

NP Lookup Library Reference

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

Table of Contents

Library Initialization/Termination	3
sceNpLookupInit	4
sceNpLookupTerm	6
Context/Request Operations	7
sceNpLookupCreateTitleCtx	8
sceNpLookupDeleteTitleCtx	9
sceNpLookupCreateRequest	10
sceNpLookupDeleteRequest	11
sceNpLookupSetTimeout	12
sceNpLookupAbortRequest	14
sceNpLookupWaitAsync, sceNpLookupPollAsync	15
Communication Processing	16
sceNpLookupNpId	17
sceNpLookupNpIdAsync	19
sceNpLookupUserProfile	21
sceNpLookupUserProfileAsync	24
sceNpLookupAvatarImage	27
sceNpLookupAvatarImageAsync	29
Constants	31
SCE_NP_LOOKUP_MAX_CTX_NUM	32
SCE_NP_LOOKUP_MAX_REQUEST_NUM	33
SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT	34

Library Initialization/Termination

SCE CONFIDENTIAL

sceNpLookupInit

Initialize the NP Lookup library

Definition

```
#include <np.h>
int sceNpLookupInit(
    SceBool usesAsync,
    SceInt32 threadPriority,
    SceInt32 cpuAffinityMask,
    void *option
);
```

Arguments

<i>usesAsync</i>	Set to SCE_TRUE if utilizing an asynchronous function. When SCE_TRUE is set, 1 thread will be generated internally.
<i>threadPriority</i>	Thread priority of the thread generated when <i>usesAsync</i> is SCE_TRUE.
<i>cpuAffinityMask</i>	CPU affinity mask of the thread generated when <i>usesAsync</i> is SCE_TRUE.
<i>option</i>	Option for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_ALREADY_INITIALIZED	0x80550701	Library has already been initialized
SCE_NP_COMMUNITY_ERROR_OUT_OF_MEMORY	0x80550703	Not enough free memory
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Invalid argument (<i>threadPriority</i> is a negative value, etc.)
SCE_NP_COMMUNITY_ERROR_NO_RESOURCE	0x80550716	Could not allocate system resources

Description

This function initializes the NP Lookup library. This function must be called before using the NP Lookup library.

Notes

This function is not multithread safe. If called simultaneously from multiple threads, even if this function terminates normally, the library may not be able to function correctly afterwards. Therefore, program so as to avoid simultaneous calling.

Examples

Below is a usage example in case asynchronous functions are not used.

```
SceInt32 ret;
ret = sceNpLookupInit(
    SCE_FALSE,
    0,
    0,
    NULL);
If (ret < 0){
    // Error handling
}
```

Below is a usage example in case asynchronous functions are used.

Set appropriate values for thread priority and CPU affinity mask.

```
SceInt32 ret;
ret = sceNpLookupInit(
    SCE_TRUE,
    SCE_KERNEL_DEFAULT_PRIORITY_USER,
    SCE_KERNEL_THREAD_CPU_AFFINITY_MASK_DEFAULT,
    NULL);
If (ret < 0){
    // Error handling
}
```

See Also

sceNpLookupTerm()

SCE CONFIDENTIAL

sceNpLookupTerm

Terminate the NP Lookup library

Definition

```
#include <np.h>
int sceNpLookupTerm(void);
```

Arguments

None

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized

Description

This function terminates the NP Lookup library.

Notes

This function is not multithread safe. If called simultaneously from multiple threads, even if this function terminates normally, the library may not be able to function correctly afterwards. Therefore, program so as to avoid simultaneous calling.

See Also

sceNpLookupInit()

Context/Request Operations

SCE CONFIDENTIAL

sceNpLookupCreateTitleCtx

Create an NP Lookup title context

Definition

```
#include <np.h>
int sceNpLookupCreateTitleCtx(
    const SceNpCommunicationId *communicationId,
    const SceNpId *selfNpId
);
```

Arguments

communicationId Pointer to the NP Communication ID
selfNpId Pointer to the NP ID structure of a login user

Return Values

Returns the ID (>0) of the NP Lookup title context upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_OUT_OF_MEMORY	0x80550703	Not enough free memory
SCE_NP_COMMUNITY_ERROR_TOO_MANY_OBJECTS	0x80550706	Created more than 32 title contexts at one time. (Check to see that <code>sceNpLookupDeleteTitleCtx()</code> was called as necessary.)
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>selfNpId</i>

Description

This function creates a title context of the NP Lookup library.

