libime Reference

PlayStation®Vita Programmer Tool Runtime Library Release 3.50

Table of Contents

Constants

Character String Size

Character string size used with libime

Definition

Value	Decimal	Description
`SCE_IME_MAX_PREEDIT_LENGTH`	30	Maximum preedit length.
`SCE_IME_MAX_TEXT_LENGTH`	2048	Maximum length of the character string that can be specified in `SceImeParam.`*`maxTextLength`*.

Description

Here is defined the maximum character string length that can be used with libime.

`SCE_IME_MAX_PREEDIT_LENGTH` is the maximum preedit length.

`SCE_IME_MAX_TEXT_LENGTH` is the maximum value that can be specified in `SceImeParam.`*`maxTextLength`*.

Return Codes

List of return codes returned by libime

Definition

Value	Hexadecimal	Description
`SCE_IME_ERROR_ALREADY_OPENED`	0x80100700	libime has already been started.
`SCE_IME_ERROR_NOT_OPENED`	0x80100701	libime has not been started yet.
`SCE_IME_ERROR_INVALID_POINTER`	0x80100702	An invalid pointer has been specified.
`SCE_IME_ERROR_INVALID_PARAM`	0x80100703	An invalid parameter has been specified.
`SCE_IME_ERROR_NO_MEMORY`	0x80100704	Memory is insufficient.
`SCE_IME_ERROR_CONNECTION_FAILED`	0x80100705	Connection with IME has failed.
`SCE_IME_ERROR_INVALID_TEXT`	0x80100706	Content of text is invalid.
`SCE_IME_ERROR_TOO_MANY_REQUESTS`	0x80100707	Too many requests for IME.
`SCE_IME_ERROR_INVALID_INPUT_METHOD`	0x80100712	*inputMethod* value is invalid.
`SCE_IME_ERROR_INVALID_SUPPORTED_LANGUAGES`	0x80100713	*supportedLanguages* value is invalid.
`SCE_IME_ERROR_INVALID_TYPE`	0x80100714	*type* value is invalid.
`SCE_IME_ERROR_INVALID_OPTION`	0x80100715	*option* value is invalid.
`SCE_IME_ERROR_INVALID_WORK`	0x80100716	*work* value is invalid.
`SCE_IME_ERROR_INVALID_ARG`	0x80100717	*arg* value is invalid.
`SCE_IME_ERROR_INVALID_HANDLER`	0x80100718	*handler* value is invalid.
`SCE_IME_ERROR_INVALID_MAX_TEXT_LENGTH`	0x80100719	*maxTextLength* value is invalid.
`SCE_IME_ERROR_INVALID_INPUT_TEXT_BUFFER`	0x8010071a	*inputTextBuffer* value is invalid.
`SCE_IME_ERROR_INVALID_RESERVED`	0x8010071b	*reserved* value is invalid.
`SCE_IME_ERROR_INVALID_ENTER_LABEL`	0x8010071c	*enterLabel* value is invalid.
`SCE_IME_ERROR_INTERNAL`	0x80100750	An internal error has occurred.

