

Music Export Library Reference

© 2012 Sony Computer Entertainment Inc.
All Rights Reserved.
SCE Confidential

Table of Contents

Structures..... **3**
 SceMusicExportParam4

Functions..... **5**
 sceMusicExportFromFile6

Callbacks..... **8**
 SceMusicExportCancelFunc.....9
 SceMusicExportProgressInfoFunc.....10

Constants..... **11**
 Character String Size12
 Work Memory Size13
 Return Codes14

000004892117

Structures

000004892117

SCE CONFIDENTIAL

SceMusicExportParam

Export parameter structure

Definition

```
#include <music_export.h>
typedef struct SceMusicExportParam {
    void reserved[128];
} SceMusicExportParam;
```

Members

reserved Reserved area (fill with all 0s)

Description

This is a structure of the parameters to be specified to `sceMusicExportFromFile()`.

reserved is a reserved area for future function expansion. It must be filled with all 0s.

See Also

`sceMusicExportFromFile()`

Functions

000004892117

SCE CONFIDENTIAL

sceMusicExportFromFile

Execute export processing through file path specification

Definition

```
#include <music_export.h>
SceInt32 sceMusicExportFromFile (
    const SceChar8 *musicfilePath,
    const SceMusicExportParam *param,
    void *workMemory,
    SceMusicExportCancelFunc cancelFunc,
    SceMusicExportProgressInfoFunc progressFunc,
    void *userdata,
    SceChar8 *exportedPath,
    SceInt32 exportedPathLength
);
```

Arguments

<i>musicfilePath</i>	Path of the music file
<i>param</i>	Pointer to SceMusicExportParam
<i>workMemory</i>	Work memory
<i>cancelFunc</i>	Pointer to cancel callback function
<i>progressFunc</i>	Pointer to the progress callback function
<i>userdata</i>	Pointer to the user data serving as an argument of the cancel callback function and the progress callback function
<i>exportedPath</i>	Buffer storing the export destination path of the music file
<i>exportedPathLength</i>	Size of the buffer storing the export destination path of the music file

Return Values

Returns SCE_OK (0) for normal termination.

Returns an error code (negative value) described in the Return Codes section for errors.

SCE_MUSIC_EXPORT_ERROR_SERVER_DOWN returns when launch of the system process executing export processing fails, or when registration processing is interrupted. For this reason, perform a retry if the SCE_MUSIC_EXPORT_ERROR_SERVER_DOWN error returns.

Description

This is a function to execute the music export function.

Export processing is executed by specifying the file path of the music file.

For *musicfilePath*, specify the path of the music file to export.

param is a reserved area for future function expansion. *param.reserved* must be filled with all 0s.

For *workMemory*, specify the work memory used by this function. Specify a memory space of SCE_MUSIC_EXPORT_MAX_MEMBLOCK_SIZE or more bytes as the work memory. If NULL is specified, memory of SCE_MUSIC_EXPORT_MAX_MEMBLOCK_SIZE bytes is acquired internally by the function.

For *cancelFunc*, specify the cancel callback function. If the cancel callback function is not required, specify NULL.

For *progressFunc*, specify the progress callback function. If the progress callback function is not required, specify NULL.

SCE CONFIDENTIAL

For *userdata*, specify the pointer to the data to be used as the argument of the cancel callback function and the progress callback function. If no argument is required for the cancel callback function and the progress callback function, specify NULL.

exportedPath, specify the buffer storing the export destination path of the music file. For *exportedPathLength*, specify the buffer size. The maximum size of the path is SCE_MUSIC_EXPORT_MAX_FS_PATH. Prepare a buffer that is of the SCE_MUSIC_EXPORT_MAX_FS_PATH size or more. When the path is longer than *exportedPathLength*, it will be cut off at SCE_MUSIC_EXPORT_MAX_FS_PATH. Note the end of the path string should be NULL.

This function is a blocking function that returns upon completion of the export processing. Therefore, in order to execute cancellation of the export processing, this function must be called from a sub-thread.

