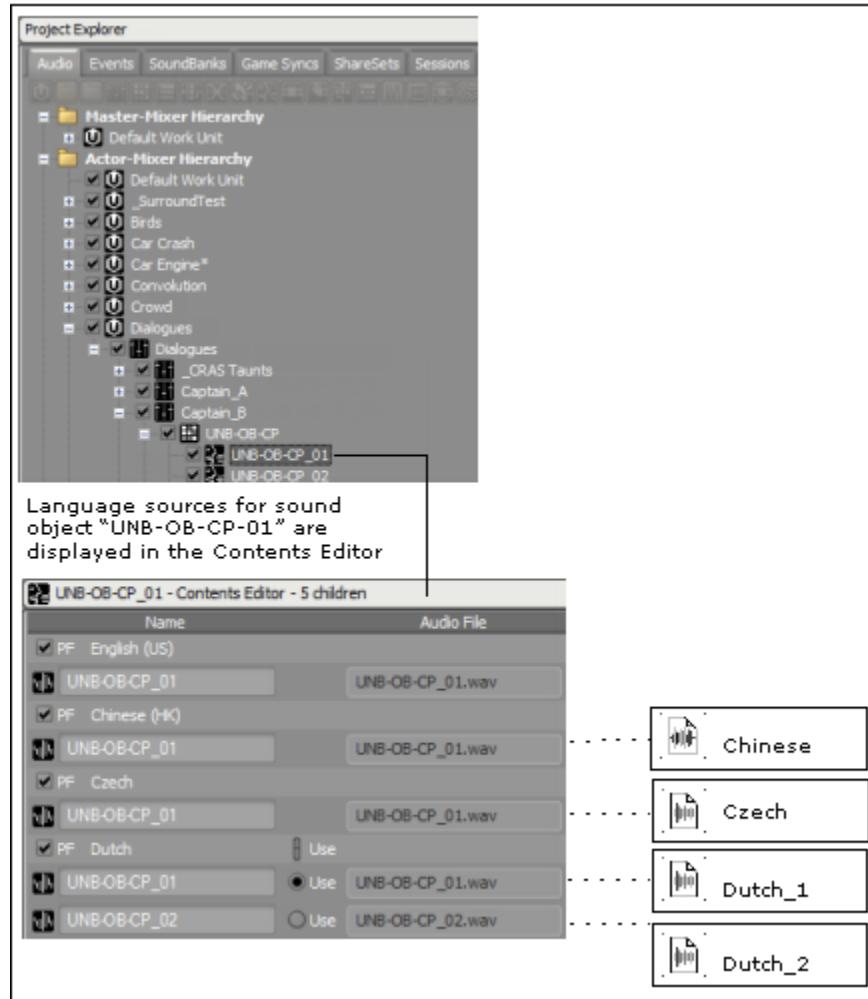
在 Wwise 中，管理工程的本地化包括以下任务：

- 为工程创建语言。
- 定义参考语言。
- 导入语言文件。
- 试听和分析语言版本。

在 Wwise 中，语言版本保存为 Sound Voice 对象的源，并与各个源的不同音频文件相关联。下图展示了声音对象、其语言源和语言文件之间的关系。



Sound Voice 对象包含各个语言的源，但语言源只在 Contents Editor 中显示。

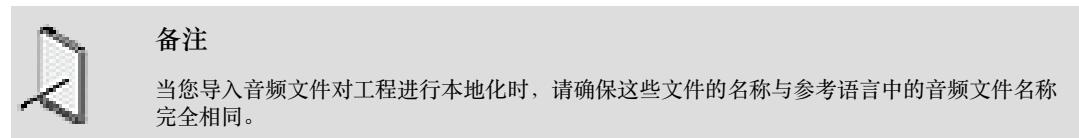
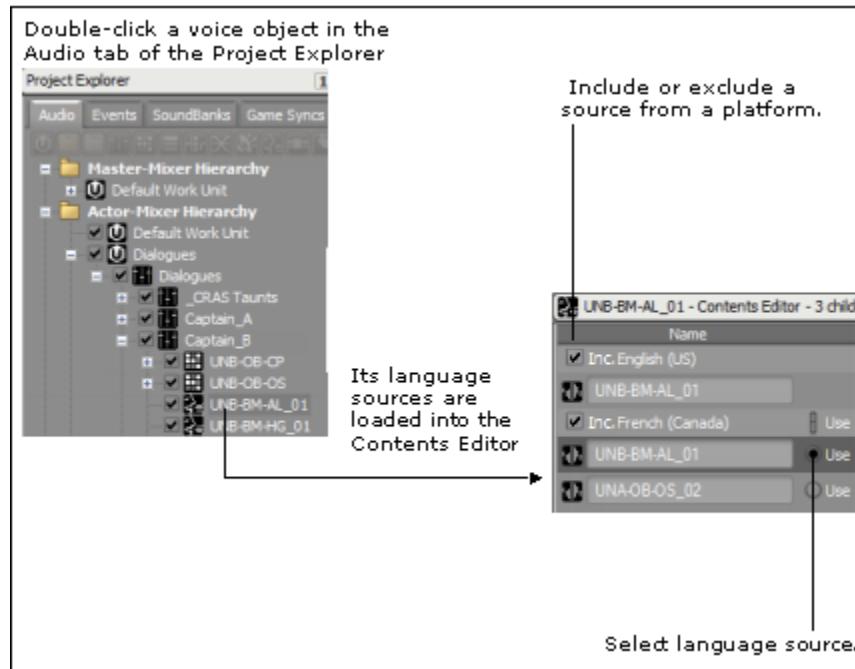


导入语言文件

当语言文件准备就绪时，您可以使用 Audio File Importer 将它们导入工程。导入这些文件时，在 Project Explorer 的 Audio 选项卡内的选定语音对象中会创建语言源，并显示在 Contents Editor 中。如果要立即对工程进行本地化，则可在层级结构的最上层导入经过本地化的文件。

在 Contents Editor 中，您可以根据平台灵活地启用或弃用语言源，以及根据平台选择语言源。有关如何根据语言和源版本进行自定义以及根据平台确定要使用的语言源的详细信息，请参阅：

- [根据平台选择源](#)
- [从平台中弃用工程元素](#).



将语言音频文件导入工程的方法是：

1. 选择您要导入语言文件的 Actor-Mixer Hierarchy 中的某个对象。



2. 从 Wwise 菜单栏中, 点击 Project > Import Audio Files... (点击Shift + I)。

此时将会打开 Audio File Importer。

3. 选择 Localize languages 作为Import Mode (导入模式)。

Import as Sound Voice 选项将自动选择, 并且 Destination language 菜单变为可用。您正在导入的语言必须是工程中已添加的语言。

4. 点击以下选项之一:

- Add Files...

File Open 对话框将打开, 可以在其中选择要导入的媒体文件——AMB、MID 或者 WAV。

- Add Folders...

File Open 对话框将打开, 可以在其中选择要导入的媒体文件所在的文件夹。

- Import Tab Delimited...

File Open 对话框将打开，可以在其中选择制表符分隔的文本文件——TXT 或 TSV——文件定义了要导入的音频文件和要创建的结构。



5. 浏览至导入的文件或文件夹所在的位置。
6. 选择后，点击 **Open**。

所选文件于是加载到 Audio File Importer 中了。

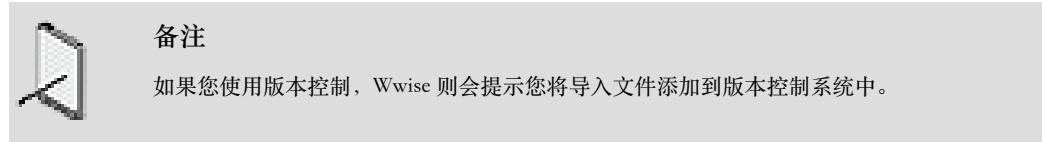
7. 点击 **Import**（导入）

此时将会打开 Importing（导入）对话框，您可以在其中查看导入过程的进度。



导入操作成功完成时，Importing 对话框会关闭，您将返回到 Audio File Importer。

8. 完成导入音频文件后，点击 **Close** 以关闭 Audio File Importer。



切换到不同的语言版本

由于 Wwise 工程中包含您已定义的所有语言，因此在开发周期中您随时都可以轻松地从一种语言切换到另一种语言。Language Selector 列表中填满了您在 [Language Manager](#) 中定义的语言。

切换语言的方法是：

1. 点击工具栏上的箭头以显示 Language Selector 列表。
2. 从 Language Selector 列表中选择新语言。

此时将显示工程的本地化版本。

试听和分析语言版本

在 Wwise 中您可以像平常一样进行试听，创建模拟，并分析各个语言版本。根据分析结果，您可以决定对平台使用特定的语言源版本或源，以获得最佳和效率最高的效果。

有关试听、模拟和分析语言版本的详细信息，请参阅以下各节：

- [第 41 章 认识 Transport Control 视图](#)
- [使用 Performance Monitor 进行监控和故障排查](#)
- [第 32 章 创建模拟](#)

版本技巧和经验总结

您最好仔细阅读以下章节，其中为您提供了一系列技巧和经验总结，在整个音频开发过程中它们可以帮助您更好地管理平台和语言版本。

对多个文件做转码

您可一次性对多个音频文件做转码，方法是在层级结构中右键点击顶层对象（例如容器），然后从快捷菜单中选择 Convert。如果您想为整组对象设置转码设置，那么您可以右键点击顶级对象，然后从快捷菜单中选择 Conversion Settings。您还可以在 Project Explorer 中选择多个对象，然后定义它们的转码设置或者同时对它们做转码。

转码设置的策略

游戏公司在开发游戏时，很多时候会审查各个平台，然后为各性能指标（例如内存占用、CPU 和带宽）分配“预算”。预算的一部分将分配给音频团队，由他们高效地针对各个平台来管理这笔预算。不同平台拿到的预算也不相同，因此音频设计师需要理解如何利用各种转码设置来高效地针对这些差异做出补偿，并仍然能设计出优秀的声音。为了了解优化效果，您可以使用 Game Profiler 来监视游戏性能，并在预算范围内根据需要来调整转码设置。

音频格式

在您的预算中还必须考虑到音频格式。审查 Wwise 支持的不同格式的特点非常有用，可以让您为项目做好合理的选择。一般规则是，文件压缩程度越小，则需要的运算资源越少；但是未经压缩的文件通常会占用更多的内存或磁盘空间。

下表为您提供了一些基本的建议和信息，帮助您决定如何编码音频文件。

文件格式	压缩率 (近似值)	CPU	内存	常见用途示例	局限性
PCM	1:1	极低	高	声音需要高保真。	无。
ADPCM	4:1	低	中等	环境声和音效。	仅在 64 个采样点边界上循环。
Vorbis	3-40:1	中等到高	中等到极低	对话、音乐、环境声和音效。	元数据开销比其它格式稍大，因此应避免将它用于极小的声音（短于几十毫秒）。需要 Seek Table（文件定位表）进行跳转。
XMA	2-20:1	低	中等到极低	不需要精确循环的环境声和音效。	较大的元数据开销。流播放声音只可跨整个文件循环播放，循环点处可能存在空白间隙。内存声音支持在 128 个采样点边界处循环播放。不适用于精确到采样点的播放。
AAC	3-23:1	高（在 iOS 中使用硬件辅助解码器时低）	中等到低	背景（非互动）音乐。	极大的元数据开销。较长的搭建时间。不适用于精确到采样点的播放。

文件格式	压缩率 (近似值)	CPU	内存	常见用途示例	局限性
OpusNX	10-60:1	极低	中等到极低	对白、音乐、环境声和音效。	<p>元数据开销比其它格式稍大，因此应避免将它用于极小的声音（短于几十毫秒）。</p>  <p>*CPU 注意事项 Nintendo Switch 版 Opus 使用 Switch 硬件进行解码。这样会明显优化 CPU 效率。与下述标准版 Opus 相比。</p>
Opus (标准版)	10-60:1	高	中等到极低	对白、简单音乐、环境声和音效。	<p>元数据开销比其它格式稍大，因此应避免将它用于极小的声音（短于几十毫秒）。</p> <p>寻址或循环后的建立时间很长，尤其在从磁盘进行流播放的情况下。因此，在对互动音乐进行流播放时应谨慎使用。此外，寻址还会临时占用大量 CPU 资源。</p>  <p>备注 在采用 Mac 版 Wwise 进行开发时，无法使用 Switch 版 OpusNX 执行媒体转码。不过，标准版 Opus 并不受此限制。</p>

结合 Vorbis 编码器使用 Seek Table

Vorbis 编码器的 **Seek table granularity (sample frames)** (寻址表粒度，单位为采样帧) 默认设为 16384。虽然并非总会用到寻址表，不过在以下情况下必须使用：

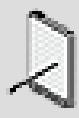
- 结合虚声部“From Elapsed Time”设置使用 Vorbis 编码器时。
- 对互动音乐文件使用 Vorbis 编码器时，特别是当声音引擎必须从除开头之外的位置开始读取文件时。
- 为 Vorbis 编码器使用 Seek Action (寻址动作) 时。

要为一系列 Vorbis 编码文件启用 Seek Table，执行以下操作：

- 在 Conversion Setting 对话框中，点击正在使用 Vorbis 编码器的平台所对应的 Edit。
- 在 Vorbis Encoder Parameter 对话框中，通过从 Seek table granularity 表中选择一个数字来指定粒度。

Seek Table 粒度越小，需要的内存越多。如果粒度太大，精确到采样点的跳转寻址（互动音乐或 Seek 动作）可能导致 CPU 峰值；如果大于磁盘流播放的粒度，则会造

成 I/O 利用率低下。根据经验，所选寻址表大小一般应小于或等于磁盘流播放粒度（即初始化过程中 `AkDeviceSettings::uGranularity` 使用的值）。



文件太小可能会导致寻址失败

若文件太小或与 Seek table granularity 大小相差无几，则寻址可能会失败。在这种情况下，请减小 Seek table granularity。

第 32 章 创建模拟

概述	563
创建模拟	564
管理模拟的播放	567
使用 Game Sync 模拟	569
在模拟中细调属性	572
创建模拟的技巧和经验总结	574

概述

在开发过程中，随时使用 Wwise 对象和事件来进行模拟可能会很有帮助。为此，Wwise 创建了一个名为 Soundcaster 的模拟环境，在此您可以异步播放声音、音乐和振动结构。这意味着用户可以控制播放的内容和时机。这对于测试事件、实时混音等操作来说非常方便。Soundcaster 是一款强大的工具，可以用于：

- 原型设计与试验；
- 开发概念验证；以及
- 同时试听声音、音乐和振动。

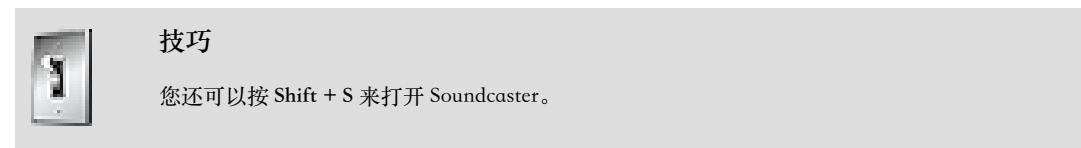
由于您可以在 Wwise 中单独进行模拟，也可以远程连接到游戏进行模拟，因此 Soundcaster 为您的模拟提供了许多不同的用途。对于任何模拟，用户可以选择：

- 有选择性地试听各个平台的音频或振动。
- 试听预转码的音频文件。
- 在播放时对音频或振动进行性能分析。
- 在 Wwise 中，通过手动模拟游戏动作来混音和测试您的音频和振动。
- 在游戏和 Wwise 中分析音频和振动性能。
- 试用与游戏对象相关联的声音、音乐和振动对象。
- 在游戏中对您的声音、音乐和振动进行混音和测试。

在 Soundcaster 中创建模拟之前，您需要新建一个 Soundcaster session，它类似于一种预设，其中包含模拟中使用的 Wwise 对象和事件。每个 Soundcaster 版面都将作为一个会话保存在 Project Explorer 的 Sessions 选项卡中，显示在其相应的 Work Unit 下面。为帮助您轻松地识别界面中的 Soundcaster 会话，Wwise 使用独特的图标来表示它。

图标	代表
	Soundcaster Session

您可以通过在 Project Explorer 中双击会话来访问 Soundcaster，也可以通过在菜单栏中点击 Views > Soundcaster 进行访问。在从菜单栏中打开 Soundcaster 时，将显示最近查看过的 Soundcaster 会话。



Soundcaster 由三个区域构成：

- 主控制
- 游戏同步器
- 对象和事件

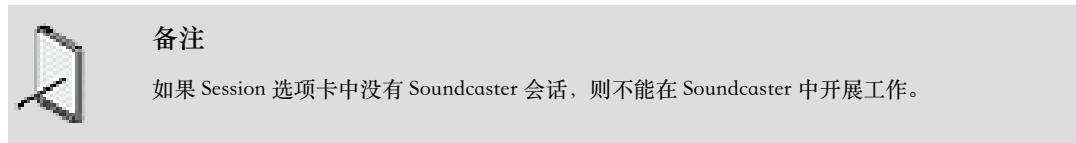


在构建模拟时，您可以使用 Soundcaster 的不同区域来使用它的混音和播放功能。

创建模拟

Soundcaster 可以提供最大的灵活性来满足您创建模拟时的各种需求。您可以添加您想要使用的任何声音、音乐或振动对象，并可根据需要随意更换它们。创建模拟的过程包括以下步骤：

- [创建 Soundcaster Session](#)
- [将对象或事件添加到 Soundcaster](#)



创建 Soundcaster Session

在可以使用 Soundcaster 前，需要创建一个 Soundcaster Session（声音选角器会话）。您可以在 Wwise 的以下两个位置中创建 Soundcaster 会话：

- Project Explorer 的 Session 选项卡。
- Soundcaster 的主控制区。

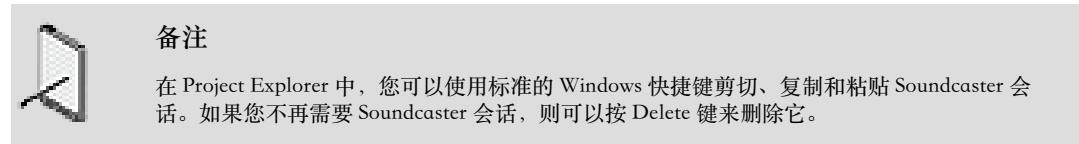
在 Project Explorer 中创建 Soundcaster 会话的方法是：

1. 在 Project Explorer 中，切换至 Sessions 选项卡。
2. 在 Soundcaster Session 标题下，执行以下操作之一：
 - 选择工作单元或虚拟文件夹，然后在 Project Explorer 工具栏中点击 Soundcaster Session 图标。
 - 右键点击工作单元或虚拟文件夹，然后在快捷菜单中点击 <New Child> Soundcaster Session。

新的 Soundcaster 会话将被添加至虚拟文件夹或工作单元。

3. 将默认名称替换成最适合 Soundcaster 会话的名称。

4. 双击新建的 Soundcaster 会话以在 Soundcaster 中打开该会话。



在 Soundcaster 中创建新的 Soundcaster 会话的方法是：

1. 在 Soundcaster 中，点击 Soundcaster Session 选择器开关 (>>) 并从列表中选择 New。

New Soundcaster Session 对话框打开了。

2. 为 SoundCaster 会话选择工作单元。
3. 在 Name 字段，将默认名称替换成最适合 Soundcaster 会话的名称。
4. 点击 OK。

Soundcaster 会话创建完成。

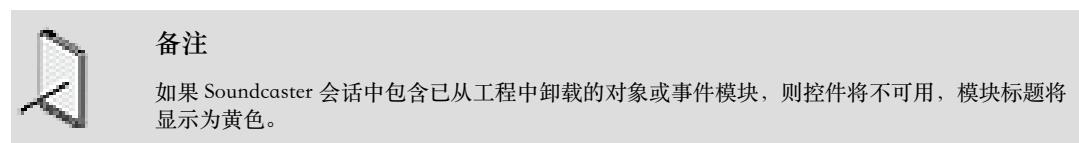
将对象或事件添加到 Soundcaster

当您为 Soundcaster 添加对象或事件时，就会创建一个模块，用来表示对象或事件及其属性和行为。它还拥有一系列的控件，可以让您修改属性，以及播放声音、音乐和振动对象。



在您准备好对您收集的一些声音、音乐或振动进行混音或测试时，便可以开始向 Soundcaster 中添加不同的模块。您可以将以下任何类型作为模块添加到 Soundcaster 中：

- 总线
- Actor-Mixer (角色混音器)
- 容器
- 声音
- 音乐容器
- Music Segment
- Event



向 Soundcaster 中添加对象/事件的方法是：

1. 执行以下操作之一：

- 将对象从 Project Explorer 的 Audio 选项卡拖到 Soundcaster 中。
- 从 Project Explorer 的 Event 选项卡中，将 Event 拖拽到 Soundcaster 内。

蓝色模块插入标志出现在您拖动对象/事件的位置。松开鼠标键时，选定的对象/事件将作为模块添加到 Soundcaster 中。

相关主题

- [创建 Soundcaster Session](#)
- [从 Soundcaster 中移除模块](#)
- [对 Soundcaster 模块进行重新排序](#)
- [播放 Soundcaster 模块](#)

对 Soundcaster 模块进行重新排序

当您为 Soundcaster 添加对象或事件时，就会创建一个模块。虽然可以按任何顺序播放模块，但您可能会想按特定顺序排列它们，以重建特定序列或为测试提供方便。

更改 Soundcaster 模块的顺序的方法是：

1. 将您要移动的模块拖到新位置。

模块移动完成，在 Soundcaster 中，先前位于该位置上的模块下移了。

相关主题

- [创建 Soundcaster Session](#)
- [从 Soundcaster 中移除模块](#)
- [将对象或事件添加到 Soundcaster](#)
- [播放 Soundcaster 模块](#)

从 Soundcaster 中移除模块

如果不再需要模块或者准备创建新的模拟，您则可以将模块从 Soundcaster 中移除。

将模块从 Soundcaster 中移除的方法是：

1. 执行以下操作之一：

- 若要移除一个模块，则点击模块标题栏中的 Close 按钮。
- 若要移除所有模块，则点击主控制区中的 Clear。

于是模块从 Soundcaster 中移除了。

相关主题

- [创建 Soundcaster Session](#)
- [将对象或事件添加到 Soundcaster](#)

- 对 Soundcaster 模块进行重新排序
- 播放 Soundcaster 模块

管理模拟的播放

在将模块添加到 Soundcaster 后，便可以开始使用混音和播放选项了。

在播放模拟时，可以执行以下任何任务：

- 试听平台专有的对象和事件。
- 试听预转码的音频文件。
- 当连接到游戏时，试听与特定游戏对象相关的对象和事件。
- 按您希望的任何顺序播放模块。
- 一次管理一个或所有模块的播放。
- 应用游戏同步器。
- 修改模块的属性。
- 在播放时修改对象的 Event Action。
- 试听互动音乐。

指定要播放的 Wwise 对象

在创建模拟时，您可以根据以下条件选择要播放的声音：

- 平台，以在模拟中将特定 Wwise 对象在不同的平台上启用或者弃用。有关使用声音和平台的详细信息，请参阅[从平台中弃用工程元素](#)。在创建模拟时，您可以选择只播放当前平台中的声音，也可以选择播放模块中的所有声音。
- 转码后的声音——将音频文件的原始版本与转码版本进行比较。在对导入的音频文件做转码时，Wwise 保留了音频文件的“原始版本”，您可以随时试听该版本。这些文件已经完成导入转码过程，但还没有针对平台进行转码。在默认情况下，Soundcaster 播放经过转码的声音；然而您可以选择播放原始导入版本。

播放特定平台的声音和振动对象的方法是：

1. 在工具栏中，验证一下是否已经为模拟选择了正确的平台。
2. 在 Project Explorer 的 Session 选项卡中，双击您要定义的 Soundcaster 会话。

选定的会话于是被加载到 Soundcaster 中。

3. 在主控制区，点击 Inc. Only.

The Inc. 只有按钮变成蓝色，并且对象和事件适用于当前平台时才能试听。



播放未转码声音的方法是：

1. 在主控制区，点击 Original。

按钮变为蓝色，导入的预转码声音将在 Soundcaster 内的所有模块中播放。



相关主题

- 在 Soundcaster 中试听
- 使用 Game Sync 模拟
- 在模拟中细调属性
- 连接到本地/远程游戏系统

在 Soundcaster 中试听

在选择您要播放的声音或 Motion 后，您可自行决定如何试听 Soundcaster 中的模块。通过使用异步播放功能，您可以按照您选择的顺序（连续、重合等）试听模块。更改播放顺序是测试、试验和帮助您找到最佳序列的一个好方法。您可以根据模拟的任务来决定如何播放这些模块。基本的播放任务是：

- 播放模块。
- 暂停模块播放。
- 停止播放模块。

播放 Soundcaster 模块

如果您要试听模块，则可以直接使用您要播放的模块中的控件。在开始按照需要的顺序播放所有需要的模块后，您可以暂停或停止播放选择的任何模块。但是不可播放 Actor-Mixer 或总线。



您可以选择使用主控件来停止或暂停所有模块。



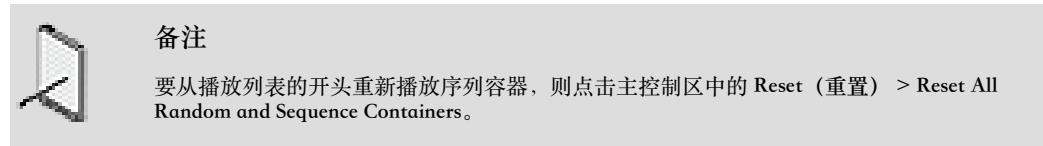
播放单个 Soundcaster 模块的方法：

1. 在模块中，点击 Play。

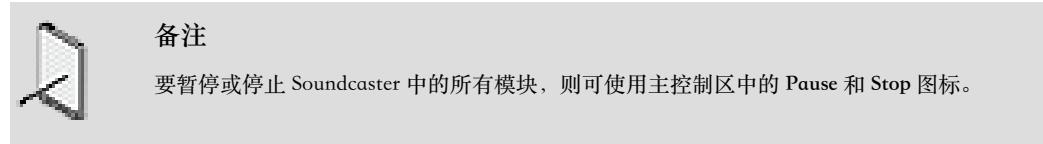
相关声音将播放到结束为止。

- 如果您要暂停播放，则点击 **Pause**。

模块和主控制区中的 **Pause** 按钮变为黄色。要继续播放，则再次点击 **Pause**。



- 要停止播放，则点击 **Stop**。



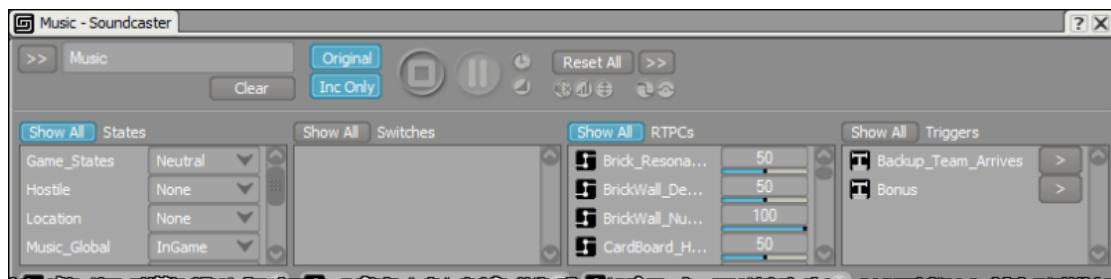
相关主题

- [指定要播放的 Wwise 对象](#)
- [在播放期间启用 State](#)
- [在播放期间指定 Switch](#)
- [模拟 Game Parameter 的变化](#)
- [实时混音和定位](#)
- [使用重置功能](#)

使用 Game Sync 模拟

在 Soundcaster 中，您为工程创建的游戏同步器（Game Sync）应有尽有，都可供试用。在游戏和 Wwise 中，您都可以试听状态、切换开关、RTPC 和触发器施加在声音上的效果。通过连接到游戏并将相应模块添加到 Soundcaster，您可以使用游戏同步器随着游戏动作实时对它们进行试听、测试和混音。

选中 **Show All** 按钮时，创建的所有游戏同步器都将显示在 Soundcaster 的 Game Syncs 区域。否则将只显示与 Soundcaster 中加载的 Wwise 对象相关联的游戏同步器。



在播放期间启用 State

在将模块添加到 Soundcaster 会话中并确定要播放的 Wwise 对象后，您可以在播放期间启用状态（State）。在将对象拖到 Soundcaster 中时，为模块中的 Wwise 对象所启

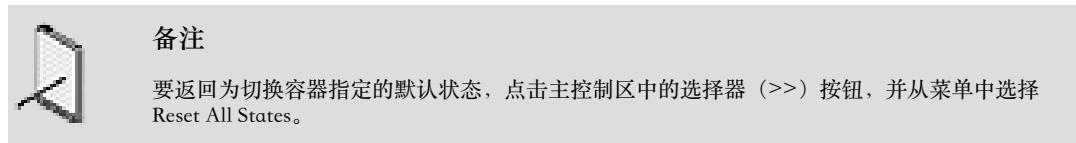
用的状态组和状态会添加到 State 列表。如果未显示状态，点击 Show All。然后，所有状态组和状态都会显示在 States 区域。要了解有关创建状态和指定对象到状态的详细信息，请参阅以下章节：

- [使用 State](#)
- [将 State 指派给对象和总线](#)

在播放期间启用状态的方法是：

1. 在 States 列表中，选择您要应用的 State。

该状态将应用于 Soundcaster 中的所有采用了该状态的 Wwise 对象。



相关主题

- [在播放期间指定 Switch](#)
- [模拟 Game Parameter 的变化](#)
- [在模拟期间调用 Trigger](#)

在播放期间指定 Switch

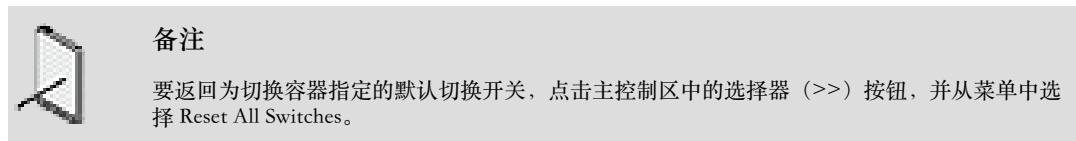
在将模块添加到 Soundcaster 会话中并确定您要试听的 Wwise 对象后，您可以启用在播放期间要应用的切换开关。在将对象拖到 Soundcaster 中时，应用到 Wwise 对象的切换开关组和切换开关会添加到 Switches 区域。如果未显示切换开关，则点击 Show All。于是所有切换开关组和切换开关都将显示在 Switches 区域。要了解有关创建 Switch 以及如何使用 Switch 的详细信息，请参阅以下章节：

- [使用 Switch](#)
- [定义切换容器的类型](#)

将切换开关指定到模块的方法是：

1. 从 Switch 列表中选择要应用的 Switch。

状态于是应用到 Soundcaster 中采用该状态的所有 Wwise 对象上去了。



相关主题

- [在播放期间启用 State](#)
- [模拟 Game Parameter 的变化](#)

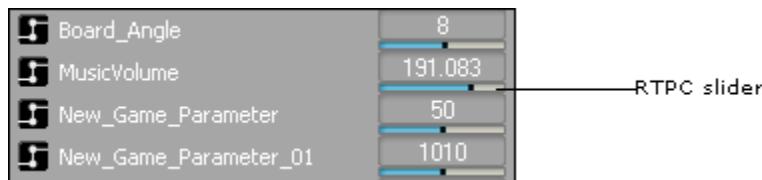
- 在模拟期间调用 Trigger

模拟 Game Parameter 的变化

采用所选切换开关组的切换容器将播放对应于所选切换开关的 Wwise 对象。在将模块添加到 Soundcaster 并确定要试听的 Wwise 对象后，便可以测试映射到游戏参数 (Game Parameter) 的属性值。

- 管理 RTPC 中使用的 Game Parameter
- 将 Wwise 属性指派给游戏参数

要了解有关创建游戏参数和将属性值映射到它们的详细信息，请参阅以下各节：如果没有显示游戏参数，请点击 Show All。您创建的所有游戏参数都将显示在 RTPCs 区域。滑杆代表游戏参数值的值域。在将这些值映射到 Wwise 属性值后，在更改这些参数值时将自动更改属性值。

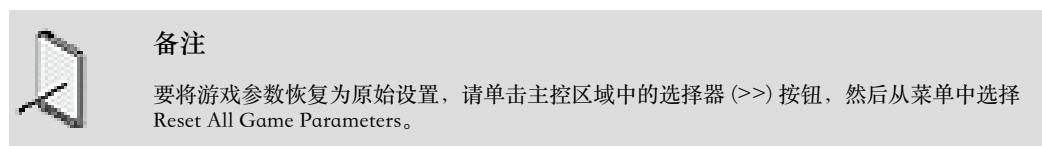


在模拟中，您可以试听这些属性在播放期间的变化。

在播放期间修改游戏参数值的方法是：

1. 在播放期间，使用 RTPC 滑杆更改游戏参数值。

相关的对象属性将根据您在游戏参数和对象属性之间创建的映射而改变。



相关主题

- 在播放期间指定 Switch
- 在播放期间启用 State
- 在模拟期间调用 Trigger

在模拟期间调用 Trigger

将模块添加到 Soundcaster 并确定要试听的 Wwise 对象后，您还可以测试为音乐触发 Stinger (插播乐句) 的 Trigger (触发器)。通过这种方式，您可以模拟游戏在关键时刻通过 Trigger 调用 Stinger 后，插播乐句与当前音乐叠加播放的情况。要了解有关创建触发器以及为它们创建您要在 Soundcaster 中试听的插播乐句的详细信息，请参阅以下各节：

- 使用 Trigger

- 第 29 章 使用 Stinger

在将对象拖到 Soundcaster 中时，相关触发器会添加到 Triggers 区域。如果未显示触发器，则点击 Show All。您创建的所有触发器将显示在 Triggers 区域。

在播放期间调用触发器的方法是：

1. 在模块中的音乐对象播放时，从 Triggers 列表中选择您要试听的触发器。



2. 点击 Call Trigger 图标。

相应的插播乐句将在当前所播放的音乐对象之上播放。您可继续选择其它触发器，播放其相应的插播乐句，以模仿游戏中插播乐句的播放。

相关主题

- [在 Soundcaster 中试听](#)
- [使用 Trigger](#)
- 第 29 章 使用 Stinger

在模拟中细调属性

您可以继续选择触发器，并播放相应的插播乐句来模拟游戏中的音乐。您可以直接在 Soundcaster 中修改各个声音、音乐和振动对象。



属性标志

您可能注意到，模块中的某些属性值旁边有若干个标志。这些标志显示出属性值是否链接到其它平台、属性值是否使用 RTPC 与游戏参数进行相关，以及是否对属性值应用了随机化器。有关在 Wwise 中如何使用这些标志的详细信息，请参阅[属性标志](#)。



有关链接/取消链接属性值、使用 RTPC 和将属性值随机化的详细信息，请参阅以下各节：

- [根据平台自定义对象属性](#)
- [使用 Game Parameter 控制属性值](#)
- [通过随机化属性值来改善播放](#)

除了这些属性标志外，Soundcaster 还包含播放标志，在播放期间，当发生特定行为或事件时，这些标志将变为蓝色。在主控制区和单个模块中可以查看它们。下表列出了 Soundcaster 中的这些额外属性和动作参数标志。

图标	名称	代表
	Delay	延迟。已对事件、随机容器或序列容器应用了延迟。
	Fade	淡变。已对事件、随机容器或序列容器应用了淡变。
	Set Voice Volume (启用旁通)	Set Voice Volume 动作应用于 Event 中的声音、音乐或振动对象。
	Set Pitch (设置音高)	Set Pitch 动作应用于 Event 中的 Sound SFX、音乐或振动对象。
	Mute (静音)	已对事件中的声音、音乐或 Motion 应用了静音动作。
	Set Low-Pass Filter (设置低通滤波器)	设置低通滤波器。已对事件中的声音或音乐对象应用了设定低通滤波动作。
	Enable Bypass (启用旁通)	对事件中的对象应用了启用旁通动作。

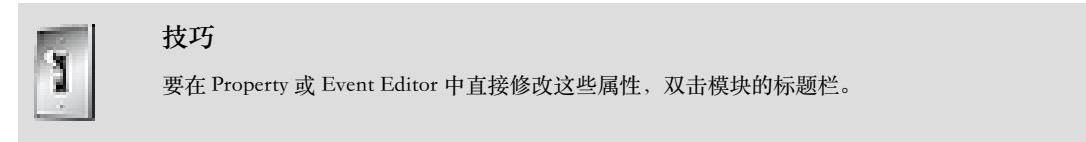
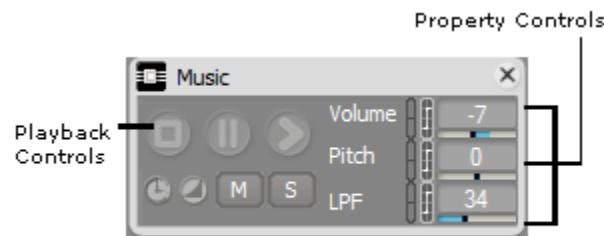
有关将这些属性添加到对象的详细信息，请参阅以下各节：

- [播放容器内的所有对象](#)
- [设置事件 Action 的属性](#)

实时混音和定位

在模拟中，您可以使用属性控件在播放前和播放期间执行以下操作：

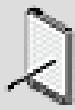
- 实时混音，以修改音量、音高、低通属性值。有关对这些属性进行混音的详细信息，请参阅[定义相对属性（音量、音高、LPF、HPF）](#)。
- 修改定位，以修改 2D 或 3D 声音或振动传播属性。有关定位的详细信息，请参阅[第 11 章 定义定位](#)。



使用重置功能

在 Soundcaster 中试听一个 Event 模块后，您可能想要改变与该事件相关的对象属性。事件动作可暂时更改对象的属性。在更改属性前，应将属性恢复到它们的初始

值。有关使用事件和事件动作的详细信息，请参阅[处理 Event](#)。您可以分别重置各个模块的事件动作，也可以在 Soundcaster 的主控制区中重置所有模块的事件动作。



备注

在主控制区，您还可以清除在 Soundcaster 中的音乐轨，以防它们被强制播放。

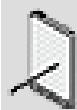
在主控制区中重置事件动作的方法是：

1. 在主控制区，点击 Reset All 旁边的选择器按钮 (>>)。

此时将打开 Reset 菜单。

2. 在 Reset 菜单中，选择以下其中一项：

- Reset All 可将所有对象恢复到原始设置。
- Reset All Random and Sequence Containers 可清空已为对象触发的所有随机和序列动作。
- Reset All Game Parameters 可清空已为对象触发的所有游戏参数。
- Reset All Set Mute 可清空已为对象触发的所有静音动作。
- Reset All Set Voice Pitch 可清空已为对象触发的所有声部音高设置动作。
- Reset All Set Voice Volume 可清空已为对象触发的声部音量设置动作。
- Reset All Set Bus Volume 可清空已为对象触发的总线音量设置动作。
- Reset All Set Voice Low-pass Filter 可清空已为对象触发的所有低通滤波器动作。
- Reset All Bypass Effect 可清空已为对象触发的所有旁通效果动作。
- Reset All States 可清空对象的所有 Set State 动作。
- Reset All Switches 可清空已为对象触发的所有 Set Switch 动作。
- Reset All Music Tracks Force Usage 不再强制播放 Soundcaster 中的特定声轨。
- Reset Position 用于将 Attenuation Preview (衰减预览) 控件内听者的定位重置为其默认定位。



备注

主控制区中的 Reset 功能包括其它命令。有关重置 Random Container 和 Sequence Container 的信息，请参阅[播放 Soundcaster 模块](#)。要重置状态、切换开关和游戏参数，请参阅[使用 Game Sync 模拟](#)。

相关主题

- [使用 Game Sync 模拟](#)
- [管理模拟的播放](#)

创建模拟的技巧和经验总结

在 Wwise 中创建模拟非常灵活，您可以使用它们高效地执行性能分析、创建概念验证以及在 Wwise 或游戏中测试当前的作品。以下是帮助您取得游戏最佳效果而需要考虑的一些概念。

实时混音和对象属性

当您连接到游戏或游戏模拟器时，可以在 Wwise 中实时修改以下相对属性的值：

- Volume (音量)
- Pitch (音高)
- Low-Pass Filter (低通滤波器)

为了能够做到这一点，您需要加载您想在 Transport Control 或 Soundcaster 中修改其属性的对象。如果未加载对象，则更改将不会生效，因为声音引擎中未注册该对象。对于 Transport Control，某些对象是无法加载的，例如角色混音器，但您可以加载角色混音器的子对象，通过这种办法可以将它的父对象注册到声音引擎中。因为可以将角色混音器加载到 Soundcaster，所以这不成问题。在注册对象后，对象将在您连接到游戏时保持注册状态。



备注

记住，如果您将对象固定在 Transport Control 中，则其它对象要到您取消固定第一个对象后才可加载。然而，如果您已将对象加载到 Soundcaster 中，则此对象将在声音引擎中注册。

对于某些属性，包括 Randomizer、Effect、Attenuation 和 Source 插件属性，只有在游戏下次播放该对象时，您所做的更改才会生效。

第 33 章 在 Wwise 中管理内存

概述	577
理解内存管理器的组件	577
设置内存池的大小	578
排查内存问题	579
优化内存池	580
内存管理技巧和经验总结	584

概述

Wwise 可以帮您创建丰富、复杂的声景，为游戏带来无限的可能性。但也要注意各平台的内存资源都是有限的，为游戏设计音频时您需要牢记这些限制。构建工程音频元素时，您可以用各种 Wwise 功能对游戏内存进行有效管理，如 Playback Limit（播放数限制）、Virtual Voice（虚声部）和 Volume Threshold（音量阈值）。

Profiler（性能分析器）中的丰富信息可以帮您排查游戏内存问题。为了充分理解 Profiler 信息、利用 Wwise 的内存节省功能，必须首先理解 Wwise 如何管理工程各组件，包括游戏内的 Media File（媒体文件）、Sound Structure（声音结构）、Event（事件）、Game Sync（游戏同步器）、Effect（效果器）等。

本章概括介绍了内存管理器中的不同组件，并就管理游戏内存的最佳方式给出了一些小技巧和经验总结。要了解内存管理器中不同组件的详情，请参阅 [Wwise SDK 文档](#)。

理解内存管理器的组件

Wwise 声音引擎中，Memory Manager（内存管理器）负责管理游戏音频专用的内存。在游戏起始阶段，音频程序员会初始化内存管理器。初始化时，程序员还须创建一定数量的内存池，来管理游戏内的不同音频组件。

Wwise 使用三种不同类型的内存池：

- **Default memory pool** -- 默认内存池。适用于高层声音结构的通用内存池。该内存池主要包含：
 - 音频结构内容。
 - 已发布事件的命令队列。
 - 针对已注册游戏对象、游戏对象位置、Listener（听者）、RTPC（实时参数控制）、Switch（切换开关）、State（状态）等各项所分配的内存。
- **Lower engine memory pool** -- 底层引擎内存池。主音频处理管线的内存池，通常用于音频播放、音频处理、为效果器分配的内存等。
- **Media memory pool** -- 媒体内存池。通用内存池，包含磁盘内媒体文件的副本。

音频程序员还必须在创建游戏内存池时设置其大小，创建之后将无法再调整大小。在内存管理器的整个生命周期内都可以创建和销毁内存池。内存池被划分为同等大小的块，每块代表最小的分配单元。

您可以用其它技术方案（比如内部研发工具或者 Unreal）来替代内存管理器。关于如何实现定制内存管理器的更多信息，请参阅 Wwise SDK 文档的[重写内存管理器](#)章节。

游戏将如何使用内存池

将 SoundBank（声音包）加载至内存后，声音引擎将执行两个最基本的功能：

- 解析 SoundBank 的 Metadata（元数据），并在 Default memory pool（默认内存池）中存储声音层级和事件信息。
- 将 SoundBank 的所有媒体文件复制到用户定义的内存池。

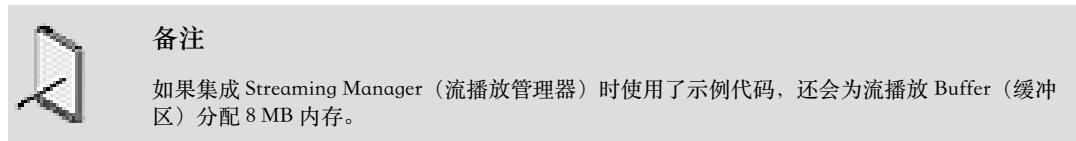
当游戏事件触发音频时，Lower engine memory pool（底层引擎内存池）将执行所需音频处理，然后播放音频。

设置内存池的大小

声音引擎内存池的默认大小非常大，这是为了确保您首次使用时有足够的资源来应对最坏的运行情况。

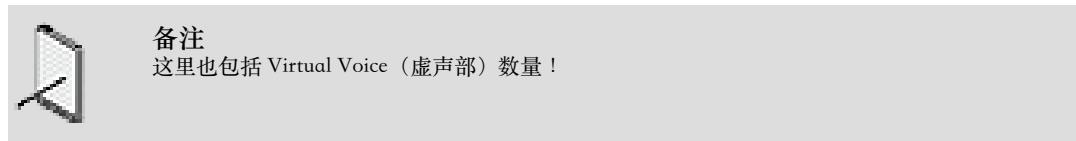
每个池的默认大小是：

- 16MB Default memory pool（默认内存池）
- 16MB Lower engine memory pool（底层引擎内存池）



默认内存池的大小最终应取决于您的游戏类型。例如，如果游戏中有几个 G 的音频素材，但超过 4 或 5 个声音同时播放的情况并不多，而且未使用效果器（或很少），那么可能会要求默认内存池较大，而仅需较小的底层引擎内存池。相比之下，如果游戏中声音较少但需同时播放所有声音，且包含很多 Reverb（混响）、Delay（延迟）和其它环境效果（如第一人称射击游戏），那么可能会要求默认内存池较小，但底层引擎内存池较大。

底层引擎内存池的大小最终应取决于您希望声音引擎进行多少处理。如果游戏要同时播放大量声音和处理很多效果，就需要该内存池提供更多内存。



在 PC 上，不加载音频包也不播放声音时，需要使用约 150kB 内存。您可以在 Wwise 中查看确切内存量，只需切换至 Profiler 布局，并在 Performance Monitor（性能监视器）中查看 Total Used Memory（总内存占用）值。当已加载资源数量、播放声部、游戏对象位置数量增加，该值将显著增长，

所需的内存池大小应取决于：音频品质、加载声音数量、同时播放声部数量、声音结构的复杂性、编码方式、需处理的效果器类型和数量、3D 定位声音数量、切换开关数量等。

内存需求量的唯一可靠获取方式如下：

- 为各内存池分配任意大小的内存。如果可以，一开始请使用默认内存大小。
- 将 Wwise 连接至游戏。
- 使用声音引擎的 Profile（调试）版本，运行游戏中声音最复杂的场景。
- 查看 Advanced Profiler（高级性能分析器）中的 Memory（内存）选项卡。Peak Used（用量峰值）一栏将提示您需要多大的内存池。通常情况下，各内存池的最优大小应超出内存占用峰值的 15-20%。

- 请刨除 Communications（通信）、Monitor（监控）和 Monitor Queue（监控）内存池的大小。其中是针对调试分配的内存，而 Release（发布）版本中不会分配。刨去这些内存池大小，即可估算 Release 版本内存占用。



备注

无需连接至游戏也可以获取声音引擎的内存占用，只需启动捕获会话，然后在 Transport Control（播放控制）或 Soundcaster（声音选角器）中播放若干个事件或对象。播放所用的内存大小将显示在 Advanced Profiler 的 Memory 选项卡中。

选择最差情况进行测试

各内存池出现最差情况的条件可能不同。例如，同时播放 64 个声音将显著增加 Lower engine memory pool（底层引擎内存池）占用，但 Default memory pool（默认内存池）占用仅稍有增加。相比之下，一次加载 20 个 SoundBank（音频包）会增加默认内存占用，但不会影响底层内存池占用。因此，我们建议测试时不要使用同一最差情况，因为游戏中情况纷杂，使用内存的方式也可能各不相同。

关卡切换时，加载和卸载 SoundBank 过程是需要测试的典型情况，因为默认内存池中的内存占用量可能出现峰值。例如，游戏在第 1 关和第 2 关之间切换时，可能会遇到内存占用峰值，因为可能存在内存中加载了两个 SoundBank 的瞬间。



注意

针对 Windows 平台测试最大内存占用时，必须确保运行游戏的 PC 的扬声器配置为 5.1 或以上。请务必注意，因为扬声器设置为 Stereo（立体声）时声音引擎会进行某些优化，所以可能比 5.1 配置下使用的内存少。

开始时应该使用多大的内存？

大多数情况下，开始最好使用默认内存池大小（各个池 16 M）。因为较大的内存池可以让您轻松了解实际内存占用量，之后可以相应地削减内存池。如果要进一步限制指定的内存容量（16 MB 过大），在开始时可使用以下值：

- Default memory pool（默认内存池）：2MB
- Lower engine memory pool（底层引擎内存池）：2MB
- Streaming management（流播放管理）：2MB

流播放 Buffer（缓冲区）大小应取决于采用流播放的声音数量，声音品质，以及能够安全完成流播放的时间要求。8MB 默认值对大部分游戏而言通常都有些浪费，但一开始最好使用较大值，您可以在逐渐明确内存占用量之后再削减内存。

内存池大小和平台

各平台的内存池大小是完全独立的，设置时，请注意内存池大小并不完全取决于平台，而是取决于音频内容和使用方式。这意味着开始时各平台内存池可以采用相同大小，然后再根据 Profiler（性能分析器）中的内存占用读数，定制特定平台的内存池大小。

排查内存问题

设置内存池初始大小后，您可以运行游戏来查看是否会有问题。在这个阶段，Profiler（性能分析器）是检测问题的实用工具。

使用 Wwise Profiler 时，游戏每次发生内存分配故障，都会在 Capture Log（捕获日志）里显示警告通知。您可以查看通知列表，随时监控游戏中哪个内存池出现了内存不足。

关于使用 Profiler 的更多相关信息，请参阅 [第 34 章 性能分析](#)。

运行时内存不足

对于不同情况下出现的“内存不足”，声音引擎也会做出不同的反应。下表说明了几种情况下出现内存不足的后果：

情况	内存不足的后果
初始化声音引擎	初始化失败。
加载 SoundBank	SoundBank（音频包）加载失败。
开始属性过渡，如音量	Transition（过渡）被跳过，参数将忽略过渡，直接跳至目标值。
播放声音	视声音引擎初始化时是否启用 Memory Threshold（内存阈值）而定。默认情况下 Memory Threshold 禁用，此时播放将失败。如果启用了 Memory Threshold，那么将应用优先原则。强烈建议启用 Memory Threshold 功能。

优化内存池

为了优化您的内存池，首先要了解可以在哪里节省内存。以下情况可能导致内存占用较高：

- 加载 SoundBank（音频包）会增加 Default memory pool（默认内存池）的内存占用。请注意，各音频包占用的内存量不同。默认内存池占用量不取决于音频包的物理大小，而取决于其中的声音和事件数量。
- 某些效果（包括混响 Reverb 和延迟 Delay）在播放时会占用底层引擎内存池中的内存。
- 同时播放多个声音会显著增加底层引擎内存池占用。
- 短时间内触发多个动作会增加默认内存池的内存占用。
- 注册游戏对象、设置对象参数和对象位置都会使用默认内存池中的少量内存。请注意，游戏对象不再使用时必须 Unregister（取消注册）从而释放其内存，否则内存占用将会不断增加。

优化内存时，您需要考虑以下内存池：

- [底层引擎内存池](#)
- [默认内存池](#)
- [媒体内存池（用于 SoundBank）](#)

底层引擎内存池

Lower Engine Memory Pool（底层引擎内存池）中的内存用于播放声音，它直接受同时播放的声音数量影响，还会受同时使用的效果数量和类型影响。因此，为了削减该内存池用量，需要先弄清楚同时希望听到多少声音。有些游戏很少同时听到超过 10 种声音，而有些游戏则会同时听到 100 种。设置内存池大小时，请使用游戏中的最差情况。

这个比例大致呈线性，但实际还取决于编解码器类型、效果器数量和其它因素。

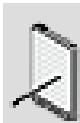
还需要明确的一点是，同时播放 170 种声音是否对游戏有益。当声音达到一定数量后，人耳将无法辨别不同声音，它们将沦为噪声。您需要弄清楚最适合游戏的声音数量，当然这需要实验。您可以对游戏中的多个场景进行 Profile（性能分析），并使用 Advanced Profiler（高级性能分析器）的 Memory（内存）选项卡来观察资源占用情况。

为降低 Lower engine memory pool（底层引擎内存池）占用，需要减少同时发声的声部数量。通过以下策略可以实现：

- **Playback Limits**（播放数量限制），位于 Advanced Settings（高级设置）中。比如说，真的需要听到 50 个子弹纷飞的声音吗？很可能不用，您可以将其限制在合理数量，比如 15 个。注意，也可以对总线设置该限制。
- **Playback Priority**（播放优先级），位于 Advanced Settings（高级设置）中。比如说，子弹声没有对话声重要，那么声音过多时，可以首先去掉子弹声。优先级可以跟播放数限制结合使用。
- **Distance-based Priority Offset**（基于距离的优先级偏置），位于 Advanced Settings（高级设置）中。远距离对象通常不如较近的对象重要，比如说，如果近处有 15 个子弹声，就不需要听到 10 米开外的子弹声了。
- **Memory Threshold**（内存阈值），在 SDK 中设置。初始化声音引擎内存池时，您可以指定其总大小的某个百分比作为内存阈值，占用达到这个百分比时，会根据优先级顺序终止声部。这样内存就有了一个强制界限，几乎所有的内存不足情况都可以通过这种可控方式来避免。请注意，由于存在内存碎片，因此内存不可能全部占用，所以开始设置的阈值最好是内存池大小的 0.9 或 90%。
- **Volume Threshold**（音量阈值），位于 Project Settings（工程设置）中。通过音量阈值可以终止微弱而无法听到的声音，它可以与 Below Threshold Behavior（低于音量阈值行为）一起使用，和 Attenuation（距离衰减）也紧密关联，因为通常声音都是越远越微弱。音量阈值默认值为 -80。
- **Below Volume Threshold Behavior**（低于音量阈值行为），位于 Advanced Settings（高级设置）中。最节省资源（CPU 和内存）的选项是“Kill Voice（终止声部）”，这对很短的声音或循环环境声很实用，会终止声部，但不会终止循环。第二个更好的选项是“Send To Virtual（发送至虚声部）”+“Play from beginning（从头播放）”，然后是“Send To Virtual”+“Resume（恢复）”，然后是“Send To Virtual”+“Play from elapsed time（从剩余时间开始播放）”。“Continue to play（继续播放）”和“Play from elapsed time”是最耗费资源的选项。请注意，“Continue To Play”是 Wwise 的默认设置。

默认内存池

Default memory pool（默认内存池）包含内存中所加载声音和事件的结构元数据，也包含 Wwise 中各行为所需工程对象的所有属性。它还包含所有注册的游戏对象及其相关信息，包括 Game Sync（游戏同步器）值、位置、方向等。当内存中加载的库越多，元数据也越多，这就需要更大的默认内存池。这个大小最终取决于同时播放声音数量最多的情况，可以是在一个场景、关卡、地图、游戏区域等。



备注

默认内存池中不包含媒体文件。

每个声音结构和事件组合平均占用默认内存池中约 300 字节的内存。例如，如果您的工程包含 50,000 个以上的声 (包括对话)，那么仅结构数据就要占用 15MB 内存。可以想见，一次将其全部加载至内存不太可行，因此需要跟处理媒体文件相同的方法来加载和卸载结构数据。

在 Wwise 工程中可以看到，内存中声音 Structure (结构) 的绝对数量较大，所以将占用默认内存池的很大部分 (约占25-50%)。当然，加载的声音越少，默认内存池也就越小。第二大占用项为 Event (事件) 的元数据，它通常占用默认内存池的 10%。事件非常小，即使有多个事件也不会占用很大的空间。Random/Sequence Container (随机/序列容器) 结构可能是最占用内存的，占用量约是简单声音的三倍。但通常 Random/Sequence Container 的数量要比声音或事件少得多，因此整体而言占用量要更少。

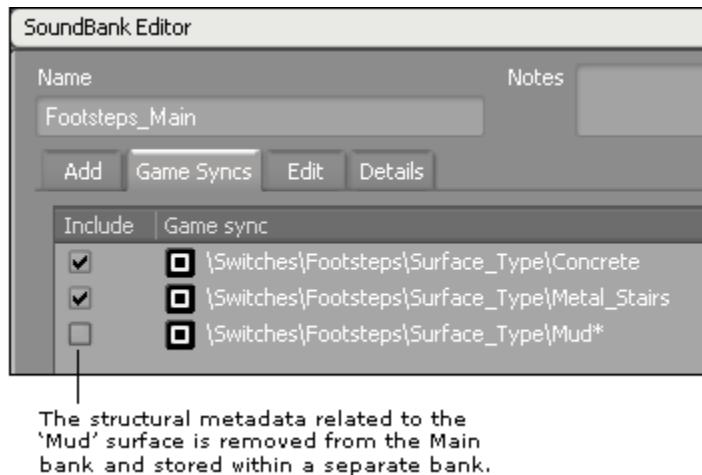
一般情况下，开放世界游戏需要播放更多的声音，因此要求更多内存，默认内存池需要 5-8 MB。其它类型游戏的内存占用会低得多，如 2-3 MB。请注意，这些数值仅供大致了解使用 Wwise 的其它游戏所用的内存量，不保证也适用于您的游戏设计。

以下经验总结可以帮您减少默认内存池占用：

- 对于包含很多声音结构和事件的大型 SoundBank (音频包)，可以将其拆分成较小的音频包，并根据需要动态地加载和卸载音频包。请确保您的音频包不仅仅根据角色组织，而要更多地按情景分类。
- 可以使用 ExecuteActionOnEvent API 来减少事件数量。与创建一对 Play/Stop Event (播放/停止事件) 相比，只创建播放事件，然后调用 ExecuteActionOnEvent 来停止可以达到同样效果，Pause/Resume (暂停/恢复) 也同样适用。
- 严格管理您的游戏对象。游戏对象完成任务后，请立即 Unregister (取消注册) 它们。要避免让冗余游戏对象处于激活状态；这样毫无益处，只会消耗内存。例如，如果一名 NPC 死亡，应该取消注册其游戏对象，不要将其重新用于其它对象，需要时请注册一个全新的游戏对象。一般情况下，如果有上千个游戏对象处于激活状态，可能就太多了。
- 使用 Virtual Folder (虚拟文件夹) 来组织声音，而不要用 Actor-Mixer (角色混音器)。Virtual Folder 不占用内存，而 Actor-Mixer 会占用。仅当一组对象确实要共享属性值 (默认值除外) 时，才有必要使用 Actor-Mixer。这种情况下 Actor-Mixer 可以节省内存，因为该属性仅在一处生效。当然，这还取决于 Actor-Mixer 是否被事件引用，如 Set Voice Volume (设置音量值) 或 Set Voice Pitch (设置音高值)。
- 请尝试减少大型层级结构的大小和复杂程度。大型层级结构的常见例子是 Impact (撞击声) 层级结构或 Footstep (脚步声) 层级结构，变量很多，规模也可能很大。遗憾的是，它无法拆分 (媒体可以拆分，但结构数据不能)。以下方法可以简化大型层级结构：
 - 如果仅需改变简单属性 (为相同的样本设置不同的音量/音高/随机化等)，则使用 RTPC，而不要复制声音结构。
 - 可以用 Dialogue Event (对白事件) 和等价的 State Group (状态组) 来替换某些 Switch (切换开关) 层级结构，或其中的一部分。如果变量频繁更改，这将很实用。例如，撞击声层级结构中，撞到哪个表面无法预先确定，只有到事件发生时才能知道。可以使用 Dialogue Event 和相同名称的 State Group 来替代 Switch Container (切换容器)。因为切换容器基于切换开关值，而 Dialogue Event 的优势之一是能轻松复用样本或子结构。而且，Dialogue Event 不会为每个游戏对象分配内存。还要记住，您可以将 Switch Container 指派给 Path (路径)，从而进一步增加变量维度。

- 以脚步声为例，Dialogue Event 可以有两个状态组，如 Step Type（脚步类型，如走路/跑动）和 Surface Type（地表类型），各条路径均可指定给“Footwear Types”切换容器（鞋子类型，开关值包括“Boots（靴子）”、“Civilian shoes（平底鞋）”“Barefoot（赤足）”等）。这是因为“Footwear”在游戏对象存在周期内不会频繁改变，但其它两个变量会经常变化。
- 该方法的另一个优势是，Dialogue Event 可以独立存在于仅有事件的音频包内，而无需关联要播放的声音结构。就是说，您可以将事件中的复杂声音结构进行拆分，置于多个音频包内。以脚步声为例，可以将 Dialogue 放入 BankEvent 音频包中，将混凝土地材质相关的结构和媒体放入 BankConcrete 音频包，泥土地材质相关的结构和媒体放入 BankDirt，以此类推。这种情况下，只需根据即将遇到的地表材质类型，动态加载和卸载音频包即可。
- 将 Switch Container 层级结构拆分为多个 SoundBank。将 Switch Container 添加至 SoundBank 时，会自动包含其所有子分支。但您可以通过 SoundBank Editor（音频包编辑器）的 Game Syncs（游戏同步器）或 Edit（编辑）选项卡，手动排除某些子分支。假如有一个 Footstep 层级结构，其中顶层 Switch 变量为地表材质类型。如果游戏中并不会一直出现所有地表材质类型，就可以将 Switch Container 层级结构拆分并置于不同的 SoundBank 中，然后根据游戏情况，仅加载所需的表面类型。例如，您可在“Footstep”SoundBank 中包含游戏中随处可见的地表材质（如水泥 Concrete 和金属阶梯 Metal Stairs），而对于游戏中某个场景或环节的特定地表材质（例如泥地 Mud），则置于其它动态加载的 SoundBank 中。

对于撞击类的各种不同声音，可以将 WAV 文件替换为等价的 SoundSeed Impact 插件。将 10 个不同的 Clang（金属碰撞的叮当声）声音替换为 1 个不仅合算，而且还能带来 10 种以上变化。不要忽视这种可能性，因为它确实很有效。



媒体内存池（用于 SoundBank）

SoundBank（音频包）的内存占用基本取决于其中包含的声音数据量。以下技巧可以控制声音媒体的内存占用：

- 将包含很多声音的大音频包拆分为较小的音频包。根据需要来动态加载和卸载音频包。
- 让更多的声音从磁盘进行流播放（要修改声音对象属性）。

- 使用 PrepareEvent/UnprepareEvent API 来准备/取消准备事件。
- 采用更高的音频压缩比（使用转码设置，编解码格式等）。
- 使用更低的采样率，还可以考虑用 Automatic Sample Rate Detection（自动采样率检测）功能（使用转码设置）。
- 用 SoundSeed Impact 插件替换撞击类声音。这样可以减少包含多个声音的 Random Container 数量，帮您节省大量内存。请注意，SoundSeed Impact 还可以用来创建某些非撞击声的变体。
- 用 SoundSeed Wind/Woosh 插件替换风声类的声音。风声作为长时间循环声音，会占用大量空间。SoundSeed Air 插件可用于创建各种气流声，包括利器挥舞声、螺旋桨声、车窗敞开时的气流声、通风系统噪声或任何嘈杂的声音，如海浪或远方高速公路的声音。

内存管理技巧和经验总结

以下章节介绍了一系列技巧和经验总结，帮您高效管理游戏中的音频内存。

使用 Memory Threshold

正如之前章节所述，您还可以为声音引擎的各内存池设置内存阈值。音频程序员可以在声音引擎的初始化参数中修改以下值，来设置内存池的 Memory Threshold（内存阈值）：

- AkInitSettings::fDefaultPoolRatioThreshold
- AkPlatformInitSettings::fLEngineDefaultPoolRatioThreshold

默认情况下，内存阈值被禁用，即使用默认值 1（即 100%）。您可以将其设置为 0 至 1 之间的任意值，来启用内存阈值。

禁用内存阈值后，内存分配器将正常工作。启用后，引擎会定期检查内存占用百分比是否低于该阈值。如果超出阈值，系统将开始忽略优先级最低的声音。

未启用内存阈值时，引擎将按优先顺序播放声音，但内存过低时，引擎可能无法播放高优先级声音，因此将忽略该声音。启用内存阈值时，低优先级的声音将被忽略，给高优先级声音腾出空间。

处理声音引擎中的内存碎片

每次在内存中添加和移除数据块时，都会产生一定数量的碎片。Wwise 本身不会处理内存碎片，但内存池的设计可以帮助减少碎片量。

加载声音结构元数据的 Default memory pool（默认内存池）里主要是小对象，因此出现碎片并不是问题。用于音频管线和处理的 Lower engine memory pool（底层引擎内存池）包含较大的数据块，但当音频终止时，这些数据块都将被释放。对于上述两种内存池，只需分配约 10% 的余量，几乎就能完全避免碎片问题。

存储在 SoundBank（音频包）中的音频数据，每次 LoadBank() 调用时都会拷贝到独立的 User-defined pool（用户定义内存池）中，以便音频程序员更好地控制音频数据的内存管理。为最大程度降低出现内存碎片的可能性，音频程序员可以尝试让内存池使用固定块大小，将 AkFixedSizeBlocksMode 标识传递给 CreatePool，并以该内存池中最大的 SoundBank 大小作为数据块大小。这样分配给各 SoundBank 的内存量都

将与最大音频包相等。当然，您还可以创建多个内存池，每个都含有特定数量的槽/块，以供容纳特定大小的音频包。AkFixedSizeBlocksMode 内存池的数据块大小全部相同，并可以通过 GetBlock() 查询。使用这一技巧可以完全避免内存碎片。

第 34 章 性能分析

概述	587
连接到本地/远程游戏系统	595
从声音引擎捕获数据	597
使用 Performance Monitor 进行监控和故障排查	606
使用 Voice Inspector 分析声部	608
使用 Game Object Explorer 跟踪对象和听者	614
通过 Game Object 3D Viewer 检查对象	618
通过 Game Sync Monitor 评估游戏同步器	623
性能分析技巧和最佳实践	624

概述

在完成工程之前，您肯定会想测试一下工程的声音效果，并查明设计方面的潜在问题。为此，可直接在 Wwise 中执行性能分析。当然，您也可以在工程前期根据需要随意使用这一功能。

在 Wwise 中，可采用各种方式来针对不同平台量身打造游戏。利用 Profiler（性能分析器）布局，测试音频和振动效果在各个平台上的性能，从而在一定程度上确保平台之间的一致性。

在制作过程中，随时都可利用 Profiler 布局，来针对平台分析游戏在特定方面的性能。您可以在 Wwise 中执行[本地性能分析](#)，也可以[通过连接到远程游戏主机来执行性能分析](#)。两种方式均可直接从声音引擎捕获性能分析信息（[如下所述](#)）。通过监控声音引擎活动，您可以检测并排查内存、声部、流播放、效果器及其他相关问题。将原型集成到游戏中之前，还可以使用 Game Simulator 和 Soundcaster 进行性能分析。

有关如何执行性能分析的详细信息，请参阅以下子页面并简要了解 Wwise 中提供的不同类型的性能分析：

- [理解 Wwise 中不同类型的性能分析](#)
- [探索游戏性能分析视图](#)
- [了解声部性能分析视图](#)
- [了解游戏对象性能分析视图](#)

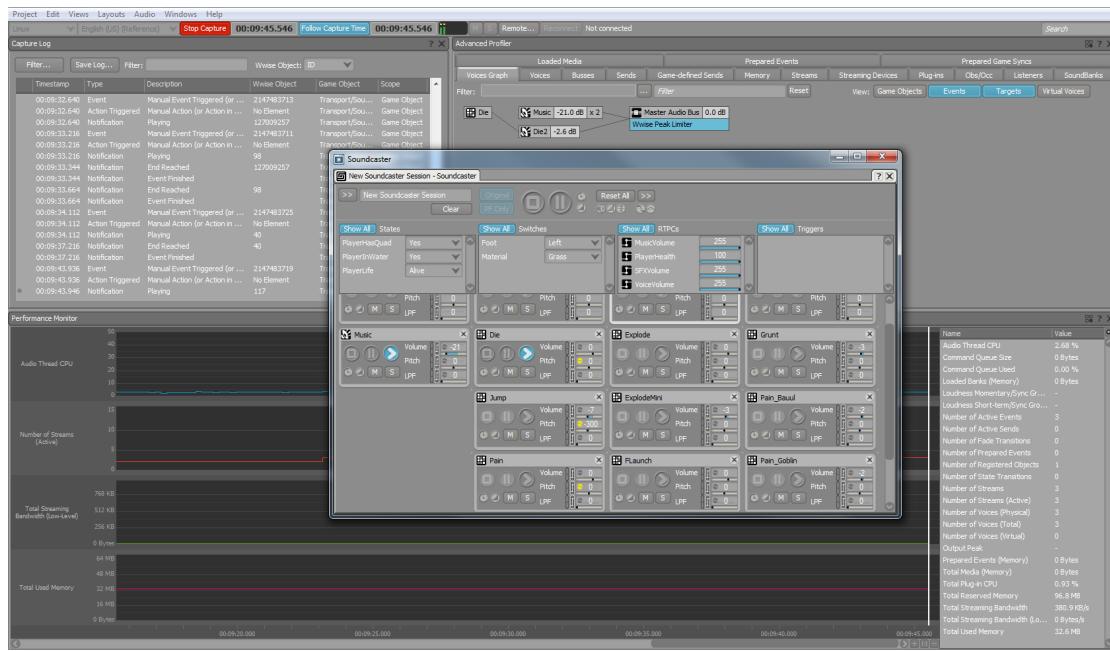
本地性能分析

音频集成到游戏中之前，通过在 Soundcaster 中触发事件、声音和振动对象以及游戏同步器，您可以对 Wwise 工程进行多方面的性能分析。

本地性能分析

1. 切换至 Profiler 布局。
2. 在 Soundcaster 中添加事件、对象和总线。
3. 点击 Start Capture（开始捕获）。
4. 触发 Soundcaster 中的事件、对象或游戏同步器。

声音引擎的相关信息将被捕获并显示在 Profiler 的各视图中。



在游戏过程中进行性能分析

将音频和振动整合到游戏后，您可以将 Wwise 连接到游戏，然后在游戏过程中进行实时性能分析。连接游戏后，还可以实时微调游戏中的对象属性并进行混音。



注意

进行性能分析时，不能使用 Release 版本配置，而且 Wwise 声音引擎的 Debug 版本配置尚未经过优化。因此，即使游戏运行的是 Debug 版本，我们也不建议您使用 Wwise 的 Debug 版本配置。建议您进行性能分析时，连接到 Wwise 声音引擎的 Profile 版本配置。

对游戏进行性能分析的方法如下：

1. 启动游戏。
2. 切换至 Profiler 布局。
3. 使用 Remote Connection（远程连接）连接到游戏。
4. 点击 Start Capture（启动捕获）。
5. 开始玩游戏。

声音引擎的相关信息将被捕获并显示在 Profiler 的各视图中。

理解 Wwise 中不同类型的性能分析

在 Wwise 中，有各种类型的性能分析，其分析范围或宽泛或具体：

- 游戏性能分析：着重从声音引擎和工程所含各种组件的角度来分析性能要求和需求。它反映了工程中的声音对平台性能的实时累积影响，并可以查看各声部的影响。
- 声部性能分析：从单个声部的角度监控相应输出。声部性能分析会详细列出不同对象的各种设置，以便确定游戏中声部的实时输出是否合理。

- 游戏对象性能分析：从单个游戏对象的角度分析声音引擎的输出。游戏对象性能分析可以跟踪游戏对象，让您实时观察这些对象的运动和行为，从而找到有问题的游戏对象。

游戏对象是游戏中的独立实体，对于游戏中所有能作为发声体或听者的元素，音频程序员都会为其注册或创建游戏对象。这些元素包括玩家角色、非玩家角色（NPC）、武器、车辆、怪物以及周围环境中的物体（例如火把）等。游戏对象经过编程后能够独立移动，它们可以关联声音、音乐、振动或能够定位的总线对象。可以同时使用多个游戏对象性能分析工具（Game Object Explorer、3D Game Object Viewer 和 Game Sync Monitor），在游戏或模拟过程中检查游戏对象。

对游戏声音进行性能分析——示例

假设您正在制作一款格斗游戏，在游戏中玩家控制巨兽参加在宏大城市中心的战斗。您可以使用游戏性能分析和游戏对象性能分析工具来分析游戏声音的性能。

用户可以用游戏性能分析工具分析下列内容：

- 怪兽、警察和旁观者相关的大量声音消耗了多少平台的流播放能力。
- 有多少背景噪声（例如建筑物倒塌声）进入了虚声部（Virtual Voice），它们是在何时进入虚声部的。
- 哪些效果器插件被施用在不同的怪兽咆哮声上，这些效果器的 CPU 占用率如何。
- 时间线上的某些特定时刻会调用哪些 API 函数。

您可以使用声部性能分析工具来执行以下分析：

- 为何听不到警笛声。
- Reverb Effect 如何影响特定位置处旁观者的尖叫声。
- API 调用如何在具体时间点影响某个特定声部。

用户可以使用游戏对象性能分析工具分析下列内容：

- 游戏中各个怪兽声音的衰减半径如何与其它怪兽声音的衰减半径相互作用。
- 警察直升机等游戏对象相对于彼此以及相对于怪兽移动到了何处。
- RTPC（例如“Panic”）如何影响与旁观者游戏对象相关联的声音播放。

这样，游戏性能分析和游戏对象性能分析工具可以让用户全面地了解运转中的游戏声景。

探索游戏性能分析视图

Wwise 的游戏性能分析工具由多个视图构成，这些视图共同帮助您分析工程的详细性能。这些视图已合并为 Profiler 布局。



技巧

在 Wwise 中，默认情况下您可以按 F6 来切换到 Profiler 布局。

Capture Log

Capture Log（捕获日志）可捕获并记录来自声音引擎的所有信息。您可以使用 Capture Log Filter 来确定 Capture Log 中显示的信息。若要限制捕获哪些数据类型，请调整 Advanced Profiler 的 Profiler Settings。

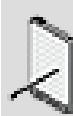
Timestamp	Type	Description	Wwise Object	Game Object	Scope
00:00:51.968	Error	Unload bank failed, request...			
00:00:51.968	Bank	Bank Unload Request Recei...	2185786256		
00:00:51.968	Bank	Bank Unloaded (from Gravel)	2185786256		
00:00:54.336	Bank	Bank Load Request Receive...	983016381		
00:00:54.357	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

相关主题

- 从声音引擎捕获数据

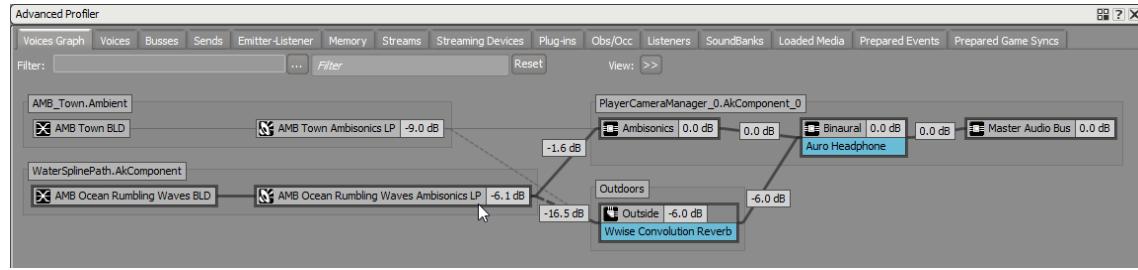
Advanced Profiler

Advanced Profiler（高级性能分析器）包含有关声音引擎执行的各个活动的性能相关信息，例如 CPU、内存和带宽。一旦从声音引擎捕获到这些信息，它们就会被实时显示出来。您可以在 Profiler Settings 对话框中确定 Advanced Profiler 的选项卡上显示的信息类型。



备注

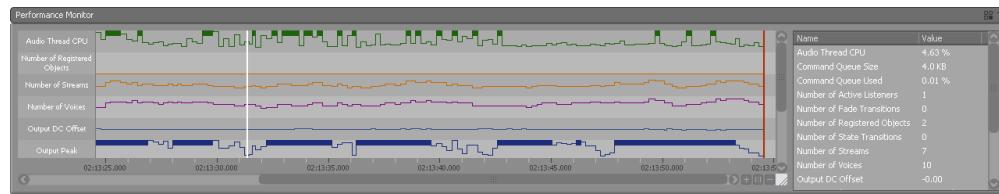
只有混音总线才会显示在 Voices Graph 和 Buses 选项卡中。有关两类总线区别的详细信息，请参阅[混音与非混音总线/](#)。



Advanced Profiler 中显示了 Wwise Audio Lab 中的一个 Capture，其中玩家站在村庄户外

Performance Monitor (性能监视器)

Performance Monitor (性能监视器) 是一整套声音引擎指标，可帮助您监控性能和排查问题。您可以配置 Performance Monitor 来让它只显示您需要的信息。



备注

如以上截屏中 Audio Thread CPU 和 Output Peak 两行所示，曲线图顶部的粗线表示当前值已经超出了 Graph Max (图表上限) 值。

相关主题

- [从声音引擎捕获数据](#)
- [使用 Performance Monitor 进行监控和故障排查](#)
- [附录 C, 快捷键](#)

