

# positioning\_base\_library

## 目次 [Table of contents]

- 目次 [Table of contents]
- 図表目次 [Table of figures]
- positioning\_base\_library [positioning\_base\_library]
  - 機能概要 [Functional overview]
  - 機能詳細 [Function detail]
  - ソフトウェア構成図 [software block]
  - ユースケースとAPI一覧 [use-case and API lists]
    - 外部要因 ユースケース一覧 [outside factor use-case list]
    - 内部処理 エラーユースケース一覧 [internal processing error use-case list]
    - 初期化と終了シーケンス一覧 [start and end use-case list]
    - メッセージキューを生成する [positioning\_base\_library\_MessageAPI\_001]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 指定したPNOにメッセージを送信する [positioning\_base\_library\_MessageAPI\_002]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 指定したプロセスにメッセージを送信する [positioning\_base\_library\_MessageAPI\_003]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 指定したPNO向けのメッセージを受信する [positioning\_base\_library\_MessageAPI\_004]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - プロセス内通信用 (非コピーAPI用) 送信バッファ取得 [positioning\_base\_library\_MessageAPI\_005]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - プロセス内通信用 (非コピーAPI用) [positioning\_base\_library\_MessageAPI\_006]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - プロセス間で排他可能な名前付きミューテックスを作成する [positioning\_base\_library\_MutexAPI\_001]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - プロセス名をPNOに変換する [positioning\_base\_library\_NamePnoCnvAPI\_001]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - PNOをプロセス名に変換する [positioning\_base\_library\_NamePnoCnvAPI\_002]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース取得判定(メッセージ制御テーブル) [positioning\_base\_library\_ResourceAPI\_001]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース取得判定(mutex制御情報) [positioning\_base\_library\_ResourceAPI\_002]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース取得判定(プロセス名-PNO変換テーブル) [positioning\_base\_library\_ResourceAPI\_003]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース解放(メッセージ制御テーブル) [positioning\_base\_library\_ResourceAPI\_004]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース解放(mutex制御情報) [positioning\_base\_library\_ResourceAPI\_005]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - リソース解放(プロセス名-PNO変換テーブル) [positioning\_base\_library\_ResourceAPI\_006]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - セマフォを生成してセマフォIDを返却する [positioning\_base\_library\_SemaphoreAPI\_001]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 指定されたセマフォIDのセマフォを取得する [positioning\_base\_library\_SemaphoreAPI\_002]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 指定されたセマフォIDのセマフォを解放する [positioning\_base\_library\_SemaphoreAPI\_003]
      - 概要 [Overview]
      - シーケンス [Sequence]
    - 共有データ領域作成 [positioning\_base\_library\_ShareDataAPI\_001]
      - 概要 [Overview]

- シーケンス [Sequence]
- 共有データ領域リンク [positioning\_base\_library\_ShareDataAPI\_002]
  - 概要 [Overview]
  - シーケンス [Sequence]
- タイマ起動指示 [positioning\_base\_library\_TimerAPI\_001]
  - 概要 [Overview]
  - シーケンス [Sequence]
- タイマ停止指示 [positioning\_base\_library\_TimerAPI\_002]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 指定した名称でイベントを生成し、そのイベントIDを返却する [positioning\_base\_library\_EventAPI\_001]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 指定されたイベントIDのイベント値が指定された範囲になるまで待つ [positioning\_base\_library\_EventAPI\_002]
  - 概要 [Overview]
  - シーケンス [Sequence]
- イベント値を設定する [positioning\_base\_library\_EventAPI\_003]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 指定されたイベントIDのイベントを削除する [positioning\_base\_library\_EventAPI\_004]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(メッセージ) [positioning\_base\_library\_GetDbgAPI\_001]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(mutex) [positioning\_base\_library\_GetDbgAPI\_002]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(タイマ) [positioning\_base\_library\_GetDbgAPI\_003]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(イベント) [positioning\_base\_library\_GetDbgAPI\_004]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(memory) [positioning\_base\_library\_GetDbgAPI\_005]
  - 概要 [Overview]
  - シーケンス [Sequence]
- dump情報取得(other) [positioning\_base\_library\_GetDbgAPI\_006]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 自スレッドのアプリケーションハンドルを取得する [positioning\_base\_library\_HandleAPI\_001]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 自スレッドのアプリケーションハンドルを設定する [positioning\_base\_library\_HandleAPI\_002]
  - 概要 [Overview]
  - シーケンス [Sequence]
- positioning\_base\_libraryの初期化 [Initialize positioning\_base\_library]
  - 概要 [Overview]
  - シーケンス [Sequence]
- Base API終了処理 [Teardown Base API]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 該当スレッドを終了させる [Stop thread]
  - 概要 [Overview]
  - シーケンス [Sequence]
- 該当プロセスを終了させる [Stop process]
  - 概要 [Overview]
  - シーケンス [Sequence]

## 図表目次 *[Table of figures]*

- 表.ユニット概要
- 表. positioning\_base\_library 機能一覧

## positioning\_base\_library *[positioning\_base\_library]*

### 機能概要 *[Functional overview]*

## 表.ユニット概要

ユニット名[Unit Name]	コンポーネント名[Component Name]	ユニット概要[Description]	オーナーディレクトリ[Owner Dir]
positioning_base_library	VehicleService	positioning_base_libraryは positioning ベース機能を提供する。 <i>positioning_base_library provides the base function for positioning.</i>	positioning_base_library

## 機能詳細 [Function detail]

提供する機能の詳細を以下に示す。

*Function detail of this guide as follows.*

### 表. positioning\_base\_library 機能一覧

NO.	種類 [Kind]	機能 [Function]
1	MessageAPI	メッセージキューを生成する <i>Create the message queue</i>
2		指定したPNOにメッセージを送信する <i>Send the message based on PNO</i>
3		指定したプロセスにメッセージを送信する <i>Send the message based on process name</i>
4		指定したPNO向けのメッセージを受信する <i>Receive the message based on PNO</i>
5		プロセス内通信用（非コピーAPI用）送信バッファ取得 <i>Get message buffer inside process (not copy API)</i>
6		プロセス内通信用（非コピーAPI用） <i>Send message inside process (not copy API)</i>
7	MutexAPI	プロセス間で排他可能な名前付きミューテックスを作成する <i>Create mutex between process</i>
8	NamePnoCnvAPI	プロセス名をPNOに変換する <i>Change process name to PNO</i>
9		PNOをプロセス名に変換する <i>Change PNO to process name</i>
10	ResourceAPI	リソース取得判定(メッセージ制御テーブル) <i>Check resource(message control table)</i>
11		リソース取得判定(mutex制御情報) <i>Check resource(mutex control information)</i>

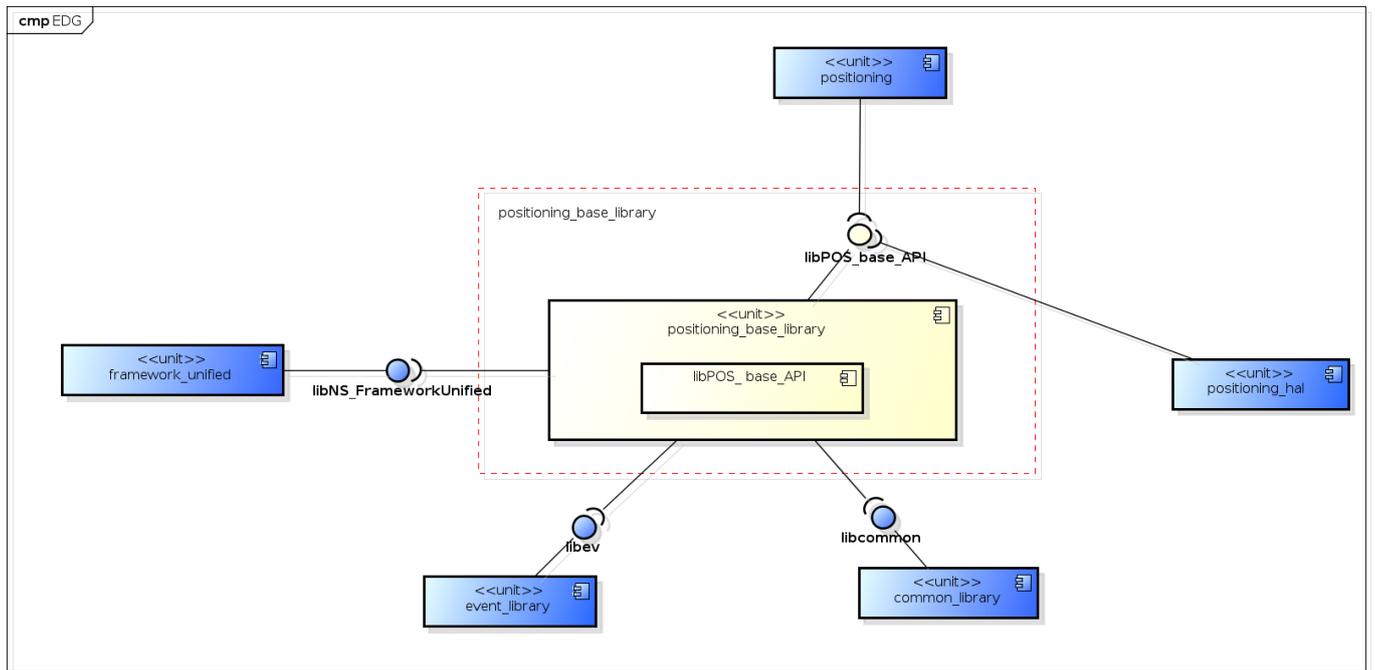
12		リソース取得判定(プロセス名-PNO変換テーブル) <i>Check resource(process name - PNO table)</i>
13		リソース解放(メッセージ制御テーブル) <i>Release resource(message control table)</i>
14		リソース解放(mutex制御情報) <i>Release resource(mutex control information)</i>
15		リソース解放(プロセス名-PNO変換テーブル) <i>Release resource(process name - PNO table)</i>
16	SemaphoreAPI	セマフォを生成してセマフォIDを返却する <i>Create semaphore and get semaphore ID</i>
17		指定されたセマフォIDのセマフォを取得する <i>Get semaphore according to semaphore ID</i>
18		指定されたセマフォIDのセマフォを解放する <i>Release semaphore according to semaphore ID</i>
19	ShareDataAPI	共有データ領域作成 <i>Create share data</i>
20		共有データ領域リンク <i>Link share data</i>
21	TimerAPI	タイマ起動指示 <i>Start timer</i>
22		タイマ停止指示 <i>Stop timer</i>
23	EventAPI	指定した名称でイベントを生成し、そのイベントIDを返却する <i>Create event according to event name, and get event ID</i>
24		指定されたイベントIDのイベント値が指定された範囲になるまで待つ <i>Wait for the event according to event ID and event value</i>
25		イベント値を設定する <i>Set the event</i>
26		指定されたイベントIDのイベントを削除する <i>Delete event according to event ID</i>
27	GetDbgAPI	dump情報取得 <i>Get dump information</i>

28	HandleAPI	自スレッドのアプリケーションハンドルを取得する <i>Get application handle</i>
29		自スレッドのアプリケーションハンドルを設定する <i>Set application handle</i>

## ソフトウェア構成図 [software block]

以下にソフトウェア構成を示す。

Software block as follows.



powered by Astah

上図の凡例) 黄色:ターゲットユニット 青色:外部ユニット

Legend of above) Yellow:target units Blue:outside units

## ユースケースとAPI一覧 [use-case and API lists]

### 外部要因 ユースケース一覧 [outside factor use-case list]

positioning\_base\_libraryのユースケース一覧を記載する。

Describe use-case list of positioning\_base\_library.

ユースケース番号 [Use-case number]	機能カテゴリ [function category]	ユースケース名 [use-case name]	ユースケースを実現するAPI [API to realize use case]	備考 [notes]
positioning_base_library_MessageAPI_001	MessageAPI <i>Message API</i>	メッセージキューを生成する <i>Create the message queue</i>	_pb_CreateMsg	
positioning_base_library_MessageAPI_002		指定したPNOにメッセージを送信する <i>Send the message based on PNO</i>	_pb_SndMsg	

positioning_base_library_MessageAPI_003		指定したプロセスにメッセージを送信する  <i>Send the message based on process name</i>	_pb_SndMsg_Ext	
positioning_base_library_MessageAPI_004		指定したPNO向けのメッセージを受信する  <i>Receive the message based on PNO</i>	_pb_RcvMsg	
positioning_base_library_MessageAPI_005		プロセス内通信用（非コピーAPI用）送信バッファ取得  <i>Get message buffer inside process (not copy API)</i>	_pb_GetZcSndBuf	
positioning_base_library_MessageAPI_006		プロセス内通信用（非コピーAPI用）  <i>Send message inside process (not copy API)</i>	_pb_ZcSndMsg	
positioning_base_library_MutexAPI_001	MutexAPI  <i>Mutex API</i>	プロセス間で排他可能な名前付きミューテックスを作成する  <i>Create mutex between process</i>	_pb_CreateMutex	
positioning_base_library_NamePnoCnvAPI_001	NamePnoCnvAPI  <i>Name and Pno API</i>	プロセス名をPNOに変換する  <i>Change process name to PNO</i>	_pb_CnvName2Pno	
positioning_base_library_NamePnoCnvAPI_002		PNOをプロセス名に変換する  <i>Change PNO to process name</i>	_pb_CnvPno2Name	
positioning_base_library_ResourceAPI_001	ResourceAPI  <i>Resource API</i>	リソース取得判定(メッセージ制御テーブル)  <i>Check resource(message control table)</i>	_pb_GetMsgResource	
positioning_base_library_ResourceAPI_002		リソース取得判定(mutex制御情報)  <i>Check resource(mutex control information)</i>	_pb_GetMutexResource	
positioning_base_library_ResourceAPI_003		リソース取得判定(プロセス名-PNO変換テーブル)  <i>Check resource(process name - PNO table)</i>	_pb_GetOtherResource	
positioning_base_library_ResourceAPI_004		リソース解放(メッセージ制御テーブル)  <i>Release resource(message control table)</i>	_pb_ReleaseMsgResource	
positioning_base_library_ResourceAPI_005		リソース解放(mutex制御情報)  <i>Release resource(mutex control information)</i>	_pb_ReleaseMutexResource	
positioning_base_library_ResourceAPI_006		リソース解放(プロセス名-PNO変換テーブル)  <i>Release resource(process name - PNO table)</i>	_pb_ReleaseOtherResource	
positioning_base_library_SemaphoreAPI_001	SemaphoreAPI  <i>Semaphore API</i>	セマフォを生成してセマフォIDを返却する  <i>Create semaphore and get semaphore ID</i>	_pb_CreateSemaphore	

positioning_base_library_SemaphoreAPI_002		指定されたセマフォIDのセマフォを取得する <i>Get semaphore according to semaphore ID</i>	_pb_SemLock	
positioning_base_library_SemaphoreAPI_003		指定されたセマフォIDのセマフォを解放する <i>Release semaphore according to semaphore ID</i>	_pb_SemUnlock	
positioning_base_library_ShareDataAPI_001	ShareDataAPI <i>Share data API</i>	共有データ領域作成 <i>Create share data</i>	_pb_CreateShareData	
positioning_base_library_ShareDataAPI_002		共有データ領域リンク <i>Link share data</i>	_pb_LinkShareData	
positioning_base_library_TimerAPI_001	TimerAPI <i>Timer API</i>	タイマ起動指示 <i>Start timer</i>	_pb_ReqTimerStart	
positioning_base_library_TimerAPI_002		タイマ停止指示 <i>Stop timer</i>	_pb_TimerStop	
positioning_base_library_EventAPI_001	EventAPI <i>Event API</i>	指定した名称でイベントを生成し、そのイベントIDを返却する <i>Create event according to event name, and get event ID</i>	_pb_CreateEvent	
positioning_base_library_EventAPI_002		指定されたイベントIDのイベント値が指定された範囲になるまで待つ <i>Wait for the event according to event ID and event value</i>	_pb_WaitEvent	
positioning_base_library_EventAPI_003		イベント値を設定する <i>Set the event</i>	_pb_SetEvent	
positioning_base_library_EventAPI_004		指定されたイベントIDのイベントを削除する <i>Delete event according to event ID</i>	_pb_DeleteEvent	
positioning_base_library_GetDebugAPI_001	GetDbgAPI <i>Get Debug API</i>	dump情報取得(メッセージ) <i>Get dump information (message)</i>	_pb_GetDebugMsgMngTbl	
positioning_base_library_GetDebugAPI_002		dump情報取得(ミューテックス) <i>Get dump information (mutex)</i>	_pb_GetDebugMutexMngTbl	
positioning_base_library_GetDebugAPI_003		dump情報取得(タイマ) <i>Get dump information (timer)</i>	_pb_GetDebugTimerMngTbl	
positioning_base_library_GetDebugAPI_004		dump情報取得(イベント) <i>Get dump information (event)</i>	_pb_GetDebugEventMngTbl	
positioning_base_library_GetDebugAPI_005		dump情報取得(メモリ) <i>Get dump information (memory)</i>	_pb_GetDebugMemoryMngTbl	
positioning_base_library_GetDebugAPI_006		dump情報取得(共通) <i>Get dump information (other)</i>	_pb_GetDebugOtherMngTbl	

positioning_base_library_HandleAPI_001	HandleAPI <i>Handler API</i>	自スレッドのアプリケーションハンドルを取得する <i>Get application handle</i>	_pb_GetAppHandle	
positioning_base_library_HandleAPI_002		自スレッドのアプリケーションハンドルを設定する <i>Set application handle</i>	_pb_SetAppHandle	

## 内部処理 エラーユースケース一覧 [\[internal processing error use-case list\]](#)

エラー番号 <a href="#">[error number]</a>	機能カテゴリ <a href="#">[function category]</a>	ユースケース名 <a href="#">[use-case name]</a>	戻り値 <a href="#">[return value]</a>	備考 <a href="#">[notes]</a>
positioning_base_library_MessageAPI_Error_001	MessageAPI <i>Message API</i>	framework_unifiedのエラー <i>Processing error of framework_unified</i>	RET_ERROR	
positioning_base_library_TimerAPI_Error_001	TimerAPI <i>Timer API</i>		RET_ERROR	

## 初期化と終了シーケンス一覧 [\[start and end use-case list\]](#)

機能カテゴリ <a href="#">[function category]</a>	ユースケース名 <a href="#">[use-case name]</a>	備考 <a href="#">[notes]</a>
初期化 <i>initialize</i>	positioning_base_libraryの初期化 <i>Initialize positioning_base_library</i>	positioning_base_libraryモジュール初期化処理。 <i>positioning_base_library moudle initial.</i>
終了 <i>destroy</i>	Base API終了処理 <i>Teardown Base API</i>	positioning_base_libraryモジュール終了化処理。 <i>positioning_base_library moudle deinitializes.</i>
	該当スレッドを終了させる <i>Stop thread</i>	
	該当プロセスを終了させる <i>Stop process</i>	

