

JAVA プログラム呼び出し関数仕様書

名前

GetProxyJNI - JNI のインターフェース取得

書式

```
#include <proxyjni/IProxyJNI.h>
IProxyJNI* GetProxyJNI()
```

説明

GetProxyJNI()は、JNI の呼び出しのインターフェースを取得します。
インターフェースとして次の関数が定義されている。
関数の呼び出し方法は、インターフェース名(IProxyJNI)_関数名()で、
第一引数に、インターフェース構造体へのポインターを渡します。

Init()

JNI の初期処理を行う

classStaticMethodCall()

java ファイル内スタティックメソッドを実行する

getInstance()

java ファイル内のクラスインスタンスを生成する

classObjectMethodCall()

java ファイル内のクラスメソッドを実行する

end()

JNI の終了処理を行う

戻り値

JNI の呼び出しインターフェース構造体へのポインター

名前

`IProxyJNI_init` - JNI の初期化

書式

```
#include <proxyjni/IProxyJNI.h>
void IProxyJNI_init(IProxyJNI* pThis)
```

説明

`IProxyJNI_init()`は、JNI の初期化処理を行う。

戻り値

無し

名前

IProxyJNI_classStaticMethodCall - クラススタティックメソッド実行

書式

```
#include <proxyjni/IProxyJNI.h>

void IProxyJNI_classStaticMethodCall(IProxyJNI* pThis, JNIReturn* pJNIReturn, char* pClassName,
char* pFuncName, JNIParams* pJNIParams)
```

説明

IProxyJNI_classStaticMethodCall()は、次の処理を実行する。

- ・パラメータ構造体に従って、JNI 用に変換する(`pJNIParams` \neq `NULL`)
- ・java クラス内のスタティックメソッドを実行する
- ・実行結果を、JNI 用から戻り構造体へ変換する

戻り値

無し

名前

IProxyJNI_getInstance - java ファイル内のクラスのインスタンス生成

書式

```
#include <proxyjni/IProxyJNI.h>

void* IProxyJNI_getInstance(IProxyJNI* pThis, char* pClassName, JNIParams* pJNIParams)
```

説明

IProxyJNI_getInstance()は、次の処理を実行する

- ・パラメータ構造体に従って、JNI 用に変換する(pJNIParams ≠ NULL)
- ・java クラス内のクラスを生成する

戻り値

生成したクラスインスタンス

名前

IProxyJNI_classObjectMethodCall - オブジェクトに対するメソッドの呼び出し

書式

```
#include <proxyjni/IProxyJNI.h>

void IProxyJNI_classObjectMethodCall(IProxyJNI* pThis, JNIReturn* pJNIReturn, void* pObj, char*
pFuncName, JNIParams* pJNIParams)
```

説明

IProxyJNI_classObjectMethodCall ()は、次の処理を実行する

- ・ パラメータ構造体に従って、JNI 用に変換する(pJNIParams ≠ NULL)
- ・ オブジェクト内のメソッドを実行する
- ・ 実行結果を、JNI 用から戻り構造体へ変換する

戻り値

無し

名前

IProxyJNI_end - JNI の終了処理

書式

```
#include <proxyjni/IProxyJNI.h>
void IProxyJNI_end(IProxyJNI* pThis)
```

説明

IProxyJNI_end()は、JNI の終了処理を行う。

戻り値

無し

名前

CProxyJNIFunction_ctor - CProxyJNIFunction のコンストラクタ

書式

```
#include <proxyjni/ CProxyJNIFunction.h>
void CProxyJNIFunction_ctor(CProxyJNIFunction* pThis, CProxyJNIImpl* impl)
```

説明

CProxyJNIFunction_ctor()は、impl を内部に保持する。

戻り値

無し

名前

CProxyJNIFunction_dtor - CProxyJNIFunction のデストラクタ

書式

```
#include <proxyjni/ CProxyJNIFunction.h>
void CProxyJNIFunction_dtor(CProxyJNIFunction* pThis)
```

説明

CProxyJNIFunction_dtor()は、CProxyJNIFunction の解放を行う

戻り値

無し

名前

CProxyJNIFunction_init - JNI の初期化

書式

```
#include <proxyjni/ CProxyJNIFunction.h>
void CProxyJNIFunction_init(IProxyJNI* pThis)
```

