

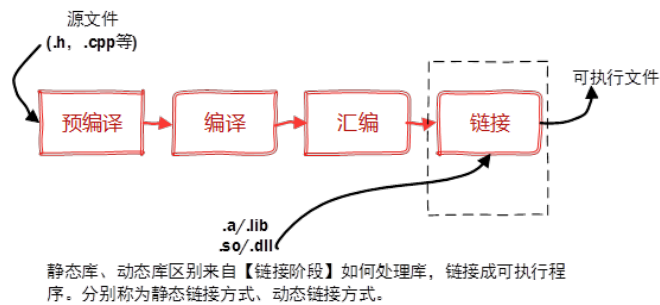
静态库&动态库的解读

王慧丽 2023. 11. 29

在编程的过程中，使用已经封装好的库函数是十分方便的，也是十分高效的，因此会使用函数库是很重要的。

1. 什么是库

库是写好的现有的，成熟的，可以复用的代码。现实中每个程序都要依赖很多基础的底层库，不可能每个人的代码都从零开始，因此库的存在意义非同寻常。本质上来说库是一种可执行代码的二进制形式。在C语言中，函数库文件分为两种类型，一种是静态库（库程序是直接注入目标程序的，不分彼此，库文件通常以.a（Linux系统）.lib（Windows系统）结尾），另一种是动态库（库程序是在运行目标程序时（中）加载的，库文件通常以.so（Linux系统）.dll（Windows系统）结尾）。所谓静态、动态是指链接。回顾一下，将一个程序编译成可执行程序步骤：

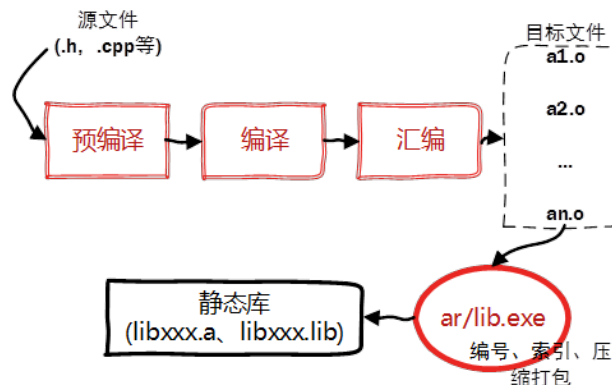


2. 静态库

之所以成为【静态库】，是因为在链接阶段，会将汇编生成的目标文件.o与引用到的库一起链接打包到可执行文件中。因此对应的链接方式称为静态链接。

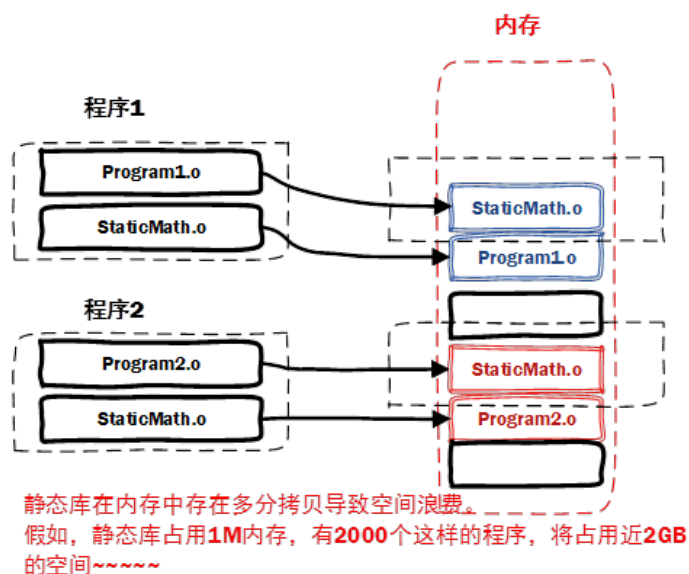
详细地，函数和数据被编译进一个二进制文件（通常扩展名为.LIB）。在使用静态库的情况下，在编译链接可执行文件时，链接器从库中复制这些函数和数据并把它们和应用程序的其他模块组合起来创建最终的可执行文件（.EXE文件）。当发布产品时，只需要发布这个可执行文件，并不需要发布被使用的静态库。

