

Dynamic-link library (DLL)

Dynamic-link library (also written without the hyphen), or **DLL**, is Microsoft's implementation of the shared library concept in the Microsoft Windows and OS/2 operating systems. These libraries usually have the file extension DLL, OCX (for libraries containing ActiveX controls), or DRV (for legacy system drivers). The file formats for DLLs are the same as for Windows EXE files — that is, Portable Executable (PE) for 32-bit and 64-bit Windows, and New Executable (NE) for 16-bit Windows. As with EXEs, DLLs can contain code, data, and resources, in any combination.

DLL (англ. **Dynamic-link library** — библиотека динамической компоновки) — понятие операционных систем Microsoft Windows и IBM OS/2; динамическая библиотека, позволяющая многократное применение различными программными приложениями. К DLL относятся также элементы управления ActiveX и драйверы. В мире UNIX аналогичные функции выполняют т. н. shared objects («разделяемые объекты»).

Формат файлов DLL придерживается тех же соглашений, что и формат исполняемых файлов, сочетая код, таблицы и ресурсы.

Creating DLL exports

```
#include <windows.h>

// DLL entry function (called on load, unload, ...)
BOOL APIENTRY DllMain(HANDLE hModule, DWORD dwReason, LPVOID lpReserved)
{
    return TRUE;
}

// Exported function - adds two numbers
extern "C" __declspec(dllexport) double AddNumbers(double a, double b)
{
    return a + b;
}
```

Using DLL imports

Make sure you include Example.lib file (assuming that Example.dll is generated) in the project (Add Existing Item option for Project!) before static linking. The file Example.lib is automatically generated by the compiler when compiling the DLL. Not executing the above statement would cause linking error as the linker would not know where to find the definition of AddNumbers. You also need to copy the DLL Example.dll to the location where the .exe file would be generated by the following code.

```
#include <windows.h>
#include <stdio.h>

// Import function that adds two numbers
extern "C" __declspec(dllimport) double AddNumbers(double a, double b);

int main(int argc, char *argv[])
{
    double result = AddNumbers(1, 2);
    printf("The result was: %f\n", result);
    return 0;
}
```

MDLLU.h

```
//-----  
#ifndef MDLLUH  
#define MDLLUH  
  
#ifdef __DLL__  
#define DLL_EI __declspec(dllexport)  
#else  
#define DLL_EI __declspec(dllimport)  
#endif  
  
extern "C" DLL_EI char * hi();  
extern "C" DLL_EI int test();  
  
typedef double matr[5][5];  
  
extern "C" DLL_EI void calcmatr(matr m);  
  
//-----  
#endif
```

MDLLU.cpp

```
//-----  
#include <vcl.h>  
#pragma hdrstop  
  
#include "MDLLU.h"  
  
//-----  
#pragma package(smart_init)  
  
char * hi()  
{return "Hi there!";}  
  
int test()  
{return 100;}  
  
void calcmatr(matr m)  
{  
    for (int i=0;i<5;i++)  
        for (int j=0;j<5;j++)  
            m[i][j]=i+j;  
}
```