NP WordFilter Library Reference

© 2011 Sony Computer Entertainment Inc.
All Rights Reserved.
SCE Confidential

Table of Contents

- 3 -

Library Initialization/Termination

sceNpWordFilterInit

Initialize the NP WordFilter library

Definition

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

Arguments

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

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 (*option* is a value other than NULL, etc.)
SCE_NP_COMMUNITY_ERROR _NO_RESOURCE	0x80550716	Could not allocate system resources

Description

This function initializes the NP WordFilter library. This function must be called before using the NP WordFilter 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 = sceNpWordFilterInit(
            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 = sceNpWordFilterInit(
            SCE_TRUE,
            SCE_KERNEL_DEFAULT_PRIORITY_USER,
            SCE_KERNEL_THREAD_CPU_AFFINITY_MASK_DEFAULT,
            NULL);
If (ret < 0){
      // Error handling
}
```

See Also

```
sceNpWordFilterTerm()
```

SCE CONFIDENTIAL

sceNpWordFilterTerm

Terminate the NP WordFilter library

Definition

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

Arguments

None

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 WordFilter library.
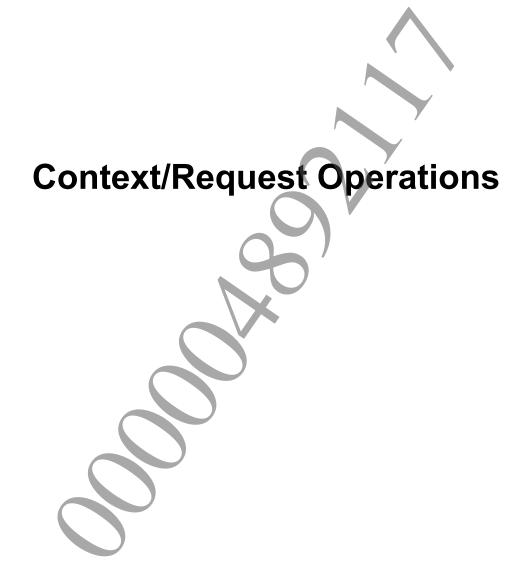
Notes

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

```
sceNpWordFilterInit()
```

SCE CONFIDENTIAL

©SCEI

Context/Request Operations

sceNpWordFilterCreateTitleCtx

Create an NP WordFilter title context

Definition

```
#include <np.h>
int sceNpWordFilterCreateTitleCtx(
        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 WordFilter title context upon normal termination.

Returns a negative value upon errors. 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 sceNpWordFilterDeleteTitleCtx() was called as necessary.)
SCE_NP_COMMUNITY_ERROR_ INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to *selfNpId*

Description

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

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

When sceNpWordFilterTerm() is called, all title contexts will be deleted automatically. However, try to call sceNpWordFilterTerm() after sceNpWordFilterDeleteTitleCtx() 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.

When multiple NP Communication IDs are used by a single title, select and set any one.

See Also

sceNpWordFilterDeleteTitleCtx(),SCE_NP_WORD_FILTER_MAX_CTX_NUM

SCE CONFIDENTIAL

sceNpWordFilterDeleteTitleCtx

Destroy the NP WordFilter title context

Definition

```
#include <np.h>
int sceNpWordFilterDeleteTitleCtx(
        SceInt32 titleCtxId
);
```

Arguments

titleCtxId   ID of the NP WordFilter title context to be destroyed

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *titleCtxId* does not exist

Description

This function destroys the NP WordFilter title context used by the NP WordFilter library.

See Also

sceNpWordFilterCreateTitleCtx()

sceNpWordFilterCreateRequest

Create an NP WordFilter request

Definition

```
#include <np.h>
int sceNpWordFilterCreateRequest(
        SceInt32 titleCtxId
);
```

Arguments

titleCtxId   NP WordFilter title context ID

Return Values

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

Returns a negative value upon errors. 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 sceNpWordFilterDeleteRequest() was called as necessary.)
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for *titleCtxId* does not exist

Description

This function creates a request of the NP WordFilter library.

A request ID is an ID that is used up for each communication processing function. A request ID must always be created before using these functions. Moreover, a used up request must be deleted with sceNpWordFilterDeleteRequest().

Up to 32 requests can exist simultaneously.