一般创建静态库的步骤如图所示：



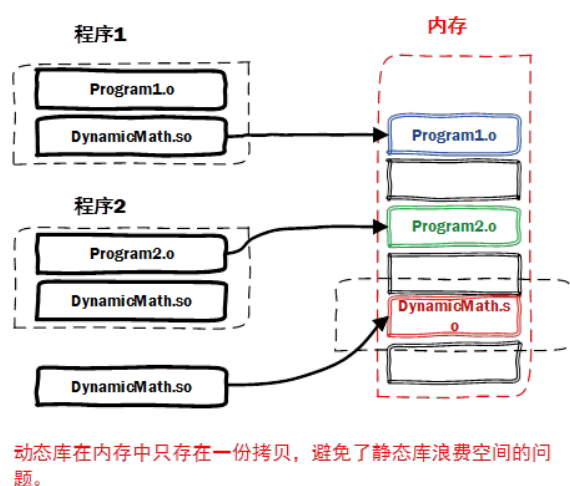
3. 动态库

为什么需要动态库，其实也是静态库的特点导致。空间浪费是静态库的一个问题。



另一个问题是静态库对程序的更新、部署和发布页会带来麻烦。如果静态库 liba.lib 更新了，所以使用它的应用程序都需要重新编译、发布给用户（对于玩家来说，可能是一个很小的改动，却导致整个程序重新下载，全量更新）。

动态库在程序编译时并不会被连接到目标代码中，而是在程序运行是才被载入。不同的应用程序如果调用相同的库，那么在内存里只需要有一份该共享库的实例，规避了空间浪费问题。动态库在程序运行是才被载入，也解决了静态库对程序的更新、部署和发布页会带来麻烦。用户只需要更新动态库即可，增量更新。



动态库特点总结：

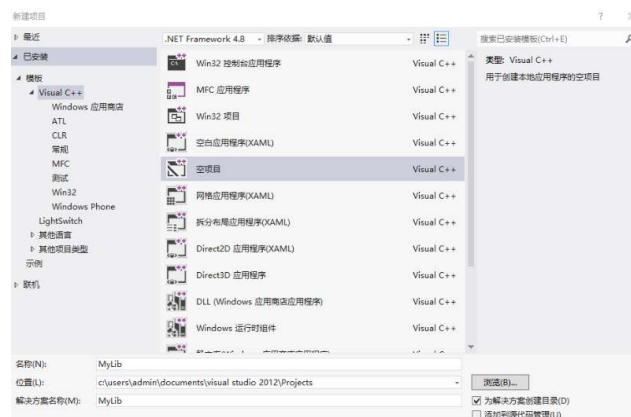
动态库把对一些库函数的链接载入推迟到程序运行的时期。可以实现进程之间的资源共享（因此动态库也称为共享库），将一些程序升级变得简单。甚至可以真正做到链接载入完全由程序员在程序代码中控制（显示调用）。与创建静态库不同的是，不需要打包工具（ar、lib.exe），直接使用编译器即可创建动态库。

4. 静态库和动态库的区别与联系

二者的不同点在于代码被载入的时刻不同。静态库的代码在编译过程中已经被载入可执行程序,因此体积比较大。动态库(共享库)的代码在可执行程序运行时才载入内存,在编译过程中仅简单的引用,因此代码体积比较小。不同的应用程序如果调用相同的库,那么在内存中只需要有一份该动态库(共享库)的实例。静态库和动态库的最大区别,静态情况下,把库直接加载到程序中,而动态库链接的时候,它只是保留接口,将动态库与程序代码独立,这样就可以提高代码的可复用度,和降低程序的耦合度。静态库在程序编译时会被连接到目标代码中,程序运行时将不再需要该静态库。动态库在程序编译时并不会被连接到目标代码中,而是在程序运行是才被载入,因此在程序运行时还需要动态库存在