了解声部性能分析视图

Wwise 的声部性能分析工具作为对[游戏性能分析](#)的补充，可帮助分析工程的基本构成元素：声部。

Voice Profiler (声部性能分析器) 布局整合了其他布局中常用的多个视图，其中有两个主要用来执行声部性能分析：[Voice Inspector](#) 和 [Voice Monitor \(声部监视器\)](#)。



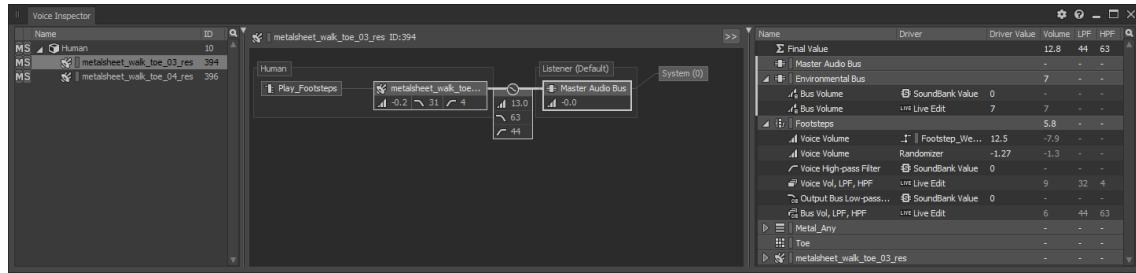
技巧

在 Wwise 中，默认情况下您可以按 F11 来切换到 Voice Profiler 布局。

Voice Inspector

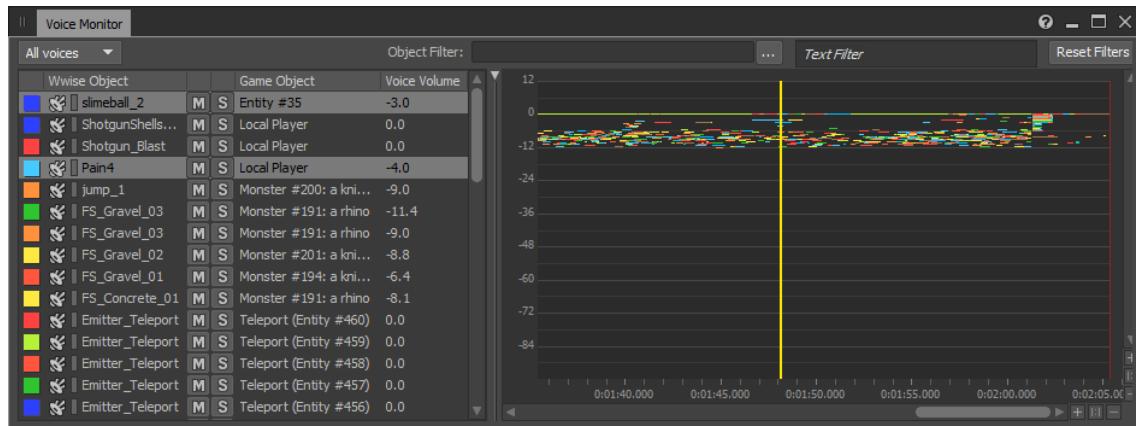
在捕获会话过程中和完毕后，Voice Inspector (声部检视器) 会根据 Performance Monitor (性能监控器) 或 Voice Monitor (声部监控器) 中选定的时间点，在第一个面板中显示捕获的活跃声部列表。您可以从这些声部中选择一个来显示在右侧的

Voice Graph (声部图) 和 Contribution List (影响列表) 面板中。每个面板均显示声部工作流和影响最终音量输出的元素的信息。



Voice Monitor (声部监视器)

在捕获会话过程中和完毕后，Voice Monitor 会显示捕获信息时间线，并列出播放的每一个声部及其相关音量。在时间线上选择某个时间点时，将列出当时活跃的所有 Wwise 和关联游戏对象及其声部音量。



相关主题

- [使用 Voice Inspector 分析声部](#)
- [从声音引擎捕获数据](#)

了解游戏对象性能分析视图

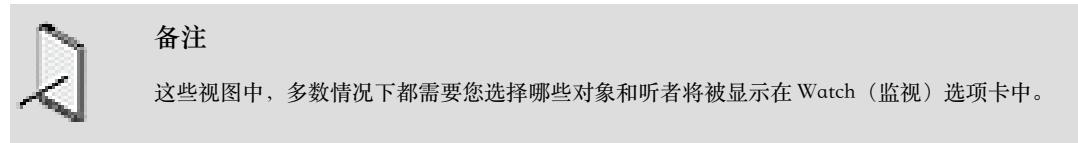
Wwise 的游戏对象性能分析工具包括三个视图，通过配合使用，可以帮助您监控游戏对象以及它们对游戏音频和振动的影响。这三个视图在 Game Object Profiler 布局中可以同时查看。



Game Object Explorer 视图

Game Object Explorer 是 Wwise 游戏对象性能分析工具的控制中心。音频程序员在游戏中注册了游戏对象，就可以在 Game Object 3D Viewer、Game Sync Monitor 和

RTPC（实时参数控制）坐标图视图以及 Attenuation Editor 中实时“监视”它们。连接到游戏后，游戏对象列表即显示出游戏当前已注册的所有游戏对象。

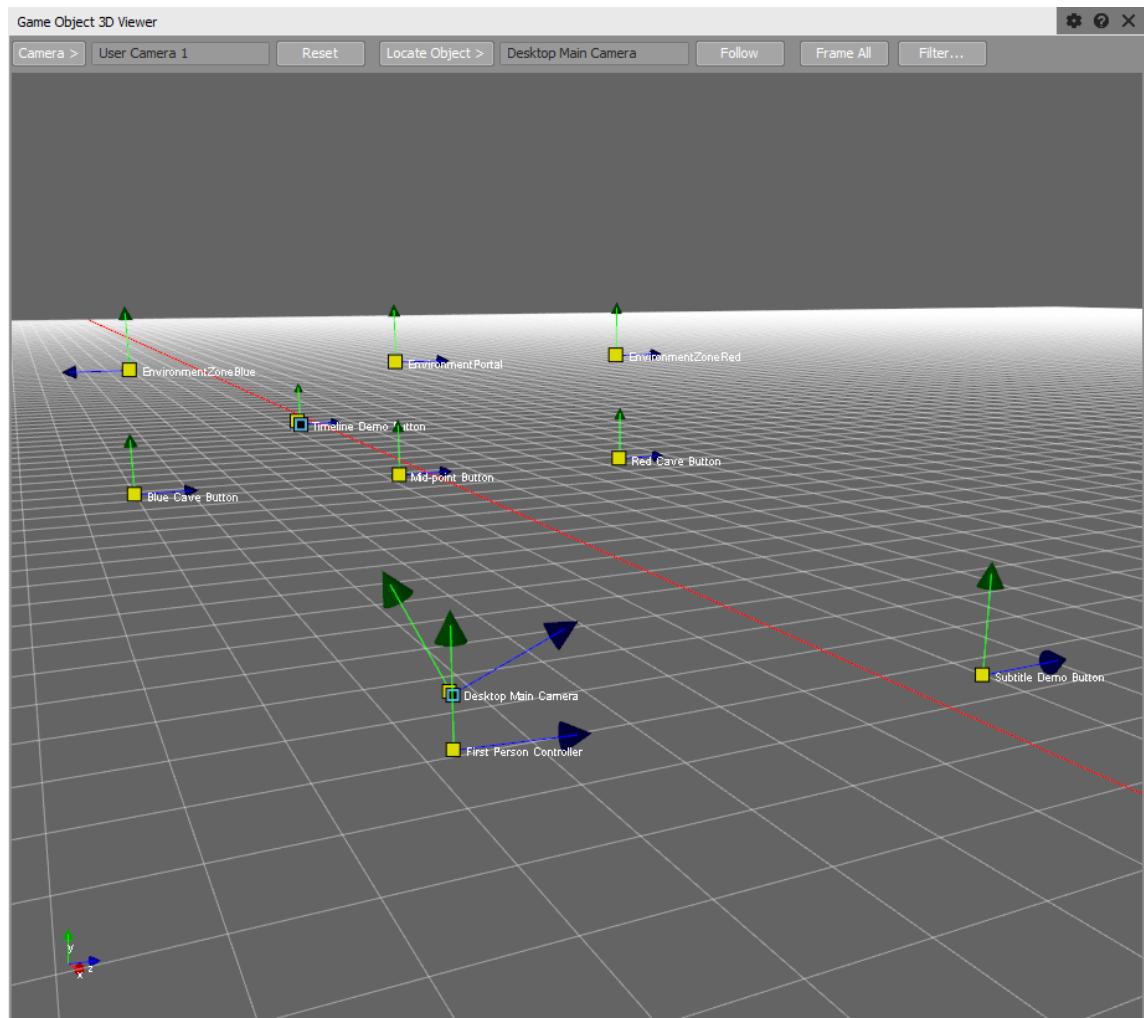


The screenshot shows the Unity Editor's Game Object Explorer window. It has tabs for 'Game Objects' and 'Watches', with 'Game Objects' selected. A button labeled 'Is Alive' is visible. The main area is a table listing registered game objects:

Name	ID	Last Registered	Last Unregistered	Is Alive
AKPhysicsTriggerWaterImpact...	M S 19136937	0:10:40.725	-	✓
AMBIENCE	M S 444443	0:10:40.725	-	✓
Boy	M S 196665	0:10:40.725	-	✓
CollisionRect2D	M S 44761606	0:10:40.725	-	✓
CollisionRect2D	M S 786709	0:10:40.725	-	✓
door	M S 26411081	0:10:40.725	-	✓
Listener (Default)	M S 10002	0:10:40.725	-	✓
pelvis	M S 196646	0:10:40.725	-	✓

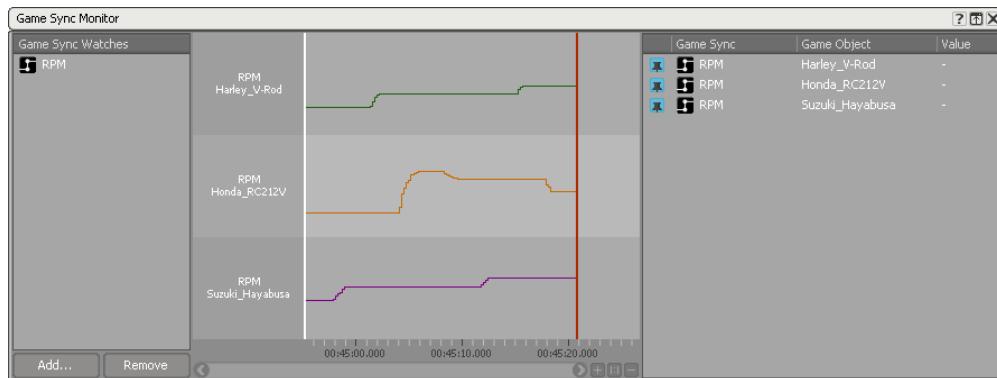
Game Object 3D Viewer 视图

Game Object 3D Viewer 即三维对象查看器，能够查看游戏对象、与其他游戏对象的关系，以及它们成为听者之后的情况。您可以对Game Object 3D Viewer 进行设置，使其显示不同类别的数据并改变不同游戏对象的呈现方式。



Game Sync Monitor 视图

Game Sync Monitor（游戏同步器监控器）是实时分析 RTPC 数值的工具。游戏中，将针对所监视的游戏对象实时显示 RTPC 变化曲线。



相关主题

- [从声音引擎捕获数据](#)

- 通过 Game Sync Monitor 评估游戏同步器
- 通过 Game Object 3D Viewer 检查对象
- 使用 Game Object Explorer 跟踪对象和听者

连接到本地/远程游戏系统

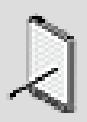
如果想要模拟不同声音在游戏中的效果，或在特定平台上对游戏各方面进行性能分析和故障排查，您首先需要连接运行游戏的设备。局域网中正在运行且未被占用的 Wwise 声音引擎都可以连接。



注意

进行性能分析时，即使游戏使用了 Debug Build，也建议连接到 Wwise 声音引擎的 Profile Build 配置，因为 Debug Build 配置的性能尚未优化。

为帮您找到设备，Wwise 会自动搜索同一子网段上的所有 PC 和游戏机，寻找当前运行的 Wwise 声音引擎。您还可以手动输入平台的 IP 地址，连接到子网段外的游戏机或 PC。



备注

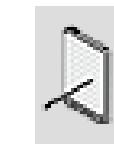
如果游戏在 Wwise 中的 Remote Connection（远程连接）对话框中没有显示，请参阅[排查远程连接问题](#)来解决问题。

连接到本地/远程游戏系统的方法如下：

1. 在工具栏中，单击 Connect to Remote Platform（连接至远程平台）按钮 。

Remote Connection 对话框将打开，列表中显示了当前网络中运行 Wwise 声音引擎的游戏系统。

2. 从列表中选择要连接的平台。

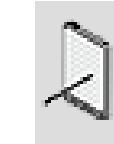


备注

设备的状态须显示为“Ready”（就绪），才能连接该设备。

3. 要想从连接起始就捕获信息，请选择 Start capture on connect（连接时开始捕获）选项。
4. 点击 Connect（连接）。

Wwise 将连接到本地或远程平台。设备的名称会显示在工具栏中。



备注

您也可以手动连接平台，点击 Connect To IP（连接到 IP）然后键入远程设备的 IP 地址即可。

从本地/远程设备上断开连接的方法如下：

1. 在工具栏上，单击 **Disconnect from Remote Platform**（从远程平台断开）按钮 。

Wwise 将断开与该设备的连接。

相关主题

- [使用远程连接历史记录列表](#)
- [从声音引擎捕获数据](#)
- [指定网络端口](#)
- [排查远程连接问题](#)

使用远程连接历史记录列表

Wwise 会将成功连接的设备/系统保存至历史记录，从而让您轻松找到曾经使用的系统。连接设备之前必须首先验证其状态，设备状态可以是以下类型之一：

- Ready（就绪）：设备处于就绪状态，可以连接。
- Busy（繁忙）：设备已存在其他连接，因此无法连接。
- Different Version（版本不同）：Wwise 版本与远程系统的游戏版本不兼容。
- Not Available：不可用。设备不再与网络相连。

当您希望设备不再出现在历史记录列表中时，可以清空历史记录。

使用历史记录列表连接到系统的方法如下：

1. 在工具栏中，单击 **Connect to Remote Platform**（连接至远程平台）按钮 。
Remote Connection 对话框将打开，其中显示当前网络中运行游戏的设备列表。
2. 切换至 History 选项卡。
3. 从 History 列表中选择要连接的设备。
4. 如果希望从连接起始就捕获信息，则请选择 **Start capture on connect** 选项。
5. 点击 **Connect**。

Wwise 将连接到本地或远程设备。设备的名称会显示在工具栏中。

清空历史记录列表的方法如下：

1. 在工具栏中，单击 **Connect to Remote Platform**（连接至远程平台）按钮 。
Remote Connection 对话框将打开，其中显示当前网络中运行游戏的设备列表。
2. 切换至 History 选项卡。

3. 从 History 列表中选择要删除的设备。
4. 执行以下操作之一：
 - 点击 Remove。
 - 按 Delete 键。

设备将从 History 列表中删除。

相关主题

- [连接到本地/远程游戏系统](#)
- [从声音引擎捕获数据](#)
- [排查远程连接问题](#)

从声音引擎捕获数据

连接到机器后，您可以直接从声音引擎捕捉数据，来为游戏音频和振动进行性能分析。所有来自声音引擎的信息都将显示在 Capture Log 中。其中将记录以下类型的信息条目：

- Notifications (通知)
- Markers (标记)
- Events (事件)
- Actions (动作)
- Properties (属性)
- States (状态)
- Switches (切换开关)
- SoundBanks (音频包)
- Events Preparation (事件准备)
- Errors (错误)
- Messages (消息)
- MIDI Events (MIDI 事件)
- API Calls (API 调用)
 - Event
 - RTPC
 - Game Syncs (游戏同步器)
 - Game Objects (游戏对象)
 - Listener Spatialization (空间定位)
 - Positions (定位)
 - Auxiliary Bus (辅助总线)
 - Obstruction/Occlusion
 - Motion (振动)
 - MIDI

- Dynamic Sequence (动态序列)
- Spatial Audio (空间音频)
- 其他

您可以使用 Performance Monitor 和 Advanced Profiler 监控这里的各条记录。这些视图包含有内存、声部和效果器的详细使用情况，以及流播放、SoundBank和插件等相关的详细信息。有关监控性能的详细信息，请参阅[使用 Performance Monitor 进行监控和故障排查](#)。您也可以使用 Wwise 来分析捕获的数据，并获取游戏或工程中特定音频元素的统计信息。有关获取统计信息的详情，请参阅[从 Capture Session 获取统计信息](#)。

Wwise 使用以下特殊标记和标颜色，来帮您对 Capture Log 中的条目进行快速分类。

界面元素	描述
标记	
圆点 	圆点的颜色随底色变化：在 Classic (经典) 主题下显示为青绿色，在 Dark (深色) 主题下显示为橙色。这样方便指示在距离 Performance Monitor 时间光标位置 100 ms 内捕获了 Capture Log 中的哪些条目。颜色越亮，条目离时间光标位置越近。您可以将光标强制移动至特定日志条目所在的时间戳：从快捷菜单(Ctrl+V)选择 Move Cursor to Timestamp (将光标移至时间戳)。或者直接双击时间戳。
互连圆点 	关联圆点的颜色随底色变化：在 Classic (经典) 主题下显示为深灰色，在 Dark (深色) 主题下显示为白色。这样方便指示 Capture Log 中哪些条目是相互关联的。这些圆点仅在选中日志中的条目时显示。
颜色	
选定条目 	Capture Log 中所选条目会高亮显示为灰色对比色调，该色调随所选主题变化 (Classic 主题下高亮显示为深灰色，Dark 主题下高亮显示为浅灰色)。  备注 您可以选择多个条目，不过会有部分显示和快捷菜单选项不可用。使用标准多选选项：在按住 Shift 的同时连续选择一系列条目，或者在按住 Ctrl 的同时逐一选择多个日志条目。
相关条目 	所有与选中条目相关的条目均高亮显示为灰色对比色调（比所选条目的颜色更深或更浅），该色调随主题变化。
Error 	捕获过程中出现的所有错误均显示为红色。
Messages (消息) 	所有消息均显示为黄色。

下图说明了各种标记和颜色在 Capture Log 中如何使用。

Errors are displayed in red.

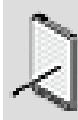
Messages are displayed in yellow.

Turquoise indicators show entries captured within 100 ms of the Performance Monitor time cursor.

Gray indicators show which entries are related. These entries are also highlighted gray.

Timestamp	Type	Description	Wwise Object Name	Wwise Object ID	Game Object Name
0:00:50.880	Message	Some errors occurred ...			(Global)
0:00:50.880	Error	Event ID not found	Monsters_Aware	3430883159	166811904 (Global)
0:00:50.880	Message	End of pre-connectio...			
0:00:50.880	Switch	Switch to "Left"	Foot	3031504797	Local Player
0:00:50.880	Switch	Switch to "Left"	Foot	3031504797	Monster #130: a rhino
0:00:50.880	Switch	Switch to "Left"	Foot	3031504797	Monster #192: a knight
0:00:50.880	Switch	Switch to "Right"	Foot	3031504797	Monster #194: a knight
0:00:50.880	Switch	Switch to "Left"	Foot	3031504797	Monster #193: a knight
0:00:50.880	Switch	Switch to "Right"	Foot	3031504797	Monster #195: a knight
0:00:50.880	Switch	Switch to "Right"	Foot	3031504797	Monster #196: a knight
0:00:52.736	Switch	Switch to "Grass"	Material	3865314626	Monster #130: a rhino
0:00:52.736	Switch	Switch to "Left"	Foot	3031504797	Monster #130: a rhino
0:00:52.736	Event	Event Triggered	Foot_Rhino	3826003282	Monster #130: a rhino
0:00:52.736	Action Triggered	Play	Footsteps	76	Monster #130: a rhino
0:00:52.736	Event	Event Triggered	Jump	3833651337	Monster #126: a goblin
0:00:52.736	Action Triggered	Play	Jump	157860496	Monster #126: a goblin
0:00:52.736	Notification	Playing	FS_Gravel_02	67	Monster #130: a rhino
0:00:52.736	Notification	Playing	Jump_4	238179141	Monster #126: a goblin
0:00:52.757	Notification	Delay Ended	Fire_Shotgun_Player [Play - S...	617065999	Local Player
0:00:52.757	Action Triggered	Play	Shotgun Shells	783703556	Local Player

将您的设计和原型集成到游戏中之前，通过在 Wwise 中的 Soundcaster（声音选角器）中直接进行捕获，还可以事先对其进行性能分析。有关使用 Soundcaster 的详细信息，请参阅[第 32 章 创建模拟](#)。



备注

对游戏进行性能分析之前，应将已卸载的工作单元全部加载到工程中。工作单元被卸载时，性能分析信息可能不完整并造成误导。

指定要捕捉的信息类型

由于传输的数据量可能很大，所以从声音引擎捕获信息可能会影响 Wwise 的性能。因此您可能需要在 Profiler Settings（性能分析设置）中进行设置，限制 Advanced Profiler 生成的信息类型。通过排除一些信息类型，您可以：

- 节省网络传输带宽。
- 节省 Wwise 中的内存空间。
- 省略数据计算，从而节省游戏中的 CPU 时间。
- 省略数据处理或显示，从而节省 Wwise 中的 CPU 时间。

您还可以设置在性能分析会话文件中，是否需要存储 Capture Session（捕获会话）中 Profiler 和 Game Object Profiler 布局中所有视图的全部内容。该会话文件内的数据可供 Wwise 稍后重新加载，从而进一步分析游戏中的音频。



备注

当超出限制数量时，Wwise 会删除最早的捕获会话。因此，如果您想保存早期的捕获会话，必须在每次性能分析后重命名最早的文件，也可以增大 Number of sessions kept。

指定生成和捕获的信息类型的方法如下：

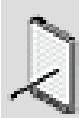
- 执行以下操作之一：

- 在 Advanced Profiler 标题栏中，点击 View Settings（视图设置）图标。
- 在菜单栏中点击 Project > Profiler Settings（工程 > Profiler 设置）。
- 按 Alt+G。

Profiler Setting 对话框将打开。

2. 从以下信息类型中选择：

- **Plug-in Data**（插件数据），捕获各种插件的相关信息。
- **Memory Data**（内存数据），捕获声音引擎 Memory Manager 中已注册内存池的相关信息。
- **Stream Data**（流播放数据），捕获声音引擎管理的播放流相关的信息。
- **Voices Data**（声部数据），捕获声音引擎管理的各声部相关的信息。
- **Listener Data**（听者数据），捕获声音引擎管理的各个听者相关的信息。
- **Obstruction/Occlusion Data**（声障/声笼数据），捕获声障和声笼影响的游戏对象相关的信息。
- **Markers Notification Data**（标记通知数据），捕获音频文件中的 Marker（标记）以及音乐段落中 Custom Cue（自定义提示点）相关的通知信息。
- **Output Data**（输出数据），捕获输出峰值和输出 DC offset（直流偏置）的相关信息。
- **SoundBanks**（声音库），捕获已加载至内存中的 SoundBank 相关的信息。
- **Loaded Media**（加载的媒体），捕获已加载至内存中的媒体相关的信息。
- **Prepared Events**（预备事件），捕获由 PrepareEvent API 函数进行“预备”操作的事件相关的信息。
- **Prepared Game Syncs**（预备游戏同步器），捕获由 PrepareGameSyncs API 函数进行“预备”操作的游戏同步器相关的信息。
- **Music Transitions**（音乐过渡），捕获音乐对象间过渡的相关信息。
- **Streaming Device Data**（流播放设备数据），捕获所有流播放设备中的数据。
- **Meter**（电平表），捕获信号测量数据的相关信息，供总线与仪表视图中的电平表所使用。
- **Auxiliary Sends Data**（辅助发送总线数据），捕获声音引擎管理的各辅助发送总线的相关信息。
- **API Calls**（API 调用），捕捉 Wwise API 调用的相关信息。



备注

在排除某个信息类型时，相应的选项卡会从 Advanced Profiler 中移除。

3. 在 Capture Log 下面键入最大内存使用量，最小为 10 MB，最大为 4000 MB。

Game Profiler 使用的内存量将不超过您为 Capture Log 指定的值。

4. 使用 Number of sessions kept（保留的会话数量）来指定将存储多少个性能分析会话文件，数量超出时，最早的文件将被删除。
5. 点击 OK。

Game Profiler 将只捕获所选信息类型的数据。

相关主题

- 启动/停止捕获过程
- 筛选 Capture Log
- 使用 Capture Log 快捷菜单
- 从 Capture Session 获取统计信息
- 保存 Capture Log
- 从已有的远程 Capture Session 加载数据

启动/停止捕获过程

开始捕获声音引擎中的信息后，Advanced Profiler、Capture Log 和 Performance Monitor 中将会自动记录数据。可以在捕获同时查看相应信息，也可以停止捕获并调查出现的问题。想要在捕获同时查看相应信息，可以使用 >Follow Capture Time>（跟随捕获时间）选项让 Capture Log 和 Performance Monitor 随捕获时间更新。



启动捕获的方法如下：

1. 执行以下操作之一：
 - 在菜单栏中，点击 Layouts（布局）> Profiler（Profiler）。
 - 按 F6 键。
 Profiler 布局将显示。
2. 想要在捕获数据的同时监控性能，请点击工具栏上的 Follow Capture Time  按钮。
3. 选择此项的作用包括：
 - 捕获到数据时 Capture Log 将自动滚动。
 - Performance Monitor 时间光标将跟随游戏时间光标。
4. 要停止捕获，请点击 Stop Capture（停止捕获）。

相关主题

- [连接到本地/远程游戏系统](#)
- 筛选 Capture Log
- 使用 Capture Log 快捷菜单
- 从 Capture Session 获取统计信息

- [保存 Capture Log](#)
- [从已有的远程 Capture Session 加载数据](#)
- [使用 Performance Monitor 进行监控和故障排查](#)

筛选 Capture Log

您可能发现 Capture Log 中显示的信息过多了，这种情况下可以筛选 Capture Log，只显示要查看的信息。例如，您可以仅显示特定游戏对象的相关信息，或事件、状态相关的信息。

筛选 Capture Log 的方法如下：

1. 在 Capture Log 中，点击 Filter（筛选器）。

Capture Log Filter（捕获日志筛选器）对话框将打开。

2. 将希望显示在 Capture Log 中的选项勾选，即可筛选捕获信息。



3. 点击 OK。

Capture Log 将根据所选条件进行筛选。

相关主题

- [启动/停止捕获过程](#)
- [使用 Capture Log 快捷菜单](#)
- [从 Capture Session 获取统计信息](#)
- [保存 Capture Log](#)
- [从已有的远程 Capture Session 加载数据](#)

使用 Capture Log 快捷菜单

在已载入信息的 Capture Log（捕获日志）中，右键单击一个或多个条目以显示快捷菜单。除标准[快捷菜单选项](#)外，Capture Log 本身还提供一些其他选项。

- **Copy Text to Clipboard**（将文本复制到剪切板）：将选中列信息（包括列标题）以制表符分隔格式复制到剪切板。
- **Move Cursor to Timestamp**（将光标移至时间戳）：将光标移至所选时间戳，并更新所有 Profiler（性能分析器）视图，包括 Performance Monitor（性能监控器）和 Advance Profiler（高级性能分析器）。
- **Move to Previous Related Item**（移至上一相关条目）：选中当前所选条目之前的第一个相关条目（如适用）。
- **Move to Next Related Item**（移至下一相关条目）：选中当前所选条目之后的第一个相关条目（如适用）。

Copy Text to Clipboard	
Move Cursor to Timestamp	Ctrl+\
Move to Previous Related Item	Ctrl+[
Move to Next Related Item	Ctrl+]
Add Game Object to Game Object Watch List - By Id	
Add Game Object to Game Object Watch List - By Name	
Mute Game Object	
● Unmute Game Object	
Solo Game Object	
● Unsolo Game Object	
Search Game Object in Voice Graph	



技巧

凡是列有游戏对象的地方（都提供对象专有快捷菜单，包含以下选项：

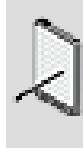
- [Add Game Object to Game Object Watch List - By ID](#): 将所选游戏对象的 ID 添加到监视列表。
- [Add Game Object to Game Object Watch List - By Name](#): 将所选游戏对象的名称添加到监视列表。
- [Mute Game Object](#): 将该游戏对象关联的所有音频对象静音。
- [Unmute Game Object](#): 为该游戏对象关联的所有音频对象取消静音。
- [Solo Game Object](#): 间接地将其他游戏对象关联的所有音频对象静音。
- [Unsolo Game Object](#): 为其他游戏对象关联的所有音频对象取消静音。
- [Search Game Object in Voice Graph](#): 打开 Advanced Profiler 的 Voices Graph 选项卡，将其 Filter 设置为指定的游戏对象。

相关主题

- [启动/停止捕获过程](#)
- [筛选 Capture Log](#)
- [从 Capture Session 获取统计信息](#)
- [保存 Capture Log](#)
- [从已有的远程 Capture Session 加载数据](#)

从 Capture Session 获取统计信息

在 Capture Session（捕获会话）期间或捕获结束后，您可以随时用 Wwise 分析捕获的信息，来获取 Wwise 工程或游戏中特定音频元素相关的统计数据。



备注

目前，仅限 Dynamic Dialogue（动态对话）的相关信息可以从捕获会话中提取。为了完全理解统计信息，必须明确 Wwise 生成 Dynamic Dialogue 时涉及的各个步骤。有关 Dynamic Dialogue 的详细信息，请参阅[理解 Dynamic Dialogue 系统](#)。

Profiler Statistics（Profiler 统计数据）视图显示了从 Capture Session 中提取的统计信息，其中包含供查询使用的预设，可以帮您获取 Capture Log 内各种信息类型的统计数据。比如，您可以获悉在 Capture Session 期间解析了哪些路径，解析次数和致使音频播放的次数。还可以筛选信息来缩小搜索范围，发现问题并确定必要的对策。

所有激活的 Capture Session 都可以用来提取统计数据，即对于游戏远程捕获的数据或 Wwise 本地捕获的数据，都可以提取相关统计信息。只要 Capture Log 中显示信息，就说明 Capture Session 处于激活状态。

从 Capture Session 收集统计信息的方法如下：

1. 启动本地或远程 Capture Session。
2. 在菜单栏中，点击 Views > Profiler Statistics（视图 > Profiler 统计数据）。

Profiler Statistics 视图将打开。

3. 从 Queries 列表中，选择以下查询功能：
 - Paths Used -- 已使用的路径。表明哪些路径在 Capture Session 期间被解析或添加到动态序列列表中，路径被解析的次数，以及致使音频播放的次数。
 - Paths Added -- 已添加的路径。表明在 Capture Session 期间触发了哪些对白事件。对于触发的各对白事件，会显示添加到动态序列列表中的相应音频对象，以及该对象被添加的次数。
 - Dialogue Events Distribution -- 对白事件分布。显示 Capture Session 期间触发的对白事件相关的统计信息，特别是各对白事件被解析的次数，以及对白事件实际致使音频对象播放的次数。
4. 点击 Run（运行）。

查询的结果会显示在 Results（结果）中。

5. 用筛选器来排除不需要的信息。这样可以缩小搜索范围，并快速确定是否应采取必要对策。

相关主题

- [启动/停止捕获过程](#)
- [筛选 Capture Log](#)
- [使用 Capture Log 快捷菜单](#)
- [保存 Capture Log](#)
- [从已有的远程 Capture Session 加载数据](#)

保存 Capture Log

每次启动捕获时 Wwise 都会清空 Capture Log，因此如果希望保留 Capture Log 中的信息，就需要将其保存到文件中。可以将其另存为 PROF 文件（默认）或文本文件，您

可以点击 [Wwise 工具栏](#) 中的 Connect to Remote Platform 按钮，然后点击 Connect To File... 将 PROF 文件加载到 Wwise Profiler 布局中。即可像正在对游戏进行性能分析一样，显示所有信息。如果选择文本文件，Wwise 将仅保存 Capture Log 中的信息，当前各列和排序也将保留。如果想进一步排序或筛选信息，可将文件导入外部电子表格程序。

保存 Capture Log 的方法如下：

1. 在 Capture Log 中，点击 Save Log（保存日志）。
2. Save As（另存为）窗口将打开。
3. 浏览至希望保存文件的位置。
4. 在 File name 中输入文件名称。选择 PROF 或 TXT 扩展名。
5. 点击 Save（保存）。

这时将把 Capture Log 的内容保存为文件。

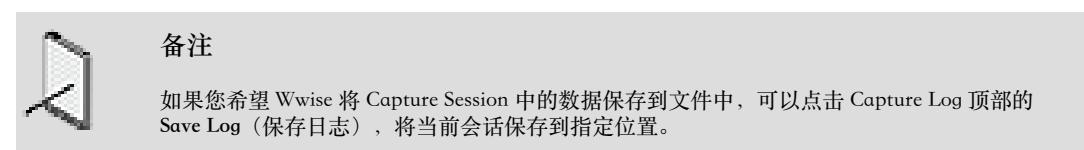
相关主题

- [启动/停止捕获过程](#)
- [筛选 Capture Log](#)
- [使用 Capture Log 快捷菜单](#)
- [从 Capture Session 获取统计信息](#)
- [从已有的远程 Capture Session 加载数据](#)
- [使用 Performance Monitor 进行监控和故障排查](#)

从已有的远程 Capture Session 加载数据

捕获游戏中的信息时，Wwise 会将其保存至名为 ProfilingSession.prof 的性能分析会话文件，稍后可以将其加载至 Wwise 中，并进一步分析游戏的音频。

每个新的 Profiling Session 都会创建新的文件，并在文件名称后附加递增的数字。保留的最大文件数可以在 Profiler Settings（Profiler 设置）中指定。



从已有的 Capture Session 加载数据的方法如下：

1. 切换至 Profiler 布局，可以执行以下操作之一：
 - 在菜单栏中，点击 Layouts > Profiler（视图 > Profiler）。
 - 按 F6 键。
2. 在工具栏中，单击 Connect to Remote Platform（连接至远程平台）按钮 。

Remote Connection 对话框将打开。

3. 点击 Connect To File (连接到文件)。
4. 选择一个 PROF 文件并点击 Open。

Capture Session 中收集的所有数据都将加载到 Profiler、Voice Profiler 和 Game Object Profiler 布局的各视图中。

从多个 Profiling Session 中提取统计信息的方法如下

1. 在 View 菜单中打开 Profiler Statistic 视图。
2. 点击工具栏上的 Connect to Remote Platform  按钮。

Remote Connection 对话框将打开。

3. 点击 Connect To File 按钮。
4. 选择所需的多个 PROF 文件并点击 Open。

将会处理所选 Capture Session 中收集的所有数据，最终的统计信息将显示在 Profiling Statistics 视图中。

相关主题

- 启动/停止捕获过程
- 筛选 Capture Log
- 使用 Capture Log 快捷菜单
- 从 Capture Session 获取统计信息
- 保存 Capture Log
- 使用 Performance Monitor 进行监控和故障排查

使用 Performance Monitor 进行监控和故障排查

Wwise 设计工具与音频引擎紧密集成，因此游戏过程中，您可以实时监控大量的关键表现指标。Game Profiler (性能分析器) 也可与 Soundcaster (声音选角器)、Game Simulator (游戏模拟器) 以及基于 SoundFrame (声音构架) 的其他应用程序一起使用，这样在您的设计和原型集成到游戏中之前，可以事先对其性能进行监控。

Wwise 捕获声音引擎活动的相关信息时，Performance Monitor (性能监控器) 中会实时显示性能图表，对应的实际数量和百分比也会显示在右侧的 Performance Data (性能数据) 列表中。

这些曲线可以让您快速找到游戏中音频性能超过限制的地方，例如平台限制。组合使用 Performance Monitor、Advanced Profiler 和 Capture Log，可以帮您诊断和解决潜在的问题。

Performance Monitor 允许自定义，即可以设置其中显示哪些性能指标和数值。有关自定义 Performance Monitor 的详细信息，请参阅[自定义 Performance Monitor](#)。

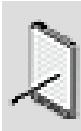


技巧

您可以在捕获数据时监控性能，或者启动 Capture Session，在图表中移动光标来调查问题。如果想在捕获到信息的同时监控性能，必须选择 Follow Capture Time 选项，来让 Capture Log 和 Performance Monitor 随捕获时间更新。

监控性能的方法如下：

1. 连接到远程运行的控制台、PC、SoundFrame 应用程序或 Game Simulator。



备注

如果正在使用 Soundcaster 对模块或原型进行性能分析，将在您播放不同模块时捕获信息。

2. 执行以下操作之一：

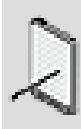
- 在菜单栏中点击 Layouts > Profiler（布局 > 性能分析）。
- 按 F6 键。

Profiler 布局将显示。

- 3.

点击工具栏中的 Start Capture (开始捕获) 按钮。

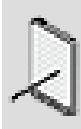
Wwise 将显示声音引擎捕获信息相关的平台性能曲线。



备注

实心方块表明此处捕获的信息已超过限制，曲线顶部的长方块表明当前值高出最大允许值，底部的方块表明当前值低于最小允许值。

4. 将 Performance Monitor 的时间光标（白色）拖至曲线上提示性能问题的位置。



备注

在曲线视图中移动光标时，Capture Log 的位置以及 Performance Data 面板和 Advanced Profiler 中显示的数据会自动更新。

5. 通过 Advanced Profiler 中的时间光标（白色圆圈）找出导致问题的事件、动作或其它声音引擎活动，或按住 Shift 并点击 Capture Log 中的条目，来将所有性能分析视图与此时间戳同步。
6. 要编辑对象和事件的内容或属性，请双击 Capture Log 中的条目。相应事件或对象将在 Event Editor（事件编辑器）或 Property Editor（属性编辑器）中打开，在其中进行所需修改即可。
7. 查看 Advanced Profiler 各选项卡，可以获取该引擎活动条目的详细性能信息。
8. 要将 Performance Monitor 中的曲线数值保存为文本文件，请右键点击曲线并选择 Save All Counters to File（将所有数值保存到文件）。数值将保存为制表符分割的文本文件，其中包含从捕获开始直至当前的所有数值，记录间隔为 200 ms。

相关主题

- [自定义 Performance Monitor](#)

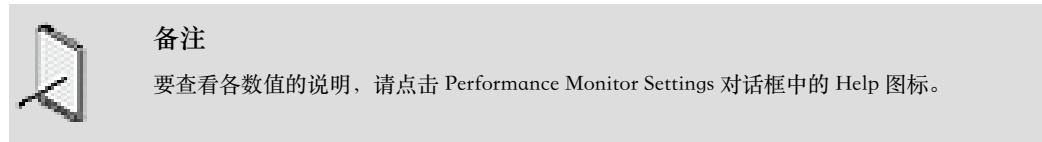
- 启动/停止捕获过程
- 保存 Capture Log

自定义 Performance Monitor

有些时候，您可能需要同时监控多个关键指标，而另一些时候则可能只想关注一个特定方面。为了满足不同监控会话的需求，可以增加和删除 Performance Monitor 中的数值显示。

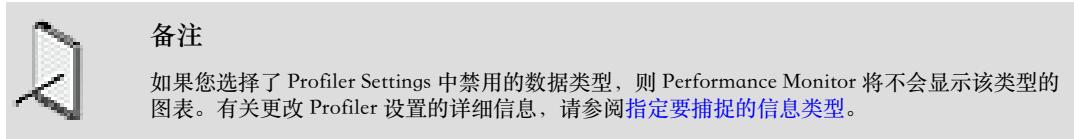
修改 Performance Monitor 显示内容的方法如下：

1. 在 Performance Monitor 标题栏中，点击 View Settings 图标。
Performance Monitor Settings（性能监控设置）对话框将打开。
2. 在 Show In Graph（显示曲线）列中，选择希望显示在 Performance Monitor 坐标图中的数值。



3. 在 Show In List（在列表中显示）列中，选择希望显示在 Performance Data 列表中的数值。
4. 为您选择的数值设置最小和最大值，即数值所对应曲线图的纵坐标范围。
5. 点击 OK。

Performance Monitor 将会更新，仅显示您选择的数值。



使用 Voice Inspector 分析声部

为什么这个声音的音量那么大？为什么我听不到这个声音？声音设计师经常会问这样的问题。经过各个层级结构的声音、衰减、Event（事件）、RTPC 和总线的影响，可能很难查明 Wwise 中音量输出电平异常问题的原因。利用 Voice Inspector，用户可以快速查出所有声部的音量驱动因素。

设置 Voice Inspector

1. 转到 Voice Profiler（声部性能分析器）布局 (F11)，或直接打开 Voice Inspector（声部检视器）视图 (Ctrl+Shift+.)。
2. 在 Voice Inspector 中，单击右上角的“设置”图标或按下 Alt+G，打开 Profiler Settings（性能分析器设置）视图。
3. 启用 Voices Data（声部数据）和 Voice Inspector Data（声部检视器数据）。

运行 Voice Inspector

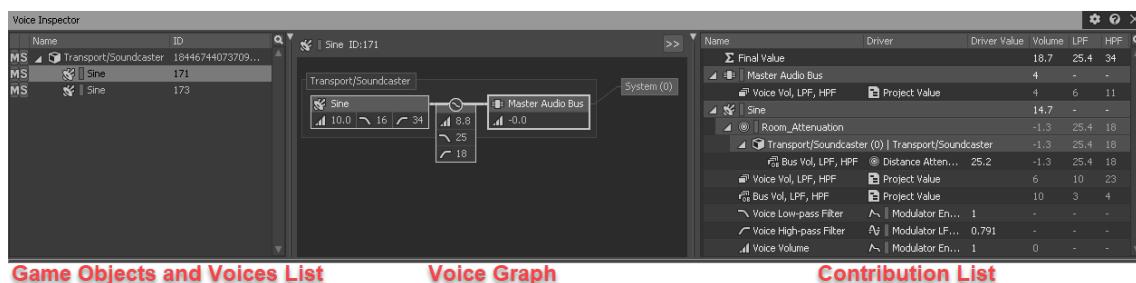
您可以在连接游戏的情况下或直接在 Wwise 设计工具中使用 Voice Inspector。

下面来简单了解一下如何实际运用 Voice Inspector：

1. 在 Project Explorer (工程资源管理器) 中, 使用源文件来添加 Sound SFX (音效)。我们已将其命名为 Sine。请将其设为无限循环。
2. 在 Property Editor (属性编辑器) 的 RTPC 选项卡中, 添加一个或多个 RTPC 曲线 (至少关联至 Volume、LPF 或 HPF 之一)。
3. 在 Property Editor 的 Positioning (定位) 选项卡中, 添加 Attenuation ShareSet (衰减共享集)。您也可以选择在 Attenuation Editor (衰减编辑器) 中启用 Cone Attenuation (声锥衰减)。
4. 在 Wwise 工具栏中, 单击 Start Capture (开始捕获)。(Alt+C)
5. 在 Transport Control (走带控制) 中, 播放声音。(空格键)
6. 调节各项设置, 比如:
 - 在 Attenuation Editor 中, 移动 Distance Attenuation (距离衰减) 和 Cone Attenuation。
 - 在 Master Audio Bus (主音频总线) 或声音的 General Settings (常规设置) 选项卡中, 更改 Volume (音量)、Pitch (音高)、Low-pass Filter (低通滤波器) 或 High-pass Filter (高通滤波器) 的基准值。
7. 停止捕获。(Alt+C)

此时, Voice Inspector 将会记下针对此声部对 Volume/LPF/HPF 所做的全部修改。若要查看这些修改, 需打开 Voice Inspector, 并在 Voice Monitor (声部监控器) 中移动时间光标:

1. Ensure you have the Voice Inspector and Voice Monitor views open. (By default, the Voice Profiler layout includes both.)
2. 在 Voice Monitor 中往回移动时间光标, 直到在 Voice Inspector 的左侧面板 (Game Objects 和 Voices List) 中看到所述循环声音。Select the sound ("Sine" in the example below) from the list.
3. 在 Game Objects (游戏对象) 和 Voices List (声部列表) 中选择声音后, 视图的中间面板 (Voice Graph) 中会显示整条声部信号流。若在 Voice Graph (声部图) 中选择节点, 则会显示具体的声部路径, 同时相应地更新右侧面板 (Contribution List)。
4. 在 Voice Monitor 中前后移动时间光标, 将显示在捕获期间对 Volume/LPF/HPF 所做的修改。



此时将会在 Voice Inspector 中显示各项信息。如需详细了解其用法, 请参阅 [使用 Voice Inspector](#)。

使用 Voice Inspector

在 Game Objects（游戏对象）和 Voices List（声部列表）中选择声音时，将在 Voice Graph（声部图）中自动选择**干声路径**，同时在 Contribution List（影响列表）中显示选定干声路径的整个层级结构。Contribution List 中将显示针对 Volume/LPF/HPF 所做的全部修改，包括 RTPC、State（状态）、Distance Attenuation（距离衰减）、Cone Attenuation（声锥衰减）和 Event Action（事件动作）。这些参数会实时更新并与 Performance/Voice Monitor（性能/声部监控器）光标保持同步，方便用户通过移动时间光标来对比这些值随时间的变化。



技巧

Voice Monitor 和 Performance Monitor 中，时间线是同步移动的。

Game Objects 和 Voices List

左侧的 Game Objects 和 Voices List 包含所有当时正在播放声音（无论虚实）的 Game Object（时间由 Voice/Performance Monitor 的时间光标指定）。在对应 Game Objects 下，可能会显示一个或多个活跃声音。在同一 Game Object 当前播放该声音的多个实例时，声音名称会在 Game Object 下出现多次。不过，每个实例都会有不同的 ID，其 Volume/LPF/HPF 也会单独进行分析。

	Name	ID
M S	Transport/Soundcaster	18446744073709551614
M S	Sine	165

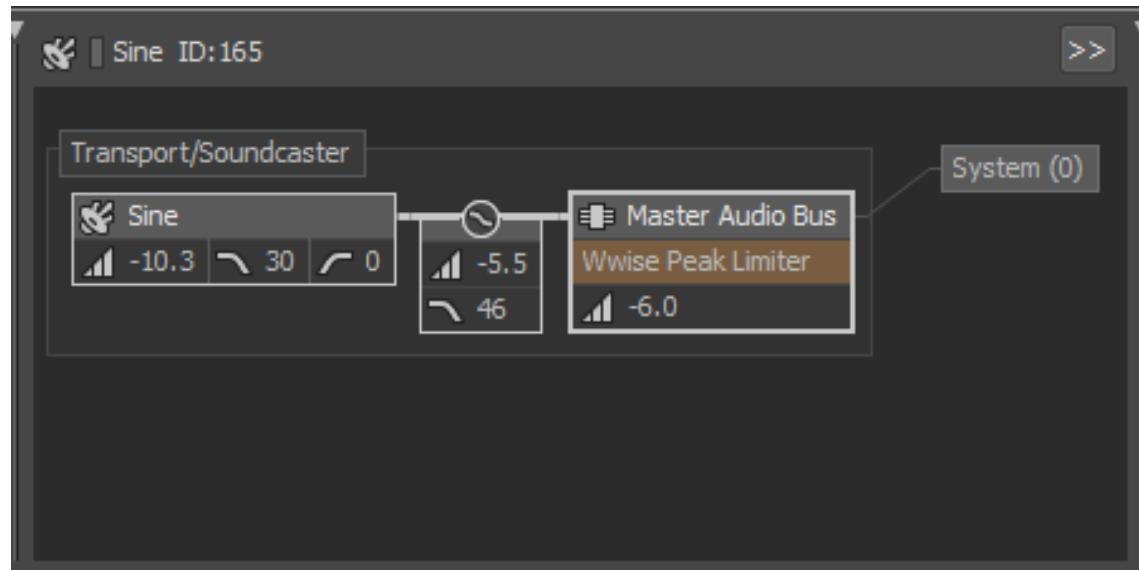
Game Objects and Voices List

在 Game Objects 和 Voices List 中选择声音时，将自动更新 Voice Graph 和 Contribution List。默认选择干声路径。每次仅可显示一个声音。

在双击声音时，将打开 Property Editor（属性编辑器）。在右键单击声音时，将显示**标准快捷菜单**。

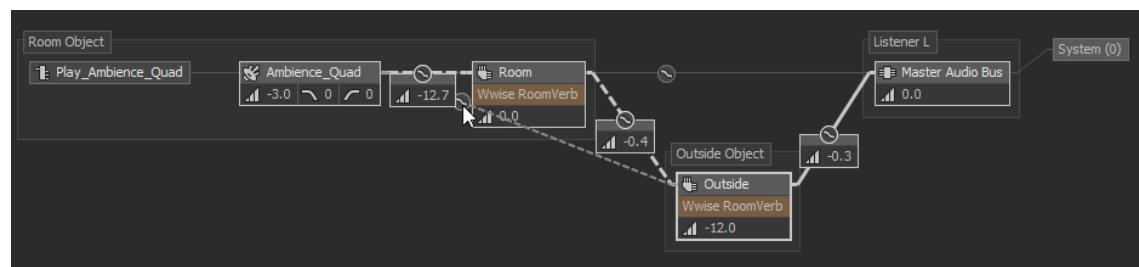
Voice Graph

视图中间的 Voice Graph 会显示从音频源到输出的所有声部信号路径。因为各个信号路径的 Volume/LPF/HPF 属性可能具有不同的值，所以同一时间仅会有一个活跃路径。对于活跃路径，声部图节点之间的连线会高亮显示为白色。右侧的 Contribution List 中会显示属性影响及选定路径。对于不属于选定路径的属性，将不予显示。



Voice Graph

若要显示不同路径的属性值，请直接单击声部图节点或所需路径对应的连线。如需检视具有多个发送的路径（常见于 Spatial Audio 场景），可同时选择多个节点（Ctrl + 单击）。

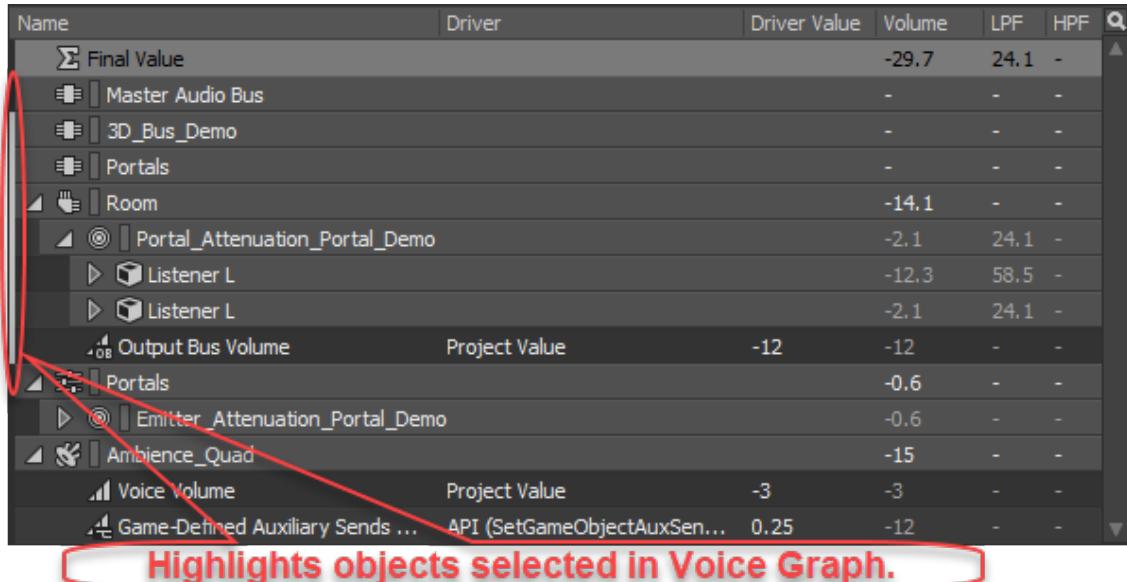


在此，可使用双击和右键单击等标准 Voice Graph 操作。



技巧

在 Voice Graph 中选择对象（如 Master Audio Bus）时，Contribution List 中会高亮显示该对象及其子对象（左侧显示白色边线）。

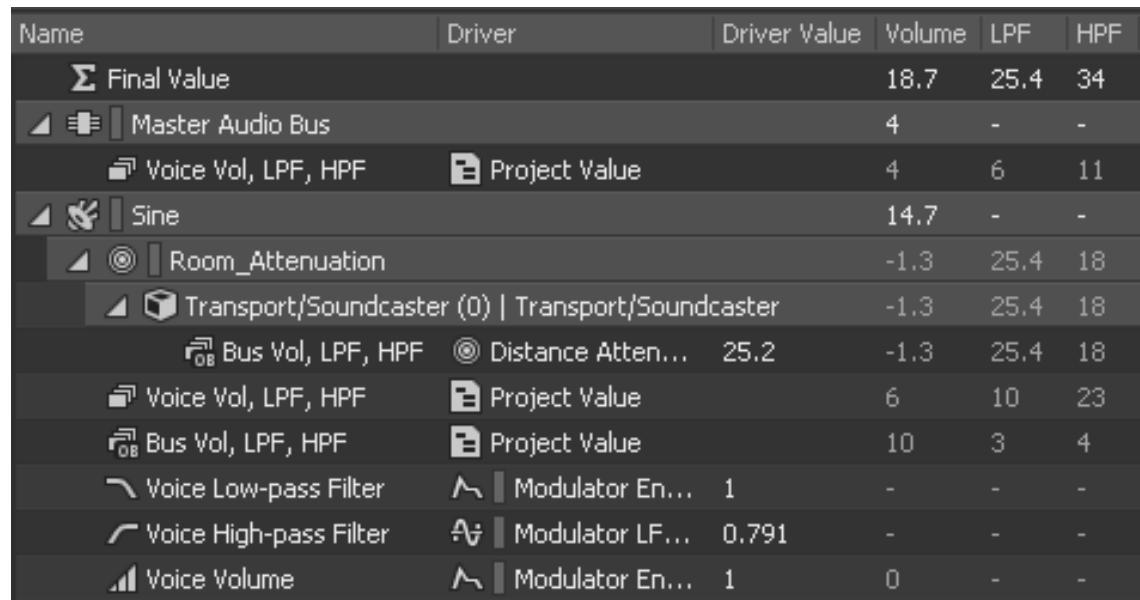


Name	Driver	Driver Value	Volume	LPF	HPF
Σ Final Value		-29.7	24.1	-	-
Master Audio Bus		-	-	-	-
3D_Bus_Demo		-	-	-	-
Portals		-	-	-	-
Room		-14.1	-	-	-
Portal_Attenuation_Portal_Demo		-2.1	24.1	-	-
Listener L		-12.3	58.5	-	-
Listener L		-2.1	24.1	-	-
Output Bus Volume	Project Value	-12	-12	-	-
Portals		-0.6	-	-	-
Emitter_Attenuation_Portal_Demo		-0.6	-	-	-
Ambience_Quad		-15	-	-	-
Voice Volume	Project Value	-3	-3	-	-
Game-Defined Auxiliary Sends ...	API (SetGameObjectAuxSen...	0.25	-12	-	-

Highlights objects selected in Voice Graph.

Contribution List

对于选定声音的选定声部路径，Contribution List 中会显示 Actor-Mixer 和 Interactive Music 层级结构中修改了 Volume/LPF/HPF 的所有父对象，包括在对选定声音进行混音的过程中涉及的所有 Audio Bus（音频总线）和 Auxiliary Bus（辅助总线）。



Name	Driver	Driver Value	Volume	LPF	HPF
Σ Final Value		18.7	25.4	34	
Master Audio Bus		4	-	-	-
Voice Vol, LPF, HPF	Project Value	4	6	11	
Sine		14.7	-	-	-
Room_Attenuation		-1.3	25.4	18	
Transport/Soundcaster (0) Transport/Soundcaster		-1.3	25.4	18	
Bus Vol, LPF, HPF	Distance Atten...	25.2	-1.3	25.4	18
Voice Vol, LPF, HPF	Project Value	6	10	23	
Bus Vol, LPF, HPF	Project Value	10	3	4	
Voice Low-pass Filter	Modulator En...	1	-	-	-
Voice High-pass Filter	Modulator LF...	0.791	-	-	-
Voice Volume	Modulator En...	1	0	-	-

Contribution List

Contribution List 会在相应层级结构的每个对象下显示 Volume/LPF/HPF 的实际值。Driver（驱动因素）和 Driver Value（驱动因素值）列会详细显示哪些因素导致了属性变化。

对于所涉及的 Master-Mixer、Actor-Mixer 和 Interactive Music 层级结构，相应的各行中会简要显示直接针对选定路径节点的属性修改。顶部的 Final Value（最终值）行会显示所有修改的汇总结果。若要转到属性页面以便查看属性变化，请双击属性名称。

对于未经修改的值，将不予显示。不过，对于曾经修改过的值，倘若恢复为中间值(0 dB)，则将持续显示。

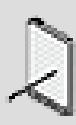
相关主题

- [有关特定属性显示方式的说明](#)

有关特定属性显示方式的说明

为了充分了解 Voice Inspector（声部检视器）中显示的值，请注意以下事项。

- 总的来说，假如某一 Driver（驱动因素）同时在影响 Volume（音量）、LPF 和 HPF，只会显示一行。
- Project Value/SoundBank Value（工程值/音频包值）：一般通过滑杆（如声音的主音量滑杆）修改的值会写入 SoundBank（音频包）并由游戏加载。在将性能分析器连接到游戏时，这些值会显示为 SoundBank Value。若仅使用性能分析器而不连接游戏（直接开始捕获），则这些基础声音属性将显示为 Project Value。在连接游戏时，任何在捕获之后对这些值所做的调节都会被视为 Live Edit。
- Live Edit（实时编辑）：假如在连接游戏时更改属性值，将在 Contribution List（影响列表）中显示为 Live Edit 条目，对应于所应用的偏置。因此，若原始音量为 -10 dB 且滑杆被移到 -4，则 Live Edit 将显示 +6，而 SoundBank Value 保持不变。不过，两者的总和会反映实际音量 (-4 dB)。此规则同样适用于未连接游戏时修改的值。在连接游戏时，Wwise 会将新的修改推送给游戏，同时显示为 Live Edit。这样便于识别需要重新生成的过时 SoundBank。



备注

The Voice Inspector reports Live Edits differently depending on when you connect to a remote platform, when SoundBanks load, and whether you reconnect or recapture.

- Values which are edited before connecting to a remote platform, where the edits are not included in generated SoundBanks, will be marked as Live Edit if they are loaded before their corresponding SoundBank loads. 同时，SoundBank Value 将被标记为 0，意味着该值被直接忽略。
- Live Edit values from the last capture may be marked as SoundBank Value when restarting the capture or reconnecting to the remote platform.

- Event Actions（事件动作）：在同时有多个活跃的 Action 时，Set/Reset Volume/LPF/HPF（设置/重置音量/LPF/HPF）会合并显示一个当前有效值。比方说，假如针对爆炸声发送了一个 Play Event，并利用第一个 Set Volume Action 将其设为 Absolute 30 dB，然后利用第二个带有 10 秒延迟的 Set Volume Action 将其设为 Relative -20 dB。那么，在触发 Event 大概 10 秒之后，爆炸声对象节点下的 Voice Vol 将会显示为 10。如需进一步查看 Set Volume 动作，请转到 Capture Log（捕获日志）。
- Attenuation（衰减）：Attenuation 的 Volume、LPF 和 HPF 值，与其发声体-听者对（也叫射线）的值对应。在存在多个发声体或多个位置时，Attenuation 值将会显示为所有 Attenuation 发声体-听者对中最大的 Volume 值和最小的滤波器值。在展开 Attenuation 对象节点及对应射线后，将详细显示射线的各项属性值，包括所有适用的 Distance Attenuation、Cone Attenuation、Obstruction 和 Occlusion 值。

- **Source**（源）：此处不会显示 Source Editor（源编辑器）中设置的 Fade In（淡入）和 Fade Out（淡出），因为其会在转码时写入音频文件。
- **Fades**（淡变）：Event Action（Play、Stop、Mute、Unmute、Pause 和 Resume）产生的淡变只有在其不断改变音量时才显示。一旦淡变完成，Contribution List 中将不再显示淡变的行。各个淡变源均会单独显示。
- **RTPC**：在 RTPC 导致属性发生变化时，会将 Game Parameter（游戏参数）显示为 Driver，并在 Driver Value（驱动因素值）列中列出当前的参数值。一旦 RTPC 曲线开始变换数值，Volume、LPF 或 HPF 列中便会显示相应的变化。
- **States**（状态）：在 State 导致属性发生变化时，将在 Driver 列中列出 State Group（状态组），并在 Driver Value 列中显示当前 State。Volume、LPF 或 HPF 列中会显示属性的相应变化。
- **HDR Window**（HDR 窗口）：因 HDR 窗口变化导致的声音衰减将会显示为 Voice Volume（HDR）条目，同时会将 HDR 总线显示为 Driver，并在 Driver Value 列中列出 HDR Window 音量。Volume 列中将会报告实际的音量变化（单位为 dB）。

使用 Game Object Explorer 跟踪对象和听者

Game Object Explorer 供您研究游戏对象，包括它们作为发声体、听者的作用，亦或两者兼具。在该视图中，您可以看到游戏中注册的游戏对象，并决定哪些对象将被 Wwise 监视。在选择需要 Watch（监视）的游戏对象后，它们将显示在 Game Sync Monitor 中，也会显示在 Game Object 3D Viewer、RTPC 视图和 Attenuation Editor 中。

使用 Game Object Explorer 跟踪游戏对象涉及：

- [理解游戏对象列表](#)
- [通过 Watch 监视对象和听者](#)

理解游戏对象列表

Game Object Explorer 中的 Game Object（游戏对象）选项卡里，包含此会话中所有活跃的游戏对象，或曾经活跃的游戏对象列表。游戏对象由音频程序员进行创建和销毁，并在创建时为其提供名称和 ID。

连接到游戏之后，此列表将自动更新。

Game Object Explorer

Name	ID	Last Registered	Last Unregistered	Is Alive
AKPhysicsTriggerWaterImpact...	19136937	0:10:40.725	-	✓
AMBIENCE	444443	0:10:40.725	-	✓
Boy	196665	0:10:40.725	-	✓
CollisionRect2D	44761606	0:10:40.725	-	✓
CollisionRect2D	786709	0:10:40.725	-	✓
door	26411081	0:10:40.725	-	✓
Listener (Default)	10002	0:10:40.725	-	✓
pelvis	196646	0:10:40.725	-	✓

处于激活状态的游戏对象会标记为“Alive”，游戏对象注册时会变成激活状态，而注销时激活状态将停止。例如，在第一人称射击游戏中，如果游戏对象被销毁或超出玩家可听到的区域范围，该对象即被视为不再处于激活状态。而在即时战略游戏中，地图上的所有游戏对象在销毁之前，可能将一直保持激活状态。



技巧

点亮筛选按键“Is Alive”，其颜色将由灰变蓝，视图中将不再显示当前未使用的游戏对象。

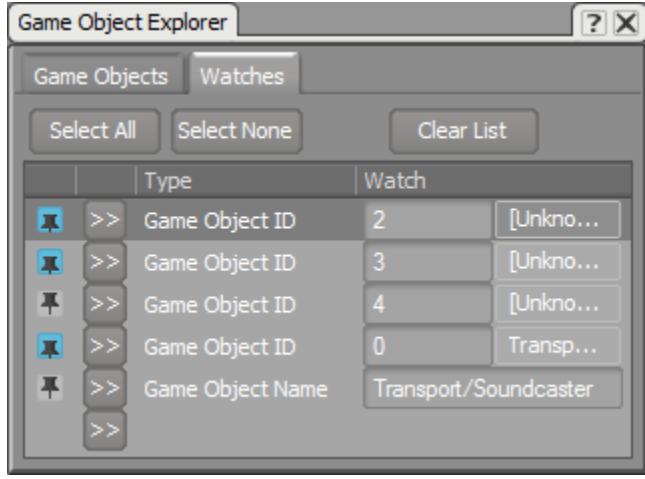
Game Objects 选项卡中的大多列中（与其他表格一样）都仅显示信息。但是，Mute 和 Solo 列有相应的按钮，可以帮助设计师单独监听或静音所选游戏对象的声音及相应 Wwise 对象。

相关主题

- [通过 Watch 监视对象和听者](#)
- [为游戏对象指派 Watch](#)
- [编辑和删除 Watch](#)

通过 Watch 监视对象和听者

监视（Watch）允许您独立监控游戏中的各游戏对象。将一个 Watch 指派给游戏对象时，该 Watch 会定期报告其详细状态。通过 Watch 得到的信息可供 Game Object 3D Viewer、RTPC 图表、Attenuation Editor 和 Game Sync Monitor 使用。如果没有监视游戏对象，将无法在游戏对象性能分析视图中看到任何数据。



游戏对象监视可以基于名称或者 ID，游戏中可能有多个游戏对象名称相同，但对象 ID 不会重名。因此，如果您希望创建一个 Watch 来监视多个相同对象（例如无人机），请使用基于名称的监视。而如果仅需监视一个特定对象，应创建基于 ID 的 Watch。

使用监视可能涉及以下任务：

- [为游戏对象指派 Watch](#)
- [编辑和删除 Watch](#)
- [选择要显示的 Watch](#)

为游戏对象指派 Watch

有两种方法为游戏对象（Game Object）指派 Watch（监视）：

- 通过游戏对象列表。
- 通过监视列表。

通过游戏对象列表指派游戏对象 Watch 的方法如下：

1. 在 Game Object Explorer 的 Game Objects 选项卡中，右键点击游戏对象并选择 Add Game Object to Game Object Watch List（添加游戏对象到监视列表）（By ID 或 By Name）。

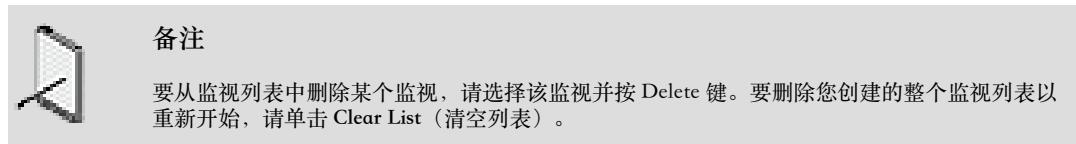
基于该游戏对象 ID 或名称的 Watch 将添加到监视列表中。

通过监视列表指派游戏对象监视的方法如下：

1. 在 Game Object Explorer 中，切换至 Watch（监视）选项卡。
2. 点击选择器按钮（>>）并选择其一：
 - Game Object Name（游戏对象名称），创建基于对象名称的游戏对象监视。
 - Game Object ID（游戏对象 ID），创建基于对象 ID 的游戏对象监视。
 - Global Game Object（全局游戏对象），创建针对全局游戏对象的监视。（如果要监视针对全局或总线所使用的值，将很有用）。

3. 在 Watch 列中，输入该监视基于哪个名称或 ID，并按 Enter 键。

该监视将被添加到监视列表中。



相关主题

- [通过 Watch 监视对象和听者](#)
- [编辑和删除 Watch](#)
- [选择要显示的 Watch](#)

编辑和删除 Watch

您可以轻松编辑或删除已创建的监视 (Watch)，注意，这不会影响原始游戏对象，而仅会影响收集其数据所用的监视。

编辑监视的方法如下：

1. 在 Game Object Explorer 中，切换至 Watch (监视) 选项卡。
2. 要更改监视类型，请点击选择器按钮 (>>) 并选择一个不同类型的监视。
3. 要更改监视名称，请点击 Watch 列并输入一个不同的名称或 ID。

删除监视的方法如下：

1. 在 Game Object Explorer 中，切换至 Watch (监视) 选项卡。
2. 选择监视并按 Delete 键。

该监视将从监视列表中删除。



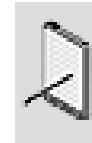
相关主题

- [通过 Watch 监视对象和听者](#)
- [为游戏对象指派 Watch](#)
- [选择要显示的 Watch](#)

选择要显示的 Watch

默认情况下，所有监视 (Watch) 都是固定显示的，即与其相关的游戏对象将自动显示在 Game Object 3D Viewer、RTPC 和 Attenuation Editor 图表中。但您可以为一些

监视取消固定，避免视图显得混乱。显示的对象越少，占用的 CPU 资源越少，从而改善 Game Object 3D Viewer 的性能。



备注

这不会影响在 Game Sync Monitor 中显示的游戏对象。Watch List 中的所有游戏对象都会显示在 Game Sync Monitor 中。

在监视列表中将监视固定或取消固定的方法如下：

1. 在 Game Object Explorer 的 Watch 选项卡中，点击监视名称左侧的固定图标。

在监视列表中固定所有监视的方法如下：

1. 在 Game Object Explorer 的 Watch 选项卡中，点击 Select All（选择所有）。

在监视列表中取消固定所有监视的方法如下：

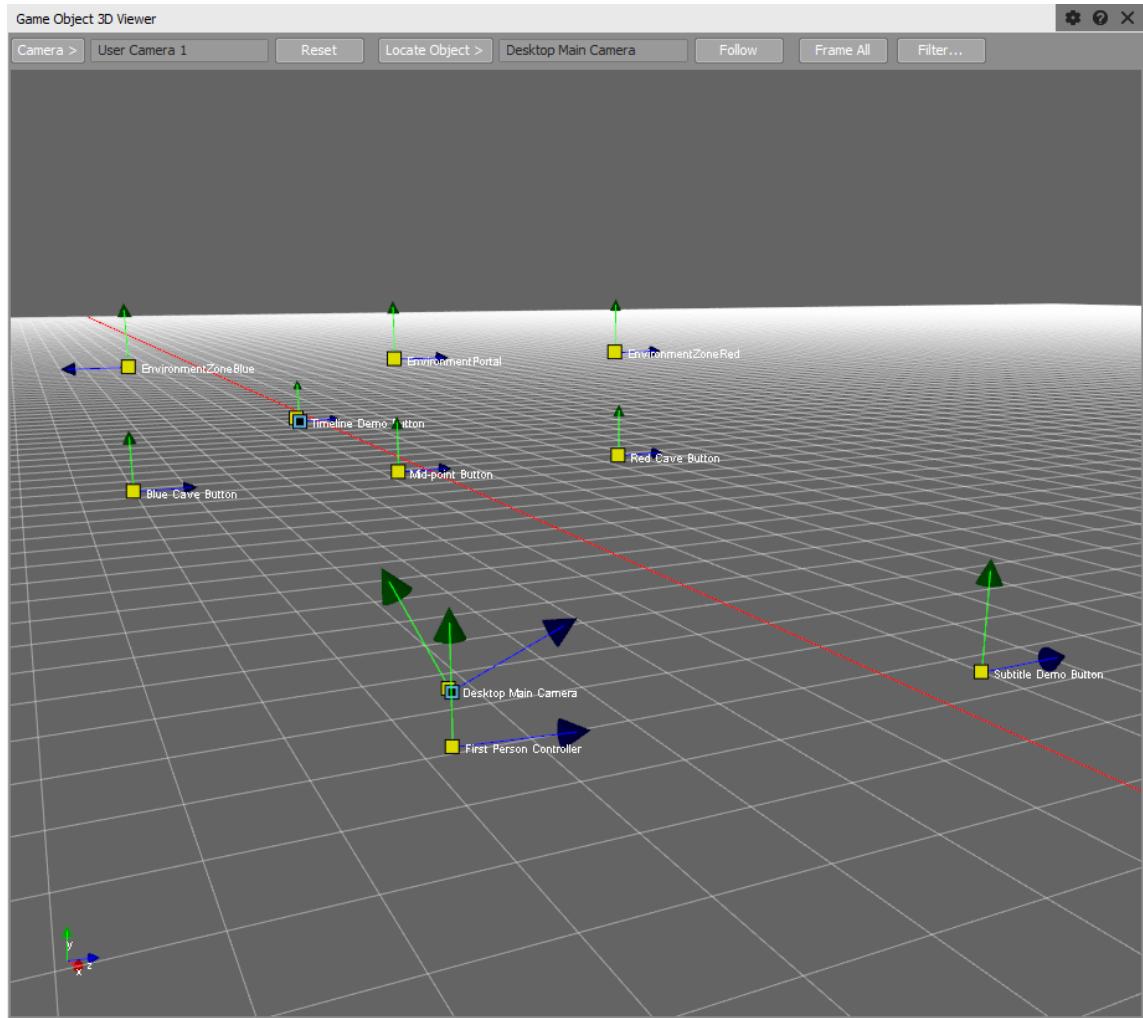
1. 在 Game Object Explorer 的 Watch 选项卡中，点击 Select None（全部不选）。

相关主题

- [通过 Watch 监视对象和听者](#)
- [编辑和删除 Watch](#)

通过 Game Object 3D Viewer 检查对象

Game Object 3D Viewer（3D游戏对象查看器）从游戏对象 Watch 中获取数据并以图形形式显示，在其中，您可以实时查看这些对象作为发声体/听者时是如何互动和相对移动的。



Game Object 3D Viewer 中将显示所监视的各个游戏对象并附以相应图标。您可以根据 Viewer Settings (视图设置) 中所选选项来自定义视图。比如，可同时查看应用于游戏对象的声锥或半径。



技巧

可以使用键盘快捷键访问 Game Object 3D Viewer 中的许多控件，快捷方式的完整列表请参阅[附录 C](#)。# # #。

在 Game Object 3D Viewer 中启用 Spatial Audio

若要在 Game Object 3D Viewer 中显示 Spatial Audio 视觉表示，则需先对其启用性能分析。

Spatial Audio 性能分析数据包括 Geometry (几何构造) 和 Portal (门户) 的视觉表示，也包括 Emitter (发声体)、Listener (听者) 和 Room (空间) 的新图标以及由 Diffraction (衍射)、Sound Propagation (声音传播) 和 Reflect 插件所创建的声音路径和虚拟对象。

指定所要分析的数据的方法如下：

1. 在 Advanced Profiler (高级性能分析器) 中，单击 Settings (设置) 图标。

Profiler Setting 对话框将打开。

2. 选择 Spatial Audio。
3. 点击 Close。

这时 Game Object 3D Viewer 中将显示相应数据。

相关主题

- [选择显示在 Game Object 3D Viewer 中的数据](#)

选择显示在 Game Object 3D Viewer 中的数据

通过更改 Game Object 3D Viewer 设置，可以按需选择要显示的数据。当然，您选择的数据越多，视图越复杂。

指定要显示在 Game Object 3D Viewer 中的数据的方法如下：

1. 在 Game Object 3D Viewer 中，点击 Setting (设置) 图标。

Game Object 3D Viewer Setting (Game Object 3D Viewer 设置) 对话框将打开。

2. 选择希望显示在 Game Object 3D Viewer 中的选项。



3. 点击 Close。

数据将根据所选条件显示在 Game Object 3D Viewer 中。

相关主题

- [操作 Camera](#)
- [筛选 Watch](#)

操作 Camera

您可以通过鼠标控制和移动 Camera (摄影机)，利用三个空间维度来探索 3D 查看器，这样，被监视的游戏对象作为发声体/听者进行互动时，您就可以进行观察了。

将游戏对象置于 Camera 镜头中央的方法如下：

1. 在 Game Object 3D Viewer 中，点击 Locate Object (查找对象)，并从列表中选择要放在 Camera 镜头中央的对象。

所选对象将显示在视野中心。

将摄影机与游戏对象绑定的方法如下：

1. 从 Game Object 3D Viewer 的 Locate Object 列表中，选择一个游戏对象。
2. 点击 Follow（跟随）。

Capture（捕捉）过程中，如果对象移动，摄影机将跟随对象。

让 Game Object 3D Viewer 显示所有游戏对象的方法如下：

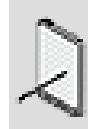
1. 在 Game Object 3D Viewer 中，点击 Frame All（显示全部）。

查看器将自动进行缩放调整，并显示所有所选游戏对象。

从游戏对象视角来观察游戏的方法如下：

1. 从 Game Object 3D Viewer 的 Camera 列表中选择一个已监视的游戏对象作为摄像机原点。

所选对象将成为摄影机视野范围的原点，且摄影机与对象的朝向相同。



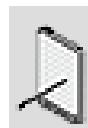
备注

如果您选择 Follow 选项或使用 WASD/箭头键来更改摄影机位置，则摄影机的朝向和原点将不再与对象一致。

平移摄影机镜头的方法如下：

1. 在 Game Object 3D Viewer 内按住 Ctrl 加鼠标左键，然后拖动。

摄影机镜头将朝上下左右方向平移。



备注

First Person 和游戏对象摄影机均不支持镜头平移。

缩放摄影机镜头的方法如下：

1. 在 Game Object 3D Viewer 内按住鼠标右键，然后拖动。

摄影机镜头将被推近和拉远。

旋转摄影机的方法如下：

1. 从 Game Object 3D Viewer 的 Camera 列表中，选择 User Camera 1（用户摄影机 1）或 User Camera 2（用户摄影机 2）。
2. 在 Game Object 3D Viewer 内按住鼠标左键，然后拖动。

摄影机将围绕中心点旋转。



技巧

可以在两个摄影机之间来回切换，而保持其各自的设置不变。

以第一人称射击游戏模式操纵摄影机的方法如下：

1. 在 Game Object 3D Viewer 的摄影机列表中选择 First Person 或任何游戏对象摄影机。

下面执行以下操作：

- 按 WASD 或箭头键来移动摄影机。
- 按住左键并拖动来调整摄影机镜头方向。
- 按住 Shift 来加快移动速度。
- 按住右键并拖动，或使用鼠标滚动轮，来推近和拉远镜头。

将摄影机重置于默认位置的方法如下：

1. 在 Game Object 3D Viewer 中，点击 Reset（重置）。

摄影机将被重置，并回到默认位置。

相关主题

- [选择显示在 Game Object 3D Viewer 中的数据](#)
- [筛选 Watch](#)