Datatypes

SceImeParam

Initialization parameters

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 sdkVersion;
        SceUInt32 inputMethod;
        SceUInt64 supportedLanguages;
        SceBool languagesForced;
        SceUInt32 type;
        SceUInt32 option;
        void *work;
        void *arg;
        SceImeEventHandler handler;
        SceImeTextFilter filter;
        SceWChar16 *initialText;
        SceUInt32 maxTextLength;
        SceWChar16 *inputTextBuffer;
        SceUChar8 enterLabel;
        SceUChar8 reserved[7];
} SceImeParam;
```

Members

sdkVersion	SDK version
inputMethod	Input method
supportedLanguages	Supported languages
languagesForced	Whether to set supported languages forcedly
type	Input type
option	Input options
work	Work buffer
arg	`void` pointer passed to event handler
handler	IME event handler
filter	Character strings filter
initialText	Initial character string
maxTextLength	Maximum length of result character strings
inputTextBuffer	Storage location of edited character string
enterLabel	Enter button label
reserved	Reserved area (fill with all 0s)

Description

This structure is used for setting parameters when starting libime with `sceImeOpen()`. Before using it, perform initialization with `sceImeParamInit()`.

Since currently there is nothing to be specified in *inputMethod*, set to 0.

supportedLanguages is either the logical OR of the following values or 0 indicating the use of all languages.

Value	Hexadecimal	Description
`SCE_IME_LANGUAGE_DANISH`	0x00000001ULL	Danish
`SCE_IME_LANGUAGE_GERMAN`	0x00000002ULL	German
`SCE_IME_LANGUAGE_ENGLISH_US`	0x00000004ULL	English (United States)
`SCE_IME_LANGUAGE_SPANISH`	0x00000008ULL	Spanish
`SCE_IME_LANGUAGE_FRENCH`	0x00000010ULL	French
`SCE_IME_LANGUAGE_ITALIAN`	0x00000020ULL	Italian
`SCE_IME_LANGUAGE_DUTCH`	0x00000040ULL	Dutch
`SCE_IME_LANGUAGE_NORWEGIAN`	0x00000080ULL	Norwegian
`SCE_IME_LANGUAGE_POLISH`	0x00000100ULL	Polish
`SCE_IME_LANGUAGE_PORTUGUESE_PT`	0x00000200ULL	Portuguese (Portugal)
`SCE_IME_LANGUAGE_RUSSIAN`	0x00000400ULL	Russian
`SCE_IME_LANGUAGE_FINNISH`	0x00000800ULL	Finnish
`SCE_IME_LANGUAGE_SWEDISH`	0x00001000ULL	Swedish
`SCE_IME_LANGUAGE_JAPANESE`	0x00002000ULL	Japanese
`SCE_IME_LANGUAGE_KOREAN`	0x00004000ULL	Korean
`SCE_IME_LANGUAGE_SIMPLIFIED_CHINESE`	0x00008000ULL	Simplified Chinese
`SCE_IME_LANGUAGE_TRADITIONAL_CHINESE`	0x00010000ULL	Traditional Chinese
`SCE_IME_LANGUAGE_PORTUGUESE_BR`	0x00020000ULL	Portuguese (Brazil)
`SCE_IME_LANGUAGE_ENGLISH_GB`	0x00040000ULL	English (United Kingdom)
`SCE_IME_LANGUAGE_TURKISH`	0x00080000ULL	Turkish

If you wish to switch among all languages specified in *supportedLanguages*, set *languagesForced* to `SCE_TRUE`, and set to `SCE_FALSE` if you wish to input with the languages in the set intersection of the languages specified in *supportedLanguages* and the languages specified by the user.

type is set to one of the following values.

Value	Decimal	Description
`SCE_IME_TYPE_DEFAULT`	0	UI for regular text input
`SCE_IME_TYPE_BASIC_LATIN`	1	UI for alphanumeric character input
`SCE_IME_TYPE_NUMBER`	2	UI for number input
`SCE_IME_TYPE_EXTENDED_NUMBER`	3	UI for extended number input
`SCE_IME_TYPE_URL`	4	UI for entering URL
`SCE_IME_TYPE_MAIL`	5	UI for entering an email address

option is set to the logical OR of the following values. Set 0 when no values are used.

Value	Hexadecimal	Description
`SCE_IME_OPTION_MULTILINE`	0x01	Multiline input option. This option is not available for libime. This can be used only for the IME Dialog library.
`SCE_IME_OPTION_NO_AUTO_CAPITALIZATION`	0x02	Prohibits automatic capitalization.
`SCE_IME_OPTION_NO_ASSISTANCE`	0x04	Prohibits input assistance UIs, such as predictive text and conversion candidate.

Set the pointer to the work buffer used by libime in *work*. Work buffer size is `SCE_IME_WORK_BUFFER_SIZE`. The work buffer must be allocated during the operation of libime.

Set the function pointer for processing IME events in *handler*.

In *filter* set the filter function pointer used when filtering input character strings.

Set the initial character string in *initialText*.

Set the maximum result character string length in *maxTextLength*. 0 at the end of the character string is not included in this length.

Set the pointer to the buffer for storing edited character strings in *inputTextBuffer*. Buffer length must be SCE_IME_MAX_PREEDIT_LENGTH + *maxTextLength* + 1. After the completion of the edit operation, the result character string is stored.

enterLabel is set to one of the following values.

Value	Decimal	Description
SCE_IME_ENTER_LABEL_DEFAULT	0	Sets the Enter button label to the default.
SCE_IME_ENTER_LABEL_SEND	1	Sets the Enter button label to "Send".
SCE_IME_ENTER_LABEL_SEARCH	2	Sets the Enter button label to "Search".
SCE_IME_ENTER_LABEL_GO	3	Sets the Enter button label to "Go".

reserved is a reserved area. Fill it with all 0s.

See Also

sceImeOpen()

SceImeRect

Information on the rectangle occupied by IME

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 x;
        SceUInt32 y;
        SceUInt32 width;
        SceUInt32 height;
} SceImeRect;
```

