InvitationDialog Library Reference

PlayStation®Vita Programmer Tool Runtime Library Release 3.50

Document serial number: 000004892117

Document serial number: 000004892117

Table of Contents

Session ID/Invitation ID

SceNpInvitationId

Invitation ID

Definition

```
#include <np_common.h>
#define SCE_NP_INVITATION_ID_MAX_SIZE  (60)
typedef struct SceNpInvitationId {
        char data[ SCE_NP_INVITATION_ID_MAX_SIZE ];
        char term;
        char padding[3];
} SceNpInvitationId;
```

Members

data	Invitation ID string (maximum 60 bytes)
term	Null terminator
padding	Reserved area

Description

This structure represents the invitation ID.

For the invitation content, refer to the "Session/Invitation Web APIs Reference" document.

SceNpSessionId

Session ID

Definition

```
#include <np_common.h>
#define SCE_NP_SESSION_ID_MAX_SIZE  (45)
typedef struct SceNpSessionId {
        char data[ SCE_NP_SESSION_ID_MAX_SIZE ];
        char term;
        char padding[2];
} SceNpSessionId;
```

Members

data	Session ID (maximum 45 bytes)
term	Null terminator
padding	Reserved area

Description

This structure represents the session ID.

For the session content, refer to the "Session/Invitation Web APIs Reference" document.

See Also

sceInvitationDialogParamInit(), sceInvitationDialogInit(), SceInvitationDialogDataParam

Parameter Settings

SceInvitationDialogAddressInfo

Invitation dialog addressee details

Definition

```
#include <invitation_dialog.h>
#define SCE_INVITATION_DIALOG_ADDRESS_USER_LIST_MAX_NUMBER (16)
typedef union {
        struct {
              const SceNpOnlineId *onlineIds;
              SceUInt32 onlineIdsCount;
        } UserSelectDisableAddress;
        struct {
              SceUInt32 onlineIdsMaxCount;
        } UserSelectEnableAddress;
} SceInvitationDialogAddressInfo;
```

Members

onlineIds	Array of target users
onlineIdsCount	Number of target users (16 or less)
onlineIdsMaxCount	Maximum number of target users (16 or less)

Description

This is a union for specifying detailed information of addressees when sending an invitation with the invitation dialog.

To specify SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERDISABLE as the editing mode for the invitation addressee list and disable user edits, set the members *onlineIds* and *onlineIdsCount* in the *UserSelectDisableAddress* structure.

For *onlineIds*, specify the address for the target user array allocated in advance. Memory area allocated for *onlineIds* must be held until the invitation dialog is terminated with sceInvitationDialogTerm().

For *onlineIdsCount*, specify the number of addressees specified in *onlineIds*.

To specify SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERENABLE as the editing mode for the invitation addressee list and enable user edits, set the *onlineIdsMaxCount* member of the *UserSelectEnableAddress* structure.

For *onlineIdsMaxCount*, specify the maximum number of addressees that users can arbitrarily specify.

In either editing modes for the invitation addressee list, the maximum number of users that can be specified as addressees is SCE_INVITATION_DIALOG_ADDRESS_USER_LIST_MAX_NUMBER.

See Also

sceInvitationDialogParamInit(), sceInvitationDialogInit(), SceInvitationDialogAddressParam

SceInvitationDialogAddressParam

Invitation dialog addressee parameters

Definition

```
#include <invitation_dialog.h>
typedef struct SceInvitationDialogAddressParam {
        SceInvitationDialogAddressType addressType;
        SceInvitationDialogAddressInfo addressInfo;
} SceInvitationDialogAddressParam;
```

Members

addressType Addressee edit mode
addressInfo Addressee details

Description

This structure is for specifying the addressee parameters when calling `sceInvitationDialogInit()` in send mode.

For *addressType*, set the edit mode of the invitation addressee list. Specify one of the following values.

Value	(Number)	Description
SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERDISABLE	1	Disable addressee list editing
SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERENABLE	2	Enable addressee list editing

If SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERDISABLE is specified for *addressType*, the user cannot edit the addressee list set by the application.

When SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERENABLE is specified for *addressType*, the user can edit the addressee list when invitation dialog is displayed.

For *addressInfo*, specify the detailed information of the addressees. For details, refer to the "SceInvitationDialogAddressInfo" section.