筛选 Watch

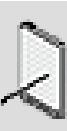
默认情况下，Game Object 3D Viewer 会显示正在监视的所有游戏对象。如果希望仅显示其中的一部分，可以筛选所显示的内容。

筛选 Game Object 3D Viewer 中显示内容的方法如下：

1. 在 Game Object 3D Viewer 中，点击 Filter。

Game Object 3D Viewer Filter 对话框将打开。

2. 要显示或隐藏某个监视（Watch），请选择或取消选择其固定图标。



备注

Game Object 3D Viewer 中将仅显示已固定的游戏对象监视。

3. 要显示所有监视，请点击 Pin All（固定全部）。
4. 要隐藏所有监视，请点击 Unpin All（取消固定全部）。

5. 点击 Close (关闭) 以返回 Game Object 3D Viewer。

相关主题

- [选择显示在 Game Object 3D Viewer 中的数据](#)
- [操作 Camera](#)

通过 Game Sync Monitor 评估游戏同步器

Game Sync Monitor (游戏同步器监视器) 性能分析工具可以在监视游戏对象时，跟踪其相关的 RTPC 数值。监视将接收这些数值并实时显示在曲线图中。

通过 Game Sync Monitor 监控游戏同步器涉及以下任务：

- [将 Game Sync 添加到监视器](#)
- [筛选监视结果](#)

将 Game Sync 添加到监视器

Game Sync Monitor 可以显示游戏中已监视对象相关的 RTPC 数据。须事先添加希望显示在此视图中的游戏同步器 (Game Sync)，才能显示数据曲线。

将 RTPC 添加到 Game Sync Monitor 中的方法如下：

1. 在 Game Sync Monitor 的 Game Syncs Watch (游戏同步器监视) 中，点击 Add。

Project Explorer - Browser 对话框将打开。

2. 选择所需 RTPC 并点击 OK。

RTPC 将被添加到 Game Sync Monitor 中。

相关主题

- [通过 Game Sync Monitor 评估游戏同步器](#)
- [筛选监视结果](#)

筛选监视结果

默认情况下，将 RTPC 添加到 Game Sync Monitor 中时，所有受其影响的被监视游戏对象都会显示在坐标图中，所以如果 RTPC 影响许多对象，可能会导致显示的内容非常拥挤。要简化此视图，您可以取消固定特定游戏对象/游戏同步器来将其隐藏。

显示或隐藏 Game Sync Monitor 中曲线的方法如下：

1. 在 Game Sync Monitor 中，点击游戏同步器旁边的固定图标来显示或隐藏其曲线。

相关主题

- [通过 Game Sync Monitor 评估游戏同步器](#)

- 将 Game Sync 添加到监视器

性能分析技巧和最佳实践

在 Wwise 中使用 Game Profiler 之前，您可能需要阅读以下章节，其中介绍了一系列可以帮您对游戏各方面进行性能分析和故障排查的技巧和最佳实践。

Game Profiler 性能

由于传输的数据量可能很大，所以从声音引擎捕获信息可能会影响 Wwise 的性能。因此您可能需要在 Profiler Settings（性能分析设置）中进行设置，限制 Advanced Profiler 生成的信息类型。通过排除一些信息类型，您可以：

- 节省网络传输带宽。
- 节省 Wwise 中的内存空间。
- 省略数据计算，从而节省游戏中的 CPU 时间。
- 省略数据处理或显示，从而节省 Wwise 中的 CPU 时间。

在捕获的同时监控性能

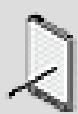
如果希望从声音引擎捕获数据并同时监控性能，须选择 Follow Capture Time 选项，否则一段时间之后，Profiler 中的视图将看似未被更新，如同没有捕获到信息一样。如果您忘了选择 Follow Capture Time，可以随时将 Performance Monitor 时间光标（白色）拖至图表上的其他位置，来滚动显示捕获的数据。拖动时间光标时，Profiler 中的其它视图将自动更新。

排查远程连接问题

- 将 Wwise 连接到游戏时，如果运行游戏的计算机 / 游戏机未显示在 Remote Connection 对话框中，请验证以下信息：
 - 游戏中，Communication（通信）模块是否正确初始化。详细信息请参阅 [Wwise SDK 文档](#) 的“Initializing Communications”（初始化通信）小节。
 - 确保游戏内初始化通信模块的 AkCommSettings 结构中，“Game Discovery Broadcast Port”（游戏发现广播端口）端口与 Wwise 工程设置中相同。详细信息请参阅 [指定网络端口](#) 和 [Wwise SDK 文档](#) 的“Initializing Communications”（初始化通信）小节。
 - 计算机 / 游戏机是否已经正确连接到网络。注意，有些游戏机会有多个网络适配器，其中一个用于调试，还有一个用于常规网络通信，请确保连接所有网络适配器，然后再用 Wwise 重新连接。
 - 运行游戏的计算机 / 游戏机与运行 Wwise 的计算机位于相同的子网段。
 - 是否有活跃的防火墙正在阻止连接。如果运行 Wwise 的计算机上有活跃的防火墙，请禁用并检验其是否正在阻止连接。如果确实是防火墙阻止了连接，请尝试将 Wwise 应用程序添加到防火墙的例外列表（请参阅防火墙的说明文档）。如果需要更改 Wwise 通信所用的端口（包括将非动态端口改为固定端口），请参阅 [指定网络端口](#) 和 [Wwise SDK 文档](#) 的“Initializing Communications”（初始化通信）小节。
 - 如果知道远程计算机 / 游戏机的 IP 地址，可以点击 Remote Connection 对话框中的 Connect to IP 并手动输入 IP 地址，如果这样可以连接到游戏，说明仅阻

止了自远程计算机 / 游戏机的广播。连接计算机/控制台后，会自动将其显示在 Remote Connection 对话框的 History 选项卡中，因此再次连接时将不必输入 IP 地址。

- 声音引擎是否在使用 Debug 或 Profile 版本。如果使用 Release 版本将无法连接。
- 如果您正使用很大的工程连接游戏并且 Wwise 停留在连接窗口，请尝试以下操作：
 - 打开工程，取消选择当前 Soundcaster Session、Mixing Desk Session 会话和加载到 Transport Control（播放控制）中的对象，并在未选中 **Synchronize modified object** 的情况下尝试连接游戏。



备注

此方法的一些背景信息：Wwise 连接到游戏时，会给游戏推送信息来确保显示在 Wwise 中的内容与游戏同步。但在未选中 **Synchronize modified object** 的情况下，Wwise 将不会推送全部内容，而仅推送 Property Editor、Transport Control 中选择的内容，以及活跃 Soundcaster Session 和 Mixing Desk Session 里的内容。如果 Wwise 工程中有包含大量数据的活跃 Soundcaster Session 或 Mixing Desk Session，连接游戏时就可能遇到此问题。

选择 Remote Connection 中的 **Synchronize modified object** 选项也可能导致此问题，因为 Wwise 会话内的所有更改都将同步至远程游戏的音频播放，这可能需要很长时间。因此，如果您遇到问题，请确保在取消选中此选项的情况下重新连接。

第 35 章 管理 SoundBank

概述	627
理解 SoundBank 如何加载到游戏中	629
构建 SoundBank	632
管理 SoundBank	647
定义 SoundBank 的自定义属性	649
为工程生成 SoundBank	656
使用 CopyStreamedFile 工具	659
SoundBank 的管理策略	660
SoundBank 技巧和经验总结	672

概述

为了高效地管理游戏的音频或振动组件，Wwise 将游戏的所有音频和振动都放到 SoundBank 中。SoundBank 其实就是一个包含音频或振动数据、媒体或两者兼有的文件。这些 SoundBank 在游戏的特定时刻加载到游戏所在平台的内存中。通过仅加载必要的内容，您可以优化每个平台上媒体文件占用的内存大小。SoundBank 是您所有工作的成果，这些成果包含的最终内容会成为游戏的一部分。

Wwise 中有两类 Bank：

- **Initialization (Init) bank** -- 初始化库。一种特殊库，其中包含有关工程的所有通用信息，包括有关总线层级结构、状态、切换开关和 RTPC 的信息。如果条件具备，则它还可能包括音频设备 ShareSet。

在每次 Wwise 生成 SoundBank 都会自动创建 Initialization Bank。Initialization Bank 通常在游戏开始时加载一次，以便在游戏期间能轻松地获取工程的所有通用信息。它必须是在启动游戏时最先加载的声音包；否则其它声音包（内容）可能会无法加载。Initialization Bank 的文件名为“Init.bnk”。

- **SoundBank** —— 此文件中同时包含事件数据、声音、音乐和振动结构数据或音频文件。与 Initialization Bank 不同，SoundBank 一般在游戏的不同时间加载和卸载，以提高平台内存的利用率。

由于所有平台各不相同，因此 Wwise 让您能轻松地针对各个平台定制 SoundBank，并同时为所有平台生成 SoundBank。Wwise 还为您提供故障排查工具，用来排查与 SoundBank 相关的任何问题，确保您遵守不同平台的限制。

每款游戏只能使用一个 Wwise 工程。如果您有多人同时处理一个大型工程，则可以将工程划分为多个独立的工作单元。有关使用工作单元的信息，请参阅[将工程分成 Work Units](#)。

Wwise 中的 SoundBank 视图

为了帮助您提高工作效率，Wwise 提供了 SoundBank 布局。此布局包含为您工程创建、管理和生成 SoundBank 所需要的所有视图，包括 SoundBank Manager、SoundBank Editor、Project Explorer 和 Event Viewer。

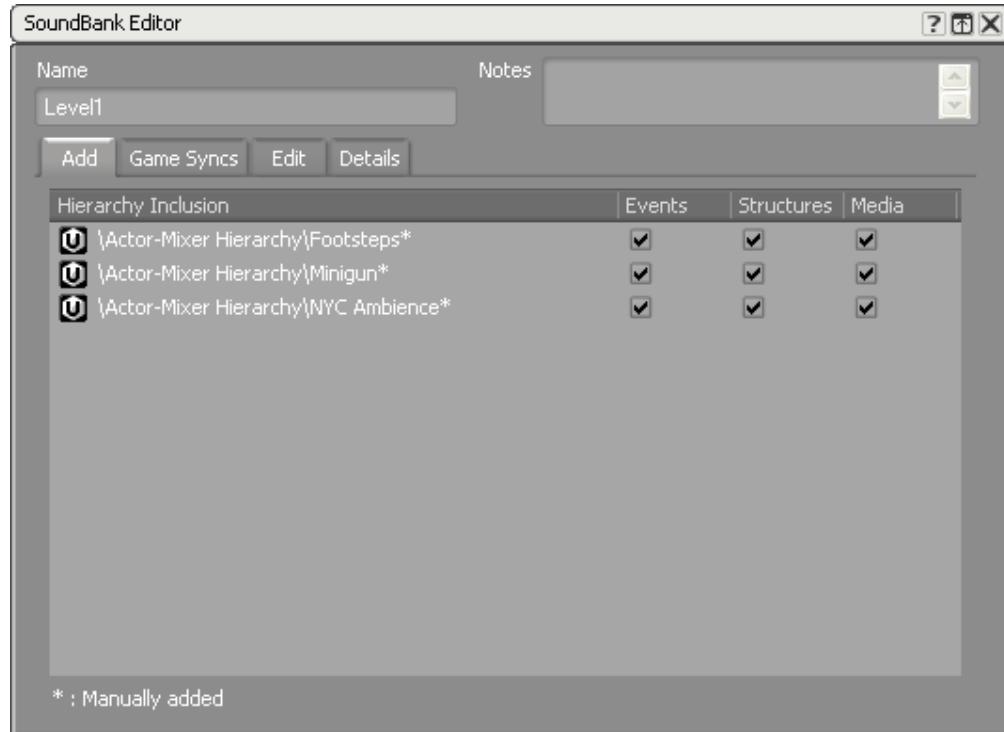


SoundBank Editor 是通过将不同事件、声音、音乐和振动结构以及音频文件添加到 SoundBank 来构建 SoundBank 的工具。您可以通过在 Project Explorer 或 SoundBank Manager 中双击 SoundBank 来打开 SoundBank Editor。

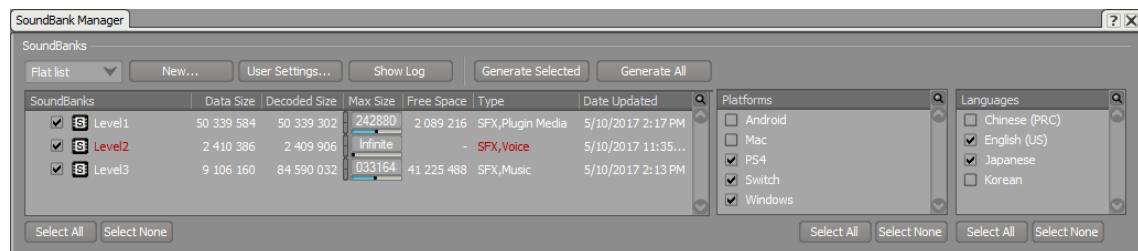
SoundBank Editor 包括以下四个选项卡：

- **Add** - 仅显示实际添加到 SoundBank 中的事件、对象层次结构、工作单元和文件夹。自动添加到 SoundBank 中的相应子对象将只显示在 Edit（编辑）选项卡中。在 Add（添加）选项卡中，用户还可以决定各个层级结构元素中那些素信息或媒体类型将包含在 SoundBank 中。

- **Add** —— 只显示已添加到 SoundBank 的实际事件、对象层级结构、工作单元和文件夹。在此选项卡上，您可以从 SoundBank 中弃用掉特定的 Game Sync。如果弃用了 Game Sync 的话，您则也弃用了与 Game Sync 相关的声音结构和媒体文件。
- **Game Sync** —— 显示与 Add 选项卡中所包含的事件和声音结构相关的信息。SoundBank 中媒体文件的其它信息也会显示出来，包括采样率、音频格式和文件大小。通过掌握这些附加信息，您可以轻松地微调各个文件的转码设置，以遵守特定平台的限制。用户可以根据语言和对象类型来对列表进行筛选，并将需要从 SoundBank 中弃用的任何工程元素取消选择。
- **Detail** —— 显示与选定 SoundBank 中不同元素的大小相关的详细信息，以及报告可能缺失或已被替换的任何文件。



在工程开发期间，您可以随时从 SoundBank Manager 中生成 SoundBank。SoundBank Manager 显示已经创建的 SoundBank 的列表，以及有关各个 SoundBank 的类型和大小的一些基本信息。其中还包括平台和语言的单独列表，一旦选中此列表，它将确定要生成的 SoundBank。



理解 SoundBank 如何加载到游戏中

在了解如何在 Wwise 中构建和生成 SoundBank 之前，理解游戏加载和管理 SoundBank 信息的不同方式非常重要。您选择什么方法将取决于许多不同的因素，包括您开发的游戏类型、运行游戏的平台以及工程团队自己设置的约束。

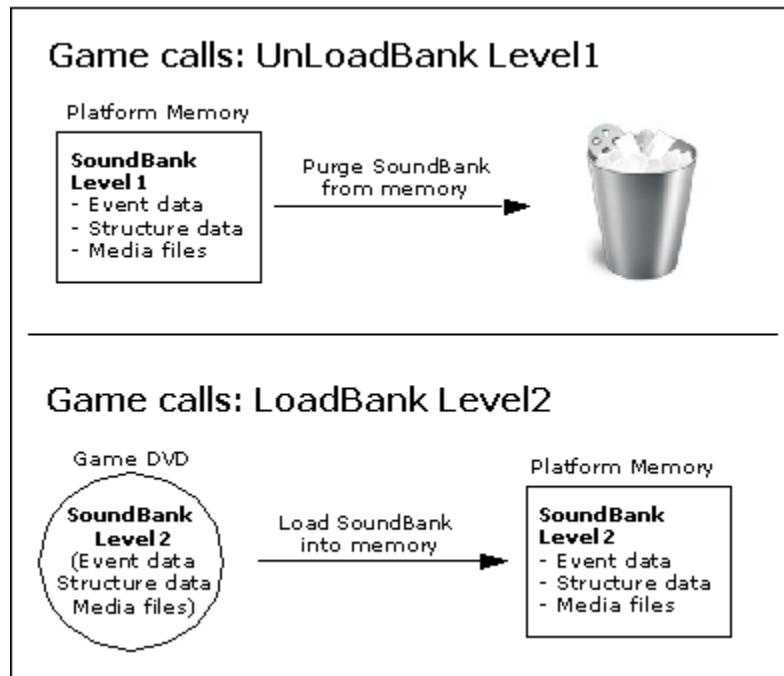
为了提高灵活性和满足几乎所有游戏类型的需求，Wwise 提供了多种将音频和振动加载到游戏中的方法，包括以下几种：

- 加载整个 SoundBank
- Prepare SoundBank（全部内容）
- Prepare 动作事件

加载整个 SoundBank

加载 SoundBank 的传统方法使用同时包含事件数据、对象结构数据和媒体的 SoundBank。这些 SoundBank 的全部内容在游戏的特定时刻加载和卸载，确保事件数据和相关媒体在被触发时就可以播放。

下图演示在游戏中使用传统方法创建的 SoundBank 如何在玩家从 1 级升级到 2 级时加载到平台内存和从平台内存中卸载。



由于特定 SoundBank 的所有数据和媒体同时加载到内存中，因此这种方法不仅确保所有数据和媒体可以在需要时播放，而且在游戏中需要执行极少的磁盘寻址，从而将磁盘腾出来执行对磁盘需求较大的其它任务。

此方法的主要缺点是在 SoundBank 的整个加载期间将占用大量的内存，使您无法灵活地处理复杂的大型游戏。此方法还显式加载 SoundBank 中的所有内容，而不验证媒体文件是否已经加载到内存。这可能导致同一媒体文件被多次加载到内存中。尽管

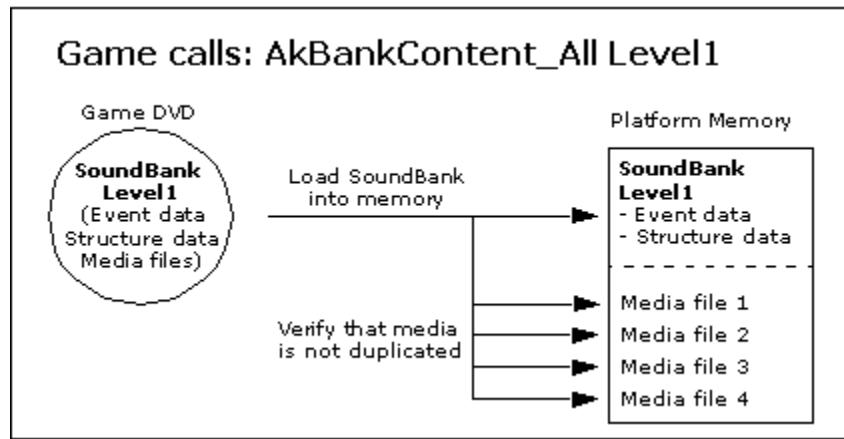
此传统方法有上述缺点，但是在许多情况下仍非常有用。例如，用于所有数据和媒体必须随时可用的经典弹珠游戏。

Prepare SoundBank (全部内容)

为了克服 LoadBank() 机制的一些缺点，您可以先采用 Prepare 操作来“预备” SoundBank 而不是使用 AkBankContent_All() 加载它们（译注：为了突出 Prepare 是一种 API 调用，文档尽量用英语原文来代表这种预备操作，而不做通篇翻译）。在使用此方法时，SoundBank 仍可包含所有内容类型（事件、结构数据和媒体文件）；但是，此方法不是立即加载媒体文件，而是通过使用 PrepareEvent() 机制来将所有媒体加载到内存中。通过使用此机制加载媒体，Wwise 首先查看媒体文件是否已经存在内存中，然后再加载它。这可以避免内存中出现媒体文件重复，从而将内存占用保持在最低水平。

除了可以节省内存外，此方法还可以保证顺序访问磁盘，这将避免在使用 PrepareEvent() 中一次 Prepare 一个 Event 时可能发生的随机磁盘寻址。

下图演示了“Prepare Bank (All Content)”（预备 SoundBank (全部内容)）机制如何将元数据和内容加载到平台内存中。



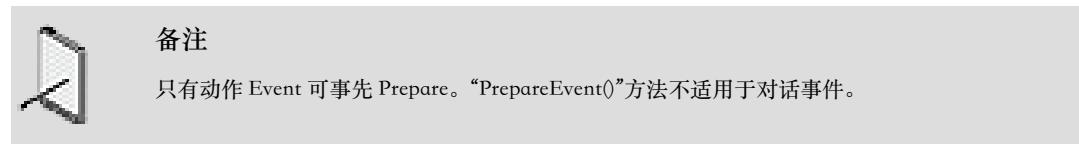
离线解压缩

在加载包含 [Vorbis](#) 编码或 [Opus](#) 编码媒体的 SoundBank 时，可利用 [Preparation_LoadAndDecode](#) 预备类型将这些媒体文件解码为非压缩 [PCM](#) 文件。这会使 SoundBank 体积变大，但当事件调用媒体文件时，将立即播放而无需进行解码。

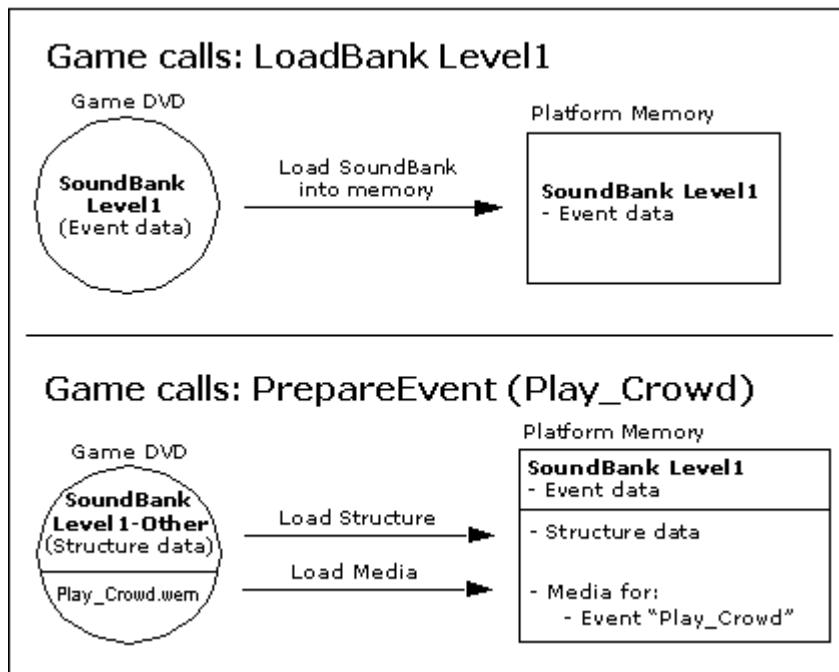
Prepare 动作事件

PrepareEvent() 方法只有在绝对需要时才会动态加载媒体。使用此方法时，动作事件元数据必须位于 SoundBank 中，相关媒体文件必须位于文件系统中可访问的位置。相应结构元数据可包含于事件所在的同一 SoundBank 中，也可包含于单独的 SoundBank 中。在使用此方法时，包含动作事件元数据的 SoundBank 使用 LoadBank() 加载并保留在内存中。在动作事件被游戏调用前，声音引擎先 Prepare (预备) 这些事件。Prepare 事件时将从文件系统中加载所有被引用的媒体文件，如果所有被引用的结构元数据尚未加载，则还将从 SoundBank 中加载这些结构元数据。当动作事件不再需要时，可将它 Unprepare (解除预备)，相应媒体文件于

是将从内存中被清除掉。为了将内存占用保持在最低水平并避免内存中出现媒体文件重复，Wwise 在加载媒体文件前会先确认它不存在于内存中。



下图说明如何事先 Preapre Event，以便只将必需的媒体文件加载到内存中。

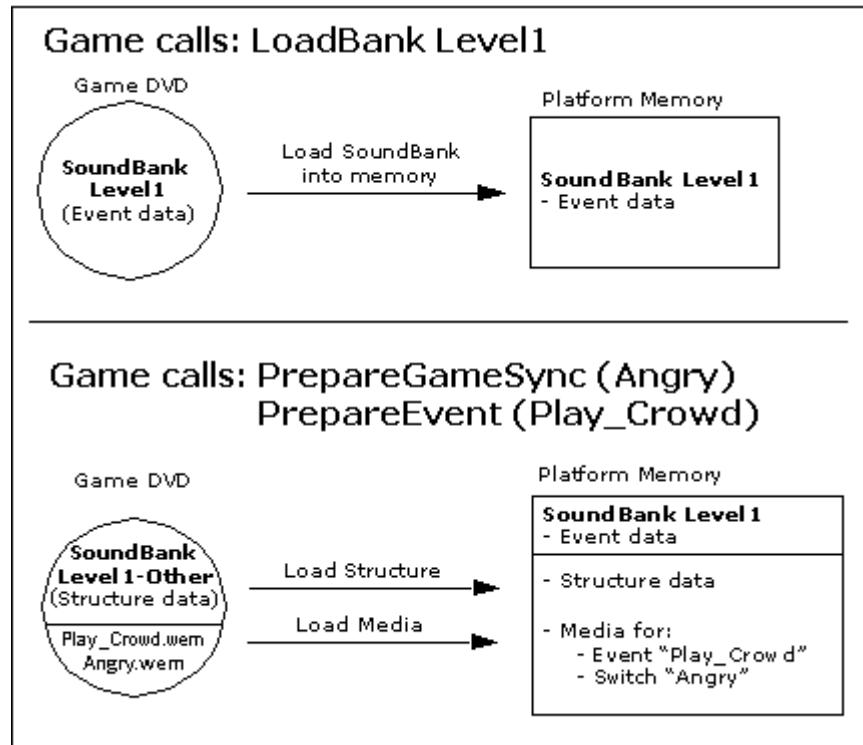


当元数据（又称结构）未存储在 Event 所在的同一 SoundBank 时，Wwise 需要一种方法来找到位于另一 SoundBank 中的相应数据。为此，Wwise 加入了对其它 SoundBanks 中存储的相应内容的引用。Wwise 可以使用名称或 ID 来引用其它 SoundBank。要在声音引擎中使用 SoundBank 名称，必须在 Project Setting 对话框的 SoundBank 选项卡上选择 Use SoundBank Names 选项。要使用 ID，则取消勾选该选项。

被 SoundBank 引用的媒体必须作为松散文件存储在磁盘中，否则必须能够被底层 IO (Low-Level IO) 解析（例如在 File Package 中）。

此方法一般在内存利用率方面非常高效，但它需要额外的磁盘寻址，当大量文件已经在磁盘上使用流来做数据传输的情况下，此方法可能就不太适合。另外，在游戏使用了 State 和 Switch 的情况下，不需要的媒体文件可能会被白白加载到内存中。例如，您针对游戏中人群的不同情绪或能量级别设置了不同的人群声音，如果游戏的特定区域中只有“愤怒”Switch 声音有效时，则所有级别的人群声还是都会被加载到内存中来，而这样您是不会愿意的。为了克服这一问题，您还可以 Prepare 特定的 State 或 Switch，只加载与 Prepare 的 State 或 Switch 相关的媒体文件。

下图显示了如何提前 Prepare Switch，以进一步限制在游戏任何特定时刻加载到内存中的媒体量。



虽然提前 Prepare Game Sync 可以优化内存占用，但注意，这也会降低媒体加载到内存中的速度。读取时间延长是因为声音引擎需要搜索磁盘，以查找与 Prepare 的 Game Sync 相对应的声音。

通过动态地只加载必需的音频文件，第二种方法可以为您处理以下情况提供更大的灵活性：处理具有大量声音的超大区域或关卡，或者以非常有限的内存量来存储事件数据、结构数据和媒体。

您可以看到，每种方法各有长短。您选择的方法取决于游戏的特定场景、要求和限制。在确定策略后（使用一种方法还是同时使用多种方法），您可以开始相应地填充和优化您的 SoundBank。



备注

LoadBank、AkBankContent_All、PrepareEvent、PrepareGameSyncs 和 AkBankContent_StructureOnly 函数均可通过 Wwise API 访问。有关加载 SoundBank 和 Prepare 事件或 Game Sync 的详细信息，请参阅 Wwise SDK 的《集成 SoundBanks》一节。

有关可用于管理 SoundBank 的不同方法的详细概述，请参阅[SoundBank 的管理策略](#)。

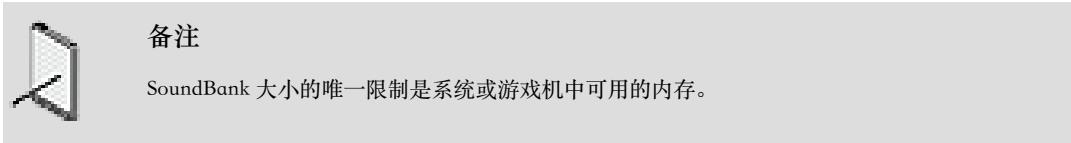
构建 SoundBank

要在声音引擎中使用 SoundBank 名称，必须在 Project Setting 对话框的 SoundBank 选项卡上选择 Use SoundBank Names 选项。接下来，您可以定义游戏所需的 SoundBank 的大小限制和数量。然后，您可以根据选择的方法以及游戏中的角色、对象、区域或关卡将事件、对象结构和媒体组织到各种 SoundBank 中。

Wwise 不限制可以加载到内存中的 SoundBank 数量。另外对 SoundBank 本身的小也没有限制。各个工程各不相同，您可以自行决定最适合您工程的方法。

构建 SoundBank 的过程涉及以下任务：

- [创建 SoundBank](#)
- [填充 SoundBank](#)
- [管理 SoundBank 的内容](#)
- [为工程生成 SoundBank](#)

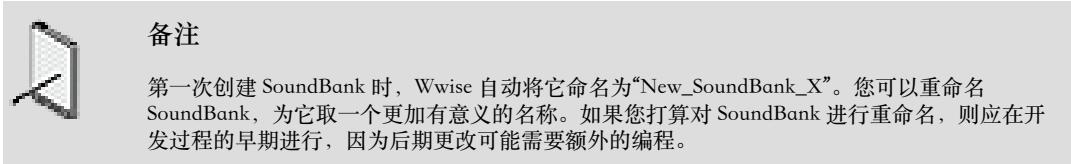


创建 SoundBank

在 Wwise 中，您可以手动创建单个 SoundBank，也可以通过导入定义文件（definition file）创建一批 SoundBank。定义文件由您正在用于将事件集成到游戏中的 3D 应用程序或关卡编辑器生成。有关导入定义文件的详细信息，请参阅[通过导入定义文件填充 SoundBank](#)。

您可以在 Wwise 的两个不同区域中手动创建 SoundBank：

- [在 Project Explorer 中创建 SoundBank 的方法是：](#)
- [在 SoundBank Manager 中创建 SoundBank 的方法是：](#)



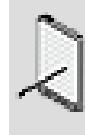
在 Project Explorer 中创建 SoundBank 的方法是：

1. 通过执行以下任一操作切换到 SoundBank 布局：
 - 在菜单栏中，点击 Layouts > SoundBank。
 - 按 F7。
2. 在 Project Explorer 中，切换到 SoundBank 选项卡。
3. 执行以下操作之一：
 - 选择虚拟文件夹或工作单元，然后点击 Project Explorer 中的 SoundBank 图标。
 - 右键点击虚拟文件夹或工作单元，然后从快捷菜单中选择 New Child > SoundBank。

此时新的 SoundBank 将高亮显示在 Project Explorer 中。

4. 输入 SoundBank 的新名称并按 Enter。

SoundBank 于是创建完成并添加到 SoundBank 列表中了。



备注

SoundBank 名称中只能包含不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

5. 在 Max Size 文本框中，指定您要为 SoundBank 分配的最大游戏内存。

在 SoundBank Manager 中创建 SoundBank 的方法是：

1. 通过执行以下任一操作切换到 SoundBank 布局：

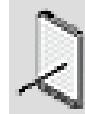
- 在菜单栏中，点击 Layouts > SoundBank。
- 按 F7。

2. 在 SoundBank Manager 中，点击 New 按钮。

此时将打开 New SoundBank 对话框。

3. 在 SoundBank 层级结构中，选择您要在其中创建 SoundBank 的工作单元。

4. 在 Name 字段中，将默认名称替换成适合新 SoundBank 的名称。



备注

SoundBank 名称中只能包含不带变音符号的罗马字母、数字和下划线。名称还必须以字母或下划线开头。

5. 点击 OK。

SoundBank 于是创建完成并添加到 SoundBank 列表中了。

6. 在 Max Size 文本框中，指定您要为 SoundBank 分配的最大游戏内存。



备注

如果您正在使用 Workgroup 版本控制插件并且新 SoundBank 所在的 Work Unit 未检出，Wwise 会询问您是否等到保存工程时才检出工作单元。

相关主题

- [填充 SoundBank](#)
- [管理 SoundBank 的内容](#)
- [监视 SoundBank 的详情](#)
- [为工程生成 SoundBank](#)

填充 SoundBank

根据您创建的游戏类型和选择的 SoundBank 策略，SoundBank 可包含不同类型的信息和媒体。您选择的策略将最终确定 SoundBank 的大小和内容。SoundBank 可包含以下任意类型的工程元素的组合：

- Event 数据
- 声音、音乐或振动结构数据

- 媒体文件

您可以使用以下任一方法填充 SoundBank：

- [通过导入定义文件填充 SoundBank](#)- 导入定义文件。
- [手动填充 SoundBank](#)- 将 Event、对象结构和工作单元等工程元素从 Project Explorer 拖到选定的 SoundBank。

通过导入定义文件填充 SoundBank

创建并填充 SoundBank 的方法之一是导入定义文件（definition file）。定义文件由您用于将事件集成到游戏中的 3D 应用程序或关卡编辑器生成。定义文件是用制表符分割的文件，其中按 SoundBank 分类别列出了游戏中的所有 Event。定义文件必须包括 SoundBank 的名称和相应的 Event 名称，它们之间通过制表符隔开。以下示例展示如何编写定义文件，以便 Wwise 能够读取它。

```
SB1#"Event_01"
SB2#"Event_02"
SB2#"Event_03"
SB3#"Event_04"
```

使用关键字在 SoundBank 中启用／弃用元素

定义文件还可包括特殊关键字，用于定义将在 SoundBank 中启用或弃用的工程元素类型。有以下关键字可供使用：

- Event** —— 指定 SoundBank 将包含 Event 信息。
- Structure** —— 指定 SoundBank 中将包含声音、音乐或振动结构信息。
- Media** —— 指定 SoundBank 中将包含媒体文件。
- GameSyncExclusion** —— 指定特定 Game Sync 将从 SoundBank 中弃用。
弃用 Game Sync 时，所有相关声音结构和媒体文件也会被弃用。关键字 -GameSyncExclusion 必须结合以下其中一个关键字使用：
 - State** —— 指定特定状态及其所有相关对象和媒体文件都将从 SoundBank 中弃用。此关键字后面必须写上 State Group 名称和 State 名称，名称之间通过制表符隔开。
 - Switch** —— 指定特定切换开关及其所有相关对象和媒体文件都将从 SoundBank 中弃用。此关键字后面必须写上切换开关组名称和切换开关名称，名称之间通过制表符隔开。
 - Trigger** —— 指定特定触发器及其所有相关对象和媒体文件都将从 SoundBank 中弃用。该关键字后面必须写上 Trigger 名称。
- DialogueEvent** —— 将在 SoundBank 中启用指定特定对白事件。此关键字后面必须写上对白事件的名称、GUID 或 short ID（短 ID）。它后面还可以写上 Event、Structure 或 Media 关键字作为包含选项。
- EffectShareset** —— 将在 SoundBank 中启用指定特定效果器 ShareSet。此关键字后面必须写上效果器 ShareSet 的名称、GUID 或 short ID。它后面还可以写上 Structure 或 Media 关键字作为包含选项。
- AuxBus** —— 指定将特定辅助总线加入 SoundBank。此关键字后面必须写上总线的名称、GUID 或 short ID。它后面还可以写上 Structure 或 Media 关键字作为包含选项。

```
SB1#"Event_01"  
SB1#"Event_02"#Structure  
SB1#"Event_03"#Media#Structure  
SB1#"Event_04"#Event#Media#Structure
```

在上图中，被称为“My_SoundBank_Normal”的第一个 SoundBank 将包含与 Event_01 相关的所有 Event 数据、结构数据和媒体文件。被称为“My_SoundBank_EventandStructure”的第二个 SoundBank 将只包含与 Event_02 相关的 Event 和结构数据。被称为“My_SoundBank_Media”的第三个同时也是最后一个 SoundBank 将只包含与 Event_03 相关的媒体文件。

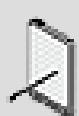


备注

定义文件中包含的所有 Event 必须事先在 Wwise 中创建。如果缺少任何事件，它们则将在导入定义日志 (Import Definition Log) 中显示为“Event Missing”。

在使用 -GameSyncExclusion 关键字时，必须另起一行为弃用的各个关键字创建单独的条目。弃用 Game Sync 还会弃用所有对应的对象结构和媒体文件。下图展示 -GameSyncExclusion 关键字在定义文件中的书写方式。(→ 表示制表符)

```
SB1#-GameSyncExclusion#State#StateGroupName#StateName  
SB1#-GameSyncExclusion#Switch#SwitchGroupName#StateName  
SB1#-GameSyncExclusion#Trigger#Trigger
```



备注

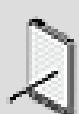
Game Sync (State Group、State、Switch Group、Switch 和 Trigger) 的名称不需要使用引号 (“”)。

```
SB1#-EffectShareSet#"effect1"#Media  
SB1#-EffectShareSet#"effect2"#Structure  
SB1#-EffectShareSet#"effect3"#Media#Structure
```

使用 ID 而不是字符串来标识事件和效果器

如果游戏未对 Event 名称使用字符串，则可以在定义文件中使用以下任一进制系统定义 Event：

- 十六进制
- 十进制



备注

十六进制和十进制系统可用于标识定义文件中的 Event，但不可用于标识 Game Sync 弃用项中的 State、Switch 和 Trigger。

下图展示如何在定义文件中使用三种不同的进制系统定义 Event。(→ 表示制表符)

```
SB1#"Event_01"  
SB1#26507443  
SB1#0x19478B3
```



技巧

SoundBank 定义文件还可用于跟踪已经集成到游戏中的、缺失的以及仍需在 Wwise 中创建的 Event。某个程序员可以从游戏中生成 Event 列表，然后您可以将定义文件导入 Wwise。您可以使用日志文件中的信息将游戏中的 Event 与在 Wwise 中创建的 Event 匹配起来。

通过导入定义文件创建 SoundBank 的方法是：

1. 在 Project Explorer 中，切换到 SoundBank 选项卡。
2. 右键点击您要在其中创建 SoundBank 的工作单元。
3. 从快捷菜单中选择 Import SoundBank Definition。

此时将会打开 Open (打开) 对话框。

4. 前往保存定义文件的位置。
5. 点击 Open。

此时将打开 Import Definition Log 对话框。

6. 仔细查看日志中的导入活动。导入活动可以为以下任意一项：
 - Inclusion Added —— 新的事件或效果已添加到现有 SoundBank。
 - SoundBank Created —— 创建了新的 SoundBank。
 - Inclusion Removed —— 某Event 或效果器已从现有 SoundBank 中移除。
 - Event Missing —— 工程中不再存在某 Event 或者某 Event 尚未创建。
 - Effect Missing —— 工程中不再存在某效果器或者某效果器尚未创建。
 - Exclusion Added —— 某 Game Sync 已从现有 SoundBank 中弃用。
 - Exclusion Deleted —— 某 Game Sync 已在现有 SoundBank 中被重新启用。
 - Exclusion Missing —— 工程中不再存在某 Game Sync 或者 Game Sync 尚未创建，因此不能添加弃用项或者从弃用列表中删除。
 - No Change Detected —— 导入的 SoundBank 与 Wwise 中已存在的 SoundBank 相同。
7. 点击 Close。

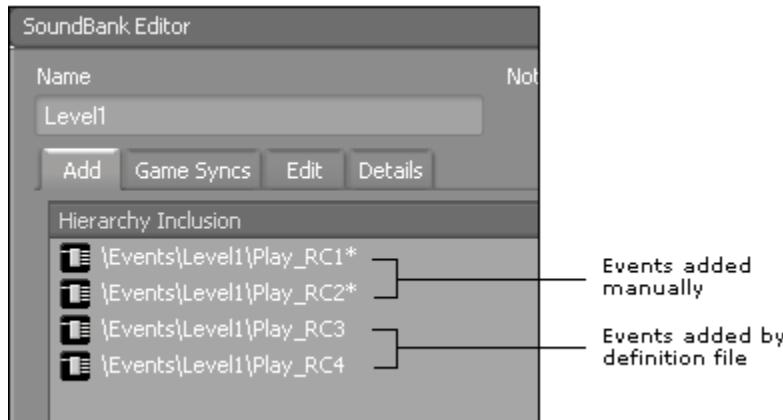
Wwise 于是会创建 definition file (定义文件) 中定义的 SoundBank，并将指定的 Event、对象结构和媒体添加到 SoundBank 中去。

相关主题

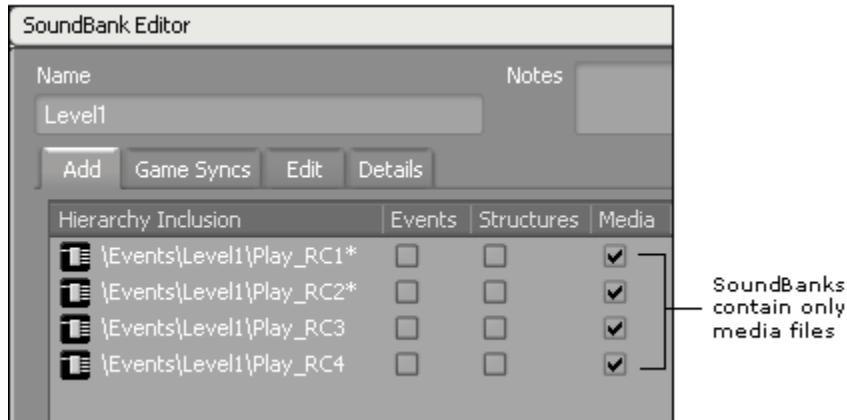
- [创建 SoundBank](#)
- [在 SoundBank 中启用/弃用工程元素](#)
- [使用 Game Sync 启用/弃用工程元素](#)
- [搜索 SoundBank 中的元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [从 SoundBank 中移除工程元素](#)
- [监视 SoundBank 的详情](#)
- [为工程生成 SoundBank](#)

手动填充 SoundBank

在通过导入定义文件创建所有 SoundBank 后，您可以手动优化它们的内容，添加和弃用单个 Event、Game Sync 和对象。为帮助您识别 SoundBank 中手动添加或弃用的工程元素，在 SoundBank Editor 的 Add 和 Game Sync 选项卡上，条目旁边会添加一个星号 (*)。

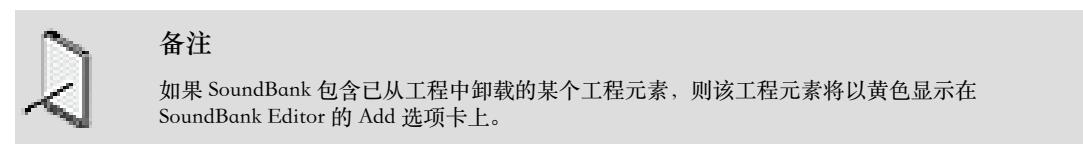


在将工程元素添加到 SoundBank 时，所有相应 Event、对象结构和媒体文件也会自动添加到 SoundBank。然而，您可以通过加入您需要的工程元素类型来修改 SoundBank 的内容。例如，您可能会需要让 SoundBank 只包含媒体。



为了帮助您更高效地工作，Wwise 允许将完整的结构、工作单元和文件夹从 Project Explorer 拖放到 SoundBank 编辑器中。如果这些工程元素包含子对象，则这些子对象也会自动添加到 SoundBank，但它们只会在 Edit 选项卡上显示。

当父对象、工作单元或文件夹添加到 SoundBank 时，它们与原始工程元素间将保持链接状态。例如，假设 SoundBank 中包含一个 Event 工作单元。当您稍后将两个 Event 添加到工作单元时，它们会自动添加到 SoundBank 中。如果将 SoundBank 加载到 SoundBank Editor 中，您会注意到 Add 选项卡仍然只显示原始的事件 Work Unit，但 Edit 选项卡中会显示其中的所有事件，包括刚刚添加的两个事件。通过保持此链接，Wwise 确保您的 SoundBank 始终包含工程的最新修改。



手动填充 SoundBank 的方法是：

1. 通过执行以下操作之一切换到 SoundBanks 布局：
 - 在菜单栏中，点击 Layouts > SoundBank。

- 按 F7。
- 2. 双击 SoundBank Manager 中的任一 SoundBank，将它加载到 SoundBank Editor 中。
- 3. 将以下任何元素从 Project Explorer 拖到 SoundBank Editor 的 Add 选项卡：

角色混音器、容器、声音、Motion 和音乐对象。

Folders (文件夹)

Event

Work Units

工程元素及其所有相关 Event 的整个层级结构、对象结构和媒体文件将自动添加到 SoundBank。现在您可以进一步优化 SoundBank 的内容，确定您希望启用的工程元素类型。

相关主题

- [创建 SoundBank](#)
- [在 SoundBank 中启用/弃用工程元素](#)
- [使用 Game Sync 启用/弃用工程元素](#)
- [搜索 SoundBank 中的元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [从 SoundBank 中移除工程元素](#)
- [监视 SoundBank 的详情](#)
- [为工程生成 SoundBank](#)

管理 SoundBank 的内容

在创建并填充 SoundBank 后，您很可能需要优化内容，在 SoundBank 中启用、弃用和移除特定元素。由于 SoundBank 可包含大量不同的工程元素，为此 Wwise 为您提供了用于启用和弃用工程元素的不同方法，以及帮助您快速定位对象的搜索工具和两个独立筛选选项。

在管理 SoundBank 的内容时，您可以执行以下任务：

- [在 SoundBank 中启用/弃用工程元素](#)
- [使用 Game Sync 启用/弃用工程元素](#)
- [搜索 SoundBank 中的元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [在 SoundBank 之间移动工程元素](#)
- [从 SoundBank 中移除工程元素](#)

在 SoundBank 中启用/弃用工程元素

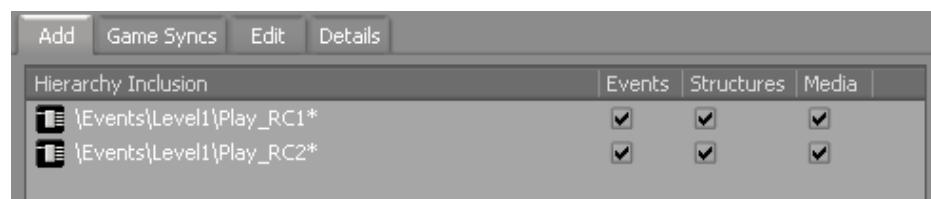
在将 Event 或声音结构添加到 SoundBank 时，Wwise 会自动启用所有相应的 Event、声音结构和音频文件。根据您对游戏所使用的 SoundBank 策略，可能需要从 SoundBank 中移除所有 Event、声音结构或媒体文件。

示例——在 SoundBank 中启用/弃用元素

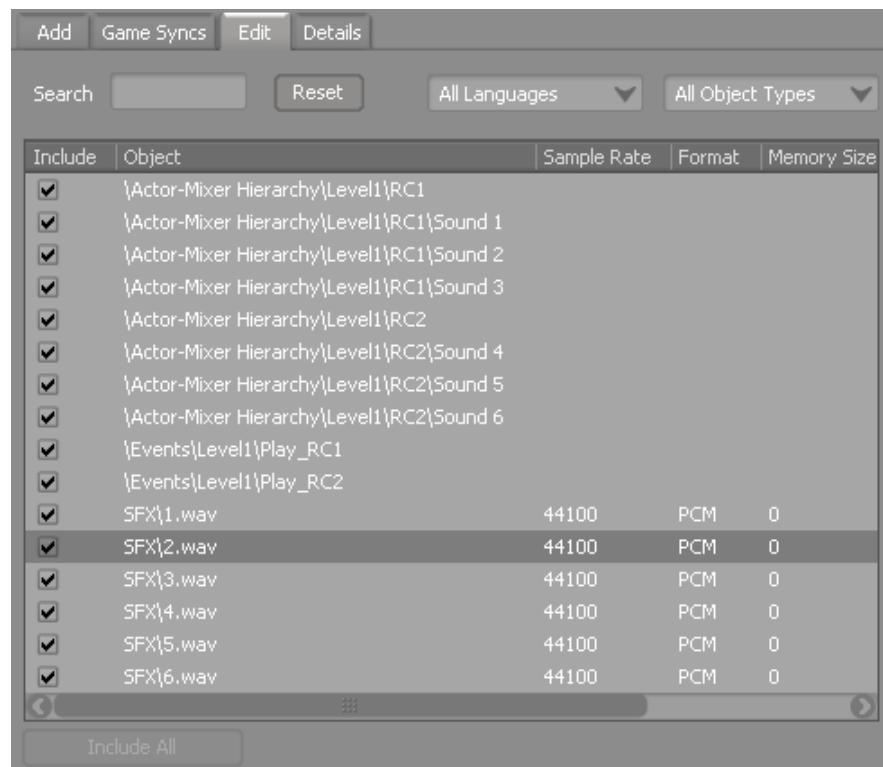
假设您拥有一个包含以下工程元素的小工程：

- 两个随机容器
- 六个声音对象
- 六个音频文件
- 两个播放 Event

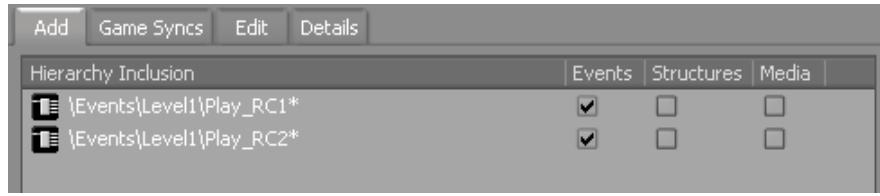
您打算在调用 Event 前先通过 Prepare Event 来动态加载媒体文件，因此希望为工程中的所有 Event 构建一个 SoundBank。在这种情况下，您可以先创建 SoundBank，然后将两个 Event 拖到 SoundBank Editor 的 Add 选项卡。



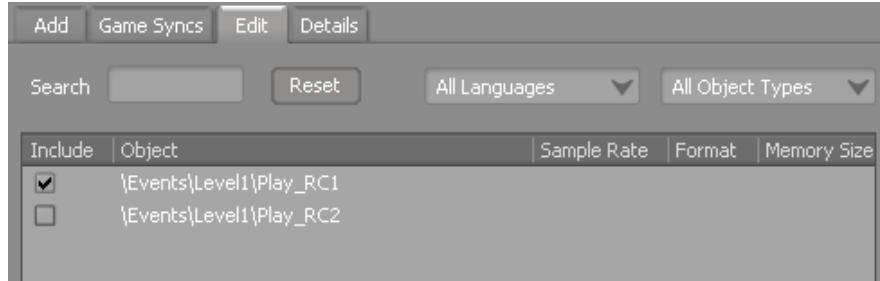
在默认情况下，两个随机容器和六个声音对象的数据，以及六个音频文件，也将添加到 SoundBank。您可以在 SoundBank Editor 的 Edit 选项卡上查看 SoundBank 的全部内容。



要在此 SoundBank 中只包含 Event，您可以返回 Add 选项卡，取消选择 Structures 和 Media 列下的复选框。



现在 SoundBank 中只启用了两个 Event 的数据。您可以进一步优化 SoundBank，在 SoundBank Editor 的 Edit 选项卡上弃用个别元素。例如，您可能只想在 SoundBank 中包含其中一个事件，因此您可以弃用不需要的另一个事件。

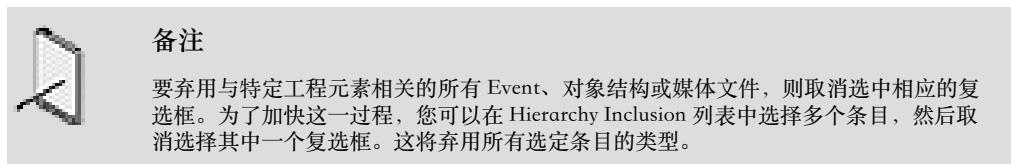


对于包含媒体的 SoundBank，您可以查看有关各个媒体文件的信息，包括它的采样率、音频格式和文件大小。通过掌握这些附加信息，您可以轻松地微调各个文件的转码设置，以遵守特定平台的限制。要更改媒体文件的转码设置，只需右键点击列表中的条目并选择 Conversion Settings 即可。

通过为游戏采取正确的策略，并可灵活地在 SoundBank 中启用和弃用单个元素以及修改单个媒体文件的转码设置，您可以有效地克服游戏的内存约束。

在 SoundBank 中启用工程元素类型的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 对于列表中的各个条目，选择以下任意或所有工程元素类型，以在您的 SoundBank 中启用它们：
 - **Event** —— 将在 SoundBank 中启用所有相应 Event 数据。
 - **Structure** —— 将在 SoundBank 中启用所有相对对象结构数据。
 - **Media** —— 将在 SoundBank 中启用所有相应媒体文件。

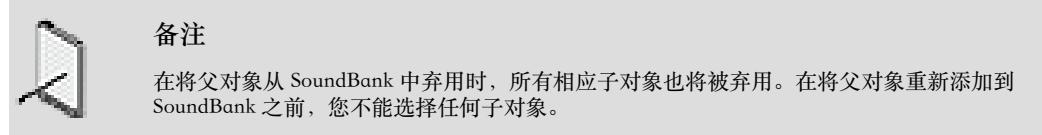


从 SoundBank 中弃用单个工程元素的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
 2. 切换到 Edit (编辑) 选项卡。
- 此时将显示 SoundBank 中包含的所有 Event、对象结构和媒体文件的完整列表。
3. 使用搜索或筛选工具定位您要从 SoundBank 中弃用的工程元素。

4. 取消选择 Event、对象结构或媒体文件的相应复选框。

工程元素及所有相关工程元素于是从 SoundBank 中弃用掉了。



5. 重复执行步骤 3 和 4，直至将您要移除的所有工程元素都从 SoundBank 中弃用出去。

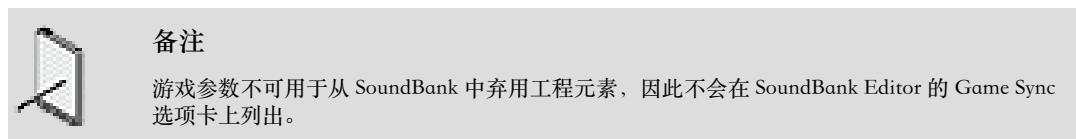
相关主题

- [使用 Game Sync 启用/弃用工程元素](#)
- [搜索 SoundBank 中的元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [在 SoundBank 之间移动工程元素](#)
- [从 SoundBank 中移除工程元素](#)

使用 Game Sync 启用/弃用工程元素

在 SoundBank 中启用或弃用声音的另一种快速、便捷的方法是使用 Game Sync。在将 Event 或声音结构添加到 SoundBank 时，SoundBank Editor 的 Game Sync 选项卡上会自动创建相关 Game Sync 的列表。您可以通过弃用特定 Game Sync 将特定元素从 SoundBank 中弃用。在将 Game Sync 从 SoundBank 弃用时，引用该 Game Sync 的所有对象结构和媒体文件也会被弃用。

通过根据游戏同步器来启用 / 弃用声音，可以更好地控制在游戏特定时刻加载的声音。这意味着您可以更好地管理游戏平台的内存限制。

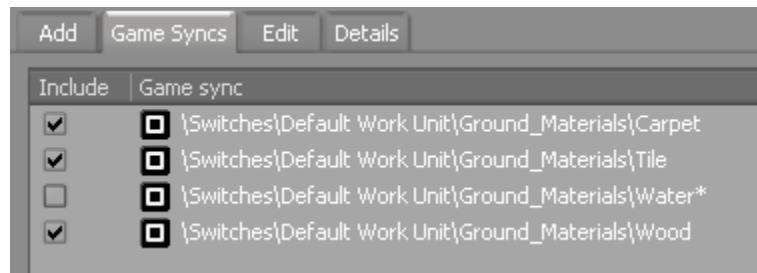


示例——使用 Game Sync 启用/弃用元素

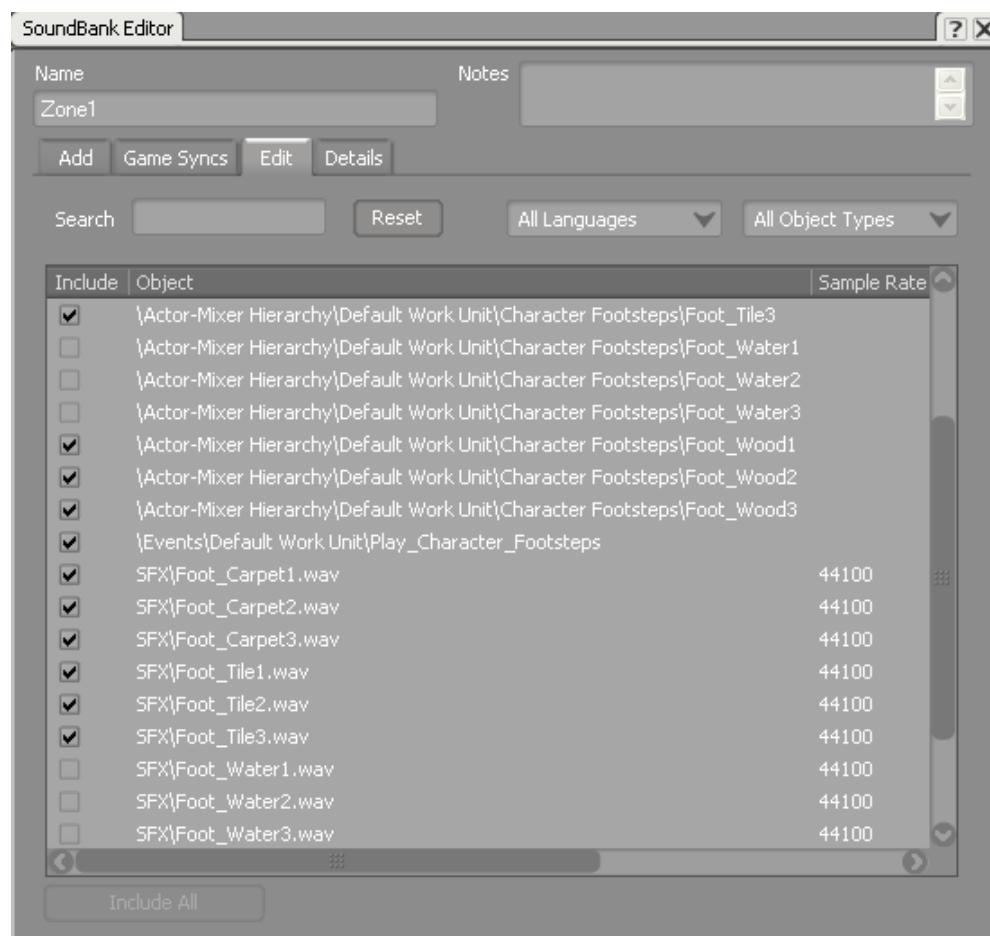
假设您的一个小游戏由三个不同的区域构成。要完成游戏通关，主人公必须穿过区域 1、2 和 3。在 Wwise 中，您使用切换容器处理主人公的脚步声。您为主人公可能行走的不同地面材质创建了以下四个切换开关：

- Carpet
- Tile
- Wood
- Water

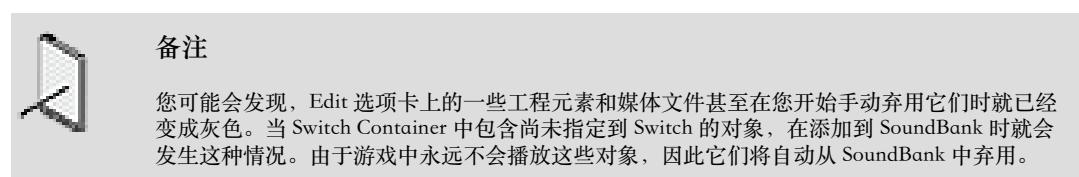
在游戏中，“Water”切换开关只用于区域 2 和 3，而其它三个切换开关用于所有三个区域。您决定创建三个 SoundBank；每个对应于一个区域。由于各个 SoundBank 中必须包含 Switch Container，因此您需要从区域 1 的 SoundBank 中移除“与水相关”的 Event、声音结构和媒体文件。为此，您可以加载区域 1 的 SoundBank，然后切换到 Game Sync 选项卡，并取消选择“Water”切换开关。



此操作将从 SoundBank “Zone1”中弃用“水”相关声音结构和媒体文件。如果您切换到 Edit 选项卡，则会发现 SoundBank 中不再包含引用此切换开关的声音结构、Event 和媒体文件。



通过导入包含 Game Sync 弃用项的定义文件也可自动将 Game Sync 从 SoundBank 弃用。有关使用定义文件的详细信息，请参阅[通过导入定义文件填充 SoundBank](#)。



使用 Game Sync 将工程元素从 SoundBank 中弃用的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 切换到 Game Syncs 选项卡。

此时将显示与 Add 选项卡上的 Event 或对象相关的 Game Sync 的完整列表。

3. 定位您要从 SoundBank 中弃用的 Game Sync，并取消选择相应的复选框。

此时 Game Sync 及其所有相关对象和媒体文件就从 SoundBank 中弃用掉了。

4. 重复步骤 3，直至从 SoundBank 中弃用掉您想移除的所有 Game Sync。

相关主题

- 在 SoundBank 中启用/弃用工程元素
- 搜索 SoundBank 中的元素
- 筛选 SoundBank 中的元素列表
- 在 SoundBank 之间移动工程元素
- 从 SoundBank 中移除工程元素

将在 SoundBank 中启用插件媒体

某些效果器（例如 Convolution Reverb）需要媒体文件才能工作。例如，Convolution Reverb 需要冲激响应文件作为它的插件媒体。这些插件媒体文件需要包含在 SoundBank 中，效果器才能在游戏中正常工作。您还需要确保在需要实例化效果器时加载包含启用了这些媒体文件的 SoundBank。

在 SoundBank 中启用插件媒体文件有很多方法：

- 您可以在 SoundBank 的启用列表中插入 ShareSet 效果器对象。
- 您可以在 SoundBank 的启用列表中插入所有者对象（总线、辅助总线或 Actor-Mixer Hierarchy 对象）。
- 如果效果器由 Actor-Mixer Hierarchy 对象所拥有，您则可以在 SoundBank 的启用列表中插入 Event。

通过查看 SoundBank Editor / Edit 选项卡可验证 SoundBank 中是否包含插件媒体文件。

搜索 SoundBank 中的元素

根据游戏的复杂程度，SoundBank 中可能即使没有数千也得有数百个不同的工程元素。因此，拥有能迅速找到特定工程元素的必要工具是至关重要的。为帮助您在 Edit（编辑）选项卡中找到元素，可以使用搜索工具。

搜索 SoundBank 中的元素的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 切换到 Edit（编辑）选项卡。
3. 在 Search 字段中，输入您要查找的工程元素的名称。

Wwise 对列表进行筛选，使列表仅包含与您输入的名称相匹配的对象。

4. 点击 **Reset**（重置）以清空 Search 字段并再次显示 SoundBank 的全部内容。

相关主题

- 在 SoundBank 中启用/弃用工程元素
- 使用 Game Sync 启用/弃用工程元素
- 筛选 SoundBank 中的元素列表
- 在 SoundBank 之间移动工程元素
- 从 SoundBank 中移除工程元素

筛选 SoundBank 中的元素列表

为帮助您查找 SoundBank 中的特定元素，您可以根据语言或对象类型筛选列表。

筛选 SoundBank 中的元素列表的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 切换到 Edit（编辑）选项卡。
3. 要根据语言筛选列表元素，则从 Language Filter 中选择以下其中一个选项：

All languages -- 全部语言。显示工程中所有语言下的工程元素

“Language” -- “语言”。仅显示选定语言下的工程元素。

4. 要根据对象类型筛选元素列表，则从 Object Type 列表中选择以下其中一个选项：

All object types -- 所有对象类型。显示 SoundBank 中的所有对象

Event -- 事件。仅显示 SoundBank 中的事件

Structure -- 结构。仅显示 SoundBank 中的对象结构。

Media -- 媒体。仅显示 SoundBank 中的媒体文件。

相关主题

- 在 SoundBank 中启用/弃用工程元素
- 使用 Game Sync 启用/弃用工程元素
- 搜索 SoundBank 中的元素
- 在 SoundBank 之间移动工程元素
- 从 SoundBank 中移除工程元素

在 SoundBank 之间移动工程元素

在常见游戏开发工程的过程中，SoundBank 的数量和内容会发生变化。因此，能否高效管理这些 SoundBank 非常重要。为了帮助您更好地管理 SoundBank 的内容，Wwise 可以让您轻松地在 SoundBank 之间移动工程元素。

在 SoundBank 之间移动工程元素的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 从 SoundBank Editor 的 Add 选项卡中选择您要移到另一个 SoundBank 中去的若干个工程元素。
3. 将选中项从 SoundBank Editor 的 Add 选项卡拖到 SoundBank Manager 中的另一个 SoundBank 中去。

选定的工程元素从原始 SoundBank 中移出，并添加到新的 SoundBank。

相关主题

- [在 SoundBank 中启用/弃用工程元素](#)
- [使用 Game Sync 启用/弃用工程元素](#)
- [搜索 SoundBank 中的元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [从 SoundBank 中移除工程元素](#)

从 SoundBank 中移除工程元素

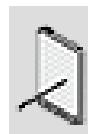
在从 SoundBank 中移除元素时，所有相应 Event、对象结构和媒体文件也会被移除。

虽然无效的工程元素在生成期间会被 Wwise 忽略，不会导致错误或占用额外空间，但应将它们从 SoundBank 中移除，以保持工程的完好度。无效的 Event 和对象结构出现在 SoundBank Editor 上，它们的名称后面会带“Missing”一词。

从 SoundBank 中移除工程元素的方法是：

1. 将 SoundBank 加载到 SoundBank Editor。
2. 在 Add 选项卡上，选择您要移除的若干个对象。
3. 按 Delete 键。

此时工程元素连同任何相应的事件、对象结构或媒体文件就从 SoundBank 中移除掉了。



备注

如果某个对象或媒体文件被 SoundBank 中的多个工程元素引用，则移除其中一个工程元素并不会自动移除该对象或媒体文件，因为其它工程元素仍在引用它。

相关主题

- [搜索 SoundBank 中的元素](#)
- [在 SoundBank 中启用/弃用工程元素](#)
- [使用 Game Sync 启用/弃用工程元素](#)
- [筛选 SoundBank 中的元素列表](#)
- [在 SoundBank 之间移动工程元素](#)

管理 SoundBank

创建初始 SoundBank 后，您可以在整个开发过程中继续更改它。在管理 SoundBank 时，您可以执行以下任务：

- 重命名 SoundBank
- 删除 SoundBank
- 监视 SoundBank 的详情

重命名 SoundBank

在创建 SoundBank 后，您可以返回，根据需要对它进行重命名。



备注

如果您打算对 SoundBank 进行重命名，则应在开发过程的早期进行，因为后期更改可能需要额外的编程。

重命名 SoundBank 的方法是：

1. 在 Project Explorer 中，切换到 SoundBank 选项卡。
2. 选择您要重命名的 SoundBank，并执行以下操作之一：

按 F2。

点击 SoundBank 名称字段。

此时 SoundBank 的名称变成高亮显示。

3. 将 SoundBank 名称替换为新名称，然后按 Enter。

相关主题

- 填充 SoundBank
- 删除 SoundBank
- 监视 SoundBank 的详情

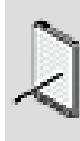
删除 SoundBank

当您不再需要某个 SoundBank 时，可以将它删除。

删除 SoundBank 的方法是：

1. 在 Project Explorer 中，切换到 SoundBank 选项卡。
2. 执行以下操作之一：
 - 右键点击您要删除的 SoundBank，并选择 Delete Selection。
 - 点击您要删除的 SoundBank，然后按 Delete 键。

选定的 SoundBank 于是被删除了。



备注

如果您误删了某个 SoundBank，则可以按 Ctrl+Z 或点击 Edit > Undo 来撤消删除操作。

相关主题

- [填充 SoundBank](#)
- [重命名 SoundBank](#)
- [监视 SoundBank 的详情](#)

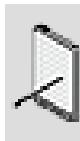
监视 SoundBank 的详情

由于遵循各个平台的内存限制很关键，因此能够管理工程中的各个 SoundBank 和对它们进行故障排查非常重要。您需要确定哪些 SoundBank 超出了最大体积、哪些语言文件缺失了、生成的 SoundBank 的总大小以及高效管理 SoundBank 所需要的其它重要信息。

Wwise 将在 SoundBank Editor 的 Detail 选项卡中显示这些详情。

SoundBank Editor											
Name	Notes										
Level1											
Add			Game Syncs			Edit	Details				
Maximum size	30000000	Date updated	4/9/2008 3:37 PM								
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	

如果您的 SoundBank 包含 Sound Voice 对象，则 Detail 选项卡上的信息将按照语言进行组织。Wwise 还会以红色显示一些信息，帮助您迅速找出潜在问题。



备注

SoundBank Manager 和 SoundBank Editor 中的所有内存信息均以字节为单位。

监视 SoundBank 的详情的方法是：

1. 在 SoundBank Manager 中，双击想要查看详细信息的 SoundBank。

相关信息会显示在 SoundBank Editor 的 Details 选项卡中。

2. 请查看 Detail 选项卡的内容了解有关下列内容的信息：

- **Type** -- 类型。SoundBank 中包含的对象类型，例如音效、声部、音乐和振动。
- **Language** -- 语言。相应 SoundBank 的语言。
- **SFX** -- 音效。音效声对象占用的内存量。
- **Music** -- 音乐。音乐对象占用的内存量。
- **Voice** -- 声部。特定语言的 Sound Voice 对象占用的内存量。

- **Data Size** -- 数据大小。特定语言的音效、语音和振动对象占用的总内存量。
- **Free Space** -- 空闲空间。SoundBank 中的剩余空间。只有指定最大体积时才会显示此值。
- **Missing Files** -- 缺失文件。特定语言缺失的媒体文件数量。
- **Files Replaced** -- 被替换文件。当前被参考语言的音频文件所替换的、缺失的 Sound Voice 音频文件的数量。
- **Memory Size** -- 内存大小。将加载到内存中的 SoundBank 数据所占用的空间量。
- **Prefetch Size** -- 预读大小。流播放文件的预读数据所占用的空间量。该部分数据通过 SoundBank 加载到内存中，以确保流媒体文件零延迟播放。
- **File Size** -- 文件大小。生成的 SoundBank 文件的总大小。

相关主题

- [填充 SoundBank](#)
- [管理 SoundBank 的内容](#)
- [删除 SoundBank](#)

定义 SoundBank 的自定义属性

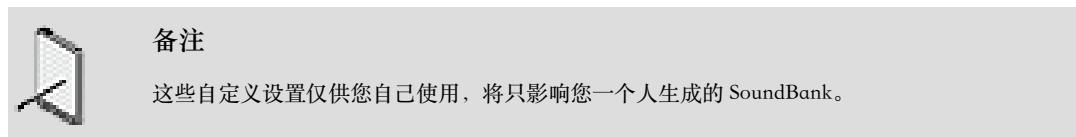
在为每种平台和语言生成 SoundBank 之前，您需要定义 SoundBank 的设置。这些设置是在 Project Settings 中针对工程全局定义的，但您也可以创建自定义用户设置来覆盖这些设置。可以定义以下 SoundBank 用户设置：

- [定义自定义 SoundBank 设置](#) —— 确定生成的 SoundBank 中所包含的信息。
- [为保存的 SoundBank 指定自定义位置](#) —— 确定硬盘或网络中保存 SoundBank 的位置。
- [定义在生成 SoundBank 前 / 后执行的自定义用户步骤](#) —— 确定在刚要生成 SoundBank 前将执行的任务。
- [定义在生成 SoundBank 前 / 后执行的自定义用户步骤](#) —— 确定在刚刚生成 SoundBank 后将执行的任务。
- [指定外部源文件的输入/输出位置](#) —— 指定用于 Template Source 插件的外部音频源的输入和输出路径。

有关在工程级别设置这些属性的信息，请参阅[定义工程的 SoundBank 设置](#)。

定义自定义 SoundBank 设置

SoundBank 设置确定哪些信息将在生成过程中被启用、启用这些信息的方式以及生成的格式。虽然这些信息是在工程级别定义的，但有时候您会不希望沿用这些信息，而是创建自己的自定义设置。您选择的设置将取决于游戏如何访问 SoundBanks 中的数据和媒体。

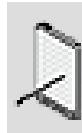


定义自定义 SoundBank 用户设置的方法是：

1. 在 SoundBank Manager 中，点击 User Settings（用户设置）。

此时将打开 SoundBank User Settings 对话框。

2. 选择 Override Project SoundBank Settings 选项。
3. 选下以下任意选项为 SoundBank 定义自定义设置：
 - Allow SoundBanks to exceed maximum size -- 允许 SoundBanks 超出最大体积。即使超出指定的最大体积时，仍将生成 SoundBank。
 - Generate SoundBank content file -- 生成 SoundBank 内容文件。用于创建列出各个 SoundBank 的内容的文件。内容文件包括有关 Event、总线、状态和切换开关的信息，以及流播放文件和内存媒体文件的完整列表。
 - Generate header file -- 生成头文件。创建将 Event、状态、切换开关和游戏参数名称映射到 ID 的头文件。
 - Max attenuation -- 最大衰减距离。在 SoundBanksInfo.xml 文件中包含各个 Event 的最大衰减距离信息。
 - Estimated duration -- 预计时长。在 SoundBanksInfo.xml 文件中包含各个 Event 的预计最大和最小时长，以及声音是无限循环播放还是只播放一次。
 - Use SoundBank Name -- 使用 SoundBank 名称。使用 SoundBank 名称（勾选时）或 ID（不勾选时）来命名生成的 .bnk SoundBank 文件，以及在一个 SoundBank 中引用另一个 SoundBank。
4. 如果选择生成头文件，则必须确定是否保存它。为此，请执行以下操作：
 - 直接在文本框中输入路径。
 - 点击 Browse 按钮 (...) 并使用浏览器前往您选择的位置。



备注

您可以使用完整路径或相对路径来指定保存头文件的位置。使用相对路径时，工程文件夹会作为路径的起始点。

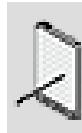
5. 如果选择生成 SoundBank 内容文件，则可以选择所需的文本文件格式和 SoundBank content file format 选项。



技巧

如果文件路径、对象名称或对象备注包含非 ANSI 字符，则应使用 Unicode 格式。

6. 点击 OK 以应用这些设置。



备注

这些自定义用户设置将用于生成 SoundBank，直至您取消选择不沿用选项。

相关主题

- [为保存的 SoundBank 指定自定义位置](#)
- [定义在生成 SoundBank 前 / 后执行的自定义用户步骤](#)

为保存的 SoundBank 指定自定义位置

工程中生成的所有 SoundBank 会自动保存在 Project Settings 的 SoundBanks 选项卡中指定的工程默认位置。如果您需要更改此位置，请创建自定义用户设置。

在为 SoundBank 指定保存路径时，您可以使用完整路径，也可以使用相对路径。在使用相对路径时，使用工程文件夹用作路径的起始位置。例如，下列完整路径和相对路径指定的位置是相同的：

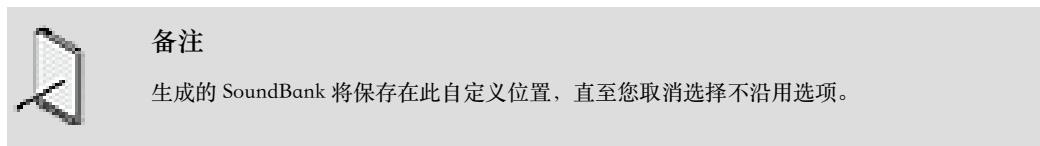
- C:\Wwise Projects\My Project\GeneratedSoundBanks\Windows
- GeneratedSoundBanks\Windows\

为保存的 SoundBank 指定新位置的方法是：

1. 在 SoundBank Manager 中，点击 User Settings（用户设置）。

此时将打开 SoundBank User Settings 对话框。

2. 选择 Override Project SoundBank Paths（不沿用工程 SoundBank 路径）选项。
3. 通过执行以下一项操作来指定路径：
 - 直接在文本框中输入路径。
 - 点击 Browse 按钮 (...) 并使用浏览器前往您选择的位置。
4. 点击 OK 以使您所做的任何更改生效。



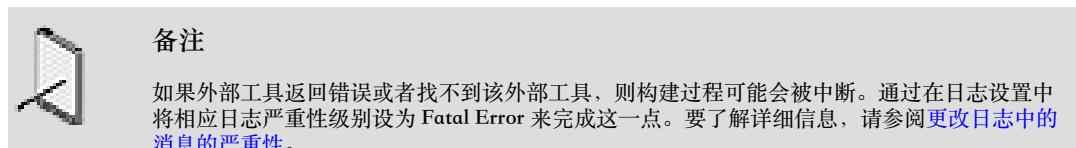
相关主题

- [定义自定义 SoundBank 设置](#)
- [定义在生成 SoundBank 前 / 后执行的自定义用户步骤](#)