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

See Also

sceNpWordFilterDeleteRequest(),SCE_NP_WORD_FILTER_MAX_REQUEST_NUM

SCE CONFIDENTIAL

sceNpWordFilterDeleteRequest

Destroy the NP WordFilter request

Definition

```
#include <np.h>
int sceNpWordFilterDeleteRequest(
        SceInt32 reqId
);
```

Arguments

reqId   NP WordFilter request ID

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *reqId* does not exist

Description

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

See Also

sceNpWordFilterCreateRequest()

sceNpWordFilterSetTimeout

Set a timeout period for communication processing functions

Definition

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

Arguments

id	NP WordFilter title context ID or NP WordFilter request ID
resolveRetry	Name resolution retry times
resolveTimeout	Name resolution timeout time (microseconds)
connTimeout	Timeout time when connecting (microseconds)
sendTimeout	Sending timeout time (microseconds)
recvTimeout	Receiving timeout time (microseconds)

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *resolveRetry*, *resolveTimeout*, *connTimeout*, *sendTimeout* or *recvTimeout*
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	Title context or request specified for *id* does not exist

Description

This function sets the timeout period for communication processing.

For *id*, specify an NP WordFilter title context ID if you want to set a uniform timeout period, or specify an NP WordFilter 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 requests that you subsequently create after this function. It will not, however, be applied to already-existing transaction requests.

Specify the timeout period in microseconds.

SCE CONFIDENTIAL

Only the following values can be set for each value.

Argument	Settable values
resolveRetry	1 or more, or SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT
resolveTimeout	1 second or more, or SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT
connTimeout	10 seconds or more, or SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT
sendTimeout	10 seconds or more, or SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT
recvTimeout	10 seconds or more, or SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT

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_WORD_FILTER_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_WORD_FILTER_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 sceNpWordFilterAbortRequest().

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 WordFilter library ends 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

sceNpWordFilterCreateTitleCtx(),sceNpWordFilterCreateRequest(),
sceNpWordFilterAbortRequest(),SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT

sceNpWordFilterAbortRequest

Abort communication processing

Definition

```
#include <np.h>
int sceNpWordFilterAbortRequest(
        SceInt32 reqId
);
```

Arguments

reqId   NP WordFilter request ID

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *reqId* 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

sceNpWordFilterCreateRequest()

SCE CONFIDENTIAL

sceNpWordFilterWaitAsync,
sceNpWordFilterPollAsync

Obtain the execution result for an asynchronous request

Definition

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

Arguments

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

Return Values

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

Returns a negative value upon errors. 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 *reqId* does not exist

Description

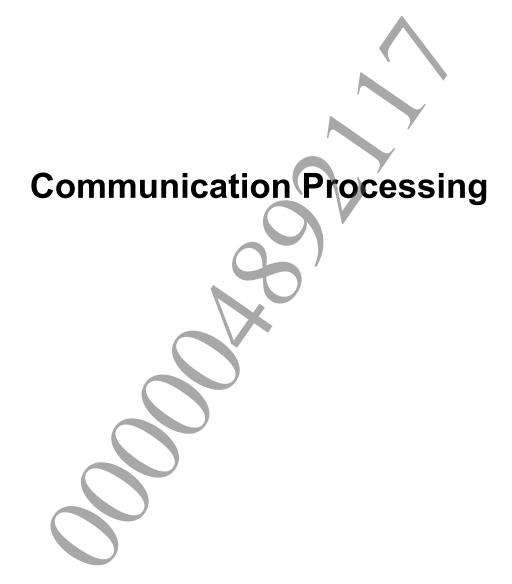This function obtains the execution result of a request executed asynchronously.

For *reqId*, specify the ID of the request started by the asynchronous function, sceNpWordFilter*XXX*Async().

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

sceNpWordFilterPollAsync() 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

```
sceNpWordFilterCreateRequest()
```

Communication Processing

sceNpWordFilterCensorComment

Check for inappropriate character strings (synchronous)

Definition

```
#include <np.h>
int sceNpWordFilterCensorComment(
        SceInt32 reqId,
        const char *comment,
        void *option
);
```

Arguments

reqId   NP WordFilter request ID (IN)
comment Pointer to UTF-8 character string to check; includes the NULL termination character. The maximum byte count, including the NULL termination character, is 1025 bytes (IN)
option   Option reserved for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *option*
SCE_NP_COMMUNITY_ERROR_ ABORTED	0x80550707	Request was aborted with sceNpWordFilterAbortRequest() or sceNpWordFilterDeleteRequest()
SCE_NP_COMMUNITY_ERROR_ INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to *comment*
SCE_NP_COMMUNITY_ERROR_ INVALID_ID	0x8055070e	ID specified for *reqId* 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_CENSORED	0x8055080b	String specified to *comment* is inappropriate. Occurs when inappropriate words or a string in the form of a telephone number is included.