See Also

SceInvitationDialogDataParam, SceInvitationDialogAddressType, SceInvitationDialogAddressInfo

SceInvitationDialogAddressType

Invitation addressee list editing mode

Definition

```
#include <invitation_dialog.h>
typedef SceInt32 SceInvitationDialogAddressType;
```

Description

These constants represent the edit mode of the invitation addressee list.

Value	(Number)	Description
SCE_INVITATION_DIALOG_ADDRESS_TYPE_INVALID	0	Invalid mode/initial value
SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERDISABLE	1	Addressee list editing disabled
SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERENABLE	2	Addressee list editing enabled

When SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERDISABLE is specified for the editing mode, the user cannot edit the addressee list set by the application.

When SCE_INVITATION_DIALOG_ADDRESS_TYPE_USERENABLE is specified for the edit mode, the user can edit the addressee list when the invitation dialog is displayed.

See Also

SceInvitationDialogAddressParam

SceInvitationDialogDataParam

Invitation send/receive data parameters

Definition

```
#include <invitation_dialog.h>
#define SCE_INVITATION_DIALOG_MAX_USER_MESSAGE_LENGTH (512)
typedef union {
        struct {
              const SceChar8 *userMessage;
              const SceNpSessionId *sessionId;
              SceInvitationDialogAddressParam addressParam;
              SceInvitationDialogOptions options;
        } SendInfo;
        struct {
              SceInvitationDialogOptions options;
              uint8_t reserved[60];
        } RecvInfo;
} SceInvitationDialogDataParam;
```

Members

userMessage	Application-specified message string (UTF-8, NULL-terminated, 512 characters or less) or NULL
sessionId	Session ID obtained from the server
addressParam	Addressee parameters
options	Options
reserved	Reserved area (fill with 0's)

Description

This data parameter union is required for sending or receiving invitations using the invitation dialog.

An arbitrary message string can be specified for *userMessage*. The string must be UTF-8, NULL-terminated, with a maximum size of SCE_INVITATION_DIALOG_MAX_USER_MESSAGE_LENGTH (characters). Specify NULL when not setting a string. The memory area allocated for *userMessage* must be held until the invitation dialog is terminated with sceInvitationDialogTerm().

For *sessionId*, specify the session ID obtained from the server. The memory area allocated for *sessionId* must be held until the invitation dialog is terminated with sceInvitationDialogTerm().

For *addressParam*, specify the addressee parameters that indicate the addressee of the invitation. For details, refer to the "SceInvitationDialogAddressParam" section.

For *addressParam*, specify the addressee parameters that indicate the addressees of the invitation. For details, refer to the "SceInvitationDialogAddressParam" section.

For *options*, set options of the invitation dialog. For details, refer to the "SceInvitationDialogOptions" section.

See Also

SceInvitationDialogParam, SceInvitationDialogAddressParam, SceInvitationDialogOptions

SceInvitationDialogMode

Invitation dialog display mode

Definition

```
#include <invitation_dialog.h>
typedef int32_t SceInvitationDialogMode;
```

Description

These constants indicate the display mode for the invitation dialog.

Value	(Number)	Description
SCE_INVITATION_DIALOG_MODE_INVALID	0	Invalid mode/initial value
SCE_INVITATION_DIALOG_MODE_SEND	1	Send mode
SCE_INVITATION_DIALOG_MODE_RECV	2	Receive mode

See Also

SceInvitationDialogParam

SceInvitationDialogOptions

Invitation dialog options

Definition

```
#include <invitation_dialog.h>
typedef SceInt32 SceInvitationDialogOptions;
```

Description

These options can be specified upon sending/receiving invitations using the invitation dialog.

Set the behavior upon calling the invitation dialog by specifying the bit OR of the following values to the *options* member of the SceInvitationDialogDataParam structure.

Value	(Number)	Description
-	0x0	No option specification
SCE_INVITATION_DIALOG_OPTION_ASSUME_SEND	0x1	In send mode, send invitation without the user having to press the send button of the invitation dialog
SCE_INVITATION_DIALOG_OPTION_SUSPEND_BLOCK	0x2	Prevent transitioning to the suspended state while sending or receiving

When SCE_INVITATION_DIALOG_OPTION_ASSUME_SEND is specified, an invitation can be sent upon calling the invitation dialog without the user having to press the send button of the invitation dialog. This option specification is equivalent to SCE_NP_MESSAGE_DIALOG_OPTION_ASSUME_SEND provided by the NP Message Dialog library.

