

Formation Python ILL

Modules externes en Python : appeler du C depuis Python

Matthieu Moy

Ensimag

octobre 2016

Appeler du C depuis Python

Exemple : `06-c-modules/code`

Plusieurs étapes :

- Écrire le code C (`demo.c`)
- Écrire le code Python (`main.py`)
- Compiler le code C (En utilisant `setup.py`)
- Exécuter l'ensemble

Exemple

```
# -*- coding: utf-8 -*-  
  
import demo  
  
print "Before calling f ..."  
demo.say_hello()  
print "After calling f."  
print demo.return_42()  
print demo.add(12, 13)
```

setuptools : “compiler” un projet Python

- Principe : décrire le projet dans `setup.py`, laisser l'outil faire le reste.
- Commandes (depuis un terminal) :
 - ▶ `python setup.py help`
 - ▶ `python setup.py build` (compiler dans le sous-répertoire `build`)
 - ▶ `python setup.py install --user` (installer pour l'utilisateur courant)
 - ▶ `python setup.py install` (installer pour tous les utilisateurs de la machine)
 - ▶ `python setup.py bdist` (créer une distribution binaire)
 - ▶ ...

Exécuter l'exemple

- Compiler :

```
python setup.py build
```

- Exécuter :

```
python build/lib.linux-x86_64-2.7/main.py
```

- Alternative : ajouter `build/lib.linux-x86_64-2.7` au PYTHONPATH (sous Spyder : menu « Tools » → « PYTHONPATH manager »).

Communication C ↔ Python

- Python et C n'ont pas la même façon de représenter les données
- ⇒ il faut faire des transformations pour passer de l'un à l'autre
- Python → C :

```
demo_add(PyObject *self, PyObject *args) {  
    ...  
    /* Lit deux entiers dans args,  
       écrit le résultat dans a et b */  
    PyArg_ParseTuple(args, "ii", &a, &b)
```

- C → Python :

```
/* Construit un entier Python */  
return Py_BuildValue("i", a + b);
```

Dire à Python où sont les fonctions : table des méthodes

```
static PyMethodDef demoMethods[] = {
    {"say_hello", demo_say_hello, METH_VARARGS, "P
    {"return_42", demo_return_42, METH_VARARGS, "R
    {"add", demo_add, METH_VARARGS, "Return the ad
    {NULL, NULL, 0, NULL} /* Last element: Sentinel
};

void initdemo(void) {
    Py_InitModule("demo", demoMethods);
}
```

Merci de votre attention !

N'oubliez pas :

- feuille de présence
- questionnaires satisfaction