## メッセージキューを生成する [\[positioning\\_base\\_library\\_MessageAPI\\_001\]](#)

### 概要 [\[Overview\]](#)

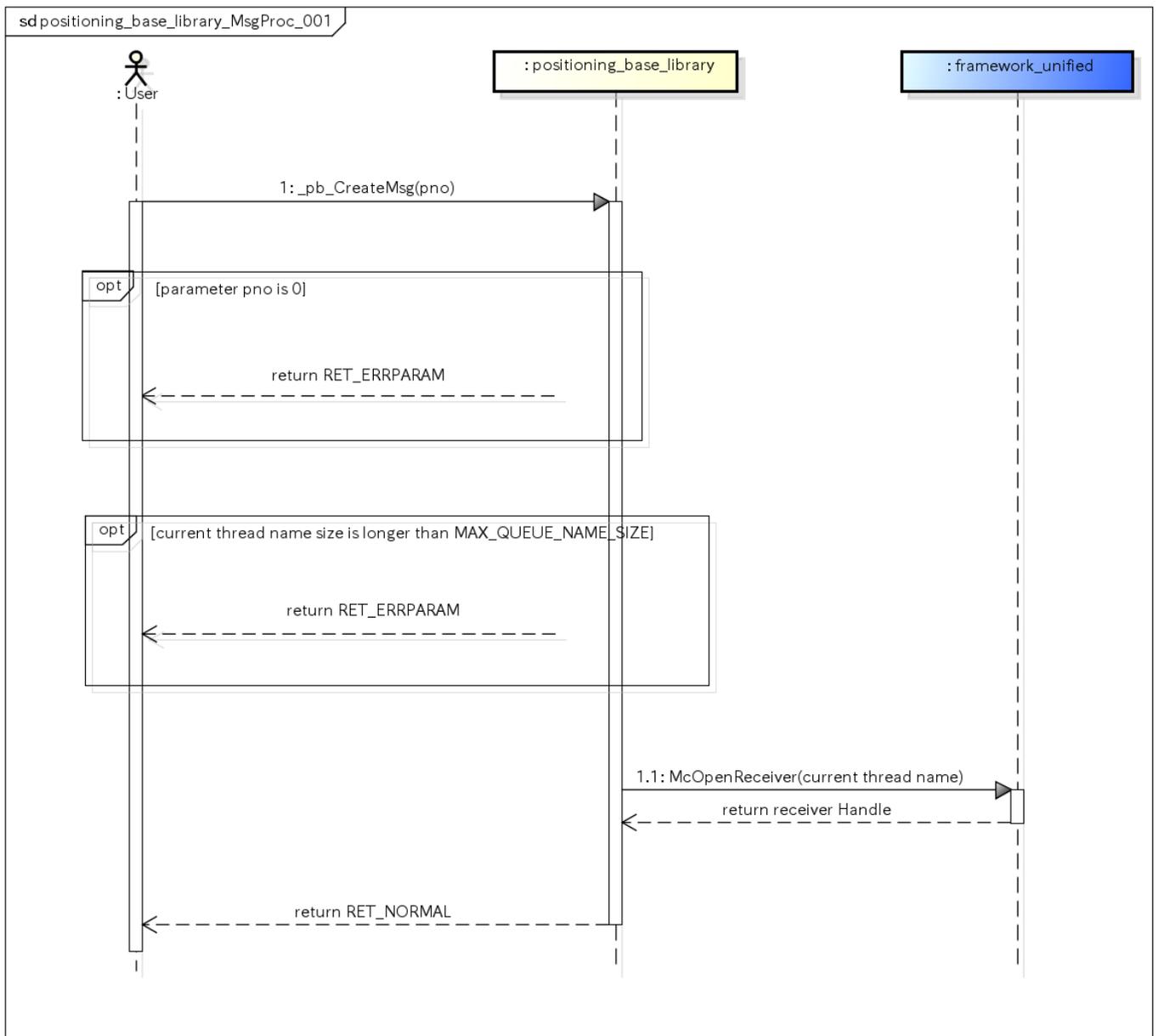
メッセージキューを生成する

*Create the message queue*

### シーケンス [\[Sequence\]](#)

メッセージキューを生成するシーケンスを以下に示す

*The sequence of create the message queue*



## 指定したPNOにメッセージを送信する [\[positioning\\_base\\_library\\_MessageAPI\\_002\]](#)

### 概要 [\[Overview\]](#)

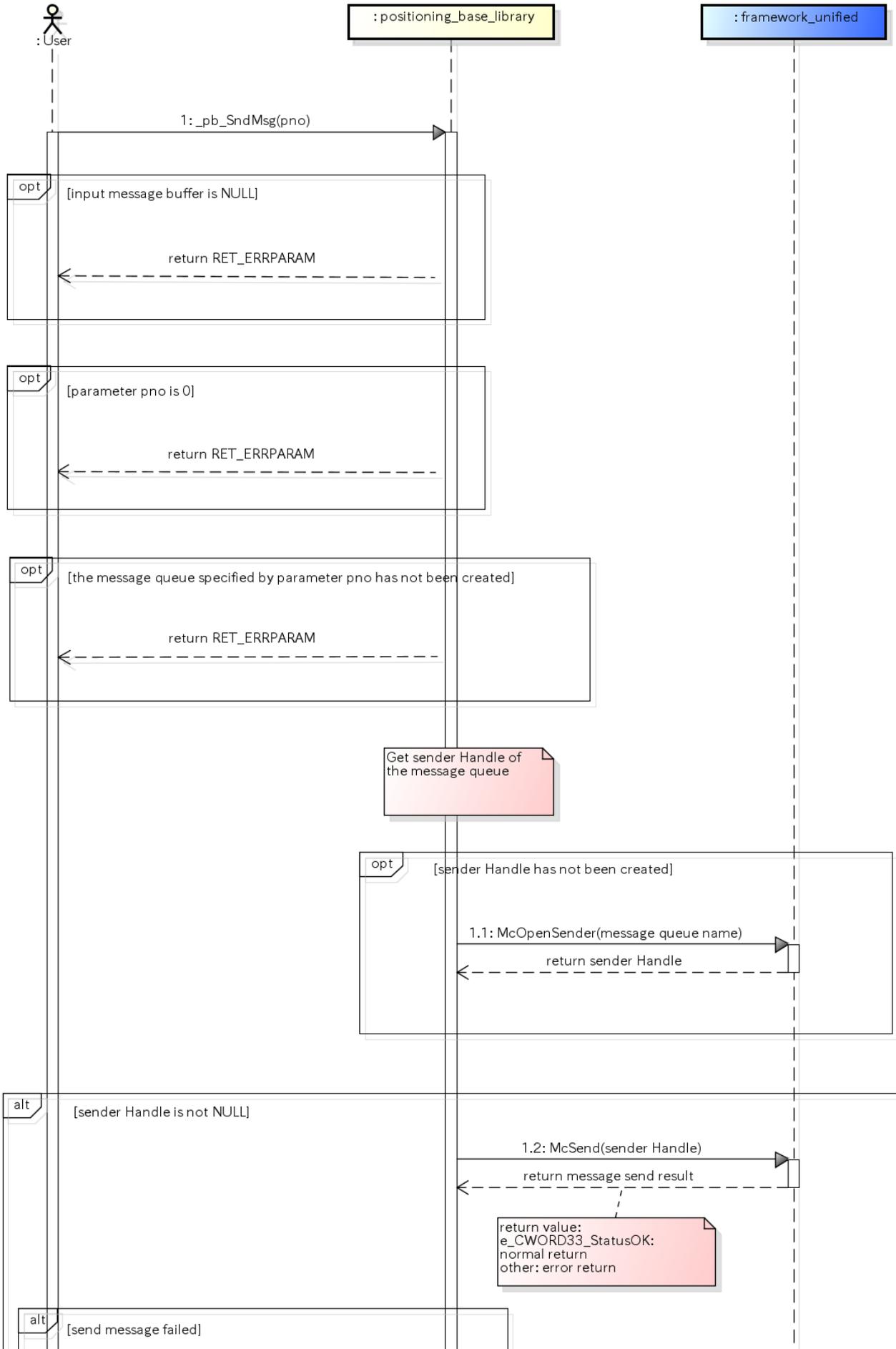
指定したPNOにメッセージを送信する

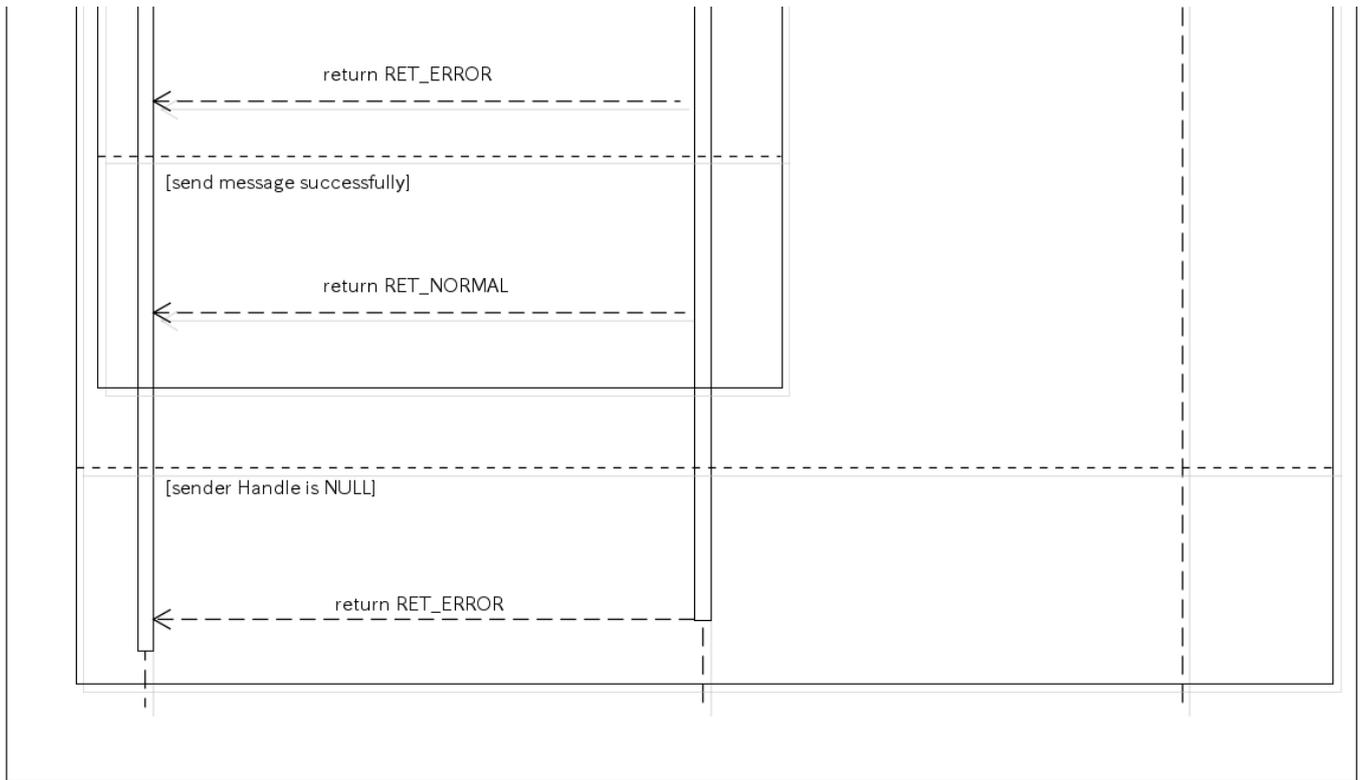
*send the message based on PNO*

### シーケンス [\[Sequence\]](#)

指定したPNOにメッセージを送信するシーケンスを以下に示す

*The sequence of send the message based on PNO*





## 指定したプロセスにメッセージを送信する [\[positioning\\_base\\_library\\_MessageAPI\\_003\]](#)

### 概要 [\[Overview\]](#)

指定したプロセスにメッセージを送信する

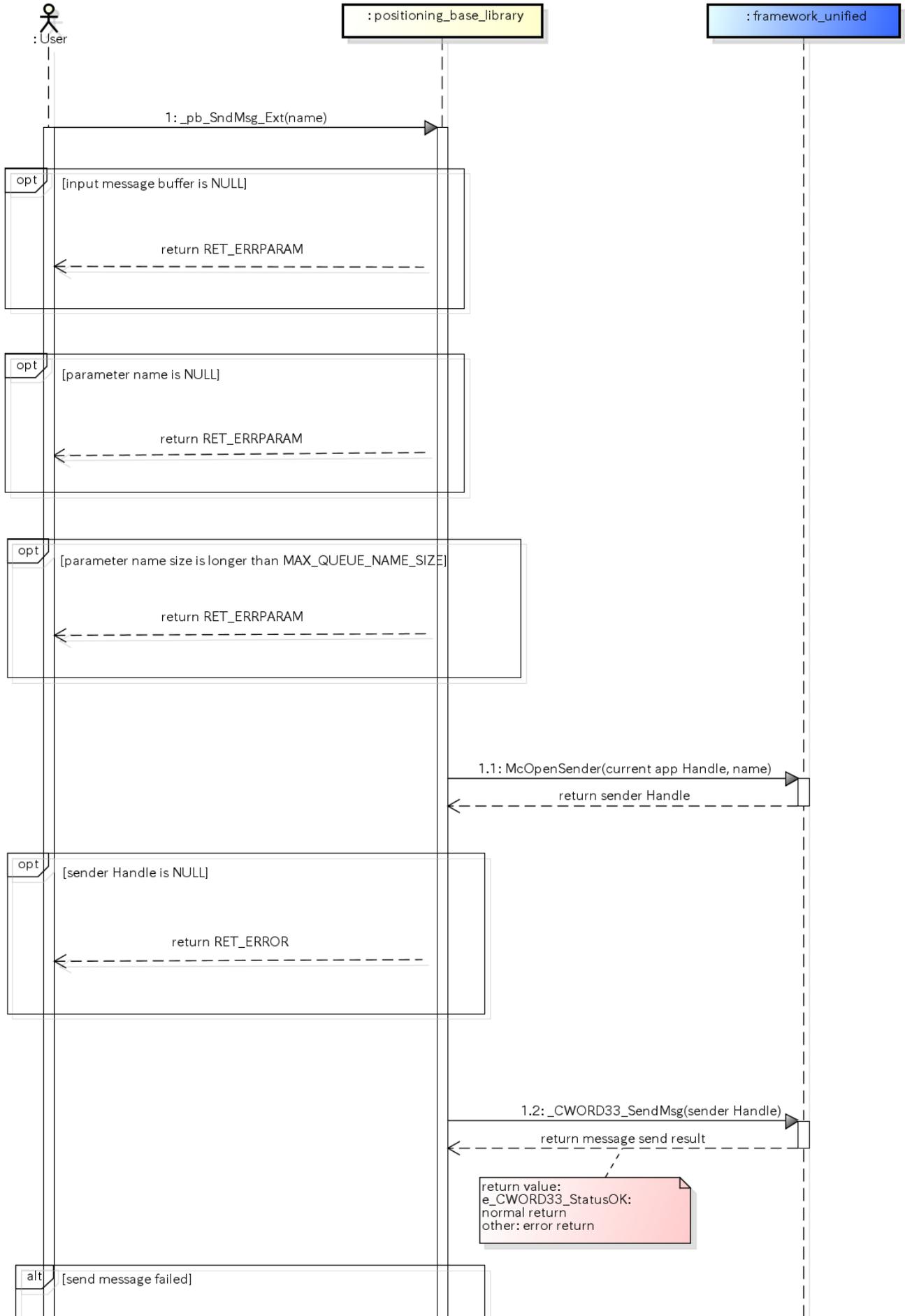
*Send the message based on process name*

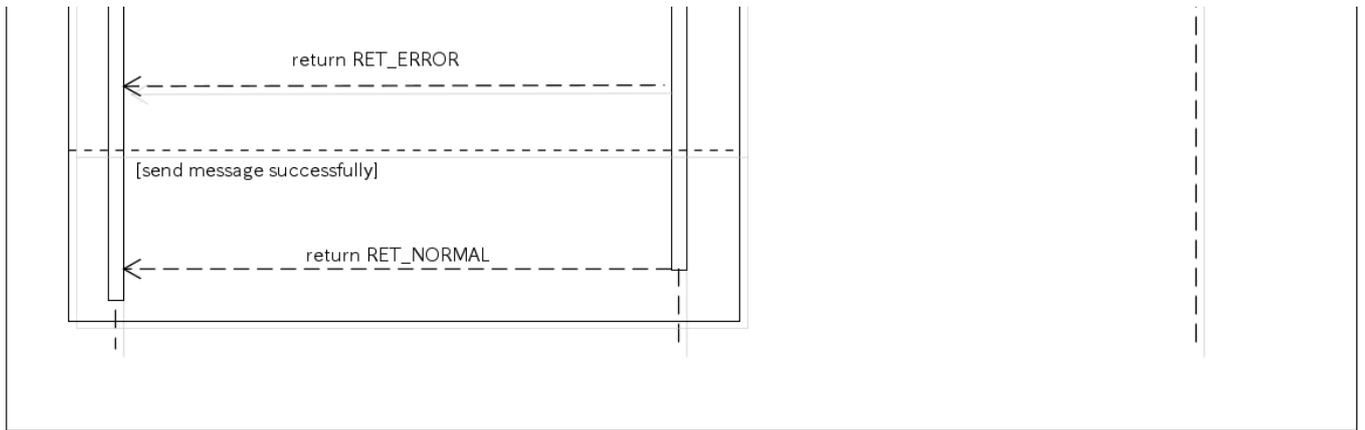
### シーケンス [\[Sequence\]](#)

指定したプロセスにメッセージを送信するシーケンスを以下に示す

*The sequence of send the message based on process name*







## 指定したPNO向けのメッセージを受信する [\[positioning\\_base\\_library\\_MessageAPI\\_004\]](#)

### 概要 [\[Overview\]](#)

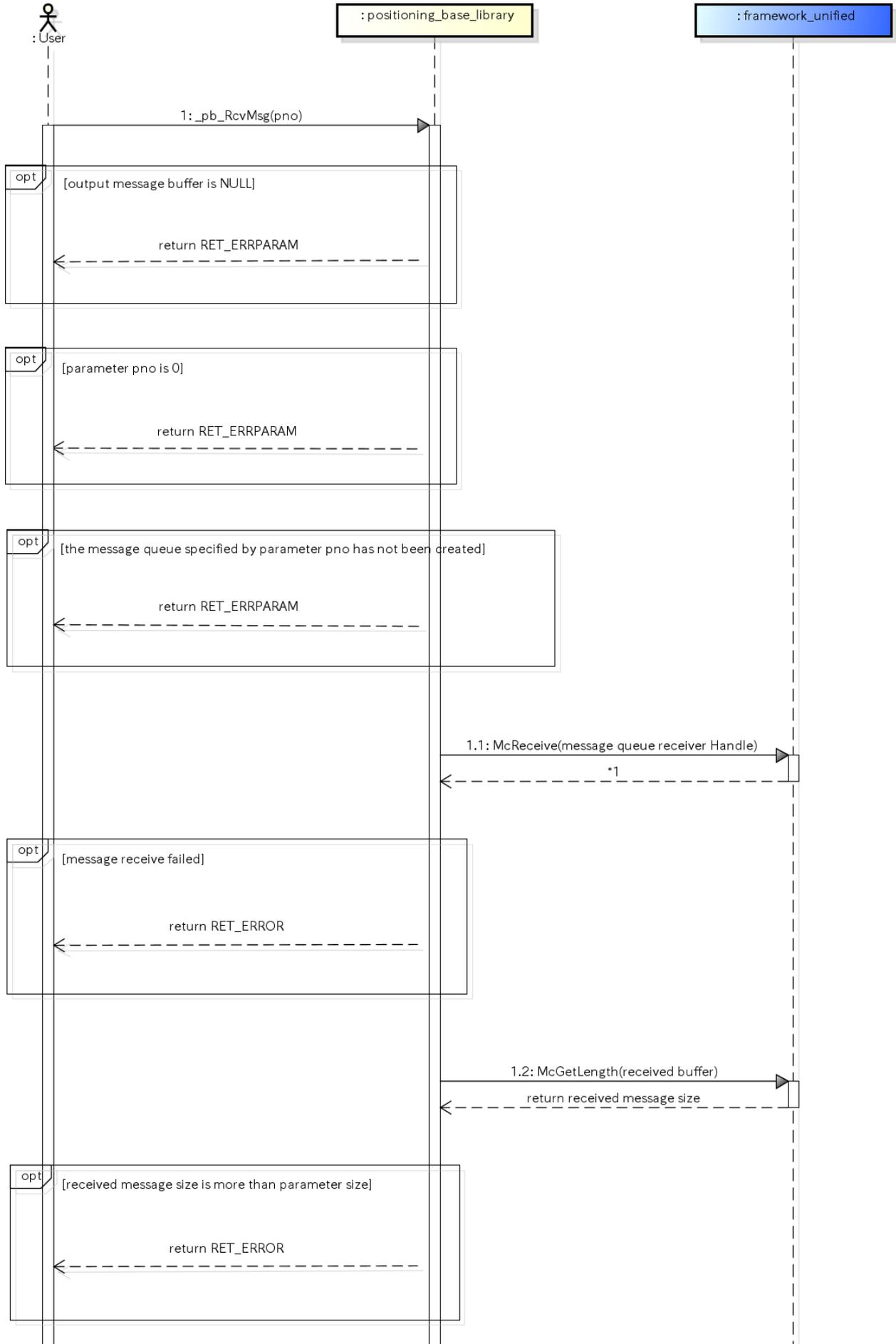
指定したPNO向けのメッセージを受信する

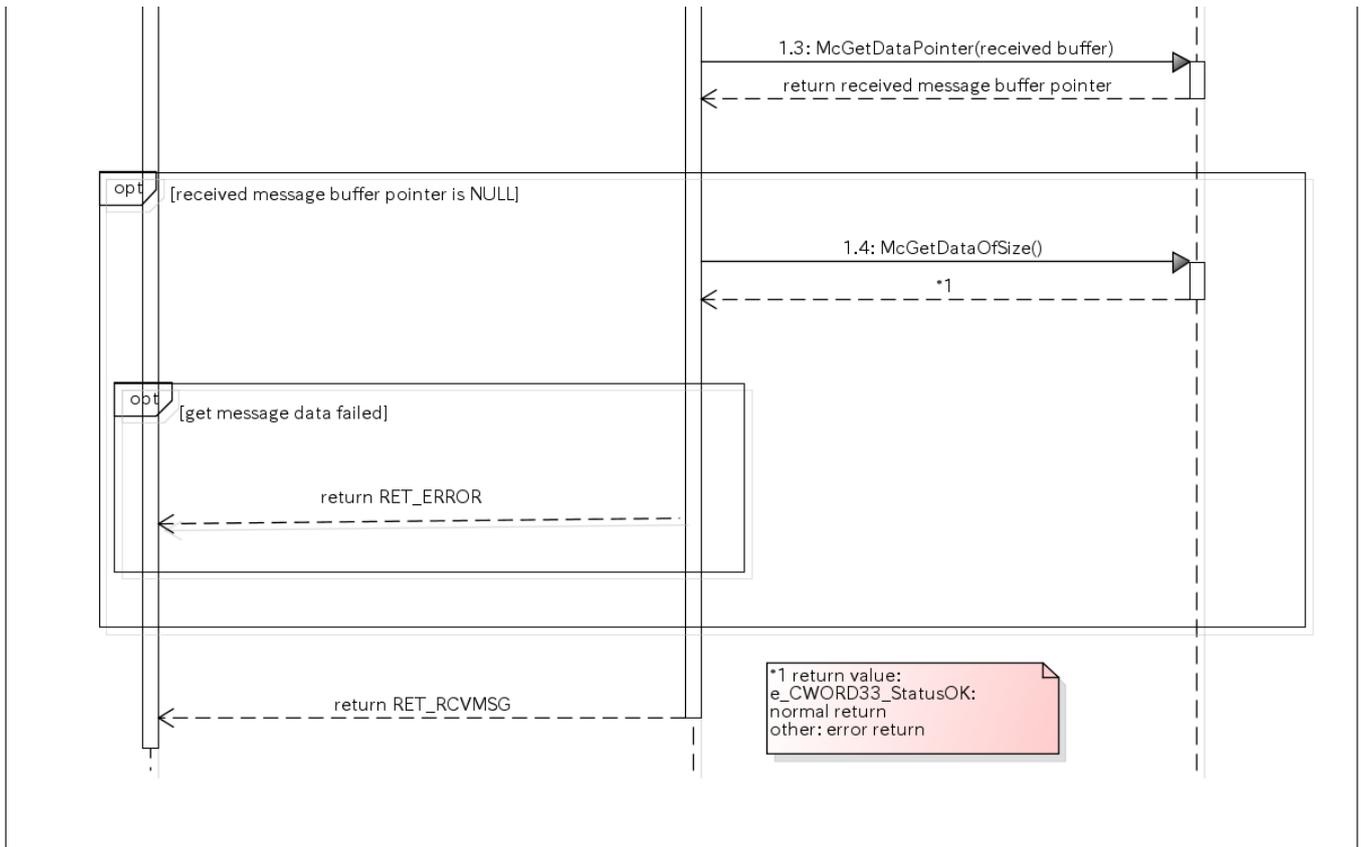
*Receive the message based on PNO*

### シーケンス [\[Sequence\]](#)

指定したPNO向けのメッセージを受信するシーケンスを以下に示す

*The sequence of receive the message based on PNO*





## プロセス内通信用（非コピーAPI用）送信バッファ取得 [\[positioning\\_base\\_library\\_MessageAPI\\_005\]](#)

### 概要 [\[Overview\]](#)

プロセス内通信用（非コピーAPI用）送信バッファ取得

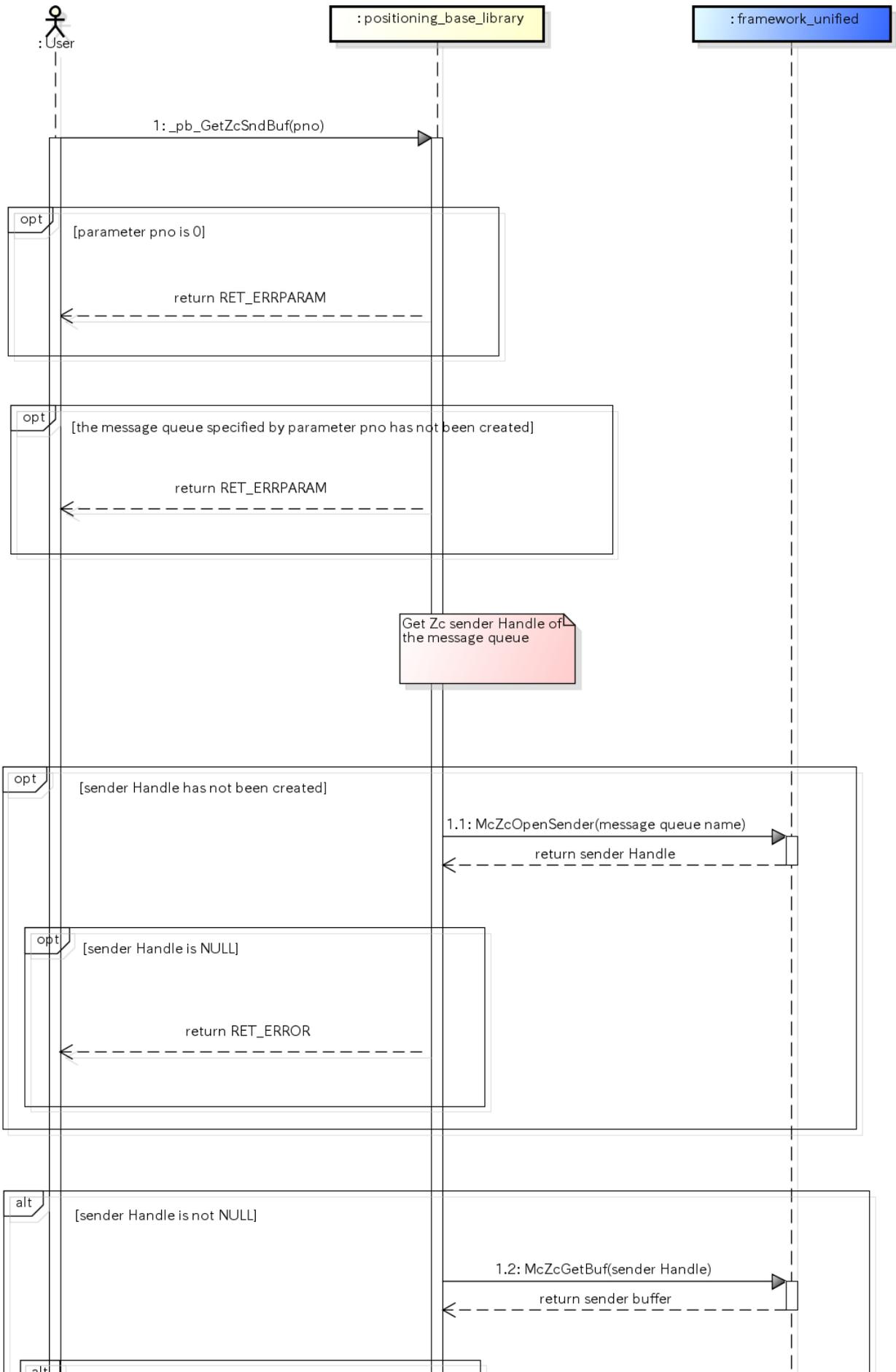
*Get message buffer inside process (not copy API)*