Members

x	Rectangle upper-left X coordinate
y	Rectangle upper-left Y coordinate
width	Rectangle width
height	Rectangle height

Description

This structure stores information on the area (rectangle) occupied by IME.

See Also

`sceImeUpdate()`,`SceImeEventParam`

SceImeEditText

Edited text information

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 preeditIndex;
        SceUInt32 preeditLength;
        SceUInt32 caretIndex;
        SceWChar16 *str;
        SceUInt32 editIndex;
        SceInt32 editLengthChange;
} SceImeEditText;
```

Members

preeditIndex	Preedit starting position
preeditLength	Preedit length
caretIndex	Caret position
str	Edited character string (0 at the end)
editIndex	Position of edited character string in the text
editLengthChange	Changes in the number of characters made through the edit operation

Description

This is the information on the text currently being edited, notified by IME event.

See Also

`sceImeUpdate(),SceImeEvent,SceImeEventParam`

SceImeEventParam

IME event parameters

Definition

```
#include <libime.h>
typedef union
{
        SceImeRect rect;
        SceImeEditText text;
        SceUInt32 caretIndex;
        SceUChar8 reserved[40];
} SceImeEventParam;
```

Members

rect	Rectangle information
text	Edited text information
caretIndex	Caret position
reserved	Reserved area

Description

This union stores information for each IME event. The union member used varies depending on event ID. Refer to the following document for what members are used for each event:

- "IME Event List" in the "IME Overview" document

See Also

`sceImeUpdate()`

SceImeEvent

IME events

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 id;
        SceImeEventParam param;
} SceImeEvent;
```

Members

id	Event ID
param	Event parameter

Description

This is the event information passed as handler argument when an IME event is issued. Perform the necessary processing based on *id* and *param*.

See Also

`sceImeUpdate()`

SceImeCaret

Caret information

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 x;
        SceUInt32 y;
        SceUInt32 height;
        SceUInt32 index;
} SceImeCaret;
```

Members

x	x coordinate
y	y coordinate
height	Height
index	Caret position

Description

This is the caret information notified to IME when the caret has moved. Set the caret's upper-left coordinates in *x* and *y*.

See Also

`sceImeSetCaret()`

SceImePreeditGeometry

Preedit geometrical information

Definition

```
#include <libime.h>
typedef struct
{
        SceUInt32 x;
        SceUInt32 y;
        SceUInt32 height;
} SceImePreeditGeometry;
```

Members

x	x coordinate
y	y coordinate
height	Height

Description

This is the geometrical information notified to IME when preedit position is changed. Set preedit upper-left coordinates in *x* and *y*.

See Also

`sceImeSetPreeditGeometry()`

Callback Functions

SceImeTextFilter

Character strings filter

Definition