説明

CProxyJNIFunction_init()は、CProxyJNIImpl_init()関数に委譲する。

戻り値

無し

名前

CProxyJNIFunction_classStaticMethodCall -クラスファイル内のクラススタティックメソッド実行

書式

```
#include <proxyjni/ CProxyJNIFunction.h>

void CProxyJNIFunction_classStaticMethodCall(IProxyJNI* pThis, JNIReturn* pJNIReturn, char*
pClassName, char* pFuncName, JNIParams* pJNIParams)
```

説明

CProxyJNIFunction_classStaticMethodCall()は、CProxyJNIImpl_classStaticMethodCall()関数に委譲する。

戻り値

無し

名前

CProxyJNIFunction_getInstance - クラスファイル内のクラスのインスタンス生成

書式

```
#include <proxyjni/ CProxyJNIFunction.h>

void* CProxyJNIFunction_getInstance(IProxyJNI* pThis, char* pClassName, JNIParams*
pJNIParams)
```

説明

CProxyJNIFunction_getInstance()は、CProxyJNIImpl_getInstance 関数に委譲する。

戻り値

CProxyJNIImpl_getInstance 関数の戻り値

名前

CProxyJNIFunction_classObjectMethodCall -オブジェクトに対するメソッドの呼び出し

書式

```
#include <proxyjni/ CProxyJNIFunction.h>

void CProxyJNIFunction_classObjectMethodCall(IProxyJNI* pThis, JNIReturn* pJNIReturn, void*
pObj, char* pFuncName, JNIParams* pJNIParams)
```

説明

CProxyJNIFunction_classObjectMethodCall()は、CProxyJNIImpl_classObjectMethodCall 関数に委譲する

戻り値

無し

名前

CProxyJNIFunction_end - JNI の終了処理

書式

```
#include <proxyjni/ CProxyJNIFunction.h>
void CProxyJNIFunction_end(IProxyJNI* pThis)
```

説明

CProxyJNIFunction_end()は、CProxyJNIImpl_end 関数に委譲する

戻り値

無し

名前

GetProxyJNIImpl – JNI 呼び出し実装のインターフェース取得

書式

```
#include <proxyjni/CProxyJNIImpl.h>
CProxyJNIImpl* GetProxyJNIImpl()
```

説明

GetProxyJNIImpl ()は、JNI 呼び出しの実装を取得します。
インターフェースとして次の関数が定義されている。
関数の呼び出し方法は、クラス名(CProxyJNIImpl)_関数名()で、
第一引数に、インターフェース構造体へのポインターを渡します。

Init()

JNI の初期処理を行う

classStaticMethodCall()

java ファイル内スタティックメソッドを実行する

getInstance()

java ファイル内のクラスインスタンスを生成する

classObjectMethodCall()

java ファイル内のクラスメソッドを実行する

end()

JNI の終了処理を行う

戻り値

JNI 呼び出し実装のインターフェース構造体へのポインター

名前

CProxyJNIImpl_ctor – CProxyJNIImpl のコンストラクタ

書式

```
#include <proxyjni/CProxyJNIImpl.h>
void CProxyJNIImpl_ctor(CProxyJNIImpl* pThis)
```

説明

CProxyJNIImpl_ctor()は、CProxyJNIImpl に各関数を設定する。

戻り値

無し

名前

CProxyJNIImpl_dtor – CProxyJNIImpl のデストラクタ

書式

```
#include <proxyjni/CProxyJNIImpl.h>
void CProxyJNIImpl_dtor(CProxyJNIImpl* pThis)
```

説明

CProxyJNIImpl_dtor()は、CProxyJNIImpl の解放する。

戻り値

無し

名前

CProxyJNIImpl_init – JNI の初期化

書式

```
#include <proxyjni/CProxyJNIImpl.h>
void CProxyJNIImpl_init(CProxyJNIImpl* pThis)
```