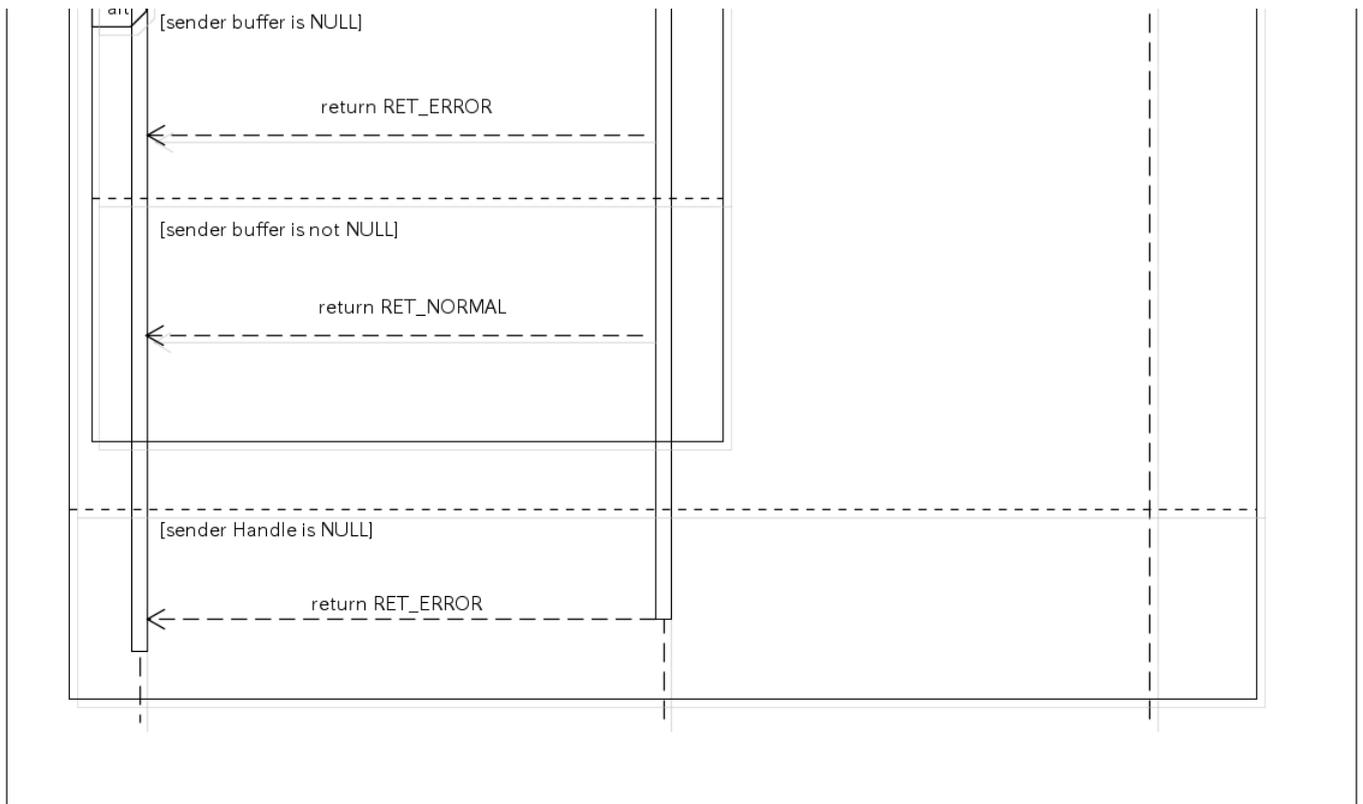
### シーケンス [\[Sequence\]](#)

プロセス内通信用（非コピーAPI用）送信バッファ取得シーケンスを以下に示す

*The sequence of get message buffer inside process (not copy API)*







## プロセス内通信用（非コピーAPI用） [\[positioning\\_base\\_library\\_MessageAPI\\_006\]](#)

### 概要 [\[Overview\]](#)

プロセス内通信用（非コピーAPI用）

*Send message inside process (not copy API)*

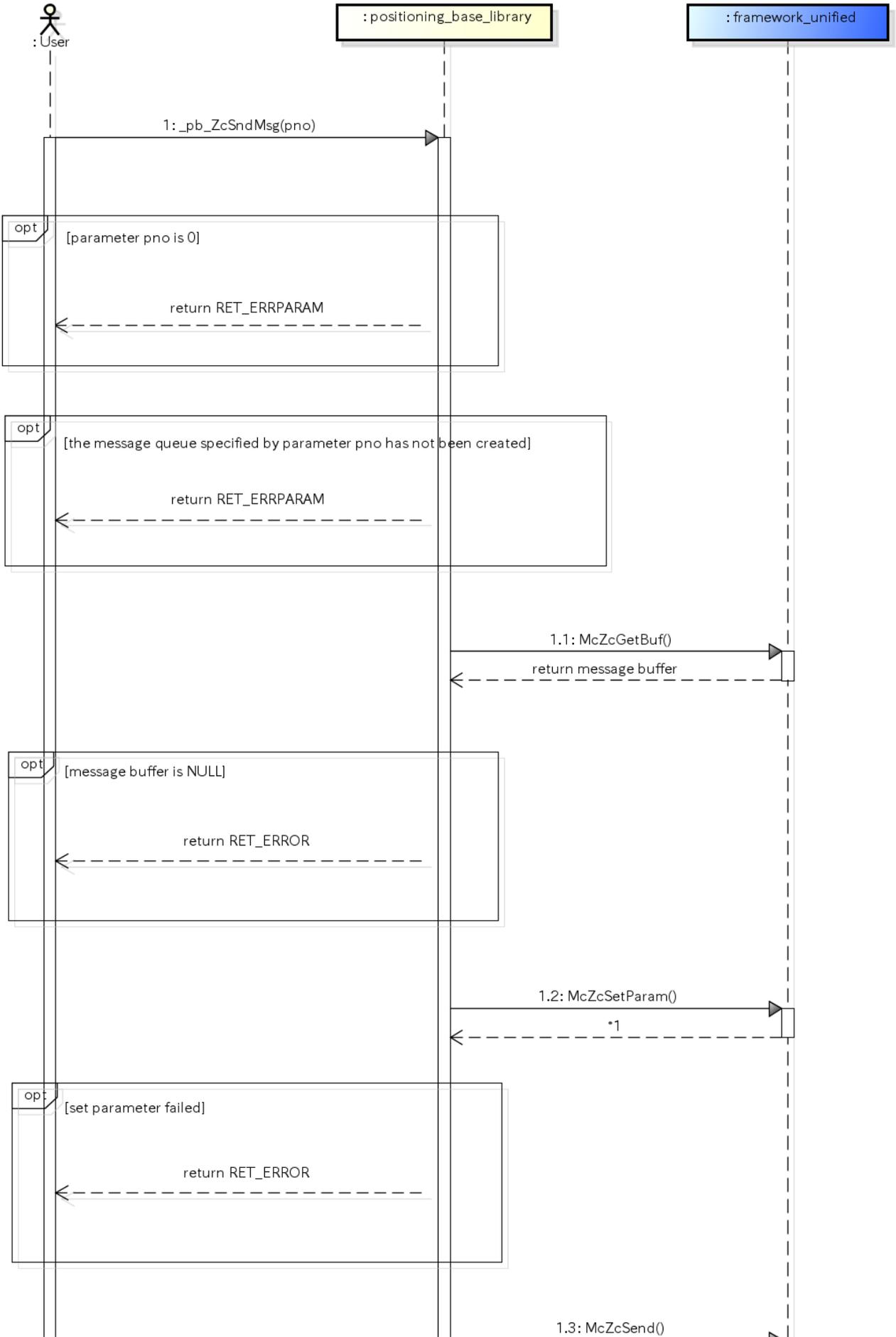
### シーケンス [\[Sequence\]](#)

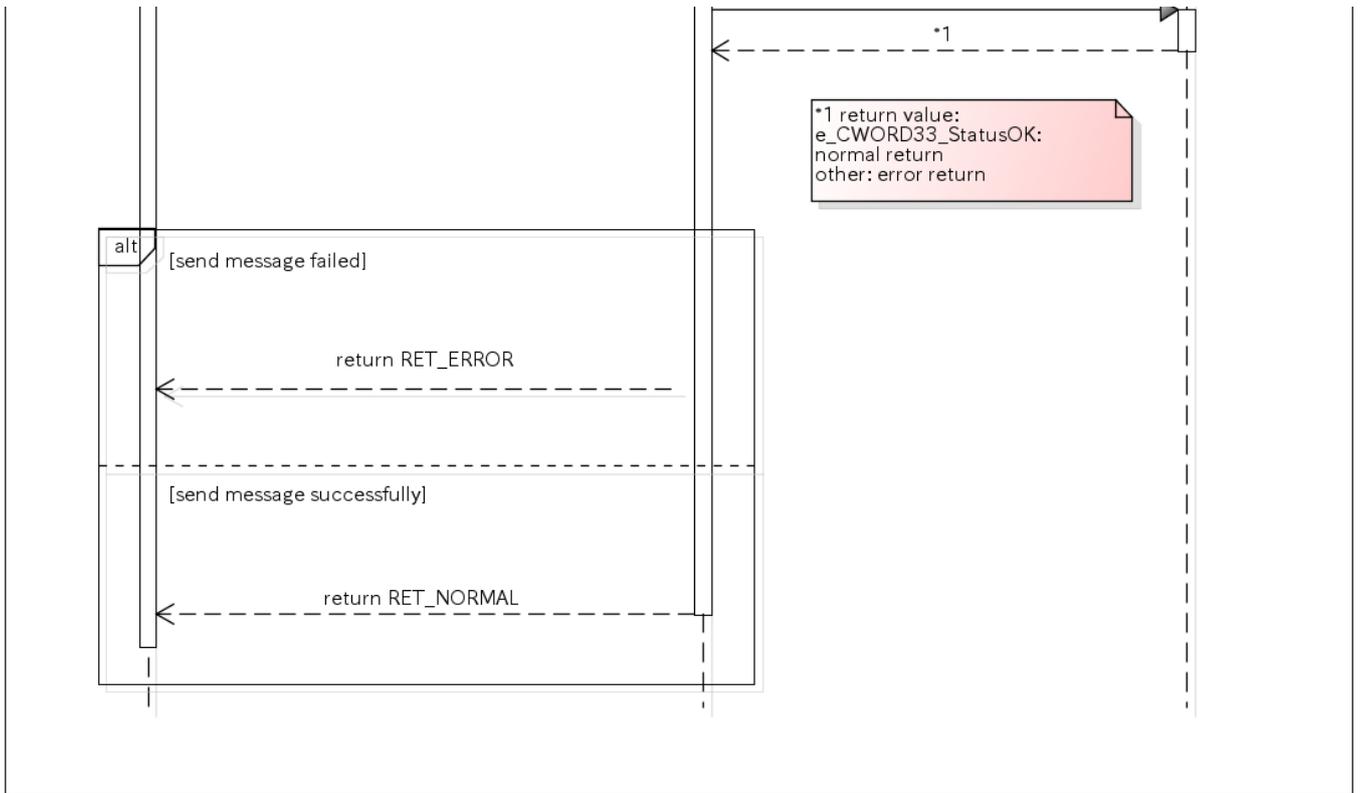
プロセス内通信用（非コピーAPI用）シーケンスを以下に示す

*The sequence of send message inside process (not copy API)*



sdpositioning\_base\_library\_MsgProc\_006





## プロセス間で排他可能な名前付きミューテックスを作成する [\[positioning\\_base\\_library\\_Mutex API\\_001\]](#)

### 概要 [\[Overview\]](#)

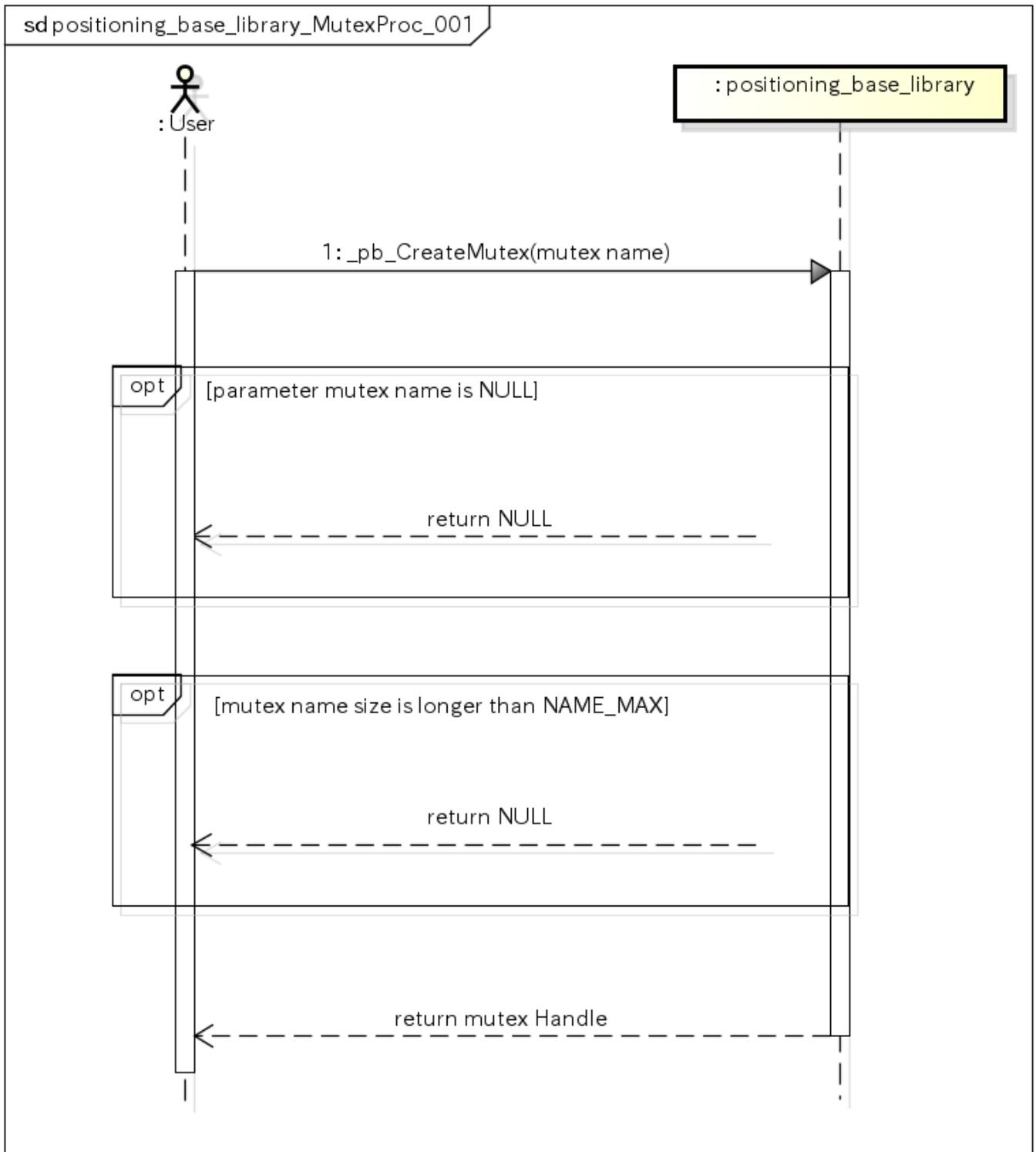
プロセス間で排他可能な名前付きミューテックスを作成する

*Create mutex between process*

### シーケンス [\[Sequence\]](#)

プロセス間で排他可能な名前付きミューテックスを作成するシーケンスを以下に示す

*The sequence of create mutex between process*



## プロセス名をPNOに変換する [\[positioning\\_base\\_library\\_NamePnoCnvAPI\\_001\]](#)

### 概要 [\[Overview\]](#)

プロセス名をPNOに変換する

*Change process name to PNO*

### シーケンス [\[Sequence\]](#)

プロセス名をPNOに変換するシーケンスを以下に示す

*The sequence of change process name to PNO*

sd positioning\_base\_library\_NamePnoCnv\_001

:User

:positioning\_base\_library

1: \_pb\_CnvName2Pno(name)

opt

[parameter name is NULL]

return 0

opt

[name size is longer than THREAD\_NAME\_LEN\_MAX]

return 0

Get pno of parameter name in management table

return pno

## PNOをプロセス名に変換する [\[positioning\\_base\\_library\\_NamePnoCnvAPI\\_002\]](#)

### 概要 [\[Overview\]](#)

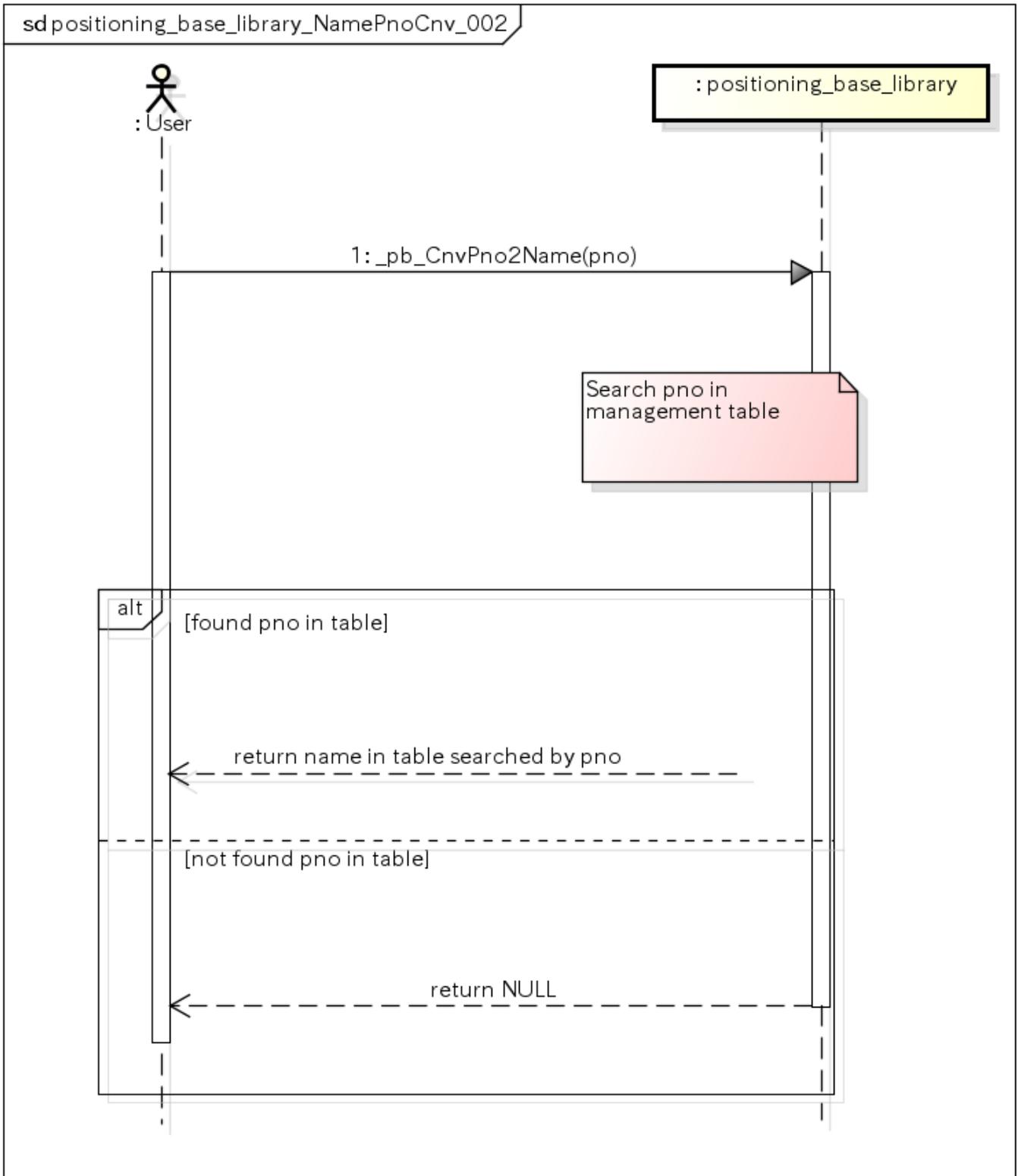
PNOをプロセス名に変換する

*Change PNO to process name*

### シーケンス [\[Sequence\]](#)

PNOをプロセス名に変換するシーケンスを以下に示す

*The sequence of change PNO to process name*



リソース取得判定(メッセージ制御テーブル) [[positioning\\_base\\_library\\_ResourceAPI\\_001](#)]

概要 [[Overview](#)]

