Dolby® Game Audio Unity Plug-in Interface Specification

Issue 1

Confidential Information

Dolby Laboratories Licensing Corporation

Corporate Headquarters

Dolby Laboratories, Inc.
Dolby Laboratories Licensing Corporation
100 Potrero Avenue
San Francisco, CA 94103-4813 USA
Telephone 415-558-0200
Fax 415-863-1373
www.dolby.com

European Licensing Liaison Office

Dolby International AB
Apollo Building, 3E
Herikerbergweg 1-35
1101 CN Amsterdam Zuidoost
The Netherlands
Telephone 31-20-651-1800
Fax 31-20-651-1801

Asia

Dolby Japan K.K.
NBF Higashi-Ginza Square 3F
13–14 Tsukiji 1-Chome, Chuo-ku
Tokyo 104-0045 Japan
Telephone 81-3-3524-7300
Fax 81-3-3524-7389
www.dolby.co.jp

Dolby Laboratories Hong Kong Limited
Unit 5407, Central Plaza
18 Harbour Road
Wanchai, Hong Kong
Telephone 852-2519-0888
Fax 852-2519-8988

Dolby Laboratories International Services (Shanghai) Co., Ltd.
05–07a, Floor 18
The Center
989 Chang Le Road
Shanghai 200031 China
Telephone 86-21-6113-3456
Fax 86-21-6113-3400
www.dolby.com.cn

Issue 1
S13/27237

The Dolby Game Audio Unity Plug-in

Dolby® audio processing technology improves audio quality, which leads to a more enjoyable listening experience for movies, music, and video games. This technology has been integrated into a number of popular portable devices, such as the Amazon™ Kindle™ Fire™ HD. To make Dolby audio processing technology available for mobile game use, Dolby has developed a plug-in for the Unity engine.

The Dolby Game Audio Unity plug-in is delivered to game developers in the form of a library that consists of two files: libDSPlugin.so and DSJavaPlugin.jar. Additionally, certain interfaces are exposed to make Dolby audio processing available to game applications. You can invoke the interface functions by using normal C# script and JavaScript function calls.

1 Overview

The Dolby Game Audio Unity plug-in is developed for the Unity game engine, with a Java interface communicating with the Dolby audio processing component integrated in a Google™ Android™ device, and with C interfaces exposed to game developers for calling associated functions. An overview of the plug-in structure is shown in the following figure.