```
#include <libime.h>
typedef SceInt32 (*SceImeTextFilter)(
        SceWChar16 *outText,
        SceUInt32 *outTextLength,
        const SceWChar16 *srcText,
        SceUInt32 srcTextLength
);
```

Arguments

outText	Filtered character string
outTextLength	Length of filtered character string
srcText	Input character string
srcTextLength	Length of input character string

Return Values

Input character strings are passed with *srcText*. Return 0 if directly using *srcText*, and 1 if using the filtered character string *outText*.

Description

This is the character strings filter set in SceImeParam.*filter*. It is called each time a character string is inputted by the user. Filter the character string *srcText* input by the user as needed and then store the filtered character string and the length of the character string into *outText* and *outTextLength* respectively. *outTextLength* stores the length of the character string that can be stored in *outText*. Since input character strings are called after preedit has been confirmed, preedit contents cannot be filtered.

See Also

SceImeParam

SceImeEventHandler

IME event handler

Definition

```
#include <libime.h>
typedef void (*SceImeEventHandler)(void* arg, const SceImeEvent *e);
```

Arguments

arg	*arg* set in SceImeParam
e	Event information

Return Values

None

Description

This is the IME event handler called each time an IME event is issued. Perform appropriate processing based on event content.

See Also

SceImeParam

Functions

sceImeParamInit

Parameter initialization

Definition

```
#include <libime.h>
void sceImeParamInit(
        SceImeParam *param
)
```

Arguments

param Initialization parameter

Return Values

None

Description

This function initializes libime starting parameters.

Before performing the various call parameter settings, always perform structure initialization by using this macro.

Examples

```
SceImeParam param;

sceImeParamInit( &param );
param.maxTextLength = MAX_TEXT_LENGTH;
param.initialText = initialText;
param.inputTextBuffer = inputTextBuffer;
if ( sceImeOpen( &param ) < 0 ) {
        // error
}
```

See Also

`SceImeParam`, `sceImeOpen()`

sceImeOpen

libime initialization

Definition

```
#include <libime.h>
SceInt32 sceImeOpen(
        SceImeParam *param
)
```

Arguments

param Initialization parameter

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
SCE_IME_ERROR_ALREADY_OPENED	0x80100700	libime has already been started.
SCE_IME_ERROR_INVALID_PARAM	0x80100703	An invalid parameter has been specified.
SCE_IME_ERROR_NO_MEMORY	0x80100704	Memory is insufficient.
SCE_IME_ERROR_CONNECTION_FAILED	0x80100705	Connection with IME has failed.
SCE_IME_ERROR_INVALID_TEXT	0x80100706	Content of text is invalid.
SCE_IME_ERROR_INVALID_INPUT_METHOD	0x80100712	*inputMethod* value is invalid.
SCE_IME_ERROR_INVALID_SUPPORTED_LANGUAGES	0x80100713	*supportedLanguages* value is invalid.
SCE_IME_ERROR_INVALID_TYPE	0x80100714	*type* value is invalid.
SCE_IME_ERROR_INVALID_OPTION	0x80100715	*option* value is invalid.
SCE_IME_ERROR_INVALID_WORK	0x80100716	*work* value is invalid.
SCE_IME_ERROR_INVALID_ARG	0x80100717	*arg* value is invalid.
SCE_IME_ERROR_INVALID_HANDLER	0x80100718	*handler* value is invalid.
SCE_IME_ERROR_INVALID_MAX_TEXT_LENGTH	0x80100719	*maxTextLength* value is invalid.
SCE_IME_ERROR_INVALID_INPUT_TEXT_BUFFER	0x8010071a	*inputTextBuffer* value is invalid.
SCE_IME_ERROR_INVALID_RESERVED	0x8010071b	*reserved* value is invalid.
SCE_IME_ERROR_INTERNAL	0x80100750	An internal error has occurred.

Description

This is a function for initializing libime.

In *`param`*, specify the initialization parameter structure setting operation mode, initial values, buffer for storing character strings etc.

Always initialize *`param`* with the `sceImeParamInit()` macro before setting the various values.

It is not necessary to allocate *`param`*'s instance after calling this function, but some pointer reference parameters must be saved until termination with `sceImeClose()`.

This function is multithread safe.

Examples

```
SceImeParam param;

sceImeParamInit( &param );
param.maxTextLength = MAX_TEXT_LENGTH;
param.initialText = initialText;
param.inputTextBuffer = inputTextBuffer;
if ( sceImeOpen( &param ) < 0 ) {
        // error
}
```

See Also

`SceImeParam`, `sceImeParamInit()`

sceImeUpdate

Updating libime

Definition