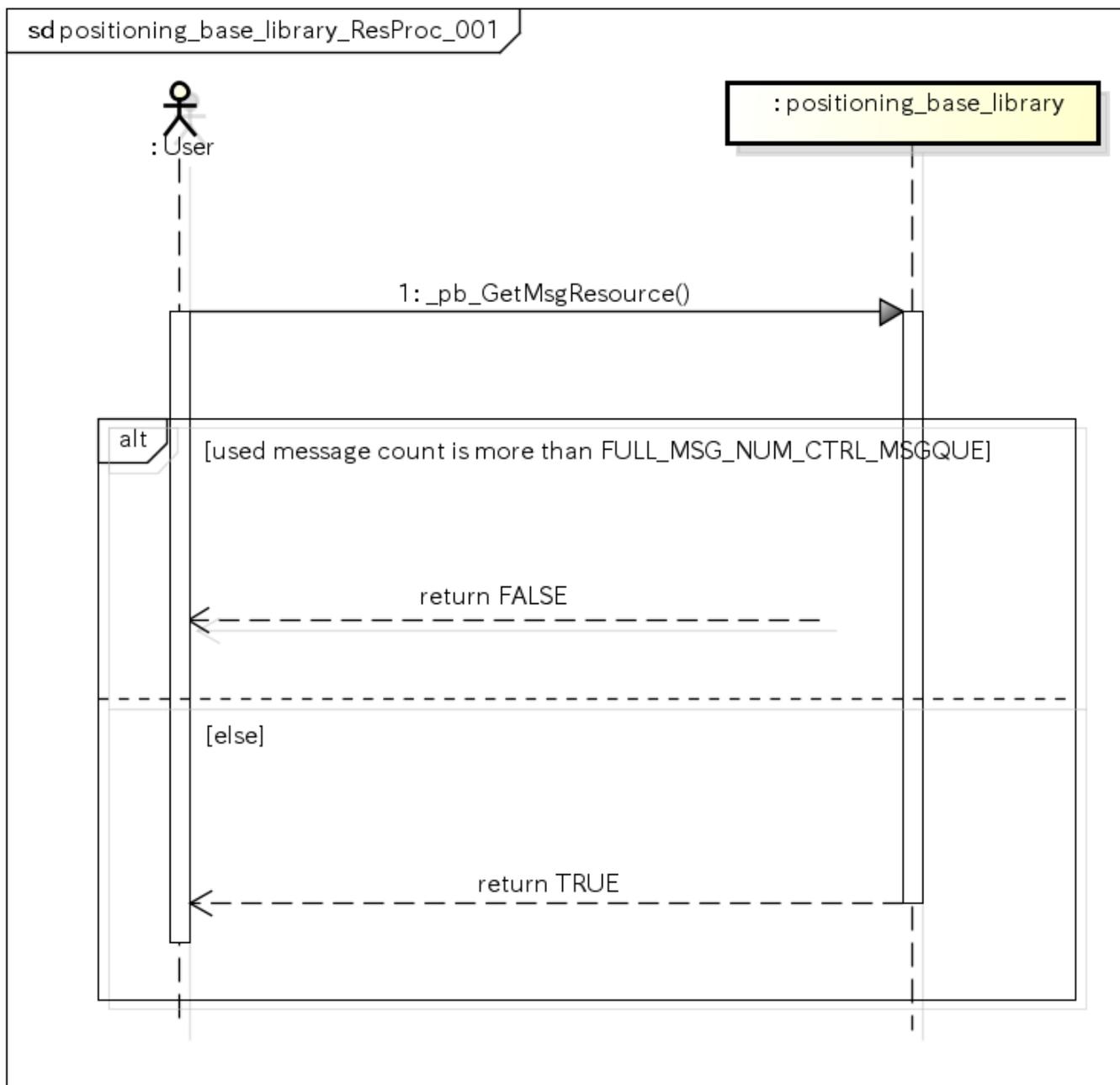
リソース取得判定(メッセージ制御テーブル)

[Check resource\(message control table\)](#)

## シーケンス [Sequence]

リソース取得判定(メッセージ制御テーブル)シーケンスを以下に示す

*The sequence of check resource(message control table)*



## リソース取得判定(mutex制御情報) [positioning\_base\_library\_ResourceAPI\_002]

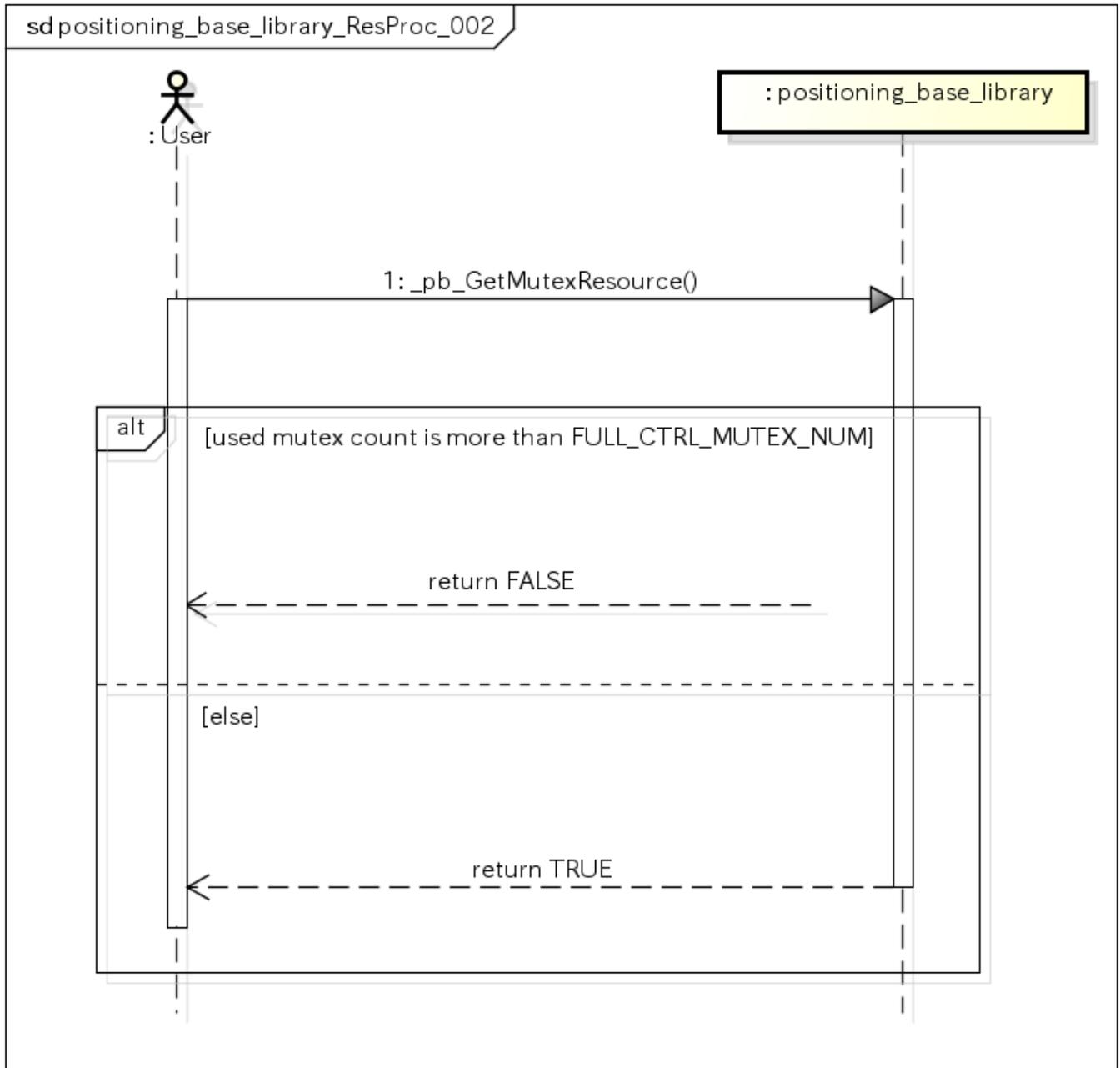
### 概要 [Overview]

リソース取得判定(mutex制御情報)

*Check resource(mutex control information)*

### シーケンス [Sequence]

リソース取得判定(mutex制御情報)シーケンスを以下に示す



### リソース取得判定(プロセス名-PNO変換テーブル) [[positioning\\_base\\_library\\_ResourceAPI\\_003](#)]

#### 概要 [[Overview](#)]

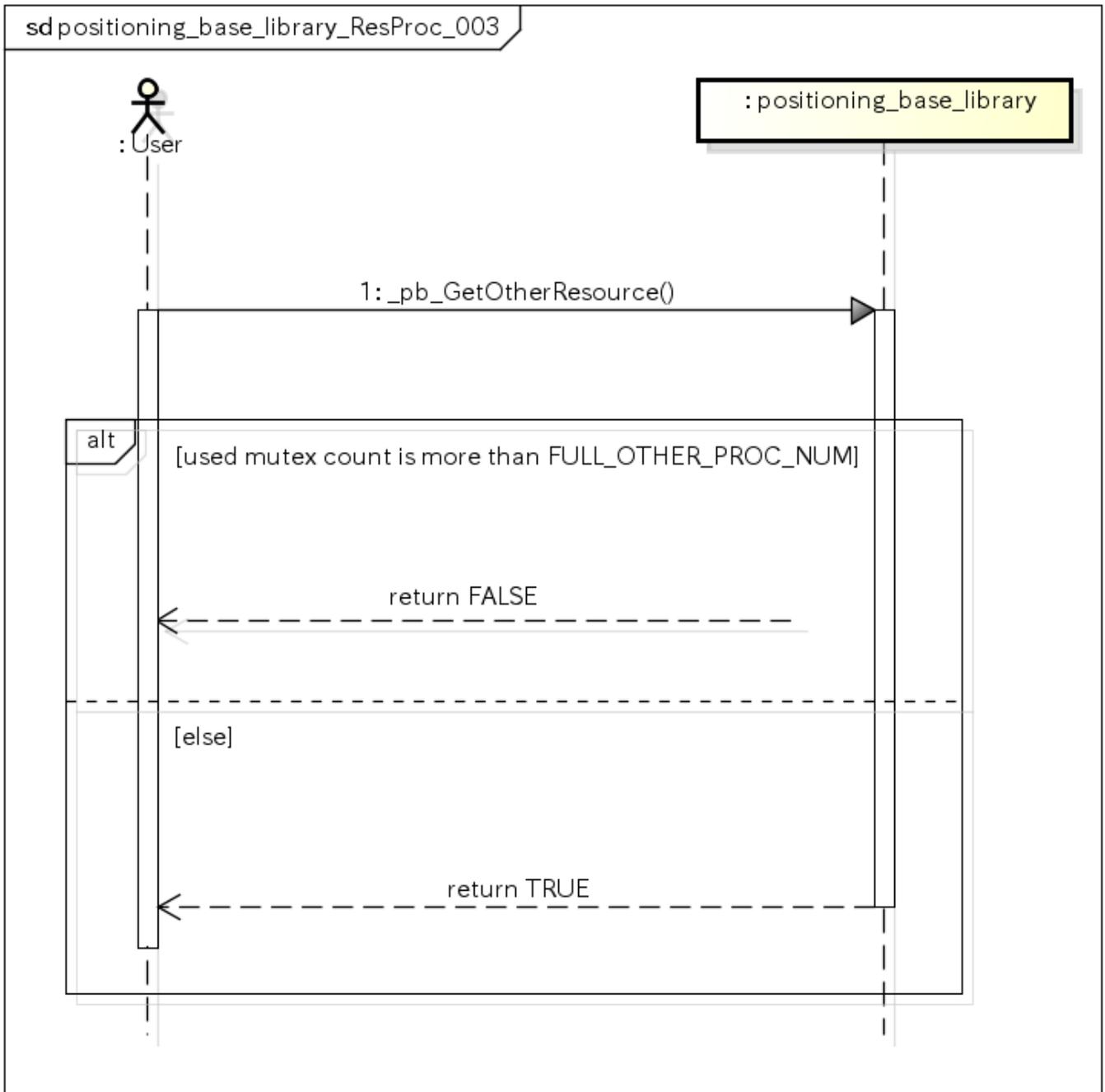
リソース取得判定(プロセス名-PNO変換テーブル)

*Check resource(process name - PNO table)*

#### シーケンス [[Sequence](#)]

リソース取得判定(プロセス名-PNO変換テーブル)シーケンスを以下に示す

*The sequence of check resource(process name - PNO table)*



## リソース解放(メッセージ制御テーブル) [\[positioning\\_base\\_library\\_ResourceAPI\\_004\]](#)

### 概要 [\[Overview\]](#)

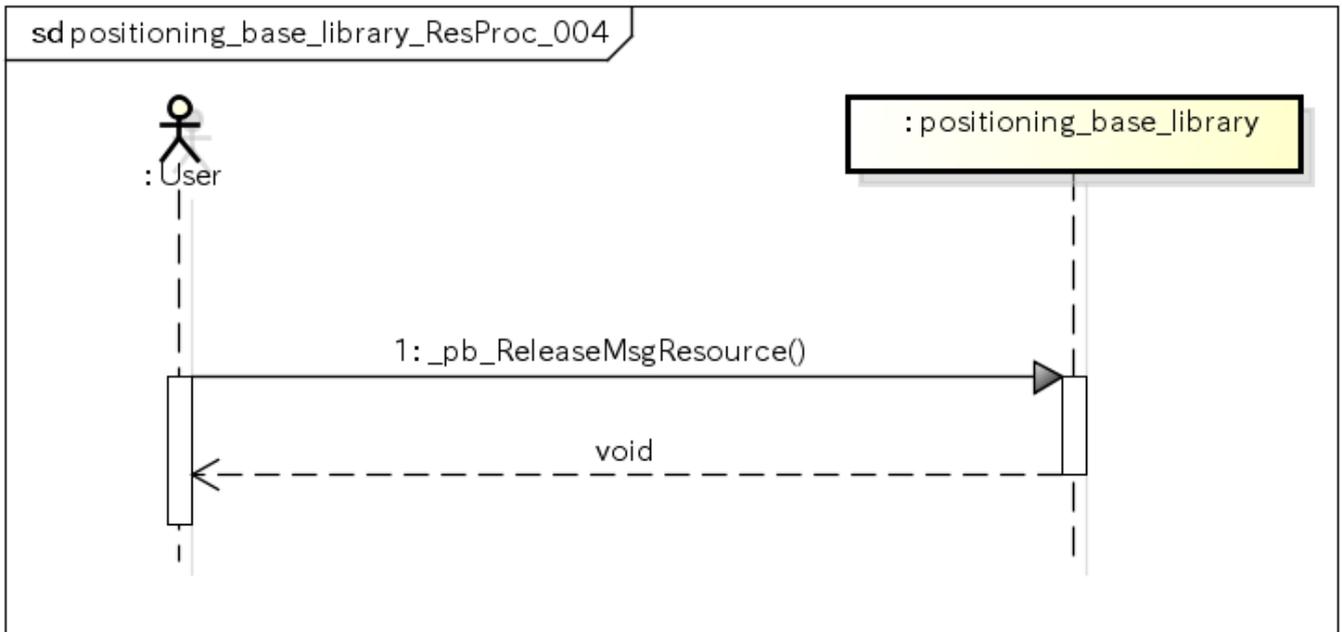
リソース解放(メッセージ制御テーブル)

*Release resource(message control table)*

### シーケンス [\[Sequence\]](#)

リソース解放(メッセージ制御テーブル)シーケンスを以下に示す

*The sequence of release resource(message control table)*



リソース解放(mutex制御情報) [[positioning\\_base\\_library\\_ResourceAPI\\_005](#)]

概要 [[Overview](#)]

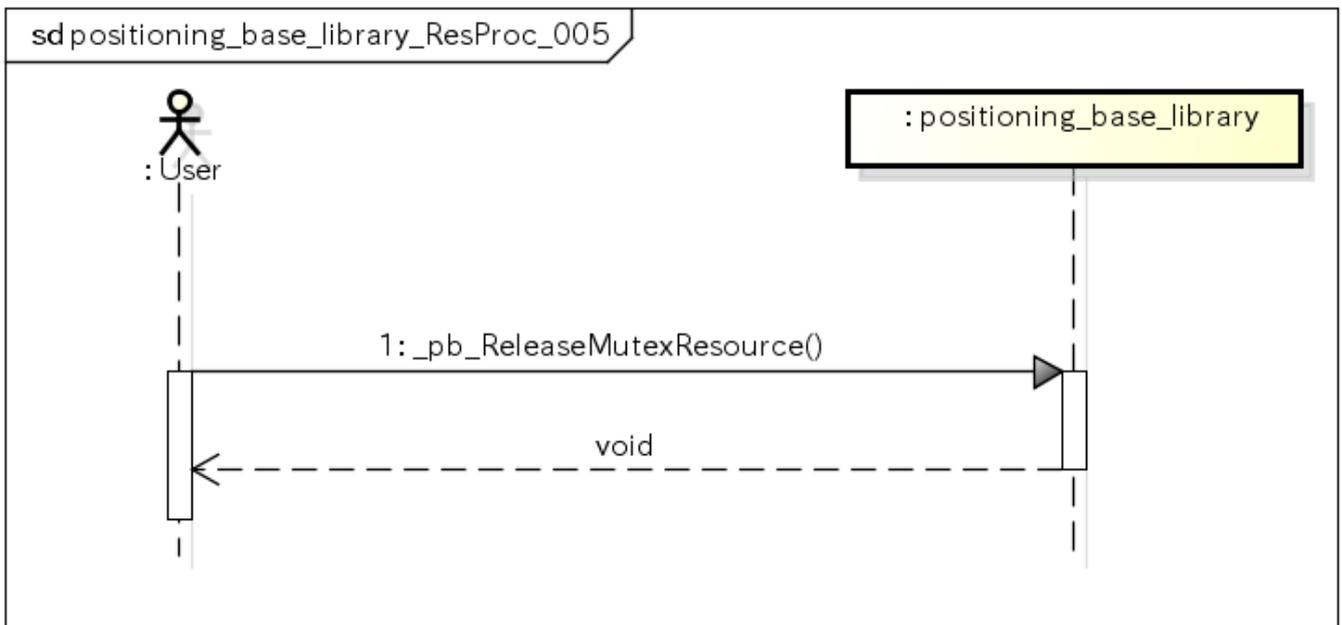
リソース解放(mutex制御情報)

*Release resource(mutex control information)*

シーケンス [[Sequence](#)]

リソース解放(mutex制御情報)シーケンスを以下に示す

*The sequence of release resource(mutex control information)*



リソース解放(プロセス名-PNO変換テーブル) [[positioning\\_base\\_library\\_ResourceAPI\\_006](#)]

## 概要 [Overview]

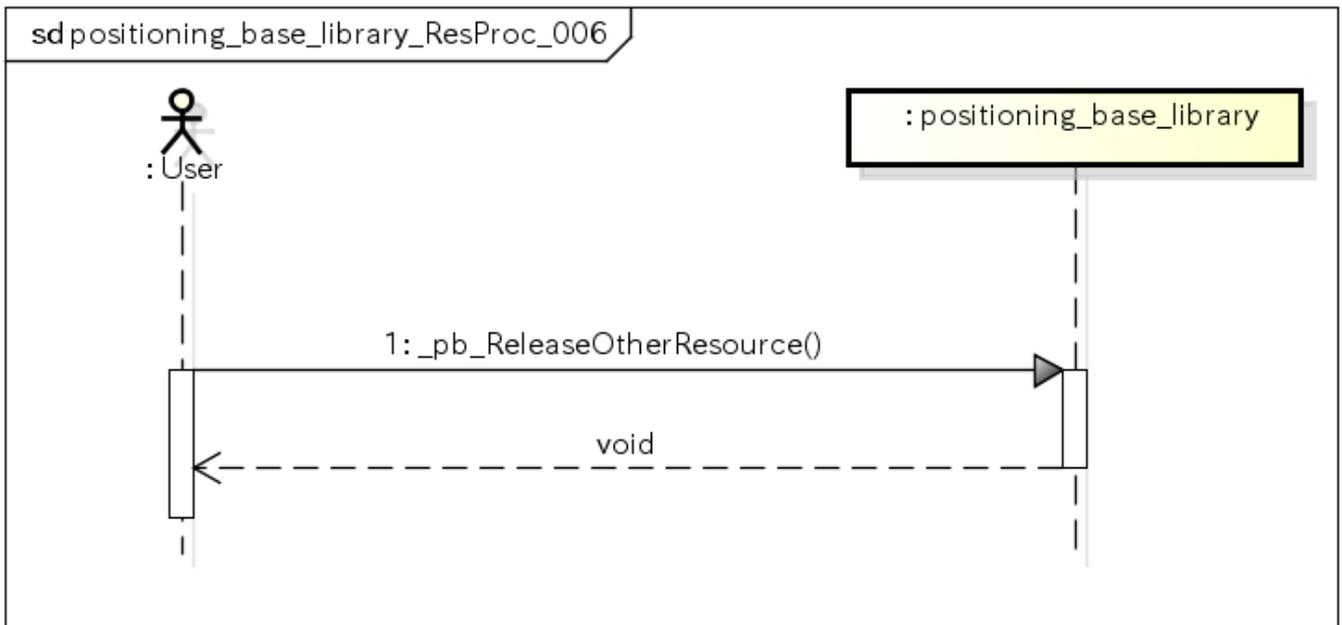
リソース解放(プロセス名-PNO変換テーブル)

*Release resource(process name - PNO table)*

## シーケンス [Sequence]

リソース解放(プロセス名-PNO変換テーブル)シーケンスを以下に示す

*The sequence of release resource(process name - PNO table)*



## セマフォを生成してセマフォIDを返却する [positioning\_base\_library\_SemaphoreAPI\_001]

### 概要 [Overview]

セマフォを生成してセマフォIDを返却する

*Create semaphore and get semaphore ID*

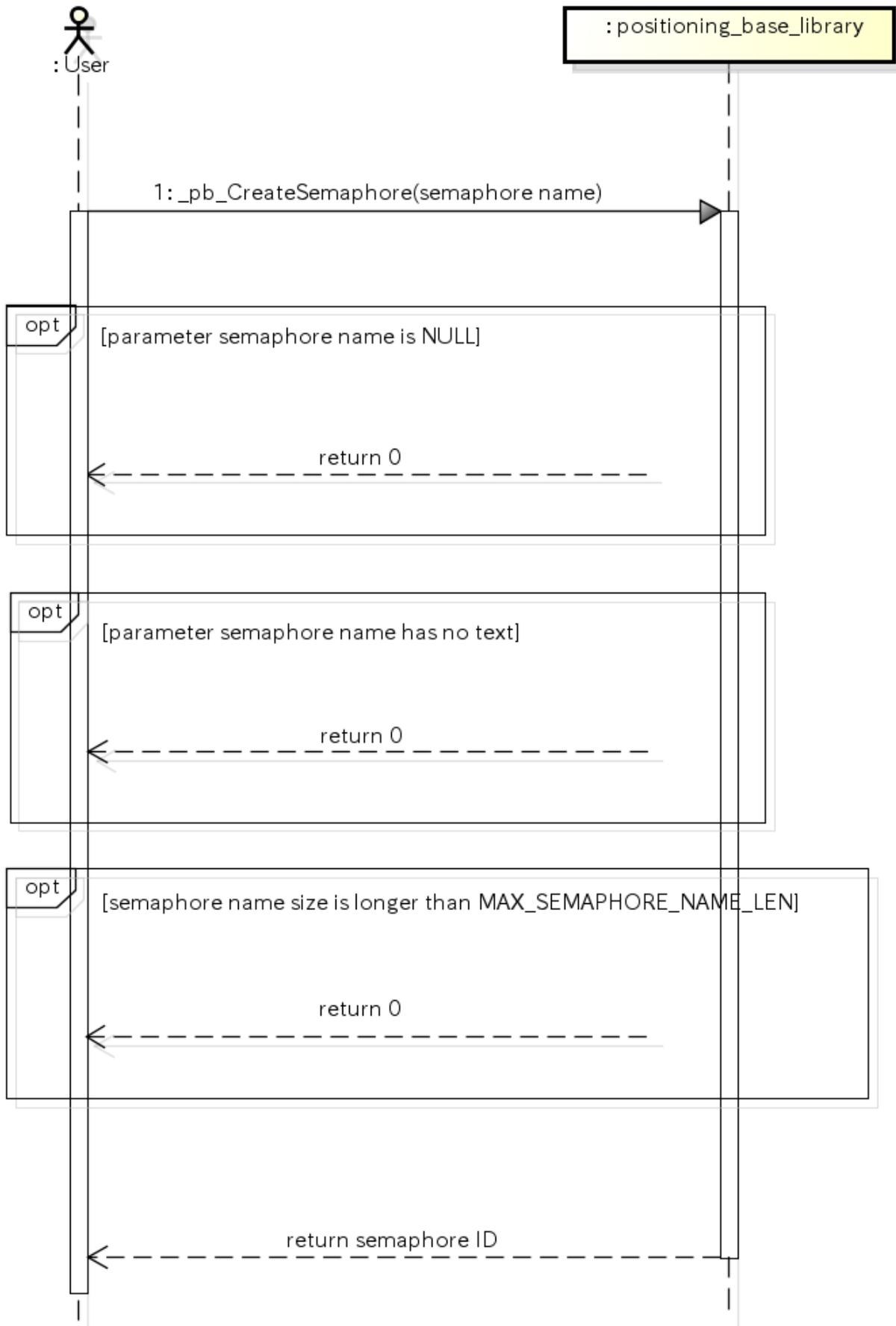
### シーケンス [Sequence]