説明

IProxyJNI_init 関数を参照

戻り値

無し

名前

CProxyJNIImpl_classStaticMethodCall –クラスファイル内のクラススタティックメソッド実行

書式

```
#include <proxyjni/CProxyJNIImpl.h>

void CProxyJNIImpl_classStaticMethodCall(CProxyJNIImpl* pThis, JNIReturn* pJNIReturn, char*
pClassName, char* pFuncName, JNIParams* pJNIParams)
```

説明

IProxyJNI_classStaticMethodCall 関数を参照

戻り値

無し

名前

CProxyJNIImpl_getInstance – クラスのインスタンス生成

書式

```
#include <proxyjni/CProxyJNIImpl.h>

void* CProxyJNIImpl_getInstance(CProxyJNIImpl* pThis, char* pClassName, JNIParams*
pJNIParams)
```

説明

IProxyJNI_getInstance 関数を参照

戻り値

クラスのインスタンス

名前

CProxyJNIImpl_classObjectMethodCall –オブジェクトに対するメソッドの呼び出し

書式

```
#include <proxyjni/CProxyJNIImpl.h>

void CProxyJNIImpl_classObjectMethodCall(CProxyJNIImpl* pThis, JNIReturn* pJNIReturn, void*
pObj, char* pFuncName, JNIParams* pJNIParams)
```

説明

IProxyJNI_classObjectMethodCall 関数を参照

戻り値

無し

名前

CProxyJNIImpl_end – JNI の終了処理

書式

```
#include <proxyjni/CProxyJNIImpl.h>
void CProxyJNIImpl_end(CProxyJNIImpl* pThis)
```

説明

IProxyJNI_end 関数を参照

戻り値

無し

名前

GetJNIJavaSysProp – システムプロパティ操作クラス取得

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
CJNIJavaSysProp* GetJNIJavaSysProp(CProxyJNIImpl* impl)
```

説明

GetJNIJavaSysProp()は、システムプロパティ操作クラスを取得する。
クラスとして、次の関数が定義している。
関数の呼び出し方法は、クラス名(CJNIJavaSysProp)_関数名()で、
第一引数にクラス構造体へのポインターを渡します。

setClassPath()

クラスパスの設定する

setSysProp()

システムプロパティの設定する

getSysProp()

指定した key で、システムプロパティを取得する

戻り値

無し

名前

CJNIJavaSysProp_ctor – CJNIJavaSysProp コンストラクタ

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
void CJNIJavaSysProp_ctor(CJNIJavaSysProp* pThis, CProxyJNIImpl* impl)
```

説明

CJNIJavaSysProp_ctor()は、CProxyJNIFunction_ctor()を呼び出す。

戻り値

無し

名前

CJNIJavaSysProp_dtor – CJNIJavaSysProp デストラクタ

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
void CJNIJavaSysProp_dtor(CJNIJavaSysProp* pThis)
```

説明

CJNIJavaSysProp_dtor()は、CJNIJavaSysProp の解放する。

戻り値

無し

名前

CJNIJavaSysProp_setClassPath – クラスパス設定をする

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
void CJNIJavaSysProp_setClassPath(CJNIJavaSysProp* pThis, char* filePath)
```

説明

CJNIJavaSysProp_setClassPath()は、filePath で指定したディレクトリ内のファイルを取得し、CJNIJavaSysProp_setSysProp()を呼び出す。

Key: java.class.path

戻り値

無し

名前

CJNIJavaSysProp_setSysProp – システムプロパティの設定

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
void CJNIJavaSysProp_setSysProp(CJNIJavaSysProp* pThis, char* key, char* value)
```

説明

CJNIJavaSysProp_setSysProp ()は、システムプロパティを設定する。

戻り値

無し

名前

CJNIJavaSysProp_getSysProp – システムプロパティの取得

書式