定义在生成 SoundBank 前／后执行的自定义用户步骤

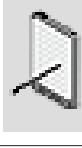
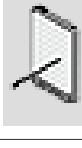
根据工作流程，在生成 SoundBank 之前或之后您可能需要立即执行特定步骤或任务。例如，在生成特定 SoundBank 文件前想从版本控制系统中 check out 它们；或者在生成后想立即将流播放文件复制到 SoundBanks 目录中。虽然这些信息通常在工程级别定义，但有时候您可能不需要沿用这些信息，而需要创建您自己的自定义设置。

在 Wwise 中，这些任务类型是通过创建命令行来定义的。Wwise 中有一个专门的命令行编辑器，方便您构建数目不限的命令行。为了进一步简化过程，编辑器中包含命令行中可使用的所有 Wwise 专用环境变量和其它 Windows 环境变量的列表。

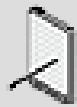


当自定义 Global opening step 失效时，整个构建进程将被中断，并且不会创建 SoundBank。不过当针对特定平台的自定义步骤失效时，则将只跳过该平台。然而，由于进程运行不完全成功，因此将跳过 Global closing step。外部进程必须返回非零值才会被视为失效。

可用于编写自定义命令行的专用 Wwise 变量如下：

命令行变量	描述
<code>\$(AllowExceedMaximum)</code>	指定当 SoundBanks 超过指定最大大小时是否生成该 SoundBanks。 当选择了 Allow SoundBanks to exceed maximum（允许 SoundBanks 超过最大大小）选项时，此变量设为 True。
<code>\$(ContentFileFormat)</code>	指定生成的 SoundBank 内容文件的文件类型。可能的值为： <ul style="list-style-type: none">• ANSI• Unicode
<code>\$(GenerateContentFile)</code>	指定是否创建文件来列出各个 SoundBank 的内容。内容文件包括有关 Event、Buss、State 和 Switch 的信息，以及流播放音频文件和内存音频文件的完整列表。 当选择了 Generate SoundBank content files（生成 SoundBank 内容文件）选项时，此变量设为 True。
<code>\$(GenerateHeaderFile)</code>	指定是否生成将 Event、状态、切换开关和游戏参数名称映射到 ID 的头文件。 当选择了 Generate Header File（生成头文件）选项时，此变量设为 True。
<code>\$(GenerateMaxAttenuationInfo)</code>	指定是否为 Event 生成最大衰减距离信息。 当选择了 Metadata Options: Max attenuation 选项时，此变量设为 True。
<code>\$(GenerateEstimatedDuration)</code>	指定是否为 Event 生成预计最大和最小时长以及时长类型信息。 当选择了 Metadata Options: Estimated Duration 选项时，此变量设为 True。
<code>\$(HeaderFileFullPath)</code>	头文件的完整路径，即 <code>\$(HeaderFilePath)\Wwise_IDs.h</code> 。
<code>\$(HeaderFilePath)</code>	保存头文件的路径或位置。 此路径来自 Header file path（头文件路径）文本框。
<code>\$(InfoFilePath)</code>	当前平台的信息文件的完整文件名。
<code>\$(IsRunningFromCmdLine)</code>	指定 Wwise 启动的命令行中是否带有“-generatesoundbanks”选项。
<code>\$(LanguageList)</code>	传送到命令行的语言的列表或者 SoundBank Manager 中的选定语言的列表。  备注 列表采用空格隔开。
<code>\$(Platform)</code>	当前平台的名称。
<code>\$(SoundBankList)</code>	传送到命令行的 SoundBank 的列表或者 SoundBank Manager 中的选定 SoundBank 的列表。  备注 列表采用空格隔开。并用双引号括起整个参数中的列表。
<code>\$(SoundBankPath)</code>	保存当前平台的 SoundBanks 的路径或位置。
<code>\$(UseSoundBankNames)</code>	指定选用 SoundBank 名称（设为 true 时）或 ID（设为 false 时）来用于命名生成的 SoundBank BNK 文件，以及在 SoundBanks 中用于引用其它 SoundBank 中的媒体。

命令行变量	描述
	当选择了 Use SoundBank names (使用声音包名称) 选项时, 此变量设为 True。
<code>\$(WwiseExeDriveLetter)</code>	Wwise 可执行程序 (Wwise.exe) 所在的电脑盘符。
<code>\$(WwiseExePath)</code>	Wwise 可执行程序 (Wwise.exe) 的路径或位置。
<code>\$(WwiseExeProcessID)</code>	Wwise 可执行程序 (Wwise.exe) 的进程 ID (数字形式)。
<code>\$(WwiseProjectDriveLetter)</code>	Wwise 工程所在的电脑盘符。
<code>\$(WwiseProjectName)</code>	当前工程的名称。
<code>\$(WwiseProjectPath)</code>	Wwise 工程的路径或位置。
<code>\$(WaapiWampPort)</code>	WAAPi 为 WAMP 协议使用的端口。
<code>\$(WaapiHttpPort)</code>	WAAPi 为 HTTP 协议使用的端口。



备注

环境变量自动进行映射, 例如 `$(WWISESDK)`。

为了尽可能提高灵活性, Wwise 支持为以下类型的步骤指定命令行:

- **Global opening step** -- 全局初始步骤。适用于所有平台并且在任何其它步骤前执行的命令行。
- **Platform-specific pre-generation step** -- 针对平台的生成前步骤。适用于特定平台并且在生成 SoundBanks 前执行的命令行。
- **Platform-specific post-generation step** -- 针对平台的生成后步骤。适用于特定平台并且在生成 SoundBanks 后执行的命令行。
- **Global closing step** -- 全局结束步骤。适用于所有平台并且在所有其它步之后执行的命令行。

在默认情况下, 各个工程都包含平台专有的生成后步骤 (post-generation step) 命令行, 该命令行将流播放文件复制到 SoundBank 目录下。然而, 您可以通过执行一个不同的命令行将任何类型的任务自动化。Wwise 还自带另一个出厂命令行, 它使用 File Packager 来生成包含 SoundBank 中所有数据和媒体的文件包。有关 File Packager 的详细信息, 请参阅[第 36 章 管理 File Package](#)。有关加载出厂命令行的详细信息, 请参阅[加载出厂设定/自定义命令行](#)。

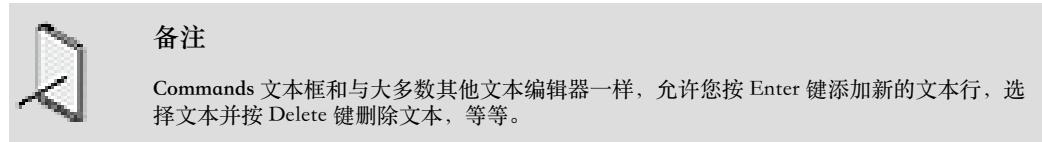
您还可以将您创建的命令行保存到文件 (WCMDLINE) 中, 以便今后在同一工程中使用, 跨工程使用, 或者与其它用户共享。有关保存命令的详细信息, 请参阅[保存自定义命令行到文件](#)。

定义在生成 SoundBank 前执行的用户任务的方法是:

1. 在 SoundBank Manager 中, 点击 User Settings (用户设置)。
- 此时将打开 SoundBank User Settings 对话框。
2. 选择 Override Project Pre-Generation Step (不沿用工程预生成步骤) 选项。
 3. 要添加或修改在 Project Settings 对话框中定义的全局起始预生成步骤, 则点击相应的 Edit 按钮 (...)。

Pre-Generation Step Editor（生成前步骤编辑器）打开。

4. 在 Description 文本框中，输入名称，名称要能清晰描述将要执行的步骤或任务。
5. 在 Commands 文本框中，根据需要编写新的命令行或者编辑现有的命令行。



6. 如果要在命令中插入内置宏和环境变量，则请执行以下操作：

在 Macros（宏）分组框中，选择以下其中一个选项：

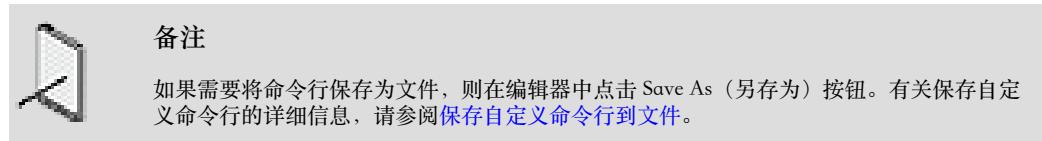
- Built-in Macros -- 内置宏。显示可用于 Wwise 命令行中的一列 Wwise 专用变量。
- Environment Variables -- 环境变量。显示可用于 Wwise 命令行中的一列 Windows 专用环境变量。

要添加变量到命令行中的话，要执行以下操作中的一项：

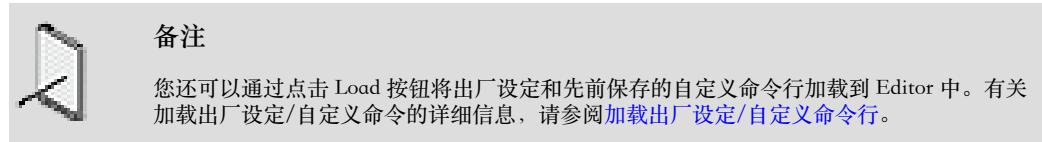
- 双击列表中的变量。
- 从列表中选择变量，然后点击 Insert（插入）。

根据需要，继续添加变量到命令行。

7. 如果需要添加另一个预生成步骤，则只需转至第一行的末尾，按 Enter，然后开始创建新的命令行即可。
8. 点击 OK 保存命令行并关闭 Pre-Generation Step Editor。



9. 要添加或修改现有平台专用的预生成步骤，则对各个平台重复执行步骤 3 至 8。



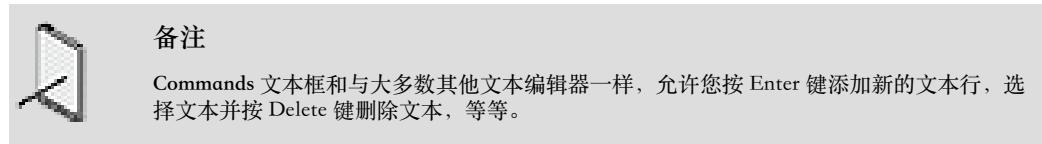
定义在生成 SoundBank 后执行的用户任务的方法是：

1. 在 SoundBank Manager 中，点击 User Settings（用户设置）。
- 此时将打开 SoundBank User Settings 对话框。
2. 选择 Override Project Post-Generation Step 选项。
 3. 要添加或修改在 Project Setting 对话框中定义的现有后生成步骤，则点击 Edit 按钮 (...) 之一。

此时将打开 Post-Generation Step Editor。

4. 在 Description 文本框中，输入名称，名称要能清晰描述将要执行的步骤或任务。

- 在 Command 文本框中，根据需要编写新的命令行或编辑当前命令行。



- 如果要在命令中插入内置宏和环境变量，则请执行以下操作：

在 Macros (宏) 分组框中，选择以下其中一个选项：

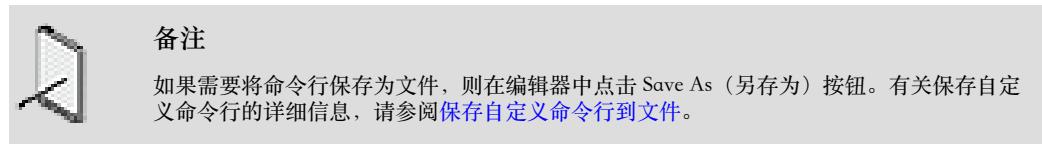
- Built-in Macros -- 内置宏。显示可用于 Wwise 命令行中的一列 Wwise 专用变量。
- Environment Variables -- 环境变量。显示可用于 Wwise 命令行中的一列 Windows 专用环境变量。

要添加变量到命令行中的话，要执行以下操作中的一项：

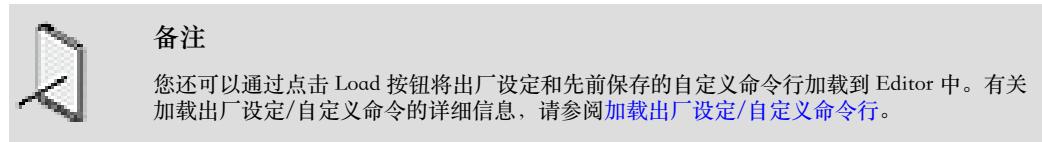
- 双击列表中的变量。
- 从列表中选择变量，然后点击 Insert (插入)。

根据需要，继续添加变量到命令行。

- 如果需要添加另一个预生成步骤，则只需转至第一行的末尾，按 Enter，然后开始创建新的命令行即可。
- 点击 OK 以保存命令行并关闭 Post-Generation Step Editor。



- 对于全局结束步骤或其它各个平台，重复执行步骤 3 至 8。



相关主题

- [定义自定义 SoundBank 设置](#)
- [为保存的 SoundBank 指定自定义位置](#)
- [加载出厂设定/自定义命令行](#)
- [保存自定义命令行到文件](#)

指定外部源文件的输入/输出位置

如果您计划使用 Template Source 插件，则必须指定在运行时与模板源相关联的外部音频文件的位置。您还必须指定保存转码后的源的文件夹，以便在玩游戏时 Wwise 能够使用它们。

指定外部音频源的输入/输出路径的方法是：

- 在 SoundBank Manager 中，点击 User Settings (用户设置)。

- 此时将打开 SoundBank Settings 对话框。
2. 切换到 External Sources 选项卡。
 3. 如果您想在 SoundBank 生成过程中对外部源转码，则选择 Convert external sources when generating SoundBank（在生成声音包时对外部源转码）选项。
 4. 要修改 Project Setting 对话框中指定的输入路径，则选择 Override Input Path 选项。
 5. 通过执行以下操作之一，指定外部源文件所在的文件夹：
 - 点击 External Source 列表，并输入音频文件所在的路径。
 - 点击对应于其中一个游戏平台的 Browse 按钮 (...)，前往包含该平台音频文件的文件夹，然后点击 Open。
 6. 对各个平台重复执行步骤 3。
 7. 要修改 Project Setting 对话框中指定的输出路径，则选择 Override Output Path（不沿用输出路径）选项。
 8. 通过执行以下操作之一指定保存外部源的文件夹：
 - 点击 External Sources Output Folder 列表，并输入要保存音频文件的路径。
 - 点击对应于其中一个游戏平台的 Browse 按钮 (...)，前往您想为该平台保存音频文件的文件夹，然后点击 OK。
 9. 对各个平台执行步骤 5。
 10. 点击 OK 以关闭 SoundBank Settings 对话框。

为工程生成 SoundBank

您可以为要创建的各个游戏平台和语言生成 SoundBank。在开发过程中，随时都可以生成 SoundBank，以便您测试如何将它们集成到各种游戏平台中。为了更加方便，Wwise 还可以同时为所有平台和所有语言生成 SoundBank。

在生成 SoundBank 时，SoundBank 可包含以下任何信息：

- Event 信息
- 声音、音乐、振动、角色混音器和容器信息
- 内存媒体的声音、音乐或振动数据
- 零延迟流播放文件的预读数据
- 流播放文件的引用

SoundBank 中包含的信息仅可在同一工程中使用，这意味着 SoundBank 只可与同一工程中生成的其它 SoundBank 一起使用。

如果 SoundBank 包含的 Event 或对象结构已从工程中删除，您则仍可以生成该 SoundBank。这些无效的工程元素在生成过程中将被 Wwise 忽略，不会导致错误或占用额外的空间。然而，您应该从 SoundBank 中移除这些无效的 Event 和对象结构，以维持工程的完好度。



备注

当 SoundBank 中的 Event 或对象结构变得无效时，相应的图标会被移除，并且在 SoundBank Editor 的 Add 选项卡上，无效 Event 或对象结构的名称后面将显示“Missing”一词。

在生成 SoundBank 前，您需要设置一系列选项来定义是否生成内容和头文件、是否可以使用 SoundBank 名称、用于保存 SoundBanks 的位置以及是否需要将流播放文件复制到 SoundBank 目录等。在 Wwise 中，这些 SoundBank 设置从两个不同的级别进行定义：

- **Project level** —— 为您的工程创建默认设置。在 Project Setting 对话框中定义这些设置。有关在工程级别定义 SoundBank 设置的详细信息，请参阅[定义工程的 SoundBank 设置](#)。
- **User level** —— 创建不沿用工程设置的自定义用户设置。在 SoundBank User Settings 对话框中定义用户设置。有关定义自定义用户设置的详细信息，请参阅[定义自定义 SoundBank 设置](#)。

备注

在为特定游戏机（例如 Xbox One 或 PlayStation 4）生成 SoundBank 前，您需要成为该平台的授权内容创作者 / 程序员，并拥有必要的平台 CAB 文件。如果安装包中不包含特定平台的 CAB 文件，则平台不会显示为 SoundBank Platform 列表中的可用选项之一。有关如何安装 Wwise 和相应 CAB 文件的详细信息，请参阅《Wwise 安装和迁移指南》的“安装问题”一节。

为工程生成 SoundBank 的方法是：

1. 通过执行以下任一操作切换到 SoundBank 布局：

- 在菜单栏中，点击 Layouts > SoundBank。
- 按 F7。

技巧

您可以从任何其它布局中打开 SoundBank Manager 的浮动视图，方法是点击 View > SoundBank Manager 或使用快捷键 (Shift+B)。

2. 在 SoundBank Manager 中，选择您要生成的 SoundBank。
3. 在 Platform 列表中，选择您要为它生成 SoundBank 的平台。
4. 在 Language 列表中，选择您要为它生成 SoundBank 的语言。
5. 要开始生成 SoundBank，则点击 Generate Selected。

此时将打开 Generating SoundBank 对话框，您可以在此查看 SoundBank 生成过程的进度。在生成 SoundBank 时，Generating SoundBanks - Completed 对话框会打开。

备注

先前生成的并且没有更改的 SoundBank 不会再生成。在这种情况下，Results 面板的 Created 列中将显示“Up to date”。

6. 仔细阅读对话框中的信息，确保所有 SoundBank 都已成功生成。
7. 作为最终的验证，仔细查看日志中的条目。SoundBank 日志中包含在生成过程中遇到的错误、警告和其它消息的完整列表。有关 SoundBank 日志的详细信息，请参阅[解决在生成期间遇到的 SoundBank 问题](#)。
8. 在阅读完 SoundBank 日志时，您可以将它随 Generating SoundBank - Completed 对话框一起关闭。

SoundBank 文件位于您指定的文件夹中，现在可以集成到您的游戏中了。

相关主题

- [定义工程的 SoundBank 设置](#)
- [定义 SoundBank 的自定义属性](#)
- [重命名 SoundBank](#)
- [监视 SoundBank 的详情](#)
- [删除 SoundBank](#)
- [使用脚本生成 SoundBank](#)
- [搜索 SoundBank 中的元素](#)
- [解决在生成期间遇到的 SoundBank 问题](#)

解决在生成期间遇到的 SoundBank 问题

将消息添加到 Log Ignore List

有时候在生成期间，您会不再想或不再需要在 SoundBank Log 中显示特定警告或消息。为防止这些类型的警告和消息出现在 SoundBank 日志中，您可以添加它们到 Log Ignore（日志忽略）列表中。如果您希望日志中重新显示这些消息时，则只需将这些消息类型从 Log Ignore List 中移除即可。

您还可以在 Project Setting 对话框中定义要加入 Log Ignore List 中的消息。有关详细信息，请参阅[管理在日志中出现的消息](#)。

将消息添加到 Log Ignore List 的方法是：

1. 在 SoundBank Log 中，选择您不再希望在日志中出现的消息类型。
2. 右键点击选中项，并选择 Add Message(s) to Ignore List。

下次为您的工程生成 SoundBank 时，这些类型的消息将不再出现在 SoundBank Log 中。

相关主题

- [解决在生成期间遇到的 SoundBank 问题](#)
- [为工程生成 SoundBank](#)
- [使用脚本生成 SoundBank](#)

如果 Wwise 在生成 SoundBank 时遇到任何问题，它将在 SoundBank Log 中显示有关各个问题的信息。在生成 SoundBank 时，您可以查看显示在 Generating SoundBank 对话框底部面板中的日志。另外，也可稍后使用 Logs（日志）视图的 SoundBank Generation（音频包生成）选项卡检查日志中的信息。

在特定情况下，您可能会需要防止特定消息出现在 SoundBank 日志中，实现方法是将您不希望出现的消息添加到 Log Ignore List 中。有关 Log Ignore List 的详细信息，

请参阅[将消息添加到 Log Ignore List](#)。您还可以限制消息在日志中显示的次数。有关设置此限制的信息，请参阅[管理在日志中出现的消息](#)。

相关主题

- [将消息添加到 Log Ignore List](#)
- [为工程生成 SoundBank](#)
- [使用脚本生成 SoundBank](#)

更改日志中的消息的严重性

在有些情况下，您可能想更改 SoundBank Generation（音频包生成）日志中显示的特定消息、警告或错误的严重性。为此，请转到 Project Settings（工程设置）对话框的 Log（日志）选项卡。

有关详细信息，请参阅[管理在日志中出现的消息](#)。

相关主题

- [解决在生成期间遇到的 SoundBank 问题](#)
- [为工程生成 SoundBank](#)
- [使用脚本生成 SoundBank](#)

使用脚本生成 SoundBank

如果您打算定期生成 SoundBank，则最好创建一个脚本来自动执行生成过程。为了做到这一点，您可以使用 Wwise SDK 从命令行来生成多个 SoundBank。要想了解此功能的完整描述，请参阅 Wwise SDK 文档中的[Wwise SDK > “深入探索”>“从命令行生成声音包”](#)。

使用 CopyStreamedFile 工具

CopyStreamedFiles 工具是一款独立的实用程序，可以将工程中的流播放文件复制到保存 SoundBank 的位置（在默认情况下，SoundBank 保存在 GeneratedSoundBanks 目录下的平台专用文件夹中）。从命令行调用此工具时，它首先扫描 SoundBanksInfo.xml 文件以找到采用流播放的文件，然后将它们从工程缓存文件夹复制到 SoundBank 目录下的相应文件夹。在默认情况下，各个工程包括用于调用 CopyStreamedFile 工具来完成此任务的后生成步骤命令行，但您可以修改此步骤或创建新的步骤来满足工程的特定需要。

CopyStreamedFile 工具位于 Wwise 可执行程序目录中。从命令行调用此工具时，需要定义特定参数，包括 SoundBanksInfo.xml 文件的位置和复制流播放文件的位置。下表列出了可用于命令行的参数。

参数	描述
-info <文件路径>	指定 SoundBanksInfo.xml 文件的位置。作为后生成步骤用于 Wwise 时，此选项设为“\$(InfoFilePath)”（带引号）。
-outputpath <文件夹路径>	指定从中复制流播放文件的基础文件夹，它应为生成 SoundBank 时所在的基础文件夹。在作为后生成步骤用于 Wwise 时，设为“\$(SoundBankPath)”（带引号）。

参数	描述
-languages "language1 languageN"	<p>指定要复制的语言列表。</p>  <p>备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。在语言列表中也可使用关键词“SFX”来复制它们。</p>
-banks "soundbank1 soundbankN"	<p>此参数为可选参数。在默认情况下复制所有语言。</p> <p>指定要为其复制流播放文件的 SoundBank 的列表。名称不包含 BNK 扩展名。或者可指定包含 SoundBank 名称列表的文本文件，方法是传入文本文件的完整路径，包括 TXT 扩展名。</p>  <p>备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。</p>
-verbose	<p>此参数为可选参数。在默认情况下复制所有流播放文件。</p> <p>启用额外游戏机文本输出。此参数为可选参数。</p>

相关主题

- [定义 SoundBank 的自定义属性](#)
- [为工程生成 SoundBank](#)
- [第 36 章 管理 File Package](#)

SoundBank 的管理策略

以下各节描述可用于生成 SoundBank 并将它们集成到游戏中的不同方法。在一款游戏中，您可以使用一种方法或多种不同方法的组合。由于各个游戏各不相同，因此您选择的方法将取决于游戏的具体需求。

您在创建 SoundBank 时所做的选择将极大地影响管理游戏音频和振动素材所需要的工作量，并将直接影响游戏的性能。强烈建议声音设计师和程序员均认真阅读本节，以便了解各种可能性。通过协作，您可以提出满足游戏特定需求的策略。记住，所有解决方案都可行，但您选择的策略应考虑到内存占用、I/O 存取和集成于游戏的便利性。每种方法各有长短，因此在大多数情况下，就是在内存占用和集成便利性之间进行平衡的问题。

以下各节中描述了以下方法：

- [方法 1：‘一体化’库](#)
- [方法 2：多个完整的 SoundBank](#)
- [方法 3：巨细无遗地管理您的媒体](#)
- [方法 4：Prepare 动作事件](#)
- [方法 5：对 Event 和 Game Sync（切换开关和状态）做 Prepare](#)



备注

另外还有其它两种方法，它们融合了现有 LoadBank() 和 PrepareEvent() 方法的长处。使用这两种方法可以 Prepare 整个 SoundBank，这意味着您可以将所有数据和媒体合并到同一

一个 SoundBank 中，避免内存中出现媒体重复，并只有在需要时才加载媒体。有关 Prepare SoundBank 的详细信息，请参阅 Wwise SDK 文档中的“声音引擎集成纵览”>“将 Wwise 元素集成到游戏中”>“Prepare SoundBanks”>“集成详情——SoundBanks”>“加载 SoundBank”>“Prepare SoundBank”。



离线解压缩

对于 Opus 和 Vorbis 音频格式，可提前解压媒体文件。虽然这将导致 SoundBank 变大，但可以缩短在调用媒体时访问媒体的时间，因为播放时不再需要等待解压缩了。

此策略可与一系列其它方法结合来优化素材的管理。

方法 1：“一体化”库

此方法适用于以下情形：

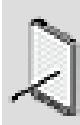
- 游戏拥有有限数量的音频或振动素材。
- 游戏拥有大量的空闲内存。

当然，大多数游戏一般没有可浪费的内存，但此方法的重大优势就是使用和维护起来非常简单。使用这种方法的主要原因是在将整个 Wwise 工程集成到游戏中时用时最少。

在 Wwise 中创建“一体化”SoundBank 的方法是：

1. 创建一个 SoundBank，然后取一个适当的名称。
2. • 将 SoundBank 加载到 SoundBank Editor 中，然后将以下项拖到“Add”选项卡中去：
 - “\Actor-Mixer Hierarchy\Default Work Unit”
 - “\Events\Default Work Unit”
 - “\Interactive Music Hierarchy\Default Work Unit”（只有使用互动音乐时才需要）
 - “\Dynamic Dialogue\Default Work Unit”（只有使用动态对白时才需要）

• 在默认情况下，Events、Structures 和 Media 列下的复选框全部选中。对于此方法，我们希望将所有内容都包含在 SoundBank 中，因此这非常有用。



备注

在新工程中，只有默认工作单元可用。如果创建了更多的工作单元，则这些工作单元也应按照需要添加到 SoundBank 中。

3. 在所有工作单元均已添加到 SoundBank 中后，您可以生成 SoundBank，然后把生成的 SoundBank 文件夹复制到游戏应用程序中。

将“一体化”SoundBank 集成到游戏中的方法是：

1. 由于此游戏只有一个 SoundBank，因此只需在初始化游戏时加载它。当然，必须首先正确初始化声音引擎。

下面的代码示例可以让您了解“一体化”SoundBank 是如何加载到游戏中的：

```
//#####
...
// #####"##"SoundBank#
AkBankID bankID; // #####
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 );
}
...
...
```

有关此方法的补充说明

下表列举了此方法的优点和缺点。

优点	缺点
从声音设计师的角度来看，维护 SoundBank 的内容最容易。 由于游戏总是加载同一个 SoundBank，因此更改 SoundBank 的内容时无需重新编译游戏。 在游戏中没有复杂的 SoundBank 加载和卸载过程。 在游戏中无需跟踪可用或者不可用的声音。	内存利用率不高，因为整个游戏的所有 Event、结构和内存媒体一直占用内存。

虽然这种方法是将音频和振动快速轻松集成到游戏中的好方法，但不要等到工程结束时才考虑换成别的方法来提高游戏内存利用率。

方法 2: 多个完整的 SoundBank

此方法适用于以下情形：

- 游戏或其音频 / 振动内容可以分成多个部分。

此方法非常适用于单人游戏，在单人游戏中，所有可能用到的声音和振动都只是由玩家在游戏中的当前位置决定的。通过将内容分割成多个 SoundBank，管理内存的效率比第一种方法更高，同时仍可轻松地将音频和振动集成到游戏中。

首先，必须确定如何划分 SoundBank。例如，可以按以下方式划分 SoundBank：

- 一个通用 SoundBank，其中包含在游戏任何时刻都可能发生的所有 Event。此 SoundBank 将一直保持载入内存。
- 每个关卡或环境配备一个 SoundBank。根据主人公实际位置变化而需要用到的声音和振动。
- 以及一些额外 SoundBank，具体取决于特定游戏的需求。

在 Wwise 中创建多个完整 SoundBank 的方法是：

1. 创建游戏所需的多个 SoundBank，并正确地为它们命名，例如“CommonEvent”、“Level_1”、“Level_2”和“Level_3”。
2. 在 Wwise 中把事件分组到虚文件夹中。为每个 SoundBank 创建一个虚文件夹，然后把每个文件夹拖到相应 SoundBank 中。通过文件夹来添加 SoundBank 内容，就

可以避免每次工程中加了新事件时都必须编辑 SoundBank 内容的情况。当虚文件夹中的内容改变时，SoundBank 会自动更新。

3. 将所有 Event 添加到各自的 SoundBank。只有在添加原始文件夹之外的 Event 时（如果有的话）才需要此步骤。如果多个 SoundBank 中需要某个 Event，则只需将它添加到必需的 SoundBank 即可。
4. 生成 SoundBank，并将生成的 SoundBank 文件夹复制到游戏应用程序。

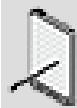
将多个完整的 SoundBank 集成到游戏中的方法是：

1. 在游戏中，只需在正确的时间加载正确的 SoundBank 即可。例如，游戏可以在开始时加载通用 SoundBank，然后在游戏中根据玩家的实际位置加载其它 SoundBank。注意，某些游戏需要具有足够的内存，以便一次加载多个“关卡”，实现关卡之间的过渡。

以下代码示例可以让您了解到在游戏中如何加载这些 SoundBank。

```
// #####
...
// ##### SoundBank#
AkBankID bankID; // #####
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 );
}

...
// #####
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" );
```



备注

作为 LoadBank() 命令的替代方法，您可以使用 AkBankContent_All 命令来做 Prepare SoundBank。对 SoundBank 做 Prepare 的优点是避免内存中可能出现媒体重复。请参阅 Wwise SDK 文档中的“Prepare SoundBank”一节。

有关此方法的补充说明

下表列举了“多个完整的”SoundBank 方法的优点和缺点。

优点	缺点
需要的内存比“一体化”SoundBank 技术少得多。 非常易于集成到游戏中。	不太适合在线游戏或基于 Event 的游戏，在这些游戏中，音频或振动需求不仅仅由主人公的位置等简单因素来驱动。 由于 SoundBank 中可能包含重复数据，因此可能导致内存中重复加载某些媒体文件。 由于不同的 SoundBank 可能含有相似的内容，因此可能增大 SoundBank 的总体磁盘占用空间。

方法 3：巨细无遗地管理您的媒体

此方法适用于以下情形：

- 游戏拥有大量驻留内存的媒体素材。
- 声音设计师很难事先预测媒体需求。
- 工程使用切换开关和状态来确定将为特定 Event 播放的声音或振动对象。
- 声音不易于划分成多个确定的部分。

有的游戏非常复杂，可能要按多种因素来触发声音，这些因素包括游戏纹理、时辰、游戏对象的运动，在多人在线游戏中，还可能包括其它玩家的操作。例如在基于 Event 或对象的环境中，声音可以根据其它一些游戏对象的距离加载到内存中。各个游戏对象都可能有一列 SoundBanks 必须加载，要么因为这些 SoundBank 对应的声源处于一个给定距离范围内，要么仅仅因为按规则只要这些声音存在就必须使用。

另外，切换开关和状态可以确定播放对象。当将播放这些对象的 Event 添加到 SoundBank 时，所有可能用到的媒体也会自动添加到 SoundBank。例如，您可能有一个被称为“Play_Footstep”的 Event。此 Event 将根据地面的当前纹理播放相应的声音，当前纹理通过更改切换开关来指定。虽然这种方法效果非常好，但是当游戏场景发生在伦敦的一座建筑内时，如果在内存中将声音“footstep_sand.wav”或“footstep_winter.wav”保持随时就绪状态是对内存的一种浪费。

在此情况下，为了避免浪费内存，您可以将 Event 或对象结构添加到 SoundBank，然后指定 SoundBank 中将要包含的相应用对象。例如，让我们对不同的纹理使用不同的脚步声。<XXXX CSOFT xxxxx>在此情况下，我们可以创建以下 SoundBank：

- “EventBank”，其中包含“Play_Footstep”Event 和结构。
- “Winter_Footstep_bank”，其中包含仅在游戏某个部分（冬天）发生脚步声所对应的媒体。
- “Desert_Footstep_bank”，其中包含仅在游戏某个部分（沙漠）发生脚步声所对应的媒体。
- “Common_Footstep_bank”，其中包含可在任何地方（木材、混凝土等）发生脚步声所对应的媒体。

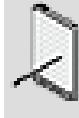
在 Wwise 中创建 SoundBank 来细致地管理媒体

假设游戏中有三种不同的纹理（雪地、沙地和混凝土）。在 Wwise 中，有一个切换容器根据切换开关“ground_texture”播放三个随机容器中的一个。三个随机容器中的每一个在给定纹理上均有四种脚步声版本。

在 Wwise 中创建用于微管理媒体的 SoundBank 的方法是：

1. 创建名为“EventBank”的 SoundBank，然后将它加载到 SoundBank Editor。
2. 将“Play_Footstep”Event 拖入 SoundBank Editor 的“Add”选项卡。
3. 取消选择此 Event 的“Media”复选框，但保持选中 Events 和 Structures 复选框。
4. 创建其它三个 SoundBank，每个纹理各一个。

- 在各个 SoundBank 中，拖动与此纹理相关的随机容器。



备注

如果不拖放随机容器，则分别拖放各个声音也同样有效。但是使用容器的优势是，当容器的内容改变时，容器中的所有声音将自动添加到 SoundBank 中，无需手动完成所有更改。

- 取消选择三个纹理 SoundBank 中的每一个所对应的“Event”和“Structure”复选框，仅保持选中 Media 类型。
- 生成 SoundBank，并将生成的 SoundBank 文件夹复制到游戏应用程序。

此时，我们有四个 SoundBank，一个包含与待播放的音频相关的 Event 和结构数据，另外三个仅包含与特定地面纹理相关的媒体。

使用微管理策略将 SoundBank 集成到游戏中的方法是：

- 在游戏开始时加载通用 SoundBank，然后在需要时加载其它 SoundBank。例如，游戏可以在开始时加载 Event SoundBank 和通用脚步声 SoundBank，然后根据玩家在游戏中的具体位置加载其它 SoundBank。

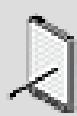
以下代码示例可以让您了解到在游戏中如何加载这些 SoundBank。

```
// ##### Eventbank#
AkBankID bankID; // #####
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 );
}

...
// #####
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" );
```

有关此方法的补充说明

这个例子只是使用此技术可以完成的众多任务之一。由于可以按对象和 Event 逐一决定各个 SoundBank 包含的内容，因此您可以全面控制各个 SoundBank 的内容。虽然您可以为游戏中的每个对象创建单独的 SoundBank，但是这样维护起来会非常困难，因为在游戏中每个新音效和振动效果都需要新的代码才能正确加载 SoundBank。每款游戏应该尝试的目标是在粒度和游戏集成便利性之间找到良好的平衡。



备注

如果您对声音逐一加载方案感兴趣，则可 Prepare Event，而无需加载整个 SoundBank。有关做 Prepare Event 的详细信息，请参阅[方法 4：Prepare 动作事件](#)。

下表列出了微管理策略在管理媒体时的优缺点。

优点	缺点
优化内存占用的最佳选择。	需要声音设计师和游戏程序员密切沟通，共同确定必须加载的 SoundBank 和加载的时机。
您可以充分控制在游戏任意时刻加载哪些媒体。	
为您提供 SoundBank 向后兼容性，支持加载较大的媒体 SoundBank。	

方法 4: Prepare 动作事件

此方法适用于以下情形：

- 要保持较低的内存占用，需要媒体具有很小的粒度级别。
- 不用为管理哪些媒体素材必须指派给哪个 SoundBank 而烦恼。

什么是准备就绪的 Action Event（动作事件）？在调用 `PrepareEvent()` 函数时，系统将分析 Action Event（动作事件），确保与此 Event 相关的所有结构和媒体都已加载到内存中。否则，系统会自动从磁盘流式传入缺失的信息。在明确撤消以前，Event 一直处于 Prepare 状态。



备注

只有动作 Event 可事先 Prepare。“PrepareEvent()”方法不适用于对话事件。

此方法要求创建至少一个 SoundBank，然而结构部件可以位于与 Event 所在的同一个 SoundBank 中，也可处于完全独立的一个 SoundBank 中。

在构建使用 PrepareEvent 机制的 SoundBank 时，条件是至少在一个 SoundBank 中能够找到所需的所有 Event 和结构，而且文件系统必须能够访问这些松散媒体素材。记住，将结构划分到多个 SoundBank 中可以提高内存的管理效率。

在 Prepare 动作 Event 前，Event 本身必须（使用 `LoadBank()`）从一个 SoundBank 加载到内存中。由于 Event 中包含在 Prepare Event 时所需的依赖项的信息，因此必须首先加载 Event 所在的 SoundBank。



备注

另外还可结合 `AK::SoundEngine::PrepareBank` 来 Prepare Event。使用 `PrepareBank` 机制的主要优势是无需将 SoundBank 划分为“Event 库”和“媒体库”。在这种方法下，所有内容包含在同一个 SoundBank 中，但是在调用 `AK::SoundEngine::PrepareBank` 时，只有 SoundBank 的元数据内容加载到内存。当游戏需要媒体时，您可以使用 `PrepareEvent` 加载媒体。

在 Prepare Event 时在 Wwise 中设置 SoundBank 的方法是：

- 创建一个名为“Event”的 SoundBank，并加载到 SoundBank Editor 中。
- 把工程中的一些动作事件添加到“Events” SoundBank 中，或仅添加事件工作单元。
- 禁用 **Media** 复选框，只将 **Events** 和 **Structures** 复选框保留为启用状态。在使用 `PrepareEvent()` 时，媒体不得位于 SoundBank 内，但可以作为松散文件直接从磁盘中加载。



备注

在本例中，所有 Event 和结构包含在一个 SoundBank 中。虽然对于小型工程，这是可以接受的，但您很可能会需要将内容拆分到多个 SoundBank。另外还可以创建独立

的“Structures”SoundBank，它无需从 SDK 命令行进行显式加载或 Prepare，因为各个 Event 都包含对需要加载的其它 SoundBank 的引用。

4. 生成 SoundBank，并将生成的 SoundBank 文件夹复制到游戏应用程序。



备注

单个 SoundBank 中包含的结构数据不可在运行时拆分。因此，如果您使用了 AK::SoundEngine::PrepareEvent()，并且需要来自独立 SoundBank 的结构数据，则该 SoundBank 中的所有结构将一次性加载。为此，您可能会需要将工程中的结构内容拆分到多个 SoundBank，最大限度地减少加载到内存中的非必需信息量。

在游戏中 Prepare Event 的方法是：

1. 在游戏中，先在游戏开始时加载 Event 的 SoundBank，然后再在游戏中需要时来 Prepare Event。相应的结构和媒体将自动加载。

下面的代码示例可以让您了解在游戏中如何 Prepare Event。

```
// #####
AkInitSettings initSettings;
AkPlatformInitSettings platformInitSettings;
AK::SoundEngine::GetDefaultInitSettings( initSettings );
AK::SoundEngine::GetDefaultPlatformInitSettings( platformInitSettings );

// #####
...
// ## PrepareEvent #####
initSettings.bEnableGameSyncPreparation = false; // #####
// ##### Prepare #####
// #####
{
    initSettings.uPrepareEventMemoryPoolID = AK::MemoryMgr.CreatePool( NULL, 4*1024*1024, 1024, AkMalloc );
    // #####
    AK::MemoryMgr::SetPoolName( initSettings.uPrepareEventMemoryPoolID, L"PrepareEventPool" );
}

AKRESULT eResult = AK::SoundEngine.Init( initSettings, platformInitSettings );
if( eResult != AK_Success )
{
    // #####
}

// #####/#####
AkBankID bankID; // #####
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 );
}
...

// #####
AkLpCtstr pEventsNameArray[1] = { L"My_Event_Name" };

// Prepare ##
eResult = AK::SoundEngine::PrepareEvent( Preparation_Load, pEventsNameArray, 1 ); // 1 #####
// ##### Prepare ##
eResult = AK::SoundEngine::PrepareEvent( Preparation_Unload, pEventsNameArray, 1 ); // 1 #####

```

有关此方法的补充说明

记住，对 `AK::SoundEngine::PrepareEvent()` 的调用必须视为 I/O 函数调用。在上例中，我们使用了阻塞函数。您可以使用

AK::SoundEngine::PrepareEvent() 函数的其它重载来实现非阻塞调用，然后用另一个回调来响应加载完成的通知。

下表列举了对 Action Event（动作事件）做 Prepare 的优点和缺点。

优点	缺点
生成 SoundBank 的过程很简单。	当逐一加载媒体素材时，可能增加磁盘上的读取和寻址次数。
媒体的粒度级别非常小。	对总体内存占用量的控制力减弱。
内存占用总体保持在低水平。	使用交互式音乐时将变得复杂。
过程自动化非常容易。	

方法 5：对 Event 和 Game Sync（切换开关和状态）做 Prepare

此方法适用于以下情形：

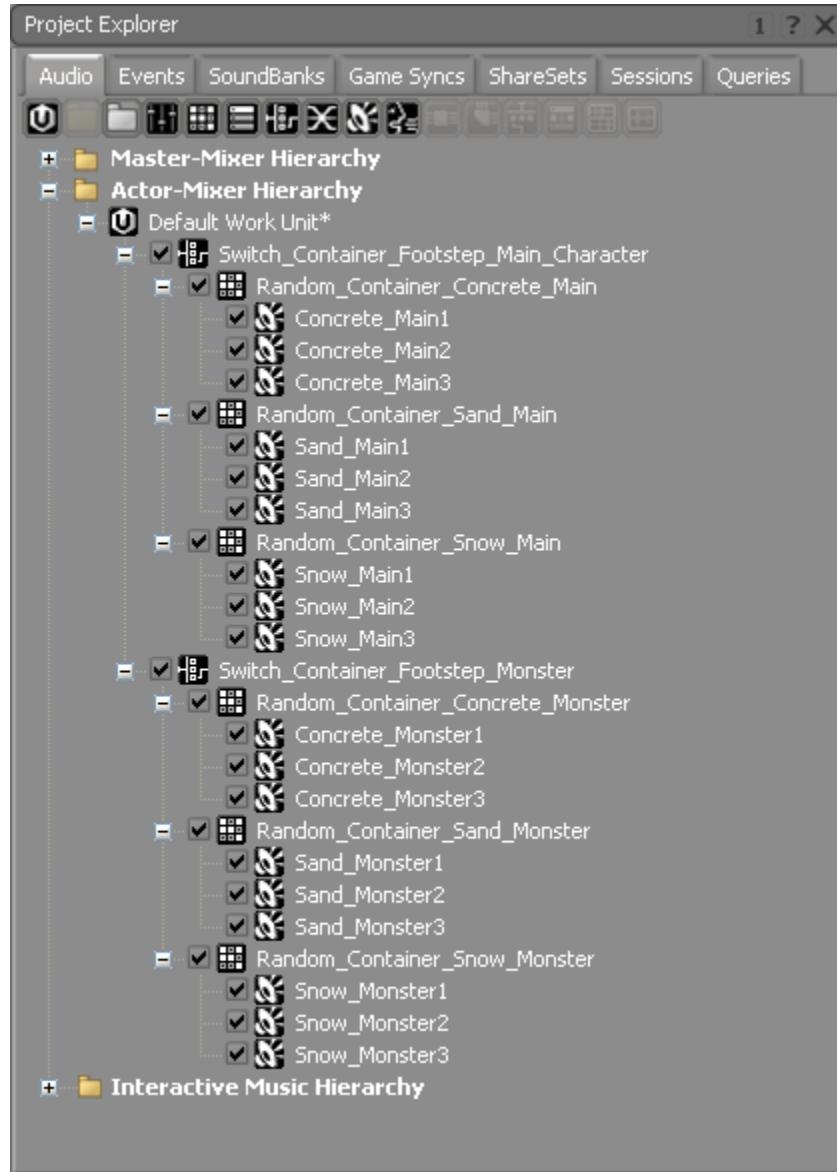
- 需要媒体具有较小的粒度级别以将内存占用保持在低水平。
- 不想为将媒体素材拆分到多个 SoundBank 而烦恼。
- 工程中的 Event 根据切换开关和状态播放不同的声音。
- 工程中包含根据切换开关和状态播放音乐的互动音乐。

此方法基本与上一种方法相同，但对 Event 做 Prepare 时加载的媒体具有更大的控制力。使用这种方法时，只有 Prepare 的事件和当前活跃的 Game Sync 同时相关的媒体才会加载到内存中。

假设一个简单的工程中有两个 Event

：“Play_Maincharacter_FootSteps”和“Play_Monster_Footsteps”。各个 Event 播放不同的切换容器，后者根据角色在行动时脚下的地面纹理来播放不同的随机声音。切换开关组名称为“GroundTexture”，拥有三个可能的状态：“Snow”、“Concrete”和“Sand”。

在 Wwise 中，切换开关容器层级结构如下所示：



在此例中，18 个声音（3 个声音一组，共 6 组）可能加载到内存中。您可以使用 [方法 4：Prepare 动作事件](#)，但粒度级别不会小于每个 Event 加载 6 个声音到内存中。

您可以使用 [方法 3：巨细无遗地管理您的媒体](#) 获得更高的粒度级别，但必须为这一简单工程创建 6 个不同的 SoundBank（在实际工作场合下的工程中，SoundBank 的数量将迅速增加）。于是当怪兽出现时，您必须查看可能用到的纹理，然后再加载相应的 SoundBank。

在当前方法中，您只需指定可能的 Event 和 Game Sync，然后只有合适的媒体才会加载。为了简便起见，可以将所有媒体集中到一个 SoundBank 中。

在对 Event 和 Game Sync 做 Prepare 时设置 SoundBank 的方法是：

1. 创建名为“Events”的 SoundBank，并加载到 SoundBank Editor。
2. 将两个 Event 拖到 SoundBank Editor 的 Add 选项卡。

3. 禁用 **Media** 复选框，只将 **Events** 和 **Structures** 复选框保留为启用状态。在使用 **PrepareEvent()** 时，媒体不得位于 SoundBank 内，但可以作为松散文件直接从磁盘中加载。

**备注**

在本例中，所有 Event 和结构包含在一个 SoundBank 中。虽然对于小型工程，这是可以接受的，但您很可能会需要将内容拆分到多个 SoundBank。另外还可以创建独立的“Structures”SoundBank，它无需从 SDK 命令行进行显式加载或 Prepare，因为各个 Event 都包含对需要加载的其它 SoundBank 的引用。

4. 生成 SoundBank，并将生成的 SoundBank 文件夹复制到游戏应用程序。

**备注**

单个 SoundBank 中包含的结构数据不可在运行时拆分。因此，如果使用 **AK::SoundEngine::PrepareEvent()**，并且需要来自单独 SoundBank 的结构数据，则该 SoundBank 中的所有结构将一次性加载。为此，您可能会需要将工程中的结构内容拆分到多个 SoundBank，最大限度地减少加载到内存中的非必需信息量。

在游戏中对 Event 和 Game Sync 做 Prepare 的方法是：

1. 在游戏需要两个 Event 前加载“Event” SoundBank。
2. 让 Concrete 纹理一直保持激活状态。
3. 始终都要 Prepare 主人公脚步 Event。
4. 当可能的地面纹理在附近时，激活 Game Sync。

下面的代码示例可以让您了解在游戏中如何对 Game Sync 做 Prepare。首先，您需要初始化声音引擎，并设置必要的设置，包括 **bEnableGameSyncPreparation** 标志。

```
// #####
AkInitSettings initSettings;
AkPlatformInitSettings platformInitSettings;
AK::SoundEngine::GetDefaultInitSettings( initSettings );
AK::SoundEngine::GetDefaultPlatformInitSettings( platformInitSettings );

// #####
...
// ## PrepareEvent #####
// bEnableGameSyncPreparation ##### true##### Preapare
// Game Sync ##### false ##### Game Sync #####
// ##### AK::SoundEngine:PrepareGameSyncs#
//
// ### true ##Game Sync #####
// ### AK::SoundEngine:PrepareGameSyncs ## Game Sync#
initSettings.bEnableGameSyncPreparation = true;

// ##### Prepare #####
// #####
{
    initSettings.uPrepareEventMemoryPoolID = AK::MemoryMgr.CreatePool( NULL, 4*1024*1024, 1024, AkMalloc );

    // #####
    AK::MemoryMgr::SetPoolName( initSettings.uPrepareEventMemoryPoolID, L"PrepareEventPool" );
}

AKRESULT eResult = AK::SoundEngine::Init( initSettings, platformInitSettings );
if( eResult != AK_Success )
{
    // #####
}
```

然后您可以加载初始化库和 Event 的 SoundBank。

```
// #####/#####
AkBankID bankID; // #####
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 );
}

// ... ##### ##### Prepare#
// #####
```

现在您可以 Prepare 所需的 Event 和 Game Sync。

```
const AkOSChar** pNameArray = new const AkOSChar*[1];

// Prepare #####
pNameArray[0] = L"Play_Maincharacter_FootSteps";
eResult = AK::SoundEngine::PrepareEvent(AK::SoundEngine::Preparation_Load, pNameArray, 1); // 1 is the array size.

// ... ### Prepare ### Event#####

// ##### Concrete#
pNameArray[0] = L"Concrete";
eResult = AK::SoundEngine::PrepareGameSyncs(AK::SoundEngine::Preparation_Load, AkGroupType::AkGroupType_Switch,
L"GroundTexture", pNameArray, 1 );

// ... ###3 ####Sound_Concrete_main_1#Sound_Concrete_main_2 # Sound_Concrete_main_3#
// #####
// #####
// #####
pNameArray[0] = L"Snow";
eResult = AK::SoundEngine::PrepareGameSyncs(AK::SoundEngine::Preparation_Load, AkGroupType::AkGroupType_Switch,
L"GroundTexture", pNameArray, 1 );

// ... ###3 ####Sound_Snow_main_1#Sound_Snow_main_2 # Sound_Snow_main_3#
// #####
// #####
// #####
pNameArray[0] = L"Play_Monster_Footsteps";
eResult = AK::SoundEngine::PrepareEvent( AK::SoundEngine::Preparation_Load, pNameArray, 1 ); // 1 is the array size.

// ... ##### 6 ####Sound_Concrete_Monster_1.2.3 # Sound_Snow_Monster_1.2.3#
// #####
// #####
// #####
pNameArray[0] = L"Snow";
eResult = AK::SoundEngine::PrepareGameSyncs( AK::SoundEngine::Preparation_Unload, AkGroupType::AkGroupType_Switch,
L"GroundTexture", pNameArray, 1 );
// ... ### 6 #####Sound_Snow_Monster_1.2.3 # Sound_Snow_main_1.2.3#
// #####
// #####
...
```

有关此方法的补充说明

下表列举了对 Game Sync 做 Prepare 的优点和缺点。

优点	缺点
声音包生成过程简单。	当逐一加载媒体素材时，可能增加磁盘上的读取和寻址次数。
媒体的粒度级别非常小。	对总体内存占用量的控制力减弱。
保持较低的总体内存占用。	在大量 Event 采用 Prepare 因而需要加载新数据时，激活 Game Sync 可能导致流播放带宽升高。
易于实现过程自动化。	
只有加载有用的媒体。	

SoundBank 技巧和经验总结

在定义 SoundBank 的内容前，您最好仔细阅读以下各节。其中提供了一系列的技巧和经验总结，可以帮助您更好地管理游戏中的 SoundBank。

更新初始化库

每次生成若干个 SoundBank 时，Wwise 会根据需要更新初始化库。这样做时为了确保两套声音包同步。初始化库包含有关工程的所有通用信息（包括有关总线层级结构的信息）以及有关状态、切换开关和 RTPC 的信息。如果更改工程中的任何这些元素，Wwise 则将随着新生成的 SoundBank 为您的游戏更新此声音包。然而，如果没有更改这些工程元素，Wwise 则将不会生成 Init.bnk 的新版本。

Initialization Bank —— 一种特殊库，其中包含有关工程的所有通用信息（包括有关总线层级结构的信息）以及有关状态、切换开关、RTPC 和环境效果器的信息。每次 Wwise 生成 SoundBank 都会自动生成 Initialization Bank。Initialization Bank 通常在游戏开始时加载，以便在游戏期间可以轻松获得工程的所有通用信息。它必须是在启动游戏时最先加载的库；否则其它库可能无法加载。在默认情况下，Initialization Bank 命名为“Init.bnk”。

SoundBank 和内存

如果在 SoundBank 中加载长声音，它则可能占用大量平台内存。为避免占用更多的内存，应尝试采用流播放长声音或音乐文件。有关如何流播放声音和音乐对象的信息，请参阅[对声音采用流播放或流播放音乐](#)。

将通用元素合并到 SoundBank 中

将通用元素全部集中到一个 SoundBank 中是一个好方法。例如，在整个游戏期间一直处于加载状态的所有元素（例如菜单、主人公等）应合并到一个 SoundBank 中。对于在玩游戏的过程中不断加载和卸载的元素，应将它们合并到在整个游戏期间可以交换的逻辑单元或构件中去。

使用工作单元和文件夹

为避免每次更改工程时必须编辑 SoundBank，您可以使用文件夹和 / 或工作单元重新创建设置 SoundBank 的方式。由于 Wwise 在 SoundBank 元素与工程元素之间保持着有效的链接，因此这些文件夹一旦添加到 SoundBank，您就无需再编辑这些 SoundBank 了，它们会自动更新。

在 SoundBank 中使用 Event ID

在工程的最后阶段，使用 Event ID 而不是 Event 字符串是一个好办法，因为在继续进行前，声音引擎无需对名称执行哈希算法，Event ID 验证起来会更快。

生成完好度报告

在生成 SoundBank 前生成完好度报告（Integrity Report）是一个好主意。完好度报告能展示工程中的错误列表，以及可能的解决方法。通过在生成 SoundBank 前解决所有工程错误，您可以尽可能地减少游戏中的音频或振动问题。

在 SoundBank Editor 中使用多选功能

如果正在使用 SoundBank Editor 的 Add 选项卡，并且需要为多个条目弃用同一类型的工程元素，则您可以在 Hierarchy Inclusion 列表中选择多项，然后取消勾选其中一个复选框。这将为所有选定项弃用该类型。您还可以使用多选功能为多个项来添加类型。

SoundBank 名称限制

要规避平台文件系统对文件名的限制，可以将 SoundBank 打成文件包。文件包是一个独立单元，它将平台的文件系统完全分离出来，消除了系统可能对文件名的任何限制。

DLC 注意事项及限制

为了让 DLC 中的声音结构符合要求，您必须了解在已有结构基础上进行更改和添加将产生哪些影响。有一些常见情况需要特别注意，下面将详细介绍。

添加子项

对于一些特定对象，在 DLC 中为其添加子项时，必须重新打包现有的 SoundBank 并包含在 DLC 中；但对一些其它对象，则可以通过新增 SoundBank 在 DLC 中为其添加子项。下表列出了需要重新打包 SoundBank 来添加子项的对象，以及可以在 DLC 中新增 SoundBank 来添加子项的对象。

需重新打包 SoundBank 才能将子项添加到 DLC 的对象	可使用新增 SoundBank 将子项添加到 DLC 的对象
<ul style="list-style-type: none"> • Random/Sequence Container (随机/序列容器) • Switch Container (切换容器) • Blend Container (混合容器) • Music Switch Container (音乐切换容器) • Music Playlist Container (音乐播放列表容器) • Music Segment (音乐段落) 	<ul style="list-style-type: none"> • Actor-Mixer (角色混音器) • Folder • Work Unit (工作单元)

例如，假设有以下层级结构：

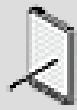
- WorkUnit
 - ActorMixer
 - ContainerA
 - Sound1
 - Sound2

如果您将 ContainerA 放在主版本的 SoundBank1 中，那么对于 DLC，可以顺利地在 Actor-Mixer Hierarchy 内添加新容器。

- WorkUnit
 - ActorMixer
 - ContainerA
 - Sound1
 - Sound2

- ContainerB
 - Sound1
 - Sound2

随后您可以创建一个新的 SoundBank2 并添加 ContainerB，加载 SoundBank1 和 SoundBank2 时，其中的结构将自动合并。但 DLC 中，不能在 ContainerA 下添加声音，并将其加入新的 SoundBank 中进行打包。如果必须添加，则必须重新生成 SoundBank1 并包含在 DLC 中，它将替换主发布版本中的 SoundBank。



备注

将 Event 和 Structure 与 Media 文件分离，这对计划发布 DLC 的游戏来说是一个很好的策略，因为重新打包 SoundBank 时，不会导致 DLC 容量剧增。

修改 Init Bank

对于 DLC，不管以何种方式修改了总线、游戏同步器（游戏参数，切换开关和状态）或环境效果后，都必须将新的 Init.bnk 文件重新打包到 DLC 中。

发布多个 DLC

在同时或依次发布多个 DLC 时需特别小心，持续更新内容时，必须基于之前的 DLC 工程来构建新的 DLC 版本。

例如，假设您发布了主版本（包括 Init.bnk + RTM.bnk）、DLC-A（包括 Init.bnk + DLC-A-specific.bnk）和 DLC-B（包括 Init.bnk + DLC-B-specific.bnk）。如果 RTM、DLC-A 和 DLC-B 中总线结构不同，则 DLC-A 中的新增总线也必须位于 DLC-B 内。

3 个版本都需要包括含有总线的 Init.bnk 文件，因为用户可能：

- 没有 DLC
- 仅有 DLC-A
- 仅有 DLC-B
- 有 DLC-A 和 DLC-B

当存在 DLC-A 和 DLC-B 时，将加载 DLC-B 中的 Init.bnk。由于 DLC-B 中的 Init.bnk 是基于 DLC-A 构建的，因此 DLC-A 所需的总线也将可用。

SoundBank 向后兼容性

在 Wwise 同一大版本下的不同小版本之间，SoundBank（音频包）是完全兼容的。比如，在 2017.1.9 版本的 Wwise 中，您可以正常加载并使用 2017.1.2 生成的 SoundBank。不过，在不同大版本的 Wwise 之间，SoundBank 可能无法兼容。比如，在 2018.1 版本的 Wwise 中，一般无法正常加载并使用 2017.1.2 生成的 SoundBank。您必须在 2018.1 中重新生成 SoundBank。

不过，这一通用规则也有例外情况。若只包含媒体信息（非 Event 或结构信息），则 2016.1 及更高版本中生成的 SoundBank 可与 Wwise 2018.1、Wwise 2017.2.7 及后续版本兼容。注意，这可能不适用于插件媒体。

对于发行时间较长的游戏，会涉及 Wwise 版本更新，那么声音设计师应该考虑使用这种[仅含媒体的 SoundBank](#)，这样升级时只需要更新含有结构和 Event 的音频包（容量要小很多）。

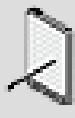
第 36 章 管理 File Package

概述	677
使用 File Packager 工程	677
管理工程内的 File Package	679
可下载内容 (DLC) 概述	683
生成 File Package	684
在命令行中使用 File Packager 参数	684
File Packager 技巧和最佳实践	687

概述

File Packager（文件打包器）是独立的应用程序，可以将 Wwise 生成的 SoundBank（声音包）、零散媒体以及流播放文件分组，并放到若干个文件包（File Package）中。文件包是将文件系统抽象化之后的独立单元，加载文件包会在内存中创建查找表，它会为 Wwise 指明媒体在 PCK 文件中的位置。您可以通过使用文件包来规避平台文件系统的一些限制，包括文件名长度和文件数量限制。文件包还可帮您更好地管理多语言版本，以及发布后可供下载的内容（DLC）。

通过将 SoundBanksInfo.xml 文件导入 File Packager 来进行检索，可以获取 Wwise 工程的 SoundBank 和流文件的所有信息。



生成 SoundbanksInfo.xml 文件

在 Project Settings 或 SoundBanks Settings 的 SoundBanks 选项卡下，须始终启用 Generate Metadata Files 选项。同时，还要启用 Generate XML Metadata 选项。利用这些设置，Wwise 可在每次成功生成 SoundBank 时自动生成 SoundbanksInfo.xml 文件。

默认情况下，所有文件都会添加到 default.pck 文件包中，该文件包将生成至各平台 SoundBank 路径的根目录下。也可以创建新的文件包，手动将文件添加至其中并修改文件包的储存位置。

您可以使用 File Packager 手动创建文件包，或者在 SoundBank 生成时使用命令行运行 File Packager 来自动创建。该命令行可以在工程级别设置，也可以在 SoundBank 的 User Setting 中自定义。关于 SoundBank 生成后通过命令行启动 File Packager 的详细信息，请参阅以下章节：

- [定义工程的 SoundBank 设置](#)
- [定义 SoundBank 的自定义属性](#)
- [在命令行中使用 File Packager 参数](#)

使用 File Packager 工程

File Packager 中创建的每个会话都可以保存为工程，以便随时存储和打开。File Packager 工程包含文件包的所有信息，包括要创建文件包的数量、文件指派设置以及各文件包的内容和文件顺序。

每次打开 File Packager 将自动创建新的会话，可以将其另存为新的工程，或打开您之前保存的工程。

打开 File Packager 的方法如下：

1. 在 Start（开始）菜单中，点击 All Programs > Audiokinetic > Wwise 20xx.x build xxxx > Tools > File Packager（所有程序 > Audiokinetic > Wwise 20xx.x build xxxx > 工具 > File Packager）。

File Packager 将打开，并创建一个新的会话。

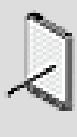
相关主题

- [将 SoundBank 信息导入工程](#)
- [打开现有 File Packager 工程](#)

- [保存 File Packager 工程](#)

将 SoundBank 信息导入工程

在开始创建文件包之前，您需要导入之前在 Wwise 中生成 SoundBank 时创建的数据。存储这些数据的 SoundBanksInfo.xml 文件位于 GeneratedSoundBanks 目录下的各平台文件夹中。Wwise 会为每个平台生成不同的 SoundBanksInfo.xml 文件。



生成 SoundbanksInfo.xml 文件

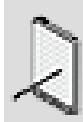
在 Project Settings 或 SoundBanks Settings 的 SoundBanks 选项卡下，须始终启用 Generate Metadata Files 选项。同时，还要启用 Generate XML Metadata 选项。利用这些设置，Wwise 可在每次成功生成 SoundBank 时自动生成 SoundbanksInfo.xml 文件。

每次打开工程时，File Packager 将自动读取 SoundBanksInfo.xml 文件并显示在 Files to package（要打包的文件）区域中。如果在 Wwise 中对 SoundBank 和流文件进行了更改，可能导致两个工程不一致。如果有文件缺失，会被标记为 missing 并显示为红色。

将 SoundBank 信息导入工程的方法如下：

1. 打开 File Packager。
2. 点击 SoundBanks Info file（SoundBank 信息文件）旁边的 Browse（浏览）按钮
Windows 系统的 Open（打开）对话框将会打开。
3. 浏览至 SoundBanksInfo.xml 文件位置，选择该文件并点击 Open（打开）。

所有 SoundBank 和流文件将被加载到 Files to package 视图中。



备注

默认情况下，所有 SoundBank 和流文件都将添加至 Default.pck 文件包。

相关主题

- [使用 File Packager 工程](#)
- [打开现有 File Packager 工程](#)
- [保存 File Packager 工程](#)

打开现有 File Packager 工程

File Packager 工程保存后，可以随时打开并继续处理。允许打开多个 File Packager 窗口，以便您同时处理多个工程。

打开现有 File Packager 工程的方法如下：

1. 打开 File Packager。
2. 从菜单栏中，点击 File > Open。

Windows 系统的 Open（打开）对话框将会打开。

3. 浏览至 File Packager 工程所在的文件夹。

4. 选择 .wfpproj 文件并点击 **Open**。

工程信息将被加载到 File Packager 中。

相关主题

- [使用 File Packager 工程](#)
- [将 SoundBank 信息导入工程](#)
- [保存 File Packager 工程](#)

保存 File Packager 工程

File Packager 中创建的每个会话都可以保存为工程，以便随时存储和打开。File Packager 工程会被存储为 WFPPROJ 文件。

保存 File Packager 工程的方法如下：

1. 执行以下操作之一：

- 在菜单栏中，点击 **File > Save**。
- 按 **Ctrl+S** 键。
- 所有工程信息都将包存至 WFPPROJ 文件。



相关主题

- [使用 File Packager 工程](#)
- [将 SoundBank 信息导入工程](#)
- [打开现有 File Packager 工程](#)

管理工程内的 File Package

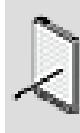
创建 File Packager 工程并将 SoundBank 和流文件信息导入后，就可以开始创建所需的文件包（File Package）并添加文件了。完成后即可生成文件包，以供游戏使用。

管理工程中的文件包，包括以下任务：

- [在工程中添加 File Package](#)
- [从工程中删除 File Package](#)
- [填充 File Package](#)
- [对 File Package 内的文件排序](#)

在工程中添加 File Package

根据需要，您可以在工程中添加任意数量的文件包（File Package）。创建后，可以通过 **Block Size**（块大小）属性来设置文件包的字节对齐，具体大小将取决于各平台 I/O 设备的要求。

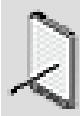


备注

有关 Block Size 和各平台 I/O 设备限制的详细信息, 请参阅 [SDK 文档](#) 的 Streaming > Low-Level I/O (流播放 > 底层 I/O) 部分。

在工程中添加文件包的方法如下:

1. 在 Packages 列表中, 点击 Add (添加)。
新文件包将被添加到 Packages 列表中。
2. 为新文件包命名, 替换默认名称。
3. 按 Enter 键。
4. 重复步骤 1-3, 创建所有文件包。
5. 要设置文件包内进行数据对齐的字节数, 可以在 Block Size 中输入数值, 具体大小将取决于各平台 I/O 设备的要求。每个平台的默认值是 1。



备注

Block Size 值为 1 时, 表示无需对齐。

相关主题

- [从工程中删除 File Package](#)
- [填充 File Package](#)
- [对 File Package 内的文件排序](#)
- [生成 File Package](#)

从工程中删除 File Package

如果您不再需要某个文件包, 可以随时将其从工程中删除。

从工程中删除文件包的方法如下:

1. 从 Packages 列表中, 选择要删除的文件包。
2. 点击 Remove (删除)。

文件包将从工程中删除。

相关主题

- [在工程中添加 File Package](#)
- [填充 File Package](#)
- [对 File Package 内的文件排序](#)
- [生成 File Package](#)

填充 File Package

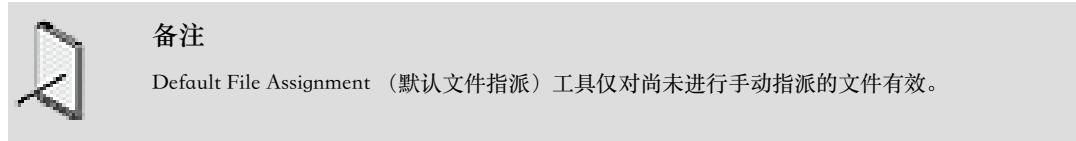
创建文件包 (File Package) 后, 就可以将 SoundBank 和流文件添加到其中了。

填充文件包时, 可以采用两种不同方式:

- [将文件自动指派给 File Package](#)
- [将文件手动添加到 File Package](#)

将文件自动指派给 File Package

将文件手动添加到文件包可能非常耗时，而 File Packager 允许您基于语言或文件类型创建规则，将一组文件自动指派给特定文件包。设置这些规则可以大大缩短填充文件包的用时。



将未指派文件自动添加到文件包中的方法如下：

1. 在 File Packager 的 Default File Assignment 视图中，将列表中各语言/音效类别下的 SoundBank 或流文件指派给特定文件包。
2. 在 Assign remaining files 区域中选择列表项，将所有剩余 SoundBank 指派给该文件包。
3. 选择列表项，将所有剩余流文件指派给该文件包。

所有尚未进行手动指派的文件都将自动指派给所选文件包。

相关主题

- [将文件手动添加到 File Package](#)
- [在工程中添加 File Package](#)
- [从工程中删除 File Package](#)
- [对 File Package 内的文件排序](#)
- [生成 File Package](#)

将文件手动添加到 File Package

某些情况下，例如创建游戏发布后的更新内容时，您可能需要将文件手动添加到文件包中。将文件手动添加到文件包中后，所有自动文件指派规则都将被改写。

手动将 SoundBank 添加到文件包中时，可以选择仅包含 SoundBank、仅包含与 SoundBank 关联的流文件，或者两者都包含。您可以随时切换至 Package Contents (文件包内容) 视图的 Resulting Contents (内容结果) 选项卡，来查看文件包中的实际内容。

将文件手动添加到文件包中的方法如下：

1. 在 Packages 视图中选择一个文件包。

文件包内容将显示在 Package Contents 视图中。

2. 将文件从 Files to package 添加到 Package Contents 视图的 Added Items (已添加项) 选项卡中。

手动添加的文件数量将会显示在 Added Item 选项卡标题栏的括号中。

注意，Resulting content 选项卡的标题栏显示的数字为文件包中的文件总数。两个数字可能不同，例如当添加的 SoundBank 引用了多个流文件时，两个数字就会有出入。

3. 将 SoundBank 添加到文件包时，可以从 Inclusion mode（包含模式）中选择需要包含的文件类型：
 - All files -- 所有文件，包含 SoundBank 文件及其关联的流文件。
 - SoundBank only -- 仅 SoundBank，只包含 SoundBank 文件。
 - Streamed files -- 仅流文件，只包含关联的流文件，而非实际添加的 SoundBank 文件。
4. 切换至 Resulting contents 选项卡查看文件包的实际内容。

相关主题

- [将文件自动指派给 File Package](#)
- [在工程中添加 File Package](#)
- [从工程中删除 File Package](#)
- [对 File Package 内的文件排序](#)
- [生成 File Package](#)

对 File Package 内的文件排序

默认情况下，所有文件都将以非特定顺序添加到文件包中。但您可能会需要某些文件位置相邻，从而减少游戏过程中的磁盘寻址时间。使用 File Order Editor（文件顺序编辑器）可以排列文件包内的文件。

由于对包内的所有文件排序费时费力且没有必要，所以 File Packager 允许仅针对需要的文件排序。通过“Remaining files inserted here（此处插入剩余文件）”占位符，可以将所有剩余文件自动添加到顺序列表末尾。将文件手动添加至顺序列表时，该文件会自动从占位符中删除。

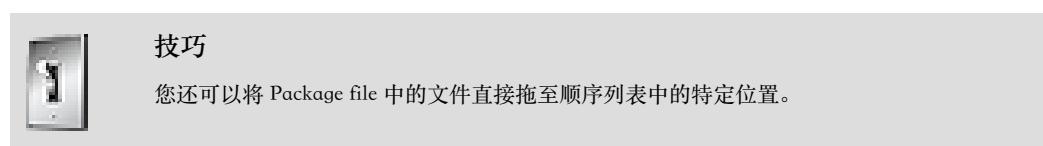
为文件包中的文件排序的方法如下：

1. 执行以下操作之一：
 - 双击 Packages 列表中的文件包。
 - 从 Packages 列表中选择一个文件包并点击 Edit file order（编辑文件顺序）。

File Order Editor 将打开。

2. 在 Package files（包内文件）中，选择要手动排序的文件。
3. 点击 Add to ordered list（添加到顺序列表）按钮。

文件会将添加至顺序列表中的空位。



4. 要给某个文件重新排序，将其拖至列表中的新位置即可。

插入点会用水平线标示。

- 继续排列并确定包内文件的最终顺序。



技巧

要从 File order (文件顺序) 列表中删除文件，选择这些文件并点击 Remove from ordered list (从顺序列表中删除) 即可。

相关主题

- [在工程中添加 File Package](#)
- [从工程中删除 File Package](#)
- [填充 File Package](#)
- [生成 File Package](#)

可下载内容 (DLC) 概述

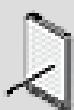
可下载内容 (DLC) 通常是在主版本（初始游戏版本，不含可下载内容）发布后，为游戏添加的新增音频内容。

在游戏的制作阶段，由 Wwise 生成的 SoundBank 和流文件可以通过 File Packager 的 Default file assignment 打包至文件包，并可以按语言种类拆分，或按流文件、零散媒体和 SoundBank 类型进行拆分。

但游戏发布后，主版本中的内容将不再使用 Default file assignment 打包。需要将所有内容手动加入文件包：

- 需手动创建主文件包
- 请将所有内容拖至主文件包

这些手动创建的文件包将包含随主版本一起发布的所有 SoundBank 和流文件。这将作为创建新内容的起始参考点，接下来请保存File Packager 工程。



备注

DLC 内容须使用主版本的 Wwise 工程进行创建，且须使用同一 Wwise 版本，来确保 SoundBank 和文件包格式兼容。

在 File Packager 中创建 DLC 的方法如下：

- 针对各 DLC，手动创建 File Packager 文件包。
- 在 File Packager 中找到新增内容。
- 将新增内容拖至手动创建的文件包（即 DLC）：

随后可以修改主版本中的内容。修改已有 SoundBank 或流文件后，须将其重新打包至新增文件包中。游戏将确保这些文件包按时间顺序加载，即先加载主版本文件包，然后是新文件包 (DLC)。同一对象有不同版本时，Wwise 会优先加载最新版本。

示例

- SoundBankA 位于主版本内，并打包在 Package #1 中。

- SoundBankA 的内容在主版本发布后进行了修改，并被打包到新的 Package #2 中，成为 DLC 的一部分。
- 包含 DLC 的游戏将先加载 Package #1，然后加载 Package #2。
- 游戏加载 SoundBankA 时，Wwise 将加载在 Package #2 中找到的 SoundBankA（其优先级高于 Package #1 中的相应 SoundBank）。

相关主题

- [DLC 注意事项及限制](#)

生成 File Package

在工程中创建文件包、添加文件并设置其顺序后，就可以生成 File Package（文件包）了。文件包将储存为 PCK 文件，位于 GeneratedSoundBanks 目录的各平台文件夹中，但若需要，也可以指定不同的位置。

如果在生成期间出现错误，将会显示在生成日志中；如果没有出现错误，则可以将这些包文件添加至游戏磁盘。

生成文件包的方法如下：

1. 要为文件包指定新的储存位置，请点击 Output directory（输出目录）旁的 Browse 按钮。

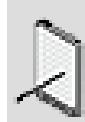
Browse For Folder（浏览文件夹）对话框将打开。

2. 浏览至您希望保存文件包的位置，选择文件夹并点击 OK。

Output directory 中将显示完整路径。

3. 在菜单栏中，点击 Generate（生成）> All packages（所有文件包）。

Generating Packages（生成文件包）对话框将打开并显示生成进度，如果出现错误，将会在日志中用黄色圆圈标记。



备注

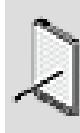
如果不生成所有文件包，您可以在 Packages（文件包）视图中选择所需文件包，然后点击 Generate（生成）> Selected Packages（所选文件包）。

相关主题

- [在工程中添加 File Package](#)
- [从工程中删除 File Package](#)
- [填充 File Package](#)
- [对 File Package 内的文件排序](#)
- [在命令行中使用 File Packager 参数](#)

在命令行中使用 File Packager 参数

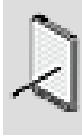
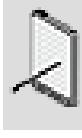
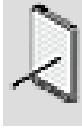
使用命令行生成文件包时，须用特定参数来定义文件包的特征。下表列出了可以在命令行中的参数。



备注

2009.2 及更早版本的 File Packager 参数与当前版本兼容。

参数	描述
-generate	<p>启动文件包生成过程，可与 File Packager 工程路径或 -info 参数一起使用。示例如下：</p> <ul style="list-style-type: none"> 通过 File Packager 工程生成文件包： -generate <project_path> 通过 SoundbanksInfo.xml 文件生成单个文件包： -generate -info <soundbank_info_filename> -output <package_name> 通过 SoundbanksInfo.xml 文件生成流文件包和库文件包： -generate -info <soundbank_info_filename> -output_stm <package_name> -output_bnk <package_name>
-info <filename>	<p>指定 SoundBanksInfo.xml 文件的位置。</p> <p>使用若干个以下参数：</p> <ul style="list-style-type: none"> -output -put_
-output <filename>	<p>生成一个包含 SoundBank 和流文件的文件包，<filename> 指定文件包名称及其保存位置。</p> <p>在没有 <projectpath> 时可以指定此参数。完整路径或仅含文件包名称均可，但都要求包含文件包扩展名。</p> <p>此参数可与 -output_stm 或 -output_bnk 一起使用来生成多个文件包，各自包含不同文件。</p> <p>虽然工程中也定义了此信息，但如果命令行显式定义了 -output 参数，其优先级将高于工程中的相应信息。</p>
-output_stm <filename>	<p>生成一个仅包含流文件的文件包，<filename> 指定文件包名称及其保存位置。</p> <p>在没有 <projectpath> 时可以指定此参数。完整路径或仅含文件包名称均可，但都要求包含文件包扩展名。</p> <p>此参数可与 -output_stm 或 -output_bnk 一起使用来生成多个文件包，各自包含不同文件。</p>
-output_bnk <filename>	<p>生成一个仅包含 SoundBank 文件的文件包，<filename> 指定文件包名称及其保存位置。</p> <p>在没有 <projectpath> 时可以指定此参数。完整路径或仅含文件包名称均可，但都要求包含文件包扩展名。</p> <p>此参数可与 -output_stm 或 -output_bnk 一起使用来生成多个文件包，各自包含不同文件。</p>
-output_loose <directory> -output_loose	<p>生成若干个仅包含零散媒体的文件包。零散文件是被 SoundBank 中的事件或结构所引用，但不包含在工程 SoundBank 中的文件。<directory> 可选，用来指定文件包的保存位置，未指定 <directory> 时将使用默认目录，即 SoundBanksInfo.xml 文件目录。</p> <p>在没有 <projectpath> 时可以指定此参数。目录可以是相对路径或绝对路径。</p> <p>此参数可与 -output、-output_stm 或 -output_bnk 一起使用来生成多个文件包，各自包含不同文件。</p>