Examples

```
#include <libsysmodule.h>
#include <music_export.h>

returnCode = sceSysmoduleLoadModule(SCE_SYSMODULE_MUSIC_EXPORT);
if (returnCode != SCE_OK ) {
    return // Error handling
}

SceChar8 exportedPath[SCE_MUSIC_EXPORT_MAX_FS_PATH];
SceChar8 *srcPath = "test.mp3";
SceMusicExportParam param;
sceClibMemset (&param, 0, sizeof(param));

returnCode = sceMusicExportFromFile(srcPath, &param,
                                   NULL, NULL,
                                   exportedPath, SCE_MUSIC_EXPORT_MAX_FS_PATH);

if (returnCode != SCE_OK) {
    // error
}

sceSysmoduleUnloadModule(SCE_SYSMODULE_MUSIC_EXPORT);
```

See Also

SceMusicExportParam, SceMusicExportCancelFunc

SCE CONFIDENTIAL

Callbacks

000004892117

SCE CONFIDENTIAL

SceMusicExportCancelFunc

Callback function for canceling export

Definition

```
#include <music_export.h>
typedef SceBool (*SceMusicExportCancelFunc) (
    void *userdata
);
```

Arguments

userdata Pointer to user data

Return Values

Return SCE_TRUE to cancel export.

Return SCE_FALSE to continue export.

Description

This is the cancel callback function specified for `sceMusicExportFromFile()`.
Export processing is canceled by returning SCE_TRUE.

See Also

`sceMusicExportFromFile()`

SCE CONFIDENTIAL

SceMusicExportProgressInfoFunc

Callback function to notify progress of export

Definition

```
#include <music_export.h>
typedef void (*SceMusicExportProgressInfoFunc) (
    void *userdata,
    SceUInt32 progress
);
```

Arguments

userdata Pointer to user data
progress Progress of the export will be specified as a value within the range of 0 - 100

Return Values

None

Description

This progress callback function is specified to `sceMusicExportFromFile()`.
It obtains the progress of the export processing.

See Also

`sceMusicExportFromFile()`

Constants

000004892117

SCE CONFIDENTIAL

Character String Size

Maximum size of the music file path

Definition

Macro	Value	Description
SCE_MUSIC_EXPORT_MAX_FS_PATH	1024	Upper limit for the size of the music file path

Description

This is the maximum size of the path for the music file to export using the music export library. The values are the size including NULL termination.

Specify all the character strings with UTF-8.

SCE CONFIDENTIAL

Work Memory Size

Size of the work memory used by the music export library

Definition

Macro	Value	Description
SCE_MUSIC_EXPORT_MAX_MEMBLOCK_SIZE	65536	Size of work memory used by the music export library

Description

This is the size of the work memory used by the music export library. Allocate a work memory of the above-noted size or larger.

Return Codes

List of return codes returned by the music export library

Definition

Macro	Hexadecimal	Description
SCE_MUSIC_EXPORT_ERROR_PARAM	0x80105301	Invalid parameter
SCE_MUSIC_EXPORT_ERROR_FILE_NOT_FOUND	0x80105302	File does not exist
SCE_MUSIC_EXPORT_ERROR_CONTENT_FULL	0x80105303	Maximum number of contents has been exceeded
SCE_MUSIC_EXPORT_ERROR_NO_MEMORY	0x80105304	Out of memory
SCE_MUSIC_EXPORT_ERROR_SERVER_DOWN	0x80105305	The server is not running
SCE_MUSIC_EXPORT_ERROR_TOO_MANY_CLIENT	0x80105306	The maximum number of connections that can be made to the server has been exceeded
SCE_MUSIC_EXPORT_ERROR_MEDIA_FULL	0x80105307	Out of storage capacity
SCE_MUSIC_EXPORT_ERROR_CREATE_FILE	0x80105308	Temporary file creation error
SCE_MUSIC_EXPORT_ERROR_NOT_SUPPORTED_FORMAT	0x80105309	Invalid file format
SCE_MUSIC_EXPORT_ERROR_CANCELED	0x8010530a	User performed cancel operation
SCE_MUSIC_EXPORT_ERROR_INTERNAL	0x8010530b	Other internal error
SCE_MUSIC_EXPORT_ERROR_MEDIA_NOT_EXIST	0x8010530c	Memory card does not exist
SCE_MUSIC_EXPORT_ERROR_DB_CORRUPTED	0x8010530d	DB corruption error
SCE_MUSIC_EXPORT_ERROR_INVALID_PATH	0x8010530e	Invalid path name

Description

The functions offered by the music export library return `SCE_OK(0)` upon normal termination, and the above error codes (negative value) upon abnormal termination.

When displaying errors returned by the music export library, make sure to use the error code display function of Message Dialog. For Message Dialog, refer to the "Message Dialog Overview" and "Message Dialog Reference" documents.