```
#include <proxyjni/ CJNIJavaSysProp.h>
char* CJNIJavaSysProp_getSysProp(CJNIJavaSysProp* pThis, char* key)
```

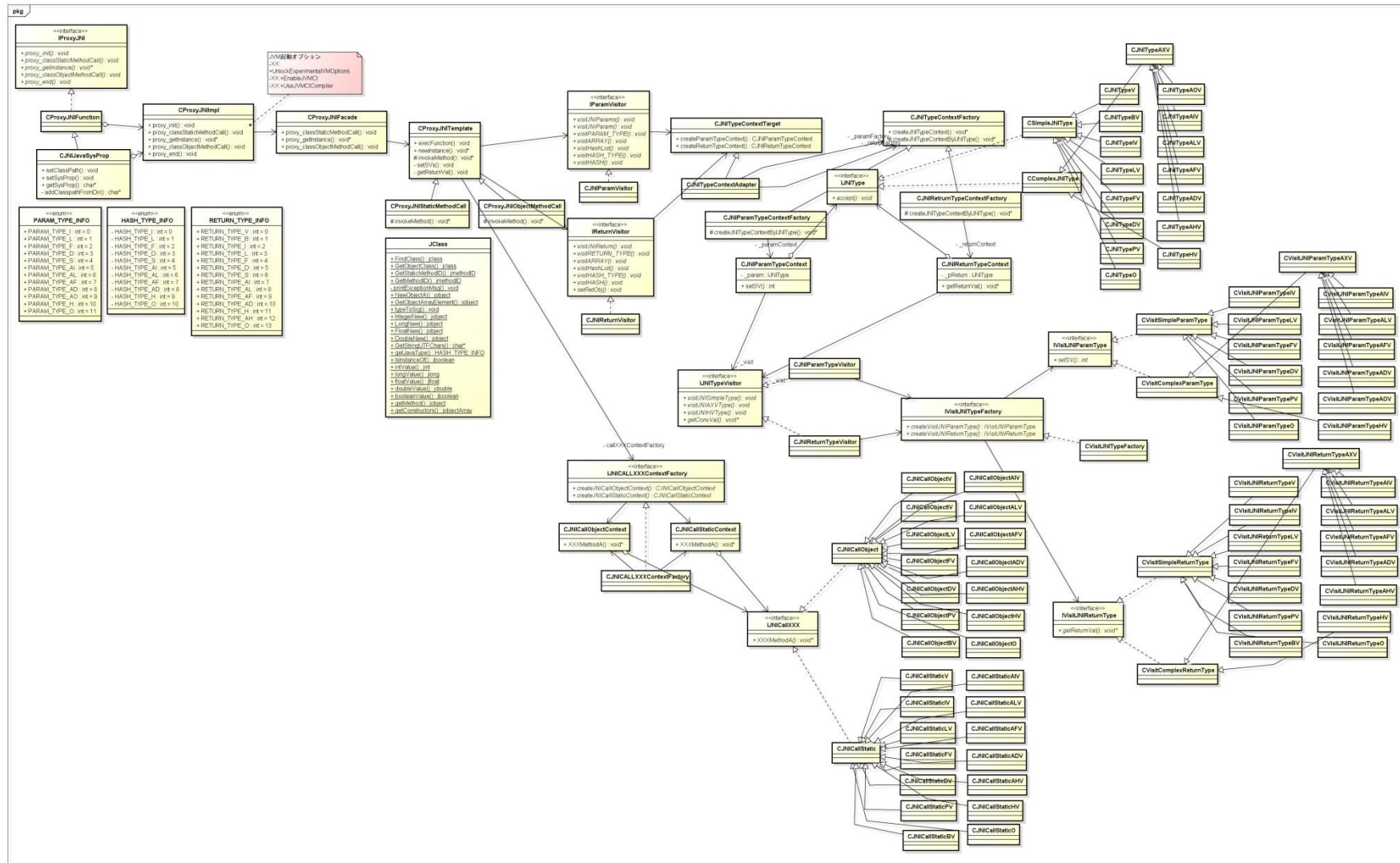
説明

CJNIJavaSysProp_getSysProp ()は、指定した **key** でシステムプロパティを取得する。

戻り値

取得した値

クラス図



パラメータ構造体

```
typedef struct _JNIPParams      JNIPParams;

struct _JNIPParams {
    int      argc;                // DynLangParam の個数
    JNIPParam* argv;             // 引数情報
};
```

