

## 15.17 传递文件名给C扩展¶

### 问题¶

你需要向C库函数传递文件名，但是需要确保文件名根据系统期望的文件名编码方式编码过。

### 解决方案¶

写一个接受一个文件名为参数的扩展函数，如下这样：

```
static PyObject *py_get_filename(PyObject *self, PyObject *args) {
    PyObject *bytes;
    char *filename;
    Py_ssize_t len;
    if (!PyArg_ParseTuple(args, "O&", PyUnicode_FSConverter, &bytes)) {
        return NULL;
    }
    PyBytes_AsStringAndSize(bytes, &filename, &len);
    /* Use filename */
    ...

    /* Cleanup and return */
    Py_DECREF(bytes)
    Py_RETURN_NONE;
}
```

如果你已经有了一个 `PyObject *`，希望将其转换成一个文件名，可以像下面这样做：

```
PyObject *obj;    /* Object with the filename */
PyObject *bytes;
char *filename;
Py_ssize_t len;

bytes = PyUnicode_EncodeFSDefault(obj);
PyBytes_AsStringAndSize(bytes, &filename, &len);
/* Use filename */
...

/* Cleanup */
Py_DECREF(bytes);
```

If you need to return a filename back to Python, use the following code:

```
/* Turn a filename into a Python object */

char *filename;    /* Already set */
int filename_len;  /* Already set */

PyObject *obj = PyUnicode_DecodeFSDefaultAndSize(filename, filename_len);
```

### 讨论¶

以可移植方式来处理文件名是一个很棘手的问题，最后交由Python来处理。如果你在扩展代码中使用本节的技术，文件名的处理方式和和Python中是一致的。包括编码/界面字节，处理坏字符，代理转换和其他复杂情况。