参数	描述
-blocksize <number>	<p>文件包内数据对齐的字节数，其大小将取决于平台 I/O 设备的要求。每个平台的默认值都是 1。</p> <div style="text-align: center;">  备注 默认值 1 表示无需对齐。 </div> <p>有关 Block Size 和各平台 I/O 设备限制的详细信息，请参阅 SDK 文档 的 Streaming > Low-Level I/O (流播放 > 底层 I/O) 部分。</p>
-hideprogressui <boolean>	<p>指定在 Wwise 内生成文件包时是否显示 Generating Packages 对话框。如果设置为 True，则进度对话框将不会显示。默认情况下，此参数为 False。</p>
<projectpath>	<p>指定 File Packager 工程路径，直接指向 WFPPROJ 文件。与 -generate 参数一起使用时，将采用此参数指定的 File Packager 工程来生成文件包，而忽略其它参数。</p> <p>此参数为可选参数。</p>
-soundbanks_dir <directory>	<p>覆盖 SoundBanksInfo.xml 文件（位于 SoundBanksInfo/RootPaths/SourceFilesRoot）中指定的 GeneratedSoundBanks 目录信息。</p> <p>默认路径为：'projectpath\GeneratedSoundBanks\platform\'。</p> <p>此参数为可选参数。</p>
-cache_dir <directory>	<p>覆盖 SoundBanksInfo.xml 文件（位于 SoundBanksInfo/RootPaths/SourceFilesRoot）中指定的 Wwise 保存已转码结果文件的目录信息。</p> <p>默认路径为：'projectpath\.cache\platform\'。</p> <p>此参数为可选参数。</p>
-languages "<language1> <languageN>"	<p>指定要打包的语言列表。</p> <div style="text-align: center;">  备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。 在语言列表中，可以使用关键字“SFX”以将其打包。 </div> <p>此参数为可选参数。默认情况下会打包所有语言。</p> <p>仅在未指定 File Packager 工程时才会使用此参数。</p>
-banks "<soundbank1> <soundbankN>"	<p>指定要打包的 SoundBank 列表，名称不包含 BNK 扩展名。</p> <div style="text-align: center;">  备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。 </div> <p>此参数为可选参数。默认情况下会打包所有 SoundBank。</p> <p>指定了 File Packager 工程时仍可以使用此参数，将自动创建含有特定 SoundBank 的文件包。</p>
-includedFilesInBanks "<soundbank1> <soundbankN>"	<p>指定 SoundBank 列表，其中的媒体应作为零散文件添加到文件包中。仅当使用 PrepareEvent API，且加载的 SoundBank 也需要此媒体时，才应使用此选项。名称不包含 BNK 扩展名。</p>

参数	描述
	 <p>备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。</p> <p>此参数为可选参数。默认情况下，SoundBank 中的媒体文件不会作为零散文件添加到文件包中。</p>
-excludedFilesInBanks "<soundbank1> <soundbankN>"	 <p>备注 列表由空格进行分隔，并用双引号括起整个参数中的列表。</p> <p>此参数为可选参数。默认情况下，对于 SoundBank 引用但未包含的媒体文件，仅在其它 SoundBank 也未引用时才会被添加到文件包中。</p>

File Packager 技巧和最佳实践

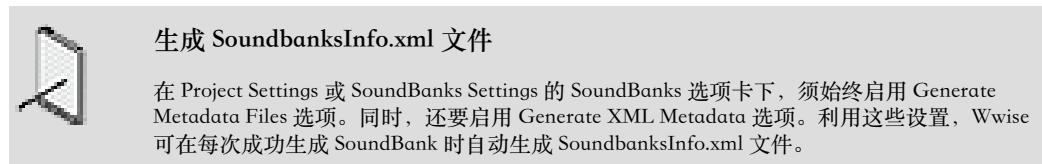
在决定使用多少文件包及储存哪些内容之前，您可能需要阅读以下章节，其中的技巧和经验总结可以帮您充分利用 File Packager。

将发布后的下载内容打包

越来越多的公司在游戏发布后，会给玩家提供 DLC（可下载内容）来延长游戏生命周期。您可以通过以下步骤来轻松管理这些新内容：

为最终发布版本内容打包的方法如下：

1. 在 Wwise 工程中为游戏的最终版本生成 SoundBank。
2. 打开 File Packager。
3. 导入 Wwise 生成的 SoundBanksInfo.xml 文件。



4. 将默认文件指派设置为 **None**（无）。
5. 手动将所有文件添加至一个文件包。

这将方便之后识别新的内容。

6. 生成文件包并将其添加至游戏磁盘。
7. 保存 File Packager 工程，作为最终版本的起始参考点。

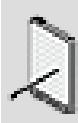
游戏发布后，打包 DLC 内容的方法如下：

1. 在 Wwise 中创建新内容后，为 DLC 生成 SoundBank。

2. 重新打开“最终版本”的 File Packager 工程。

默认情况下，File Packager 将扫描 SoundBanksInfo.xml 文件并在 Files to package 中显示。注意，所有新内容尚未指派给文件包。

3. 为可下载内容创建若干个文件包，并手动添加新的 SoundBank 或流文件。
4. 生成新文件包并进行部署。
5. 再次保存 File Packager 工程，作为此 DLC 版本的新起始参考点。



备注

使用相同的方法，可以继续为您的游戏部署新增内容。只需记住要手动将所有文件指派至一个文件包，并保存 File Packager 工程来为各版本创建新的起始参考点。



37. 认识 Project Explorer 视图	692
概述	693
Project Explorer 中的可视元素	695
在 Project Explorer 中工作	695
38. 认识 Event Viewer 视图	699
概述	700
使用 Event Viewer	700
39. 认识 Property Editor 视图	705
概述	706
使用 Property Editor	723
40. 认识 Contents Editor 视图	726
概述	727
使用 Contents Editor	734
41. 认识 Transport Control 视图	741
概述	742
设置播放属性	744
在 Transport Control 中平移对象	746
播放/暂停/停止内容	746
在播放期间使用 Game Sync	747
42. 认识 Schematic View 视图	752
概述	753
自定义 Schematic View 视图	753
使用 Schematic View 视图	755
43. 了解坐标图视图	758
概述	759
更改坐标图的显示	760
在坐标图视图中使用控制点	764
使用坐标图视图中的曲线	766
44. 认识时间线	770
概述	771
使用时间线进行定位	773
使用 Music Segment Editor 时间线	774
45. 使用 Search、Query 和 Reference	777
概述	778
搜索工程中的元素	778
查找引用特定对象的工程元素	780
使用 Query	784
查询 —— 技巧和经验总结	791
46. 使用 Preset	792
概述	793
使用 Preset	793
47. 使用控制设备	797
概述	798
将控制设备连接到 Wwise	798
创建 Control Surface Session	799
理解控制设备绑定	799
创建控制设备绑定	800
理解 Control Surface 的 View Group	803
处理控制 Control Surface Session 中的冲突	804

使用 Control Surface 工具栏 804

第 37 章 认识 Project Explorer 视图

概述	693
Project Explorer 中的可视元素	695
在 Project Explorer 中工作	695

概述

Project Explorer 是管理 Wwise 工程中所有元素的工具。在它的各个选项卡中，您可以创建、重命名、剪切、复制、粘贴和删除选项卡层级结构中显示的独特元素。各个选项卡还包含工具栏，您可以通过它快速将父对象和子对象添加到工程层级结构。在此中心位置，您可以组织和管理 Wwise 工程的各种元素，包括音频、音乐和振动素材、Bus、Event、SoundBank、Game Sync 等。

在 Project Explorer 中，您还可以将工程元素分成工作单元，让团队的每个成员同时处理工程的不同部分。有关创建工作单元的详细信息，请参阅[将工程分成 Work Units](#)。

Project Explorer 包含以下选项卡：

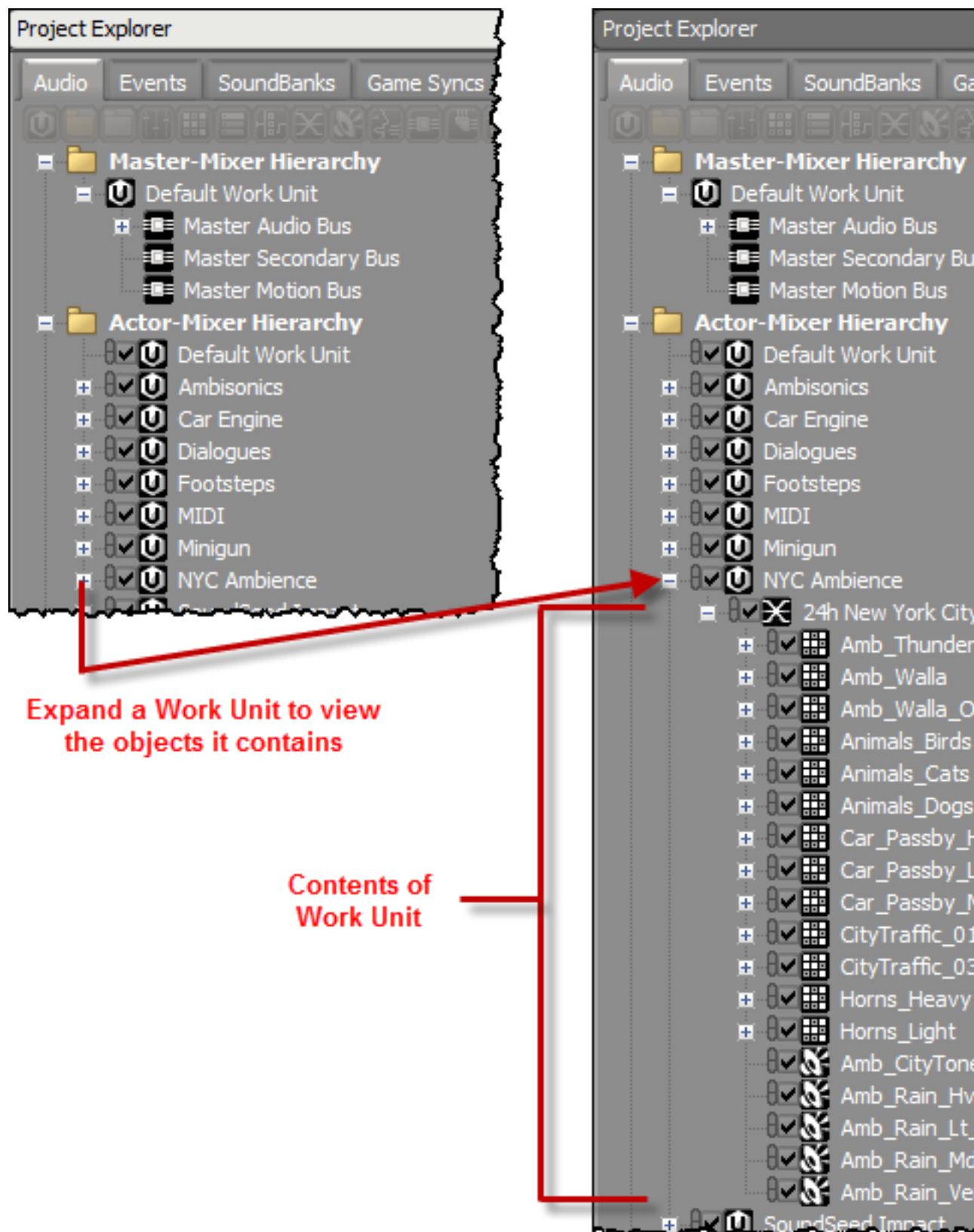
- **Audio** —— 包含工程中的所有声音、音乐、振动和 Bus 结构。
- **Events** —— 包含工程中的所有事件（动作和对白）。
- **SoundBanks** —— 包括工程中的所有 SoundBank。
- **Game Syncs** —— 包含工程中的所有 State、Switch、Trigger 和 Game Parameter。
- **ShareSets** —— 包含工程中的所有效果和 Attenuation ShareSet。
- **Sessions** —— 包含工程中的所有 Soundcaster 会话。
- **Queries** —— 包含工程中的所有查询。

若要在 Project Explorer 中浏览不同层级，您可以通过点击各个对象旁边的加号 (+) 和减号 (-) 来展开和折叠这些分组。在按住 Ctrl 的同时单击可展开/折叠所有子节点。



快速展开和折叠分组

按下数字小键盘中的向右键或向左键或加号 (+) 和减号 (-) 键，可将所选分组的上/下级节点展开或折叠。此外，还可按下数字小键盘中的斜线 (/) 和星号 (*) 键，来展开和折叠分组内的所有子节点。



Project Explorer 中的可视元素

Wwise 使用不同的图标表示工程中的各个对象和元素，方便您在 Project Explorer 和界面其它位置中找到不同对象和元素类型。有关所有图标的完整列表和描述，请参阅[理解 Wwise 中的视觉元素](#)。

Project Explorer 中还使用其它可视元素（例如颜色）来帮助识别特定对象的状态。例如，对象的颜色可指明它是否具有相关源，或者是否针对特定平台进行了转码。下表包含 Project Explorer 中使用的可视元素的列表。

可视元素	示例	用途	选项卡
星号		对工作单元中的一个或多个工程元素进行了更改。保存工程时，星号将消失。	全部
Link 图标		Link 图标为灰色时，表示该对象设置适用于所有平台。Wwise 可以让您同时针对多个平台进行制作，其方法是链接（Link）和取消链接（Unlink）与对象相关联的大多数属性值。默认情况下，对于所有活动平台，所有属性值都是链接在一起的，因此属性在各平台上的值相同。取消链接属性值可让您为特定平台自定义属性值。 关于链接和取消链接的详细信息，请参阅 根据平台自定义对象属性 。	音频和 ShareSet（用于效果器和 Audio Device）
复选标记		当对象名称旁边显示复选标记时，当前平台中包含该对象。当没有显示复选标记时，当前平台中不包含该对象。	音频和 ShareSet（用于效果器和 Audio Device）
对象名称为红色		创建了一个 Sound SFX 或音乐音轨，但没有与源相关联。	Audio
对象名称为蓝色		与源相关联的 Sound SFX 或 Music Track 尚未针对当前平台进行转码。	Audio
对象名称为白色		与源相关联的声音或音乐轨尚未针对当前平台进行转码。	Audio

如果您使用了工作组插件，在 Project Explorer 中，工作单元上方将显示特殊的叠加图标，帮助您确定工程文件的状态。例如，标记为 Add、Check-out 和 Check-in 的文件将使用不同的图标。有关叠加图标的详细信息，请参阅[使用 Workgroup 插件管理工程文件](#)。

在 Project Explorer 中工作

在 Project Explorer 中，您可以使用快捷菜单执行标准的 Windows 资源管理器（或 Mac Finder）命令，例如重命名、剪切、复制和粘贴。您还可以在选项卡内拖放工程元素，或者将工程元素拖放到 Wwise 界面中的其它视图中。请记住，每当您移动对

象时，将影响父子关系。有关创建和管理这些元素及其关系的信息，请参阅以下各节：

- 构建 Actor-Mixer Hierarchy
- 第 23 章 构建 Interactive Music Hierarchy
- 第 15 章 管理 Event
- 第 16 章 管理动态对话
- 第 35 章 管理 SoundBank
- 使用 Switch
- 第 17 章 使用 State
- 第 19 章 使用 RTPC
- 第 20 章 使用 Trigger
- 第 21 章 将 State 和 State Group 用于动态对话
- 使用效果器
- 管理多份衰减
- 创建模拟
- 创建查询

除了标准命令之外，Project Explorer 快捷菜单还由例如包含和排除之类的命令，以及当前选项卡的特定命令，例如 Audio 选项卡中导入和转码音频文件的命令，和 SoundBank 选项卡中导入 SoundBank Definition 的命令。



设置显示选项

为帮助您管理工程中的复杂层级结构，您可能会需要设置 Project Explorer 以特定方式显示，以便您在需要时快速查找特定工程元素。例如，您可以设置自动折叠各个分组，以便在 Actor-Mixer 层级结构中更加轻松地进行导航。

更改 Project Explorer 显示的方法是：

1. 在 Project Explorer 的任何选项卡中，右键点击对象。

此时将会显示快捷菜单。

2. 在快捷菜单中，选择 Expand Options。

此时将显示 Expand Options 子菜单。

3. 选择以下选项之一：

- Expand Descendants（展开子对象）：展开层级结构中的所有对象。
- Collapse Descendants（折叠子对象）：折叠层级结构中的所有对象。

您还可以使用以下键盘快捷键快速在 Project Explorer 中的不同层级导航。

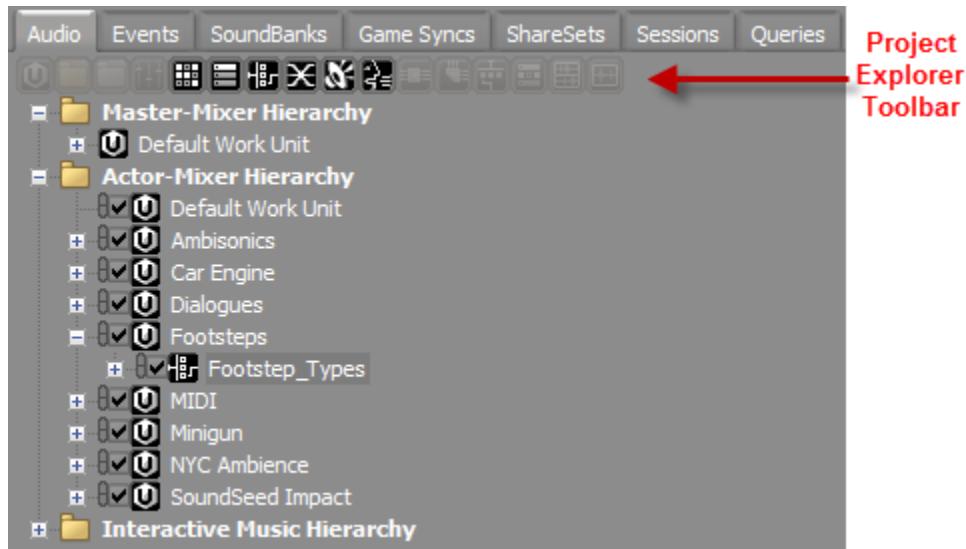
键盘快捷键	操作
向上箭头	在对象列表中向上移动。
向下箭头	在对象列表中向下移动。
数字小键盘中的向右键或加号键	展开对象。
数字小键盘中的向左键或减号键	折叠对象。
数字小键盘中的斜线键	折叠对象中所有级别的节点。
数字小键盘中的星号键	展开对象中所有级别的节点。

相关主题

- [使用 Project Explorer 工具栏](#)
- [Project Explorer 中的可视元素](#)

使用 Project Explorer 工具栏

Project Explorer 的每个选项卡包含一个图标工具栏。这些图标表示可以添加到层级结构中去的工程元素，这些元素可以添加成为选定节点的父项或子项。不同的图标将根据层级结构中选定对象的类型而激活。



在默认情况下，该工具栏显示可作为选定节点的子项加以添加的所有工程元素。要显示可以添加的父工程元素，请按 Shift 键。

使用工具栏将子对象添加到层级结构中的方法是：

1. 在 Project Explorer 中，选择层级结构中的对象或其它工程元素。

这时在工具栏中，可添加成为选定元素子项的工程元素所对应的图标激活了。

2. 点击活跃图标之一来创建选定元素的新子项。
3. 为子对象取一个恰当的名称，然后按 Enter。

使用工具栏将父对象添加到层级结构中的方法是：

1. 在 Project Explorer 中，选择层级结构中的对象或其它工程元素。

2. 按 Shift 以显示可添加成为选定元素父项的工程元素。
3. 点击活跃图标之一以创建选定元素的新父项。
4. 为父对象取一个恰当的名称，然后按 Enter。

相关主题

- [设置显示选项](#)
- [Project Explorer 中的可视元素](#)

第 38 章 认识 Event Viewer 视图

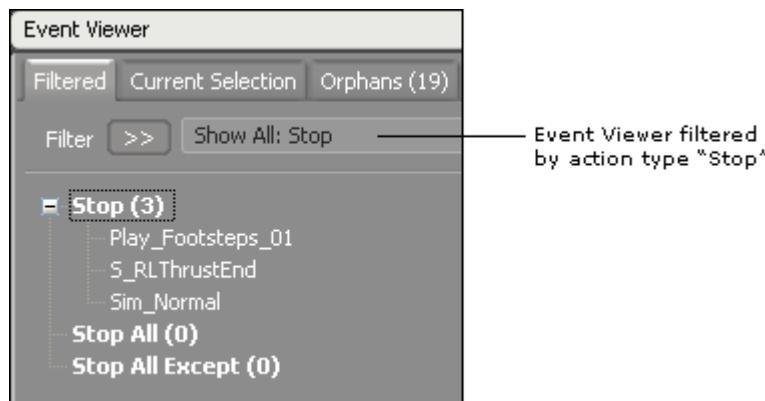
概述	700
使用 Event Viewer	700

概述

Wwise 使用 Action Event（动作事件）对工程层级中的不同声音、音乐和振动结构采取特定的动作。常见的游戏中，这样的事件可能有数百个，因此能够快速找到所需事件变得至关重要。您可以使用 Event Viewer（事件浏览器）中的排序和筛选工具来查找当前工程中的不同事件。

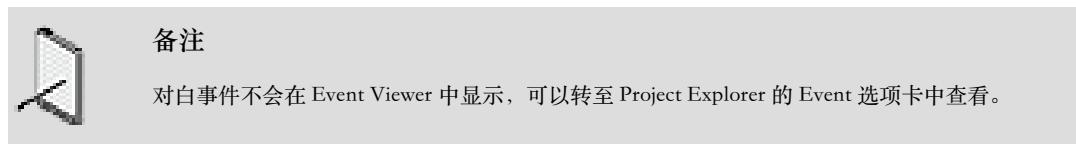
Event Viewer 有三个选项卡，分别以不同的方式筛选事件：

- **Filtered**（筛选）选项卡——按字母顺序显示所有事件。您还可以使用 Show All: Sorted 选项，按 Action Type（动作类型）对列表排序或筛选，只显示含特定动作类型的事件。点击加号（+）和减号（-）来展开和折叠文件夹，即可浏览筛选后的事件。



- **Current Selection**（当前选中）选项卡——对于当前 Project Explorer 的 Audio 选项卡中选定的对象，与其相关的事件将显示在列表中。
- **Orphans**（落单事件）选项卡——显示当前没有与特定对象关联的落单事件。

您还可以删除事件、打开 Event Editor，或者将若干事件从 Event Viewer 拖放到 Wwise 中的其他视图（例如 Soundcaster Editor 或 SoundBank Editor）。

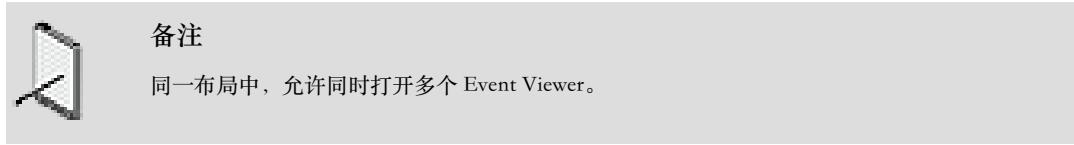


使用 Event Viewer

Event Viewer（事件浏览器）显示工程中创建的所有事件。由于工程可以有许多事件，因此您需要对它们进行排序和筛选，才能迅速找到想要编辑、播放的事件。

以下章节将帮助您熟悉 Event Viewer 中的不同工具和显示选项。

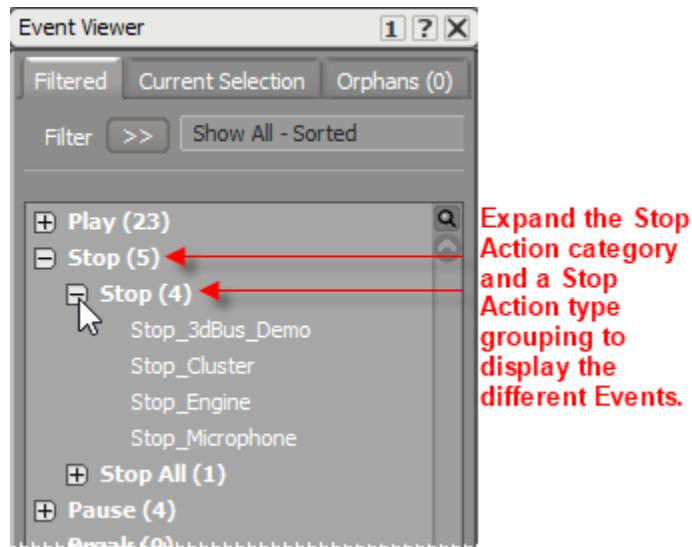
- [浏览 Event Viewer](#)
- [对事件列表进行排序](#)
- [对事件列表进行筛选](#)



浏览 Event Viewer

根据事件列表的排序或筛选方式不同，Event Viewer 中的信息可按多种不同的方式显示。有些筛选和排序选项会在 Event Viewer 中显示层级结构。要浏览不同的层级，可以点击每个 Action (动作) 类别/类型旁边的加号 (+) 和减号 (-) 来展开和折叠分组。

例如，当选定的 Filter 为 Show All - Sorted 时，要查看 Stop 动作须执行以下操作：



快速浏览

您还可以在 Event Viewer 中使用以下键盘快捷键，迅速浏览不同层级。

键盘快捷键	操作
向上箭头	在 Event 列表中向上移动。
向下箭头	在 Event 列表中向下移动。
向右箭头	展开 Event 的动作类别或类型分组。
向左箭头	折叠 Event 的动作类别或类型分组。

更改 Event Viewer 显示

视情况或个人偏好而异，您可能需要更改 Event Viewer 的显示方式。例如，可以展开或折叠所有分组，或者让 Event Viewer 每次筛选或排序时自动展开各个分组。

更改 Event Viewer 显示的方法如下：

1. 在 Event Viewer 中右键点击组标题之一。

快捷菜单将打开。

2. 选择以下选项之一：

- Auto Expand —— 每次筛选或排序时自动展开各个分组。
- Expand All —— 展开所有类别和类型分组。
- Collapse All —— 折叠所有类别和类型分组。

相关主题

- [对事件列表进行排序](#)
- [对事件列表进行筛选](#)

对事件列表进行排序

Event Viewer 默认显示 Filtered（筛选）选项卡，工程中的所有事件以字母顺序列出。要想更快地找到事件，您可以选择 Show All - Sorted 筛选器，按事件动作进行排序。也就是说所有 Play 动作分成一组，所有 Stop 动作分成一组，以此类推。

对事件列表进行排序的方法如下：

1. 在 Event Viewer 中，点击 Filtered 选项卡。

默认情况下，所有事件会按字母顺序列出。

2. 点击 Filter 按钮，显示筛选和排序选项。
3. 点击 Show All - Sorted 选项。

工程中的所有事件将按动作类别及类型排序。

相关主题

- [对事件列表进行筛选](#)
- [浏览 Event Viewer](#)
- [更改 Event Viewer 显示](#)

对事件列表进行筛选

您可以对事件列表进行筛选，进一步优化对事件的搜索。筛选事件的方法如下：

- [按动作类型筛选列表](#)
- [按当前选定对象对列表进行筛选](#)
- [按落单事件对列表进行筛选](#)

按动作类型筛选列表

您可以按 Action Type（动作类型）筛选事件列表。例如，可以让列表仅显示 Stop All 动作或 Mute 动作。

按动作类型对事件列表进行筛选的方法如下：

1. 在 Event Viewer 中，点击 Filtered 选项卡。

默认情况下，所有事件会按字母顺序列出。

2. 点击 Filter 按钮，显示筛选和排序选项。
3. 选择动作类别，然后点击列表中的动作类型。

将按您的选择筛选事件。

相关主题

- [按当前选定对象对列表进行筛选](#)
- [按落单事件对列表进行筛选](#)
- [浏览 Event Viewer](#)
- [对事件列表进行排序](#)
- [更改 Event Viewer 显示](#)

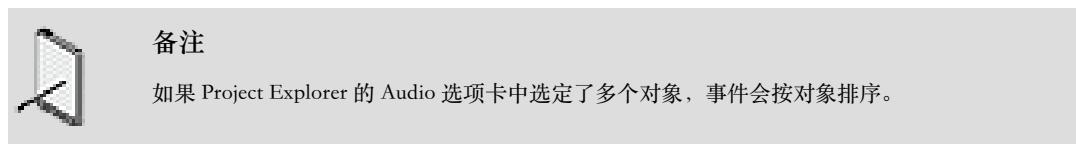
按当前选定对象对列表进行筛选

对于 Project Explorer 的 Audio 选项卡中当前选定对象，您可以查找与其相关的事情，对列表进行筛选。

按当前选定对象对事件列表进行筛选的方法如下：

1. 在 Project Explorer 的 Audio 选项卡中，选择若干对象。
2. 在 Event Viewer 中，点击 Current Selection 选项卡。

与选定对象相关的事件将显示在事件列表中。



相关主题

- [按动作类型筛选列表](#)
- [按落单事件对列表进行筛选](#)
- [浏览 Event Viewer](#)
- [对事件列表进行排序](#)
- [更改 Event Viewer 显示](#)

按落单事件对列表进行筛选

您可以对列表进行筛选来查找 Orphan Event（落单事件），其中包含没有关联到任何对象的动作。Orphans 选项卡标题括号中的数字表示工程中的落单事件总数。

按落单事件对列表进行筛选的方法如下：

1. 在 Event Viewer 中，点击 Orphans 选项卡。

落单事件将显示在事件列表中。

相关主题

- [按动作类型筛选列表](#)
- [按当前选定对象对列表进行筛选](#)
- [浏览 Event Viewer](#)
- [对事件列表进行排序](#)
- [更改 Event Viewer 显示](#)

第 39 章 认识 Property Editor 视图

概述	706
使用 Property Editor	723

概述

Property Editor 是一组属性和行为选项的集合，这些选项可用于定义特定对象或 Game Sync 总体特性。

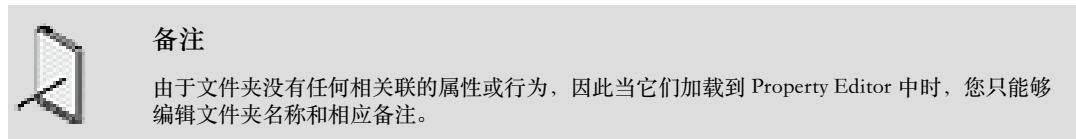
当音频对象被加载到 Property Editor 中时，根据对象类型不同，可能显示以下多个选项卡：

- General Settings
- Conversion
- Effects
- 定位 (仅适用于 Actor-Mixer Hierarchy, Interactive Music Hierarchy 和大多数 Master-Mixer Hierarchy 对象)
- RTPC (实时参数控制)
- States (状态)
- Auto-ducking (仅适用于 Audio Bus)
- HDR (仅适用于 Audio Bus)
- Transitions (过渡) (仅限于音乐对象)
- Stingers (仅限于音乐对象)
- Mixer Plug-in
- MIDI (仅适用于 Actor-Mixer Hierarchy 和 Interactive Music Hierarchy 对象)
- Advanced Settings (高级设置) (仅适用于除 Auxiliary Bus 外的所有对象)
- All Properties (所有属性)



当 Game Sync 加载到 Property Editor 时，根据您正在编辑的 Game Sync 的类型将显示不同的属性。下表描述为各个 Game Sync 所显示的属性：

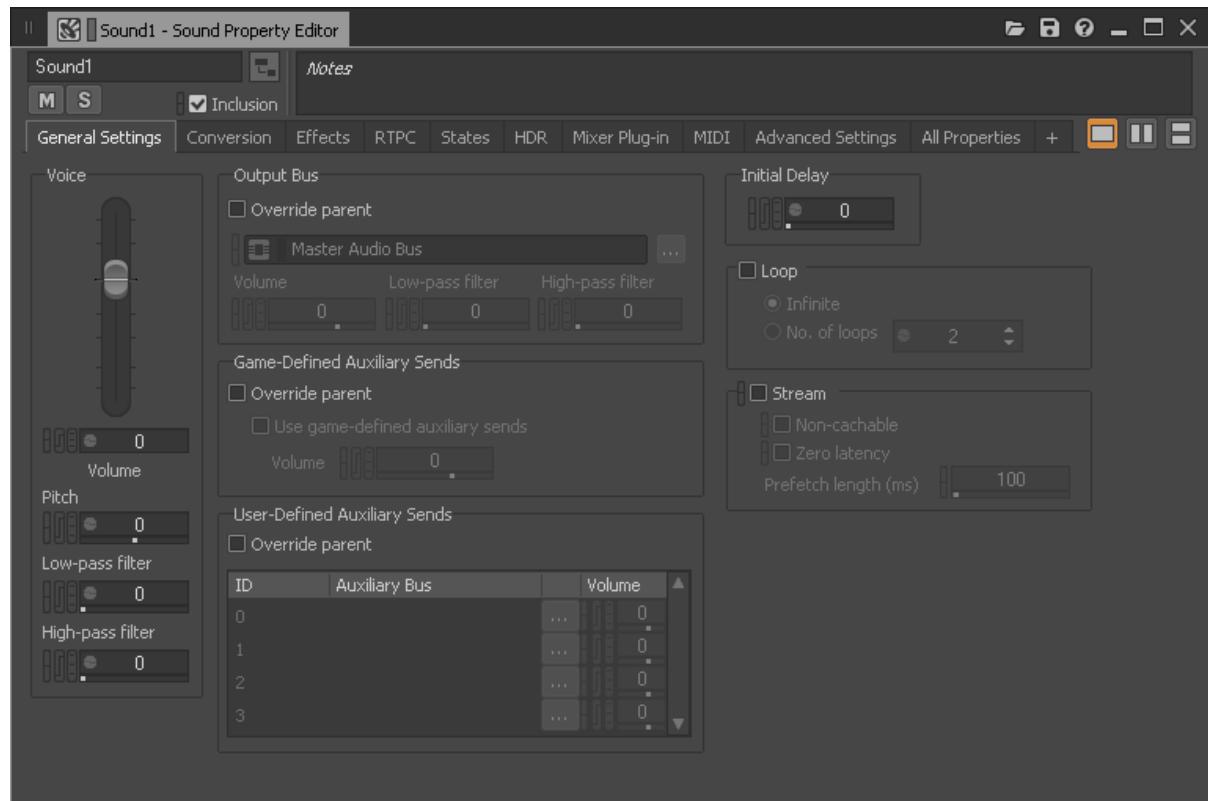
Game Sync	Property Editor 内容
Switch Group	Game Parameter 值映射的名称 (Name)、备注 (Notes) 和 Switch。
Switch	名称和备注。
State Group	名称、备注和 State 之间的过渡设置。
State	名称、备注和 State 值副本。
Game Parameter (游戏参数)	名称；备注；要绑定的内置参数；游戏参数的最小值、最大值和默认值；以及插值模式值。
Trigger (触发器)	名称和备注。



General Settings

General Setting（常规设置）选项卡是对象 Property Editor 中的主要选项卡。其中包含一系列选项，通过这些选项，您可以查看和定义工程层级结构中的对象的一般特性。例如，您可以设置对象的音量和音高值。您还可以为层级结构中的不同对象定义特定的播放行为。

General Settings 选项卡划分为属性和行为两部分。属性显示在选项卡左侧，行为显示在右侧。属性进一步分为两组：绝对属性和相对属性。绝对属性显示在左侧，相对或累加属性显示在右侧。



有关相对和绝对属性的完整描述，请参阅[工程层级结构中的属性介绍](#)。

General Settings 选项卡也是根据上下文环境而定的，这意味着此选项卡中显示的属性和行为选项将根据您当前正在编辑的对象类型而不同。

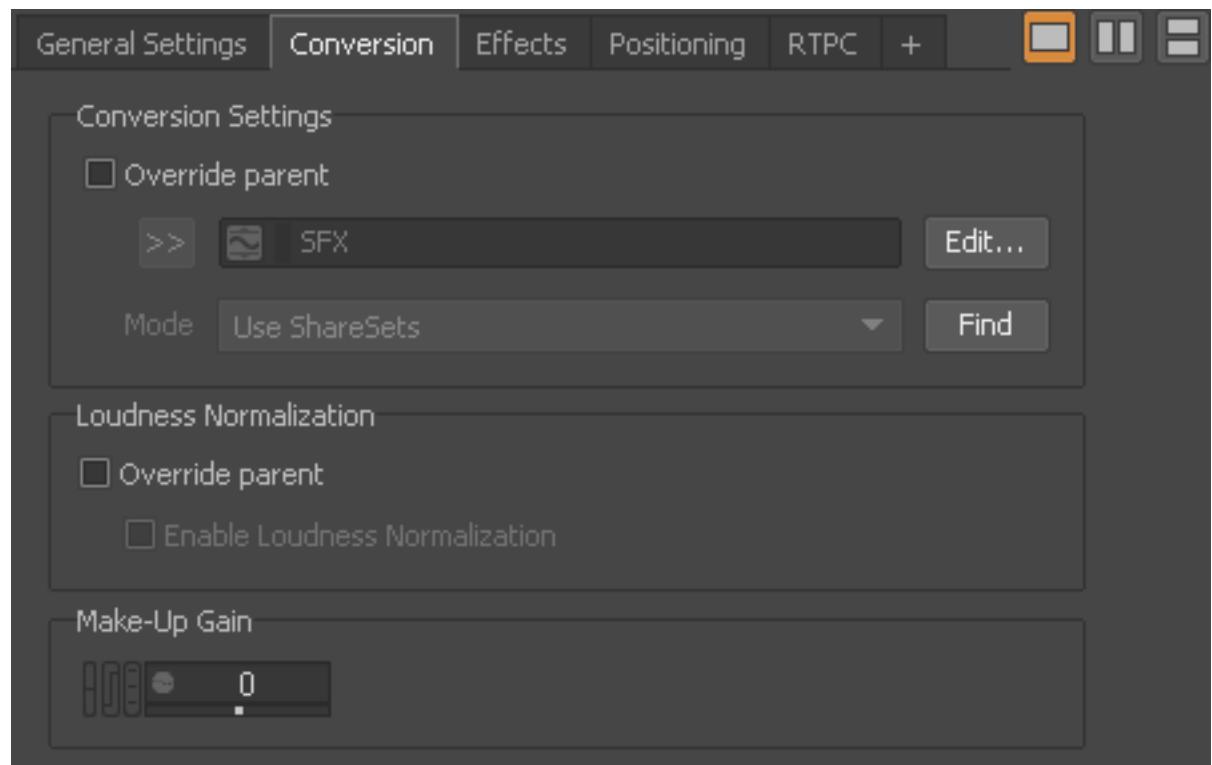


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

Conversion

Conversion 选项卡允许您为对象应用和编辑 Conversion Settings。其中还可以使用响度归一化和补偿增益来调节音量。



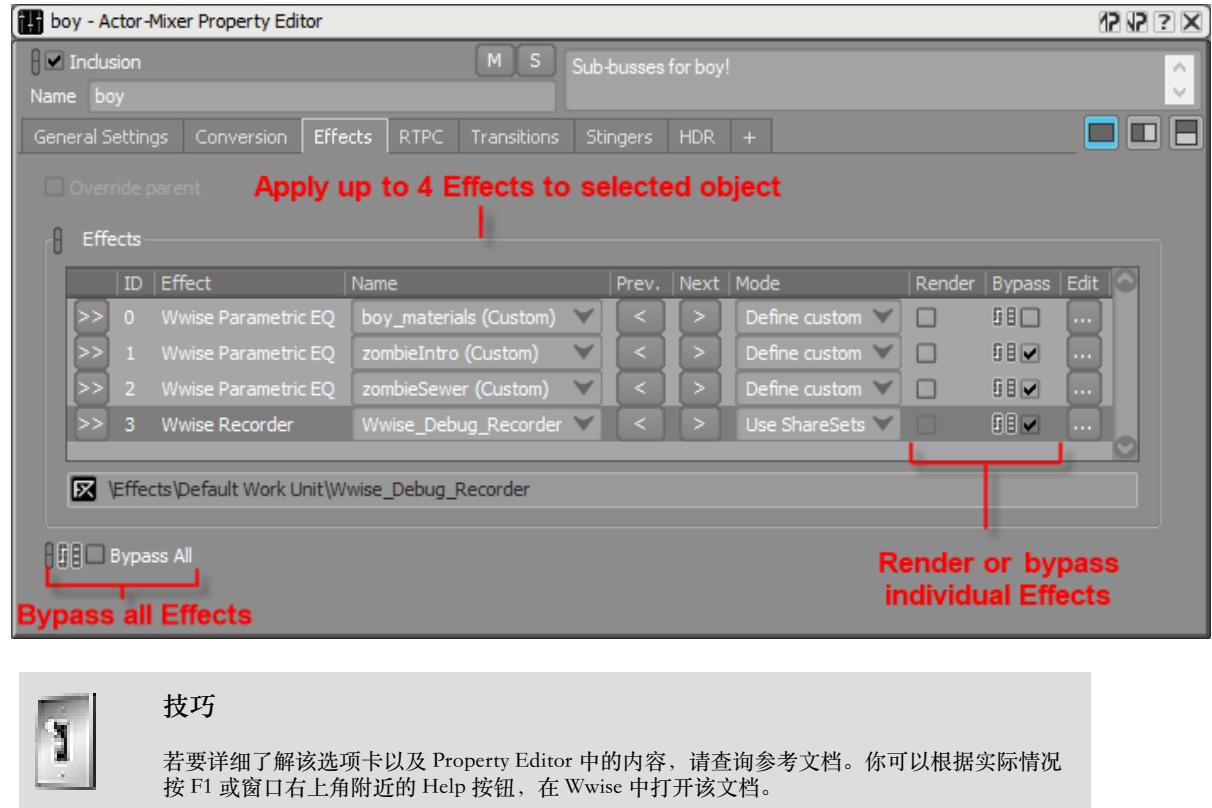
技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

Effects

Effects (效果器) 选项卡用于将一个或多个效果器应用于工程层级结构中的对象或总线。在应用效果器时，必须决定是应用效果器的 ShareSet 还是自定义实例。您还可以旁通、渲染或编辑效果器。

认识 Property Editor 视图

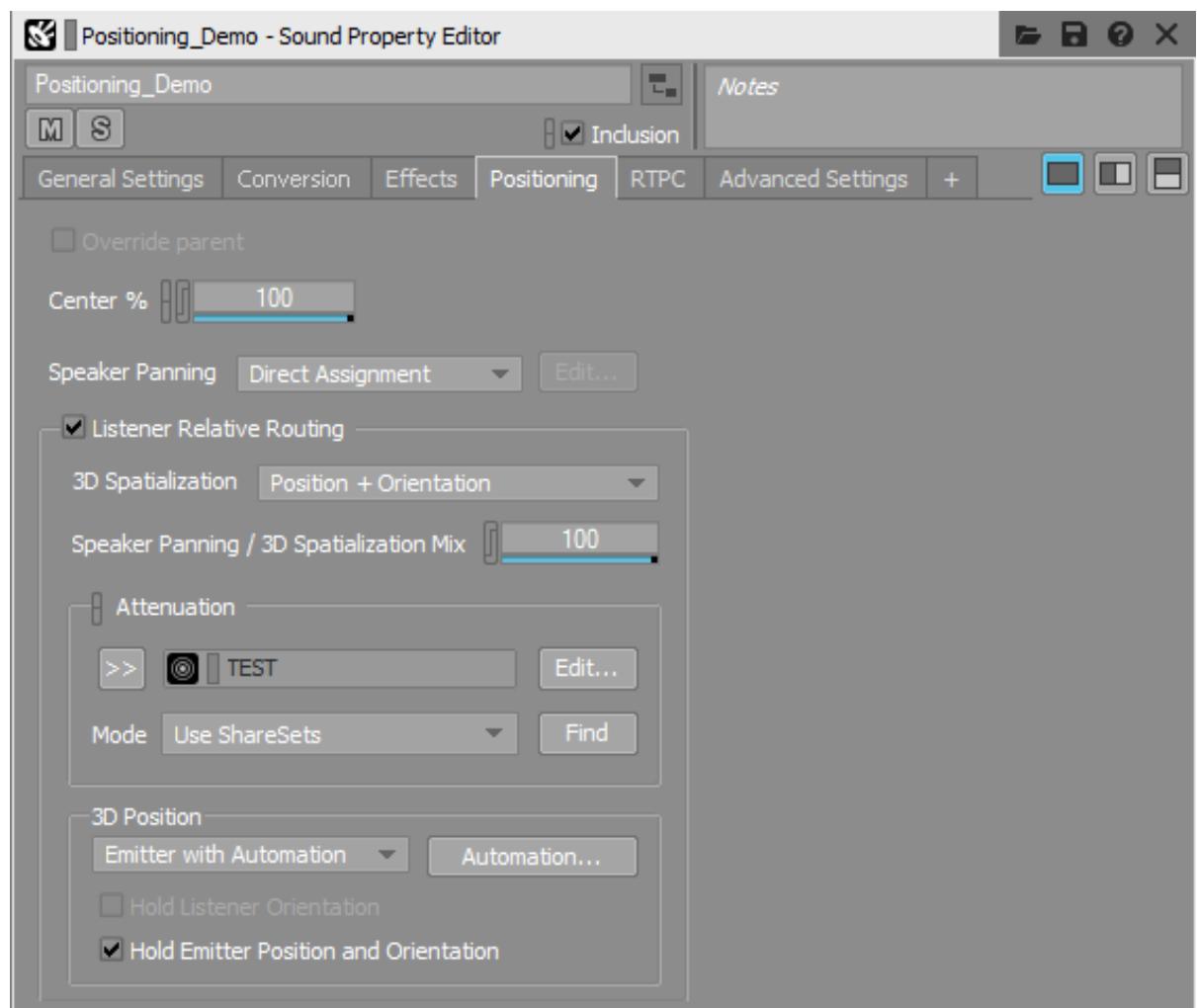


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

定位

Positioning (定位) 选项卡包含一系列选项，在此您可以查看和定义工程对象的定位特性。您可以设置对象是使用 Speaker Panning (扬声器声像摆位) 和 3D Position (3D 定位) 中的一种还是两种。然后可以设置如何应用这两种类型的定位。对于对象的音频信号分配，您还可以指定通过中置扬声器播放的百分比。

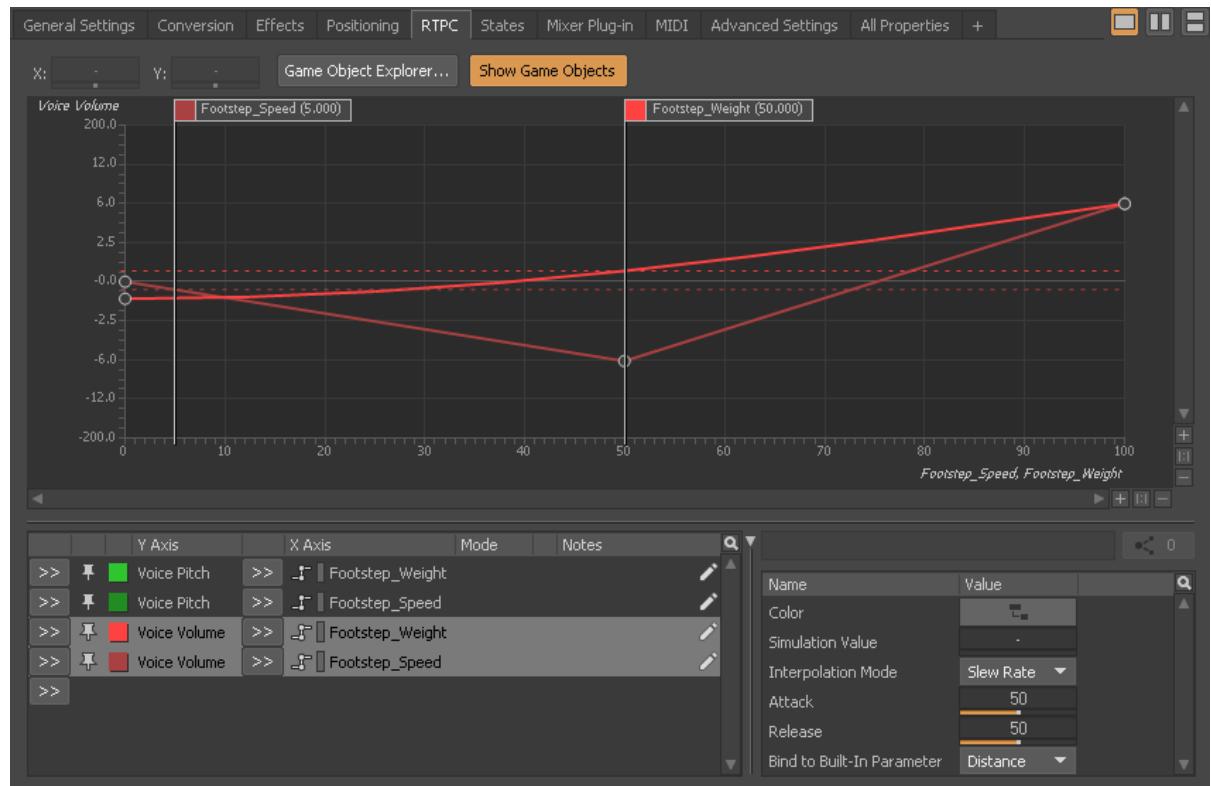


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

RTPC

通过 RTPC（实时参数控制）选项卡上的控件，您可以根据游戏中发生的实时参数值变化实时编辑特定的声音属性。参数值显示在坐标图视图中，其中一个轴表示 Wwise 中的属性值，另一个轴表示游戏中的参数值。您可以一次处理一条曲线，也可以同时编辑所有曲线。



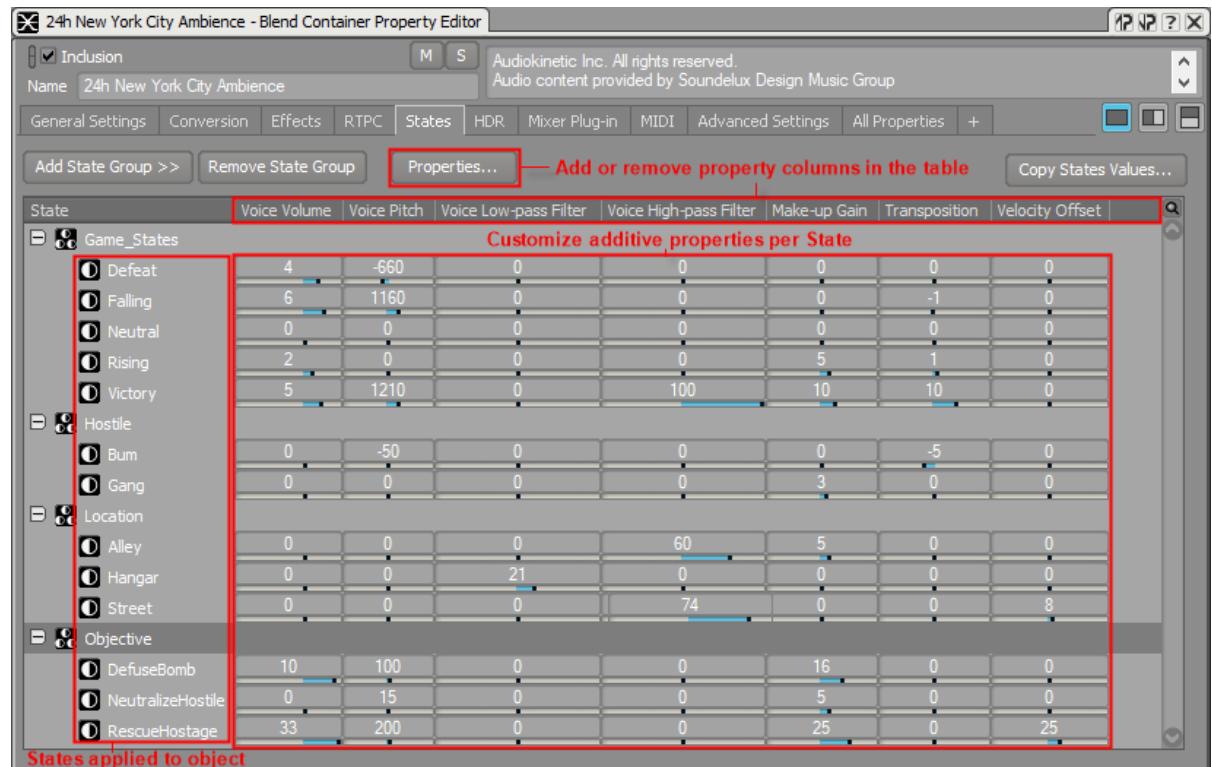
技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

States (状态)

States 选项卡允许您为当前对象添加包含各种 State 的 State Group。然后，您可以为每个状态自定义累积属性值，例如音高和音量，来进一步定义对象的特征。通过单击 Properties 按钮，即可根据所选对象类型来添加或删除对象属性。

认识 Property Editor 视图

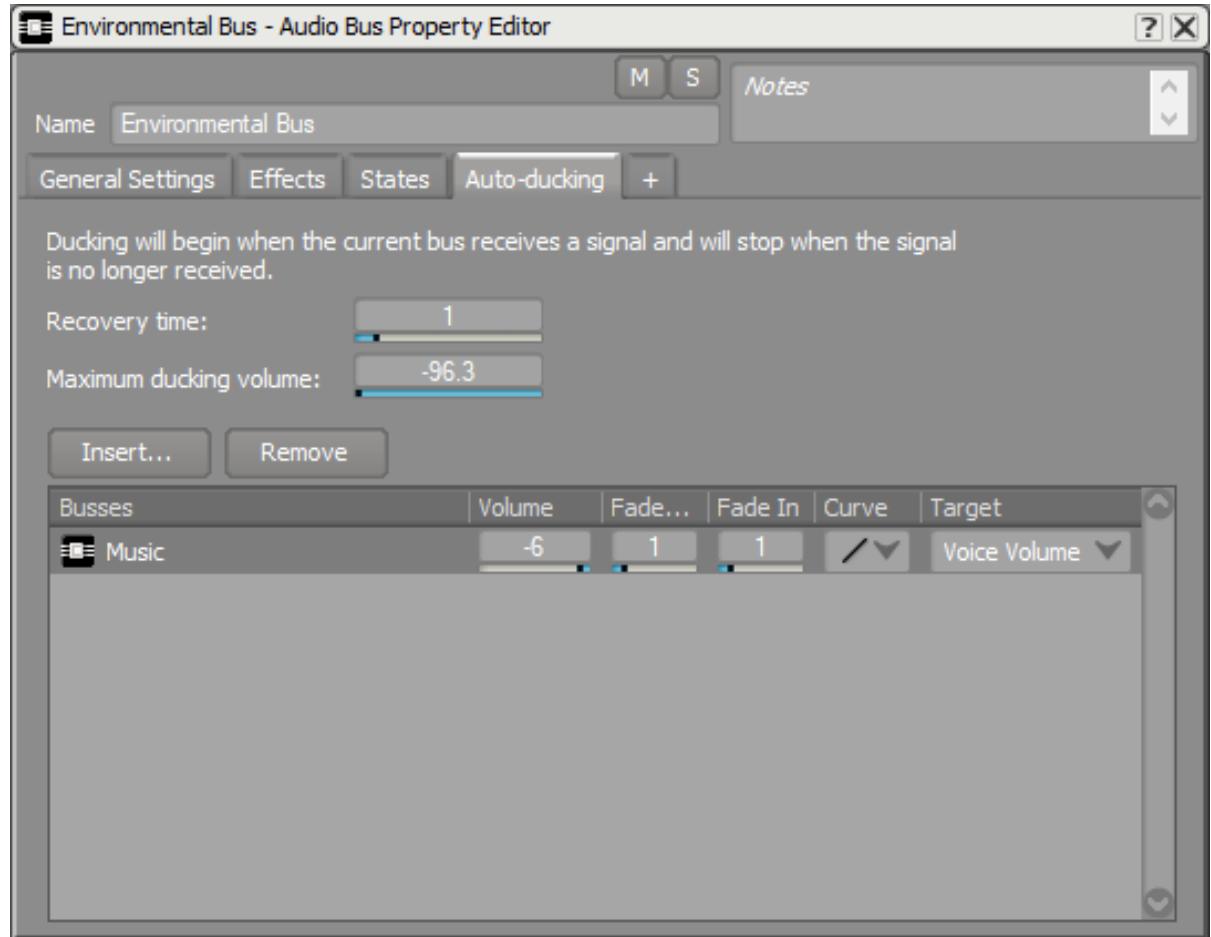


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

Auto-duking

在 Audio Busses 的 Auto-duking 选项卡中，您可以在当前总线输出时降低其它音频信号的音量，从而在同时播放时，使当前总线信号更加突出。



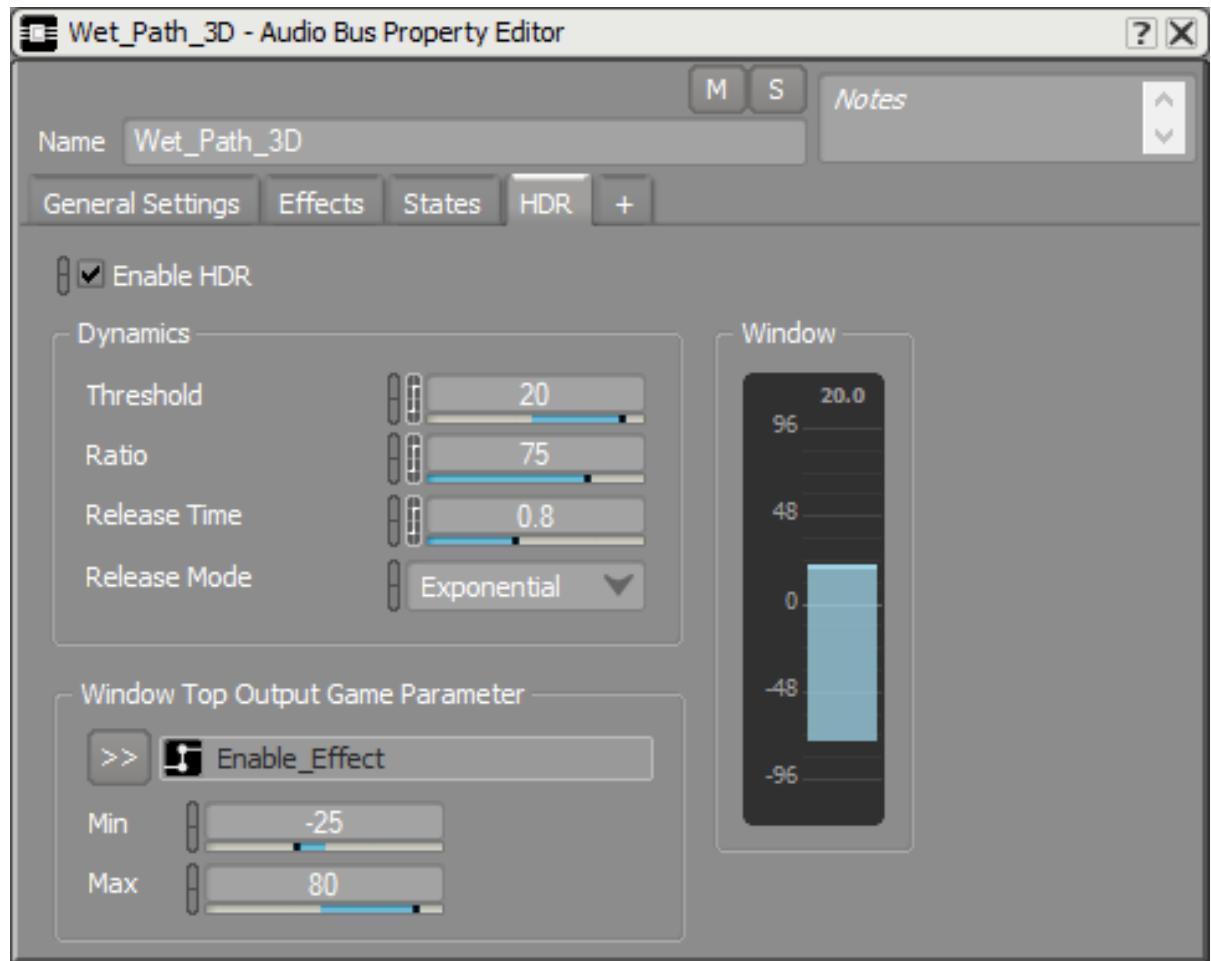
技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

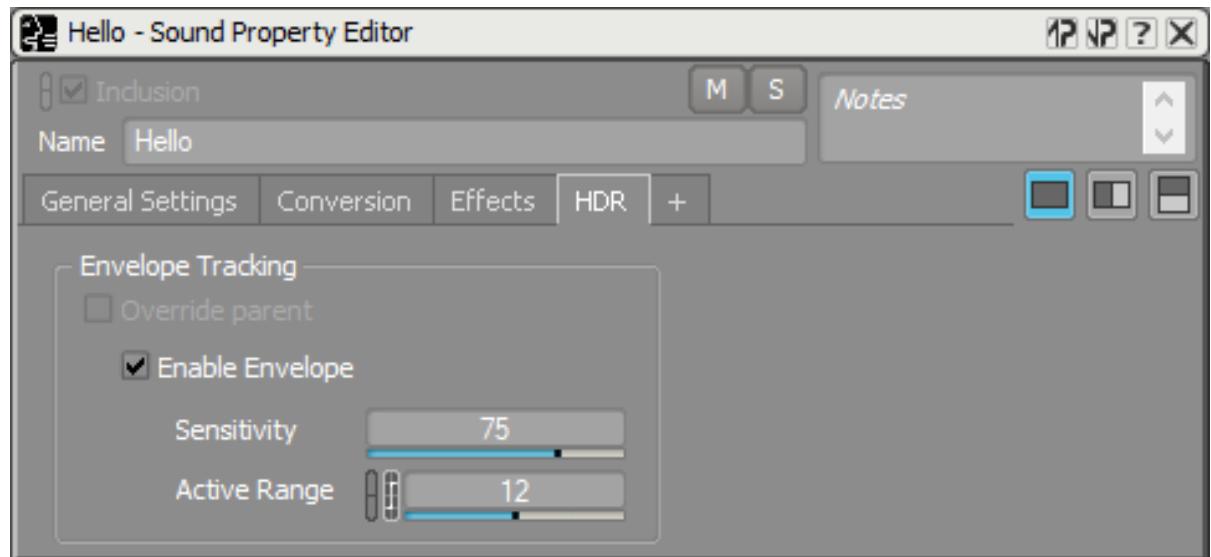
HDR

高动态范围音频（HDR 音频）是一种利用自然声音所跨越的大动态范围电平值来进行混音设计的技术。HDR 也是一个实时系统，可以将宽泛的电平范围动态地映射至更适合于您的声音系统数字输出的范围。HDR 选项卡用于定义总线的 HDR 行为和属性，或者定义音频对象的 HDR 行为和属性。

认识 Property Editor 视图



Audio Bus 的 HDR 选项卡



音频对象的 HDR 选项卡

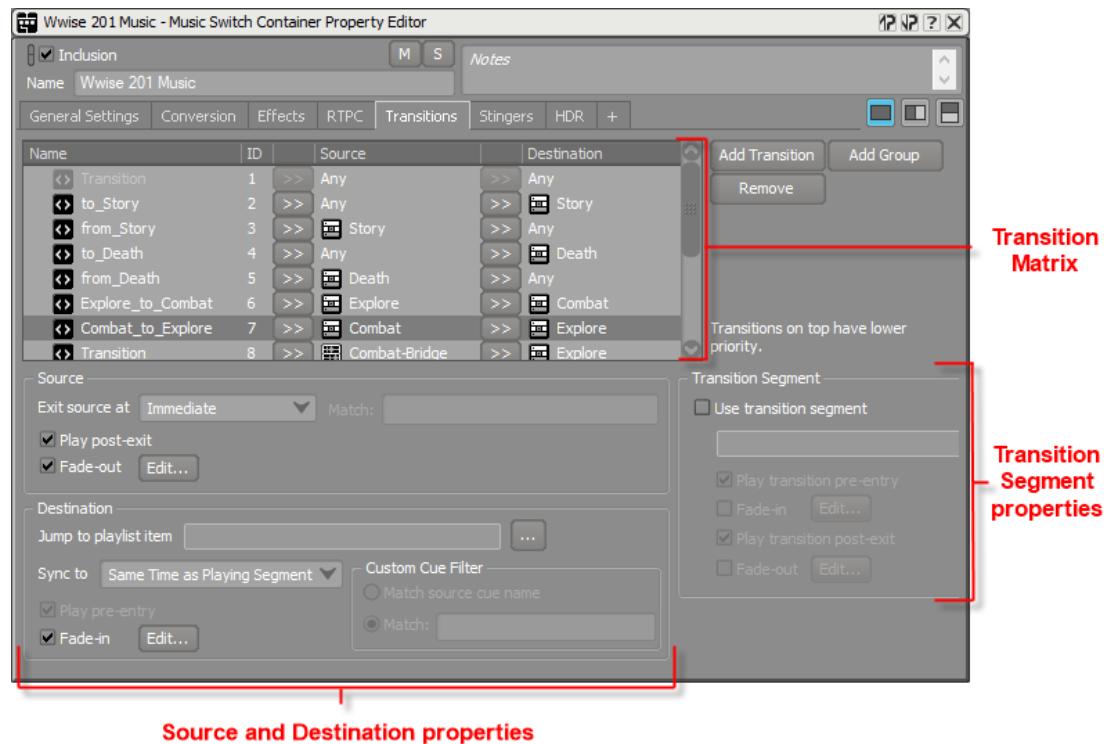


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

Transitions (过渡)

Transitions (过渡) 选项卡的主要部件是 Transition Matrix。它是一个规则列表，定义 Music Switch 或 Playlist Container 中的各个对象如何过渡到容器中的其它各对象。它还包含控件用来为过渡中的各个源和目标自定义属性，以及指定过渡段落用途和属性。



Source and Destination properties

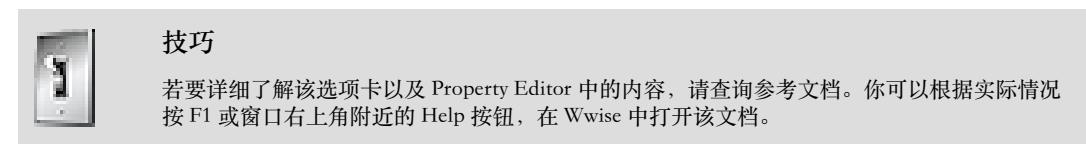
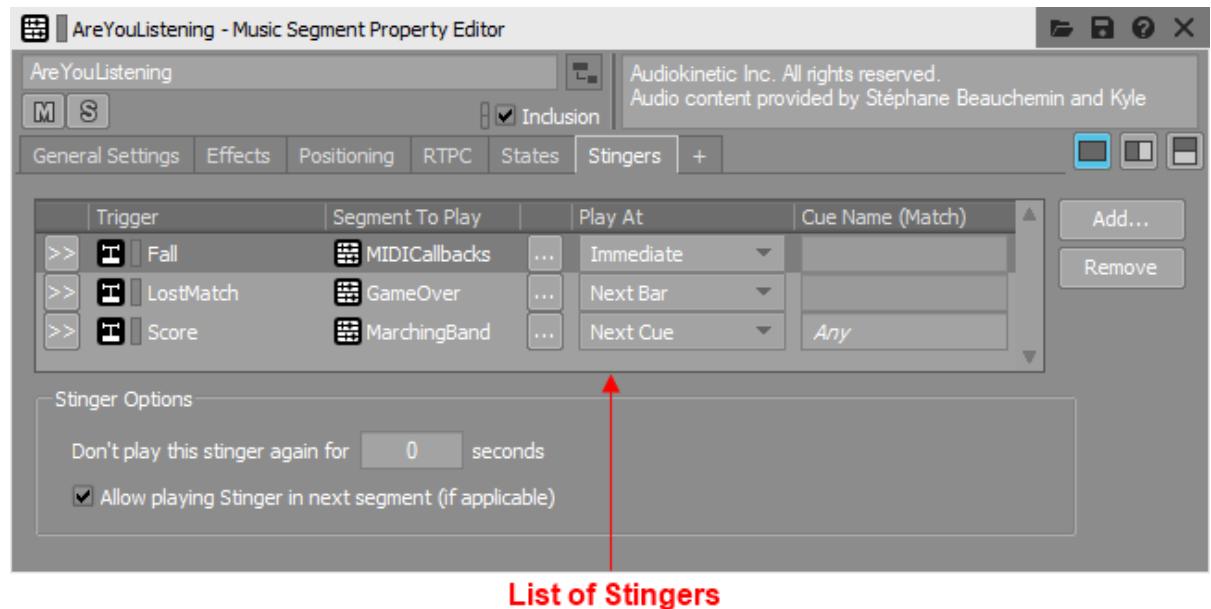


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

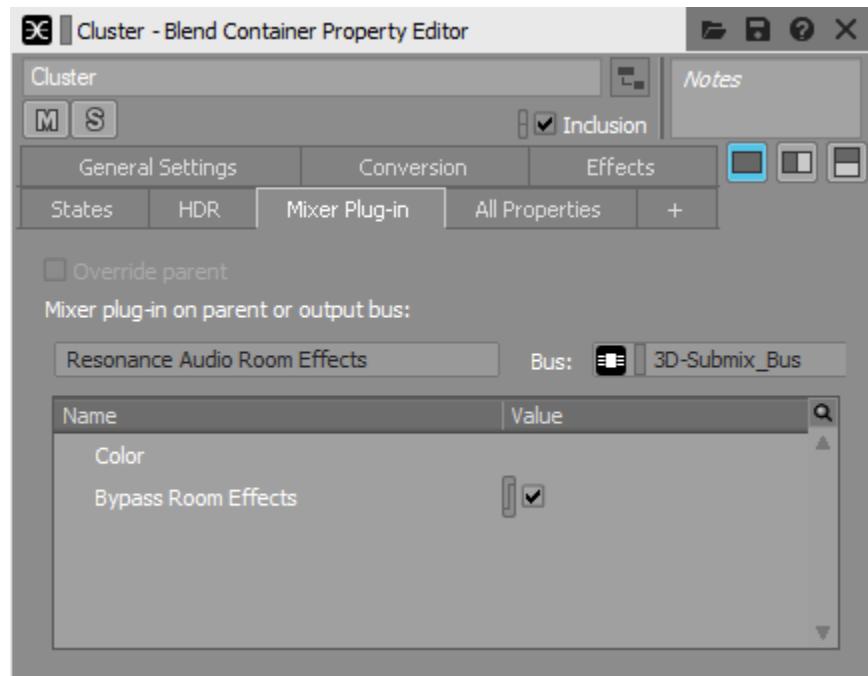
Stingers

在 Stinger (插播乐句) 选项卡中，您可以创建触发器列表，为它们指定要播放的段落，以及指定播放它们的时间。您还可以为各个 Stinger 设置播放选项。



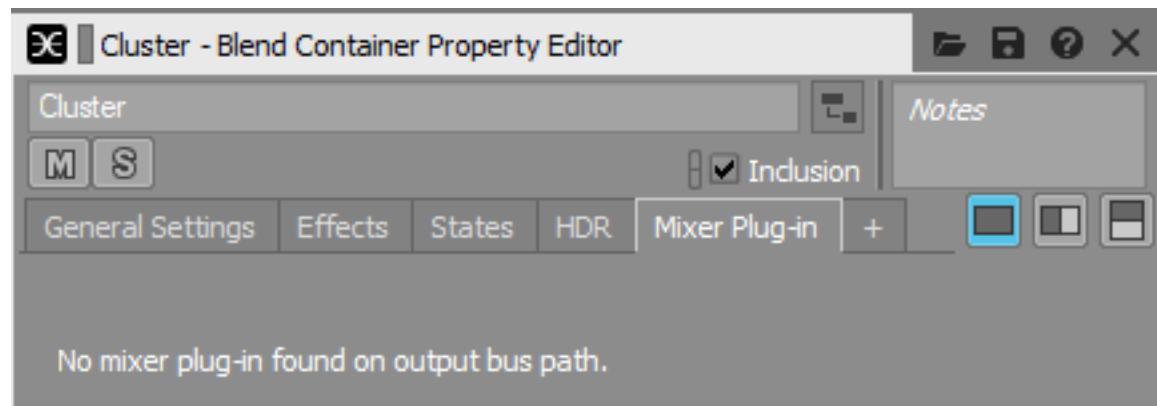
Mixer Plug-in

在总线的 Mixer Plug-in（混音器插件）选项卡中，可以应用混音器插件。对于 Mixer 插件所在的总线，通过其输出的每个音频对象或其他总线都将具有各自的附加属性，可以在 Mixer Plugin 选项卡中根据需要进行调整。可编辑的属性因插件而异。



Attachable Properties displayed only when the bus or object has a parent or output bus with a mixer plug-in

Mixer Plug-in 选项卡，插件的附加属性被应用在对象的总线输出路径中



对于输出路径中没有使用 Mixer Plug-in 的对象，Mixer Plug-in 选项卡将不显示属性

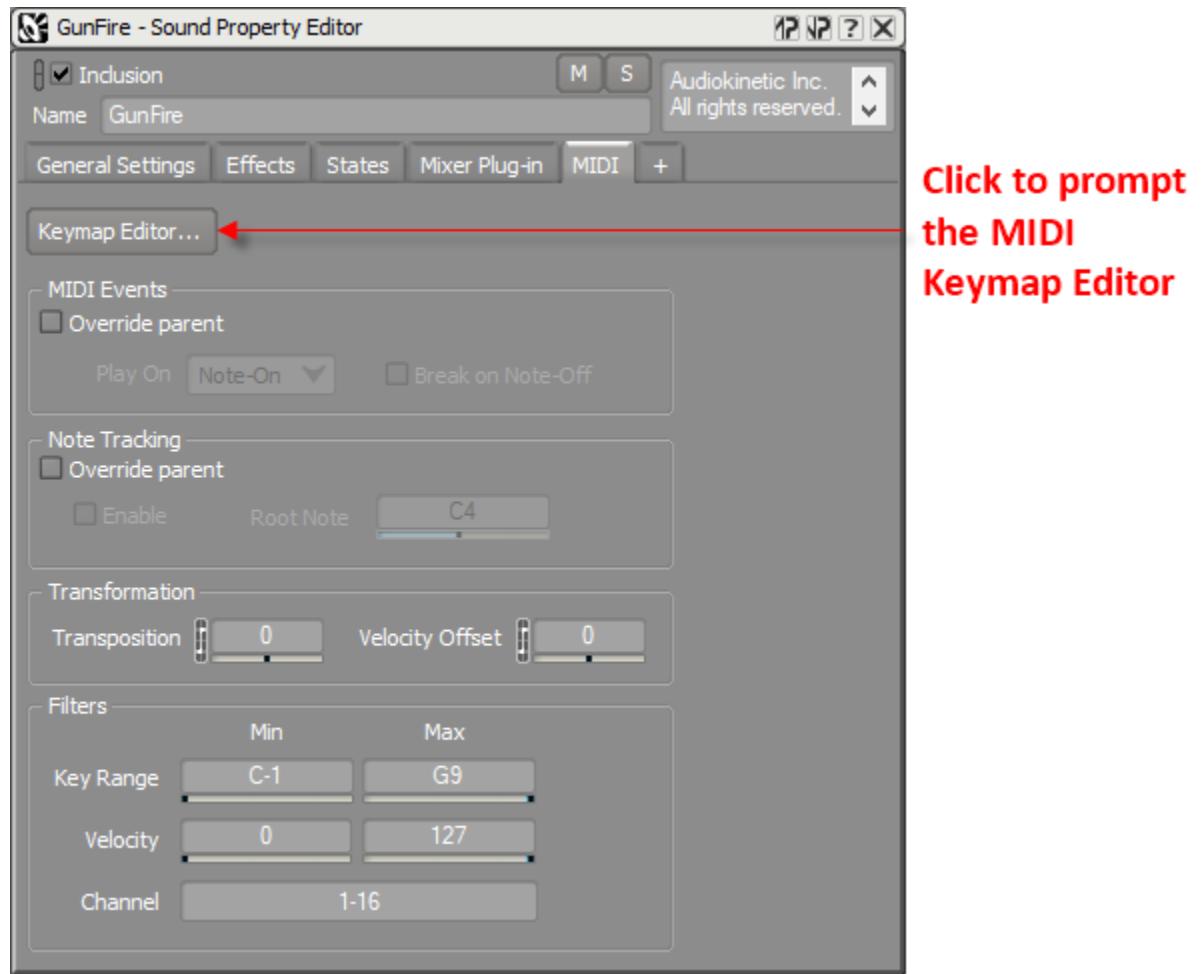


技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

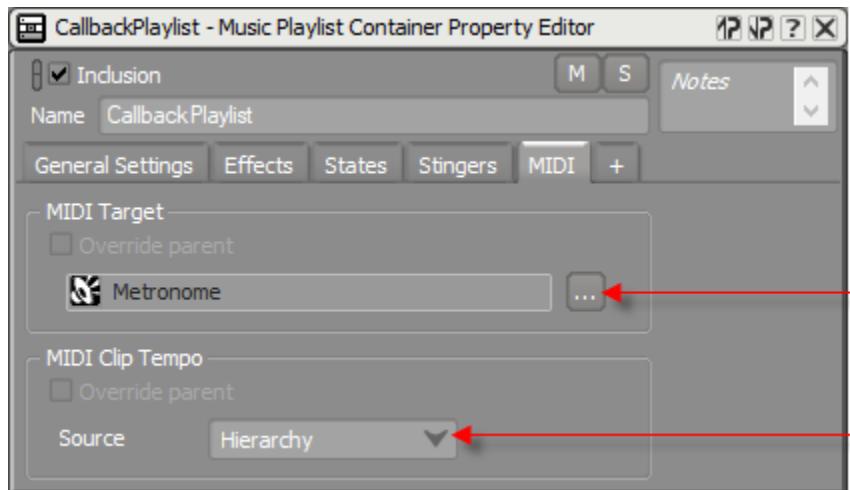
MIDI

对于 Actor-Mixer Hierarchy 中的对象，其 MIDI 选项卡中包含一系列控件，允许您将按照 MIDI 乐器来设置对象行为。



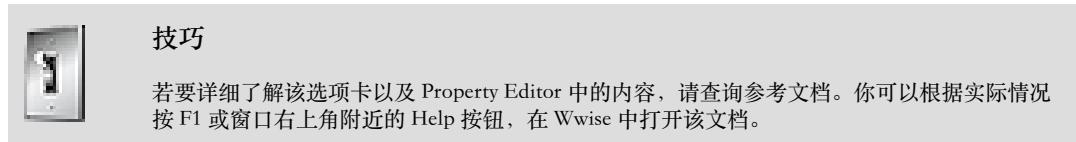
Actor-Mixer Hierarchy 对象的 MIDI 选项卡

对于 Interactive Music Hierarchy 中的对象，其 MIDI 选项卡允许您指定对象的 MIDI 内容速度，以及播放目标。目标可以是 Actor-Mixer Hierarchy 中任何可播放的对象。速度既可指定为 Interactive Music 对象中所有 MIDI 内容的全局值，也可从 MIDI 文件中获取。