```
typedef struct _JNIPParam      JNIPParam;

struct _JNIPParam {
    PARAM_TYPE_INFO param_type;
    void* param_value;
};
```

PARAM_TYPE_INFO	型名	param_value 値
PARAM_TYPE_I	Int 型	Int 型のポインター
PARAM_TYPE_L	Long 型	Long 型のポインター
PARAM_TYPE_F	Float 型	Float 型のポインター
PARAM_TYPE_D	Double 型	Double 型のポインター
PARAM_TYPE_S	文字列型	Char 型のポインター
PARAM_TYPE_AI	Int 型の配列	ARRAY_TYPE 型のポインター
PARAM_TYPE_AL	Long 型の配列	ARRAY_TYPE 型のポインター
PARAM_TYPE_AF	Float 型の配列	ARRAY_TYPE 型のポインター
PARAM_TYPE_AD	Double 型の配列	ARRAY_TYPE 型のポインター
PARAM_TYPE_AO	Object 型の配列	ARRAY_TYPE 型のポインター
PARAM_TYPE_H	ハッシュ型	HASHLIST_TYPE 型のポインター
PARAM_TYPE_O	Object 型	

```
typedef struct _Array_Type      ARRAY_TYPE;
struct _Array_Type {
    int len;
    void* value;
};
```

PARAM_TYPE_INFO HASH_TYPE_INFO RETURN_TYPE_INFO	Value 値
PARAM_TYPE_AI HASH_TYPE_AI RETURN_TYPE_AI	Int 型のポインター
PARAM_TYPE_AL HASH_TYPE_AL RETURN_TYPE_AL	Long 型のポインター
PARAM_TYPE_AF HASH_TYPE_AF RETURN_TYPE_AF	Float 型のポインター
PARAM_TYPE_AD HASH_TYPE_AD RETURN_TYPE_AD	Double 型のポインター
PARAM_TYPE_AO	JNIParam 型のポインター
RETURN_TYPE_AH	HASHLIST_TYPE 型のポインター

```
typedef struct _HashList_Type    HASHLIST_TYPE;
struct _HashList_Type {
    HASH_TYPE* hashInfo;
};
```

```
typedef struct _Hash_Type        HASH_TYPE;
struct _Hash_Type {
    HASH_TYPE* next;
    char* key;
    HASH_TYPE_INFO value_type;
    void* value;
};
```

HASH_TYPE_INFO	型名	Value 値
HASH_TYPE_I	Int 型	Int 型のポインター
HASH_TYPE_L	Long 型	Long 型のポインター
HASH_TYPE_F	Float 型	Float 型のポインター
HASH_TYPE_D	Double 型	Doule 型のポインター
HASH_TYPE_S	文字列型	Char 型のポインター
HASH_TYPE_AI	Int 型の配列	ARRAY_TYPE 型のポインター
HASH_TYPE_AL	Long 型の配列	ARRAY_TYPE 型のポインター
HASH_TYPE_AF	Float 型の配列	ARRAY_TYPE 型のポインター
HASH_TYPE_AD	Double 型の配列	ARRAY_TYPE 型のポインター
HASH_TYPE_H	ハッシュ型	HASHLIST_TYPE 型のポインター
HASH_TYPE_O	Object 型	

戻り構造体

```
typedef struct _JNIReturn      JNIReturn;
struct _JNIReturn {
    RETURN_TYPE_INFO return_type;
    void* return_value;
};
```

RETURN_TYPE_INFO	型名	return_value 値
RETURN_TYPE_V	無し	
RETURN_TYPE_B	Boolean 型	Char 型のポインター
RETURN_TYPE_I	Int 型	Int 型のポインター
RETURN_TYPE_L	Long 型	Long 型のポインター
RETURN_TYPE_F	Float 型	Float 型のポインター
RETURN_TYPE_D	Double 型	Doule 型のポインター
RETURN_TYPE_S	文字列型	Char 型のポインター
RETURN_TYPE_AI	Int 型の配列	ARRAY_TYPE 型のポインター
RETURN_TYPE_AL	Long 型の配列	ARRAY_TYPE 型のポインター
RETURN_TYPE_AF	Float 型の配列	ARRAY_TYPE 型のポインター
RETURN_TYPE_AD	Double 型の配列	ARRAY_TYPE 型のポインター
RETURN_TYPE_H	ハッシュ型	HASHLIST_TYPE 型のポインター
RETURN_TYPE_AH	ハッシュ型の配列	ARRAY_TYPE 型のポインター
RETURN_TYPE_O	Object 型	