セマフォを生成してセマフォIDを返却するシーケンスを以下に示す

*The sequence of create semaphore and get semaphore ID*



sd positioning\_base\_library\_SemProc\_001



## 指定されたセマフォIDのセマフォを取得する [\[positioning\\_base\\_library\\_SemaphoreAPI\\_002\]](#)

### 概要 [\[Overview\]](#)

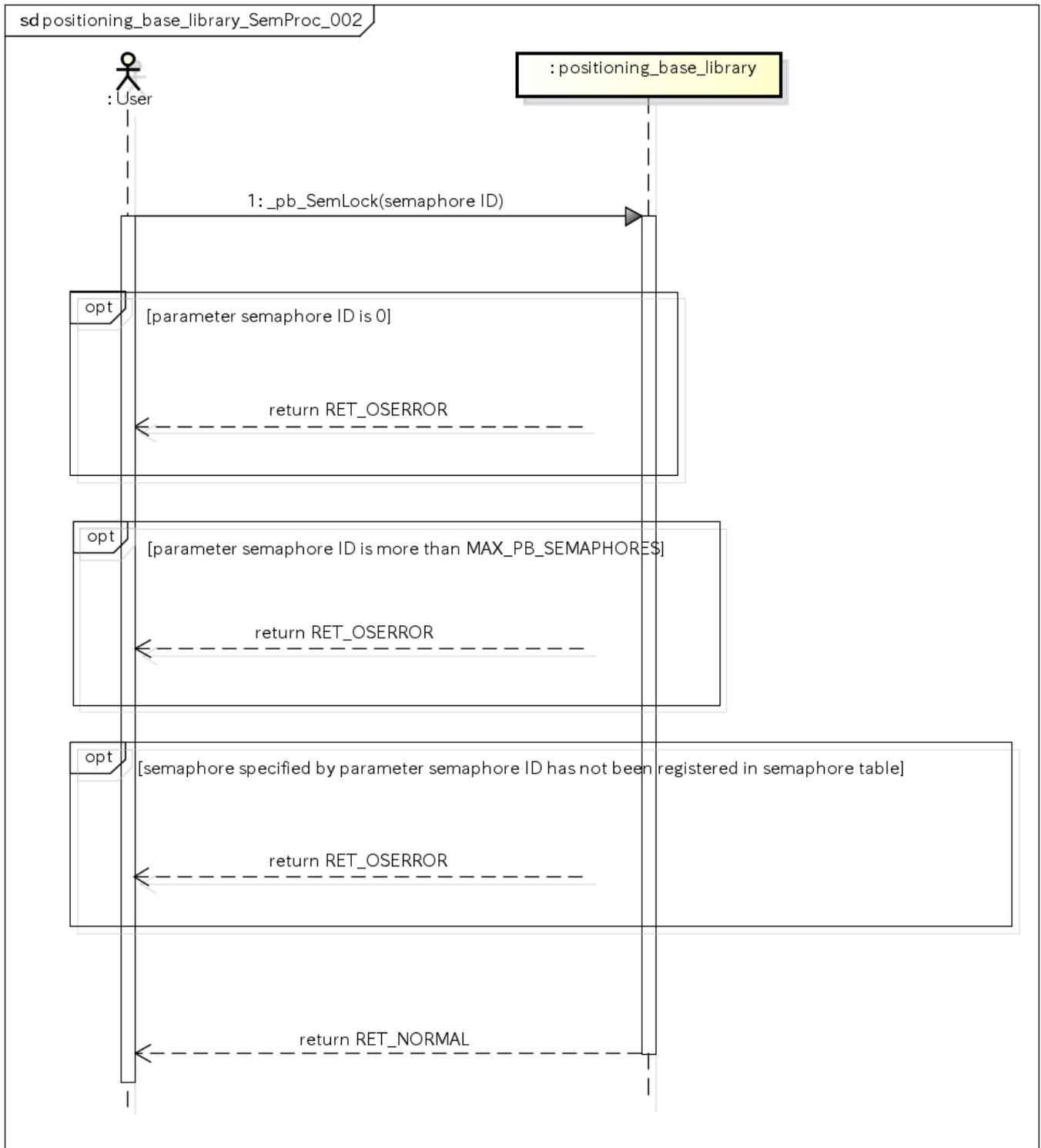
指定されたセマフォIDのセマフォを取得する

*Get semaphore according to semaphore ID*

### シーケンス [\[Sequence\]](#)

指定されたセマフォIDのセマフォを取得するシーケンスを以下に示す

*The sequence of get semaphore according to semaphore ID*



## 指定されたセマフォIDのセマフォを解放する [\[positioning\\_base\\_library\\_SemaphoreAPI\\_003\]](#)

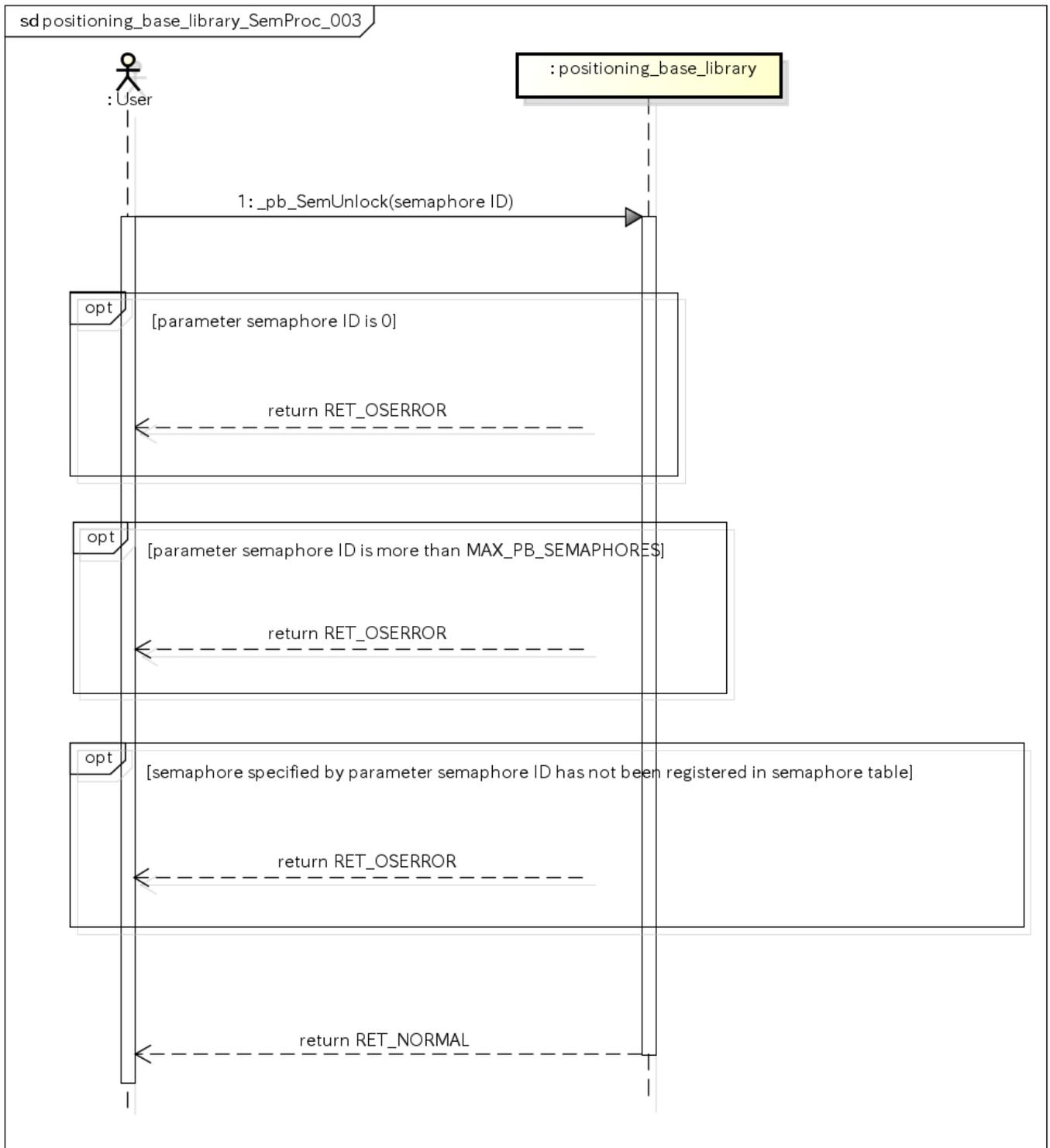
### 概要 [\[Overview\]](#)

指定されたセマフォIDのセマフォを解放する

*Release semaphore according to semaphore ID*

### シーケンス [\[Sequence\]](#)

指定されたセマフォIDのセマフォを解放するシーケンスを以下に示す



## 共有データ領域作成 [[positioning\\_base\\_library\\_ShareDataAPI\\_001](#)]

### 概要 [[Overview](#)]

共有データ領域作成

[Create share data](#)

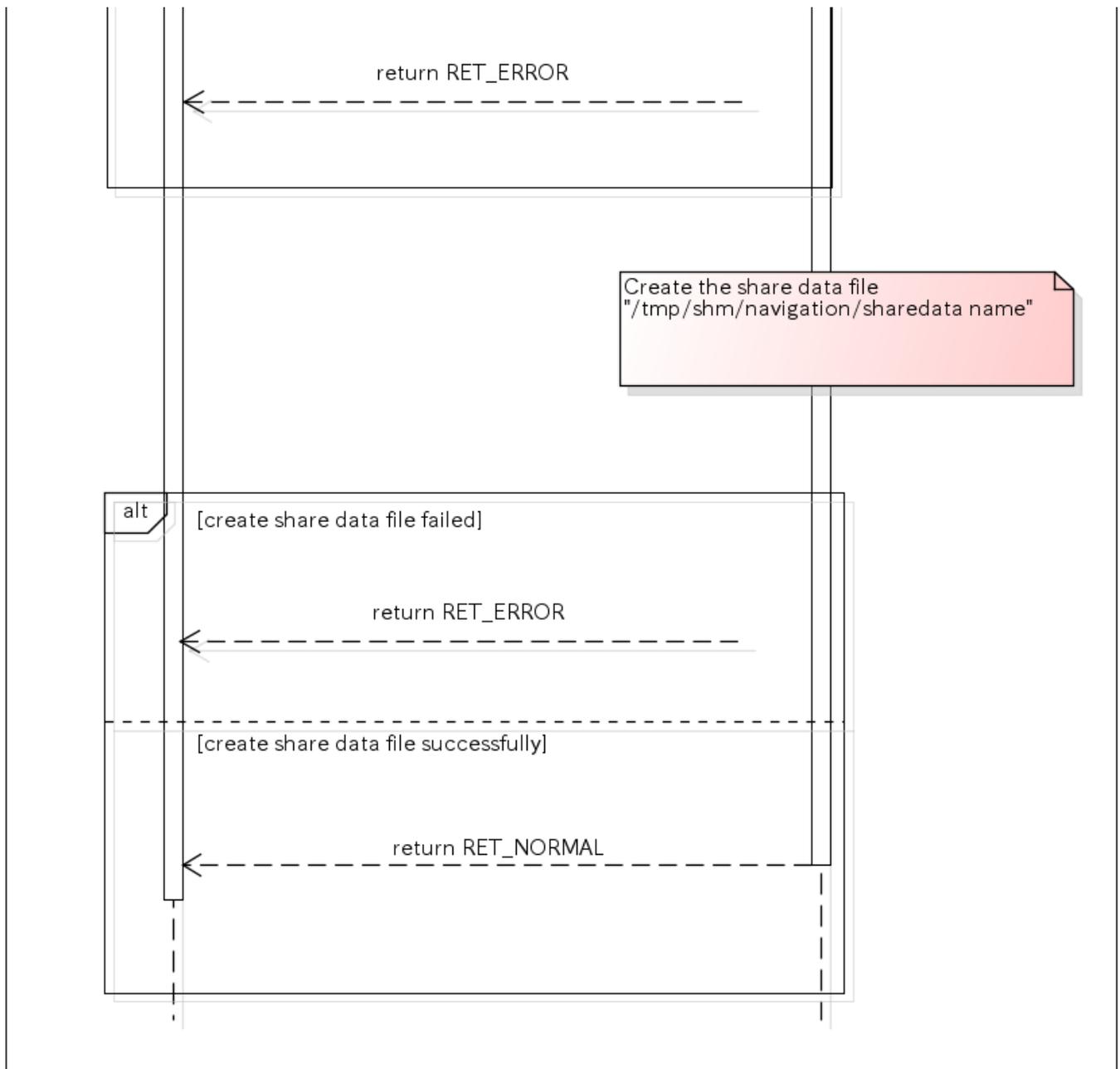
### シーケンス [[Sequence](#)]

共有データ領域作成シーケンスを以下に示す

*The sequence of create share data*







## 共有データ領域リンク [\[positioning\\_base\\_library\\_ShareDataAPI\\_002\]](#)

### 概要 [\[Overview\]](#)

共有データ領域リンク

*Link share data*

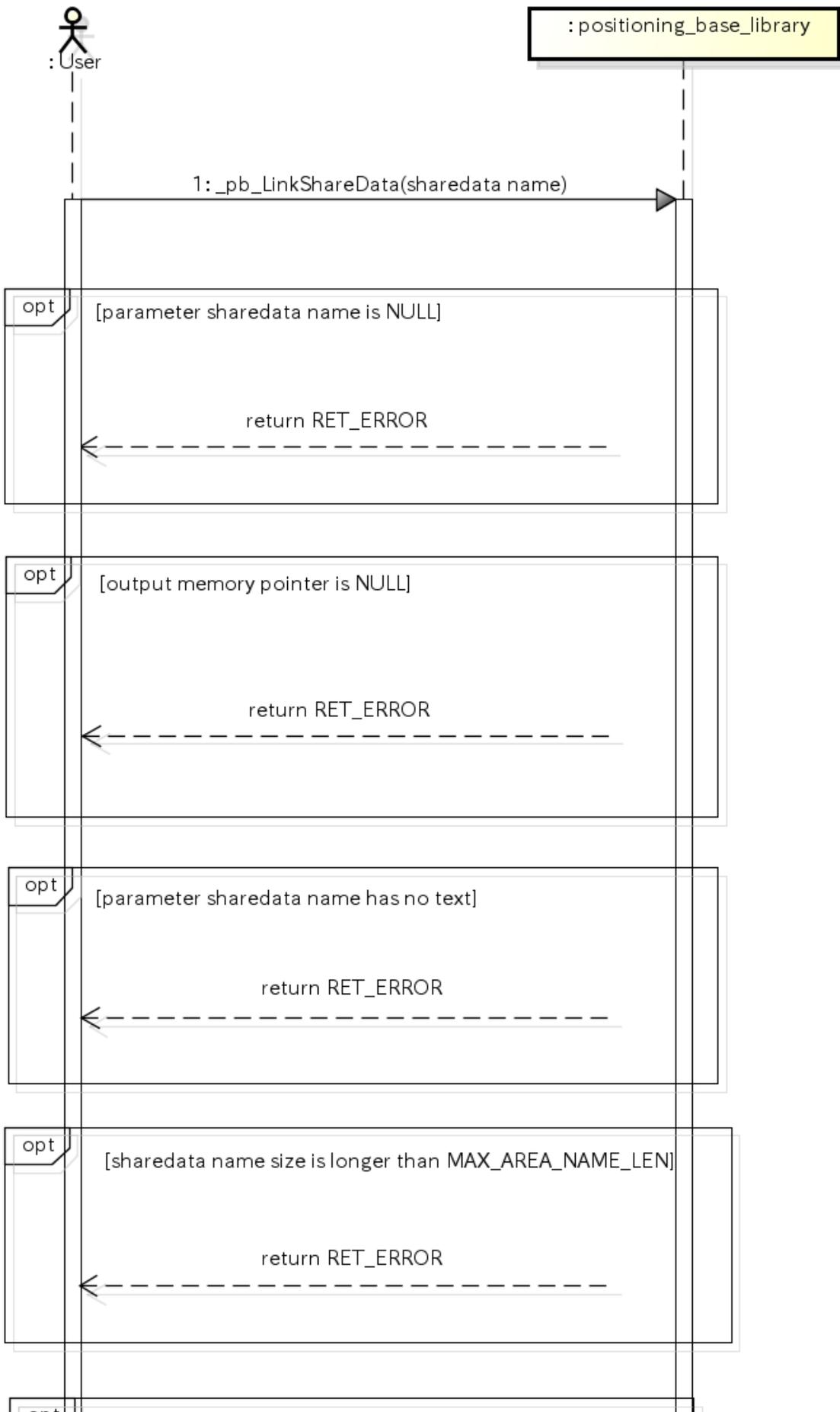
### シーケンス [\[Sequence\]](#)

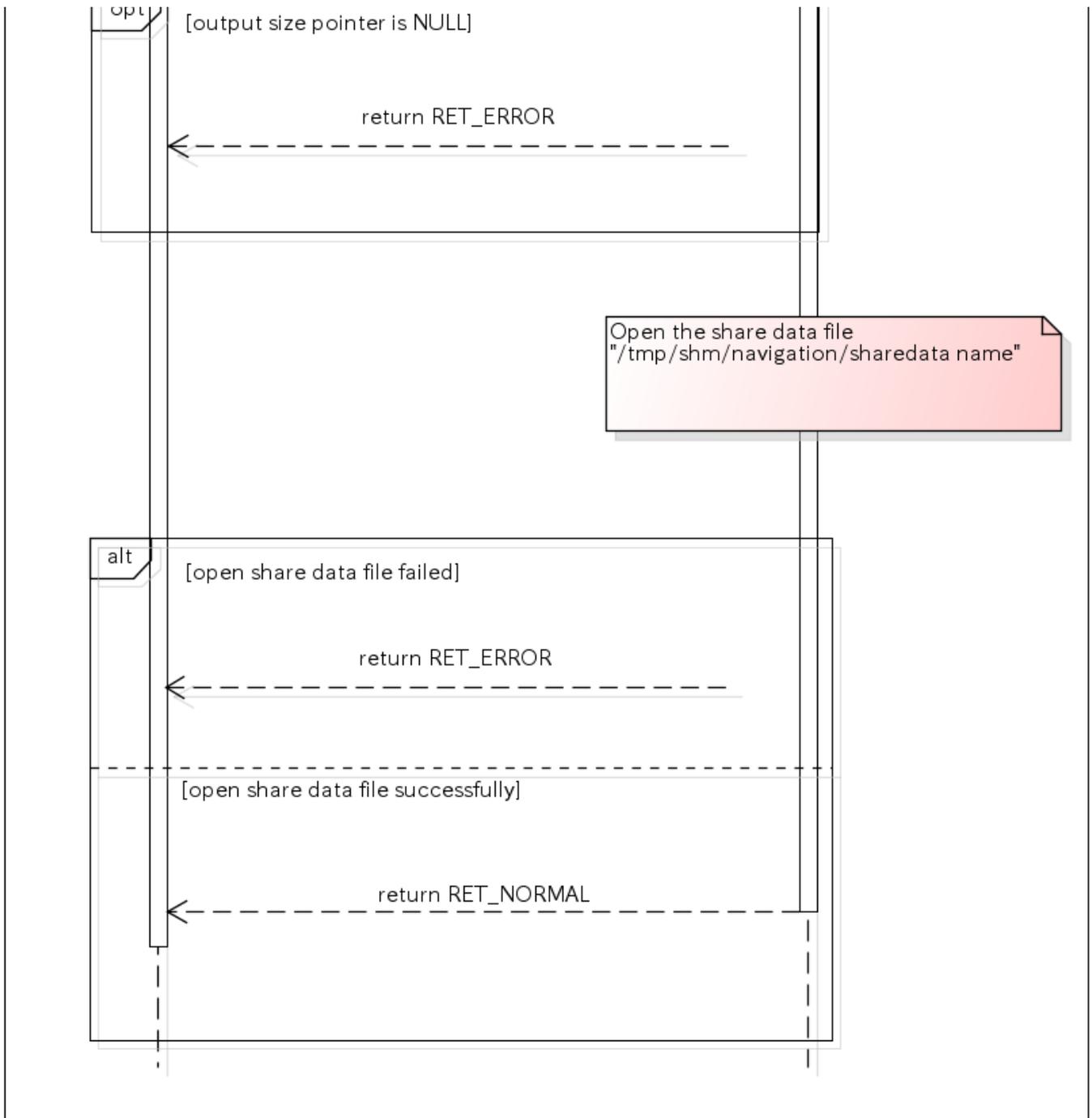
共有データ領域リンクシーケンスを以下に示す。

*The sequence of link share data*



sdpositioning\_base\_library\_ShmProc\_002





## タイマ起動指示 [\[positioning\\_base\\_library\\_TimerAPI\\_001\]](#)

### 概要 [\[Overview\]](#)

タイマ起動指示

*Start timer*

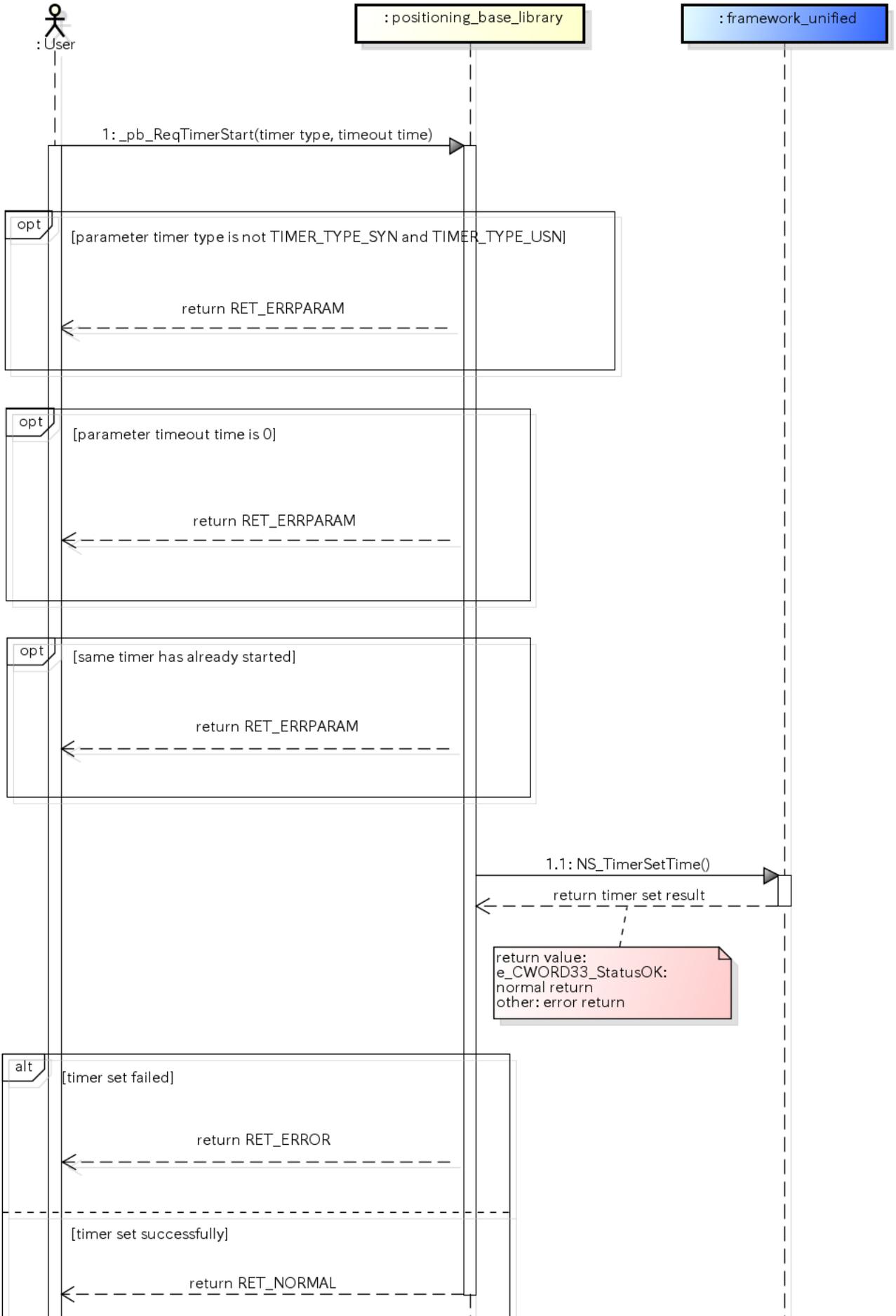
### シーケンス [\[Sequence\]](#)