When SCE_INVITATION_DIALOG_OPTION_SUSPEND_BLOCK is specified, PlayStation®Vita will not transition to the suspended state while an invitation is being sent/received.

See Also

SceInvitationDialogDataParam

SceInvitationDialogParam

Invitation dialog parameters

Definition

```
#include <invitation_dialog.h>
typedef struct SceInvitationDialogParam {
        SceUInt32 sdkVersion;
        SceCommonDialogParam commonParam;
        SceInvitationDialogMode mode;
        void * callbackArg;
        const SceInvitationDialogDataParam *dataParam;
        SceUChar8 reserved[64];
} SceInvitationDialogParam;
```

Members

sdkVersion	SDK version
commonParam	Common dialog base parameters
mode	Invitation dialog display mode
callbackArg	Application-defined data or NULL
dataParam	Data parameters required for sending or receiving invitations
reserved	Reserved area (fill with 0's)

Description

This structure is for specifying parameters for the invitation dialog when the dialog is displayed with `sceInvitationDialogInit()`. Before using this structure, it must be initialized using `sceInvitationDialogParamInit()`.

For *mode*, specify the display mode of the invitation dialog. Specify one of the following values.

Value	(Number)	Description
SCE_INVITATION_DIALOG_MODE_SEND	1	Send mode
SCE_INVITATION_DIALOG_MODE_RECV	2	Receive mode

An arbitrary pointer can be specified for *callbackArg*. The pointer specified here will be stored as-is in the *callbackArg* member of the `SceInvitationDialogResult` structure when the call result is obtained with `sceInvitationDialogGetResult()`. Specify NULL when not requiring application-defined data.

For *dataParam*, specify the data parameters (invitation message, addressees, etc.) required for sending or receiving invitations. For details, refer to the "`SceInvitationDialogDataParam`" section.

For *commonParam*, specify common parameters for the Common dialog.

Although the display status of the info bar should be specified to *commonParam.infobarParam*, always specify NULL for this value as the info bar is always displayed in the invitation dialog and the application cannot control its display status.

Although the background color in ARGB format should be specified to *commonParam.bgColor*, when using the invitation dialog in a game, only 0 (completely transparent) or 255 (completely opaque) can be specified to *commonParam.bgColor.a* (alpha component of the background color).

For details on common parameters for the Common dialog, refer to the "Common Dialog Reference" document.

reserved is a reserved area. This area must be filled with 0's.

See Also

`SceInvitationDialogDataParam`, `sceInvitationDialogParamInit()`, `sceInvitationDialogInit()`

sceInvitationDialogParamInit

Macro for initializing call parameters

Definition

```
#include <invitation_dialog.h>
#define sceInvitationDialogParamInit sceInvitationDialogParamInitialize
static inline
void sceInvitationDialogParamInitialize(SceInvitationDialogParam *param)
{
        sceClibMemset( param, 0x0, sizeof(SceInvitationDialogParam) );
        _sceCommonDialogSetMagicNumber( &param->commonParam );
        param->sdkVersion = SCE_PSP2_SDK_VERSION;
        param->mode = SCE_INVITATION_DIALOG_MODE_INVALID;
}
```

Arguments

param Call parameters to initialize

Return Values

None

Description

This macro function initializes parameters for calling the invitation dialog.

Before making individual settings to call parameters, always use this macro to initialize the structure. The appropriate SDK version will be set and the reserved area will be filled with 0's.

Examples

```
SceInvitationDialogParam param;
sceInvitationDialogParamInit(&param);
```

See Also

SceInvitationDialogParam

Call/Termination of the Invitation Dialog

sceInvitationDialogInit

Call the invitation dialog

Definition

```
#include <invitation_dialog.h>
SceInt32 sceInvitationDialogInit(
        const SceInvitationDialogParam *param
)
```

Arguments

param Call parameters

Return Values

Returns SCE_OK (0) as the value of the function for normal termination.

Returns one of the following error codes (a negative value) for an error.

Value	(Number)	Description
SCE_COMMON_DIALOG_ERROR_BUSY	0x80020401	Another Common dialog is running
SCE_COMMON_DIALOG_ERROR_NULL	0x80020402	NULL was specified for *param*
SCE_COMMON_DIALOG_ERROR_INVALID_ARGUMENT	0x80020403	The content of *param* is invalid
SCE_COMMON_DIALOG_ERROR_UNEXPECTED_FATAL	0x8002047F	Internal error

Description

This function calls the invitation dialog.