Up to 32 title contexts can be created at one time, however, try to use around one title context wherever possible.

When `sceNpLookupTerm()` is called, all title contexts will be deleted automatically. However, try to call `sceNpLookupTerm()` after `sceNpLookupDeleteTitleCtx()` to have the title contexts explicitly deleted beforehand.

Notes

When NULL is passed to the NP communication ID of this function, the NP communication ID set with the `sceNpInit()` function of the NP library is used.

Although data based on the NP Communication ID is not handled in current specifications, the correct NP Communication ID must be input. When multiple NP Communication IDs are used by a single title, select and set any one.

See Also

`sceNpLookupDeleteTitleCtx()`, `SCE_NP_LOOKUP_MAX_CTX_NUM`

©SCEI

SCE CONFIDENTIAL

sceNpLookupDeleteTitleCtx

Destroy the NP Lookup title context

Definition

```
#include <np.h>
int sceNpLookupDeleteTitleCtx(
    SceInt32 titleCtxId
);
#define sceNpLookupDestroyTitleCtx(id) \
    sceNpLookupDeleteTitleCtx(id)
```

Arguments

titleCtxId ID of the NP Lookup title context to be destroyed

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>titleCtxId</i> does not exist

Description

This function destroys the NP lookup title context used by the NP Lookup library.

See Also

`sceNpLookupCreateTitleCtx()`

SCE CONFIDENTIAL

sceNpLookupCreateRequest

Create an NP Lookup request

Definition

```
#include <np.h>
int sceNpLookupCreateRequest(
    SceInt32 titleCtxId
);
#define sceNpLookupCreateTransactionCtx(id) \
    sceNpLookupCreateRequest(id)
```

Arguments

titleCtxId NP Lookup title context ID

Return Values

Returns the ID (>0) of the NP Lookup request upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_OUT_OF_MEMORY	0x80550703	Not enough free memory
SCE_NP_COMMUNITY_ERROR_TOO_MANY_OBJECTS	0x80550706	Created more than 32 requests at one time. (Check to see that <code>sceNpLookupDeleteRequest()</code> was called as necessary.)
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>titleCtxId</i> does not exist

Description

This function creates a request of the NP Lookup library.

A request ID is an ID that is used up for each communication processing function, such as retrieving NP ID or retrieving Avatars. A request ID must always be created before using these functions. Moreover, a used up request must be deleted with `sceNpLookupDeleteRequest()`.

Up to 32 requests can exist simultaneously.

When `sceNpLookupTerm()` is called, all requests will be deleted automatically. However, try to call `sceNpLookupTerm()` after delete the title contexts explicitly beforehand.

See Also

`sceNpLookupDeleteRequest()`, `SCE_NP_LOOKUP_MAX_REQUEST_NUM`

SCE CONFIDENTIAL

sceNpLookupDeleteRequest

Destroy the NP Lookup request

Definition

```
#include <np.h>
int sceNpLookupDeleteRequest(
    SceInt32 reqId
);
#define sceNpLookupDestroyTransactionCtx(id) \
    sceNpLookupDeleteRequest(id)
```

Arguments

reqId NP Lookup request ID

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_OUT_OF_MEMORY	0x80550703	Not enough free memory
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist

Description

This function destroys the NP Lookup request used by the NP Lookup library.

See Also

`sceNpLookupCreateRequest()`

sceNpLookupSetTimeout

Set a timeout period for communication processing functions

Definition

```
#include <np.h>
int sceNpLookupSetTimeout (
    SceInt32 id,
    SceInt32 resolveRetry,
    SceUInt32 resolveTimeout,
    SceUInt32 connTimeout,
    SceUInt32 sendTimeout,
    SceUInt32 recvTimeout
);
```

Arguments

<i>id</i>	ID of an NP Lookup title context or NP Lookup request
<i>resolveRetry</i>	Name resolution retry times
<i>resolveTimeout</i>	Name resolution timeout time (microseconds)
<i>connTimeout</i>	Timeout time when connecting (microseconds)
<i>sendTimeout</i>	Sending timeout time (microseconds)
<i>recvTimeout</i>	Receiving timeout time (microseconds)

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Invalid value was specified for <i>timeout</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	Title context or request specified for <i>id</i> does not exist

Description

This function sets the timeout period for communication processing.