Description

This function checks whether inappropriate strings are included. If an inappropriate string is included, SCE_NP_COMMUNITY_SERVER_ERROR_CENSORED is returned. If no inappropriate string is included, SCE_OK(0) is returned.

This function performs synchronous processing. It blocks until communication completes and the result can be obtained. Once returning from this function, destroy the used request with sceNpWordFilterDeleteRequest().
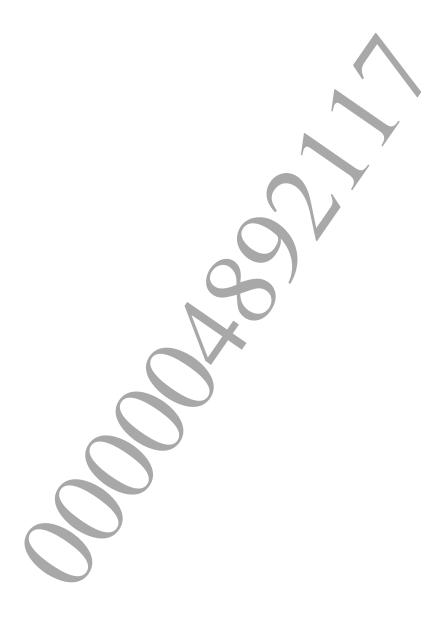
Notes

It is not necessary to check strings used in chatting, for example. Use this function to check strings that will be visible to all players and will remain visible over a long time. Note that the rules regarding inappropriate words are subject to change without notice.

See Also

```
sceNpWordFilterCreateRequest(),sceNpWordFilterDeleteRequest(),
sceNpWordFilterAbortRequest(),SCE_NP_WORD_FILTER_CENSOR_COMMENT_MAXLEN
```

©SCEI

sceNpWordFilterCensorCommentAsync

Check for inappropriate character strings (asynchronous)

Definition

```
#include <np.h>
int sceNpWordFilterCensorCommentAsync(
        SceInt32 reqId,
        const char *comment,
        void *option
);
```

Arguments

reqId	NP WordFilter request ID (IN)
comment	Pointer to UTF-8 character string to check; includes the NULL termination character. The maximum byte count, including the NULL termination character, is 1025 bytes (IN)
option	Option reserved for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *option*
SCE_NP_COMMUNITY_ERROR_ ABORTED	0x80550707	Request was aborted with sceNpWordFilterAbortRequest() or sceNpWordFilterDeleteRequest()
SCE_NP_COMMUNITY_ERROR_ INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to *comment*
SCE_NP_COMMUNITY_ERROR_ INVALID_ID	0x8055070e	ID specified for *reqId* 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_CENSORED	0x8055080b	String specified to *comment* is inappropriate. Occurs when inappropriate words or a string in the form of a telephone number is included.

Description

This function checks whether inappropriate strings are included. If an inappropriate string is included, SCE_NP_COMMUNITY_SERVER_ERROR_CENSORED is returned. If no inappropriate string is included, SCE_OK(0) is returned.

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 sceNpWordFilterWaitAsync() or sceNpWordFilterPollAsync(). Once the result is received using one of these functions, destroy the request.

©SCEI

SCE CONFIDENTIAL

Notes

It is not necessary to check strings used in chatting, for example. Use this function to check strings that will be visible to all players and will remain visible over a long time. Note that the rules regarding inappropriate words are subject to change without notice.