For **param*, specify the invitation dialog parameter structure. Have the structure initialized in advance with sceInvitationDialogParamInit() and set with the required parameters (operation mode, for example).

This function can only be called while other Common dialogs (including the invitation dialog itself) are not called. When called at any other time, SCE_COMMON_DIALOG_ERROR_BUSY returns.

When the call of this function succeeds, the operation status will immediately transition to SCE_COMMON_DIALOG_STATUS_RUNNING. For details on operation statuses, refer to the "sceInvitationDialogGetStatus" section and the "Common Dialog Overview" document.

Examples

```
SceInvitationDialogParam param;

sceInvitationDialogParamInit( &param );
param.mode = SCE_INVITATION_DIALOG_MODE_RECV;

// Display the invitation dialog
ret = sceInvitationDialogInit(&param) ;
if (ret < 0) {
        // Error handling
}
```

See Also

SceInvitationDialogParam, sceInvitationDialogParamInit()

sceInvitationDialogTerm

Terminate the invitation dialog

Definition

```
#include <invitation_dialog.h>
SceInt32 sceInvitationDialogTerm()
```

Arguments

None

Return Values

Returns SCE_OK (0) as the value of the function for normal termination.
Returns one of the following error codes (a negative value) for an error.

Value	(Number)	Description
SCE_COMMON_DIALOG_ERROR_NOT_FINISHED	0x80020410	Called at a time other than during the appropriate operation status
SCE_COMMON_DIALOG_ERROR_NOT_IN_USE	0x80020411	sceInvitationDialogInit() has not been called

Description

This function terminates the invitation dialog. The invitation dialog must always be terminated with this function after being called with sceInvitationDialogInit().

This function can only be called while the operation status of the invitation dialog is SCE_COMMON_DIALOG_STATUS_FINISHED. If the invitation dialog has not been called, SCE_COMMON_DIALOG_ERROR_NOT_IN_USE returns. If this function is called at times other than the above, SCE_COMMON_DIALOG_ERROR_NOT_FINISHED returns.

When the call of this function succeeds, the operation status will immediately transition to SCE_COMMON_DIALOG_STATUS_NONE.
For details on operation statuses, refer to the "sceInvitationDialogGetStatus" section.

Examples

```
SceCommonDialogStatus stat;

while(1) {
        stat = sceInvitationDialogGetStatus();
        if( stat == SCE_COMMON_DIALOG_STATUS_FINISHED ) {
              sceInvitationDialogTerm();
              break;
        }
}
```

See Also

sceInvitationDialogInit(),sceInvitationDialogGetStatus()

Aborting the Invitation Dialog

sceInvitationDialogAbort

Abort call of the invitation dialog

Definition

```
#include <invitation_dialog.h>
SceInt32 sceInvitationDialogAbort()
```

Arguments

None

Return Values

Returns SCE_OK (0) as the value of the function for normal termination.

Returns the following error code (a negative value) for an error.

Value	(Number)	Description
SCE_COMMON_DIALOG_ERROR_NOT_IN_USE	0x80020411	sceInvitationDialogInit() has not been called

Description

This function aborts the invitation dialog. Use this function to immediately abort display of the invitation dialog when, for example, interrupt processing must suddenly be carried out.

This function can be called at any time after calling sceInvitationDialogInit() and before calling sceInvitationDialogTerm(). If this function is called at times other than the above, SCE_COMMON_DIALOG_ERROR_NOT_IN_USE returns.

When the call of this function succeeds, the process to close the displayed invitation dialog will start and the operation status will transition to SCE_COMMON_DIALOG_STATUS_FINISHED after close processing completes.

For details on operation statuses, refer to the "sceInvitationDialogGetStatus" section.

When this function is used to close the invitation dialog, the call result obtained with sceInvitationDialogGetResult() will be SCE_COMMON_DIALOG_RESULT_ABORTED.

Examples

```
SceCommonDialogStatus stat;

while(1) {
        stat = sceInvitationDialogGetStatus();
        if( stat == SCE_COMMON_DIALOG_STATUS_RUNNING ) {
              if( need_abort ) {
                    sceInvitationDialogAbort();
                    break;
              }
        }
        else if( stat == SCE_COMMON_DIALOG_STATUS_FINISHED ) {
              sceInvitationDialogTerm();
              break;
        }
}
```

See Also

`sceInvitationDialogGetResult()`

Obtaining the Invitation Dialog Operation Status

sceInvitationDialogGetStatus

Get invitation dialog operation status

Definition

```
#include <invitation_dialog.h>
SceCommonDialogStatus sceInvitationDialogGetStatus()
```

Arguments

None

Return Values

Returns one of the following operation statuses as the value of the function.