For *id*, specify an NP Lookup title context ID if you want to set a uniform timeout period, or specify an NP Lookup request ID if you want to set a timeout period per request. When setting a timeout period for a title context, it will be applied to all transaction contexts that you subsequently create after this function. It will not, however, be applied to already-existing transaction contexts.

Specify the timeout period in microseconds.

Only the following values can be set for each value.

Argument	Settable values
<i>resolveRetry</i>	1 or more, or SCE NP LOOKUP TIMEOUT NO EFFECT
<i>resolveTimeout</i>	1 second or more, or SCE NP LOOKUP TIMEOUT NO EFFECT
<i>connTimeout</i>	10 seconds or more, or SCE NP LOOKUP TIMEOUT NO EFFECT
<i>sendTimeout</i>	10 seconds or more, or SCE NP LOOKUP TIMEOUT NO EFFECT
<i>recvTimeout</i>	10 seconds or more, or SCE NP LOOKUP TIMEOUT NO EFFECT

SCE CONFIDENTIAL

Timeout time settings will result in error if *resolveTimeout* is set to a value under 1,000,000 (1 second), or other values are set to a value below 10,000,000 (10 seconds).

When `SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT` is set in any of the arguments, the internal default value will be used as setting value of that argument.

It is not possible to set `SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT` in all arguments.

Notes

Instead of using this function to set your own timeout period, try to use the default timeout period wherever possible. When using it to avoid user operation from becoming impossible for an extensive period of time, consider implementing user cancellation using `sceNpLookupAbortRequest()`.

With the values set with this function, it is possible to set the timeout time for each processing, but not the total timeout time.

Most communication processing of the NP Lookup library end within two seconds. However, an extensive delay may occur depending on the user's network environment and it is recommended that you set a timeout period of at least 30 seconds.

See Also

`sceNpLookupCreateTitleCtx()`, `sceNpLookupCreateRequest()`,
`sceNpLookupAbortRequest()`, `SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT`

sceNpLookupAbortRequest

Abort communication processing

Definition

```
#include <np.h>
int sceNpLookupAbortRequest (
    SceInt32 reqId
);
#define sceNpLookupAbortTransaction(id) \
    sceNpLookupAbortRequest(id)
```

Arguments

reqId NP Lookup request ID

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist

Description

This function aborts communication processing.

For *reqId*, specify the ID of the request for the communication processing you wish to abort.

See Also

sceNpLookupCreateRequest ()

SCE CONFIDENTIAL

sceNpLookupWaitAsync, sceNpLookupPollAsync

Obtain the execution result for an asynchronous request

Definition

```
#include <np.h>
int sceNpLookupWaitAsync (
    SceInt32 reqId,
    SceInt32 *result
);
int sceNpLookupPollAsync (
    SceInt32 reqId,
    SceInt32 *result
);
```

Arguments

reqId NP Lookup request ID
result Pointer to a variable storing the result of the request that was asynchronously executed

Return Values

Value	Description
0	Asynchronous processing completed
1	Asynchronous processing still in execution

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist

Description

This function obtains the execution result of a request executed asynchronously.

For *reqId*, specify the ID of the request started by the asynchronous function, `sceNpLookupXXXAsync()`.

`sceNpLookupWaitAsync()` waits for this request to complete if it hasn't already done so, stores the result of the request in **result* and returns 0.

`sceNpLookupPollAsync()` returns 1 if the request has not completed yet. In this case, the value of **result* does not change. If the request is already completed, the result of the request will be stored in **result* and the return value will be 0.

See Also

`sceNpLookupCreateRequest()`

Communication Processing

SCE CONFIDENTIAL

sceNpLookupNpId

Look up an NP ID (synchronous)

Definition

```
#include <np.h>
int sceNpLookupNpId(
    SceInt32 reqId,
    const SceNpOnlineId *onlineId,
    SceNpId *npId,
    void *option
);
```

Arguments

reqId NP Lookup request ID (IN)
onlineId Target Online ID (IN)
npId Pointer to the structure storing the obtained NP ID (OUT)
option Option for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>onlineId</i> or <i>npId</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Request used for communication processing was specified. Make sure to destroy the request used for communication processing.
SCE_NP_COMMUNITY_SERVER_ERROR_NO_SUCH_USER_NPID	0x8055081b	User does not exist

Description

This function looks up the user corresponding to the specified Online ID string, and obtains his/her NP ID.