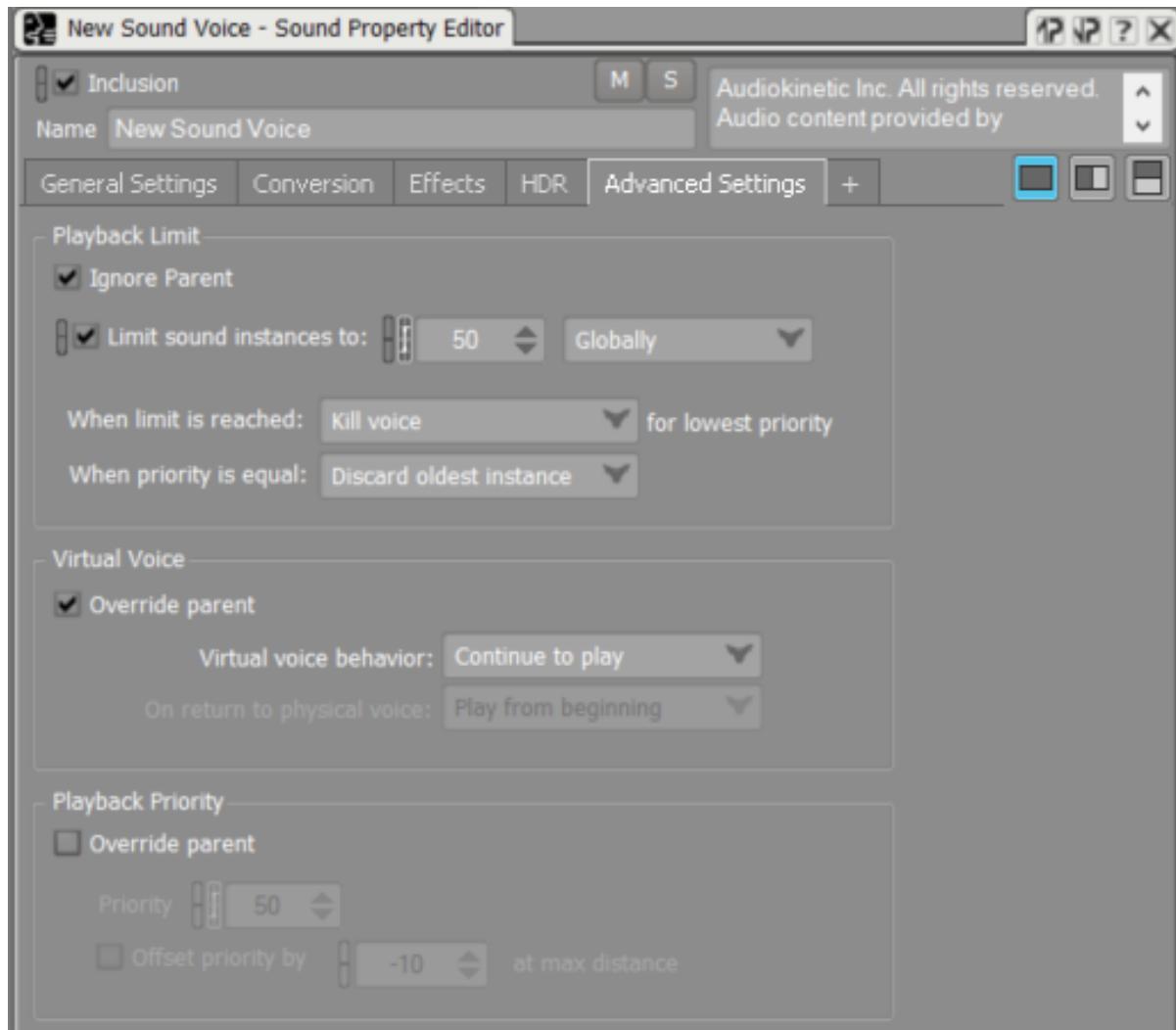
Select any playable object from the Actor-Mixer Hierarchy
Specify the tempo source

Interactive Music Hierarchy 对象的 MIDI 选项卡



Advanced Settings (高级设置)

Advanced Settings 选项卡包含一系列的高级行为选项，它们可以帮助您高效地管理在游戏中同时播放的声音、音乐或振动对象数量、优先播放的对象，以及当对象的音量低于预定音量阈值时的响应。



技巧

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

使用高级设置和动态混音

声音结构的高级设置在对象 Property Editor 的 Advanced Setting 选项卡中指定，可限制同时播放的声音数量和指定它们在无法被听到时的行为，这两种设置分别有以下优点：

- 提升性能和帮助清理混音
- 节省内存和 CPU 资源

高级设置和混音

不应等到生产过程后期再调整高级设置。更准确地说，至少应在混音前或混音期间执行第一轮高级设置调整。如果您调整混音过早，结果在后期发现音频占用资源过多，则您最终可能为了将音频处理保持在合理范围内，而过度使用高级设置，这将对混音造成巨大的影响。

播放数限制实际上可帮助您实现混音。将它们当作动态混音的一种形式进行使用，帮助玩家集中注意力于重要的方面，而不是用大量声音来淹没玩家。您还可以使用总线闪避、Set Volume 动作、State 或 RTPC 来清理您的混音。

播放数限制、优先级和 under Volume Threshold（低于音量阈值）行为

声音结构上使用的 Playback Limit 可帮助您限制同时播放的声音数量，可以限制游戏对象的声音数量，或是全局范围内的声音数量，在 Actor-Mixer 和 Interactive Music Hierarchy 中都可以指定。（在 Master-Mixer Hierarchy 中，限制范围只能是全局）。它的逻辑以播放的声音数量为唯一依据。但是，声音设计师可以在整个层级结构中为不同的对象应用单独的数量限制。而且根据需要，还可以通过选择 Ignore Parent 选项来创建独立于上级数量限制的层级结构。总而言之，多种级别的播放限制能让声音设计师更容易控制所播放的内容。

在试图播放声音之前将检查播放数限制条件。当声音即将开始播放，并已达到播放数限制时，将停止播放此声音或另一个声音。第一个条件是声音的优先级。当两个候选声音具有相同的优先级时，声音引擎停止播放最早或最新的声音，具体由 When limit is reached（达到限制时）和 When priority is equal（优先级相同时）属性指定。



备注

当声音由于播放数限制而被终止时，播放次数低于此限制则不会重新播放该声音。因此需要小心处理无限循环的环境声音。

优先级设置可搭配播放数限制一起使用。在层级结构中微调优先级，以平衡播放数限制系统。不应由播放数限制系统终止的声音（例如配音、背景音乐或循环环境声音）应具有最高的优先级。另外，已经生效的优先级还可以受到声音和听者之间的距离影响。

“under Volume Threshold”行为与播放数限制和优先级无任何关系。它只是决定当声音无法被听见时的声音行为。为了确定声音能否被听见，Wwise 只会查看音量元数据，它是所有层级结构、总线、状态、RTPC 和 Set Voice Volume 动作对音量产生的影响总和。它不分析 WAV 数据。

动态混音技术

在 Advanced Setting 选项卡中可应用下面的动态混音技术。

- 限制总线或主角混音结构上的播放次数，为重要声音留出空间

例如，当有大量动作、爆炸或玩家应关注的任何元素上时，应减少环境声音和拟音的数量。查找环境声和拟音与枪声和爆炸声并存的总线，对该总线设置限制，并降低前者的优先级。

- 对优先级使用距离偏置

例如，在环境声音中，结合基于距离的优先级使用播放数限制，以将焦点集中在更近的声音上。Offset priority by 选项指定最大距离处的优先级偏置值，此值插值于 0 和 at max distance 之间。

- 闪避不重要的声音的音量

有时候不可使用播放数限制。例如，如果您在游戏关卡一开始就启动无限循环的环境声音，则应该避免使用播放数限制功能来停止这些声音，因为通过这种方式停止

后，它们就不会再重新播放了。在这种情况下或者您认为合适的任何情况下，当游戏音频的其它更重要区域中有活动时，应使用需要的技术来闪避它们的音量。例如，在配音或格斗期间关闭环境声音。当炸弹在您附近爆炸时，您无需听到电灯的滋滋声。此过程可视为元数据侧链。使用控制总线上的总线闪避功能或特定事件中的 Set Voice Volume 动作可用来触发音量变化。

在特定场合中，一旦通过降低不重要声音的音量来清理混音后，微调它们的“under Volume Threshold”行为，以在它们无法被听见时使用尽可能少的 CPU 和内存资源。

- 实现代码侧动态混音系统

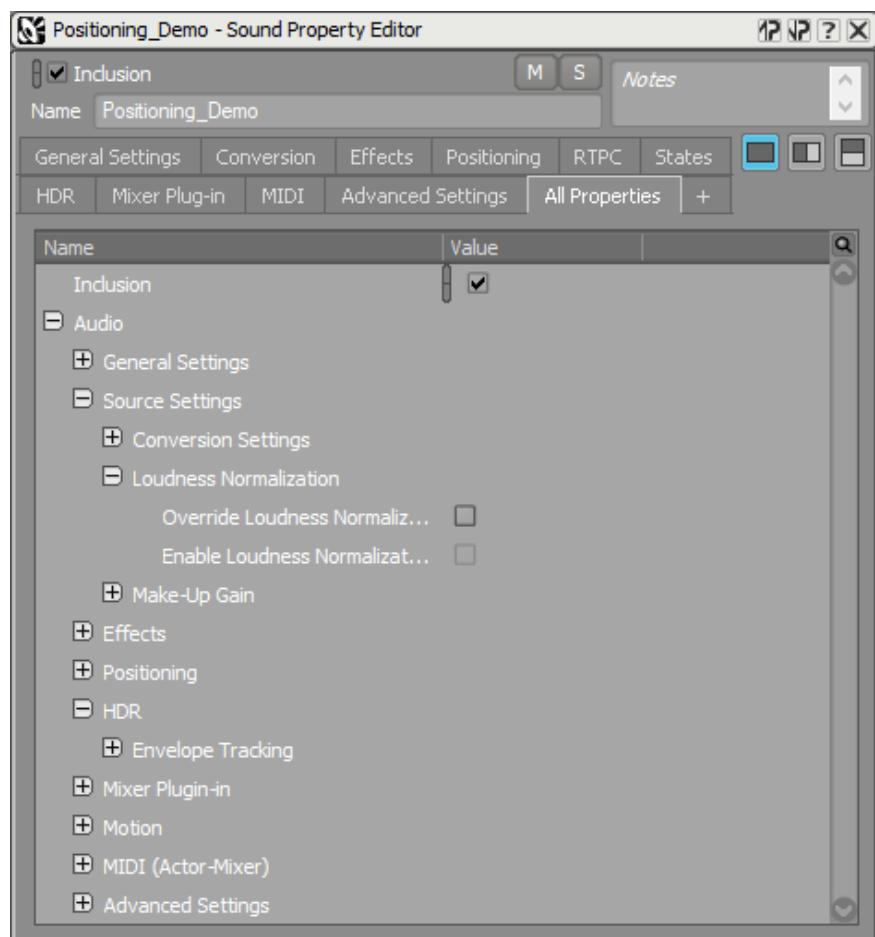
注意，播放数限制、优先级和优先级偏置可通过 RTPC 将控制接口交给游戏。您可以无障碍地实现一个系统，以根据游戏中的情况来调整这些设置。

相关主题

- [使用旁链压缩](#)

All Properties (所有属性)

All Properties 选项卡允许用户编辑对象的任何可编辑属性。如前面的标签页文档所述，可用的属性因对象类型而异。



**技巧**

若要详细了解该选项卡以及 Property Editor 中的内容，请查询参考文档。你可以根据实际情况按 F1 或窗口右上角附近的 Help 按钮，在 Wwise 中打开该文档。

使用 Property Editor

Property Editor 可包含一系列的选项、字段、滑杆、列表、按钮和坐标图视图，您可以用它们定义工程中不同对象和 Game Sync 的属性和行为。如果需要这些工具的使用帮助，请参阅以下各节：

- [使用文本框](#)
- [使用列表](#)
- [使用滑杆](#)
- [第 43 章 了解坐标图视图](#)

在 Property Editor 中修改属性值时，您可以选用随机化器或跨平台链接或取消链接该值。在 Wwise 中，这些功能通过独特的图标进行表示。Wwise 还通过图标指示某属性值是否通过 RTPC 指派到了某个游戏参数上。下表介绍了这些图标。

图标	名称	描述
	Inclusion	决定是否包含该对象。如勾选，则包含该元素。如未勾选，则不会包含该元素。在默认情况下，此设置将应用于所有平台。可以使用 Link Indicator （链接图标，位于复选框左侧）来查看或设置平台专有属性。 未勾选此选项时，Property Editor（属性编辑器）中各属性和行为选项都将不可用。
	Link	链接。属性值已链接到其它有效游戏平台的值。
	Unlink	取消链接。属性值没有链接到其它有效游戏平台的值。
	Partial Unlink	部分取消链接。当前平台的属性值已链接到其它有效平台，但其它平台的若干个相应值已取消链接。
	Link Mixed	有些选定的对象具有不同的链接状态。有些可能是链接的，而另一些则是取消链接或部分取消链接的。
	RTPC 已禁用	该属性值未绑定至游戏内参数值。
	RTPC 已启用	游戏内参数值已绑定至该属性值。这意味着，例如游戏赛车的速度可直接绑定至 Wwise 中的音调属性。当游戏中的赛车速度提高时，Wwise 中的音调也将实时提高。
	RTPC 部分启用	Multi Editor 中只有部分对象为该属性绑定了游戏参数值。Property Editor 或 Contents Editor 中不会看到这个标识。
	Randomizer 已启用	随机化器效果已应用到的属性值。
	Randomizer 已禁用	尚未应用随机化器效果的属性值。
	随机化器 Mixed	Multi Editor 中只有部分对象为该属性值启用了 Randomizer 效果。Property Editor 或 Contents Editor 中不会看到这个标识。
	State 已禁用	此属性值未与 State 绑定。

图标	名称	描述
	State 已启用	State Group 已与此属性值绑定。也就是说，所述属性（如 Volume）可能会随应用的 State 变化。
	State 混合情形	State Group 绑定到了 Multi Editor 中加载的一个或多个对象（并非全部）的属性值。Property Editor 或 Contents Editor 中不会看到这个标识。

随机化属性值

为了增强游戏中的声音真实感，您可以在 Property Editor 中对大多数属性值应用随机化器。随机化器指定可用于特定属性的可能值域。每次播放对象时，Wwise 将在指定的值域内选择不同的属性值。对各个不同的属性值应用随机化器将确保每次在播放对象时对象听起来会不相同。

有关随机化器和如何使用它们的详细信息，请参阅[通过随机化属性值来改善播放](#)。

链接/取消链接属性值

Wwise 还可以让您同时针对多个平台进行制作，其方法是链接（Link）和取消链接（Unlink）与对象相关联的大多数属性值。默认情况下，对于所有活动平台，所有属性值都是链接在一起的，因此属性在各平台上的值相同。取消链接属性值可让您为特定平台自定义属性值。



有关链接和取消链接属性值的详细信息，请参阅[根据平台自定义对象属性](#)。

将属性值指定到游戏参数

如上所述，您可以将游戏参数指定到 Wwise 中的特定属性值。它们被称为实时参数控制（RTPC）。Wwise 为您提供视觉反馈来提示点哪些属性已指派给了 RTPC。在将属性值指派给 Game Parameter（游戏参数）后，属性的 RTPC 标志会变为主题特定颜色（Classic 中为蓝色，Dark 中为橙色）。



有关使用 RTPC 的详细信息，请参阅[使用 Game Parameter 控制属性值](#)。

显示工程元素的属性

Property Editor 显示与工程中特定工程元素相关的所有属性。您可以在许多不同的视图（例如 Project Explorer 的 Audio 选项卡、Contents Editor 和 Capture Log）中通过双击对象将特定对象加载到 Property Editor。您还可以在 Project Explorer 的 Game Syncs 选项卡中通过双击下列任何一项将 Game Sync 加载到 Property Editor 中：

- Switch Group (切换开关组)
- Switch (切换开关)
- State Group
- State (状态)
- Game Parameter (游戏参数)
- Trigger (触发器)

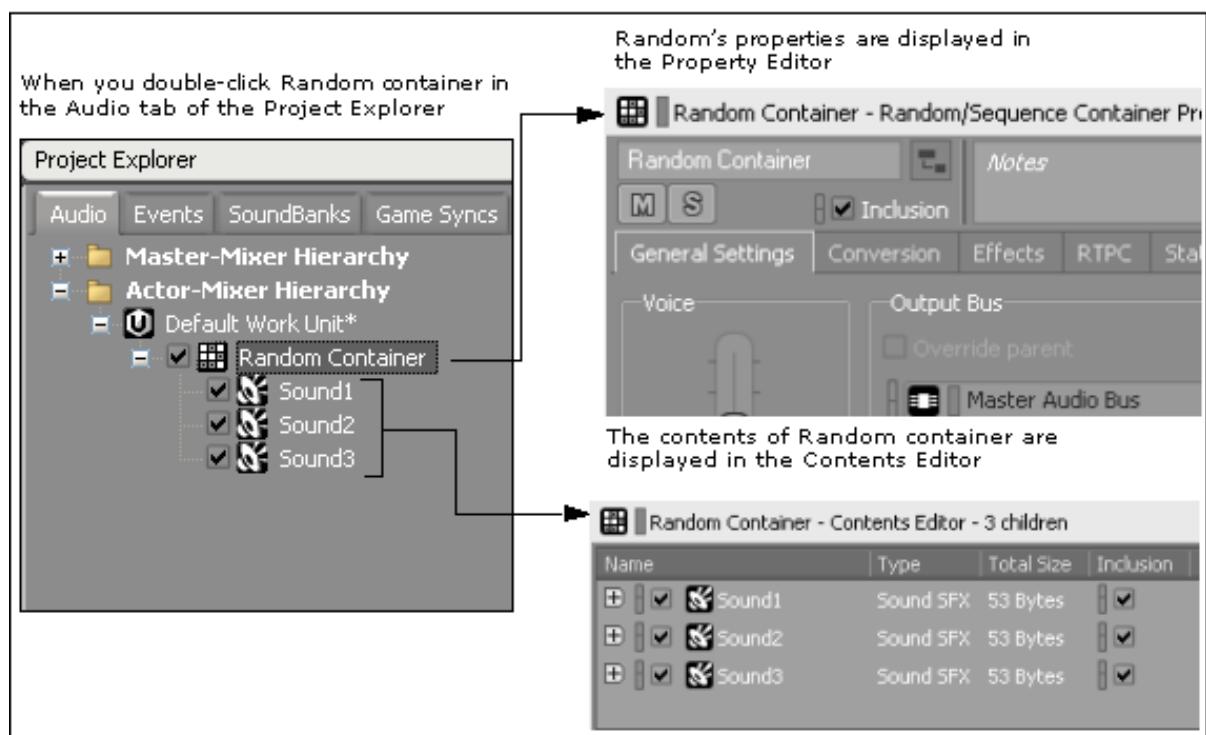
在 Designer 布局中显示工程元素属性的方法是：

1. 在 Designer 布局中，执行以下操作之一：

- 在 Project Explorer 的 Audio 选项卡中双击对象。
- 在 Project Explorer 的 Game Syncs 选项卡中双击某个 Game Sync。

此时将发生下列情况：

- 对象的属性将显示在 Property Editor 中。
- 对象或 Game Sync 内包含的所有子对象将显示在 Contents Editor 中。



第 40 章 认识 Contents Editor 视图

概述	727
使用 Contents Editor	734

概述

父级对象加载到 Property Editor (属性编辑器) 时, Contents Editor (内容编辑器) 中会显示其子对象。Property Editor 可以加载不同对象结构和其它工程元素, 因此 Contents Editor 也会根据加载的对象类型显示不同的布局。

当声音、音乐或振动结构和其它工程元素加载到 Contents Editor 时, 您可以快速访问与各对象相关的常见属性, 例如音量。通过在 Contents Editor 中进行设置, 可以直接编辑子对象, 而不必将其再分别加载到 Property Editor 中。Contents Editor 中还允许启用或弃用对象、设置 Playlist (播放列表) 和 Switch (切换开关) 行为, 以及管理音频、振动源和源插件。

下面的章节介绍了声音、音乐和振动结构及其它工程元素在 Contents Editor 中的不同视图。

- [Sound Object](#)
- [Random Container \(随机容器\)](#)
- [Sequence Container \(序列容器\)](#)
- [Switch Container](#)
- [Blend Container](#)
- [Actor-Mixer](#)
- [Folder](#)
- [Bus](#)
- [Event](#)
- [Switch Group \(切换开关组\)](#)
- [State Group](#)
- [Music Track \(音乐轨\)](#)
- [Music Segment \(音乐段落\)](#)
- [Music Playlist Container \(音乐播放列表容器\)](#)
- [Music Switch Container](#)

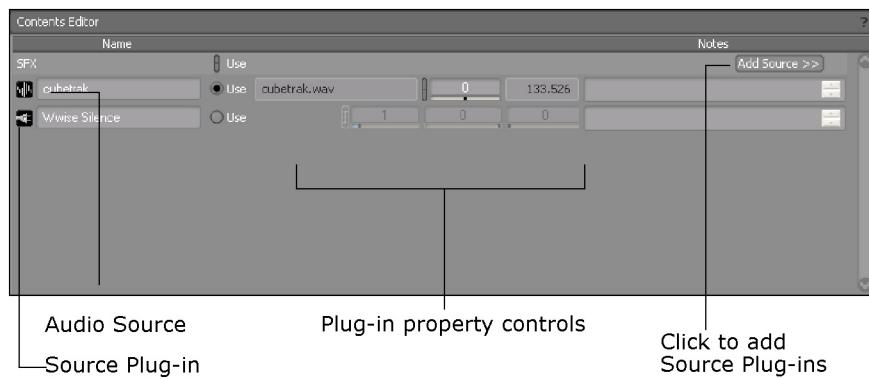
Sound Object

当您将 Sound Object (声音对象) 加载到 Property Editor 时, 它的 Source (源) 将显示在 Contents Editor 中。声音对象可以包含多个源, 包括:

- **Take** -- 版本。选择使用哪个源之前, 您可以针对同一声音对象的不同版本进行试听和测试。这些音频源可以链接到音频文件、Silence (空白) 源、插件或三者的组合。
- **Language** —— 工程的不同本地化语言版本。关于如何在 Wwise 中使用语言版本, 详细信息请参阅[对工程进行本地化](#)。

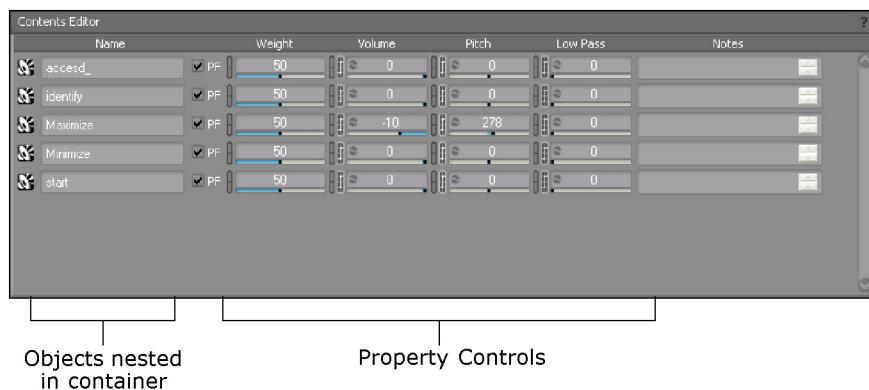
Contents Editor 中, 对于每种不同的源类型, 其控件和属性的显示也不同。

认识 Contents Editor 视图



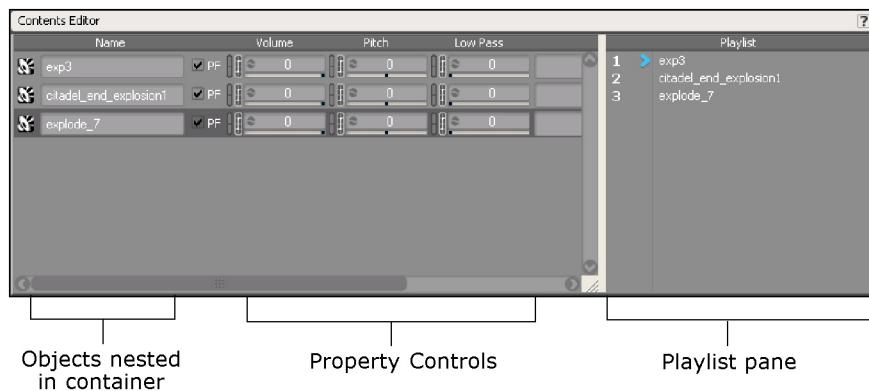
Random Container (随机容器)

当您将 Random Container (随机容器) 加载到 Property Editor 时，它的子对象将显示在 Contents Editor 中，可以在其中编辑各对象的属性。



Sequence Container (序列容器)

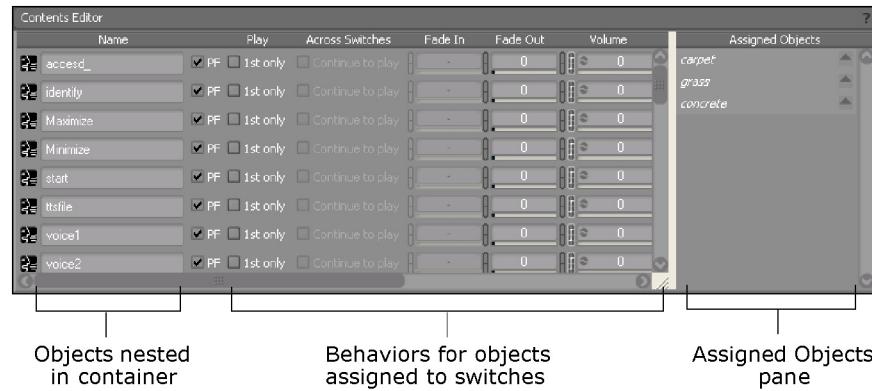
当您将 Sequence Container (序列容器) 加载到 Property Editor 时，子对象将显示在 Contents Editor 中。可以编辑各对象的属性，并为 Sequence Container 创建播放列表。



认识 Contents Editor 视图

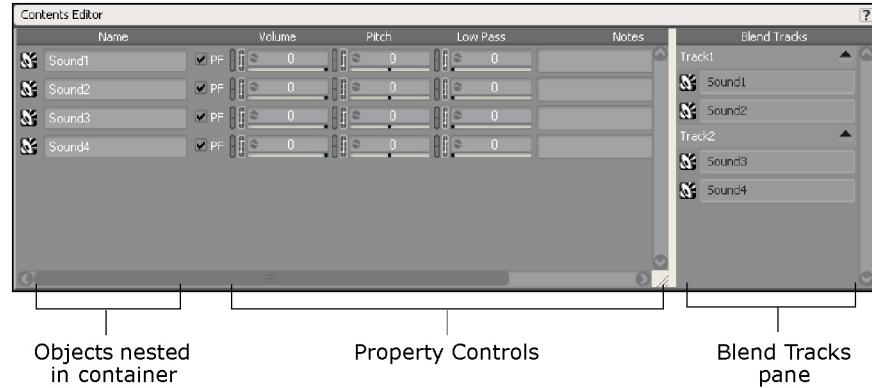
Switch Container

当您将 Switch Container（切换容器）加载到 Property Editor 时，子对象将显示在 Contents Editor 中。可以编辑子对象的属性，将它们指派给 Switch，并设置各对象在游戏调用 Switch 时的行为。



Blend Container

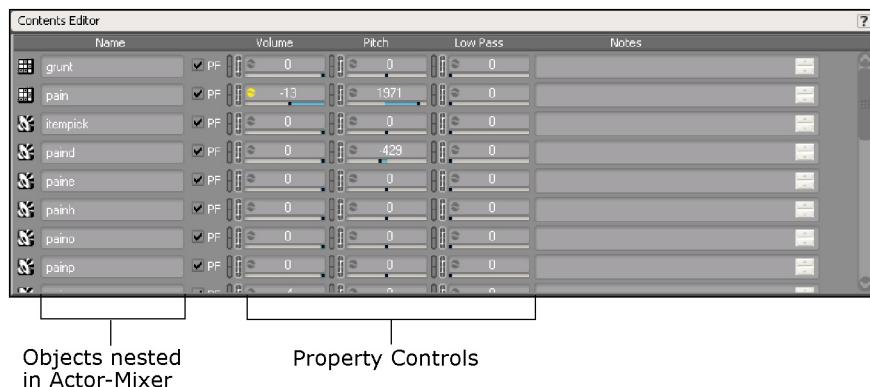
当您将 Blend Container（混合容器）加载到 Property Editor 时，子对象将显示在 Contents Editor 中。可以编辑子对象的属性，并将它们分配至不同的 Blend Track（混合轨）。



Actor-Mixer

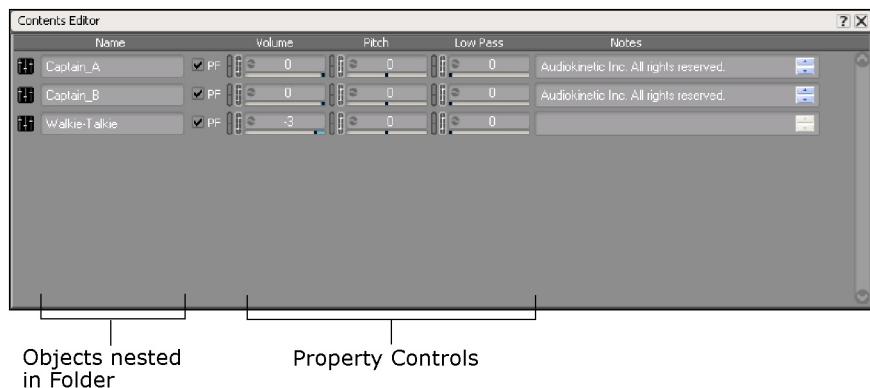
当您将 Actor-Mixer（角色混音器）加载到 Property Editor 中时，子对象将显示在 Contents Editor 中，可以在其中编辑 Actor-Mixer 子对象的属性。

认识 Contents Editor 视图



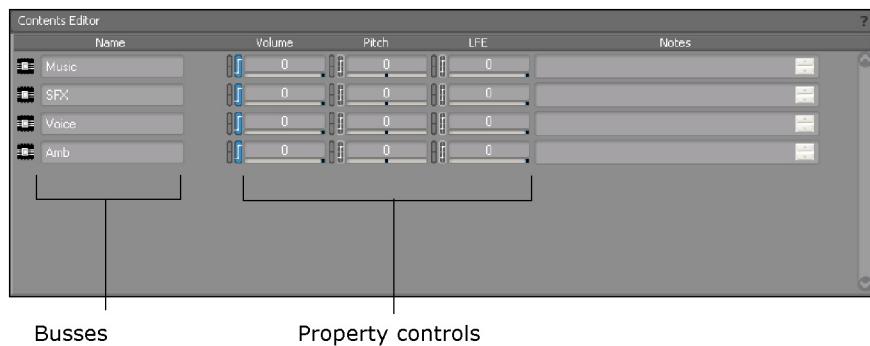
Folder

当您将 Folder (文件夹) 加载到 Property Editor 时，子对象将显示在 Contents Editor 中，可以在其中编辑子对象的属性。



Bus

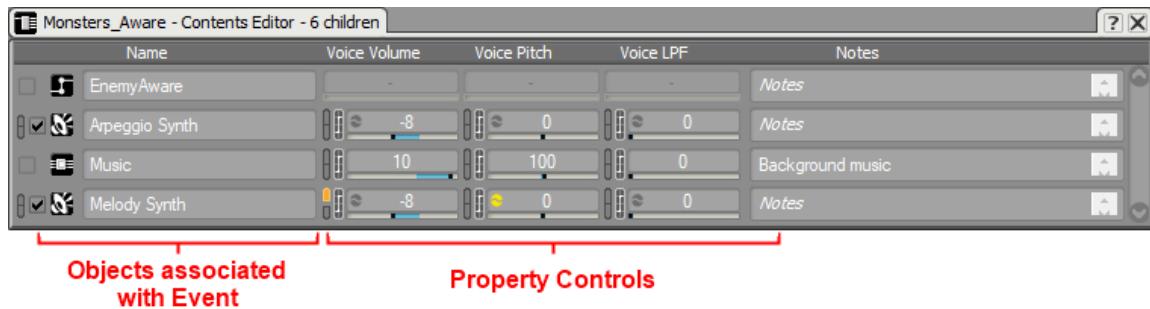
当您将 Master Bus (主总线) 或父级总线加载到 Property Editor 中时，其子总线将加载到 Contents Editor 中，可以在其中编辑子总线属性。



Event

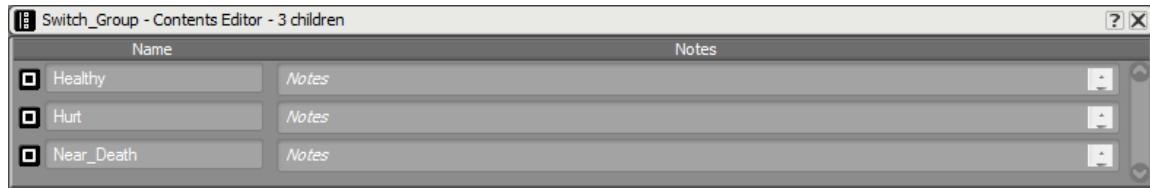
当您将 Event (事件) 加载到 Event Editor 中时，事件关联的对象将显示在 Contents Editor 中，可以编辑它们的属性。

认识 Contents Editor 视图



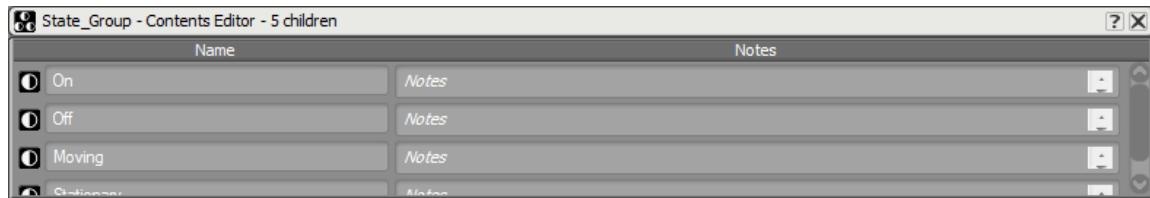
Switch Group (切换开关组)

当您将 Switch Group (切换开关组) 加载到 Property Editor 时，其中的 Switch 将显示在 Contents Editor 中。可以编辑各个 Switch 的名称，并添加备注。



State Group

当您将 State Group (状态组) 加载到 Property Editor 时，其中的状态将显示在 Contents Editor 中，可以编辑它们的属性。



Music Track (音乐轨)

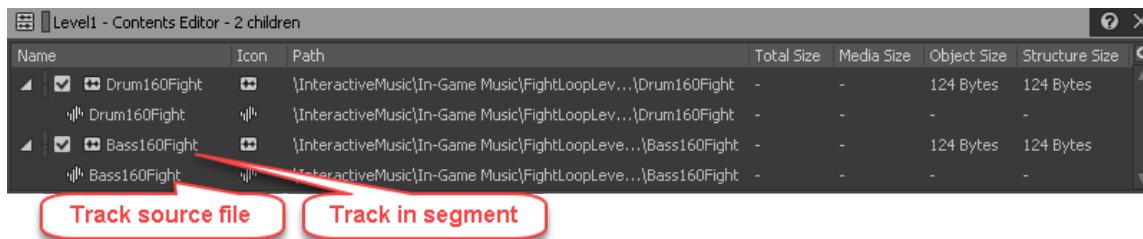
当您将 Music Track (音乐轨) 加载到 Property Editor 中时，音轨将显示在 Contents Editor 中。虽然也会显示属性，但只有备注可以编辑。

认识 Contents Editor 视图



Music Segment (音乐段落)

当您将 Music Segment (音乐段落) 加载到 Property Editor 中时，其中的音轨将显示在 Contents Editor 中，可以在此编辑其属性。



Music Playlist Container (音乐播放列表容器)

当您将 Music Playlist Container (音乐播放列表容器) 加载到 Property Editor 中时，其子对象将显示在 Contents Editor 中，可以在此编辑其属性。

认识 Contents Editor 视图

The screenshot shows the Contents Editor interface with a list of 15 objects in a playlist. Each object has columns for Name, Volume, LFE, and Low Pass, along with a checkbox for PF and notes. The objects are:

Name	Volume	LFE	Low Pass	Notes
Intro_01	0	0	0	Audiodkinetic Inc. All rights reserved.
Intro_02	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 01a (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 01b (B)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 01d (B)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 01c (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 02a (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 02b (B)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 02c (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 02d (B)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 03a (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 03b (B)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 03c (A)	0	0	0	Audiodkinetic Inc. All rights reserved.
Seg 03d (B)	0	0	0	Audiodkinetic Inc. All rights reserved.

Below the table, two labels with arrows point to specific areas: "Objects in playlist" points to the list of objects, and "Property Controls" points to the columns of sliders.

Music Switch Container

当您将 Music Switch Container（音乐切换容器）加载到 Property Editor 中时，其子对象将显示在 Contents Editor 中，可以在此编辑其属性。

The screenshot shows the Contents Editor interface with a list of 8 objects within a container. The objects are:

Name	Volume	LFE	Low Pass	Notes
Stealth	-5	0	0	Audiodkinetic Inc. All rights reserved.
Fight	0	0	0	Audiodkinetic Inc. All rights reserved.
Bonus Music Punches	-3	0	0	Audiodkinetic Inc. All rights reserved.
Stress	-5	0	0	Audiodkinetic Inc. All rights reserved.
Death	0	0	0	
Interlude	-3	0	0	
Fight_to_Menu_Transition	0	0	0	
Outro_to_Stealth_or_Stress	0	0	0	Audiodkinetic Inc. All rights reserved.

Below the table, two labels with arrows point to specific areas: "Objects in container" points to the list of objects, and "Property Controls" points to the columns of sliders.

使用 Contents Editor

Contents Editor（内容编辑器）中显示了一系列字段、列表、选项和滑杆，可以用它们设置对象、State 和 Switch 的属性和行为。如果需要这些工具的使用帮助，请参阅以下各节：

- [使用文本框](#)
- [使用列表](#)
- [使用滑杆](#)

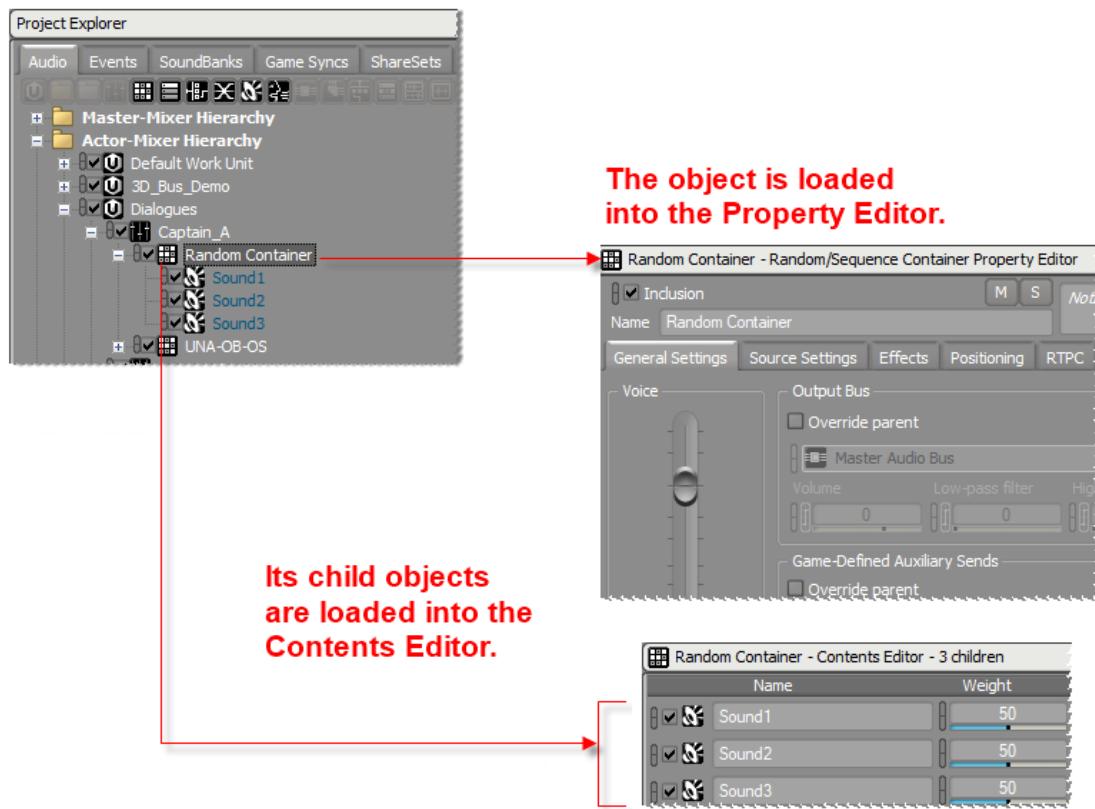
在 Contents Editor 中，您可以播放每个对象，也可以重新排序、复制、粘贴和删除视图中的对象。根据当前加载的对象类型，视图中显示的选项或属性将有所不同。多数视图中，一些属性滑杆旁边会显示特定图标，来表示它们具有特定属性。下表介绍了这些图标。

图标	名称	描述
	Link	链接。属性值已链接到其它有效游戏平台的值。
	Unlink	取消链接。属性值没有链接到其它有效游戏平台的值。
	Partial Unlink	部分取消链接。当前平台的属性值已链接到其它有效平台，但其它平台的若干个相应值已取消链接。
	Link Mixed	有些选定的对象具有不同的链接状态。有些可能是链接的，而另一些则是取消链接或部分取消链接的。
	RTPC 已禁用	该属性值未绑定至游戏内参数值。
	RTPC 已启用	游戏内参数值已绑定至该属性值。这意味着，例如游戏赛车的速度可直接绑定至 Wwise 中的音调属性。当游戏中的赛车速度提高时，Wwise 中的音调也将实时提高。
	RTPC 部分启用	Multi Editor 中只有部分对象为该属性绑定了游戏参数值。Property Editor 或 Contents Editor 中不会看到这个标识。
	State 已禁用	此属性值未与 State 绑定。
	State 已启用	State Group 已与此属性值绑定。也就是说，所述属性（如 Volume）可能会随应用的 State 变化。
	State 混合情形	State Group 绑定到了 Multi Editor 中加载的一个或多个对象（并非全部）的属性值。Property Editor 或 Contents Editor 中不会看到这个标识。
	Randomizer 已启用	随机化器效果已应用到的属性值。
	Randomizer 已禁用	尚未应用随机化器效果的属性值。
	随机化器 Mixed	Multi Editor 中只有部分对象为该属性值启用了 Randomizer 效果。Property Editor 或 Contents Editor 中不会看到这个标识。

在 Contents Editor 中显示对象

当层级结构中的对象加载到 Property Editor（属性编辑器）中时，其子对象将显示在 Contents Editor（内容编辑器）中。例如，如果您在 Project Explorer 的 Audio 选项卡中选择 Random Container（随机容器）并加载到 Property Editor，它的子对象将加载到 Contents Editor。

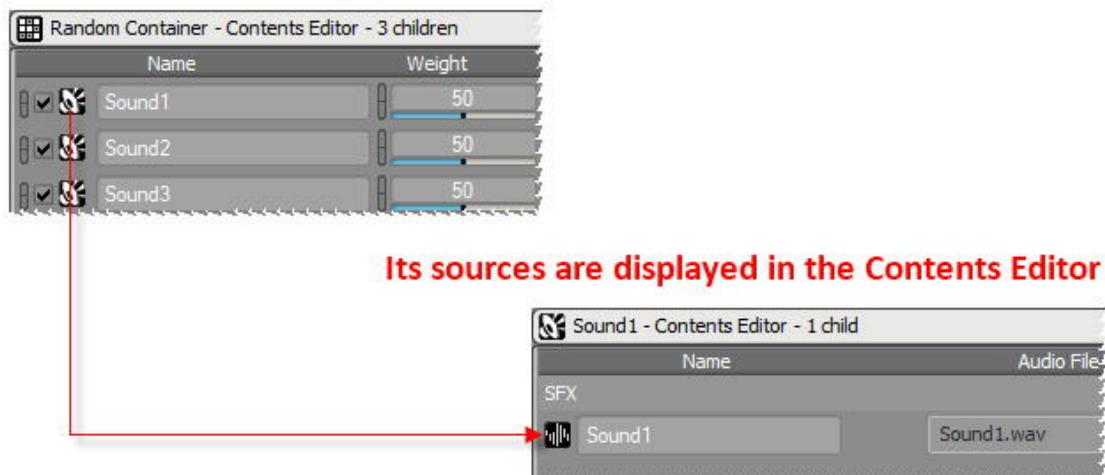
Double-click an object in the Audio tab of the Project Explorer.

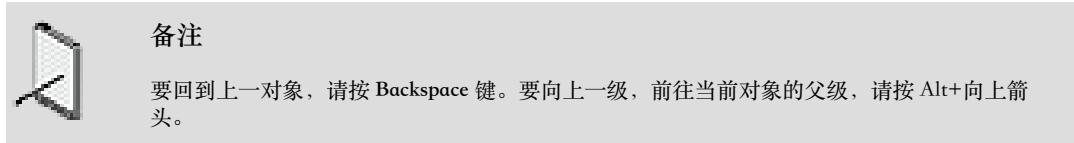


您还可以在 Contents Editor 中双击对象，直接浏览工程层级的下级结构。

可以继续沿工程树向下展开，找到与对象关联的源。

Double-click an object in the Contents Editor



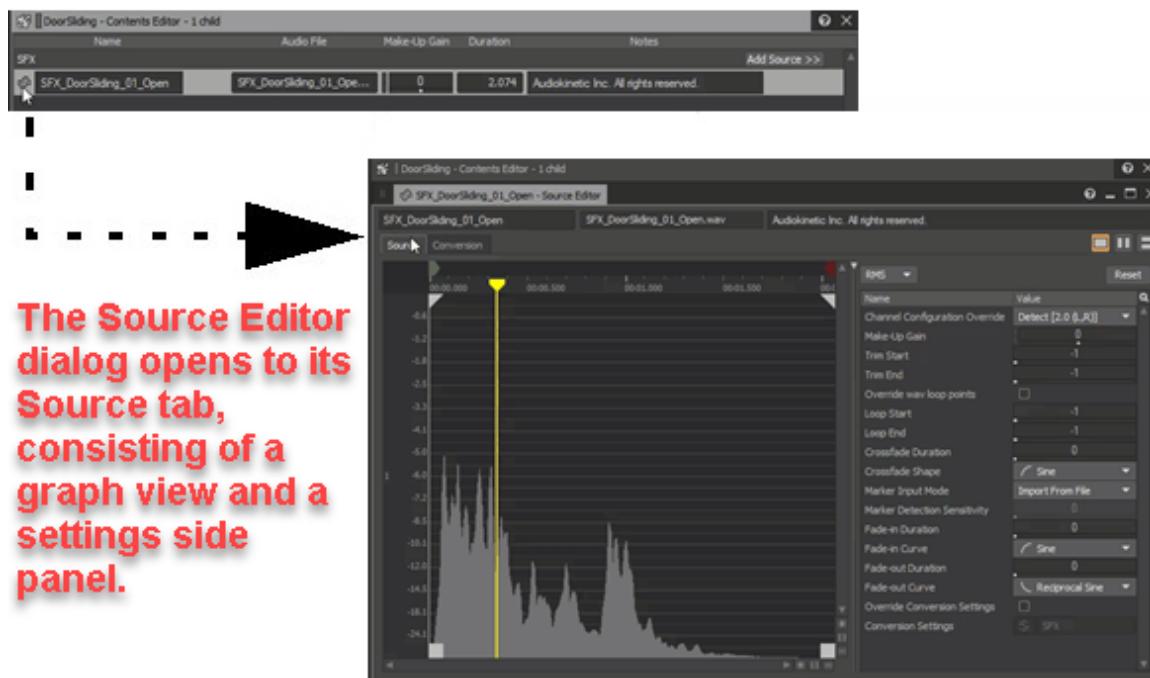


在 Contents Editor 中使用源

在 Contents Editor 中，一旦到了源层级，就可以进行音频源或源插件设置。

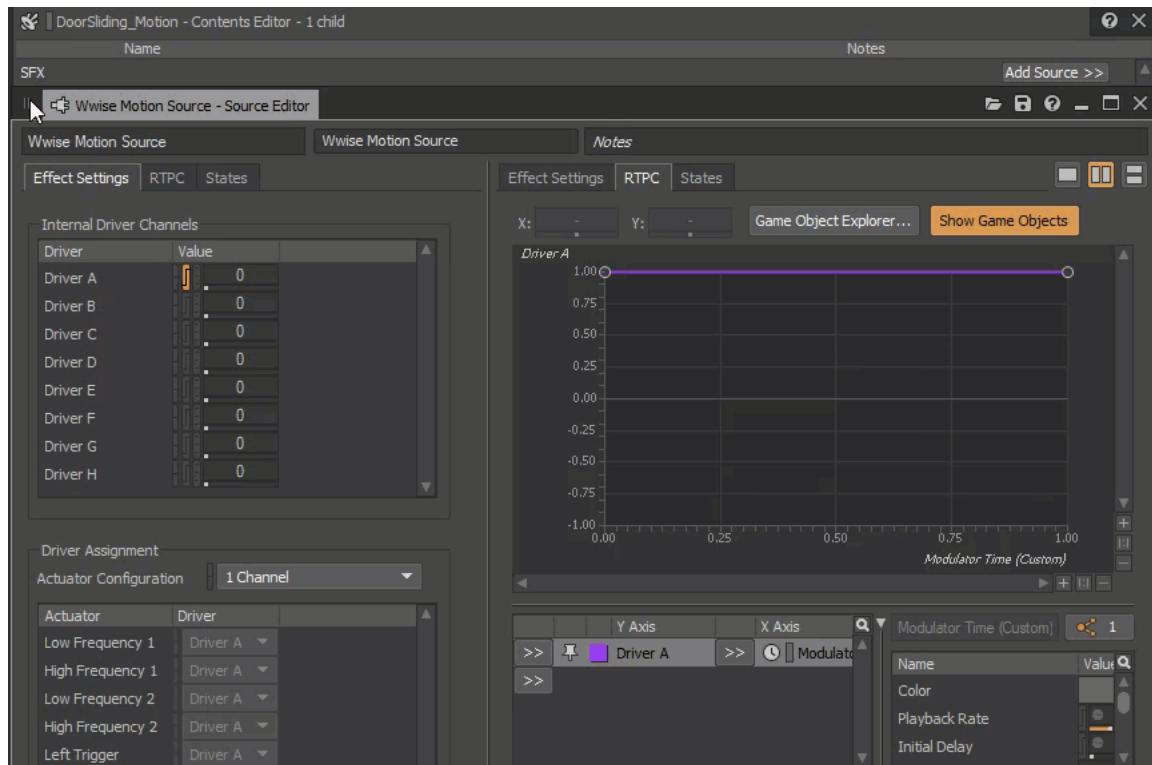
双击音频源时，Source Editor 对话框将打开。您可以在坐标图或侧面板中编辑音频源。另外，还可使用 Conversion (转码) 选项卡来调节转码设置。

Double-click the source in the Contents Editor.



双击源插件时，插件属性将显示在 Property Editor 中，可以在其中编辑属性。

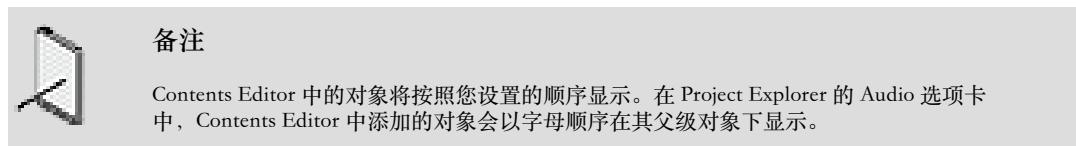
认识 Contents Editor 视图



将对象添加到 Contents Editor

对象加载到 Property Editor 中时，它的内容将显示在 Contents Editor 中。也可以将对象直接添加到 Contents Editor 中，将它们从 Project Explorer 拖放到 Contents Editor 中即可。记住，将对象拖放到 Contents Editor 中时，您是在将其从层级结构中的当前位置移到新位置，即 Contents Editor 中其它对象的父级对象下面。如果不想移动对象，可以按住 Ctrl 同时将对象拖拽到 Contents Editor 中，这样会复制对象，并将其添加到 Contents Editor 中。

源也可以添加到对象，可以直接将其导入到 Contents Editor 中或添加源插件。



对 Contents Editor 中的对象重新排序

您可以根据需要更改 Contents Editor 中对象或源的顺序。这将不会影响它们在 Project Explorer 中的顺序。

更改对象顺序的方法如下：

1. 在 Contents Editor 中，将您要移动的对象拖放到新位置。

红色标志线将显示在对象插入位置的上方。

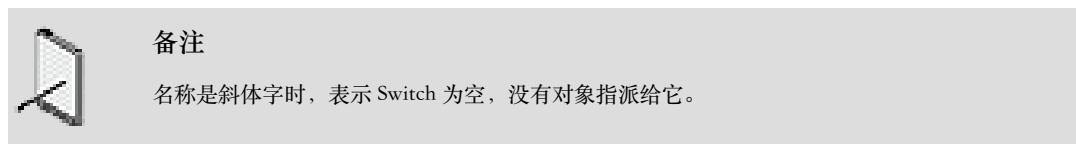
相关主题

- [将对象添加到 Contents Editor](#)
- [在 Contents Editor 的窗格之间拖动对象](#)
- [展开/折叠列表](#)
- [删除对象](#)
- [在 Contents Editor 中试听对象和源](#)

在 Contents Editor 的窗格之间拖动对象

选中 Switch Container（切换容器）、Sequence Container（序列容器）和 Blend Container（混合容器）对象时，Contents Editor 分成两个独立的窗格。第一个窗格包含容器的子对象，而右侧窗格会包含列表。在创建 Playlist（播放列表），或将对象指派到 Switch（切换开关）或 Blend Track（混合轨）时，您可以将对象从一个窗格拖到另一个窗格。

将对象指派给 Switch 时，可以将对象从 Object 窗格拖放到 Assign Object 窗格，也可以将对象从 Project Explorer 的 Audio 选项卡拖放到 Switch 上。关于使用 Switch 的详细信息，请参阅[定义 Switch Container 的内容和行为](#)。



为 Sequence Container 创建播放列表时，您可以将对象从 Object 窗格拖放到 Playlist 窗格中。关于使用播放列表和 Sequence Container 的详细信息，请参阅[创建播放列表](#)。

将对象指派到 Blend Track 时，您可以将对象从 Object 窗格拖放到 Blend Track 窗格，也可以从 Project Explorer 的 Audio 选项卡中，将对象拖放到混合轨上。关于使用 Blend Track 的详细信息，请参阅[为 Blend Track 添加和移除对象](#)。

展开/折叠列表

为了方便在 Contents Editor 窗格中操作对象，您可以展开或折叠列表。此功能适用范围如下：

- **Switch Container** —— 展开和折叠 Assigned Object 列表。
- **Blend Container** —— 展开和折叠 Blend Tracks 列表。
- **Language** —— 展开和折叠每种语言的源列表。

展开或折叠列表的方法如下：

1. 执行以下操作之一：
 - 点击想要折叠的列表旁边的箭头。
 - 列表将折叠。
 - 点击想要展开的列表旁边的箭头。

列表将展开。

展开或折叠所有语言列表：

1. 右键点击任一语言标题栏。

快捷菜单将打开。

2. 执行以下操作之一：

- 要展开所有语言列表，请点击 **Expand All**。
- 要折叠所有语言列表，请点击 **Collapse All**。

相关主题

- [对 Contents Editor 中的对象重新排序](#)
- [在 Contents Editor 的窗格之间拖动对象](#)
- [将对象添加到 Contents Editor](#)
- [删除对象](#)
- [在 Contents Editor 中试听对象和源](#)

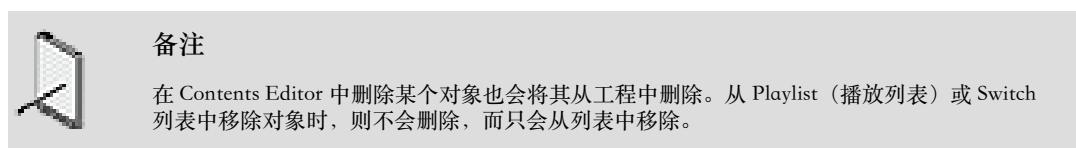
删除对象

如果不再需要某个对象或源，您可以在 Contents Editor 中删除它。处于测试目的，您可以导入多个源，决定需要哪个版本后再删除其余的源。在 Contents Editor 中删除对象或源时，也会在工程中删除它们。此操作不会自动删除工程 .cache 文件夹中的相关已转码音频文件。要删除 Orphan File（落单文件），您需要清空音频缓存。关于如何管理这些 Orphan File 的详细信息，请参阅[清除缓存](#)。

在 Contents Editor 中删除对象的方法如下：

1. 选择您要删除的对象所对应的图标。
2. 执行以下操作之一：
 - 按 **Delete** 键。
 - 右键点击对象，并在快捷菜单中选择 **Delete**。

选定的对象或源将从 Contents Editor 和 Project Explorer 的 Audio 选项卡中移除。



相关主题

- [对 Contents Editor 中的对象重新排序](#)
- [在 Contents Editor 的窗格之间拖动对象](#)
- [展开/折叠列表](#)
- [将对象添加到 Contents Editor](#)

- 在 Contents Editor 中试听对象和源

在 Contents Editor 中试听对象和源

在 Contents Editor（内容编辑器）中，您可以使用 Transport Control（播放控制）试听各对象和源。如果想确定声音或振动对象应该使用哪个源，或者想试听您创建的播放列表，那么试听功能会非常有用。

在 Contents Editor 中试听对象的方法如下：

1. 将对象加载到 Transport Control 中。
2. 在 Transport Control 中，点击 Play 图标。

选定的对象将开始播放。

3. 使用 Transport Control 中的其它控件暂停或停止播放。

关于使用 Transport Control 进行试听的详细信息，请参阅 [第 41 章 认识 Transport Control 视图](#)。

在 Contents Editor 中试听源的方法如下：

1. 将声音、语音、振动或音乐对象加载到 Transport Control。
2. 如果有多个源，请为你要播放的源选中旁边的 Use 选项。
3. 在 Transport Control 中，点击 Play 图标。

选定的源将开始播放。

4. 使用 Transport Control 中的其它控件暂停或停止播放。

关于使用 Transport Control 进行试听的详细信息，请参阅 [第 41 章 认识 Transport Control 视图](#)。

相关主题

- [将对象添加到 Contents Editor](#)
- [对 Contents Editor 中的对象重新排序](#)
- [在 Contents Editor 的窗格之间拖动对象](#)
- [展开/折叠列表](#)
- [删除对象](#)

第 41 章 认识 Transport Control 视图

概述	742
设置播放属性	744
在 Transport Control 中平移对象	746
播放/暂停/停止内容	746
在播放期间使用 Game Sync	747

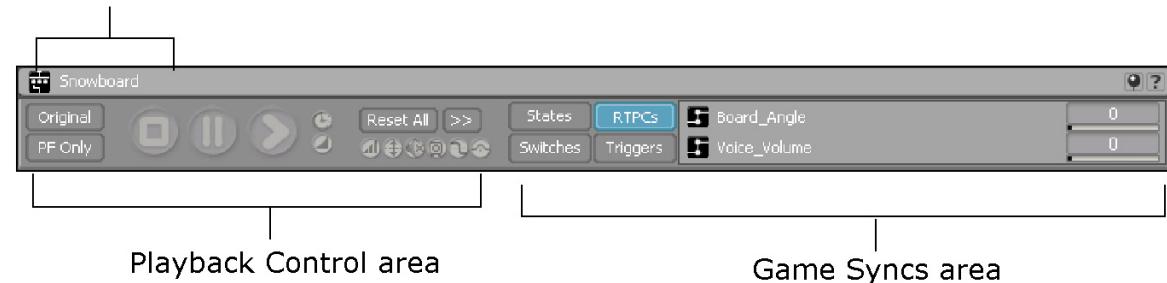
概述

在编辑声音、音乐或振动属性时，您需要能够试听您的作品。在 Wwise 中，当您选择一个声音、容器或事件时，它会自动加载到 Transport Control（播放控制）中，您可以在其中试听它。对象的名称将与它的关联图标一起显示在标题栏。

Transport Control 包括两个不同区域：

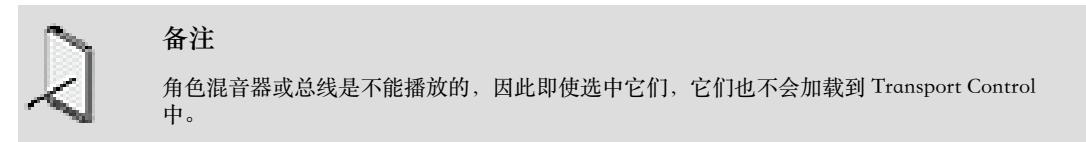
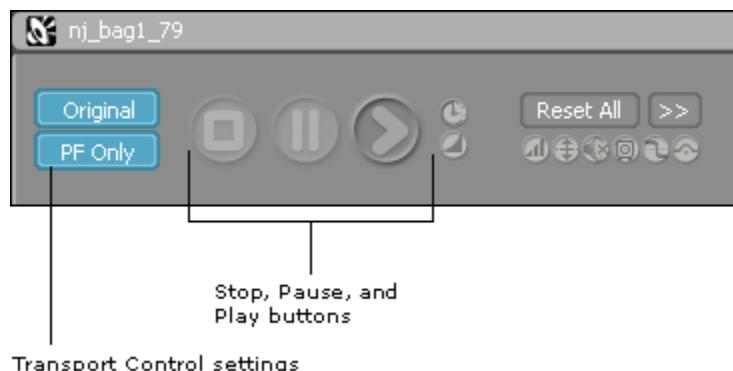
- 播放控制区
- Game Sync 区

Name and icon of
Wwise object



播放控制区

Transport Control 包含与播放音频相关的传统控件，例如播放、停止和暂停按钮。您还可以使用 Transport Control 设置确定播放对象的方式。通过选择或取消选择这些设置，您可以指定是播放原始对象还是经过转码的对象，以及是否播放已从当前平台弃用的对象。



Playback Control 区域还包含一系列标志，当特定属性或行为先前已作用于正在播放的对象时，它们会改变颜色。下表列出了 Transport Control 中的属性和动作参数指示器。

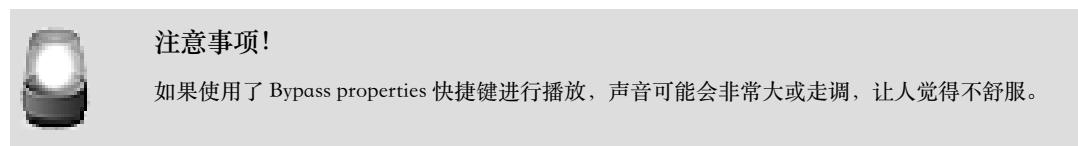
图标	名称	代表
	Delay	已对事件、随机容器和序列容器中的对象应用延迟。
	Fade	已对事件、随机容器或序列容器中的对象应用渐变。
	Set Voice Volume (启用旁通)	对事件中的对象应用了Voice Volume动作。
	Set Pitch (设置音高)	对事件中的对象应用了Set Pitch 动作。
	Mute (静音)	对事件中的对象应用了Mute 动作。
	Set Low-Pass Filter (设置低通滤波器)	已对事件中的对象应用了设定低通滤波动作。
	Enable Bypass (启用旁通)	对事件中的对象应用了启用旁通动作。

有关编辑对象的这些属性的详细信息，请参阅以下各节：

- [播放容器内的所有对象](#)
- [设置事件 Action 的属性](#)

另外，请记得进行声音设计时可以使用 Play-Bypass Properties 快捷方式。

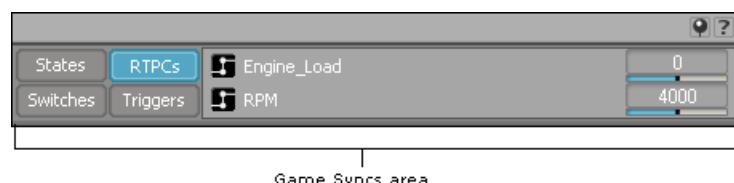
单击 Play 的同时按住 Shift，Wwise 在播放加载对象时将绕过其属性。换言之，与某些 DAW 中的 PFL (Pre-Fader Listen, 推子前监听) 类似，Wwise 在播放对象时不会考虑其层级结构属性（包括音量、音高、滤波器、延迟、Effect、Auxiliary Send、Attenuation Curve、RTPC 曲线、State、定位和总线通路），但会原样保留其行为（如 Fade、Trim、循环点以及随机和序列播放）。



单击 Transport 的 Play 按钮会从播放光标的位置播放片段。当播放光标前进时单击 Play 按钮将添加新的播放光标，并从第一个播放光标的起始位置同时播放。

Game Sync 区

除传统播放控件外，Transport Control 还有一个 Game Sync 区，其中包含与当前选定对象相关的所有 State、Switch、Trigger 和 RTPC。您可以使用 Transport Control 作为微型模拟器，来测试您的声音、音乐和振动，以及模拟在游戏中的变化。在播放期间，您可以在 State 和 Switch 之间切换，并试听游戏参数及其映射值。



有关使用 Game Sync 的详细信息，请参阅以下各节：

- [第 17 章 使用 State](#)
- [管理 Switch/State 的内容](#)
- [管理 RTPC 中使用的 Game Parameter](#)
- [使用 Trigger](#)

设置播放属性

通过启用 Transport Control 中的各种控件，您可以执行以下任何操作：

- [在播放期间使用原始音频文件](#)
- [播放/不播放音频和振动内容](#)
- [重置 Transport Control](#)

在播放期间使用原始音频文件

在转码导入的音频文件时，Wwise 将保留音频文件的原始版本，需要时您可以随时试听它们。原始版本未针对平台进行转码。在默认情况下，Transport Control 播放原始声音；然而，您可以选择播放转码版本。

播放原始声音的方法是：

1. 在 Transport Control 中，点击 Original。

此时该按钮变成蓝色。

2. 点击 Play 图标。

Transport Control 中将播放对象的原始转码前的声音。



相关主题

- [播放/不播放音频和振动内容](#)
- [重置 Transport Control](#)

播放/不播放音频和振动内容

在创建音频、音乐和振动结构时，您可以决定播放或不播放若干个平台中的特定对象。有关使用平台的详细信息，请参阅[从平台中弃用工程元素](#)。当您播放声音、音乐或振动对象时，可以选择仅播放当前平台中的内容，或者播放加载到 Transport Control 中所有平台的声音、音乐或振动对象。

播放平台专有的内容的方法如下：

1. 从工具栏中的 Platform Selector 列表中，为您要试听的对象选择平台。

2. 在 Transport Control 中，点击 Inc. Only.