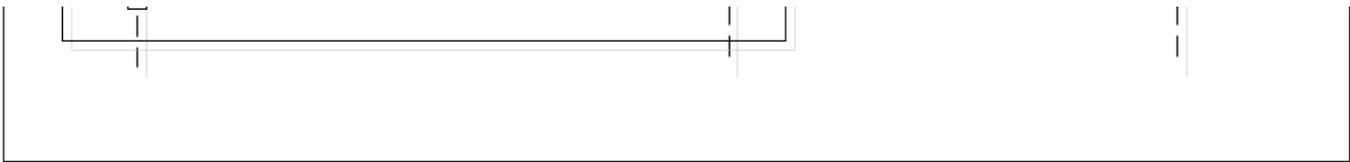
タイマ起動指示シーケンスを以下に示す

*The sequence of start timer*



sdpositioning\_base\_library\_TimerProc\_001





## タイマ停止指示 [positioning\_base\_library\_TimerAPI\_002]

### 概要 [Overview]

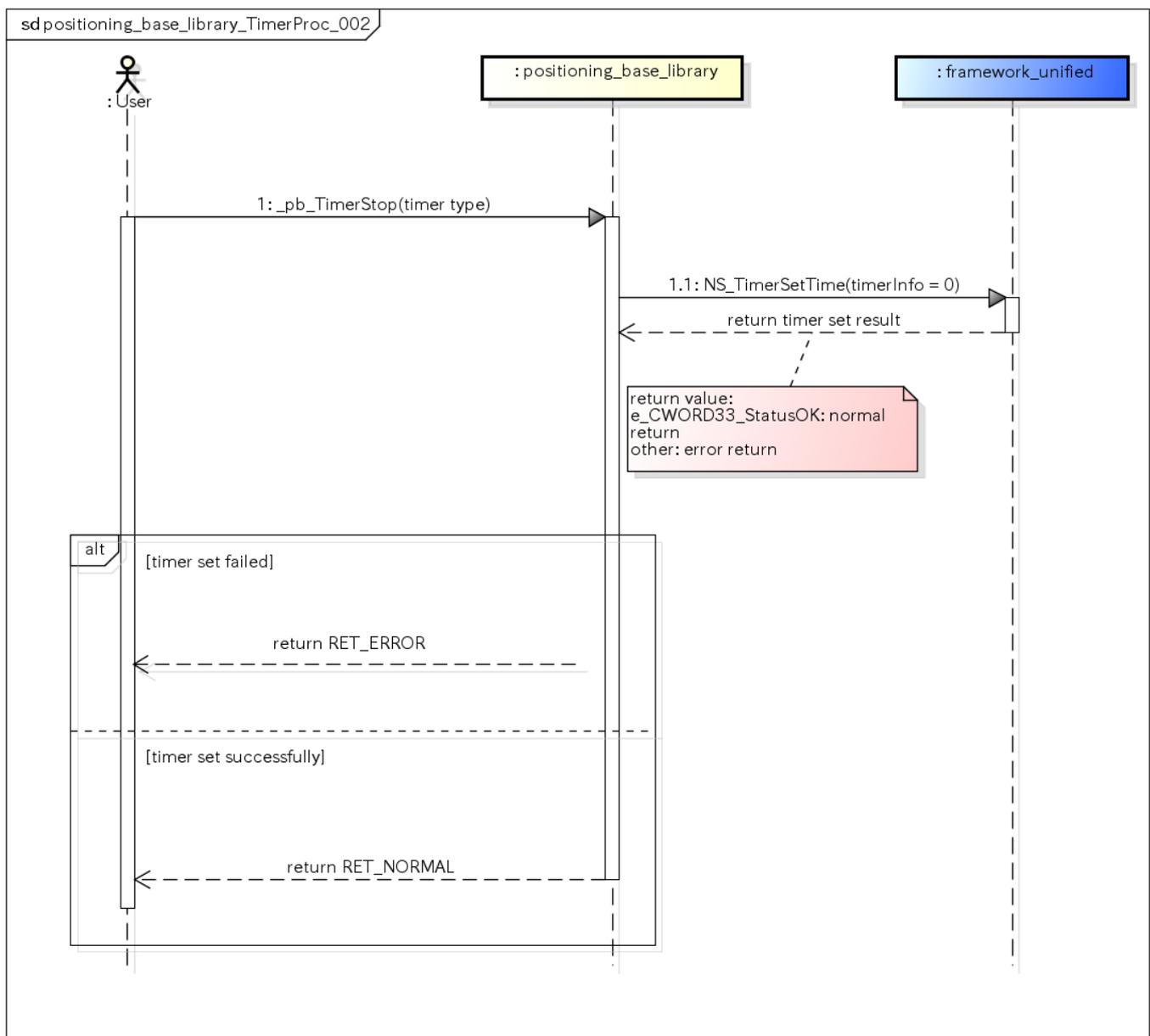
タイマ停止指示

Stop timer

### シーケンス [Sequence]

タイマ停止指示シーケンスを以下に示す

The sequence of stop timer



## 指定した名称でイベントを生成し、そのイベントIDを返却する [\[positioning\\_base\\_library\\_EventAPI\\_001\]](#)

### 概要 [\[Overview\]](#)

指定した名称でイベントを生成し、そのイベントIDを返却する

*Create event according to event name, and get event ID*

### シーケンス [\[Sequence\]](#)

指定した名称でイベントを生成し、そのイベントIDを返却するシーケンスを以下に示す

*The sequence of create event according to event name, and get event ID*



:User

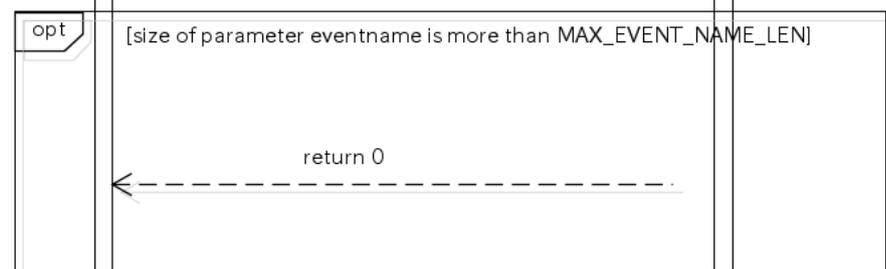
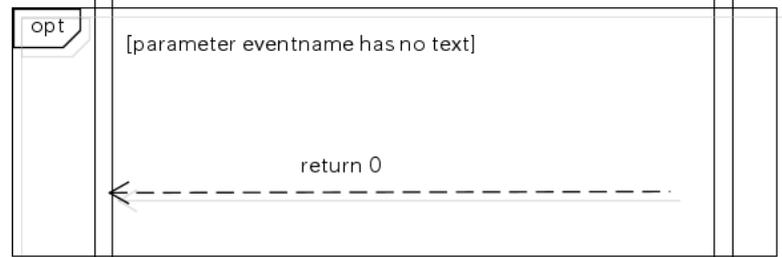
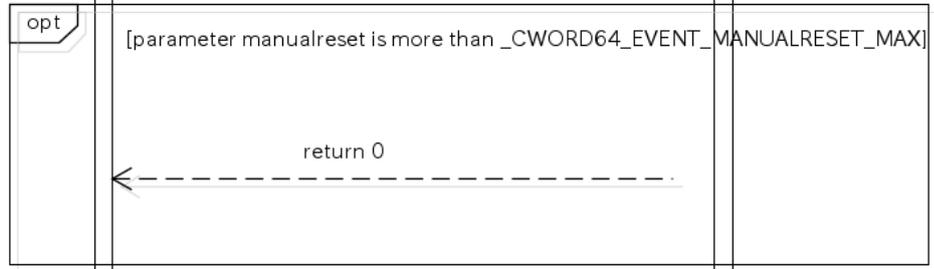
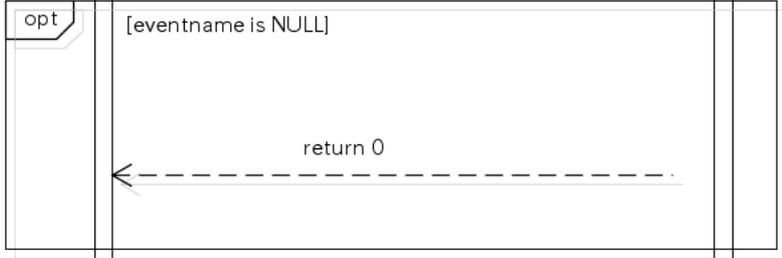


:positioning\_base\_library



:event\_library

1: \_pb\_CreateEvent(eventname)

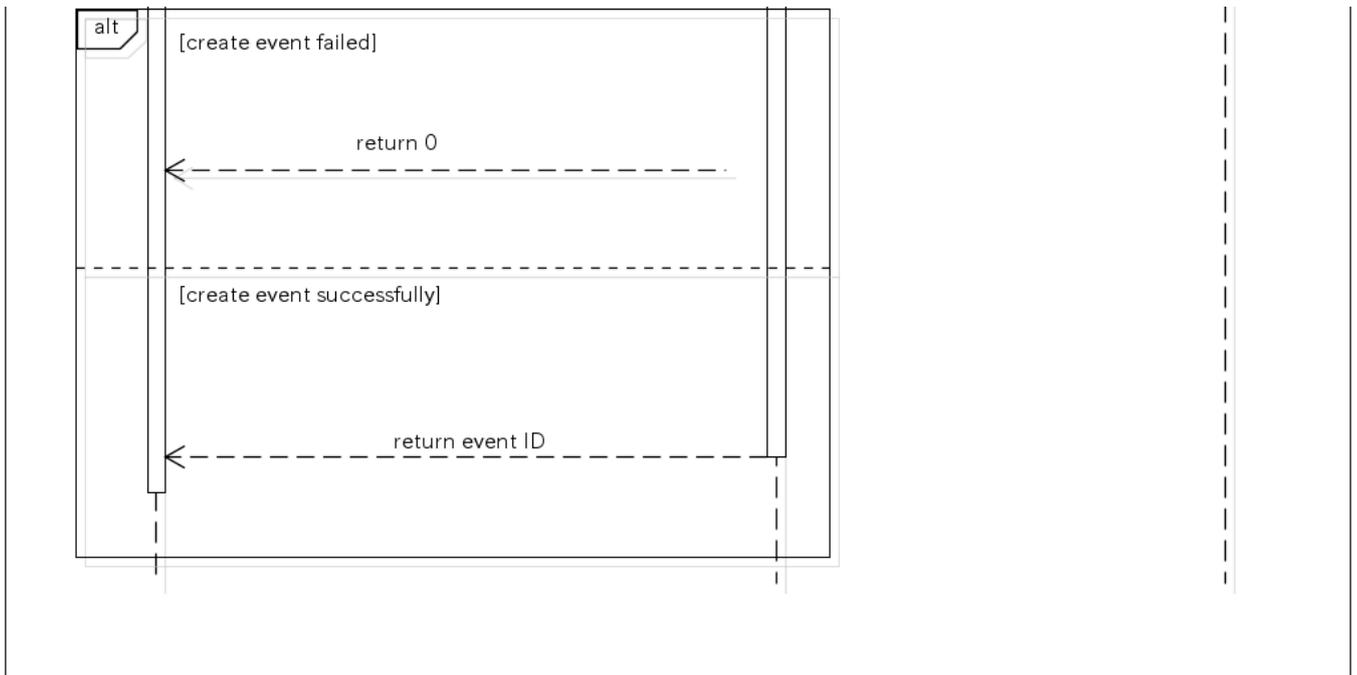


Mock project

1.1: EV\_create\_flag\_auto\_id()

return event create result

Return value:  
EV\_OK: normal return,  
other : error return



**指定されたイベントIDのイベント値が指定された範囲になるまで待つ** [[positioning\\_base\\_library\\_EventAPI\\_002](#)]

**概要** [[Overview](#)]

指定されたイベントIDのイベント値が指定された範囲になるまで待つ

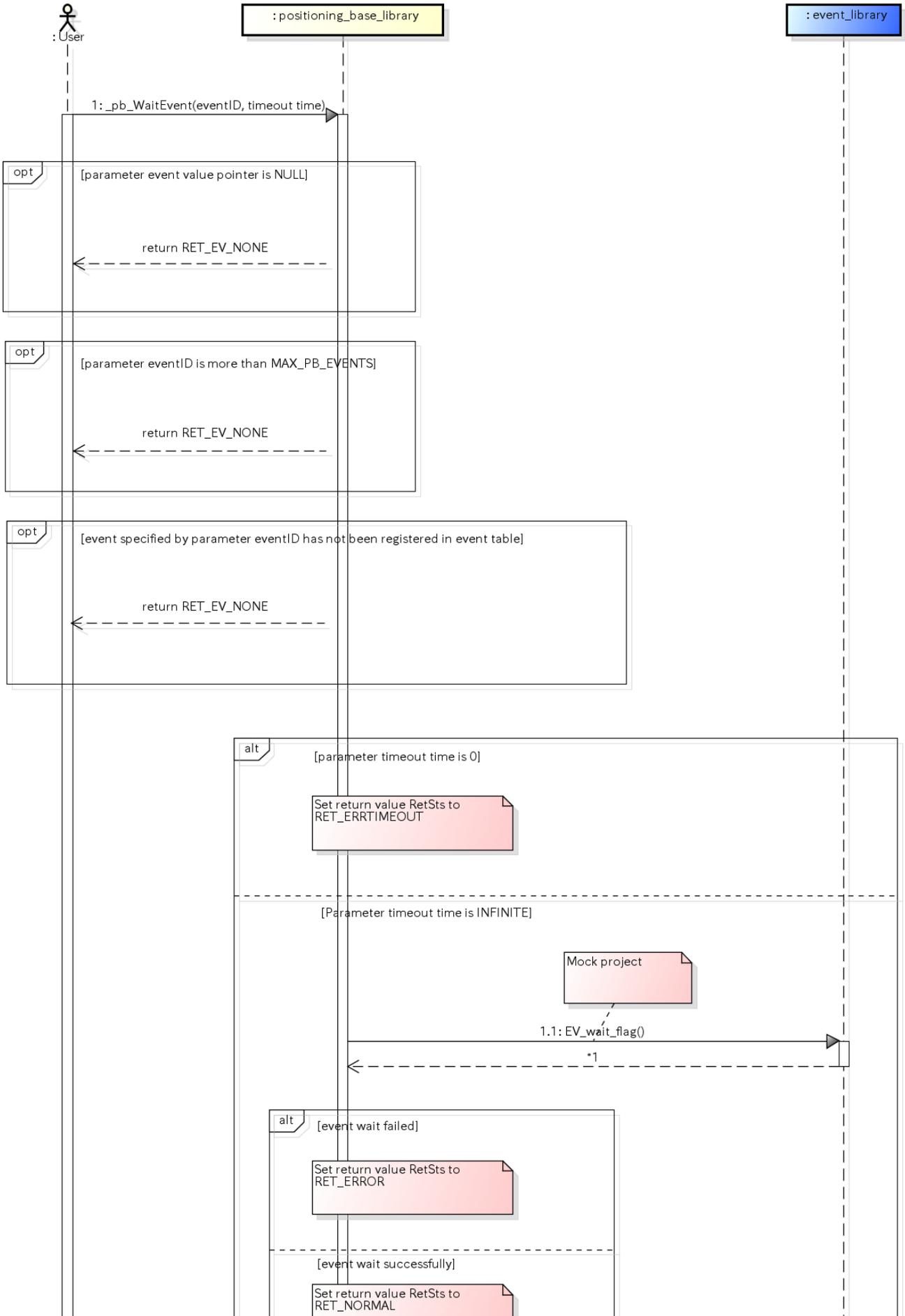
*Wait for the event according to event ID and event value*

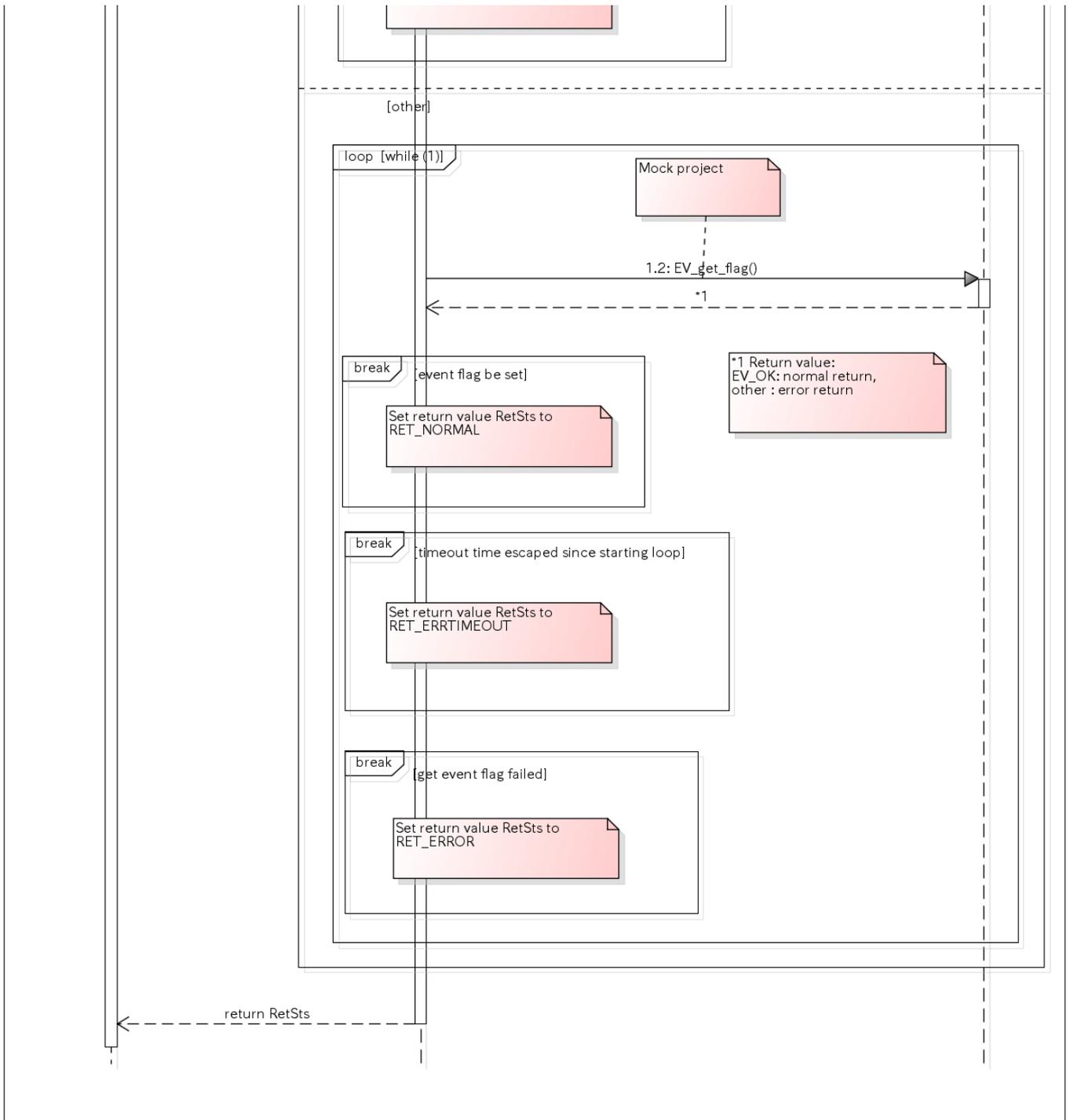
**シーケンス** [[Sequence](#)]