This function performs synchronous processing. It blocks until communication completes and the NP ID can be obtained. Once returning from this function, destroy the used request with `sceNpLookupDeleteRequest()`.

SCE CONFIDENTIAL

See Also

`sceNpLookupCreateRequest()`, `sceNpLookupDeleteRequest()`,
`sceNpLookupAbortRequest()`

000004892117

SCE CONFIDENTIAL

sceNpLookupNpIdAsync

Look up an NP ID (asynchronous)

Definition

```
#include <np.h>
int sceNpLookupNpIdAsync (
    SceInt32 reqId,
    const SceNpOnlineId *onlineId,
    SceNpId *npId,
    void *option
);
```

Arguments

reqId NP Lookup request ID (IN)
onlineId Target Online ID (IN)
npId Pointer to the structure storing the obtained NP ID (OUT)
option Option for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>onlineId</i> or <i>npId</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Request used for communication processing was specified. Make sure to destroy the request used for communication processing.
SCE_NP_COMMUNITY_SERVER_ERROR_NO_SUCH_USER_NPID	0x8055081b	User does not exist

Description

This function looks up the user corresponding to the specified Online ID string, and obtains his/her NP ID.

This function performs asynchronous processing. It starts the request and returns immediately without waiting to obtain the result from the server. The processing result must be received using `sceNpLookupWaitAsync()` or `sceNpLookupPollAsync()`. Once the result is received using one of these functions, destroy the request.

SCE CONFIDENTIAL

Notes

In order to use this function, it is necessary to set *usesAsync* of `sceNpLookupInit()` to `SCE_TRUE`, and to generate a thread for asynchronous functions.

See Also

`sceNpLookupCreateRequest()`, `sceNpLookupAbortRequest()`, `sceNpLookupWaitAsync()`,
`sceNpLookupPollAsync()`

000004892117

SCE CONFIDENTIAL

sceNpLookupUserProfile

Obtain a user profile (synchronous)

Definition

```
#include <np.h>
int sceNpLookupUserProfile(
    SceInt32 reqId,
    SceInt32 avatarSizeType,
    const SceNpId *npId,
    SceNpUserInformation *userInfo,
    SceNpAboutMe *aboutMe,
    SceNpMyLanguages *languages,
    SceNpCountryCode *countryCode,
    void *avatarImageData,
    SceSize avatarImageDataMaxSize,
    SceSize *avatarImageDataSize,
    void *option
);
```

Arguments

<i>reqId</i>	NP Lookup request ID (IN)
<i>avatarSizeType</i>	Size type of the Avatar to be obtained (details below) (IN)
<i>npId</i>	Target NP ID (IN)
<i>userInfo</i>	Pointer to the structure storing the obtained user information (OUT)
<i>aboutMe</i>	Pointer to the structure storing the obtained self introduction (OUT)
<i>languages</i>	Pointer to the structure storing the obtained language preference (OUT)
<i>countryCode</i>	Pointer to the structure storing the obtained country or region (OUT)
<i>avatarImageData</i>	Pointer to the structure storing the obtained Avatar image (OUT)
<i>avatarImageDataMaxSize</i>	Buffer size specified in <i>avatarImageData</i> (details below) (IN)
<i>avatarImageDataSize</i>	Pointer to the variable storing the size of the actually obtained Avatar (OUT)
<i>option</i>	Option for future extension. Always specify NULL

Specify one of the following values in *avatarSizeType*.

Value	Description
SCE_NP_AVATAR_SIZE_LARGE	Large-sized Avatar image
SCE_NP_AVATAR_SIZE_MIDDLE	Middle-sized Avatar image
SCE_NP_AVATAR_SIZE_SMALL	Small-sized Avatar image

Specify one of the following values in *avatarImageDataMaxSize* based on the value of *avatarSizeType*.