此时 Inc. 按钮变成蓝色，并且 Transport Control 中仅播放当前平台中的对象和事件。

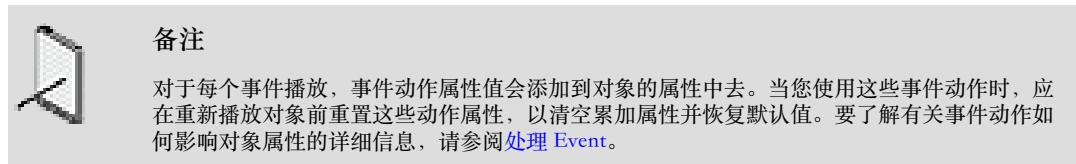


相关主题

- [播放/不播放音频和振动内容](#)
- [在播放期间使用原始音频文件](#)

重置 Transport Control

在 Transport Control 中播放对象时，您可以访问对象的一系列属性、行为和游戏同步器，这可以帮助您模拟游戏内的体验。当您连接到游戏时，某些游戏同步器、效果器和事件也可能影响先前定义的对象属性。Transport Control 中的属性标志为您提供有关在播放期间仍然有效的行为或动作的反馈。要将对象恢复为先前的设置，可使用重置功能。



重置特定 Transport Control 属性的方法是：

1. 在 Transport Control 中，点击选择器图标。

此时将显示 Reset 菜单。

2. 从 Reset 菜单中选择以下其中一项：

- Reset All 可将所有对象恢复到原始设置。
- Reset All Random and Sequence Containers 可清空已为对象触发的所有随机和序列动作。
- Reset All Game Parameters 可清空已为对象触发的所有游戏参数。
- Reset All Set Mute 可清空已为对象触发的所有静音动作。
- Reset All Set Voice Pitch 可清空已为对象触发的所有声部音高设置动作。
- Reset All Set Voice Volume 可清空已为对象触发的声部音量设置动作。
- Reset All Set Bus Volume 可清空已为对象触发的总线音量设置动作。
- Reset All Set Voice Low-pass Filter 可清空已为对象触发的所有低通滤波器动作。
- Reset All Bypass Effect 可清空已为对象触发的所有旁通效果动作。
- Reset All States 可清空对象的所有 Set State 动作。
- Reset All Switches 可清空已为对象触发的所有 Set Switch 动作。
- Reset All Music Tracks Force Usage 不再强制播放 Soundcaster 中的特定声轨。

- **Reset Position** 用于将 Attenuation Preview (衰减预览) 控件内听者的定位重置为其默认定位。



技巧

要在不显示 Reset 菜单的情况下恢复所有对象的默认设置，点击 **Reset All**。

相关主题

- [播放/不播放音频和振动内容](#)
- [在播放期间使用原始音频文件](#)

在 Transport Control 中平移对象

Transport Control 自动加载 Property Editor 中当前显示的任何对象。但是，当您在 Project Explorer 中选择任何对象或事件时，在默认情况下，它将替代 Transport Control 中的内容。如果您希望无论您选择任何其它内容，Transport Control 中都会一直将对象保持为加载状态，则您可以固定 (pin) 该对象来阻止加载其它对象。



备注

在实时混音时，您需要将对象加载到 Transport Control 中，以使针对该对象的相对属性的更改生效。确保在实时混音之前未固定对象。

使用固定功能——示例

假设您正在试听 Random Container，很想知道混响效果器对于容器中特定对象的声音效果。您可以将对象固定 (pin) 在 Transport Control 中。然后在 Property Editor 中对父对象应用混响，单独试听该对象的效果。如果您不固定对象，则容器中的所有对象将以随机模式播放，并且您想试听的对象可能不会播放。通过固定对象，您可以修改父级设置，然后随时再次试听对象。

固定 Transport Control 中加载的对象的方法是：

1. 将对象加载到 Transport Control 中。
2. 在 Transport Control 中，点击 Pin。

Pin 图标将变成红色，即使您将其他对象或事件加载到 Property Editor 中或者在 Project Explorer 中选择另一个对象，此对象或事件仍将保留在 Transport Control 中。

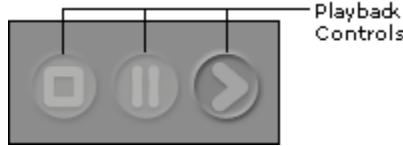


技巧

按 **Ctrl+Alt+P** 以加载对象并将它固定在 Transport Control。要取消固定对象，则按 **Ctrl+P**。

播放/暂停/停止内容

如果您想试听对象，则可使用 Transport Control 中的标准播放控件。



Transport Control 视图中的主要控制按钮都有对应快捷键，以方便您在实时编辑属性时启动、暂停和停止播放。要获取键盘快捷方式的完整列表，请参阅 [附录 C, 快捷键](#)。

在 Transport Control 中播放声音的方法是：

1. 将对象加载到 Transport Control 中。
2. 执行以下任何一项操作：
 - 点击 Play 图标。
 - 相关内容将一直播放到结束为止。
 - 要暂停播放，点击 Pause 图标。
 - 此时 Pause 图标变成黄色。要继续播放，则再次点击 Pause。
 - 若要停止回放，点击 Stop 图标。

相关主题

- [在播放期间使用原始音频文件](#)
- [播放/不播放音频和振动内容](#)
- [在 Transport Control 中平移对象](#)
- [在播放期间使用 Game Sync](#)

在播放期间使用 Game Sync

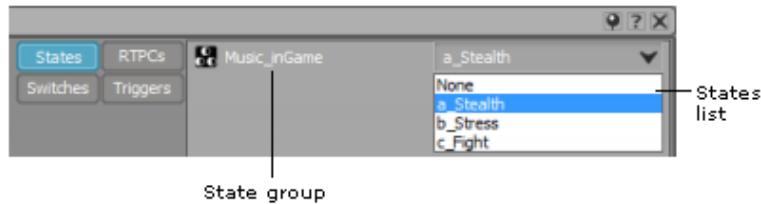
在 Transport Control 中，您可以访问为工程创建的 Game Sync。要在 State、Switch 和 RTPC 应用到声音、音乐和振动对象时试听它们，可执行以下任何一项操作：

- [在播放期间启用 State](#)
- [在播放期间指定 Switch](#)
- [在播放期间更改游戏参数值](#)
- [播放期间调用触发器](#)

在播放期间启用 State

当对象加载到 Transport Control 时，您可以从 State Group 和 State 的列表中选择对象所采用的 State Group 和 State，以模拟播放期间游戏中将发生的 State 和 State 更改。这意味着，在播放对象时，您可以试听 State 属性，还可以切换 State 来试听 State 更改。要了解有关创建 State、您将在 Transport Control 中试听的 State 属性和过渡，以及将对象指派到 State 的详细信息，请参阅以下各节：

- [使用 State](#)
- [将 State 指派给对象和总线](#)



在播放期间启用状态的方法是：

1. 将对象加载到 Transport Control 中。
2. 在 States 列表中，选择您要应用的 State。

在播放期间，State 将作用于 Transport Control 内采用该 State 的所有声音。

3. 点击 Play 图标。

播放期间将应用您选择的 State。在对象播放时，您可以继续切换 State 来模拟游戏。



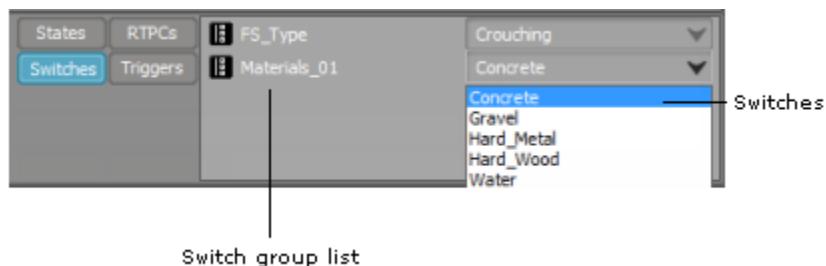
相关主题

- [在播放期间指定 Switch](#)
- [在播放期间更改游戏参数值](#)

在播放期间指定 Switch

当对象加载到 Transport Control 时，您可以从 Switch Group 和 Switch 的列表中选择对象已指定的 Switch 和 Switch Group，以模拟播放期间游戏中将发生的 Switch 切换。这意味着在播放对象时，您可以切换 Switch，并试听切换。要了解有关创建 Switch 以及如何使用 Switch 的详细信息，请参阅以下章节：

- [使用 Switch](#)
- [定义切换容器的类型](#)



在播放期间指定 Switch 的方法是：

1. 将对象加载到 Transport Control 中。

2. 从 Switch 列表中选择要应用的 Switch。

状态于是应用到 Soundcaster 中采用该状态的所有对象上去了。

3. 点击 Play 图标。

播放期间将应用您选择的 Switch。当播放对象时，您可以继续切换 Switch 来模拟游戏。



相关主题

- [在播放期间启用 State](#)
- [在播放期间更改游戏参数值](#)

在播放期间更改游戏参数值

当对象加载到 Transport Control 时，相关的 RTPC 将显示在 Games Syncs 区。该区域提供一个滑块，您可以在播放对象时更改游戏参数。由于您已经将这些值映射至 Wwise 属性值，因此更改游戏参数值时，会自动改变对象属性值。这样就会模拟当游戏参数改变时发生了什么，从而确认属性映射在游戏中的效果。在将模块添加到 Soundcaster 并确定要试听的 Wwise 对象后，便可以测试映射到游戏参数（Game Parameter）的属性值。

- [管理 RTPC 中使用的 Game Parameter](#)
- [将 Wwise 属性指派给游戏参数](#)



在模拟中，您可以试听这些属性在播放期间的变化。

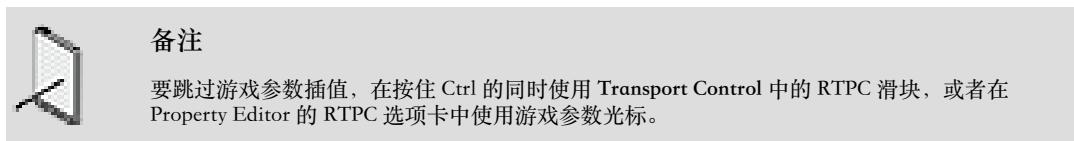
在播放期间修改游戏参数值的方法是：

1. 将对象加载到 Transport Control 中。
2. 在 Game Syncs 区，点击 RTPCs。

此时将显示已映射到对象的游戏参数。

3. 点击 Play 图标。

当对象播放时，您可以使用 RTPC 滑块更改游戏参数值，以了解声音如何对更改做出反应。



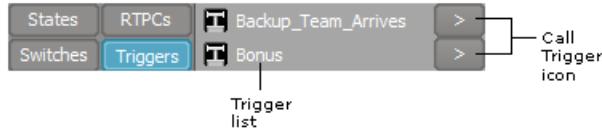
相关主题

- [在播放期间指定 Switch](#)
- [在播放期间启用 State](#)

播放期间调用触发器

除了试听此对象外，您还可以从触发器列表中选择 Trigger 来试听 Stinger。插播乐句是简短的乐句，与当前播放的音乐进行叠加，混合播放。通过这种方式，您可以模拟游戏在关键时刻通过 Trigger 调用 Stinger 后，插播乐句与当前音乐叠加播放的情况。要了解有关创建触发器以及为它们创建您要在 Transport Control（播放控制）中试听的插播乐句的详细信息，请参阅以下各节：

- [使用 Trigger](#)
- [第 29 章 使用 Stinger](#)



在播放期间调用触发器的方法是：

1. 将音乐对象加载至 Transport Control 中。
2. 点击 Play 图标。

随即播放加载至 Transport Control 中的音乐对象。

3. 在 Game Syncs 区域中，点击 Triggers 按钮以显示触发器列表。



4. 点击 Call Trigger 图标。

相应的插播乐句将在当前所播放的音乐对象之上播放。您可以选择其他触发器并播放相应插播乐句来模拟游戏中的音乐。

相关主题

- [播放/暂停/停止内容](#)
- [使用 Trigger](#)

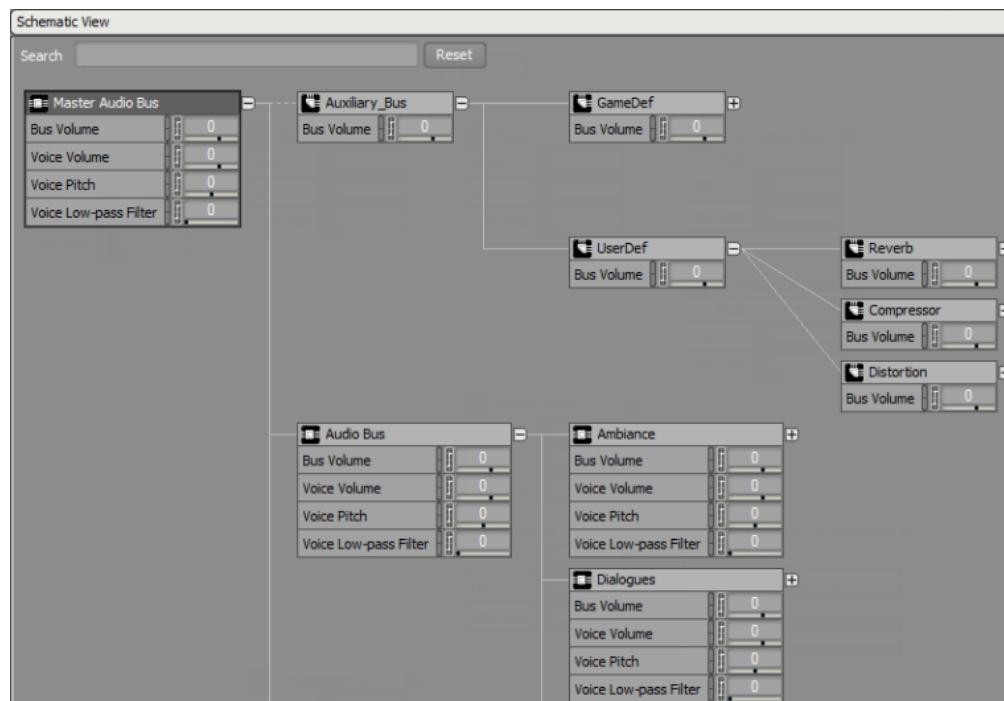
- 第 29 章 使用 Stinger

第 42 章 认识 Schematic View 视图

概述	753
自定义 Schematic View 视图	753
使用 Schematic View 视图	755

概述

Schematic (对象网络) 视图是 Wwise 工程结构的图形表示。您可以使用 Schematic View 获取工程的总览，查找工程对象，或通过逐一分析对象来分析工程结构。Schematic View 包括代表各个工程对象的图标、对象名称以及代表它们之间关系的线和节点。您还可以自定义 Schematic View 来显示工程对象详情。



自定义 Schematic View 视图

Schematic View 包含工程对象的视觉表示，并提供工具来帮助自定义此视图。它还提供便于快速定位对象的搜索功能。

要自定义 Schematic View，您可以：

- **标识工程对象和连线**
- **设置 Schematic View 显示选项**

标识工程对象和连线

Schematic View 由代表工程对象的图标和代表它们之间相互关系的连线构成。下表描述了工程对象之间的连线。有关代表工程对象的图标的信息，请参阅[理解 Wwise 中的视觉元素](#)。

图标	名称	描述
	实线	这些线用于连接父对象和子对象。
	虚线	此类线用于将总线连接到子对象，以演示通路。
	加号（白色）	点击将展开示意图，并显示对象的所有子项。

图标	名称	描述
	加号 (黄色)	如果当前没有显示对象的所有子项，则点击它将显示所有子项。
	减号	点击它将折叠对象的所有子项。

设置 Schematic View 显示选项

您可以通过使用 Schematic View Setting 来自定义示意图中各个工程对象要显示的信息。

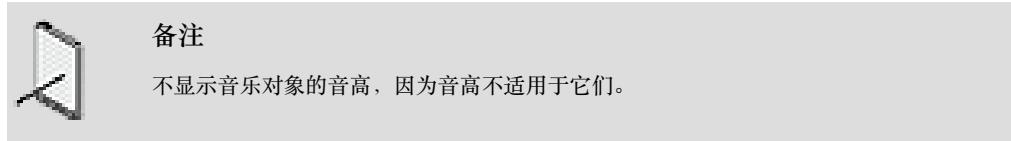
指定在 Schematic View 中显示的信息的方法是：

1. 点击 Schematic View 右上角中的选项图标。

此时将打开 Schematic View Setting 对话框。

2. 从以下信息类型中选择：

- **Icon Strip** —— 显示一排代表对象属性的图标。如果对象属性已经改变，则图标显示为白色，您则可以将鼠标悬浮在它上方以了解详细信息。如果属性仍为默认值，图标则会变成灰色。
- **Mute/Solo** —— 显示各个对象的静音和 Solo 按钮。
- **Bus** —— 显示为对象指定通路的总线。
- **Conversion Setting** —— 显示对象正在使用的转码设置 ShareSet。
- **Effect** —— 显示已应用到对象的任何效果器。
- **Positioning Type (定位类型)**：显示应用于对象的定位类型（No Positioning、3D Emitter、3D Emitter with Automation、3D Listener with Automation）。
- **Game Parameters (游戏参数)**：显示通过 RTPC 影响对象的 Game Parameter。
- **State Group** —— 显示对象指定的状态组。
- **Advanced Settings** —— 显示对对象高级设置所做的任何更改（例如播放数限制或音量阈值）。
- **Volume** —— 显示指定到对象的音量。
- **Pitch** —— 显示指定到对象的音高。



- **Low Pass** —— 显示指定到对象的低通滤波器。

3. 点击 OK。

Schematic View 显示您为各个工程对象选择的信息。

相关主题

- [标识工程对象和连线](#)
- [搜索工程对象](#)

- [显示工程对象](#)
- [编辑工程对象](#)

使用 Schematic View 视图

Schematic View 为您提供大量的选项用于查找、检查和使用其中显示的工程对象。

在使用 Schematic View 时，您可以：

- [搜索工程对象](#)
- [显示工程对象](#)
- [编辑工程对象](#)

以下各节中的许多命令已映射到键盘快捷键。快捷方式的完整列表请参阅[附录 C, 快捷键](#)。

搜索工程对象

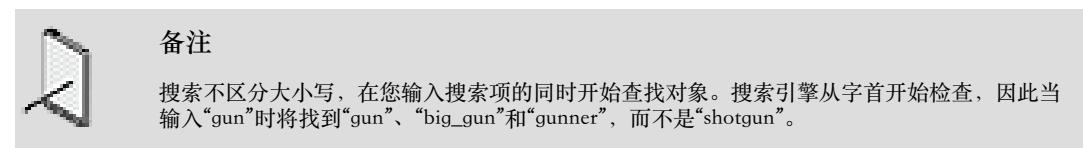
在 Schematic View 中，您可以使用搜索功能来快速查找工程对象。

在 Schematic View 中查找工程对象的方法是：

1. 在 Search 字段中，输入您要查找到的工程对象的名称。

Wwise 于是将高亮显示与您输入的名称相匹配的工程对象。

2. 点击 Reset 以清空 Search 字段，并重置 Schematic View。



相关主题

- [设置 Schematic View 显示选项](#)
- [标识工程对象和连线](#)
- [显示工程对象](#)
- [编辑工程对象](#)

显示工程对象

您可以通过指定要高亮显示的一个工程对象来简化 Schematic View 中显示的工程示意图。

在 Schematic View 中高亮显示工程对象的方法是：

1. 在 Schematic View 中，右键点击工程对象。

此时将会显示快捷菜单。

2. 从快捷菜单中选择 Show in Schematic View。

此时将显示选定对象的总线通路。对象高亮显示，并且无关的对象被隐藏。

3. 要重置 Schematic View，点击 Clear。



技巧

您还可以将对象从 Project Explorer 中拖到 Schematic View 中进行显示。还可以在 Property Editor 或 Contents Editor 中点击右键，然后选择 Show in Schematic View。

相关主题

- [标识工程对象和连线](#)
- [设置 Schematic View 显示选项](#)
- [搜索工程对象](#)
- [编辑工程对象](#)

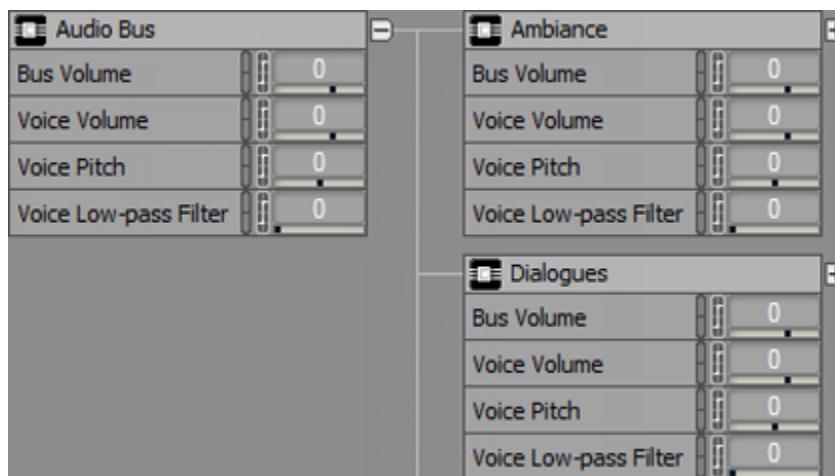
编辑工程对象

您可以使用以下两种方式编辑 Schematic View 中显示的工程对象：

- [直接编辑工程对象](#)
- [在 Property Editor 中编辑工程对象](#)

直接编辑工程对象

在 Schematic View 中，您可以使用各个工程对象下显示的控件来编辑对象。这些控件的工作原理与 Property Editor 中的控件相同。有关如何使用这些控件的详细信息，请参阅[Wwise 界面基础知识](#)。



使用 Schematic View Setting 显示这些控件。有关这些设置的详细信息，请参阅[设置 Schematic View 显示选项](#)。

在 Property Editor 中编辑工程对象

如果您喜欢直接编辑对象属性，则可以从 Schematic View 快速打开工程对象的 Property Editor。

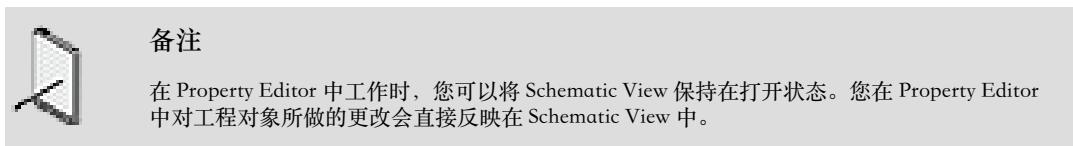
为工程对象打开 Property Editor 的方法是：

1. 在 Schematic View 中，右键点击工程对象。

此时将会显示快捷菜单。

2. 从快捷菜单中选择 Edit。

此时将为该工程对象打开 Property Editor。



相关主题

- [标识工程对象和连线](#)
- [设置 Schematic View 显示选项](#)
- [搜索工程对象](#)
- [显示工程对象](#)

第 43 章 了解坐标图视图

概述	759
更改坐标图的显示	760
在坐标图视图中使用控制点	764
使用坐标图视图中的曲线	766

概述

Wwise 的多种视图中都包含坐标图。您可使用这些坐标图映射两个变量之间的关系、规划空间定位，也可通过坐标图监控性能。

Wwise 中包含多种不同的坐标图。虽然这些坐标图通常支持很多相同的功能，但各有不同。不同的坐标图位于不同的视图中：

- Project Settings (Obstruction/Occlusion 选项卡)
- Property Editor (RTPC 选项卡)
- Blend Tracks Editor (混合轨编辑器)
- Attenuation Editor (衰减编辑器)
- Position Editor (位置编辑器)
- Switch Group Property Editor (切换开关属性编辑器)
- Music Fade Editor (音乐淡变编辑器)
- Performance Monitor (性能监视器)
- Game Sync Monitor 视图
- Wwise Motion Generator (不建议使用) Property Editor
- Wwise Reflect - Effect Editor (Wwise Reflect 效果编辑器)
- SoundSeed Air Source Editor (SoundSeed Air 源编辑器)

下表说明了各个坐标图特有的功能。

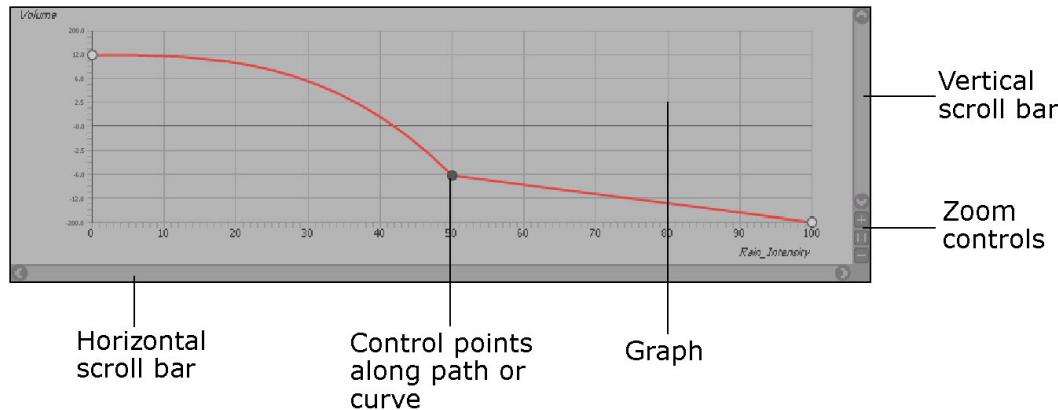
使用该坐标图	用途
Project Settings (Obstruction/Occlusion 选项卡)	定义声音对象的声障和声笼因素将如何影响音量与 LPF。
Property Editor (RTPC 选项卡)	定义游戏参数值与 Wwise 中属性值之间的关系。
Blend Tracks Editor (混合轨编辑器)	混合轨编辑器。将对象编组至混合轨中，并让它们随若干个 RTPC (实时参数控制) 曲线变化，或在对象之间进行交叉淡变。
Attenuation Editor (衰减编辑器)	衰减编辑器。定义了听者与声源之间的距离将如何影响声音的某些属性，用来创建真实的声音衰减效果。
Position Editor	位置编辑器。创建 3D 环境内的路径，用来在环绕声扬声器环境内进行声音定位。
Switch Group Property Editor (切换开关属性编辑器)	切换开关属性编辑器。将切换开关组内的不同切换开关映射至特定游戏参数。
Music Fade Editor	音乐淡变编辑器。定义音乐过渡中两段音乐之间的淡入/淡出曲线。
Performance Monitor	性能监视器。当游戏、游戏模拟器或 Soundcaster (声音选角器) 触发游戏元素时，查看与声音引擎的性能相关的信息。 Performance Monitor 中的信息为只读信息，仅用于诊断用途。因此，您无法添加和移除控制点，或编辑任何曲线。
Game Sync Monitor 视图	游戏同步器监视器。当 RTPC 值应用至您正在观察的游戏对象时，跟踪这些值。 Game Sync Monitor 中的信息为只读信息，仅用于诊断用途。因此，您无法添加和移除控制点，或编辑任何曲线。
Wwise Motion Generator Property Editor)	Wwise 振动生成器属性编辑。创建一组曲线，来为各平台控制器所用的不同马达定义各自的振动强度。
Wwise Reflect - Effect Editor (Wwise Reflect 效果编辑器)	允许编辑五种基于距离的不同曲线，分别用来定义音量 Attenuation (衰减)、Spread (散布) 百分比和声音高低切截止频率。

使用该坐标图	用途
SoundSeed Air —— Woosh Source Plugin Editor (对象路径坐标图)	SoundSeed Air - Woosh 源插件编辑器。创建定义 Woosh 场景内物体的轨迹的路径，并且为声音远离时定义该声音的衰减。
SoundSeed Air (Wind/Woosh) Source Plugin Editor	SoundSeed Air (Wind/Woosh) 源插件编辑器。创建一组自动化曲线，用来定义生成的 wind 或 woosh 声音的属性。
Voice Monitor	声部监视器。在时间线视图上查看所播放的 Wwise 对象声部的音量曲线。

当创建声音路径或监控性能，以及使用音乐段落时，还需要定义时间范围。此时，坐标图视图与时间线相关联。有关在 Wwise 中使用时间线的详细信息，请参阅 [Getting to Know the Timeline](#)。

坐标图的通用元素说明

虽然 Wwise 中的各个坐标图视图稍有不同，但它们包含某些通用元素，如缩放控制、滚动栏，以及曲线或路径的控制点。下图显示了坐标图中的各个通用元素。



更改坐标图的显示

您可以更改坐标图显示方式，从而更准确地放置控制点，或更仔细地检查特定区域。可使用以下工具更改坐标图显示方式：

- [缩放和平移坐标图视图](#)
- [定义坐标图视图的坐标显示方式](#)
- [显示/隐藏网格线](#)
- [显示/隐藏光标](#)

缩放和平移坐标图视图

您以可使用平移和缩放工具沿路径放置特定控制点，也可更仔细地检查特定区域内的信息。可以放大和缩小、进行上下左右平移。

所有平移和缩放控件均有相关的键盘快捷键。

操作	使用此快捷方式
放大	Z + 选取框
垂直放大——以当前鼠标位置为中心	Ctrl + 向上滚动鼠标滚轮

操作	使用此快捷方式
垂直缩小——以当前鼠标位置为中心	Ctrl + 向下滚动鼠标滚轮
水平放大——以当前鼠标位置为中心	Ctrl + Shift + 向上滚动鼠标滚轮
水平缩小——以当前鼠标位置为中心	Ctrl + Shift + 向下滚动鼠标滚轮
重置缩放（放大时）	Z + 点击
向左平移视图（放大时）	Shift + 向上滚动鼠标滚轮
向右平移视图（放大时）	Shift + 向下滚动鼠标滚轮
向上平移视图（放大时）	向上滚动鼠标滚轮
向下平移视图（放大时）	向下滚动鼠标滚轮
任意平移（放大时）	X + 拖动

快捷方式的完整列表请参阅[附录 C, 快捷键](#)。

放大坐标图视图的方法如下：

1. 在坐标图视图工具栏中，点击 Zoom In 图标。

坐标图朝向坐标图视图中心放大。

缩小坐标图视图的方法如下：

1. 在坐标图视图工具栏中，点击 Zoom Out 图标。

坐标图从坐标图视图中心缩小。

重设坐标图视图中的平移和缩放的方法如下：

1. 在坐标图视图工具栏中，点击 Reset 图标。

坐标图视图重设，以显示所有现有点。

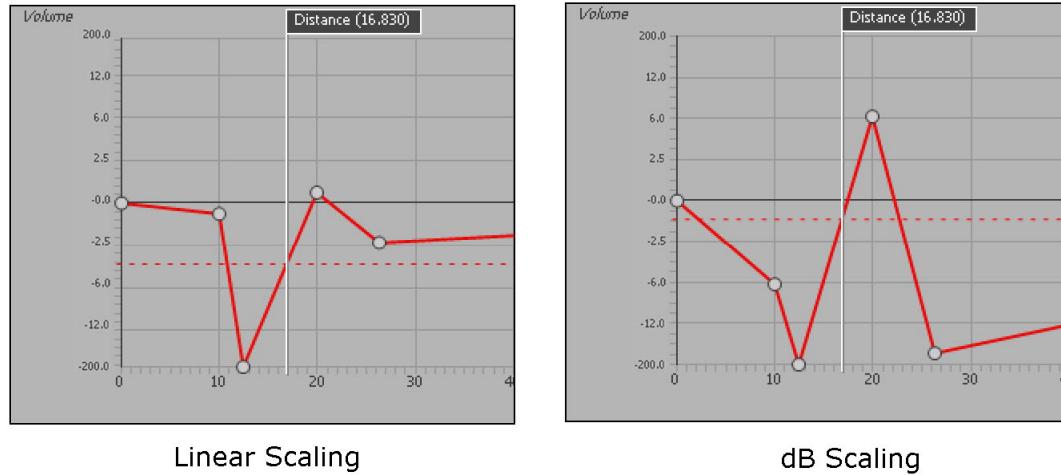
平移坐标图视图的方法如下：

1. 放大坐标图视图。
2. 执行以下操作之一：
 - 要向上平移，请向上移动鼠标滚轮。
 - 要向下平移，请向下移动鼠标滚轮。
 - 要向右平移，请按下 Shift 并向下移动鼠标滚轮。
 - 要向左平移，请按下 Shift 并向上移动鼠标滚轮。
 - 要任意平移，请按下 X 并在坐标图视图内拖动鼠标。

定义坐标图视图的坐标显示方式

当查看以分贝为单位的曲线时，您可以选择以严格的线性方式显示曲线，还是以更常用的（对数）方式显示声音分贝数。在线性坐标下，Y 轴上间隔距离相等。在分贝坐标下，Y 轴坐标将模拟人耳将声音解释成不同分贝数的方式。在接近于零时，间隔距离较大，而在较低分贝值下，间隔距离较小。

下图显示了以分贝坐标和线性坐标显示同一音量曲线之间的不同。



由于分贝坐标会影响分贝单位沿 Y 轴的分布，所以坐标显示方式还将影响控制点如何在坐标图视图内移动。例如，与 -80 dB 相比，当点位于 0 dB 时，分贝坐标中 -5 dB 的变化将要求移动距离更大。但在线性定标中，-5 dB 的变化将总是导致沿 Y 轴的移动距离相同。

在大多数情况下，分贝坐标将更准确地表示出特定游戏参数值下所听到的声音。但当创建游戏参数与 Wwise 属性之间的直接关系时，就需要使用线性坐标。例如，假设您要将 Voices 总线的音量映射至游戏中的音量滑杆，允许游戏玩家提高或降低人声的音量。此时您需要将声音音量直接映射至游戏的音量参数。由于沿 X 轴的音量是线性的，因此同样需要让 Y 轴以线性显示。



备注

坐标显示方式会影响 Y 轴上的单位，因此如果同时显示两个或多个使用不同显示显示的曲线，则 Y 轴上的单位将不会显示。

定义坐标图视图的坐标显示方式的方法如下：

1. 在坐标图视图中，右键点击曲线。

此时将会显示快捷菜单。

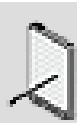
2. 选择以下任一一种坐标显示方式：

- **dB scaling** 根据人耳解释声音的通用（对数）方式，以分贝数显示曲线。
- **Linear scaling** 以严格的线性方式显示曲线。

坐标图视图将根据您选择的方法来显示曲线。

显示/隐藏网格线

部分坐标图视图使用网格线，可以帮助您更准确地放置控制点。可以选择显示或隐藏这些网格线，这并不会影响坐标图中的点。



备注

默认情况下将显示网格线。

隐藏坐标图视图中的网格线的方法如下：

1. 右键点击坐标图视图。

此时将会显示快捷菜单。

2. 选择以下任一选项：

- Display Grid (Vertical), 移除垂直网格线。

- Display Grid (Horizontal) , 移除水平网格线。

3. 重复步骤 1-2 可移除剩余的网格线。

显示坐标图视图中的网格线的方法如下：

1. 右键点击坐标图视图。

此时将会显示快捷菜单。

2. 选择以下任一选项：

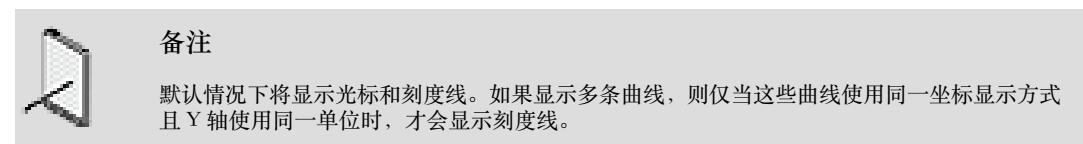
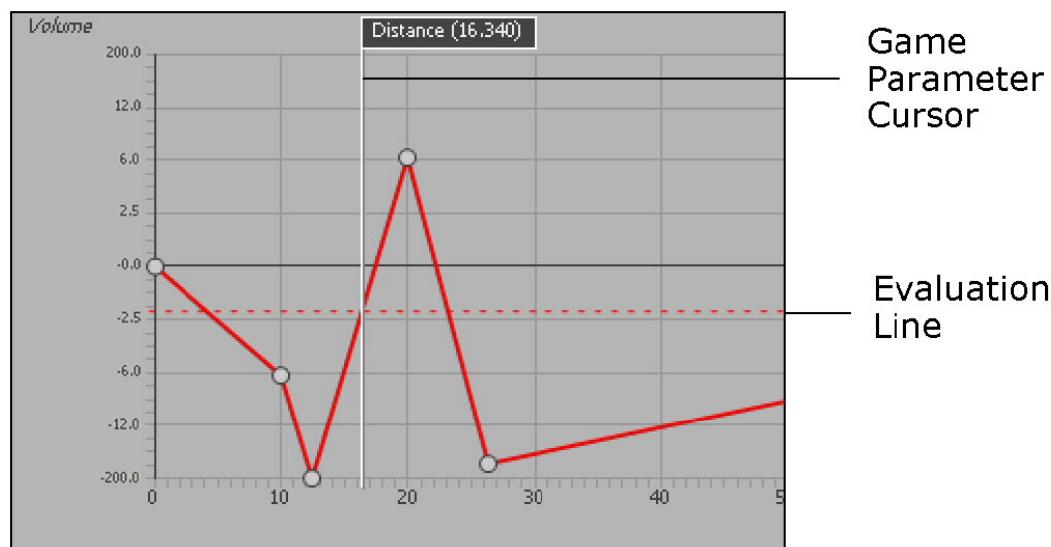
- Display Grid (Vertical), 添加垂直网格线。

- Display Grid (Horizontal), 添加水平网格线。

3. 重复步骤 1-2 可添加剩余的网格线。

显示/隐藏光标

RTPC 视图包含各条曲线的游戏参数光标和刻度线。在游戏参数光标顶部有一个标识框，标有相应游戏参数的名称，及其在曲线上的当前值。您可以在播放过程中前后拖动光标，以模拟游戏参数值的变化。刻度线可以帮助您确定对应于特定游戏参数值的相应属性值。您可以选择显示或隐藏光标，这不会影响坐标图上的点。



在坐标图视图中隐藏光标的方法如下：

1. 右键点击坐标图视图。

快捷菜单将打开。

2. 选择 Show cursors。

光标从坐标图中移除。

在坐标图视图中显示光标的方法如下：

1. 右键点击坐标图视图。

快捷菜单将打开。

2. 选择 Show cursors。

光标将显示在坐标图视图中。

在坐标图视图中使用控制点

坐标图视图中的控制点可以用来定义路径或曲线的形状，也可以创建两个变量间的特定关系曲线。您可随时添加、移动和删除曲线上的点。也可以同时选择多个点，来操控或删除若干条曲线上的特定部分。

下列章节中提供了控制点的使用须知：

- [添加控制点](#)
- [选择控制点](#)
- [移动控制点](#)
- [删除控制点](#)

添加控制点

您可沿曲线在任意位置添加控制点，以定义其形状或创建不同变量之间的关系。由于您需要在各个坐标图视图中定义不同类型的信息，因此添加控制点的方式会稍有不同。下表显示了如何在不同的坐标图视图中添加点。

坐标图	添加点	在两点间插入点
Project Settings (Obstruction/Occlusion 选项卡)	不适用	在曲线上的两个点之间双击。
RTPC/Blend Tracks Editor (RTPC/混合轨编辑器)	不适用	在曲线上的两个点之间双击。
Position Editor (位置编辑器)	双击	Ctrl + 在曲线上的两个点之间双击。
Attenuation Editor (衰减编辑器)	不适用	在曲线上的两个点之间双击。
Switch Group Property Editor (切换开关属性编辑器)	不适用	在曲线上的两个点之间双击。
Music Fade Editor (音乐淡变编辑器)	不适用	不适用
Performance Monitor (性能监视器)	不适用	不适用
Game Sync Monitor 视图	不适用	不适用

坐标图	添加点	在两点间插入点
Wwise Motion Generator Source Plugin Editor (源插件编辑器)	不适用	在曲线上的两个点之间双击。
SoundSeed Air —— Woosh Source Plugin Editor (对象路径坐标图)	双击	Ctrl + 在曲线上的两个点之间双击。
SoundSeed Air (Wind/Woosh) Source Plugin Editor (源插件编辑器)	不适用	在曲线上的两个点之间双击。

选择控制点

移动或删除坐标图视图中的控制点之前，必须先选择控制点。您每次可选择一个、多个或所有控制点。如果在坐标图视图中同时显示多条曲线，您则可选择不同曲线上的多个点，来移动或删除它们。

在坐标图视图中选择控制点的方法如下：

1. 在坐标图视图中，点击控制点，以选中该控制点。
所选控制点将变为黑色。

在坐标图视图中选择多个控制点的方法如下：

1. 在坐标图视图中，执行以下任一操作：
 - 拖拽选取框，使其包括要选择的点。
 - Ctrl + 点击要选择的点。

所选的点将变为黑色。

选择坐标图视图中的所有控制点的方法如下：

1. 点击坐标图视图以激活该视图。
2. 按下 Ctrl+A。

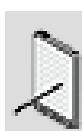
坐标图视图中的所有点将被选中，并变为黑色。

移动控制点

通过移动坐标图内任意位置的若干个控制点，可操控若干条曲线的形状。但当坐标图视图缩放至最小时，您将受到坐标图的外框限制。

移动坐标图中的控制点的方法如下：

1. 在坐标图中，选择若干个控制点。
所选控制点将变为黑色。
2. 将控制点拖至坐标图边框内的任意位置。



备注

也可使用箭头键，或者直接在 X 和 Y 坐标字段中输入值以移动控制点。如果选择了多个点，则在 X 和 Y 字段中输入的值将使各点坐标在其原始位置基础上进行增减。

删除控制点

如果要从曲线或路径中移除点，则可以将其删除。但某些点无法删除。下表说明了在各个坐标图视图中可删除的点。

坐标图	可删除的点
Project Settings (Obstruction/Occlusion 选项卡)	除第一个点和最后一个点以外的所有点。
RTPC/Blend Tracks Editor (RTPC/混合轨编辑器)	除第一个点和最后一个点以外的所有点。
Attenuation Editor (衰减编辑器)	除第一个点和最后一个点以外的所有点。
Position Editor (位置编辑器)	除第一个点以外的所有点。
Switch Group Property Editor (切换开关属性编辑器)	除第一个点和最后一个点以外的所有点。
Music Fade Editor	不适用 —— Music Fade Editor 中没有控制点。
Performance Monitor (性能监视器)	不适用 —— Performance Monitor 中没有控制点。
Game Sync Monitor 视图	不适用 —— Game Sync Monitor 中没有控制点。
Wwise Motion Generator Source Plugin Editor (Wwise 振动发生器源插件编辑器)	除第一个点和最后一个点以外的所有点。
SoundSeed Air —— Woosh Source Plugin Editor (对象路径坐标图)	除第一个点以外的所有点。
SoundSeed Air (Wind/Woosh) Source Plugin Editor (源插件编辑器)	除第一个点和最后一个点以外的所有点。

从曲线删中除控制点的方法如下：

1. 选择曲线或路径上的若干个点。

所选控制点将变为黑色。

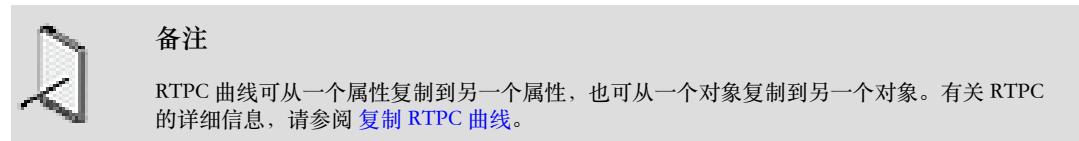
2. 按 Delete 键。

控制点将从曲线中移除。

使用坐标图视图中的曲线

由于您可以同时使用同一坐标图视图中的多条曲线，因此掌握以下基本要点十分重要：

- 在坐标图视图中显示曲线
- 将曲线固定至坐标图视图的方法如下：
- 指定控制点之间曲线的形状



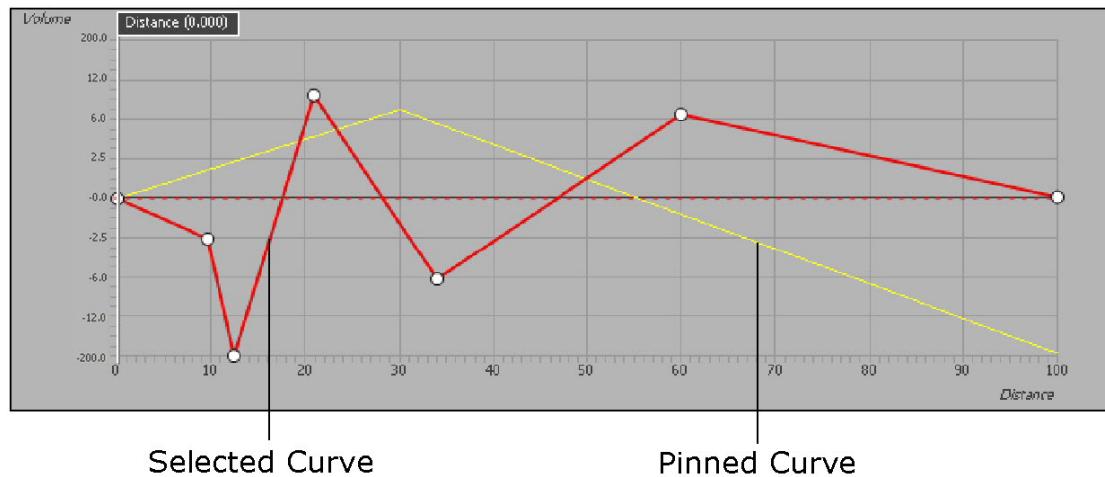
在坐标图视图中显示曲线

在 Wwise 中的大部分坐标图视图中，您都可以使用多条曲线来代表不同的路径，或代表游戏参数与 Wwise 属性之间不同的关系。从列表中选择曲线时，该曲线会显示

在坐标图视图中。如果坐标图支持多条曲线显示，如 RTPC 和 Attenuation Editor 坐标图视图，您则可以按下 Ctrl 并点击多条曲线，以在坐标图视图中同时显示所有曲线。

如果希望曲线在不被选中的情况下仍然显示在坐标图视图中，则可将其固定至坐标图视图。固定曲线后，它的轮廓会总会保留在坐标图视图中。但不会显示曲线控制点。必须选择曲线后才能编辑该曲线的控制点。

下图显示了 RTPC 坐标图视图中已选和固定曲线之间的差异。



如果坐标图视图中显示了多条曲线，则不会显示 X 和 Y 轴上的单位。当游戏参数和曲线的属性单位不同时会发生这种情况。例如，如果同时显示音高和音量曲线，则 Y 轴上的单位将消失，因为一个单位是音分，而另一个单位是分贝。



备注

如果坐标图视图中显示的曲线的坐标显示方式不同，则不会在 Y 轴上显示单位。有关坐标显示方式的详细信息，请参阅 [定义坐标图视图的坐标显示方式](#)。

在坐标图视图中显示曲线的方法如下：

1. 在坐标图视图中，从曲线列表中选择曲线。

曲线将显示在坐标图视图中。

2. 要显示更多曲线，请按住 Ctrl 并点击列表中的曲线。

这些曲线随即显示在坐标图视图中。如果几条曲线使用不同的单位，则 X 或 Y 轴上的单位将消失。

将曲线固定至坐标图视图的方法如下：

1. 在曲线列表中，点击要固定至坐标图视图的曲线的固定图标（左侧的图钉图标）。

固定图标变为蓝色，该曲线显示在坐标图视图中。

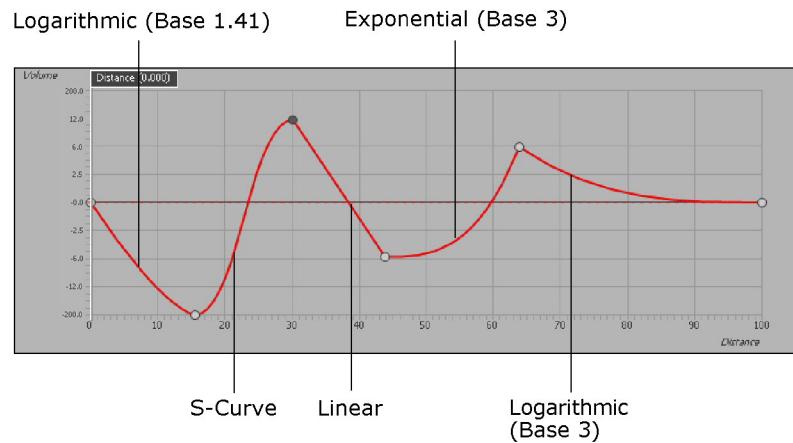
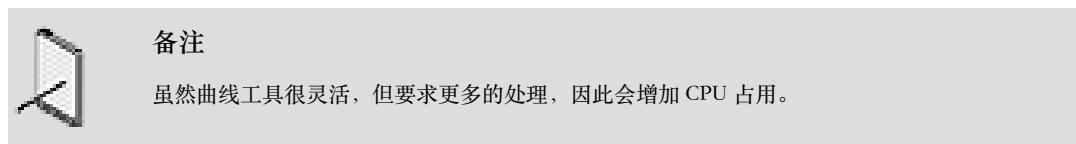
选择其它曲线时，已固定曲线的轮廓将保留显示在坐标图视图中，直至取消固定。

相关主题

- 指定控制点之间曲线的形状
- 缩放和平移坐标图视图
- 定义坐标图视图的坐标显示方式

指定控制点之间曲线的形状

为更好、更灵活地控制工程中的曲线，Wwise 可以定义各条曲线段落的形状。曲线段是两个控制点之间的部分。您可选择各种曲线形状，包括线性曲线、恒定曲线、对数曲线、幂数曲线和 S 曲线。



指定控制点间曲线的形状的方法如下：

1. 在坐标图视图中，右键点击曲线的段落。

随即显示快捷菜单

2. 从菜单中，选择以下任一选项：

- Logarithmic (Base 3)（底数为 3 的对数曲线）
- Sine (Constant Power Fade In) -- 恒定功率淡入的正弦曲线。当用于曲线的“淡入”部分时，该 Sine 曲线形状仅提供恒定功率交叉淡变。
- Logarithmic (Base 1.41)（底数为 1.41 的对数曲线）
- Inverted S-Curve (反转 S 曲线)
- Linear (线性)
- Constant (恒定)

- S-Curve (S 曲线)
- Exponential (Base 1.41) (底数为 1.41 的指数曲线)
- Sine (Constant Power Fade Out) -- 恒定功率淡出的正弦曲线。当用于曲线的“淡出”部分时，该 Sine 曲线形状仅提供恒定功率交叉淡变。
- Exponential (Base 3) (底数为 3 的指数曲线)

所选曲线形状将应用至该曲线段落。

3. 根据需要，继续应用曲线形状至曲线的其它段落。

相关主题

- [在坐标图视图中显示曲线](#)
- [缩放和平移坐标图视图](#)
- [定义坐标图视图的坐标显示方式](#)
- [添加控制点](#)
- [移动控制点](#)
- [删除控制点](#)

第 44 章 认识时间线

概述	771
使用时间线进行定位	773
使用 Music Segment Editor 时间线	774

概述

Wwise 中的多个视图内有一条时间线，以便您可以在工程中使用基于时间的信息来实现不同的目的。下列视图内有时间线：

- Music Segment Editor (音乐段落编辑器)
- Position Editor (位置编辑器)
- Performance Monitor (性能监视器)
- Game Sync Monitor 视图
- Voice Monitor (声部监视器)

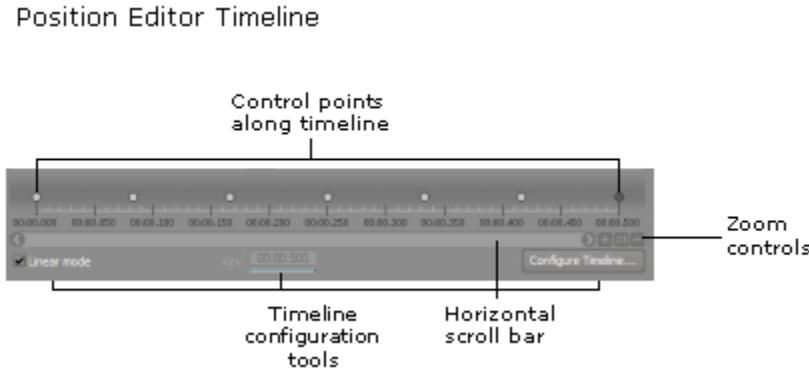
虽然这些时间线一般支持很多相同的功能，但这些时间线之间也存在差异。下表描述各条时间线的独特用途。

使用此时间线	操作
Music Segment Editor (音乐段落编辑器)	编排和同步互动音乐组件。
Position Editor (位置编辑器)	定义声音路径的时间段。
Performance Monitor (性能监视器)	从 Performance Monitor 中查看基于时间的捕获信息。
Game Sync Monitor 视图	游戏同步器监视器。查看被监视游戏对象的基于时间的 RTPC 值变化。
Voice Monitor (声部监视器)	查看 Wwise 对象声部的基于时间的音量电平。

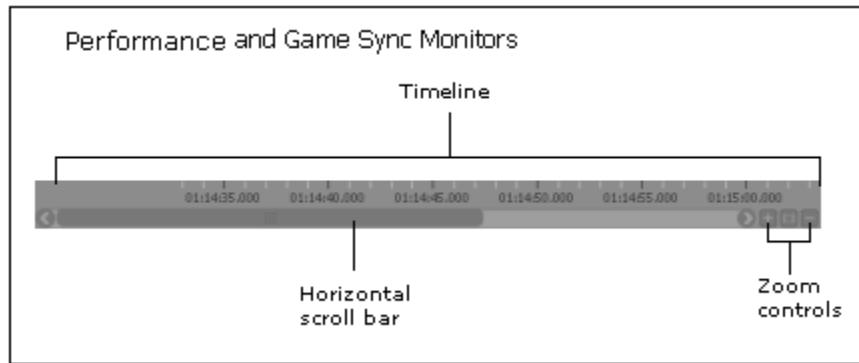
在创建声音路径和监控性能时，时间线结合坐标图视图一起使用。有关坐标图视图的详细信息，请参阅[在坐标图视图中使用控制点](#)。

描述时间线的元素

由于 Wwise 中有三种不同的时间线，下面将分别对它们进行讨论。在 Position Editor (3D Automation) 中，添加至坐标图的控制点也会显示在时间线中。时间线显示控制点所对应的时间。通过从空间和时间上绘制控制点，您不仅可以定义声音的位置，而且还可以定义声音沿路径传输所需要的时间。下图显示了 Position Editor (3D Automation) 时间线中的不同元素。

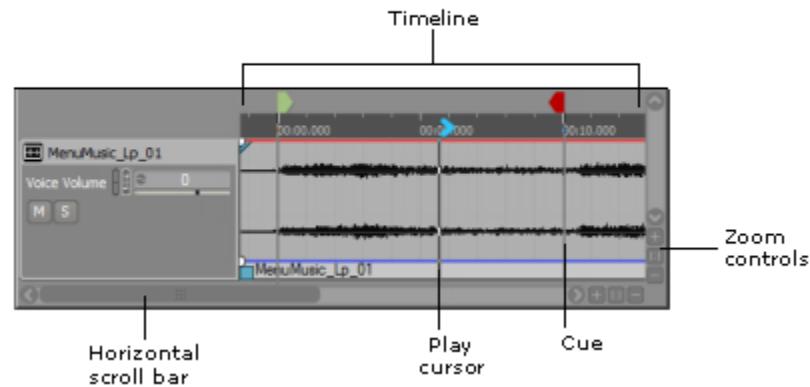


在 Performance Monitor 中，时间线帮助您定位从声音引擎中捕获并在日志中记载的各个动作和通知。下图显示 Performance Monitor 时间线中的不同元素。



在 Music Segment Editor 中，音乐片段沿时间线分轨显示。时间线和波形本身帮助您定位可发生过渡、状态切换和插播乐句的最适当音乐部分。入口提示点、出口提示点和任意数量的自定义提示点可沿时间线放置在这些特定的位置上。

Music Editor Timeline



平移和缩放时间线

您可以使用平移和缩放工具仔细检查信息，帮助您在非常具体的时间添加控制点或提示点。您可以执行放大、缩小、左移和右移操作。Music Segment Editor 中的时间线沿 X 轴和 Y 轴上布置了这些工具。

所有平移和缩放控件均有相关的键盘快捷键。

操作	使用此快捷方式
放大	Z + 选取框
垂直放大——以当前鼠标位置为中心	Ctrl + 向上滚动鼠标滚轮
垂直缩小——以当前鼠标位置为中心	Ctrl + 向下滚动鼠标滚轮
水平放大——以当前鼠标位置为中心	Ctrl + Shift + 向上滚动鼠标滚轮
水平缩小——以当前鼠标位置为中心	Ctrl + Shift + 向下滚动鼠标滚轮
重置缩放（放大时）	Z + 点击
向左平移视图（放大时）	Shift + 向上滚动鼠标滚轮
向右平移视图（放大时）	Shift + 向下滚动鼠标滚轮
向上平移视图（放大时）	向上滚动鼠标滚轮

操作	使用此快捷方式
向下平移视图（放大时）	向下滚动鼠标滚轮
任意平移	X + 拖动

快捷方式的完整列表请参阅[附录 C, 快捷键](#)。

放大时间线的方法是：

1. 在时间线工具栏中，点击 Zoom In 图标。

时间线将以时间线的中心为中心进行放大。

缩小时间线的方法是：

1. 在时间线工具栏中，点击 Zoom Out 图标。

时间线将以时间线的中心为中心进行缩小。

重置平移和放大时间线的方法是：

1. 在时间线工具栏中，点击 Reset 图标。

时间线重置为显示所有控制点。

平移时间线的方法是：

1. 放大时间线。
2. 执行以下操作之一：
 - 要向上平移，请向上移动鼠标滚轮。
 - 要向下平移，请向下移动鼠标滚轮。
 - 要向右平移，请按下 Shift 并向下移动鼠标滚轮。
 - 要向左平移，请按下 Shift 并向上移动鼠标滚轮。
 - 要做任意平移，则按 X 并用鼠标按住时间线进行拖动。

相关主题

- [配置定位时间线](#)
- [配置 Music Segment Editor 时间线](#)
- [在时间线上延长和缩短音乐轨](#)

使用时间线进行定位

在创建空间路径时，需要定义对象沿路径传输的时间。Position Editor 时间线可让您指定各个控制点所对应的时间。与坐标图视图一样，您可以缩放和平移时间线，以便更加准确地定位控制点。您还可以为您创建的各条路径配置时间线长度。

- [配置定位时间线](#)

配置定位时间线

您可以配置时间线的属性和行为。例如，您可以指定时间线的长度或者定义时间线上的控制点在添加新控制点时的行为。时间线的长度自动定义选定路径的长度。您可以配置时间线，使您创建的各条路径具有不同的时长。

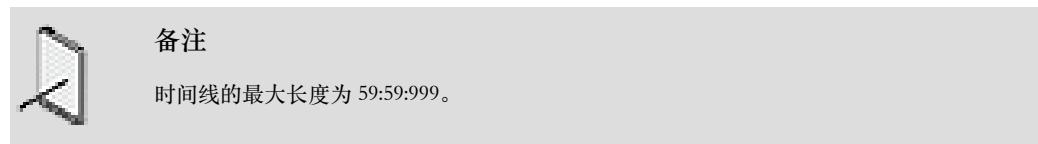
当时间线处于线性模式时，您只可定义时间线长度，因为控制点在时间线上的行为是预先确定的。

配置时间线的方法是：

1. 在 Position Editor (3D Automation) 中，单击 **Configure Timeline** (配置时间线) 按钮。

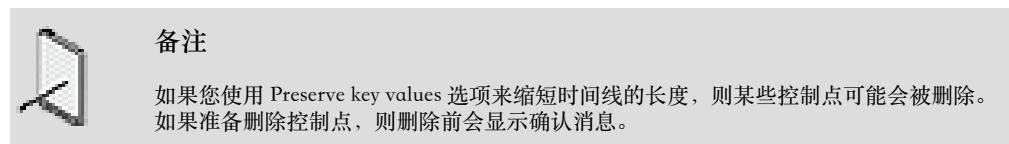
此时将打开 Timeline Configuration 对话框。

2. 在 Length 字段中，以 mm:ss.ms 为格式输入时间线的时长。



3. 如果您要更改时间线的长度，并且时间线处于非线性模式下，则选择以下其中一个选项：

- Stretch proportionally -- 按比例拉伸。重新定位现有控制点，使它们之间保持相对比例。
- Preserve key values -- 保留关键值。保持现有控制点的位置。



4. 在 Insert Key Every 字段中，输入您要在上一个现有控制点与时间线上插入的任何新控制点之间添加的时间量。

5. 点击 OK 以接受更改。

时间线于是按照新的设置进行重新配置。

相关主题

- [平移和缩放时间线](#)
- [配置 Music Segment Editor 时间线](#)
- [在时间线上延长和缩短音乐轨](#)

使用 Music Segment Editor 时间线

Music Segment Editor 时间线是互动音乐工程中用于管理编曲的关键。时间线为您使用段落 (Segment)、音轨 (Track) 和片段 (Clip) 提供重要反馈。

在使用 Music Segment Editor 时间线时，您可以执行以下任务：

- [配置 Music Segment Editor 时间线](#)
- [在时间线上延长和缩短音乐轨](#)

配置 Music Segment Editor 时间线

要优化排列音乐，您可以按照以下方式配置时间线：

- Seconds —— 以秒为单位显示时间线。当您放大此视图时，时间线将以十分之一、百分之一和千分之一秒递增显示时间。
- Bars and Beats —— 以小节和节拍为单位显示时间线。如果您使用的音乐素材必须与音乐节拍一致，则应使用此设置。时间线上的增量由相应 Property Editor 中指定的拍号（Time Signature）决定。

在 Music Segment Editor 中将时间线的单位配置为秒的方法是：

1. 右键点击时间线，然后选择 Seconds。

时间线的显示单位被重新配置为秒。



在 Music Segment Editor 中将时间线的单位配置为小节和节拍的方法是：

1. 右键点击时间线，然后选择 Bars and Beats。

时间线的显示单位被重新配置为小节和节拍。



相关主题

- [平移和缩放时间线](#)
- [配置定位时间线](#)
- [在时间线上延长和缩短音乐轨](#)

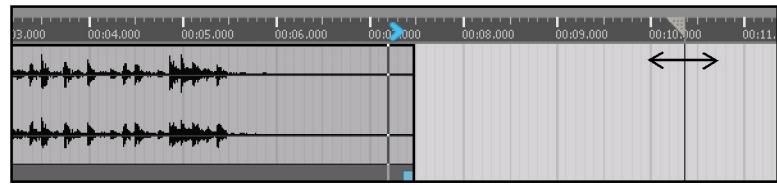
在时间线上延长和缩短音乐轨

在时间线上使用片段时，您可以调整时间线的长度。

在 Music Segment Editor 中调整时间线长短的方法是：

1. 

沿时间线将 End Cursor 图标拖到期望的位置。



Drag end cursor icon to extend or shorten the segment tracks

相关主题

- 平移和缩放时间线
- 配置定位时间线
- 配置 Music Segment Editor 时间线

第 45 章 使用 Search、Query 和 Reference

概述	778
搜索工程中的元素	778
查找引用特定对象的工程元素	780
使用 Query	784
查询——技巧和经验总结	791

概述

Wwise 工程可能包含数千种声音和振动效果、容器、事件和其他对象。随着工程扩大，能够迅速找到特定工程元素将变得越来越重要。为此，Wwise 提供三种强大高效的方法来查找工程中的元素：

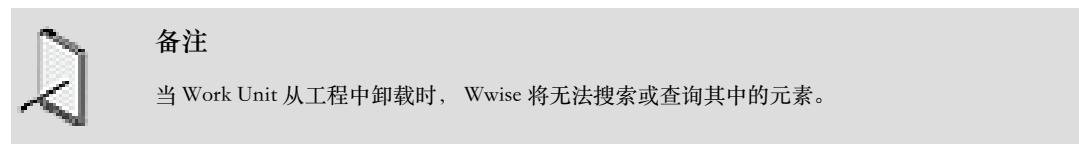
- **Search** —— 搜索。一款基于文本的快速搜索工具，在您输入搜索内容的同时就会显示匹配项。
- **Query** —— 查询。一款更加强大的搜索工具，它使用一系列的条件或标准进行搜索。
- **Reference** —— 引用。一款搜索工具，可以查找工程中包含直接引用特定对象或元素的元素。

Search 工具使用起来快速、简便。您只需在 Search 字段中输入一些文本，Wwise 就会搜索各个工程元素的名称和备注字段，以期找到匹配项。在您输入时，所有匹配项会自动显示在 Results 列表中。

Reference 工具使用起来也很快速、简便。您只需右键点击对象，选择 Find All Reference（查找所有引用），Wwise 就会创建一张列表，表中包含直接引用当前对象的对象和其它工程元素。引用列表将显示在 Reference 视图中。

而 Query 工具可为您提供更加精密和强大的搜索引擎。在 Query 中，您可以定义一系列标准（Criteria），以便搜索工程中的具体元素。例如，您可以创建一个 Query 来查找所有 Prefetch 长度为 100 ms 并启用了流播放的声音对象，然后使用 Show in Multi Editor 命令将其全部更改为 150 ms。Query 虽然设置时间略长，但一旦创建即可发挥强大的作用，您可以随时在开发周期中使用它们。

通过使用这两种搜索工具，您能够查找工程中的几乎任何单个或一系列元素。



搜索工程中的元素

您可以使用搜索工具快速轻松地查找工程中的任何元素。搜索（Search）工具是一种基于简单文本的搜索技术。您只需在搜索字段中输入文本，Wwise 就会检索整个工具来查找名称（Name）或备注（Notes）匹配的所有元素。如果找到了匹配项，则会自动将结果显示在 Result 列表中。

搜索工具通过查看名称或备注字段中各个新词的开头来寻找匹配项。在 Wwise 中，如果单词中包含非字母数字字符（例如空格、下划线、破折号等）或存在大小字变化，它们则将被视为不同的单词。了解这一点非常重要，搜索工具在查找匹配项时将考虑这一点。下表展示了几个不同的示例：

搜索字段文本	匹配	不匹配
Big	Big_Car	Verybigcar

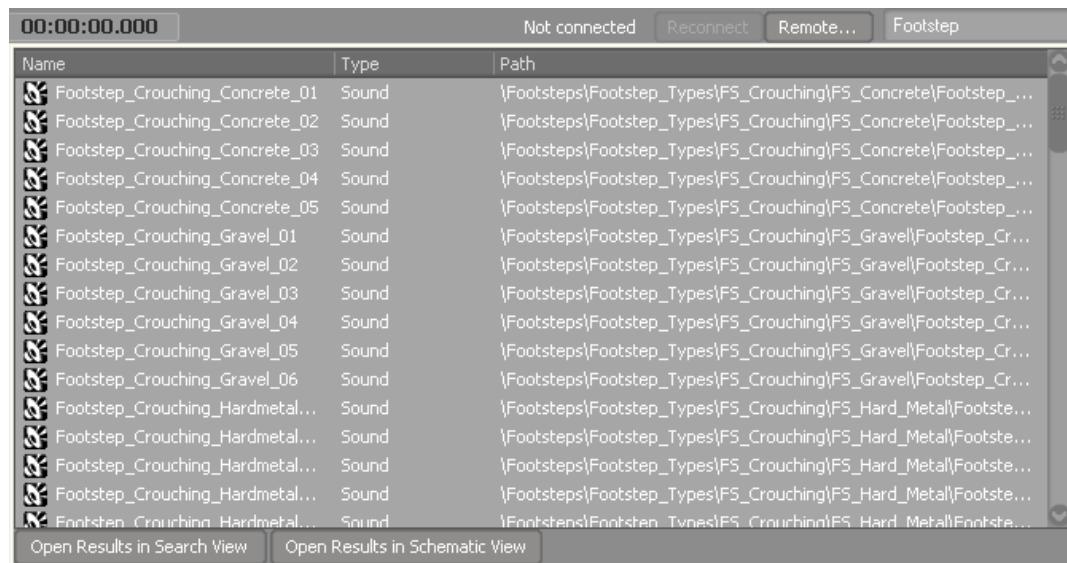
搜索字段文本	匹配	不匹配
	Big Car Bigcar VeryBigCar	
Small Car	Small Fast Car Small_Car Very small car	Smallcar Verysmallcar
mediumcar	mediumcar VeryMediumcar Very_Mediumcar	Medium (数值) Car

您可以使用 List View（列表视图）来执行搜索，不过您还可以使用位于 Wwise 工具栏右侧的 Search 搜索栏来执行快速搜索。从结果列表中，您可以选中工程元素将其加载到相应的编辑器。在使用列表视图时，您还可以右键点击列表中的一系列选项来执行各种任务，包括 Multi-Edit（多项编辑）、Convert（转码）和 Delete（删除）。

搜索工程中元素的方法是：

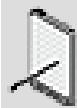
1. 在 Wwise 工具栏中，在 Search 栏中输入工程元素的名称、备注或其中的一部分。

在输入时，搜索字段下面将显示名称/备注匹配的工程元素的列表。



2. 执行以下操作之一：

- 点击 Open Results in List View 可在 List View 中显示结果的完整列表。
- 点击 Open Results in Schematic View 可在 Schematic View 中显示结果的完整列表。
- 点击您要查找的工程元素以将它加载到相应的编辑器。
- 使用箭头键浏览 Results (结果) 列表，然后按 Enter 将所选元素加载到相应编辑器。



备注

如果在 List View 中显示结果，则选择一项可自动地将它加载到 Transport Control 中以便立即播放。您还可以右键点击列表中的一系列条目来执行各种任务，包括 Multi-Edit（多项编辑）、Convert（转码）和 Delete（删除）。

相关主题

- [创建查询](#)
- [定义并运行查询](#)
- [删除查询](#)
- [使用表格](#)
- [使用批量重命名](#)

查找引用特定对象的工程元素

有时候，在项目开发期间，您会需要查找工程中直接引用特定对象的所有元素。例如，您可能想要查找哪些事件引用了特定的对象，或者哪些 SoundBank 引用了特定的事件。在 Wwise 中，使用 Find all reference（查找所有引用）命令非常简便。在大多数右键快捷菜单中可以找到此命令。引用特定对象的所有元素显示在 Reference 视图中，在此您可以轻松地打开各个工程元素，必要时更改该元素，然后继续其它操作。



备注

仅当工程元素直接引用了特定对象或元素时，才会显示在引用视图中。

下表列出了所有 Wwise 工程元素以及可能引用它们的相应元素。

图标	工程元素	可以由以下进行引用:
	Virtual Folder（虚拟文件夹）	<ul style="list-style-type: none"> • SoundBank（音频包）
Master Mixer 图标		
	Audio Bus（音频总线）	<ul style="list-style-type: none"> • Actor-Mixer（角色混音器） • Audio Bus（音频总线） • Audio Source（音频源） • 容器 • Mixing Session（混音会话） • Music Playlist Container（音乐播放列表容器） • Music Segment（音乐段落） • Music Switch Container（音乐切换容器） • Music Track（音乐轨） • Sound SFX（音效声） • SoundBank（音频包） • Soundcaster Session（声音选角器会话） • Source Plug-in（源插件）
	Auxiliary Bus（辅助总线）	<ul style="list-style-type: none"> • Actor-Mixer（角色混音器） • Audio Source（音频源） • 容器

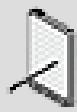
图标	工程元素	可以由以下进行引用:
		<ul style="list-style-type: none"> • Mixing Session (混音会话) • Music Playlist Container (音乐播放列表容器) • Music Segment (音乐段落) • Music Switch Container (音乐切换容器) • Music Track (音乐轨) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话) • Source Plug-in (源插件)
角色混音器图标		
	Audio Source (音频源)	<ul style="list-style-type: none"> • 不适用
	Source Plug-in (源插件)	<ul style="list-style-type: none"> • 不适用
	Sound SFX (音效声)	<ul style="list-style-type: none"> • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Sound Voice (语音声)	<ul style="list-style-type: none"> • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Work Unit (工作单元)	<ul style="list-style-type: none"> • SoundBank (音频包)
	Actor-Mixer (角色混音器)	<ul style="list-style-type: none"> • Audio Source (音频源) • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Blend Container (混合容器)	<ul style="list-style-type: none"> • Audio Source (音频源) • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Random Container (随机容器)	<ul style="list-style-type: none"> • Audio Source (音频源) • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Sequence Container (序列容器)	<ul style="list-style-type: none"> • Audio Source (音频源) • Dialogue Event (对白事件) • Event

图标	工程元素	可以由以下进行引用:
		<ul style="list-style-type: none"> • Mixing Session (混音会话) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Switch Container (切换容器)	<ul style="list-style-type: none"> • Audio Source (音频源) • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
Interactive Music (互动音乐) 图标		
	Music Track	<ul style="list-style-type: none"> • Mixing Session (混音会话) • SoundBank (音频包)
	Music Segment (音乐段落)	<ul style="list-style-type: none"> • Mixing Session (混音会话) • Music Playlist Container (音乐播放列表容器) • Music Switch Container (音乐切换容器) • SoundBank (音频包)
	Music Playlist Container (音乐播放列表容器)	<ul style="list-style-type: none"> • Mixing Session (混音会话) • Music Switch Container (音乐切换容器) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Music Switch Container (音乐切换容器)	<ul style="list-style-type: none"> • Event • Music Track (音乐轨) • Mixing Session (混音会话) • SoundBank (音频包)
其它工程元素图标		
	Event	<ul style="list-style-type: none"> • Event • SoundBank (音频包) • Soundcaster Session (声音选角器会话) • Work Unit (工作单元)
	Dialogue Event (对白事件)	<ul style="list-style-type: none"> • Event • SoundBank (音频包) • Soundcaster Session (声音选角器会话) • Work Unit (工作单元)
	SoundBank (音频包)	<ul style="list-style-type: none"> • 不适用
	Switch Group (切换开关组)	<ul style="list-style-type: none"> • Audio Source (音频源) • 容器 • Event • Mixing Session (混音会话) • Music Switch Container (音乐切换容器) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	Switch (切换开关)	<ul style="list-style-type: none"> • Audio Source (音频源) • 容器 • Event • Mixing Session (混音会话)

图标	工程元素	可以由以下进行引用:
		<ul style="list-style-type: none"> • Music Switch Container (音乐切换容器) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	State Group	<ul style="list-style-type: none"> • Audio Source (音频源) • 容器 • Dialogue Event (对白事件) • Event • Mixing Session (混音会话) • Music Switch Container (音乐切换容器) • Sound SFX (音效声) • SoundBank (音频包) • Soundcaster Session (声音选角器会话)
	State (状态)	<ul style="list-style-type: none"> • Actor-Mixer (角色混音器) • Audio Source (音频源) • 容器 • Dialogue Event (对白事件) • Event • Music Switch Container (音乐切换容器) • Sound SFX (音效声) • Soundcaster Session (声音选角器会话) • Source Plug-in (源插件) • State (状态) • State Group
	Game Parameter (RTPC) (游戏参数 (RTPC))	<ul style="list-style-type: none"> • Actor-Mixer (角色混音器) • Audio Bus (音频总线) • 容器 • Effect (效果器) • Event • Sound SFX (音效声) • Switch Group (切换开关组)
	Trigger (触发器)	<ul style="list-style-type: none"> • Event • Music Playlist Container (音乐播放列表容器) • Music Segment (音乐段落)
	Effect ShareSet (效果共享集)	<ul style="list-style-type: none"> • Auxiliary Bus (辅助总线) • Sound SFX (音效声) • SoundBank (音频包)
	Attenuation ShareSet (衰减共享集)	<ul style="list-style-type: none"> • Actor-Mixer (角色混音器) • Audio Source (音频源) • 容器 • Sound SFX (音效声)
	Conversion Settings ShareSet (转码设置共享集)	<ul style="list-style-type: none"> • Actor-Mixer (角色混音器) • Audio Bus (音频总线) • 容器 • Effect (效果器) • Event • Music Playlist Container (音乐播放列表容器) • Music Segment (音乐段落) • Music Switch Container (音乐切换容器) • Sound SFX (音效声)

图标	工程元素	可以由以下进行引用:
		<ul style="list-style-type: none"> • Sound Voice (语音声) • Switch Group (切换开关组)
	Soundcaster Session (声音选角器会话)	• 不适用
	Mixing Session (混音会话)	• 不适用
	Query (查询)	• 不适用

在大多数情况下，您主要搜索引用某一个特定对象的元素，但您也可以查找引用一系列对象的所有元素。



备注

引用列表不会自动更新，因此，如果对工程做了更改，则需要点击 Refresh 按钮以手动更新引用列表。

查找引用特定对象的工程元素的方法是：

1. 执行以下操作之一：

- 右键点击单个或多个工程元素，然后从快捷菜单中选择 Find all references (查找所有引用)。
- 按 Shift+F3 打开 Reference 视图，然后将一个或一系列对象拖到 References to: (引用) 字段。
- 按 Shift+F3 打开 Reference 视图，点击 References to: 字段旁边的 Browse 按钮，选择工程元素，然后点击 OK。



备注

如果在 Project Explorer、Property Editor 或 Reference View 中选择一个对象或元素的同时按下 Shift+F3 组合键，Wwise 则会自动搜索引用选定对象的对象。

引用选定对象的一系列工程元素显示在 Reference 视图中。

相关主题

- [搜索工程中的元素](#)
- [使用 Query](#)

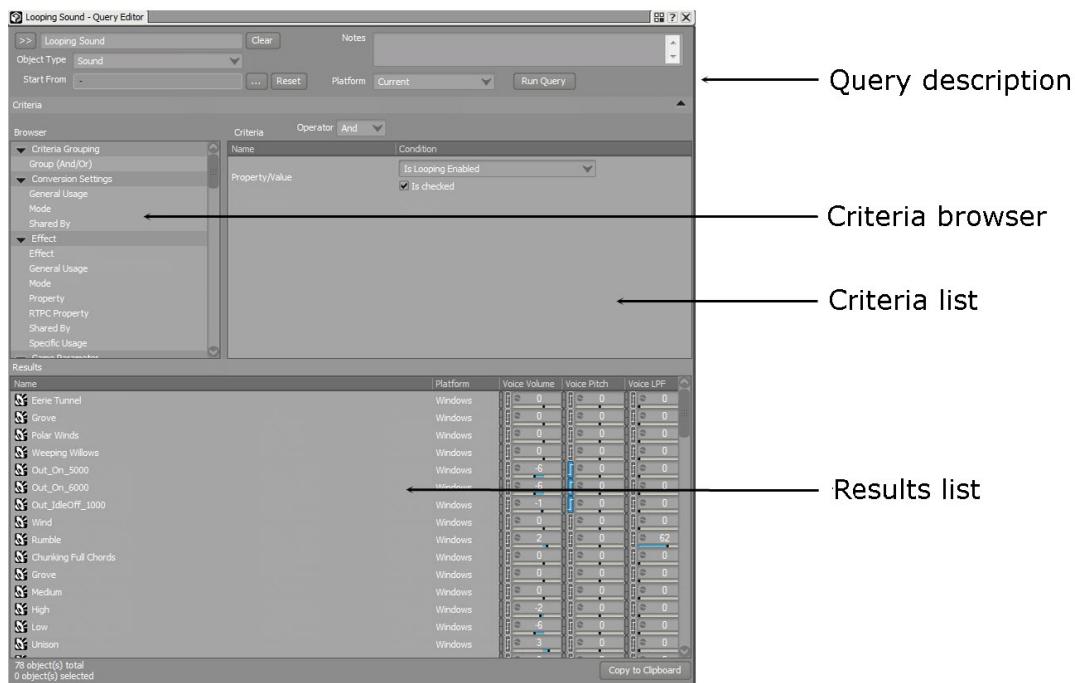
使用 Query

当您需要查找工程中的特定对象而 Search 工具又找不到时，您可以在 Query Editor (查询编辑器) 中创建查询。Query Editor 是一款复杂、强大的工具，可帮助您在工程进行搜索，查找您需要的任何对象。您的查询条件既可以设置得宽泛一些，也可以明确一些，具体取决于需要。您还可以保存和重新加载查询条件，让您的工作流程变得更加高效。在运行查询条件后，您可以一次性编辑 Results 窗格中的所有对象。例如，您可以使用 Multi Editor 开启或关闭流功能，也可以更改 Results 窗格中整组对象的转码设置。

例如，您可以创建并运行查询来执行以下任务：

- 要查找工程中以“sword”开头的所有声音，创建以 SFX 为声音类型、sword* 为名称的声音查询。
- 要查找输出至给定总线的所有 Music Segment，请创建一个针对 Music Segment 的 Query 并查询该总线名称。
- 要查找低通滤波器属性大于 10 的所有声部，创建 sound type（声音类型）为 Voice、Low-Pass Filter 属性值大于 10 的声音查询。

Query Editor 包括以下章节：



创建查询

在 Wwise 中创建查询时，您即建立了一个独特的搜索来查找工程中的对象。在对查询命名后，该查询将保存起来供今后使用。

在 Wwise 中，有两个地方可以创建查询：

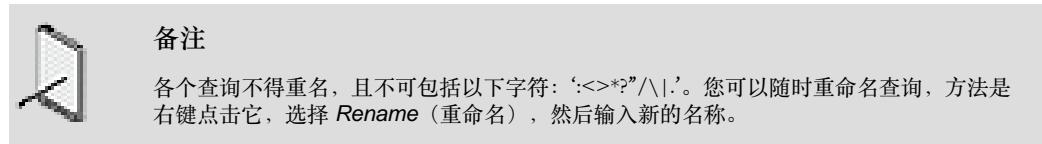
- Project Explorer 的 Queries 选项卡
- 在 Query Editor 中

创建新的查询的方法是：

- 在 Project Explorer 中，切换到 Queries 选项卡。
- 执行以下操作之一：
 - 选择工作单元或虚拟文件夹，然后点击 Project Explorer 工具栏中的 Query 图标。
 - 右键点击工作单元或虚拟文件夹，然后从快捷菜单中选择 New Child > Query（新建子项 > 衰减）。

新的 Query 将被添加到选定的 Work Unit 或 Virtual Folder。

3. 将默认名称替换成最适合该查询操作的名称，然后按 Enter。

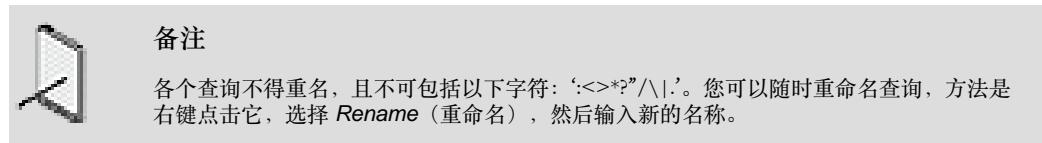


在 Query Editor 中创建新的查询的方法是：

1. 在 Query Editor 中，点击 New...（新建...）

此时将打开 New Query 对话框。

2. 选择想要创建 Query 的 Work Unit。
3. 输入查询的名称。



4. 点击 OK。

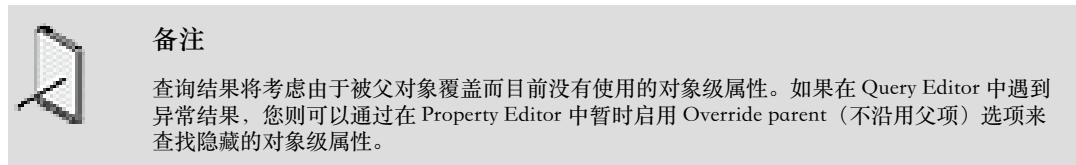
此时将创建新的查询。

相关主题

- [定义并运行查询](#)
- [使用搜索条件组创建高级查询](#)
- [使用出厂设定的查询](#)
- [删除查询](#)
- [搜索工程中的元素](#)

定义并运行查询

第一次运行查询时，需要决定使用什么标准来搜索工程。这些标准会自动保存，在您今后再次运行此查询时可以为您节省时间。



定义查询的内容的方法是：

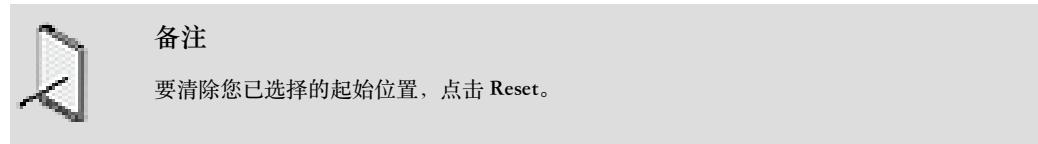
1. 将查询加载到 Query Editor 中。
2. 在 Notes 字段中输入有关查询的任何附加信息。
3. 从 Object Type（对象类型）列表中，选择您要搜索的对象或其它工程元素的类型。如果您不想搜索特定类型的对象，则选择 All Objects（全部对象）。
4. 要在层级结构中指定查询的起始位置，则需执行以下操作：

在 Start From Here (从此处开始) 框附近，点击 Browse 按钮 (...)。

Project Explorer - Browser 对话框将打开。

前往您希望 Wwise 开始搜索的文件夹。

点击 OK。



5. 从 Platform 列表中选择以下其中一个选项：

- All (全部)，以搜索用于工程所包含的任何平台上的对象。
- Current (当前)，以搜索用于当前选定平台上的对象。

6. 在浏览器列表中，选择查询所依据的各个条件。执行以下操作：

双击某个标准，以将它加载到 Criteria 列表。

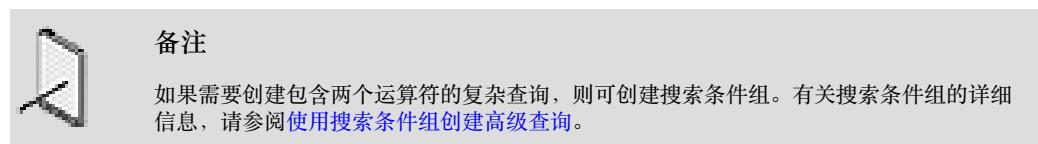
为此标准输入或选择条件。在输入标准时，您可以使用通配符 (*) 来替代词的一部分。