Value	(Number)	Description
SCE_COMMON_DIALOG_STATUS_NONE	0	Invitation dialog is not running
SCE_COMMON_DIALOG_STATUS_RUNNING	1	Invitation dialog is running
SCE_COMMON_DIALOG_STATUS_FINISHED	2	Invitation dialog operation has finished

Description

This function obtains the invitation dialog operation status.

The operation status will be SCE_COMMON_DIALOG_STATUS_NONE before the call of the invitation dialog.

The operation status will transition to SCE_COMMON_DIALOG_STATUS_RUNNING when the call of the invitation dialog succeeds with sceInvitationDialogInit().

The operation status will transition to SCE_COMMON_DIALOG_STATUS_FINISHED a little while after the close of the invitation dialog by user operation or sceInvitationDialogAbort(). sceInvitationDialogGetResult() and sceInvitationDialogTerm() can only be called while the operation status is SCE_COMMON_DIALOG_STATUS_FINISHED.

The operation status will immediately transition to SCE_COMMON_DIALOG_STATUS_NONE when sceInvitationDialogTerm() is called.

See Also

sceInvitationDialogInit(),sceInvitationDialogAbort(),
sceInvitationDialogGetResult(),sceInvitationDialogTerm()

Obtaining the Invitation Dialog Call Result

SceInvitationDialogOnlineIdList

Online ID list handled by the invitation dialog

Definition

```
#include <invitation_dialog.h>
#define SCE_INVITATION_DIALOG_ADDRESS_USER_LIST_MAX_NUMBER (16)
typedef struct SceInvitationDialogOnlineIdList {
        SceUInt32 count;
        SceNpOnlineId
onlineId[SCE_INVITATION_DIALOG_ADDRESS_USER_LIST_MAX_NUMBER];
} SceInvitationDialogOnlineIdList;
```

Members

count Number of online IDs
onlineId Online IDs

Description

This structure represents the list of online IDs handled as a batch by the invitation dialog.

See Also

SceInvitationDialogResult

SceInvitationDialogResult

Invitation dialog call result

Definition

```
#include <invitation_dialog.h>
typedef struct SceInvitationDialogResult {
        void * callbackArg;
        SceInt32 errorCode;
        SceCommonDialogResult result;
        SceInvitationDialogOnlineIdList *sentOnlineIds;
        SceUChar8 reserved[32];
} SceInvitationDialogResult;
```

Members

callbackArg	Application-defined argument specified upon calling the invitation dialog
errorCode	Invitation dialog close status
result	Invitation dialog call result
sentOnlineIds	List of addressees to whom message was actually sent or NULL
reserved	Reserved area (fill with 0's)

Description

This structure is for obtaining the invitation dialog call result with `sceInvitationDialogGetResult()`. This structure must be used after all values are filled with 0's and the structure is initialized in advance.

In *callbackArg*, the value for the *callbackArg* member of the argument *param* that was specified when `sceInvitationDialogInit()` was called will be stored as-is. This member can be used by applications for arbitrary purposes.

In *errorCode*, the error that occurred upon termination of the invitation dialog will be stored. `SCE_OK` (0) will be stored for normal termination, and a non-0 value will be stored for a fatal error. The main error codes are shown below. Note, however, that the application must not malfunction even if other error codes are returned.

For error codes, also refer to the list of return codes in the "NpWebApi Library Reference" document, include/net/errno.h, include/libnetctl.h, and the list of error codes in the "Session/Invitation Web APIs Reference" document.