<i>avatarSizeType</i> Value	Value to be specified	Value
SCE_NP_AVATAR_SIZE_LARGE	SCE_NP_AVATAR_IMAGE_MAX_SIZE_LARGE	200*1024
SCE_NP_AVATAR_SIZE_MIDDLE	SCE_NP_AVATAR_IMAGE_MAX_SIZE_MIDDLE	100*1024
SCE_NP_AVATAR_SIZE_SMALL	SCE_NP_AVATAR_IMAGE_MAX_SIZE_SMALL	10*1024

SCE CONFIDENTIAL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>npId</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Invalid value was specified for <i>avatarSizeType</i> , or request used for communication processing was specified. Make sure to destroy the request used for communication processing.
SCE_NP_COMMUNITY_SERVER_ERROR_NO_SUCH_USER_NPID	0x8055081b	User does not exist

Description

This function obtains profile information and an Avatar image of the user from the server with the specified NP ID and the Avatar size type. The URL of an Avatar of a size specified in *avatarSizeType* is stored in *icon* of *userInfo*.

Make sure to always allocate the size specified with *avatarImageDataMaxSize* in *avatarImageData*.

When an Avatar image is not necessary, specify NULL in *avatarImageData* and *avatarImageDataMaxSize*. In this case, processing will be omitted, decreasing the load.

If not necessary, it is also possible to specify NULL for *userInfo*, *aboutMe*, *languages* and *countryCode*. However, since this information is handled collectively, the load will not decrease.

This function performs synchronous processing. It blocks until communication completes and the profile information can be obtained. Once returning from this function, call `sceNpLookupDeleteRequest()` to destroy the used request.

Notes

The Avatar image, if the *SceNpAvatarUrl* structure of desired size type has already been obtained, can be obtained using `sceNpLookupAvatarImage()` or `sceNpLookupAvatarImageAsync()`. Moreover, the application can directly obtain image data from the URL of the Avatar, however, make sure to note the data size in this case.

SCE CONFIDENTIAL

Examples

```

SceInt32 ret;
SceInt32 reqId, titleCtxId;
SceNpId npId;
SceNpUserInformation userInfo;
SceNpAboutMe aboutMe;
SceNpMyLanguages languages;
SceNpCountryCode countryCode;
void *avatarImageData=NULL;
SceSize avatarImageDataSize=0;

// reqId, titleCtxId, npId assumed to store an appropriate value

avatarImageData = malloc(SCE_NP_AVATAR_IMAGE_MAX_SIZE_SMALL);
if (avatarImageData == NULL){
    // Error handling
}

ret = sceNpLookupUserProfile(
    reqId,
    SCE_NP_AVATAR_SIZE_SMALL,
    &npId,
    &userInfo,
    &aboutMe,
    &languages,
    &countryCode,
    avatarImageData,
    SCE_NP_AVATAR_IMAGE_MAX_SIZE_SMALL,
    &avatarImageDataSize,
    NULL);
If (ret < 0){
    // Error handling
}
// Use of retrieved data

```

See Also

```

sceNpLookupCreateRequest(), sceNpLookupAbortRequest(),
sceNpLookupUserProfileAsync()

```

SCE CONFIDENTIAL

sceNpLookupUserProfileAsync

Obtain a user profile (asynchronous)

Definition

```
#include <np.h>
int sceNpLookupUserProfileAsync (
    SceInt32 reqId,
    SceInt32 avatarSizeType,
    const SceNpId *npId,
    SceNpUserInformation *userInfo,
    SceNpAboutMe *aboutMe,
    SceNpMyLanguages *languages,
    SceNpCountryCode *countryCode,
    void *avatarImageData,
    SceSize avatarImageDataMaxSize,
    SceSize *avatarImageDataSize,
    void *option
);
```

Arguments

<i>reqId</i>	NP Lookup request ID (IN)
<i>avatarSizeType</i>	Size type of the Avatar to be obtained (details below) (IN)
<i>npId</i>	Target NP ID (IN)
<i>userInfo</i>	Pointer to the structure storing the obtained user information (OUT)
<i>aboutMe</i>	Pointer to the structure storing the obtained self introduction (OUT)
<i>languages</i>	Pointer to the structure storing the obtained language preference (OUT)
<i>countryCode</i>	Pointer to the structure storing the obtained country or region (OUT)
<i>avatarImageData</i>	Pointer to the structure storing the obtained Avatar image(OUT)
<i>avatarImageDataMaxSize</i>	Size of the buffer specified in <i>avatarImageData</i> (details below) (IN)
<i>avatarImageDataSize</i>	Pointer to the variable storing the size of the actually obtained Avatar (OUT)
<i>option</i>	Option for future extension. Always specify NULL

Specify one of the following values in *avatarSizeType*.