指定されたイベントIDのイベント値が指定された範囲になるまで待つシーケンスを以下に示す

*The sequence of wait for the event according to event ID and event value*







## イベント値を設定する [positioning\_base\_library\_EventAPI\_003]

### 概要 [Overview]

イベント値を設定する

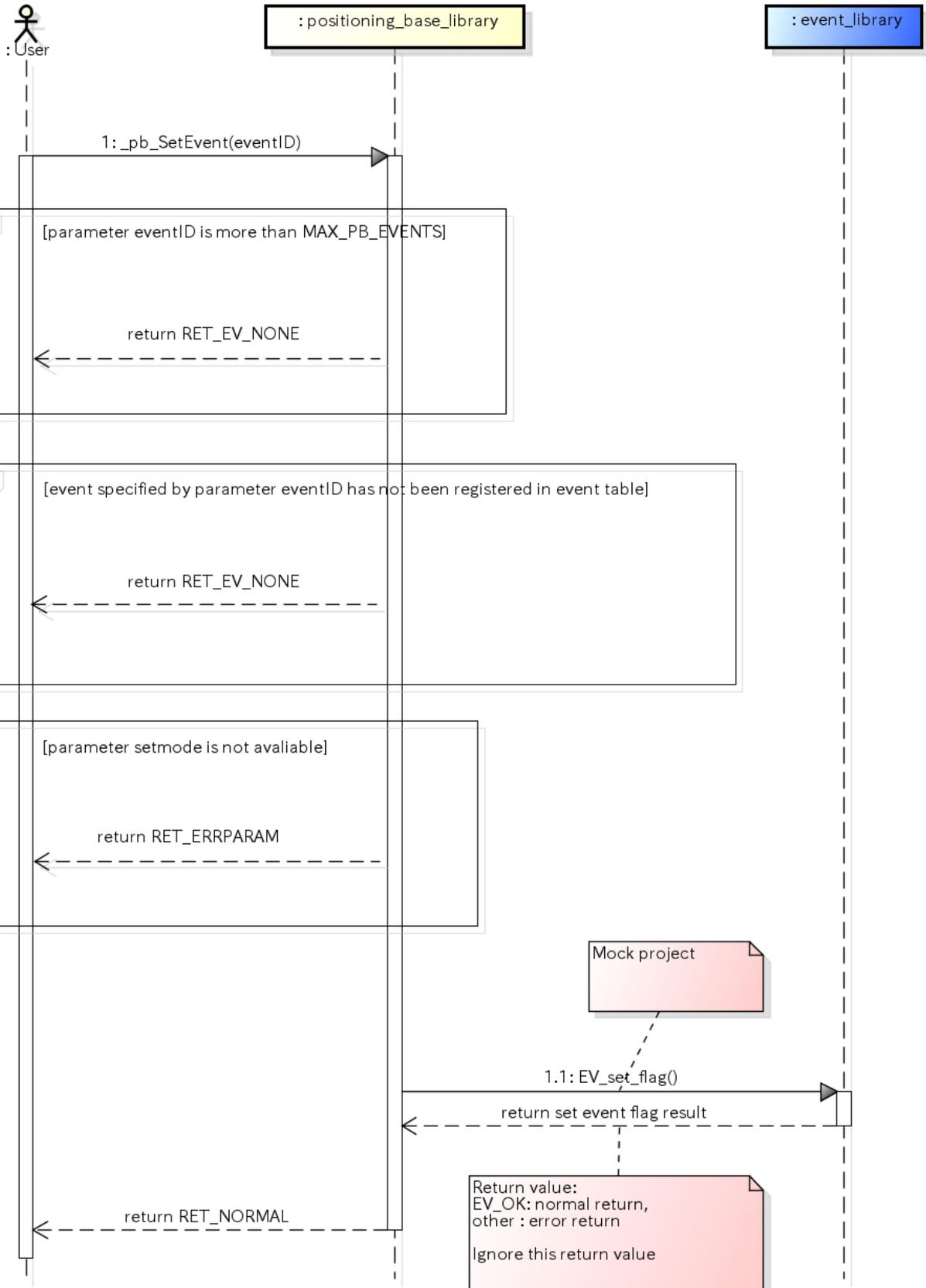
*Set the event*

### シーケンス [Sequence]

イベント値を設定するシーケンスを以下に示す

*The sequence of set the event*

sd positioning\_base\_library\_EventProc\_003



## 指定されたイベントIDのイベントを削除する [\[positioning\\_base\\_library\\_EventAPI\\_004\]](#)

### 概要 [\[Overview\]](#)

指定されたイベントIDのイベントを削除する

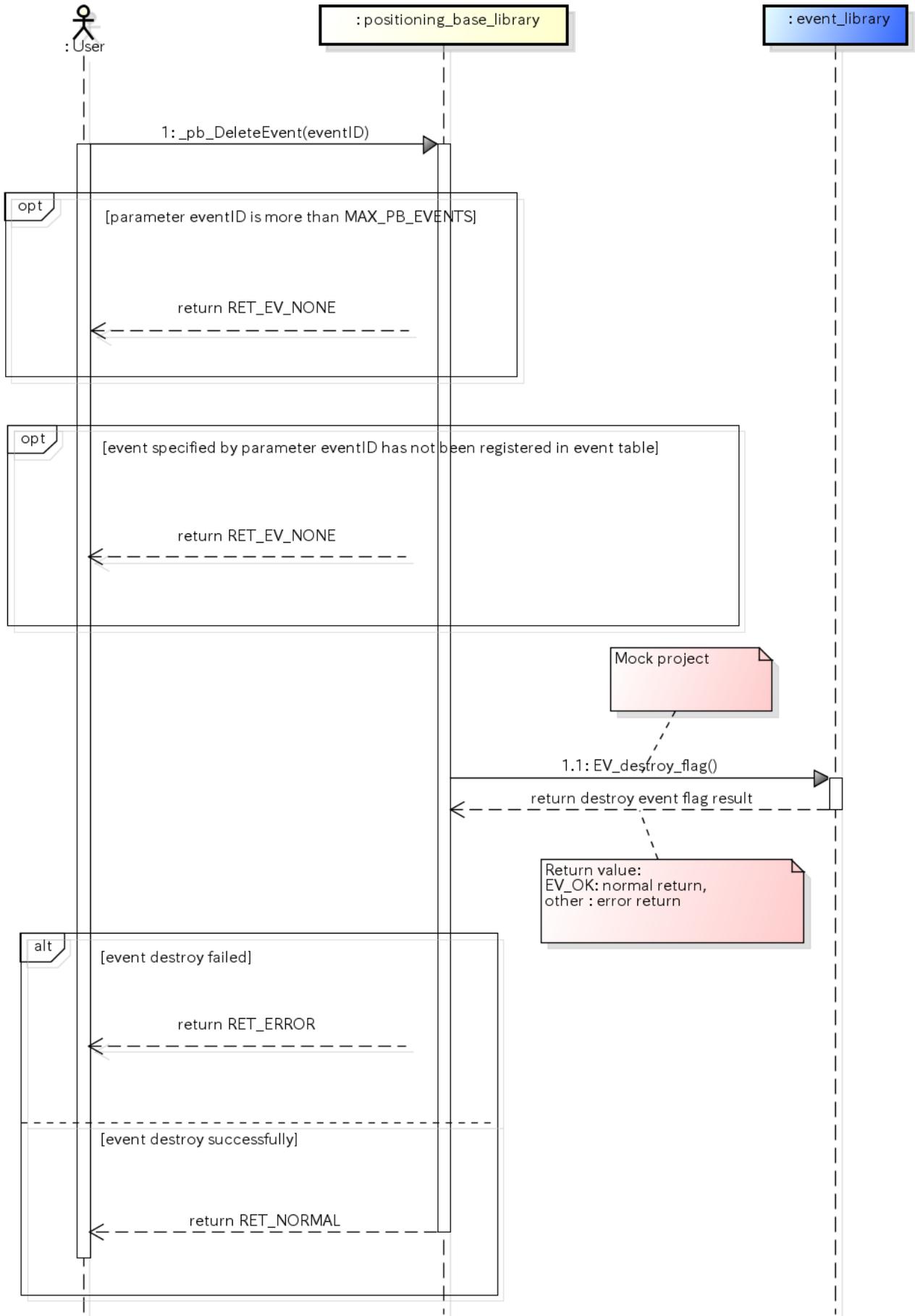
*Delete event according to event ID*

### シーケンス [\[Sequence\]](#)

指定されたイベントIDのイベントを削除するシーケンスを以下に示す

*The sequence of delete event according to event ID*





## dump情報取得(メッセージ) [\[positioning\\_base\\_library\\_GetDbgAPI\\_001\]](#)

### 概要 [\[Overview\]](#)

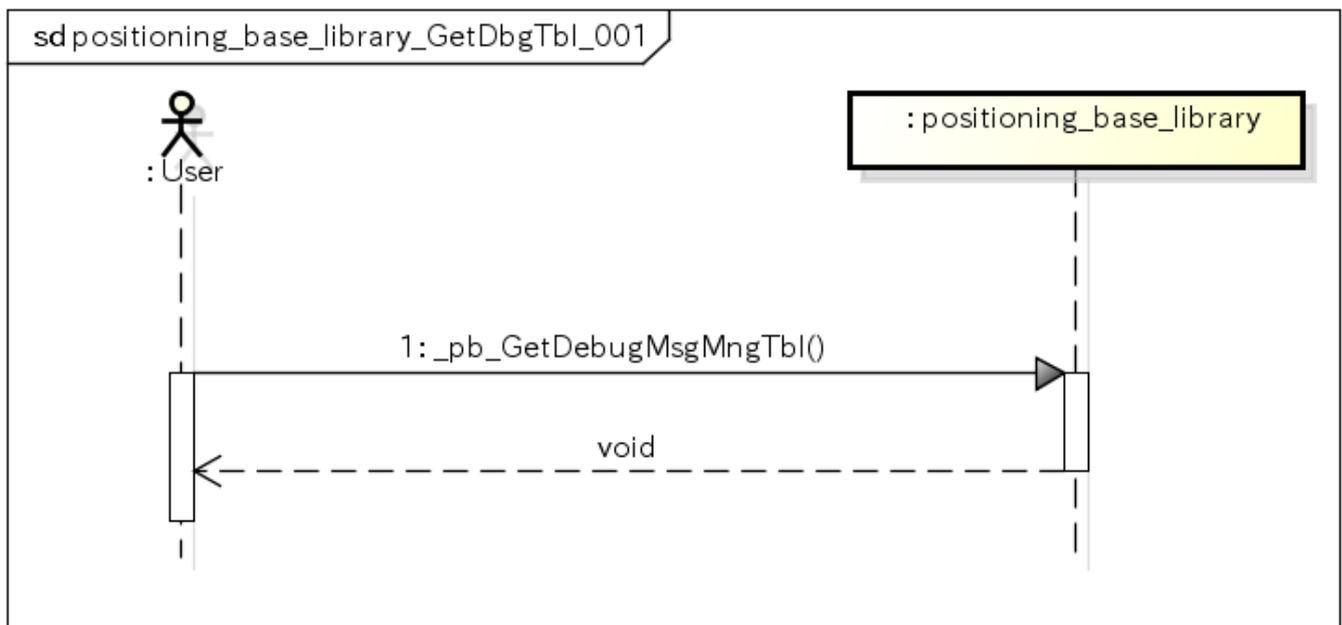
dump情報取得(メッセージ)

*Get dump information(message)*

### シーケンス [\[Sequence\]](#)

dump情報取得(メッセージ)シーケンスを以下に示す

*The sequence of get dump information(message)*



## dump情報取得(mutex) [\[positioning\\_base\\_library\\_GetDbgAPI\\_002\]](#)

### 概要 [\[Overview\]](#)

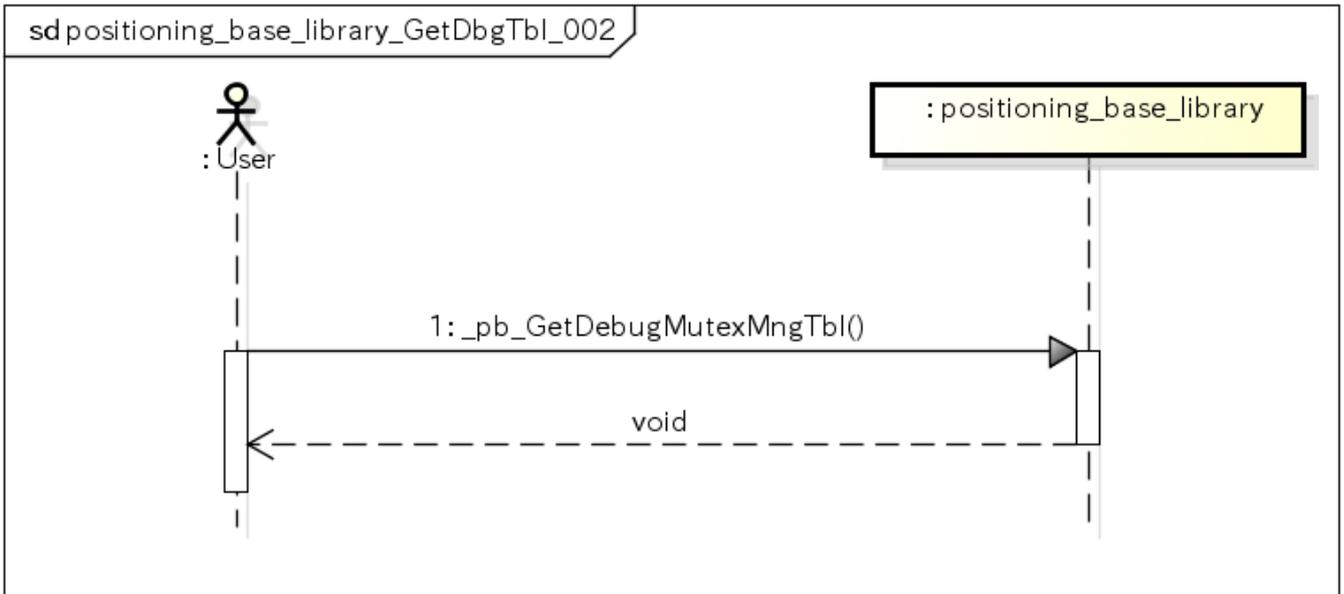
dump情報取得(mutex)

*Get dump information(mutex)*

### シーケンス [\[Sequence\]](#)

dump情報取得(mutex)シーケンスを以下に示す

*The sequence of get dump information(mutex)*



## dump情報取得(タイマ) [\[positioning\\_base\\_library\\_GetDbgAPI\\_003\]](#)

### 概要 [\[Overview\]](#)

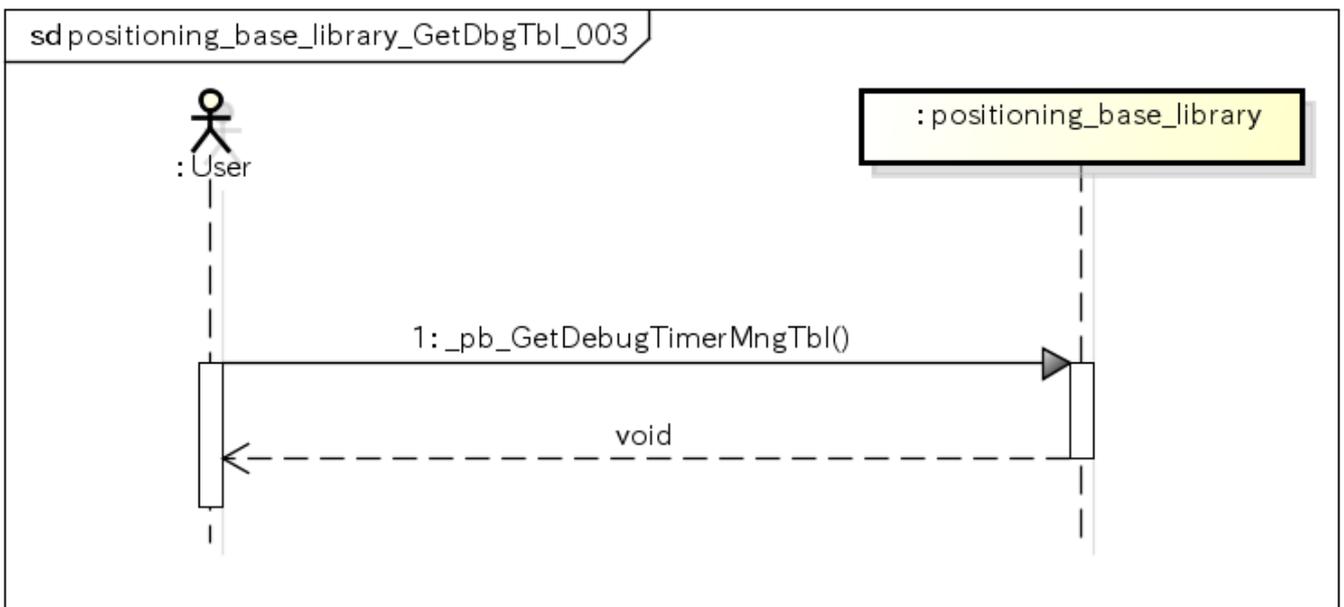
dump情報取得(タイマ)

*Get dump information(timer)*

### シーケンス [\[Sequence\]](#)

dump情報取得(タイマ)シーケンスを以下に示す

*The sequence of get dump information(timer)*



## dump情報取得(イベント) [\[positioning\\_base\\_library\\_GetDbgAPI\\_004\]](#)

## 概要 [Overview]

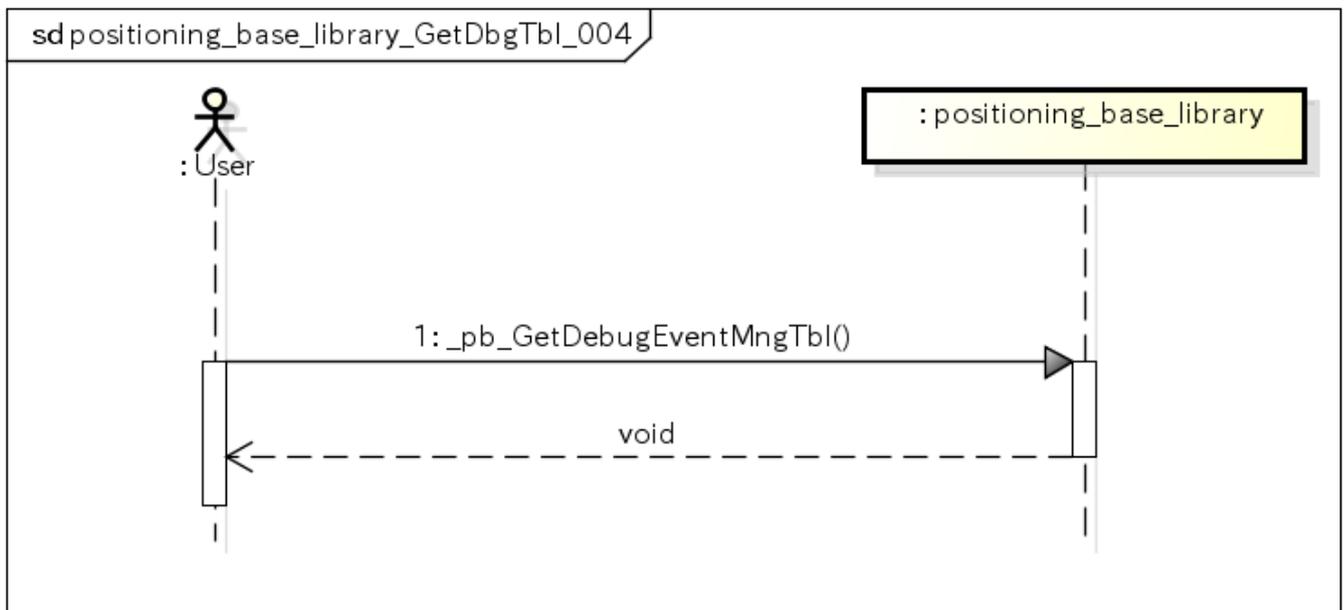
dump情報取得(イベント)

*Get dump information(event)*

## シーケンス [Sequence]

dump情報取得(イベント)シーケンスを以下に示す

*The sequence of get dump information(event)*



## dump情報取得(memory) [positioning\_base\_library\_GetDbgAPI\_005]

### 概要 [Overview]

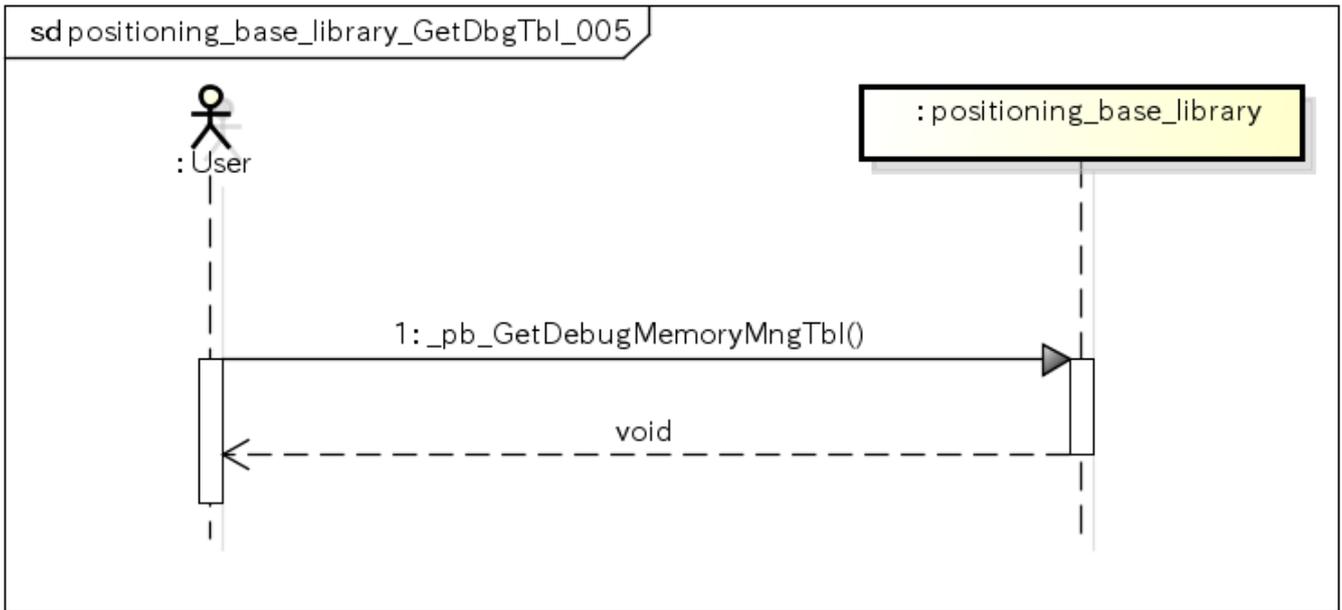
dump情報取得(memory)

*Get dump information(memory)*

### シーケンス [Sequence]

dump情報取得(memory)シーケンスを以下に示す

*The sequence of get dump information(memory)*



## dump情報取得(other) [[positioning\\_base\\_library\\_GetDbgAPI\\_006](#)]

### 概要 [[Overview](#)]

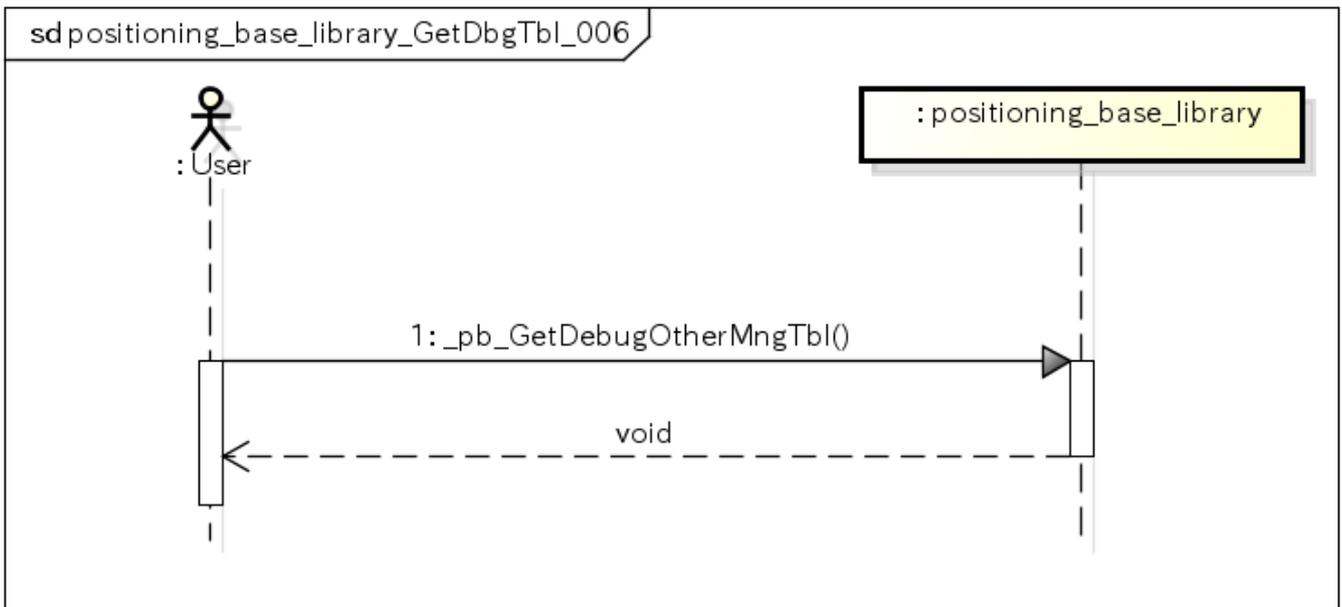
dump情報取得(other)