Value	(Number)	Description
SCE_NP_MANAGER_ERROR_ NEED_CALL_NETCHECK_DIALOG	0x8055050b	Called without carrying out sign-in processing with Network Check Dialog
SCE_NET_ERROR_EIPADDRCHANGED	0x804101a3	Connection to the network was disconnected
SCE_NET_ERROR_EINACTIVEDISABLED		
SCE_NET_CTL_ERROR_WIFI_DISABLED	0x80412113	Wi-Fi is off
SCE_NP_WEBAPI_SERVER_ERROR_ SESSION_INVITATION_BLOCKED_USER_EXISTS	0x82204185	Attempted to send to a user registered to the blocked user list
SCE_INVITATION_DIALOG_ERROR_ LACK_OF_LIBHTTP_POOL_SIZE	0x80109201	Remaining amount of the libhttp memory pool is insufficient
SCE_INVITATION_DIALOG_ERROR_ LACK_OF_LIBSSL_POOL_SIZE	0x80109202	Remaining amount of the libssl memory pool is insufficient

result is the invitation dialog call result. One of the following values will be stored.

Value	(Number)	Description
SCE_COMMON_DIALOG_RESULT_OK	0	User carried out one of the following "OK" operations - Send completed and the dialog closed - Performed operation to join session and closed the dialog
SCE_COMMON_DIALOG_RESULT_USER_CANCELED	1	User carried out one of the following cancel operations - Closed the dialog with the close button at the upper right - Pressed the cancel button during a send in the SCE_INVITATION_DIALOG_OPTION_ASSUME_SEND mode and closed the dialog
SCE_COMMON_DIALOG_RESULT_ABORTED	2	Aborted with sceInvitationDialogAbort()

For *sentOnlineIds*, specify a pointer to the SceInvitationDialogOnlineIdList structure. When the invitation send processing by the invitation dialog succeeds, the number of addressees to whom the send was actually performed and a list of their online IDs will be stored in *sentOnlineIds*. If the online IDs of the addressees to whom send processing was performed are not needed, specify NULL for *sentOnlineIds*.

reserved is a reserved area. This area must be filled with 0's.

See Also

sceInvitationDialogGetResult()

sceInvitationDialogGetResult

Get invitation dialog call result

Definition

```
#include <invitation_dialog.h>
SceInt32 sceInvitationDialogGetResult(
        SceInvitationDialogResult *result
)
```

Arguments

result Destination to store the obtained call result

Return Values

Returns SCE_OK (0) as the value of the function for normal termination.

Returns one of the following error codes (a negative value) for an error.

Value	(Number)	Description
SCE_COMMON_DIALOG_ERROR_NULL	0x80020402	NULL was specified to *result*
SCE_COMMON_DIALOG_ERROR_NOT_FINISHED	0x80020410	Called at a time other than during the appropriate operation status
SCE_COMMON_DIALOG_ERROR_NOT_IN_USE	0x80020411	sceInvitationDialogInit() has not been called

Description

This function obtains the invitation dialog call result.

This function can only be called when the invitation dialog operation status is SCE_COMMON_DIALOG_STATUS_FINISHED. Otherwise, SCE_COMMON_DIALOG_ERROR_NOT_FINISHED will return.

For details on operation statuses, refer to the "sceInvitationDialogGetStatus" section.

The invitation dialog call result will be stored in **result* upon normal termination of this function. For details on the call result, refer to the "SceInvitationDialogResult" section.

**result* must always be initialized before it is passed to this function.

See Also

SceInvitationDialogResult, sceInvitationDialogGetStatus()

Constants

Return Codes

List of return codes returned by the InvitationDialog library

Definition

Value	(Number)	Description
SCE_COMMON_DIALOG_ERROR_BUSY	0x80020401	Another Common dialog is in use
SCE_COMMON_DIALOG_ERROR_NULL	0x80020402	NULL was specified for argument
SCE_COMMON_DIALOG_ERROR_INVALID_ARGUMENT	0x80020403	The content of argument is invalid
SCE_COMMON_DIALOG_ERROR_NOT_FINISHED	0x80020410	Called at a time other than during the appropriate operation status
SCE_COMMON_DIALOG_ERROR_NOT_IN_USE	0x80020411	sceInvitationDialogInit() has not been called
SCE_COMMON_DIALOG_ERROR_UNEXPECTED_FATAL	0x8002047F	Internal error
SCE_NP_MANAGER_ERROR_NEED_CALL_NETCHECK_DIALOG	0x8055050b	Called without carrying out sign-in processing with the Network Check dialog

In addition, error codes of the NpWebApi library will return to *errorCode* of SceInvitationDialogResult that can be obtained with sceInvitationDialogGetResult().

For error codes, also refer to the list of return codes of the "NpWebApi Library Reference" document, include/net/errno.h, include/libnetctl.h, and the list of error codes of the "Session/Invitation Web APIs Reference" document.