Value	Description
SCE_NP_AVATAR_SIZE_LARGE	Large-sized Avatar image
SCE_NP_AVATAR_SIZE_MIDDLE	Middle-sized Avatar image
SCE_NP_AVATAR_SIZE_SMALL	Small-sized Avatar image

Specify one of the following values in *avatarImageDataMaxSize* based on the value of *avatarSizeType*.

<i>avatarSizeType</i> Value	Value to be specified	Value
SCE_NP_AVATAR_SIZE_LARGE	SCE_NP_AVATAR_IMAGE_MAX_SIZE_LARGE	200*1024
SCE_NP_AVATAR_SIZE_MIDDLE	SCE_NP_AVATAR_IMAGE_MAX_SIZE_MIDDLE	100*1024
SCE_NP_AVATAR_SIZE_SMALL	SCE_NP_AVATAR_IMAGE_MAX_SIZE_SMALL	10*1024

SCE CONFIDENTIAL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>npId</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Invalid value was specified for <i>avatarSizeType</i> , or request used for communication processing was specified. Make sure to destroy the request used for communication processing.
SCE_NP_COMMUNITY_SERVER_ERROR_NO_SUCH_USER_NPID	0x8055081b	User does not exist

Description

This function obtains profile information and an Avatar image of the user from the server with the specified NP ID and the Avatar size type. The URL of an Avatar of the size specified in *avatarSizeType* is stored in *icon* of *userInfo*.

Please make sure to allocate the size specified with *avatarImageDataMaxSize* in *avatarImageData*.

When an Avatar image is not necessary, specify NULL in *avatarImageData* and *avatarImageDataMaxSize*. In this case, processing will be omitted, decreasing the load.

If not necessary, it is also possible to specify NULL for *userInfo*, *aboutMe*, *languages*, and *countryCode*. However, since this information is handled collectively, the load will not decrease.

This function performs asynchronous processing. When the request is started, the function returns without waiting to obtain the result of the processing from the server. The result of the processing can be obtained by `sceNpLookupWaitAsync()` or `sceNpLookupPollAsync()`. After receiving the result from one of these functions, destroy the used request.

Notes

The Avatar image, if the `SceNpAvatarUrl` structure of desired size type has already been obtained, can be obtained using `sceNpLookupAvatarImage()` or `sceNpLookupAvatarImageAsync()`. Moreover, the application can directly obtain image data from the URL of the Avatar, however, make sure to note the data size in this case.

In order to use this function, it is necessary to set *usesAsync* of `sceNpLookupInit()` to `SCE_TRUE`, and to generate a thread for asynchronous functions.

SCE CONFIDENTIAL

See Also

`sceNpLookupCreateRequest()`, `sceNpLookupAbortRequest()`, `sceNpLookupWaitAsync()`,
`sceNpLookupPollAsync()`

000004892117

SCE CONFIDENTIAL

sceNpLookupAvatarImage

Obtain an Avatar image (synchronous)

Definition

```
#include <np.h>
int sceNpLookupAvatarImage (
    SceInt32 reqId,
    const SceNpAvatarUrl *avatarUrl,
    SceNpAvatarImage *avatarImage,
    void *option
);
```

Arguments

<i>reqId</i>	NP Lookup request ID (IN)
<i>avatarUrl</i>	Target Avatar URL (IN)
<i>avatarImage</i>	Structure for storing the obtained Avatar image (OUT)
<i>option</i>	Option for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>avatarUrl</i> or <i>avatarImage</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Request used for communication processing was specified. Make sure to destroy the request used for communication processing.

Description

This function obtains the Avatar image specified by *avatarUrl* from the server. Do not use this function for obtaining any other files.