7. 在 Operator (运算符) 列表中，执行以下操作之一：

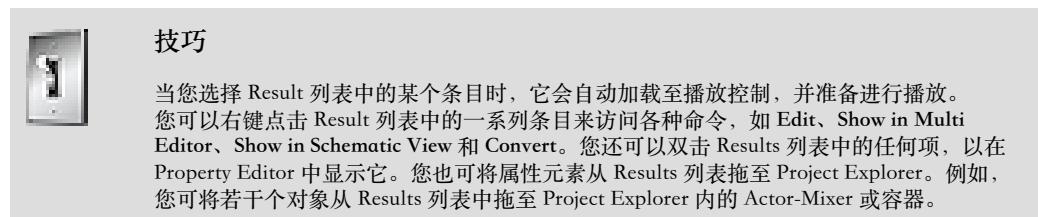
And，以返回满足所有标准的结果。

Or，以返回满足至少一个标准的结果。



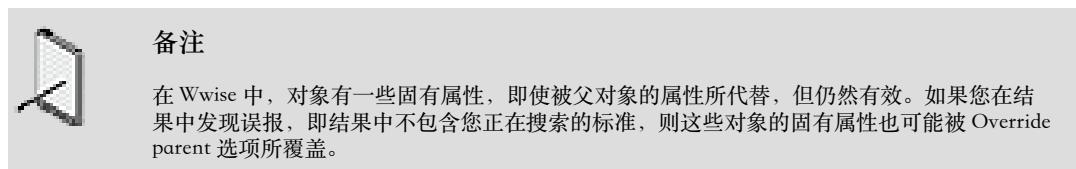
8. 点击 Run Query (运行查询)。

此时将会显示查询结果。您可以通过点击各个列标题来对结果进行排序。



9. 要保留查询结果以便在另一应用程序中使用，则点击 Copy to Clipboard (复制到剪贴板)。

10. 要清空标准和搜索结果，则点击 **Clear**。

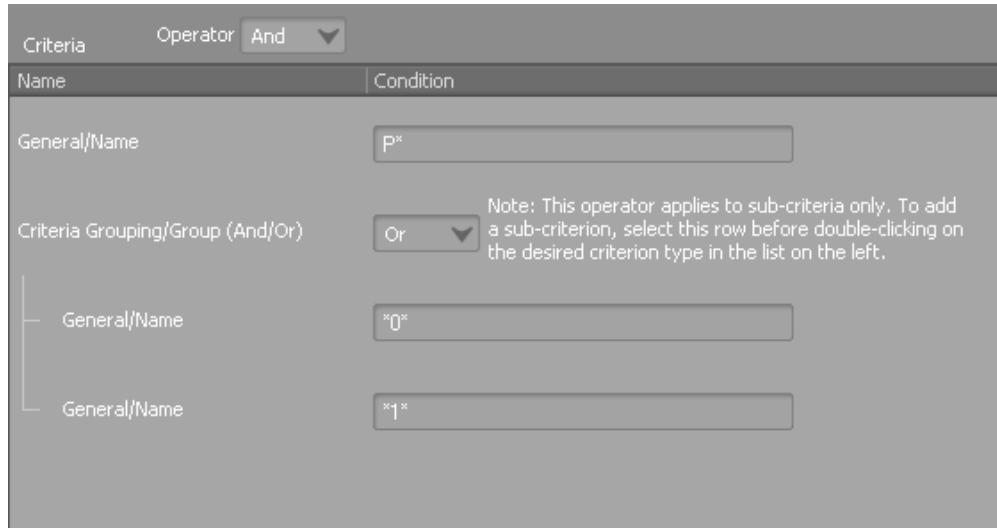


相关主题

- [创建查询](#)
- [使用搜索条件组创建高级查询](#)
- [使用出厂设定的查询](#)
- [删除查询](#)
- [搜索工程中的元素](#)

使用搜索条件组创建高级查询

在某些情况下，使用一个运算符的基本查询不够具体，无法准确地找到您正在查找的内容。在这些情况下，您可以使用搜索条件组来创建更加高级的搜索。搜索条件组是具有独立运算符控制的标准的子集。它可以让您使用不同类型的运算符来创建查询。例如，您可以创建一个查询来搜索名称以字母 P 开头并 (AND) 包含数字 0 或 (OR) 1 的所有对象。



您可以在该查询中使用若干个条件组来查找工程中的特定对象或其它元素。搜索条件组甚至可以与其它搜索条件组进行嵌套，创建出极其复杂的查询。

在查询中创建搜索条件组的方法是：

1. 将查询加载到 Query Editor 中。
2. 在 Browser 列表中，双击 Criteria Grouping (标准编组) 部分中的 Group (And/Or) (编组 (与/或)) 选项。

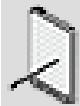
新的搜索条件组添加到 Criteria 列表中。

3. 在 Browser 列表中，通过执行以下操作添加若干个标准到搜索条件组：

选择 Criteria 列表中的搜索条件组。

双击标准以将它添加到搜索条件组。

为此标准输入或选择条件。在输入标准时，您可以使用通配符（*）来替代词的一部分。



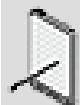
备注

要对组中的标准重新排序，只需将标准拖到新位置即可。标准一旦添加到编组中，就不能将它拖到编组外。

4. 从 Criteria Group Operator（搜索条件组运算符）列表中，选择以下其中一项：

And（与），以返回满足编组中所有标准的结果。

Or——用于返回至少满足编组中一个标准的结果。



备注

要在编组外添加另一个标准，在编组外进行点击以取消选中它，然后双击 Browser 列表中的标准。

5. 点击 Run Query（运行查询）。

此时将会显示查询结果。您可以通过点击各个列标题来对结果进行排序。



技巧

当您选择 Result 列表中的某个条目时，它会自动加载至播放控制，并准备进行播放。您可以右键点击 Result 列表中的一系列条目来访问各种命令，如 Edit、Show in Multi Editor、Show in Schematic View 和 Convert。您还可以双击 Results 列表中的任何项，以在 Property Editor 中显示它。您也可将属性元素从 Results 列表拖至 Project Explorer。例如，您可将若干个对象从 Results 列表中拖至 Project Explorer 内的 Actor-Mixer 或容器。

6. 要保留查询结果以便在另一应用程序中使用，则点击 Copy to Clipboard（复制到剪贴板）。
7. 要清空标准和搜索结果，则点击 Clear。

相关主题

- [定义并运行查询](#)
- [使用出厂设定的查询](#)
- [删除查询](#)

使用出厂设定的查询

Wwise 为许多常见查询提供了一些预定义出厂预设。使用它们比从头开始定义查询节省时间。

运行出厂定义的查询的方法是：

1. 在 Project Explorer 中，切换到 Queries 选项卡。
2. 在 Factory Queries（工厂定义的查询）列表中，双击查询。
该查询加载到 Query Editor 中。
3. 根据需要为标准输入或选择条件。在输入标准时，您可以使用通配符 (*) 来替代词的一部分。
4. 点击 Run Query（运行查询）。

此时将会显示查询结果。您可以通过点击各个列标题来对结果进行排序。



技巧

当您选择 Result 列表中的某个条目时，它会自动加载至播放控制，并准备进行播放。您可以右键点击 Result 列表中的一系列条目来访问各种命令，如 Edit、Show in Multi Editor、Show in Schematic View 和 Convert。您还可以双击 Results 列表中的任何项，以在 Property Editor 中显示它。您也可将属性元素从 Results 列表拖至 Project Explorer。例如，您可将若干个对象从 Results 列表中拖至 Project Explorer 内的 Actor-Mixer 或容器。

5. 要保留查询结果以便在另一应用程序中使用，则点击 Copy to Clipboard（复制到剪贴板）。
6. 要清空标准和搜索结果，则点击 Clear。

相关主题

- [创建查询](#)
- [使用搜索条件组创建高级查询](#)
- [删除查询](#)
- [搜索工程中的元素](#)

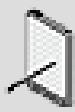
删除查询

您可以删除不再需要的查询。

删除查询的方法是：

1. 在 Project Explorer 中，切换到 Queries 选项卡。
2. 右键点击您要删除的查询，并选择 Delete Selection（删除选中对象）。

选中的查询被删除。



备注

如果您无意中删错了查询，则可以按 Ctrl+Z 或点击 Edit（编辑）> Undo（撤消）来撤消删除操作。

相关主题

- [创建查询](#)

- [使用搜索条件组创建高级查询](#)
- [使用出厂设定的查询](#)
- [定义并运行查询](#)
- [搜索工程中的元素](#)

查询 —— 技巧和经验总结

在 Wwise 中创建查询时，有许多选项可用。在设计查询时运用特定策略，可以准确、快速、稳定地获得您需要的结果。下面是在工程中使用效果器时您可能需要考虑的一些策略。

对象类型详情

Browser 列表中的许多标准都有条件，可以让您进一步缩小查询范围。然而，这些条件根据您选择的对象类型变化。例如，对于 Property Value 标准，请考虑以下三种情形：

- 如果选择“Audio Source”作为对象类型，则有包括位深和采样率等一系列条件可供您选择。
- 如果选择“Event”作为对象类型，则无条件可用，因为它对于此对象类型不是有效的标准（在 Wwise 中，事件没有属性）。

因此，对象类型的选择非常重要，它决定了您可以用于查询的条件。

加快查询速度

几个简单的选择可以让您的查询运行速度变得更快。

- 指定对象类型。
- 如果您要查找平台专有的内容，则选择平台。
- 使用 **Start From Here**（从此处开始）来选择层级结构中尽可能深的位置。

在查询中使用通配符

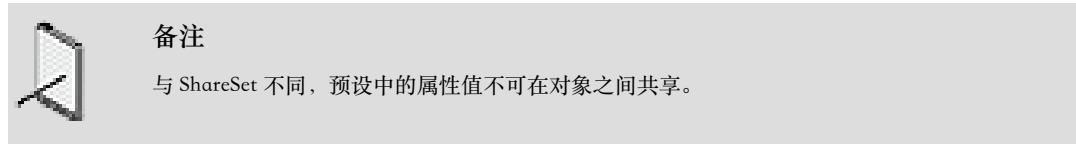
通配符是一些符号，它们通过扩展搜索词的参数来提高关键词搜索的灵活性。在您试图查找包含以特定字母或数字开头或结尾的不同工程元素时，可以使用通配运算符来代替搜索词的一部分。在 Wwise 中，通配符是星号 (*)。例如，要查找名称中包含 LOOP 的所有对象，应在条件文本框中输入 *LOOP*。而要查找以字母 LOOP 开头的所有对象，应在条件文本框中输入 LOOP*。最后，要查找以字母 LOOP 结尾的所有对象，应在条件文本框中输入 *LOOP。

第 46 章 使用 Preset

概述	793
使用 Preset	793

概述

Preset（预设）是与对象或效果相关的一组特定属性值，这些属性值保存在一个专门的文件中，以便稍后在同一个工程中重复使用它们。通过使用预设，您无需重新创建您想重新用于工程中其它对象的特定属性设置。您只需要设置一次这些属性值，保存预设，然后将预设应用于工程中的其它对象即可。这可以提高工作效率，即省时又省力。



使用 Preset

预设图标显示在各个视图的标题栏中，在标题栏中可以保存和加载预设。



保存预设

在 Wwise 中，可以为以下任何元素保存预设：

- Actor-Mixer Hierarchy 和 Interactive Music Hierarchy 中对象和源的属性值和设置
- 定位值
- 衰减设置
- 效果器值

保存预设时，Wwise 会保存视图中各个选项卡上的各个值。保存预设后，预设将按以下其中一个类别分组：

- Audio Bus（音频总线）
- Auxiliary Bus（辅助总线）
- Actor-Mixer（角色混音器）
- Random or Sequence Container（随机或序列容器）
- Switch Container（切换容器）
- Blend Container（混合容器）
- Sound SFX/Voice（音效 / 语音对象）
- Audio Source Plug-in（音频源插件）
- Music Switch Container（音乐切换容器）
- Music Playlist Container（音乐播放列表容器）
- Music Segment（音乐段落）
- Music Track（音乐轨）
- 定位（3D 自动化）

- 定位 (2D)
- Attenuation (衰减)
- Effects

打开 Save Preset 对话框时，将对预设过滤，只显示属于同一类别的预设。

保存预设的方法是：

1. 在视图标题栏中，点击 Save Preset 图标。

此时将打开 Save Preset 对话框。

2. 在 Name 字段中，输入预设的名称。
3. 在 Notes 字段中，输入进一步描述预设的任何信息。
4. 点击 Save 或按 Enter。

预设于是得到保存，在同一工程中可随时重新使用该预设。

相关主题

- [加载预设](#)
- [删除预设](#)

预设的内容

预设于是得到保存，在同一工程中可随时重新使用该预设。下表描述了所有预设及其保存的信息。

Preset (预设)	层级结构中的位置	内容
对象/源属性	顶层对象	Property Editor 中各个选项卡上的所有属性值。
	子对象	Property Editor 中各个选项卡上的所有属性值。 若保存 Preset 时未启用 Override parent 选项，则 Override 设置下的内容将不会保存到 Preset 中。若保存到 Preset 时启用了该选项，则 Preset 中将包含所有相应的 Override 属性值。但是，系统默认不启用 Override 选项。只有启用 Override parent 后，才能使用 Override 属性。
Effect (效果器)	-	Effect 的所有属性值。
定位 (2D)	-	点声源的位置。
定位 (3D 自动化)	-	Position Editor (3D Automation) 中的所有定位路径、设置和值。
Attenuation (衰减)	-	所有曲线和所有衰减设置。

加载预设

保存预设后，可以将它们应用于工程中的其它对象或效果器。保存预设后，预设将按以下其中一个类别分组：

- Audio Bus (音频总线)

- Auxiliary Bus (辅助总线)
- Actor-Mixer (角色混音器)
- Random or Sequence Container (随机或序列容器)
- Switch Container (切换容器)
- Blend Container (混合容器)
- Sound SFX/Voice (音效 / 语音对象)
- Audio Source Plug-in (音频源插件)
- Music Switch Container (音乐切换容器)
- Music Playlist Container (音乐播放列表容器)
- Music Segment (音乐段落)
- Music Track (音乐轨)
- 定位 (3D 自动化)
- 定位 (2D)
- Attenuation (衰减)
- Effects

打开 Load Preset 对话框时，将对预设过滤，只显示属于同一类别的预设。

加载预设的方法是：

1. 在视图标题栏中，点击 Load Preset 图标。

此时将打开 Load Preset 对话框。

2. 从列表中选择预设。
3. 点击 Load 或按 Enter。

此预设已应用到对象或效果。

相关主题

- [保存预设](#)
- [删除预设](#)

删除预设

如果不再需要某个预设，则可以删除它。

删除预设的方法是：

1. 在视图标题栏中，点击 Load Preset 图标。

此时将打开 Load Preset 对话框。



技巧

还可以从 Save Preset 对话框中删除预设。

2. 在 Preset 列表中，点击您要删除的预设。

此时 Delete 按钮变成已启用。

3. 点击 Delete。

该预设于是从 Preset 列表和工程中被删除掉了。

相关主题

- [保存预设](#)
- [加载预设](#)

第 47 章 使用控制设备

概述	798
将控制设备连接到 Wwise	798
创建 Control Surface Session	799
理解控制设备绑定	799
创建控制设备绑定	800
理解 Control Surface 的 View Group	803
处理控制 Control Surface Session 中的冲突	804
使用 Control Surface 工具栏	804

概述

控制设备（Control Surface）可以用来控制 Wwise 的功能或工程属性。Wwise 支持 MIDI 协议和 Mackie 协议。在用 TouchOSC Bridge 配对后，还可以将 Wwise 与 iOS 设备和 Android 设备上的 TouchOSC 应用程序一起使用，该应用程序支持 MIDI。

将控制设备连接到 Wwise

Wwise 支持两类控制设备协议：

- MIDI 协议
- Mackie HUI MIDI 映射协议（MCU Pro）

在您开始前：

- 确保设备物理连接到计算机。
- 确保设备已打开。
- 确保您已为设备安装正确的驱动程序。



将设备连接到 Wwise 的方法如下：

1. 从 Project（工程）菜单中，选择 Control Surface Devices（控制设备）。
2. 点击 Add（添加）按钮。
3. 为设备命名。
4. 点击 OK（确定）。

设备现在添加到列表里了。

5. 在 Receive From（输入端）列中，选择 MIDI IN（MIDI 输入）设备。

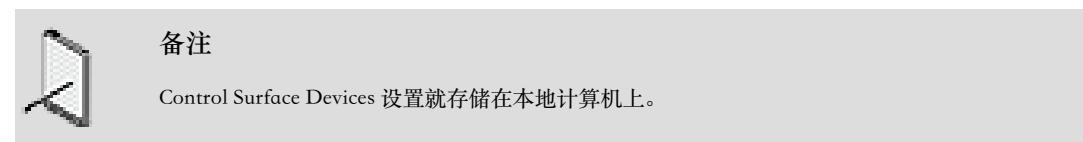
此时将会显示 Connected（已连接）消息。

6. 在 Send To（输出端）列中，选择 MIDI OUT（MIDI 输出）设备。

此时将会显示 Connected 消息。

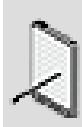
7. 点击 Close（关闭）。

设备现在就可以用了。



创建 Control Surface Session

Control Surface Session（控制设备会话）定义硬件控件连接到 Wwise 功能或工程属性的方式。Control Surface Session 定义一列绑定。各个绑定都将一个硬件控件（按钮、滑杆、旋钮、键等）连接到 Wwise 元素（属性或命令）上。



备注

Control Surface Sessions 存储在工程内，并且可用在任何使用此工程的计算机上。

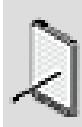
创建控制 Control Surface Sessions 的方法如下：

1. 从 View（视图）菜单中，打开 Control Surface Binding（控制设备绑定）视图（Ctrl+Shift+Q）。
2. 点击 [>>] 按钮打开选择器菜单。
3. 选择 New...（新建...）
4. 为会话输入名称。
5. 按 OK。

此时将会创建并加载会话。

在单个 Wwise 工程中可以创建多个 Control Surface Sessions，允许这样做有一些理由：

- Wwise 工程可能有多个用户，这些用户会使用不同的硬件设备。
- 不同的使用情形。
- 不同的用户有不同的需求或偏好。



备注

尽管一个工程中可以有多个会话，但一次只能有一个会话处于激活状态。

理解控制设备绑定

通过 Control Surface 绑定，可以将一个硬件控件（按钮、滑杆、旋钮、键等）连接到 Wwise 元素（属性或命令）上。

每个绑定有三个元素：

- **Property/Command**（属性/命令）：
 - Object Property：对象属性。目标对象上要修改的属性名。
 - Object Command：对象命令。目标对象上要启动的命令或操作。
 - Global Command：全局命令。要全局启动的命令或操作（不针对对象）。
- **Object/Index**：对象/索引。指定目标对象。

- **Controller Assignment:** 控制器指派。用 MIDI 消息 ID 来识别硬件控件元素。

绑定项存储在三个不同组内，每个组都具有定义目标对象的不同机制。

- **Global:** 全局。目标对象直接在绑定中指定。
- **Current Selection:** 当前选中项。目标对象是 Wwise 中最新选中的对象。
- **View Group:** 视图组。根据当前加载 Binding Group 的视图（如 Mixing Desk 和 Mixing Session，或 Soundcaster 和 Soundcaster Session），决定控制哪些目标对象。视图中加载的每个对象都有一个索引。

相关主题

- [为键盘快捷键创建绑定。](#)
- [创建绑定以修改特定对象属性值](#)
- [创建绑定以修改当前选中项](#)

创建控制设备绑定

创建 Binding（绑定）可以有多种情况：

- [为键盘快捷键创建绑定。](#)
- [创建绑定以修改当前选中项](#)
- [创建绑定以修改特定对象属性值](#)

为键盘快捷键创建绑定。

Global Binding（全局绑定）可用于触发全局命令，这些命令和 Keyboard Shortcuts（键盘快捷键）管理器中的命令相同。您可以创建一个由 Keyboard Shortcut 管理器和 Control Surface Session 的绑定触发的全局命令。

在 Control Surface Binding 视图中创建全局绑定的方法如下：

1. 选择 Global（全局）组。
2. 点击 Add & Learn Binding（添加和学习绑定）按钮。

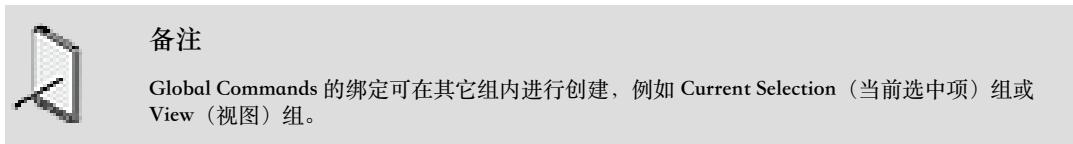
绑定条目将被添加，并且 Learn 按钮也将激活。

Properties/Command 可在 UI 变绿时进行选择。

3. 点击 [>>] 按钮选择命令。
4. 从菜单中选择 Global commands...（全局命令...）
5. 浏览找到要启动的全局命令。
6. 点击 OK。
7. 使用所需的硬件控件来作绑定。

Learn（学习）按钮处于已停用状态。

绑定现在创建好并且就绪了。



创建绑定以修改特定对象属性值

可创建 Control Surface Binding（控制设备绑定），以指向工程中特定对象的属性值。这对于把控制器指派到 Game Parameter 这样的对象上来说非常有用。

创建控制 Game Parameter 模拟值的绑定的方法如下：

1. 选择 Global 组。
2. 点击 Add & Learn Binding（添加和学习绑定）按钮。

绑定条目将被添加，并且 Learn 按钮也将激活。

Properties/Command 可在 UI 变绿时进行选择。

3. 通过任一操作即可选择 Property/Command：
 - 点击 Transport Control（播放控制）中的游戏参数的模拟值。
 - 点击 [>>] 按钮并浏览找到该属性：
 - 从菜单中选择 Object Properties...（对象属性...）
 - 浏览找到 Games Syncs > Game Parameter > Simulation Value（游戏同步 > 游戏参数 > 模拟值）。
 - 点击 OK。
4. 使用所需的硬件控件来作绑定。

Learn（学习）按钮处于已停用状态。

绑定现在创建好并且就绪了。

创建绑定以修改当前选中项

Control Surface Session 可定义用于将当前选中项的属性值连接到硬件控制器控件（滑杆、旋钮和按钮）的绑定。

典型设置是将控制设备的四个滑杆指派给：

- Voice Volume（声部音量）。
- Voice Pitch（声部音高）。
- Voice Low-pass Filter（声部低通滤波器）。
- Voice High-pass Filter（声部高通滤波器）。

在创建绑定后，四个滑杆将自动连接到所选对象的这些属性。如果您有电动推子（例如：Mackie Control Universal Pro），则在当前选中项更改时这些推子将自动移至当前值。

创建修改当前选中项 Voice Volume 的绑定的方法如下：

1. 选择 Current Selection (当前选中项) 组。
2. 点击 Add & Learn Binding (添加并学习绑定) 按钮。

绑定条目将被添加，并且 Learn 按钮也将激活。

Properties/Command 可在 UI 变绿时进行选择。

3. 通过任一操作即可选择 Property/Command：
 - 点击 Property Editor 中的 Voice Volume (声部音量)。
 - 点击 [>>] 按钮并浏览找到该属性：
 - 从菜单中选择 Object Properties...
 - 浏览找到 Audio > General Settings > Voice Volume (音频 > 常规设置 > 声部音量)。
 - 点击 OK。
4. 使用所需的硬件控件来作绑定。

Learn (学习) 按钮处于已停用状态。

绑定现在创建好并且就绪了。

当前选中项绑定还可用于触发当前选中项上的命令。例如，您可以将按钮阵列映射到：

- Mute (静音)
- Solo (单独播放)
- Play
- Stop (停止)

创建对当前选中项静音的绑定的方法如下：

1. 选择 Current Selection (当前选中项) 组。
2. 点击 Add & Learn Binding (添加并学习绑定) 按钮。

绑定条目将被添加，并且 Learn 按钮也将激活。

Properties/Command 可在 UI 变绿时进行选择。

3. 选择属性：
 - 点击 [>>] 按钮。
 - 从菜单中选择 Object commands > Mute (对象命令 > 静音)。
4. 使用所需的硬件控件来作绑定。

Learn (学习) 按钮处于已停用状态。

绑定现在创建好并且就绪了。

如果您有控制器按钮，则在静音时指示灯灯将会点亮而在取消静音时指示灯灯将会熄灭。

理解 Control Surface 的 View Group

View Group（视图组）可用于将特定视图的内容（对象）绑定到 Binding Group（绑定组）内容。例如，您可以将 Mixing Desk（调音台）的对象映射到 Control Surface（控制设备）。

View Group 包含与视图中的对象关联的绑定，这些对象由视图中的索引进行标识。

以下是支持 View Group 的视图以及这些视图指派索引的方式：

- Mixing Desk：调音台。各个垂直条带一个索引。
- List View, Query Editor, Reference View, Master-Mixer Console：列表视图、查询编辑器、引用视图、主混音器控制台。每行一个索引。
- Property Editor, Effect Editor, Source Editor, Modulator Editor：属性编辑器、效果器编辑器、源编辑器、调制器编辑器。只能使用 1 号索引，代表当前对象。
- Soundcaster：声音选角器。每个模块一个索引。

创建 View Group 的方法如下：

1. 在 Control Surface Binding 视图中，点击 View Group 文件夹
2. 点击 Add Group（添加组）按钮。
3. 对组进行命名。
4. 按 OK。
5. 使用 Add & Learn Binding 按钮创建组内绑定。

有关详细信息，请参阅[创建控制设备绑定](#)。

以下是在 Mixing Desk、List View 或 Soundcaster 内的 Control Surface 的 View Group 示例：

- 混音 View Group：
 - 绑定：Property:Voice Volume - Index:1 - Key:MyController Ch.1 CC 0
 - 绑定：Property:Voice Volume - Index:2 - Key:MyController Ch.1 CC 1
 - 绑定：Property:Voice Volume - Index:3 - Key:MyController Ch.1 CC 2
 - 绑定：Property:Voice Volume - Index:4 - Key:MyController Ch.1 CC 3
 - 绑定：Command:Solo - Index:1 - Key:MyController Ch.1 CC 32
 - 绑定：Command:Solo - Index:2 - Key:MyController Ch.1 CC 33
 - 绑定：Command:Solo - Index:3 - Key:MyController Ch.1 CC 34
 - 绑定：Command:Solo - Index:4 - Key:MyController Ch.1 CC 35
 - 绑定：Command:Mute - Index:1 - Key:MyController Ch.1 CC 64
 - 绑定：Command:Mute - Index:2 - Key:MyController Ch.1 CC 65
 - 绑定：Command:Mute - Index:3 - Key:MyController Ch.1 CC 66
 - 绑定：Command:Mute - Index:4 - Key:MyController Ch.1 CC 67

此 View Group 会将：

- 4 个硬件滑杆映射到 4 个对象的 Voice Volume。
- 4 个硬件按钮映射到 4 个对象的 Solo。
- 4 个硬件按钮映射到 4 个对象的 Mute。

将 View Group 与 Mixing Desk 相关联的方法如下：



备注

您必须有活跃的 Control Surface Session，并至少为这些步骤创建一个视图组。

1. 从 View 菜单中，打开 Mixing Desk 视图（Ctrl+Shift+M）。
2. 点击 Mixing Desk 右上角标题栏上的 [>>] 按钮。
3. 从选择器菜单中，选择要使用的 View Group。

此时已加载该 View Group。



备注

加载到视图中的视图组也会显示在 Wwise 工具栏中。这些视图组可从工具栏中加以激活和停用。

处理控制 Control Surface Session 中的冲突

输入的多个活跃的绑定有可能与其 Controller Assignments（控制器指派）发生冲突。Control Surface 系统不允许同时加载的多个绑定具有相同的键位。



备注

通过 Control Surface Session 进行的绑定会从上到下进行加载。在发生冲突时，将只会加载起冲突的几个绑定中的第一个。系统会优先考虑最先找到的绑定。

发现冲突时：

- 在 Wwise 工具栏中的组名旁会显示黄色三角。
- 在 Control Surface Session 视图内的绑定旁会显示黄色消息。

处理冲突的方式有很多种：

- 对 Control Surface Session 视图中的绑定重新排序，以更改优先级。
- 使用绑定条目上的快捷菜单来解决冲突。
- 为起冲突的绑定中的某一个指派新的控制器。



技巧

在某些情况下，把同一个控制器指派给同一组中的不同绑定可能会有用。例如，您可以创建一个 Effect View Group 用来绑定不同的效果器属性。因为 Effect Editor 一次只能加载一个效果器，所以绑定不会发生冲突。

使用 Control Surface 工具栏

Control Surface（控制设备）工具栏显示以下元素：

界面元素	描述
后退按钮	前往上一选定的对象（Backspace）。
前进按钮	在导航列表中前进（Shift+Backspace）
当前选中对象	显示当前所选对象。
Last Property Changed	显示上次使用 Control Surface 绑定修改的对象、属性和值（用滑杆表示）。
已加载的绑定组	<p>这些按钮显示以下各绑定组是否处于活跃状态：</p> <ul style="list-style-type: none">• Global：全局。目标对象在全局绑定中定义。• Current Selection：当前选中对象。目标对象为当前选中对象。• View Group：视图组。目标对象通过指定视图在指定组中加载。例如，按钮可能会显示为 YourGroup: Soundcaster，表示工程中包含名为“YourGroup”的视图组，并且用户已将其用于 Soundcaster。可能有多个视图加载了多个 View Group，因此显示的按钮可能不止一个。 <p>点击可更改组活跃的组名。</p> <ul style="list-style-type: none">• Active：活跃。蓝色• Inactive：不活跃。灰色



备注

Control Surface 工具栏仅在 Control Surface Binding 视图中加载了 Control Surface Session 时才可见。

部分 VIII. 附录



A. 下混行为	808
标准配置	809
下混表	810
B. 正则表达式快速参考指南	814
C. 快捷键	816
可缩放的编辑器	817
Attenuation Editor 视图	817
音频文件管理	817
Contents Editor 视图	818
Game Object 3D Viewer 视图	818
Game Profiler	818
全局——上下文（活跃视图和选定对象）	818
Music Segment Editor 视图	819
Position Editor (3D Automation)	819
Project Explorer 视图	820
RTPC 坐标图视图	820
Schematic View 视图	820
Soundcaster 视图	820
Transport Control 视图	820

附录 A. 下混行为

目录

标准配置	809
下混表	810

当在不进行平移/定位的情况下使用不同的声道配置将声部或总线（输入）连到总线（输出）时，Wwise 将根据下文所述的规则进行下混（Downmixing）。

标准配置

标准配置全部为典型配置，例如立体声和 5.1，其中各个声道的名称都提示点了相应扬声器的布置。例如，“Height Back Left”声道放置在听者的背后和左侧，并位于空中。

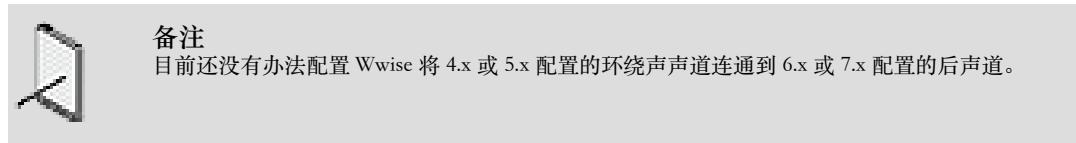
LFE

LFE (.1) 不会被下混。如果输入和输出配置都有 LFE 声道，则输入的 LFE 声道信号将被直接复制到输出的相应声道。如果输出没有 LFE 声道，则丢弃输入的 LFE 声道信号。

环绕声道与侧声道

在 4.x 和 5.x 配置中，只有一对扬声器不在前方，因此“侧”声道或“后”声道之间没有区别。它们都是“环绕”的同义词。对于这些配置，我们选择始终调用环绕声声道 SL 和 SR，其中“S”代表“surround”（环绕声）。这导致做出了有些武断的决定，即将 4.x 和 5.x 配置的环绕声声道 SL 和 SR 映射到 6.x 和 7.x 配置的侧声道 SL 和 SR，而不是后声道 BL 和 BR。这种偏好的合理性在于 5.1 配置中的环绕声扬声器的首选角度（正如 Dolby 所提倡的）可以匹配 7.1 配置中的侧扬声器的角度。

详情请参阅 <http://www.dolby.com/us/en/guide/surround-sound-speaker-setup/5-1-setup.html> 和 <http://www.dolby.com/us/en/guide/surround-sound-speaker-setup/7-1-setup.html>。



归一化

鉴于归一化系数的随意性，Wwise 下混不执行归一化。因此，从具有较多声道的配置下混到具有较少声道的配置后常常会造成响度增大的情况。设计师应该根据他们的喜好在总线音量上使用 State 或 RPTC 来处理归一化。

具有高度声道的配置

目前具有“高度声道”（HXX）的配置不存在下混表。高度声道像 LFE 声道一样处理。这意味着当这些声道存在于输出中时，则直接映射这些声道，否则丢弃之。

例如，如果将 7.0.4 输入（FL-FR-C-SL-SR-BL-BR-HFL-HFR-HBL-HBR）连通到 5.0.2 输出（FL-FR-C-SL-SR-HFL-HFR）就会发生这种情况。

- 根据[至 5.x](#) 表格的等式，输入的 BL 和 BR 下混到输出的 SL 和 SR。
- 输入的 HFL 和 HFR 传递到输出的 HFL 和 HFR。
- 由于输入的 HBL 和 HBR 在输出中不存在，因此将丢弃之。

其它配置类型

匿名和 Ambisonics 配置不执行下混。通路的工作方式与标准配置中的 LFE 声道相似。如果输出中存在相应声道，则直接复制之；否则丢弃之。

下混表

下混等式是基于 [AC-3](#) 标准。无论信号是直接下混（例如 7.1 下混为立体声）还是分步下混（例如 7.1 下混到 5.1，然后再下混到立体声），它们的最终结果都一样。

如何阅读下混表

请考虑下表，表中描述了将 5.1 输入连通到立体声输出时使用的下混等式。

输入声道	L	R
FL	1	0
FR	0	1
C	$1/\sqrt{2}$	$1/\sqrt{2}$
SL	$1/\sqrt{2}$	0
SR	0	$1/\sqrt{2}$

下混获得的立体声信号的左右声道是这些等式的输入值的加权和：

$$\begin{aligned} L_{\text{OUT}} &= FL + 1/\sqrt{2} * C + 1/\sqrt{2} * SL \\ R_{\text{OUT}} &= FR + 1/\sqrt{2} * C + 1/\sqrt{2} * SR \end{aligned}$$

参考表

与上面公式一样，您可以计算各个下混格式的输出信号，方式是将下文下混参考表中的各个输出声道列的总计加起来：

- [至单声道](#)
- [至立体声](#)
- [至 3.x](#)
- [至 4.x](#)
- [至 5.x](#)
- [至 6.x](#)
- [至 7.x](#)

至单声道

输入声道	C
FL	$1/\sqrt{2}$
FR	$1/\sqrt{2}$
C	1
SL	$1/2$
SR	$1/2$
BL	$1/2$
BR	$1/2$

下混行为

输入声道	C
Top	1
HFL	1/2
HFC	1/ $\sqrt{2}$
HFR	1/2
HBL	1/(2 $\sqrt{2}$)
HBR	1/(2 $\sqrt{2}$)

至立体声

输入声道	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

至 3.x

输入声道	L	R	C
FL	1	0	0
FR	0	1	0
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

至 4.x

输入声道	L	R	SL	SR
FL	1	0	0	0
FR	0	1	0	0

下混行为

输入声道	L	R	SL	SR
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	$1/\sqrt{2}$

至 5.x

输入声道	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	$1/\sqrt{2}$

至 6.x

输入声道	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	$1/\sqrt{2}$

至 7.x

输入声道	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	$1/\sqrt{2}$

附录 B. 正则表达式快速参考指南

锚点	
^	字符串的开头或者多行模式中的行首
\$	字符串的末尾，或者多行模式中的行末尾
\b	单词（word）边界
\B	非单词边界

字符类别	
\s	空格
\S	非空格
\d	数字
\D	非数字
\w	单词
\W	非单词
\x	十六进制数 用于匹配十六进制代码字符，例如： • “\xA9”将搜索到“0xA9”，代表版权符号 • “\x5B”将搜索到“0x5B”，代表正方括号

数量词	
*	空格
+	非空格
?	数字
{3}	正好 3 个
{3,}	3 个或 3 个以上
{3,5}	3 个、4 个或 5 个

转义字符序列	
\	对下列字符进行转义 ^ . \$ { * (\ +) ? < >

特殊字符	
\n	换行
\r	回车
\t	选项卡

分组和范围	
.	除换行（\n）之外的任何字符
\r	回车
(a b)	a 或 b

分组和范围	
(...)	Group
(?:...)	被动（非捕获）组
[abc]	范围（a 或 b 或 c）
[^abc]	非（a 或 b 或 c）
[a-q]	小写字母 a 到 q
[A-Q]	大写字母 A 到 Q
[0-7]	数字 0 到 7

替换字符串	
\$n	第 n 个非被动组
\$2	/^(abc(xyz))\$/ 中的“xyz”
\$1	/^(?:abc)(xyz)\$/ 中的“xyz”
\$`	匹配的字符串之前
\$'	匹配的字符串之后
\$+	上一个匹配的字符串
\$&	整个匹配的字符串

附录 C. 快捷键

目录

可缩放的编辑器	817
Attenuation Editor 视图	817
音频文件管理	817
Contents Editor 视图	818
Game Object 3D Viewer 视图	818
Game Profiler	818
全局——上下文（活跃视图和选定对象）	818
Music Segment Editor 视图	819
Position Editor (3D Automation)	819
Project Explorer 视图	820
RTPC 坐标图视图	820
Schematic View 视图	820
Soundcaster 视图	820
Transport Control 视图	820

Wwise 中的许多命令或操作在键盘上都有对应的按键或按键组合。您可以使用这些键盘快捷键代替鼠标来执行其中的任意操作或命令。

大多数命令（见下文）可使用 Keyboard Shortcut Manager（键盘快捷键管理器）进行重新映射。

可缩放的编辑器

操作	使用此快捷方式
放大	Z + 选取框
垂直放大——以当前鼠标位置为中心	Ctrl + 向上滚动鼠标滚轮
垂直缩小——以当前鼠标位置为中心	Ctrl + 向下滚动鼠标滚轮
水平放大——以当前鼠标位置为中心	Ctrl + Shift + 向上滚动鼠标滚轮
水平缩小——以当前鼠标位置为中心	Ctrl + Shift + 向下滚动鼠标滚轮
重置平移和缩放（放大时）	Z + 点击
向上平移视图（放大时）	向上滚动鼠标滚轮
向下平移视图（放大时）	向下滚动鼠标滚轮
向左平移视图（放大时）	Shift + 向上滚动鼠标滚轮
向右平移视图（放大时）	Shift + 向下滚动鼠标滚轮
任意平移（放大时）	X + 拖动
重置平移（放大时）	X + 点击
插入点	双击
移动选定点	方向键
更准确地移动点	Shift + 拖动
选择非连续点	Ctrl + 点击
选择所有点	Ctrl+A
将选定项移到下一个点	选项卡
将选定项移到上一个点	Shift+Tab
将选定项锁定到 X 或 Y 轴	Alt + 拖动

Attenuation Editor 视图

请参阅 [可缩放的编辑器](#)。

音频文件管理



操作	使用此快捷方式
创建新的 Sound Voice	Shift + 拖动
在不打开 Audio File Importer 的情况下导入音效或音乐音频文件。	Ctrl + 拖动
在不打开 Audio File Importer 的情况下导入 Sound Voice 音频文件	Ctrl + Shift + 拖动
强制替换音效声或音乐源	Alt + 拖动

快捷键

操作	使用此快捷方式
强制替换 Sound Voice	Shift+Alt+drag

Contents Editor 视图

操作	使用此快捷方式
向上/向下移动选定项	向上/向下方向键
将对象添加到选定项（向上/向下）	Shift + 向上/向下箭头键
将选项定向右移动到下一个控制点	选项卡
将选定项向左移动到下一个控制点	Shift+Tab
将选定项向下移动到下一个控制点	Ctrl+Tab
将选定项向上移动到下一个控制点	Ctrl+Shift+Tab

Game Object 3D Viewer 视图

只有 Game Object 3D Viewer 活动时才可使用以下快捷键。

操作	使用此快捷方式
打开 Game Object 3D Viewer Setting 对话框	V
恢复摄像机的默认位置	Ctrl+R
设置摄像机在拍摄过程中跟踪选定的游戏对象。	Ctrl+F
将所有游戏对象和听者框在视图中。	Ctrl+A
显示 Game Object 3D Viewer Filter 对话框	Alt+F
向前导航（在第一人称或听者摄像机中）	W
向后导航（在第一人称或听者摄像机中）	S
向右导航（在第一人称或听者摄像机中）	A
向左导航（在第一人称或听者摄像机中）	D
加速查看器中的移动	Shift
切换到摄像机用户 1	Ctrl+1
切换到摄像机用户 2	Ctrl+2
切换到摄像机前视图	F
切换到摄像机顶视图	T
切换到第一人称视图	P

Game Profiler

操作	使用此快捷方式
打开 Profiler Settings 对话框	Alt+G
前往下一个蓝色圆圈	Ctrl + 向上箭头键
向下前往下一个蓝色圆圈	Ctrl + 向下箭头键
强制使用时间光标捕获日志条目	Shift + 点击

全局——上下文（活跃视图和选定对象）

操作	使用此快捷方式
对于属性值	

快捷键

操作	使用此快捷方式
恢复默认值	Ctrl + 点击
使用超级滑杆来细调值	Shift + 拖动
使用超级滑杆来偏置多个选中项的值。	Alt + 拖动

Music Segment Editor 视图

操作	按下
跳到段落的开头	Home
跳到段落的末尾	End
跳到第一轨	Shift+Home
跳到最后一轨	Shift+End
跳到左上	Ctrl+Home
跳到右下	Ctrl+End
将播放光标移到入口提示点 (Entry cue)	0
将播放光标移到出口提示点 (Exit cue)	1
将播放光标移到自定义提示点	2 到 8
将播放光标移到段落开头	. (点号)
锁定到小节/节拍	Q
锁定到提示点	W
锁定到片段/循环	E
在光标位置分割	S
放大	+
放大 (选定项)	Z + 选取框
缩小	-
平移	X + 鼠标拖动
在播放光标处插入提示点	Insert
设置入口提示点	Ctrl + 点击标尺
设置出口提示点	Alt + 点击标尺
设置新的自定义提示点	Shift + 点击标尺
移动入口/出口提示点到选定区域	P

另请参阅 [可缩放的编辑器](#)。

Position Editor (3D Automation)

Position Editor (3D Automation) 的快捷方式分为以下部分：

- 在 [Path List \(路径列表\)](#) 中

请参阅 [可缩放的编辑器](#) 了解路径编辑器和时间线视图中的快捷键。

在 Path List (路径列表) 中

操作	使用此快捷方式
向上和向下导航	上/下方向箭头

Project Explorer 视图

操作	使用此快捷方式
折叠子对象	数字小键盘中的斜线键
展开节点	数字小键盘中的向右键或加号键
折叠节点	数字小键盘中的向左键或减号键
向上导航	向上箭头
向下导航	向下箭头
展开子对象	数字小键盘中的星号键

RTPC 坐标图视图

请参阅 [可缩放的编辑器](#)。

Schematic View 视图

操作	使用此快捷方式
更改选定项，浏览工程，展开或折叠层级结构	方向箭头
转至兄弟对象组的第一个子对象	Home
转至兄弟对象组的最后一个子对象	End
前往 Master Audio	Ctrl+Home
在 Property Editor 中编辑选定项	Enter
播放	空格键
前往搜索字段	Ctrl+F
（从搜索字段）转至工程层级结构	Enter

Soundcaster 视图

操作	使用此快捷方式
Play - Bypass 属性	Shift + 点击 Play
全部暂停/全部恢复	Ctrl + Shift + 空格键
开关 Original (原始) 模式	Alt+O
Toggle Inc. Only	Alt+I
全部重置	Ctrl+Alt+R

Transport Control 视图

操作	使用此快捷方式
播放/停止	空格键
Play - Bypass 属性	Shift + 空格键 (在任何位置) Shift + 点击 Play (在 Transport 内)
暂停/恢复	Ctrl + 空格键
开关 Original (原始) 模式	Alt+O
Toggle Inc. Only	Alt+I
固定/取消固定	Ctrl+P

快捷键

操作	使用此快捷方式
加载并固定在 Transport Control 中	Ctrl+Alt+P
切换到 State	Alt+S
切换到 Switch	Alt+W
切换到 RTPC	Alt+R
切换到 Trigger	Alt+T
全部重置	Ctrl+Alt+R

词汇表

此词汇表旨在定义整个帮助文档中使用的技术术语。其中包括声音设计与编程词汇的通用术语，以及涵盖了 Wwise 特有（包括 Audiokinetic 专利）对象和概念的 Wwise 专用术语。

通用术语

AAC	Wwise 中可用于 Mac 和 iOS 平台上的一种感知编码音频压缩方法。据说在比特率相似的情况下，AAC 声音品质优于 MP3。压缩是可变的，取决于内容，品质设置可由“quality”（品质）滑杆控制。在 iOS 中，如果有可用的硬件辅助编解码器，则 AAC 会通过该硬件解码器解码。注意，iOS 硬件一次只可解码一个 AAC 声音。
AC-3	针对声音编码算法的一种压缩标准。有关详细信息，请参阅详述此 数字音频压缩标准 的文档。
ADPCM (Adaptive Differential Pulse Code Modulation)	自适应差分脉冲编码调制。一种音频文件编码方法，用于量化声音信号与根据该声音信号所做的预测之间的差异。ADPCM 量化步长是自适应的，与直接量化信号的 PCM 编码不同。基本上，ADPCM 以音质为代价，显著缩减了容量和 CPU 占用量。因此，它通常用于移动平台。
Ambisonics	Ambisonics 是一种环绕声技术，可以覆盖水平面声场以及听者上方和下方的区域。通过其 B-format 声场表示法，Ambisonics 能够独立于扬声器配置发挥效果。
ADSR (Attack-Decay-Sustain-Release)	包络的形状由该声音的起音时间（和 Wwise 中的起音曲线）、衰减时间、延音电平和释音时间来决定。
ATRAC9	一种感知编码方法，支持以各种比特率编码音频文件，同时保持非常好的感知声音品质。
位深	用于描述数字音频文件内每个采样点的位数。在 PCM 音频中，位深决定信号的最大可能动态范围。
Bit Rate (比特率)	每秒发送或接收的数据量（即位数）。比特率越高，处理的文件数据越多，通常分辨率也越高。
Cent (音分)	一种音高度量单位，相当于半音程的 1/100。八度音阶由 1200 个音分组成。
DC Offset (直流偏置)	直流表示声音波形的中心。偏置表示到波形中心 0.0 点的距离百分比。
dBFS (满量程分贝)	(分贝满量程) 数字系统中的分贝振幅电平，这些系统具有最高的可用电平（如 PCM 编码）。
Delay Line (延迟线)	Delay Line 处理器单元可以将信号延迟若干采样点后输出。

词汇表

Depot (文档库)	Perforce 版本控制服务器中的中央文件存储库。它包含提交到服务器的所有文件的所有版本。
Dithering (抖动)	在量化前添加到信号中的噪声，用于减少量化过程导致的失真和噪声调制。虽然噪声电平会略有上升，但频谱形状的抖动可让这种上升变得尽可能不明显。这种噪声没有失真的危害大，可以让低电平信号听得更清楚。
Downmix (下混)	将多声道源中的所有声道混合成具有较少声道的兼容版本（例如立体声或单声道）的过程。
干声路径	从声音直接到总线的声部信号基本路径。
Dry Signal (干声信号)	完全由未经处理的原始信号构成的输出。
Echo Density (回声密度)	混响算法每秒产生的回声量。
Frequency (频率)	单位时间对应的循环次数。
Gain (增益)	信号功率或振动的变化。
Harmonics (谐波)	基频的倍数。举例来说，如果基频是 50 Hz，则第二谐波是 100 Hz，第三谐波是 150 Hz，以此类推。
Headroom (裕量空间)	放大器或音频设备中的正常工作电平与削波电平之间的电平差（单位：dB）。
High-Pass Filter (高通滤波器)	一种递归滤波器，用于衰减低于截止频率的频率。此滤波器的单位代表已应用的高通滤波效果器的百分比，其中 0 表示高通滤波（信号不受影响），100 表示最大衰减。
HMD (Head Mounted Device)	头戴显示器，为虚拟现实 (VR) 开发的设备，旨在基于玩家的头部运动来提供视觉与音频输入。
高动态范围音频 (HDR 音频)	一种利用自然声音所跨越的大动态范围电平值来进行混音设计的技术。HDR 也是一个实时系统，可以将宽泛的电平范围动态地映射至更适合于您的声音系统数字输出的范围。
HRTF (头部相关传递函数)	声音通过不同物理介质从空间某点传递到耳朵的方式。成对的 HRTF 可合成双耳声，使声音好像源自于空间中的特定某点。
Inharmonicity (不和谐度)	局部谐波音高与真正谐波之间的偏置比例（基频的整数倍）。
Intensity Stereo (强度立体声)	一种失真音频编码技术，它去除高频内容的相位校准信息，以将它合并到单声道信号中，同时保留强度信息来重构立体声信号。
Latency (延迟)	在计算机内部处理或生成音频信号时所固有的延迟。
LFE	低频效果声道。专门针对 10-120 Hz 低沉声音设计的音频声道的名称。LFE 声道是完全独立的声道，必须以 x.1 媒体文件形式导入 Wwise。

词汇表

Limiting (限幅)	一种极端的压缩形式，其中输入/输出关系变得非常扁平（10:1 或更高）。它对信号电平做出硬性限制。
Low-Pass Filter (低通滤波器)	一种递归滤波器，用于衰减高于截止频率的频率。此滤波器的单位代表已经应用的低通滤波效果器的百分比，其中 0 表示低通滤波（信号不受影响），100 表示最大衰减。
Modal density (模态密度)	模态指音频信号在频域表示法中的峰值。当模拟大多数的声学空间时，增加模态密度可改善混响的逼真度。减低模态密度可能导致嗡嗡声。
Noise Shaping (噪声整形)	一种有助于将数字信号位深减少所造成的杂音最小化的技术。
Nyquist 频率	可准确采样的最高音频频率，相当于采样率的二分之一。Nyquist 采样定理表明，采样率必须至少是输入样本最高频率的两倍才能准确地重构原始信号。
Opus	一个低延迟的音频编解码器，针对语音和通用音频进行了优化，压缩时音质损失方面优于其他编解码器。此版本的 Opus 编解码器基于所发布的标准版本，并适用于所有平台。
OpusNX	一个低延迟的音频编解码器，针对语音和通用音频进行了优化，压缩时音质损失方面优于其他编解码器。此版本在 Nintendo Switch 硬件上解码，并与通用版 Opus 软件解码器兼容。
Outro (结尾部分)	声音的末尾部分，对应于开头部分。
Passband (通带)	基本无衰减地通过滤波器的频带。
PCM (脉冲编码调制)	一种音频文件编码方法，采用独特的二进制表示法即脉冲编码。通过测量两个编码点之间的值对它们进行量化；选择与最近点相关的值。
PCM 帧 (PCM frame)	PCM 帧包括在给定时刻上所有声道的采样。每帧代表 1/采样率秒。
Pitch	声部音高。对象的播放速度。
Prediction filter (预测滤波器)	根据语音信号行为的可预测性对语音信号进行建模的方法，可用作有损音频压缩技术。
Quantization (量化)	将音频文件的值域细分为子域的过程，各个子域由指定值表示。
Recursive Filter (递归滤波器)	一种滤波器，使用先前计算的输出值和当前输入值计算最近的输出。也称为反馈滤波器。
Ripple (in passband) ((通带内) 波纹)	通带内最大衰减和最小衰减之差。
RMS power (RMS 功率)	均方根 (Root Mean Square)。它是一种测量信号平均振幅的方法，在多数情况下，信号功率比使用峰值振幅时更接近。获得此值

	的方法是将给定时间窗口期间的采样值的平方加起来，然后计算结果的平方根。
Side-Chaining (旁链)	监视音频信号的电平，并使用音频信息实时操控另一个音频信号。此技术用于自动控制不重要声音的音量，从而在最终混音中更多地强调更加重要的声音。
Source Control (版本控制)	用于管理各种文件（如 Wwise 工程，工作单元和音频文件）的修改，其中每个文件的修订版本都标记了版本识别号、修改时间和修改人。大多数版本控制系统中都可以比较、还原和合并修订版本。
标准配置	标准离散声道配置，如立体声，4.0，5.1，7.1 和 7.1.4。Ambisonics 不属于标准配置。
Tab-delimited File (用制表符分割的文件)	一种特殊类型的纯文本文件，其中信息通过制表符分割成列。
Tap (延迟线分流)	信号从 Delay Line (延迟线) 中被提取出来的分流点。
VBAP (Vector Base Amplitude Panning)	矢量基幅值平移。一种常用算法，用于多个扬声器配置下，对若干不同方向的虚拟声源进行精准 3D 定位。
Volume	音量。音频输出的振幅或强度级别。
Vorbis	一种感知编码方法，支持以各种比特率编码音频文件，同时保持非常好的感知声音品质。通过使用 Quality Factor (品质因数) 或指定每声道的最大、最小和平均比特率来控制数据压缩效率和感知声音品质的平衡。
	 Note Audiokinetic 的 Vorbis 专用版本针对所有平台进行了高度优化。
WAVEFORMATEXTENSIBLE (具有两个以上声道的格式定义波形音频数据格式的结构)	输入时保留 WAV 文件的特定多声道配置，必须将 WAV 头文件中的声道信息定义为扩展型波形文件格式定义结构的声道掩码的一部分。
湿声路径	从基本干声路径中分支后通过 Auxiliary Bus 的声部信号路径，一般会先应用 Effect (如 Reverb) 和其他修改，然后再发送到总线。
Wet Signal (湿声信号)	完全由经过处理的声音构成的输出。
World Builder (世界编辑器)	用于创建游戏虚拟环境的应用程序。
XMA	针对 Xbox 360 的、硬件支持的感知编码音频压缩方法。XMA 是 Windows Media Audio Pro 的游戏机优化版本。压缩是可变的，取决于内容，可通过“Compression Quality”滑杆来控制品质设置。最新的版本 XMA 2.0 包含新的、用于创建寻址表 (seek table) 的块大小参数。寻址表可以方便查找 XMA 数据。

Wwise 专用术语

Absolute Property (绝对属性)	这些属性通常在顶层父对象上定义，它们会自动下传到各个父项的子对象，例如定位和播放优先级。您可以不沿用顶层父属性，而在层级结构的不同层级定义这些属性。
Actor-Mixer (角色混音器)	角色混音器。若干个声音、振动对象、容器或角色混音器所构成的层级结构。您可以使用角色混音器来控制它下面的所有对象的属性。
Additive Type Property (累积型属性)	此类 Wwise 属性可以通过如 RTPC 和 State 这样的不同设置进行多次更改，各种属性偏置值经过累积得到最终数值。
Audio Input (音频输入)	此处的 Audio Input 是一种源插件的示例，允许通过 Wwise 管线发送游戏生成的音频内容，并由声音引擎处理。
Attenuation (衰减)	当远离发声源时，声音、音乐或者振动对象的音量会相应地减小。
Attenuation ShareSet (衰减共享集)	基于音源与听者之间相对距离而进行的音量衰减设定。这些设定保存为共享集，可在工程间共享。
Audio Bus (音频总线)	一种可以在 Master Audio Bus 下添加的总线，允许和其他音频总线、辅助总线一起进行编组，以便有效地组织混音。您可重命名、移动和删除这些总线；还可以在总线上加效果器。
Audio Source (音频源)	音频源。音频文件和声音对象之间的独立抽象层。音频源链接到导入到工程中的音频文件。音频源保持链接到您导入工程中的音频文件，以便您可以在随时引用它。
Auto-ducking	自动闪避。此动作将降低一个音频信号的音量电平，以便让另一并行音频信号更加突出。
Auxiliary Bus (辅助总线)	一种特殊类型的音频总线，通常用于应用混响和延迟等效果器来模拟环境效果器或进行动态混音（旁链）。
Auxiliary Send (辅助发送)	用于将音频信号发送到辅助总线的音频信号通路技术。可根据声音对象来控制辅助发送，在使用 SDK API 时，也可根据游戏对象来控制辅助发送。
Blend Container (混合容器)	该组中的对象或容器将被同时播放。此容器中的对象可编组到 Blend Track (混合轨) 中，然后使用 RTPC 将属性映射到游戏参数。在同一条Blend Track上的各个对象之间也可以基于游戏参数值来进行交叉淡变。
Cache (缓存)	一个工程文件夹，其中包含您正在开发的平台的所有转码结果文件。在默认情况下，此文件夹存储在本地，但您可以修改它的位置。多个用户不得同时访问缓存文件夹。
Child Object (子对象)	层级结构中位于上层对象或父对象内的对象。
Clip	代表音频源的音乐对象。片段按音乐轨进行排列。

压缩器	一种音频效果插件，通过削弱输入信号高出预定义阈值的任何部分来缩小信号的动态范围。
Container (容器)	若干个对象的组合，包括根据其中既定行为播放的声音、振动对象或容器。Wwise 中有多种不同类型的 Container，包括 Random Container (随机容器)、Sequence Container (序列容器)、Switch Container (切换容器)、Blend Container (混合容器)、Music Switch Container (音乐切换容器) 以及 Music Playlist Container (音乐播放列表容器)。
Control Surface Session (控制器会话)	一种可保存的工程元素，用来定义 Wwise 如何与使用了 MIDI 或 Mackie 协议的外部控制器设备进行交互。连接了兼容设备并设置了绑定后，即可打开 Control Surface Session (控制器会话) 并使用该设备来控制 Wwise 的功能。
Conversion Settings (转码设置)	音频文件参数的组合，包括采样率、音频格式和声道数量，它们定义各个平台的音频文件的总体品质、内存和 CPU 占用。
Convolution Reverb (卷积混响)	一种音频效果器，它利用 IR (冲激响应) 模拟真实空间（例如音乐厅、建筑物、街道、汽车内饰、房间、户外、森林等）的声学特性。
Cue (提示点)	附加在音乐段落上的标记，用于指示关键点，例如进入点或退出点。
Custom Cue (自定义提示点)	附加到音乐段落上的用户创建的标记，用于指示关键同步点。Entry cue (入口提示点) 和 Exit cue (出口提示点) 不属于自定义提示点。
dB Scaling (分贝定标)	在表示以分贝为单位的属性时，用于以对数刻度显示坐标图视图 X 轴的选项。
Default Work Unit	XML 文件，其中包含工程中与它们所针对的特定元素相关的所有信息。例如，Event 的 Default Work Unit.wwu 文件包含与 Event 相关的所有信息，State 的 Default Work Unit.wwu 文件包含与 State 相关的所有信息，以此类推。在创建工程时将创建默认工作单元。
Definition File (定义文件)	按 SoundBank 分类列出游戏中所有事件的文本文件。
Delay	一种音频效果器插件，通过将音频信号延迟指定时间段来添加回声。
Dialogue Event (对白事件)	使用一组规则或条件来触发游戏音频的方法，这些规则或条件通过与游戏内的可能条件相匹配的切换开关或状态值来表达。这些切换开关或状态值被安排在路径中，然后被指派到 Wwise 对象。当游戏调用 Dialogue Event 时，游戏先将其当前情形与 Dialogue Event 中定义的情形进行匹配，然后播放相应的对白片段。
Diffraction	对 Wwise 来说，衍射是内建的 Game Parameter (游戏参数)，由 Spatial Audio 计算声音和 Shadow Boundary (阴影边界) 之间的偏

	差角度，大致模拟声波遇到转角或其他障碍物时，衍射声路径的偏移角度。
Effect Chain (效果器链)	以特定顺序应用到对象或总线的一系列效果器。
Effect Instance (效果器实例)	可保存并应用到其他对象或总线上的一组自定义效果器属性。效果器实例属性还可跨对象共享。
Effect ShareSet (效果共享集)	效果共享集。音频效果插件设定，可用于增强游戏中音频效果。这些设定保存为共享集，可在工程间共享。
Empty Event	不包含任何动作或对象的事件。
Environmental Effect (环境效果器)	根据游戏对象在游戏几何图形中的位置更改该对象所生成声音的属性集合的效果器。
Event (Action Event) (动作事件)	触发游戏中音频的方法，使用一个或一系列动作（如播放、静音和暂停）来控制若干个 Wwise 对象。
扩展器 (Expander)	Wwise Expander 插件通过削弱输入信号低于预定义阈值的任何部分来扩大信号的动态范围。当信号很弱并低于阈值时，扩展器开始降低信号的增益。当信号等于或高于阈值时，不对信号应用增益衰减。
External Source (外部源)	在运行时将声音对象与音频文件相关联的源插件。通过它可管理原本需要大量开销的大量对白行。它有助于节省运行时的内存，简化在为游戏生成 DLC 内容时有时候需要更换音频文件的过程。
File Manager (文件管理器)	显示有关工程文件和原始导入源文件的信息，以及在适用的情况下管理许多版本控制插件功能的对话框。
Flanger (镶边效果器)	将两个相同信号混合在一起的音频效果，其中一个信号以微小的渐变时间量延迟，以产生扫频式梳状滤波效果。
Flat View (扁平视图)	停靠在布局中的视图。
Focus	百分比值，用于收缩由扩散值生成的虚拟发声体。对于 0% 焦点，虚发声体保持不变，但值越高，各个虚拟点距离源声道原始位置越近。
Folder	Actor-Mixer Hierarchy 中的上层结构，用于管理角色混音器、容器等其它结构。
Game Object	界面、触发器或声音等元素可附加到其上的游戏实体。
Game Parameter (游戏参数)	使用 RTPC 可映射到 Wwise 属性值的游戏参数，例如赛车游戏中的速度和 RPM。
Game Sync (游戏同步器)	一组 Wwise 元素，包括 State、切换开关、RTPC、触发器和变量，根据游戏中的条件调用它们，并相应地修改音频和振动。
Game Unit (游戏计量单位)	用于计算游戏几何构造的基本长度单位。例如，秘密潜入游戏的 FPS 可使用节拍作为游戏计量单元，而太空征服游戏可使用光年。

Guitar Distortion (吉他失真效果器)	一种音频效果器，它更改音频波形的形状，引入原始信号中不存在的频率分量。Wwise 吉他失真效果器模拟常用失真“stomp boxes”的行为来获取典型的吉他失真声音。
Harmonizer (和声效果器)	将几个有音调的声部添加到输入信号的音频效果器。
Imported folder (导入文件夹)	隐藏的工程 .cache 文件夹，其中包含导入工程中的、经过特别导入转码过程的音频文件。
Image Source (声源虚像)	发声体相对于房间墙壁所成镜像的位置，在 Wwise Reflection 中根据 声源成像技术 计算得出。
标志	Wwise 界面中的特定图标，用于指示特定属性值的状态。例如，RTPC 标志显示属性值是否具有关联的 RTPC。
Initialization Bank (初始化库)	一种特殊类型的库，其中包含工程的所有常规信息（包括有关总线层级结构的信息）和有关状态、切换开关和 RTPC 的信息。每个工程只有一个初始化库，默认情况下它被命名为“Init.bnk”。初始化库必须是启动游戏时加载的第一个库。如果第一个加载的不是它，则后续的 SoundBank 可能会拒绝加载。
Input Range (输入范围)	可为属性输入的完整值域，与滑杆范围相反。
Integrity Report (完好度报告)	Wwise 中生成的一种报告，它显示工程中的错误或问题，并提供修复方案建议。
Interactive Music (互动音乐)	一种音乐创作和编曲方法，用于创建响应于游戏内动作的、模块化的配乐。
Invalid Events (无效事件)	已从工程中删除但 SoundBank 中仍包含的 Event。
IR (冲激响应)	测量某个位置（例如音乐厅）真实声学特性所生成的音频文件。卷积混响效果中使用冲激响应来启用对输入信号应用特定位置的声学特性。
Layout (布局)	为方便完成特定任务或工作而组合在一起的一系列视图。
Listener (听者)	游戏中的虚拟话筒或振动传感器，帮助将声音指定到特定扬声器或特定马达的振动来模拟 3D 环境。
Look-ahead time (预读时间)	在流播放中，它指为声音引擎查找流播放数据所预留的时间。
Master-Mixer Hierarchy (主混音器层级结构)	工程层级结构顶层的总线或一系列总线，您可以根据游戏中的主要类别来编组许多不同的声音、音乐和振动结构。例如，您可以将所有语音或音乐结构组合在一条音频总线下，所有声音效果器组合在另一条音频总线下，以此类推。
该总线是主音频总线	在 Master Mixer 层级结构中，Master Audio Bus 位于默认工作单元下的顶层。声音信号通过复杂的子总线结构之后，最终的音频输出由 Master Audio Bus 的设置及其应用的效果器决定。

词汇表

master secondary bus（主二路总线）	Master secondary bus 是指在 Master Mixer 层级结构的默认工作单元下的一种顶层总线，供二路输出（如游戏控制器）使用。
矩阵混响器	专为游戏制作而优化的独特混响效果，它可以平衡品质和性能，包括实时编辑和 RTPC 映射功能。
Meter Effect（电平表效果器）	在不修改信号的情况下测量信号电平的音频效果，另可将此电平作为游戏参数输出。该插件非常实用，可实现旁链，测量后的总线电平可通过 RTPC 驱动其它总线的音量。
Meter (Peak)（电平表（峰值））	显示各个声道的音频信号电平的一系列电平表。虽然音频和辅助总线显示输出信号，但动态效果（例如压缩器和限幅器）通常显示音频输入、音频输出和已应用的增益衰减。
Mixing Desk	一种灵活强大的调音控制台，它将各种总线和对象属性编组到单一视图中，用于实时优化游戏的音频混音。
Mixing Session（混音会话）	您选择的在调音台中使用的一组 Wwise 对象，可在任何时候保存和重新使用它们。
调制器包络	在 MIDI 或常规音频播放中，使用预定义的 ADSR（包络）调整波形的一种技巧。
调制器 LFO	在 MIDI 或常规音频播放中，使用 LFO（低频振荡器）调整波形或音频属性的一种技巧。
振动对象	Wwise 中您为工程创建的单个振动素材的一种表示。通常，这些对象控制着主机的游戏控制器的振动。每个振动对象可以包含数个声源，这些声源决定了游戏中实际生成什么样的振动。
Music Clip（音乐片段）	音乐轨的基本组件，显示为矩形区域，表示一个 WAV 文件。
Music Playlist Container（音乐播放列表容器）	随机或按顺序播放的若干个音乐对象或容器的组合。
Music Segment（音乐段落）	一种多声轨音乐对象，它是 Interactive Music 层级结构的基本单元。
Music Switch Container（音乐切换容器）	根据调用的 Switch 或 State 播放的若干个音乐对象或容器的组合。
Music Track（音乐轨）	可以包含多个独立音乐片段音乐对象，并以波形形式显示它们，使您能够在音乐片段中以视觉方式进行调整。
	Wwise 中有多种不同类型的 Music Track，包括 Random Music Track 、 Sequence Music Track 和 Switch Music Track （切换开关音乐轨）。
Nested Object（嵌套对象）	位于另一个对象内的对象。

词汇表

Nested Object (嵌套工作单元)	嵌套在另一个工作单元内的工作单元。它可以让工程文件的粒度更细，减少在合并版本控制文件时工作组环境中存在的潜在冲突。
Noise Gate (噪声门效果器)	在 Expander 效果插件中创建的一种效果器，它几乎可以完全消除输出信号中的声音。创建噪声门效果器的方式是，设置高扩展率（大于 10:1），阻拦增益衰减到此程度的声音。
Object (对象)	Wwise 中如声音、Motion、角色混音器和容器这样的元素，用于容纳、组合和定义工程层级结构中的声音、语音、振动和音乐。
声障 (Obstruction)	指游戏中对象（如柱子等）部分挡住声音对象与听者之间空间的一种状况。
Occlusion (声笼)	指游戏中对象（如墙壁等）完全挡住声音对象与听者之间空间的一种状况。
Originals (原始音频) 文件夹	此文件夹中包含导入到工程中的音频文件未经改动的副本。此文件夹通常要做版本控制。
Orphan file (落单文件)	不再与声音 Motion 或音乐对象相关联的音频文件。在您删除声音对象时不会自动删除这些文件。要删除这些文件，则要清空音频 .cache 文件夹。
Output buffer latency (输出缓冲延迟)	音频播放期间引入的延迟，由 Wwise 使用的输出缓冲去数量决定。
Parametric EQ (参数均衡器)	一种音频效果器插件，通过它可以应用各种滤波器来对音频频谱塑形。
Parent Object (父对象)	层级结构中的对象，其中包含子对象。
Peak Limiter (峰值限幅器)	控制音频信号动态范围的音频效果器。实现方法是削弱对音频信号中暂时超出预定义信号峰值阈值的部分。
Physical Folder (实文件夹)	硬盘上位于 Wwise 工程根下的目录，可包含工程中使用的其他实文件夹或工作单元。实文件夹不能作为容器、Motion 或声音对象的子对象。
Physical Voice (实声部)	声音引擎播放和处理音频和振动所在的游戏物理环境。当音量电平变得极低时，声音和振动对象可转移到虚拟环境中，由声音引擎进行管理和监视，但不执行音频处理。
Pitch Shifter (移调器)	一种音频效果，可更改音高，但不影响所获音频信号的时长。
Point Source (点声源)	如同从单点发出声音或振动的声源。
Post-exit (后尾段)	出口提示点 (exit cue) 后的段落区域，可用于互动音乐中的过渡。
Pre-entry (前导段)	入口提示点 (entry cue) 前的段落区段，可用于互动音乐中的过渡。
Prefetch Time (预取时间)	在流播放中，它指一种小型缓冲区，照顾预取剩余文件数据所需要的延迟时间。

Preset (预设)	对象、效果和声音传播的一组自定义属性，可保存起来供随时复用。
Query (查询)	包含一组特定的搜索条件，用于查找特定对象或工程元素。
Random Container (随机容器)	按随机顺序播放的若干个声音、振动对象或容器的组合。
Randomizer (随机化器)	Wwise 中作用于属性值的一种特殊效果器，通过它您可以定义每次播放对象时可随机使用的可能值域。
Random Music Track	每次播放其父段落时，将按随机顺序播放其子音乐轨。
Relative Property (相对属性)	可在工程层级结构的各个层级上定义的属性，例如音量和音高。这些属性值是累加的，即父项的属性值将累加到子项的属性值上。
Reverb	一种音频效果器插件，用于模拟特定房间或空间的声学特性。
Room Coupling (房间配对)	Wwise Spatial Audio 中的概念，描述声音如何受到其所在房间的声学特性影响，并通过门户（房间开口）和墙壁（经过不同程度的阻塞和遮挡）以漫射场形式传播到相邻房间。
RoomVerb 混响器	一种多功能高品质混响效果器插件，用于模拟特定房间或空间的声学特性。
RTPC (实时参数控制)	实时参数控制 (Real Time Parameter Controls) 的缩写。一种互动方法，用于通过将游戏参数值映射到 Wwise 中的属性来驱动游戏中的音频。RTPC 还可用于通过将游戏参数映射到切换开关组来驱动切换开关的切换。
采样率	在将模拟信号转换成数字信号或数字转码期间，每秒采样音频信号的频率。
Send Volume	发送到辅助总线的音频信号的电平或振幅。
Sequence Container (序列容器)	根据特定播放列表播放的若干个声音、振动对象或容器的组合。
Sequence Music Track	每次播放其父级段落时，都将按照顺序播放其子音轨。
Shadow boundary (阴影边界)	View Region (照明区域，即声波路径未发生改变的区域) 和 Shadow Region (阴影区域，即声波由于衍射而发生偏折的区域) 的分界线。
ShareSet (共享集)	可在对象之间共享、用于定义效果器或衰减等属性的一组属性。
ShortID	短 ID。在 Wwise 工程内添加对象时，会在相关 Work Unit 文件中创建相应条目，并为其赋予一个 128 位的全局唯一标识符 (GUID) 编号。ShortID 可以是对象 GUID 的 FNV 哈希值，也可以是对象名称的 FNV 哈希值，即生成 32 位标识符。ShortID 的创建取决于对象的类型：

- 对于 Event、Game Sync 和 SoundBank，哈希值为隐式值，不会在 Work Unit 文件 (XML) 中声明。它是 Wwise 工程中对象名称的 32 位 FNV 哈希值。
- 对于所有其他对象 (WorkUnit、Sound、Container、Bus 等)，ShortID 为 GUID 字节的 32 位 FNV 哈希值。

您可以在 SDK 中查看 FNV 哈希算法的 C++ 实现方式: \SDK\include\AK\Tools\Common\AkFNVHash.h

Silence Source (空白源)	一种插件音源，可以指定时长，但在此期间内不会产生声音和振动。
Slider Range (滑杆范围)	使用滑块设置属性时的默认值范围，如果在字段中手动输入此范围之外的值，将会改变滑块范围。
SoundBank (音频包)	事件数据、声音、音乐、振动结构数据或媒体的编组，可在游戏中特定时刻一起加载至游戏平台内存中。
Soundcaster	Wwise 中的一种视图，可以在其中根据需要插入和移除对象或事件，提供了播放控制、试听和在 Wwise 或游戏中的实时混音等功能。
Soundcaster Session (声音选角器会话)	可被保存的工程元素，包含使用 Soundcaster 创建的特定模拟中所用的 Wwise 对象和事件。
SoundFrame (声音构架)	一种独特的插件接口，使外部应用程序能够无缝地与 Wwise 通信。可用 SoundFrame 创建的一些工具的例子有 SFTest 和 CarSim。
声音对象	Wwise 工程中您创建的单个音频素材的一种表示。每个声音对象都包含若干声源，即游戏中播放的实际音频内容。请注意，Wwise 文档中使用的大写“Sound”是指音频对象，可以是 Sound SFX 或 Sound Voice。
Sound SFX (音效声)	Actor-Mixer Hierarchy 中包含音效、音乐和环境声的声音对象。
Sound Voice (语音声)	包含对话或游戏语音的声音对象。
Source Plug-in (源插件)	由 Wwise 外部的插件创建的音频或振动源。
Spatialization (空间化)	Wwise 中用于确定声音或音乐对象在游戏 3D 环境中的实际位置的一种功能。
Spread	扩散到附近扬声器的音频量或百分比，以使声音能够随着距离的增加从低扩散的点声源变为完全扩散的传播源。对于多声道声音，各个声道单独扩散。
State (状态)	根据游戏中的物理和环境条件的变化，对游戏音频属性进行全局偏置或调整。
State Group	将相关状态进行分组，用来管理游戏环境中的全局更改。

词汇表

Stereo Delay (立体声延迟器)	一种音频效果器，与内置滤波器一起提供双声道延迟。它有反馈和交叉回馈控件来控制延迟信号从一个声道发送到另一个声道，创建立体声效果。
Stinger (插播乐句)	与当前正在播放音乐叠加并混音的简短乐句。
Switch (切换开关)	Switch 代表游戏中特定元素的替代项，用于帮助管理这些替代项的相对对象。举例而言，如果角色在混凝土表面上奔跑，然后进入草地，则切换容器中的脚步声应随之改变，以匹配地面的变化。
Switch Container (切换容器)	该容器应用了一系列 Switch 或 State，每一个都包含对应于游戏环境特定变化的一组声音、振动对象或容器。例如，角色脚步声的 Switch Container 可包含角色在游戏中可能行走的草地、混凝土、木材等表面所对应的 Switch。
Switch Group (切换开关组)	将相关的切换开关进行分组，用来管理游戏内指定元素在不同条件下的替代选项。
Switch Music Track (切换开关音乐轨)	根据相关联的切换开关组播放其子音乐轨。
Time Stretch	一种音频效果器，可改变时长但不影响所获音频信号的音高。
过渡	在互动音乐中，使用 Transition 的目的是在源音乐段落和目标音乐段落之间实现无缝过渡。
Transition Time (过渡时间)	同一状态组中从一个状态过渡到另一个状态所使用的时间。在过渡期间，将发生两个状态属性的插值。
Transmission (传输)	Wwise Spatial Audio 中的一个概念，指的是声能通过障碍物后剩下的比例。
Tremolo (震音)	使用无极载波信号调制输入信号振幅的音频效果。
Trigger (触发器)	一种 Game Sync，响应游戏中的突发事件并播放 Stinger (插播乐句)。
Virtual Folder (虚拟文件夹)	起组织作用的对象，显示为一个文件夹，位于工作单元或其子对象中。您可以在虚拟文件夹中放置其他对象，如虚拟文件夹、角色混音器、容器、振动对象或声音。虚拟文件夹不能作为容器、振动或声音的子对象，且在硬盘上没有对应的目录。
Virtual Voice (虚声部)	一种虚拟环境，在此声音和振动由声音引擎管理和监视但不执行处理。当对象的音量电平低于音量阈值时，它们将进入虚声部。
Voice (声部)	单独或离散的音频或振动播放实例。
Voice Starvation (声部匮乏)	当声音引擎无法及时向平台硬件缓冲区提供音频数据时，Capture Log 中显示的一种错误消息类型。这种问题发生在 CPU 使用过量的场合。例如，平台 CPU 尝试混合过多声源或者同时使用过多音频效果器。

词汇表

Volume Threshold (音量阈值)	一个特定音量电平，低于此电平，可以明确地确定声音、音乐和振动对象的行为。例如，低于音量阈值的声部可以继续播放、终止或者将其发送到虚声部列表。这些行为可以在对象的 Property Editor 的 Advanced Settings 选项卡中来定义。
Watch (监视)	可用于监视游戏中的游戏对象和听者的单一进程。
Workgroup (工作组)	同时处理同一 Wwise 工程的一组人员。
Work Unit (工作单元)	一个独特的 XML 文件，其中包含与工程中特定部分或元素相关的信息。