*Get dump information(other)*

### シーケンス [[Sequence](#)]

dump情報取得(other)シーケンスを以下に示す

*The sequence of get dump information(other)*



## 自スレッドのアプリケーションハンドルを取得する [positioning\_base\_library\_HandleAPI\_001]

### 概要 [Overview]

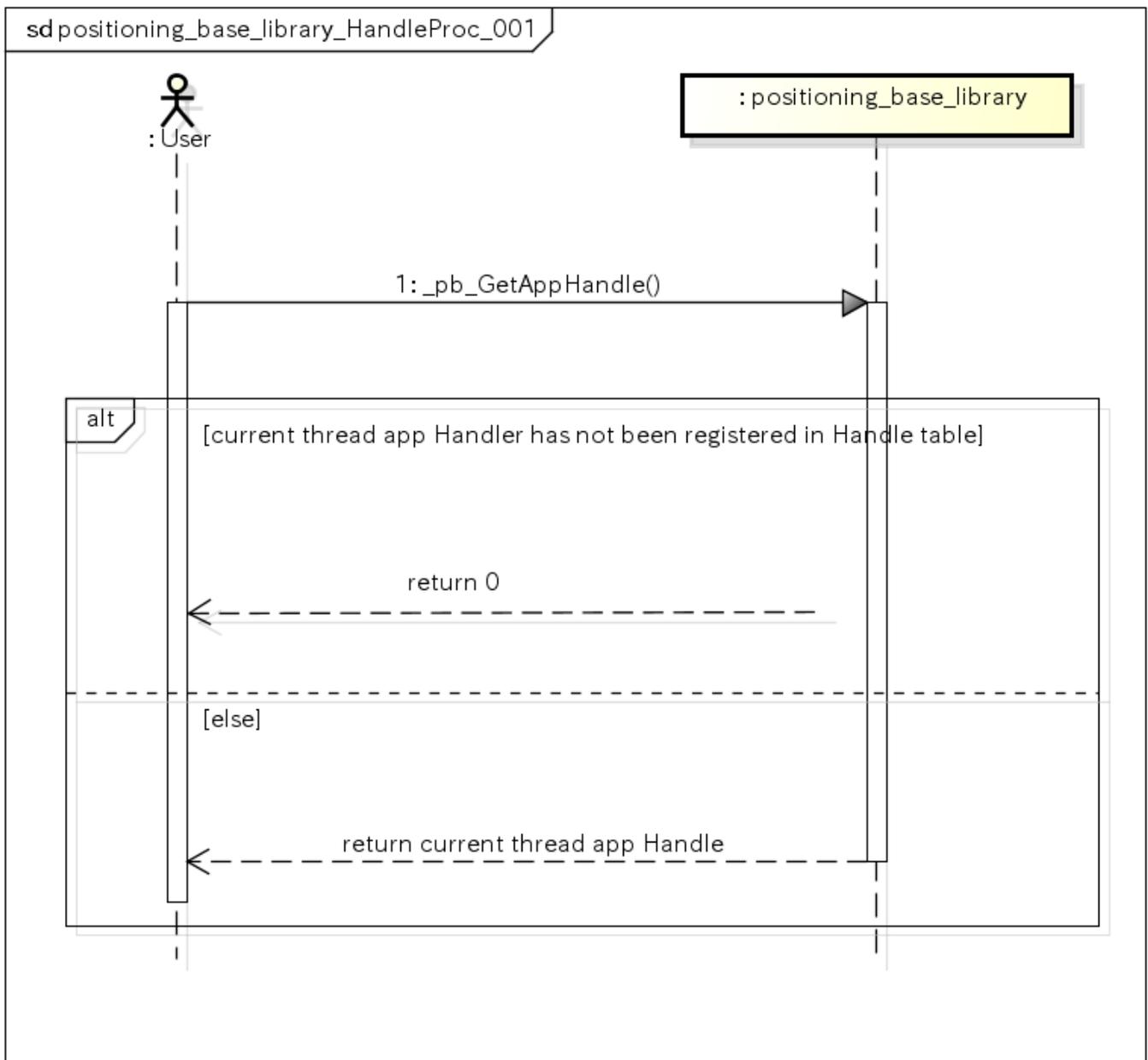
自スレッドのアプリケーションハンドルを取得する

*Get application handle*

### シーケンス [Sequence]

自スレッドのアプリケーションハンドルを取得するシーケンスを以下に示す

*The sequence of get application handle*



## 自スレッドのアプリケーションハンドルを設定する [positioning\_base\_library\_HandleAPI\_002]

## 概要 [Overview]

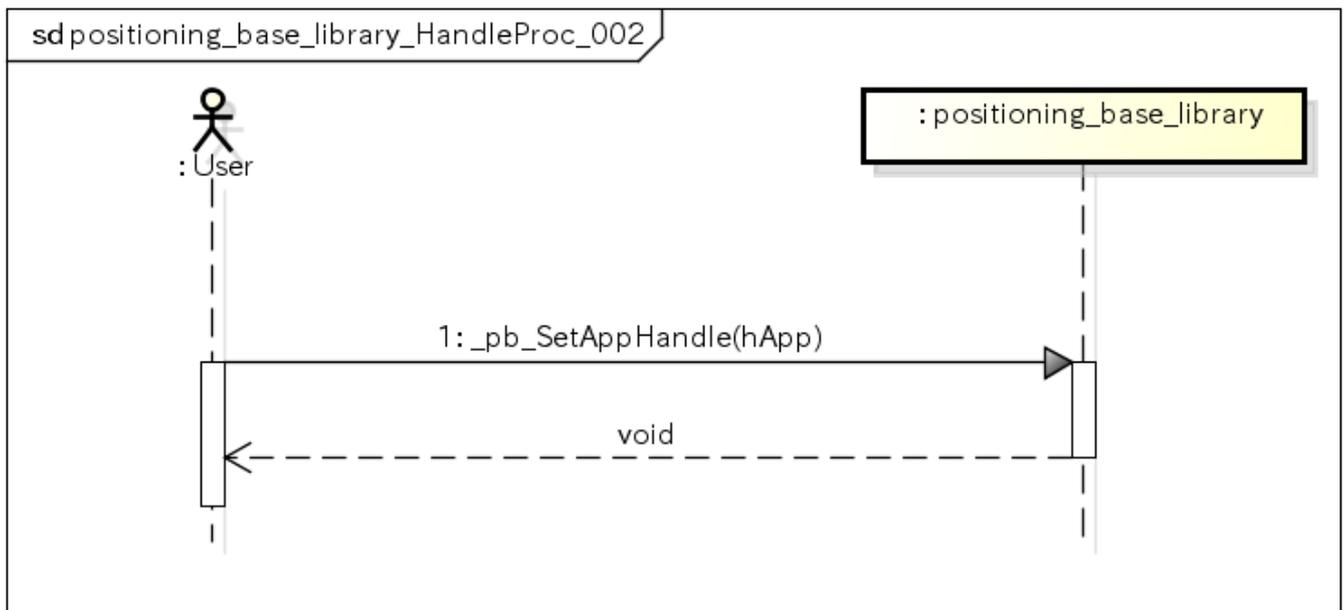
自スレッドのアプリケーションハンドルを設定する

*Set application handle*

## シーケンス [Sequence]

自スレッドのアプリケーションハンドルを設定するシーケンスを以下に示す

*The sequence of set application handle*



## positioning\_base\_libraryの初期化 [Initialize positioning\_base\_library]

### 概要 [Overview]

positioning\_base\_libraryの初期化

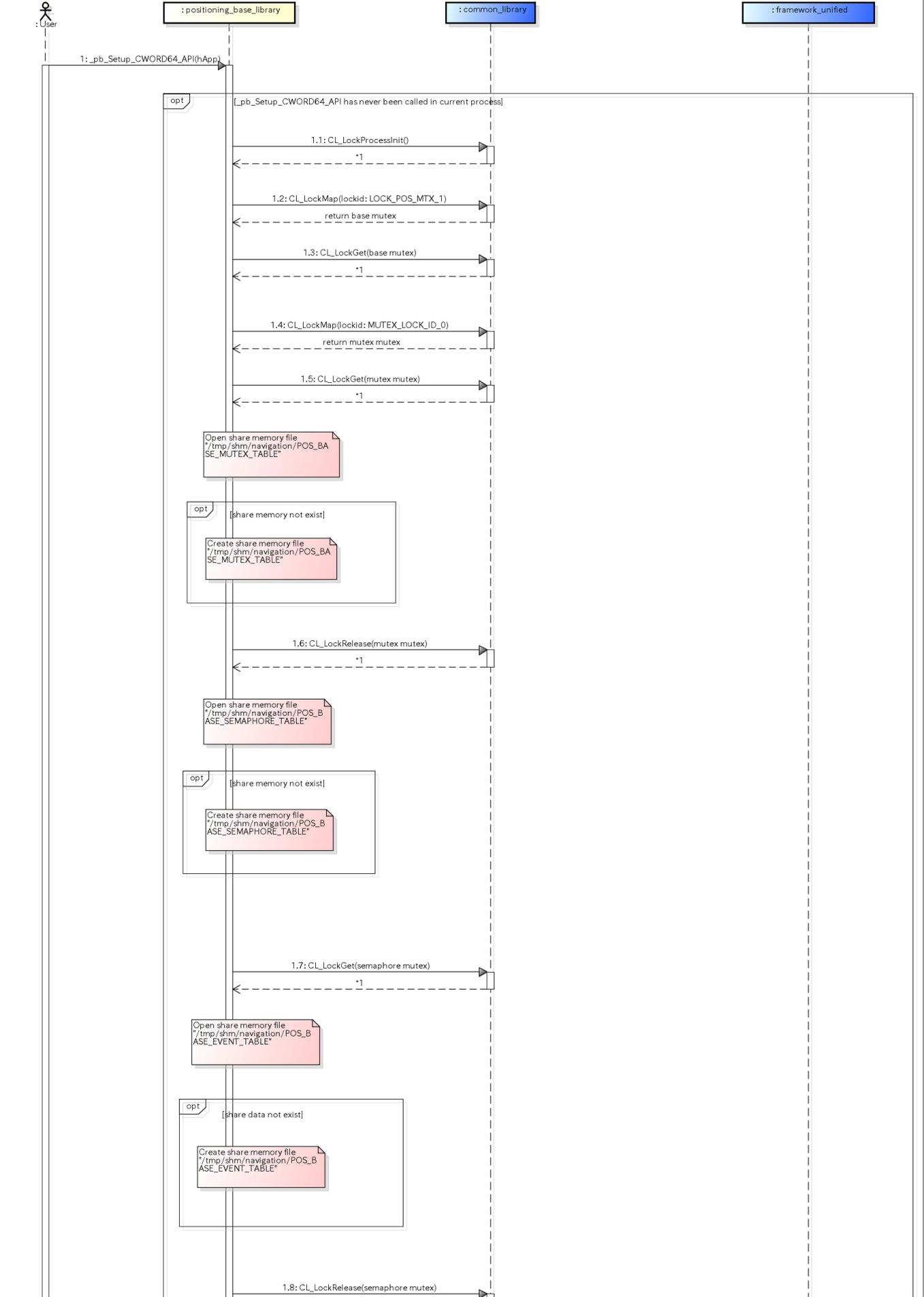
*Initialize positioning\_base\_library*

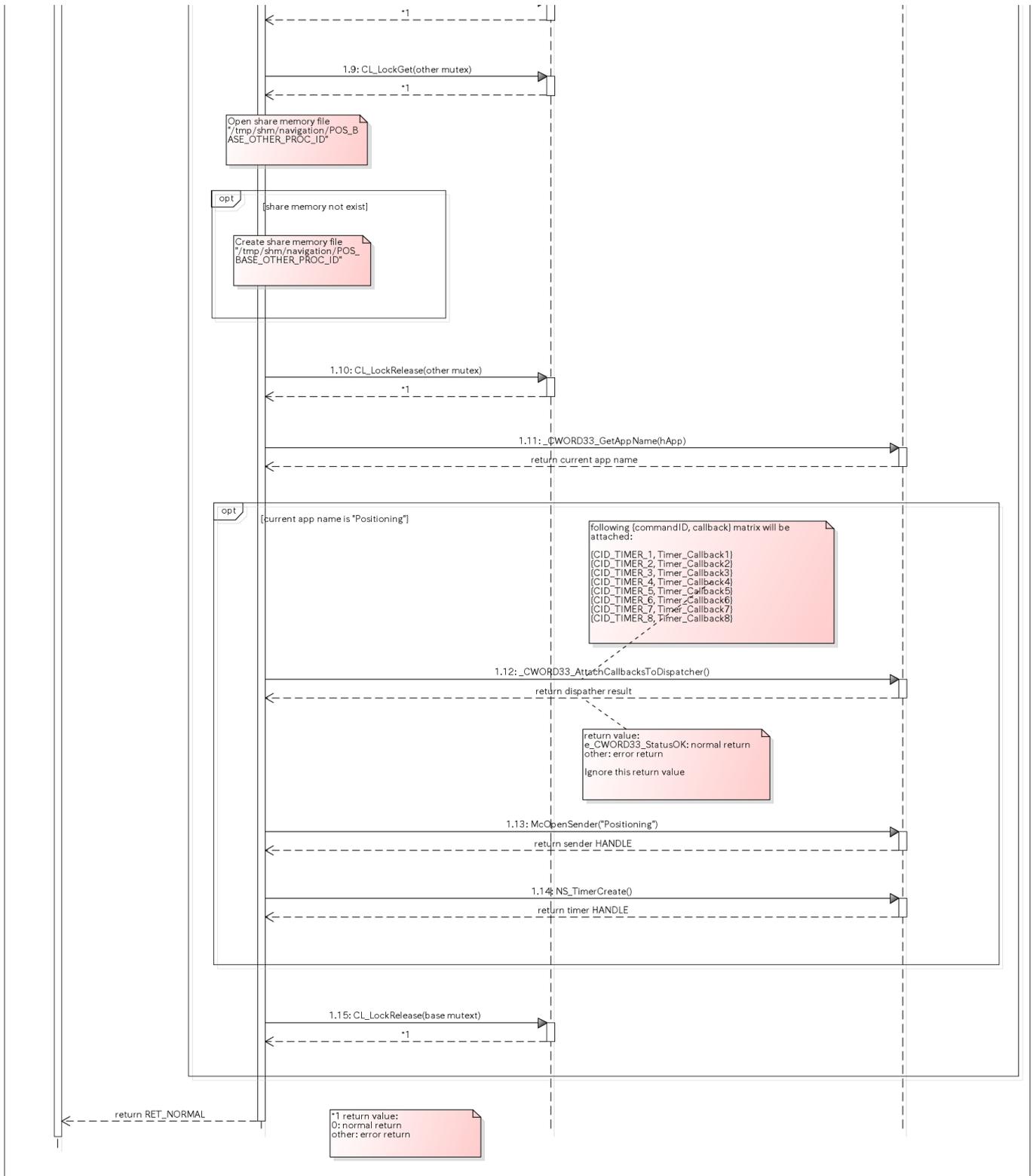
### シーケンス [Sequence]

positioning\_base\_libraryの初期化シーケンスを以下に示す

*The sequence of positioning\_base\_library initialization*







## Base API終了処理 [Teardown Base API]

### 概要 [Overview]

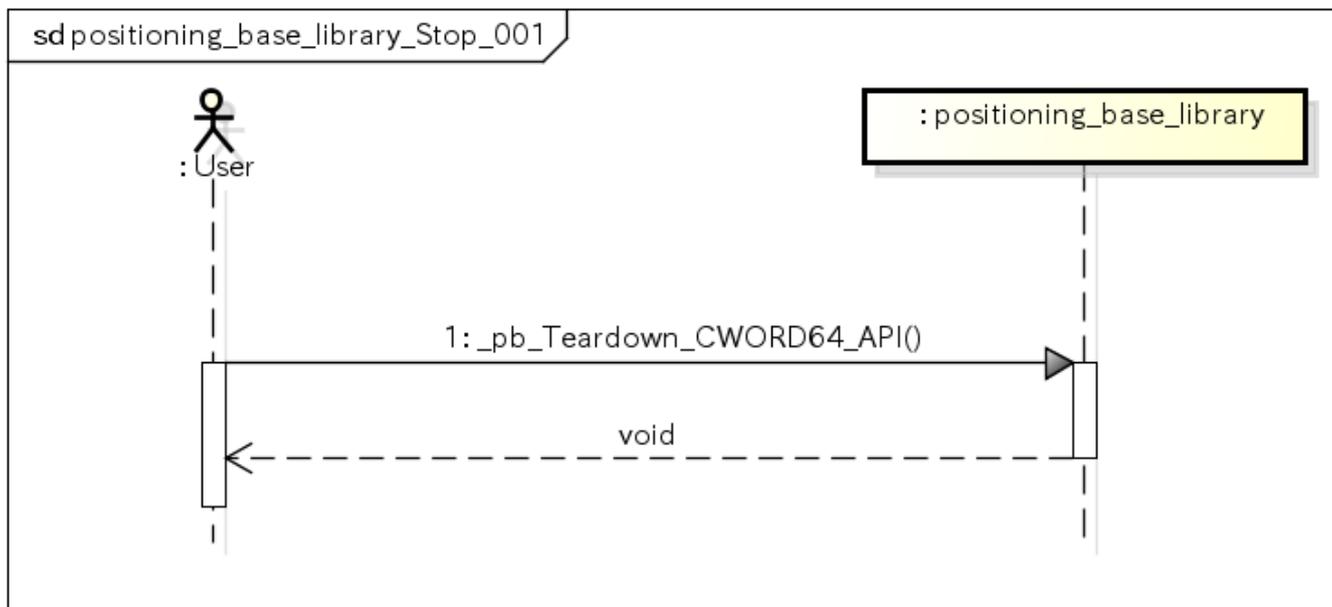
Base API終了処理

[Teardown Base API](#)

## シーケンス [Sequence]

Base API終了処理シーケンスを以下に示す

*The sequence of Teardown Base API as follows*



## 該当スレッドを終了させる [Stop thread]

### 概要 [Overview]

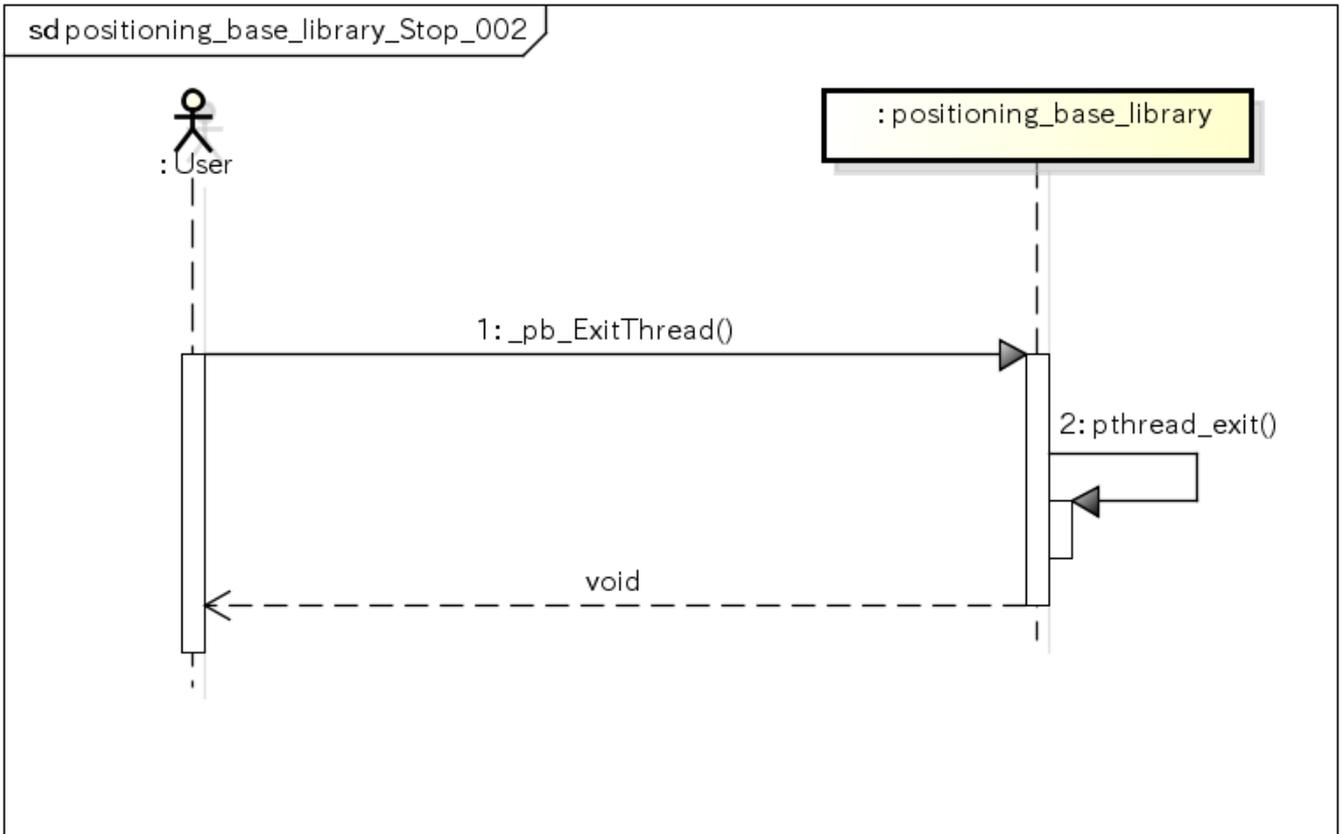
該当スレッドを終了させる

*Stop thread*

### シーケンス [Sequence]

該当スレッドを終了させるシーケンスを以下に示す

*The sequence of stop thread as follows*



## 該当プロセスを終了させる *[Stop process]*

### 概要 *[Overview]*

該当プロセスを終了させる

*Stop process*

### シーケンス *[Sequence]*

該当プロセスを終了させるシーケンスを以下に示す

*The sequence of stop process as follows*

sd positioning\_base\_library\_Stop\_003