This function performs synchronous processing. It blocks until the obtainment of the Avatar image completes. Once returning from this function, call `sceNpLookupDeleteRequest()` to destroy the used request.

Notes

In addition to obtaining the Avatar image, this function prevents the problem of an application's operation stopping because of the change in the user's Avatar size. The application can obtain the

SCE CONFIDENTIAL

Avatar image on its own using libhttp, for example, but in that case, note that the Avatar size will increase to SCE_NP_AVATAR_IMAGE_MAX_SIZE.

See Also

`sceNpLookupCreateRequest()`, `sceNpLookupDeleteRequest()`,
`sceNpLookupAbortRequest()`

000004892117

sceNpLookupAvatarImageAsync

Obtain an Avatar image (asynchronous)

Definition

```
#include <np.h>
int sceNpLookupAvatarImageAsync (
    SceInt32 reqId,
    const SceNpAvatarUrl *avatarUrl,
    SceNpAvatarImage *avatarImage,
    void *option
);
```

Arguments

<i>reqId</i>	NP Lookup request ID (IN)
<i>avatarUrl</i>	Target Avatar URL (IN)
<i>avatarImage</i>	Structure for storing the obtained Avatar image (OUT)
<i>option</i>	Option for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value for an error. The main error codes are shown below. (The application must not malfunction even if other error codes are returned.)

Value	Hexadecimal	Description
SCE_NP_COMMUNITY_ERROR_NOT_INITIALIZED	0x80550702	Library is not initialized
SCE_NP_COMMUNITY_ERROR_INVALID_ARGUMENT	0x80550704	Value other than NULL was specified to <i>option</i>
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with <code>sceNpLookupAbortRequest()</code> or <code>sceNpLookupDeleteRequest()</code>
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to <i>avatarUrl</i> or <i>avatarImage</i>
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for <i>reqId</i> does not exist
SCE_NP_COMMUNITY_ERROR_INVALID_TYPE	0x80550711	Request used for communication processing was specified. Make sure to destroy the request used for communication processing.

Description

This function obtains the Avatar image specified by *avatarUrl* from the server. Do not use this function for obtaining any other files.

This function performs asynchronous processing. When the transaction is started, the function returns without waiting to obtain the result of the processing from the server. The result of the processing can be obtained by `sceNpLookupWaitAsync()` or `sceNpLookupPollAsync()`. After receiving the result from one of these functions, destroy the used request.

SCE CONFIDENTIAL

Notes

In addition to obtaining the Avatar image, this function prevents the problem of an application's operation stopping because of the change in the user's Avatar size. The application can obtain the Avatar image on its own using libhttp, for example, but in that case, note that the Avatar size will increase to `SCE_NP_AVATAR_IMAGE_MAX_SIZE`.

In order to use this function, it is necessary to set *usesAsync* of `sceNpLookupInit()` to `SCE_TRUE`, and to generate a thread for asynchronous functions.

See Also

`sceNpLookupCreateRequest()`, `sceNpLookupAbortRequest()`, `sceNpLookupWaitAsync()`,
`sceNpLookupPollAsync()`

Constants

000004892117

SCE CONFIDENTIAL

SCE_NP_LOOKUP_MAX_CTX_NUM

Maximum number of contexts

Definition

```
#include <np.h>

#define SCE_NP_LOOKUP_MAX_CTX_NUM          (32)
```

Description

This constant indicates the maximum number of contexts that can exist at one time in the NP Lookup library.

See Also

```
sceNpLookupCreateTitleCtx()
```


SCE CONFIDENTIAL

SCE_NP_LOOKUP_MAX_REQUEST_NUM

Maximum number of requests

Definition

```
#include <np.h>

#define SCE_NP_LOOKUP_MAX_REQUEST_NUM      (32)
```

Description

This constant indicates the maximum number of requests that can exist at one time in the NP Lookup library.

See Also

```
sceNpLookupCreateRequest()
```

SCE CONFIDENTIAL

SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT

System default values

Definition

```
#include <np.h>

#define SCE_NP_LOOKUP_TIMEOUT_NO_EFFECT    (0)
```

Description

This is a constant set to the arguments for which one wishes to use system default values among the arguments of `sceNpLookupSetTimeout()` of the NP Lookup library.

See Also

`sceNpLookupSetTimeout()`