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

See Also

```
sceNpWordFilterCreateRequest(),sceNpWordFilterDeleteRequest(),
sceNpWordFilterAbortRequest(),sceNpWordFilterWaitAsync(),
sceNpWordFilterPollAsync(),SCE_NP_WORD_FILTER_CENSOR_COMMENT_MAXLEN
```

©SCEI

---

sceNpWordFilterSanitizeComment

Correct inappropriate character string (synchronous)

Definition

```
#include <np.h>
int sceNpWordFilterSanitizeComment(
        SceInt32 reqId,
        const char *comment,
        char *sanitizedComment,
        void *option
);
```

Arguments

reqId	NP WordFilter request ID (IN)
comment	Pointer to UTF-8 character string to correct; includes the NULL termination character. The maximum byte count, including the NULL termination character, is 1025 bytes (IN)
sanitizedComment	Address of memory for storing the UTF-8 character string after correction. Prepare a size of strlen(*comment*)+1 (OUT)
option	Option reserved for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *option*
SCE_NP_COMMUNITY_ERROR_ ABORTED	0x80550707	Request was aborted with sceNpWordFilterAbortRequest() or sceNpWordFilterDeleteRequest()
SCE_NP_COMMUNITY_ERROR_ INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to *comment* or *sanitizedComment*
SCE_NP_COMMUNITY_ERROR_ INVALID_ID	0x8055070e	ID specified for *reqId* 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

While sceNpWordFilterCensorComment() returns SCE_NP_COMMUNITY_SERVER_ERROR_CENSORED upon identifying an inappropriate comment, this function converts the inappropriate characters to '*' and stores the converted string in *sanitizedComment*. Even if an inappropriate character string is included, the processing returns 0. To evaluate whether an inappropriate character string was included or not can only be confirmed by comparing whether the character string specified in *comment* is the same as what is stored in *sanitizedComment*.

Because the multi-byte character of UTF-8 can be converted to '*'(0x2a), the number of bytes for the UTF-8 character string stored in *sanitizedComment* can become smaller than the byte size of the UTF-8 character string specified in *comment*.

The function is blocked until communication completes and scoreboard information is obtained. Once returning from this function, destroy the used request with sceNpWordFilterDeleteRequest().

Notes

It is not necessary to check strings used in chatting, for example. Use this function to check strings that will be visible to all players and will remain visible over a long time. Note that the rules regarding inappropriate words are subject to change without notice.

See Also

sceNpWordFilterCreateRequest(),sceNpWordFilterDeleteRequest(),
sceNpWordFilterAbortRequest(),SCE_NP_WORD_FILTER_SANITIZE_COMMENT_MAXLEN

sceNpWordFilterSanitizeCommentAsync

Correct inappropriate character string (asynchronous)

Definition

```
#include <np.h>
int sceNpWordFilterSanitizeCommentAsync(
        SceInt32 reqId,
        const char *comment,
        char *sanitizedComment,
        void *option
);
```

Arguments

reqId	NP WordFilter request ID (IN)
comment	Pointer to UTF-8 character string to correct; includes the NULL termination character. The maximum byte count, including the NULL termination character, is 1025 bytes (IN)
sanitizedComment	Address of memory for storing the UTF-8 character string after correction. Prepare a size of `strlen(comment)`+1 (OUT)
option	Option reserved for future extension. Always specify NULL

Return Values

Returns 0 upon normal termination.

Returns a negative value upon errors. 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 *option*
SCE_NP_COMMUNITY_ERROR_ABORTED	0x80550707	Request was aborted with `sceNpWordFilterAbortRequest()` or `sceNpWordFilterDeleteRequest()`
SCE_NP_COMMUNITY_ERROR_INSUFFICIENT_ARGUMENT	0x8055070c	NULL was specified to *comment* or *sanitizedComment*
SCE_NP_COMMUNITY_ERROR_INVALID_ID	0x8055070e	ID specified for *reqId* 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

While `sceNpWordFilterCensorComment()` returns SCE_NP_COMMUNITY_SERVER_ERROR_CENSORED upon identifying an inappropriate comment, this function converts the inappropriate characters to '*' and stores the converted string in *sanitizedComment*. Even if an inappropriate character string is included, the processing returns 0. To evaluate whether an inappropriate character string was included or not can only be confirmed by comparing whether the character string specified in *comment* is the same as what is stored in *sanitizedComment*.

Because the multi-byte character of UTF-8 can be converted to '*'(0x2a), the number of bytes for the UTF-8 character string stored in *sanitizedComment* can become smaller than the byte size of the UTF-8 character string specified in *comment*.