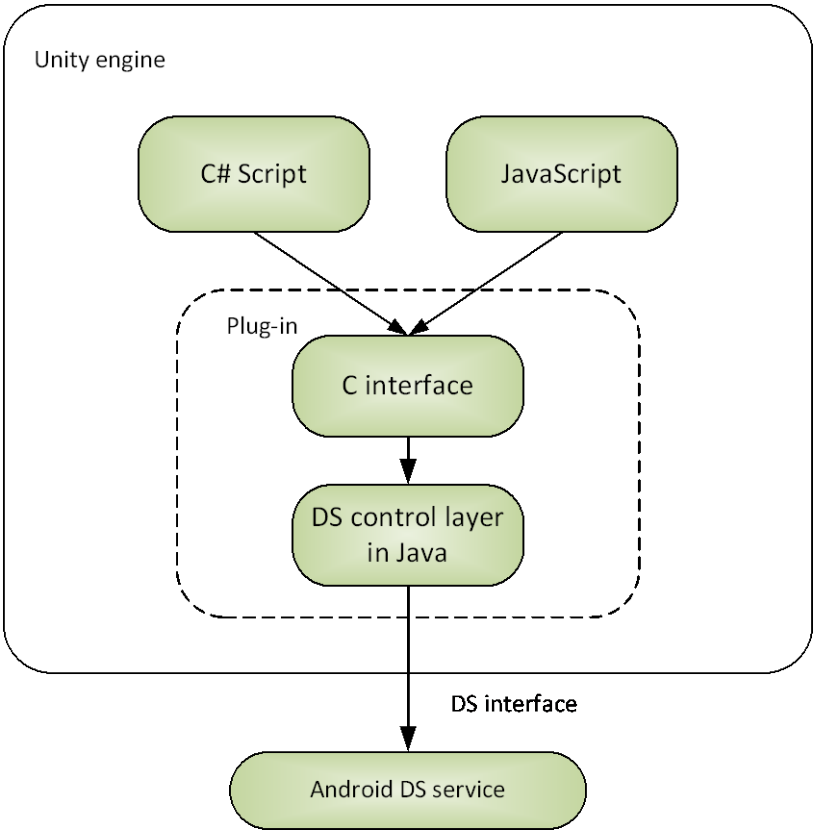

Figure 1 Dolby Audio Processing Unity Plug-in Structure

To implement the game-developer interface, use the information at http://docs.unity3d.com/Documentation/Manual/PluginsForAndroid.html.

2 Dolby Audio Processing Application Programming Interface Description

These sections describe the Dolby audio processing application programming interface (API) functions.

2.1 isAvailable()

Enables the application to check whether Dolby audio processing has been integrated into a device. If Dolby audio processing is integrated in the device, this function returns a value of true; otherwise, it returns a value of false. You can then use the returned value in

other code components. For example, the application could determine whether to display the Dolby logo based on this return value.

Syntax

```
const bool isAvailable()
```

Parameters

None

Preconditions

None

Return Values

True	Dolby audio processing is integrated in the device.
False	Dolby audio processing is not integrated in the device.

Error Codes

None

Notes

`isAvailable()` must be called before any other Dolby functions.

2.2 Initialize()

Enables an application to initialize Dolby audio processing.

Syntax

```
const int initialize()
```

Parameters

None

Preconditions

The application must call `isAvailable()` to determine whether Dolby audio processing is present on the device before calling `initialize()`.

Return Values

`0`	The function succeeded.
`-1`	The function failed.

Error Codes

None

Notes

None

2.3 setProfile()

Enables a corresponding Dolby audio processing profile of a given profile index.

Syntax

```
const int setProfile(
    int profileid
)
```

Parameters

`profileid`	[in] Profile ID of a given index. The index starts from 0, and the maximum is the profiles count minus 1.

Preconditions

None

Return Values

Return Value	Description
`0`	The function succeeded.
`-1`	The function failed.

Notes

If the input profile index is out of range, the return value is –1.

Following is the mapping table of profile IDs and corresponding profile names.

Profile ID	Profile Name
`0`	Movie
1	Music
`2`	Game
`3`	Voice

Error Codes

None

2.4 suspendSession()

Enables a device to pause and save the current Dolby audio processing game configuration when a game is switched to the background, but not exited, and enables it to restore the default configuration that was used before the game was started or resumed.

Syntax

```
const int suspendSession()
```

Parameters

None

Preconditions

None

Return Values

Return Value	Description
`0`	Pausing Dolby audio processing succeeded.
`-1`	Pausing Dolby audio processing failed.

Error Codes

None

Notes

The application must call this function before the game is switched to the background in order to save the Dolby game configuration and restore the default Dolby configuration that was in use before the game was launched or resumed. The user may switch the game to the background, or a program with a higher priority may cause the game to be switched. The current game configuration can be restored by calling the `restartSession` function.

2.5 restartSession()

Enables the device to restore the Dolby game configuration saved by `suspendSession`.

Parameters

None

Preconditions

None

Return Values

Return Value	Description
`0`	Restoration of the Dolby game configuration succeeded.
`-1`	Restoration of the Dolby game configuration failed.

Error Codes

None

Notes

The application must call this function when the user resumes the game, or when the application that caused the game to be sent to the background has exited.

2.6 release()

Enables a device to reclaim the memory that Dolby audio processing uses when a game is active and restores the configuration that was in use before the game was launched.

Syntax

```
void release()
```

Parameters

None

Preconditions

None

Return Values

None

Error Codes

None

Notes

The application must call `release()` when the game exits.