

Utilisation du Wrapper Galunity

Importation du wrapper dans un projet Unity

L'importation du wrapper dans votre projet se fait en deux étapes :

- Importer le pack **Galunity.unityEngine** pour ajouter le wrapper et le plugin à votre projet.
- Extraire le contenu de l'archive **Dlls & Device Manager.zip** (contient toutes les dlls et le device manager de la Galalib) à la racine de votre projet.

Vous pouvez alors faire appel aux fonctions du wrapper pour interagir avec la Galalib.

Fonctions du wrapper

Toutes les fonctions du wrapper sont définies en statiques et doivent être précédé de « Galunity. » pour pouvoir être appelées (voir le projet Exemple). Certaines fonctions retournent ou prennent en paramètres des variables de types **NSPValue** qui sont en réalité des **IntPtr**. Pour pouvoir utiliser le mot clé **NSPValue** vous devez au préalable déclarer « using NSPValue = System.IntPtr » avant la déclaration de votre classe MonoBehaviour (voir le projet Exemple).

La fonction *void InitInterface()* doit être appelée en premier.

Les fonctions ci-dessous sont les principales fonctions du wrapper et permettent d'interagir avec la Galalib :

int CreateNamespace(string appName, string appVersion, string appCreator)

void SetApplicationName(string appName)

void SetApplicationVersion(string appVersion)

void SetApplicationCreator(string appCreator)

string GetApplicationName()

string GetApplicationVersion()

string GetApplicationCreator()

```

int CreateParameter(string address, int count)

int SetParameter(string address, int index, NSPIAttr attribute, NSPValue nspValue)

int GetParameter(string address, int index, NSPIAttr attribute, NSPValue nspValue)

int CreateObserver(string address, int index, NSPIAttr attribute, NspCallback valueC, NspCallback addressC)

int SendValue(string address, int index, NSPIAttr attribute, NSPValue nspValue)

int SaveToXml(string filepath)

int LoadFromXml(string filepath)

int MappingFromXml(string filepath)

int PresetsFromXml(string filepath)

```

Certaines de ces fonctions prennent en paramètres des variables de type NSPValue.

Utilisez les fonctions ci-dessous pour créer des variables de type NSPValue à partir des types simples de C# :

```

NSPValue BoolToNsp(bool val)

NSPValue BoolArrayToNsp(bool[] array)

NSPValue StringToNsp(string str)

NSPValue StringArrayToNsp(string[] array)

NSPValue FloatToNsp(float val)

NSPValue FloatArrayToNsp(float[] array)

NSPValue IntToNsp(int val)

NSPValue IntArrayToNsp(int[] array)

```

Utilisez les fonctions ci-dessous pour créer des variables de type simples C# à partir de variables de type NSPValue :

```

bool NspToBool(NSPValue val)

bool[] NspToBoolArray(NSPValue nspValue)

string NspToString(NSPValue nspValue)

string[] NspToStringArray(NSPValue nspValue)

float NspToFloat(NSPValue val)

float[] NspToFloatArray(NSPValue nspValue)

int NspToInt(NSPValue val)

```

```
int[] NspToIntArray(NSPValue nspValue)
```

Utilisation du wrapper

Voir le projet exemple pour plus de détails.