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 `sceNpWordFilterWaitAsync()` or `sceNpWordFilterPollAsync()`. Once the result is received using one of these functions, destroy the request.
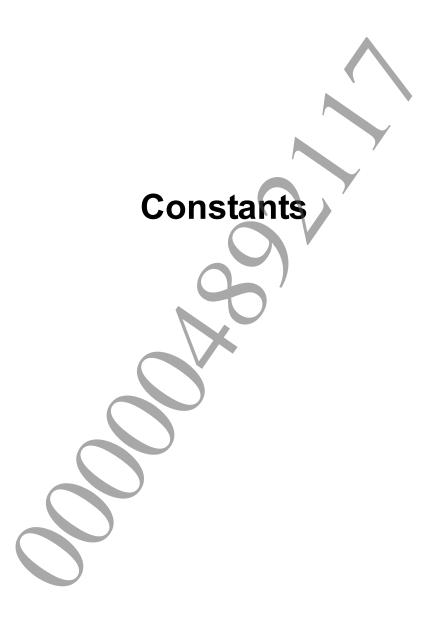
Notes

It is not necessary to check strings used in chatting, for example. Use this function to check strings that will be visible to all players and will remain visible over a long time. Note that the rules regarding inappropriate words are subject to change without notice.

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

See Also

```
sceNpWordFilterCreateRequest(),sceNpWordFilterDeleteRequest(),
sceNpWordFilterAbortRequest(),sceNpWordFilterWaitAsync(),
sceNpWordFilterPollAsync(),SCE_NP_WORD_FILTER_SANITIZE_COMMENT_MAXLEN
```

©SCEI

Constants

SCE_NP_WORD_FILTER_MAX_CTX_NUM

Maximum number of contexts

Definition

```
#include <np.h>
#define SCE_NP_WORD_FILTER_MAX_CTX_NUM    (32)
```

Description

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

See Also

```
sceNpWordFilterCreateTitleCtx()
```

SCE CONFIDENTIAL

SCE_NP_WORD_FILTER_MAX_REQUEST_NUM

Maximum number of requests

Definition

```
#include <np.h>
#define SCE_NP_WORD_FILTER_MAX_REQUEST_NUM (32)
```

Description

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

See Also

```
sceNpWordFilterCreateRequest()
```

SCE_NP_WORD_FILTER_CENSOR_COMMENT_MAXLEN

Maximum length for a comment to be checked

Definition

```
#include <np.h>
#define SCE_NP_WORD_FILTER_CENSOR_COMMENT_MAXLEN  (1024)
```

Description

This constant represents the maximum length of the character string part of a comment that can be checked by sceNpWordFilterCensorComment() and sceNpWordFilterCensorCommentAsync() of the NP WordFilter library. Because comments are set in UTF-8, length is represented in bytes rather than the length of character strings.

The above length is the byte length not including the NULL termination character. The maximum byte count of the comment, including the NULL termination character, is SCE_NP_WORD_FILTER_CENSOR_COMMENT_MAXLEN + 1.

See Also

sceNpWordFilterCensorComment(), sceNpWordFilterCensorCommentAsync()

SCE CONFIDENTIAL

SCE_NP_WORD_FILTER_SANITIZE_COMMENT_MA XLEN

Maximum length for a comment to be corrected

Definition

```
#include <np.h>
#define SCE_NP_WORD_FILTER_SANITIZE_COMMENT_MAXLEN        (1024)
```

Description

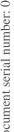This constant represents the maximum length of the character string part of a comment that can be corrected by sceNpWordFilterSanitizeComment() and sceNpWordFilterSanitizeCommentAsync() of the NP WordFilter library. Because comments are set in UTF-8, length is represented in bytes rather than the length of character strings.

The above length is the byte length not including the NULL termination character. The maximum byte count of the comment, including the NULL termination character, is SCE_NP_WORD_FILTER_SANITIZE_COMMENT_MAXLEN + 1.

See Also

sceNpWordFilterSanitizeComment(), sceNpWordFilterSanitizeCommentAsync()

SCE_NP_WORD_FILTER_TIMEOUT_NO_EFFECT

System default values

Definition

```
#include <np.h>
#define SCE_NP_WORD_FILTER_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 sceNpWordFilterSetTimeout() of the NP WordFilter library.

See Also

sceNpWordFilterSetTimeout()