```
#include <libime.h>
SceInt32 sceImeUpdate()
```

Arguments

None

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
`SCE_IME_ERROR_NOT_OPENED`	0x80100701	libime has not been started yet.
`SCE_IME_ERROR_CONNECTION_FAILED`	0x80100705	Connection with IME has failed.
`SCE_IME_ERROR_INTERNAL`	0x80100750	An internal error has occurred.

Description

This is a function for updating libime.

It updates libime and, when an IME event is present, notifies the IME event to the handler specified in `SceImeParam.`*`handler`*.

This function is multithread safe.

See Also

`sceImeOpen(),sceImeClose()`

sceImeSetCaret

Notification of caret information to IME

Definition

```
#include <libime.h>
SceInt32 sceImeSetCaret(
        const SceImeCaret *caret
)
```

Arguments

caret Caret information

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
SCE_IME_ERROR_NOT_OPENED	0x80100701	libime has not been started yet.
SCE_IME_ERROR_INVALID_PARAM	0x80100703	An invalid parameter has been specified.
SCE_IME_ERROR_TOO_MANY_REQUESTS	0x80100707	Too many requests for IME.
SCE_IME_ERROR_INTERNAL	0x80100750	An internal error has occurred.

Description

This is a function for notifying caret information to IME.

If the caret has been moved by an IME event or user operation, caret position must be notified to IME by using this function.

This function is multithread safe.

See Also

`SceImeCaret`

sceImeSetPreeditGeometry

Notification of preedit geometrical information to IME

Definition

```
#include <libime.h>
SceInt32 sceImeSetPreeditGeometry(
        const SceImePreeditGeometry *preedit
)
```

Arguments

preedit Preedit geometrical information

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
SCE_IME_ERROR_INTERNAL	0x80100750	Undefined error not listed below.
SCE_IME_ERROR_NOT_OPENED	0x80100701	libime has not been started yet.
SCE_IME_ERROR_INVALID_PARAM	0x80000703	Parameter is invalid.
SCE_IME_ERROR_TOO_MANY_REQUESTS	0x80100707	Too many requests for IME.

Description

This is a function for notifying preedit geometrical information to IME.

If preedit geometrical information has changed because of an IME event, preedit geometrical information must be notified to IME by using this function.

This function is multithread safe.

See Also

`SceImePreeditGeometry`

sceImeSetText

Change text being edited

Definition

```
#include <libime.h>
SceInt32 sceImeSetText(
        const SceWChar16 *text,
        SceUInt32 length
)
```

Arguments

text	Text
length	Character string length of text

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
`SCE_IME_ERROR_NOT_OPENED`	0x80100701	libime has not been started yet.
`SCE_IME_ERROR_INVALID_POINTER`	0x80100702	NULL was specified for *text*.
`SCE_IME_ERROR_INVALID_PARAM`	0x80100703	An invalid parameter has been specified.
`SCE_IME_ERROR_INVALID_TEXT`	0x80100706	Content of text is invalid.
`SCE_IME_ERROR_TOO_MANY_REQUESTS`	0x80100707	Too many requests for IME.
`SCE_IME_ERROR_INTERNAL`	0x80100750	An internal error has occurred.

Description

This is a function for changing a text being edited.

Call this function when it is required to change the text being edited with IME through the user operation, etc. The caret will be positioned at the end of the text.

This function is multithread safe.

See Also

`sceImeOpen()`

sceImeClose

Termination of libime

Definition

```
#include <libime.h>
SceInt32 sceImeClose()
```

Arguments

None

Return Values

Returns `SCE_OK(0)` for success.

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

Value	Hexadecimal	Description
`SCE_IME_ERROR_INTERNAL`	0x80100750	An undefined error not listed below.
`SCE_IME_ERROR_NOT_OPENED`	0x80100701	libime has not been started yet.

Description

This is a function for terminating libime.

When the IME event `SCE_IME_EVENT_PRESS_CLOSE` is received, libime must be terminated immediately by calling `sceImeClose()`.

This function is multithread safe.

See Also

`sceImeOpen()`