\_\_declspec(dllexport)(转)

(2010-10-22 23:06:15)

[[http://simg.sinajs.cn/blog7style/images/common/sg_trans.gif](javascript:;)转载](javascript:;)

|  |  |
| --- | --- |
| 标签：  [杂谈](http://uni.sina.com.cn/c.php?t=blog&k=%D4%D3%CC%B8&ts=bpost&stype=tag) | 分类： [VC基础](http://blog.sina.com.cn/s/articlelist_1789558404_3_1.html) |

转自：http://hi.baidu.com/jk\_cau/blog/item/43f7e6b3b3082da1d9335abe.html  
  
  
一、\_\_declspec(dllexport):  
  
将一个函数声名为导出函数，就是说这个函数要被其他程序调用，即作为DLL的一个对外函数接口。  
  
通常它和extern "C" 合用，形式如下：  
  
extern "C"  
{  
\_\_declspec(dllexport) RETURN\_TYPE FUNCTION()  
{  
}  
}  
这是由于在制作DLL导出函数时由于C++存在函数重载，因此\_\_declspec(dllexport) FUNCTION(int,int) 在DLL会被decorate，例如被decorate成为 function\_int\_int，而且不同的编译器decorate的方法不同，造成了在用GetProcAddress取得FUNCTION地址时的不便，使用extern "C"时，上述的decorate不会发生，因为C没有函数重载，如此一来被extern"C"修饰的函数，就不具备重载能力。  
  
二、以下内容摘自MSDN：  
  
“在 32 位编译器版本中，可以使用\_\_declspec(dllexport) 关键字从 DLL 导出数据、函数、类或类成员函数。\_\_declspec(dllexport)将导出指令添加到对象文件(即obj文件)，若要导出函数，\_\_declspec(dllexport)关键字必须出现在调用约定关键字的左边（如果指定了关键字）。例如：  
  
\_\_declspec(dllexport) void \_\_cdecl Function1(void);  
  
若要导出类中的所有公共数据成员和成员函数，关键字必须出现在类名的左边，如下所示：  
  
class \_\_declspec(dllexport) CExampleExport : public CObject  
{ ... class definition ... };  
  
生成 DLL 时，通常创建一个包含正在导出的函数原型和/或类的头文件，并将\_\_declspec(dllexport)添加到头文件中的声明。  
若要提高代码的可读性，请为\_\_declspec(dllexport)定义一个宏并对正在导出的每个符号使用该宏：  
#define DllExport \_\_declspec( dllexport ) ”  
  
三、\_\_declspec(dllexport) 与 .def  
  
模块定义 （.def） 文件是包含一个或多个描述各种 DLL 属性的 Module 语句的文本文件。  
1、二者的目的都是将公共符号导入到应用程序中或从 DLL 导出函数。  
2、添加 \_\_declspec(dllexport)是为了提供不使用.def文件从 .EXE 或 .DLL 导出函数的简单方法。  
3、如果不使用 \_\_declspec (dllimport) 或 \_\_declspec(dllexport) 导出 DLL 函数，则 DLL 需要.def文件。  
4、并不是任何时候选择添加 \_\_declspec(dllexport)而放弃.def的方式都是好的。如果DLL是提供给VC++用户使用的，只需要把编译DLL时产生的.lib提供给用户，它可以很轻松地调用你的DLL。但是如果DLL是供VB、PB、Delphi用户使用的，那么会产生一个小麻烦。因为VC++对于\_\_declspec(dllexport) 声明的函数会进行名称转换，如下面的函数：  
\_\_declspec(dllexport) int \_\_stdcall IsWinNT()  
会转换为IsWinNT@0，这样你在VB中必须这样声明：  
Declare Function IsWinNT Lib "my.dll" Alias "IsWinNT@0" () As Long  
@的后面的数由于参数类型不同而可能不同。这显然不太方便。所以如果要想避免这种转换，就要使用.def文件方式。