5. 静态链接库的创建和使用

(1) 创建一个空项目 MyLib

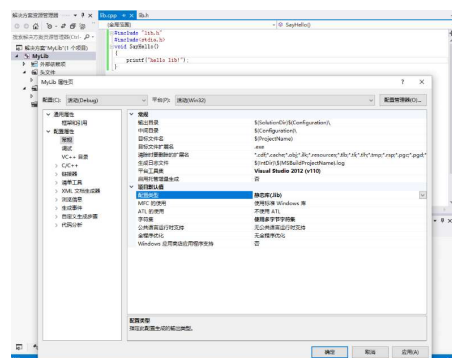


添加头文件 lib.h 和源文件 lib.cpp

```
lib.h
#pragma once
void SayHello();

lib.cpp
#include "lib.h"
#include<stdio.h>
void SayHello()
{
    printf("hello lib!");
}
```

(2) 配置项目属性 将“应用程序”修改为“静态库”。



(3) 编译生成解决方案后,可以看到工程目录下已生成.lib 文件。

电脑 > 文档 > Visual Studio 2012 > Projects > MyLib > Debug			
名称	修改日期	类型	大小
MyLib.lib	2023/11/29 14:28	Object File Library	5 KB

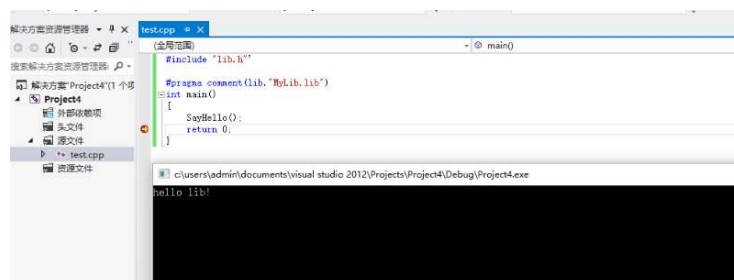
- (4) 测试，新创建一个项目，将生成的静态库文件 MyLib.lib 和头文件 lib.h 拷贝到工程目录下。

电脑 > 文档 > Visual Studio 2012 > Projects > Project4 > Project4			
名称	修改日期	类型	大小
Debug	2023/11/29 14:33	文件夹	
lib.h	2023/11/29 14:21	C/C++ Header	1 KB
MyLib.lib	2023/11/29 14:28	Object File Library	5 KB
Project4.vcxproj	2023/11/29 14:32	VC++ Project	4 KB
Project4.vcxproj.filters	2023/11/29 14:32	VC++ Project Fil...	1 KB
test.cpp	2023/11/29 14:33	C++ Source	1 KB

- (5) 加载静态库并进行调用：main.cpp

```
main.cpp
#include "lib.h"
#pragma comment(lib, "MyLib.lib")
int main()
{
    SayHello();
    return 0;
}
```

编译运行得到结果：



6. 动态链接库的创建和使用

- (1) 同样创建一个空项目，添加 dll.h dll.cpp 和 dllmain.cpp

注：dll 入口 DllMain 函数参数的详解见：

<https://blog.csdn.net/friendan/article/details/7659190>

```

dll.h
#pragma once
#define WIN32_LEAN_AND_MEAN // 从 Windows 头文件中排除极少使用的内容
// Windows 头文件
#include <windows.h>
|
_declspec(dllexport) void SayHello();

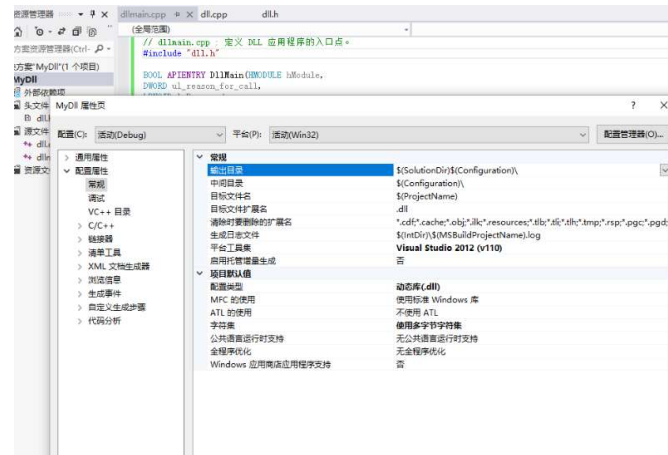
dll.cpp
#include "dll.h"
#include<stdio.h>
void SayHello()
{
    printf("hello dll!");
}

dllmain.cpp
// dllmain.cpp : 定义 DLL 应用程序的入口点。
#include "dll.h"

BOOL APIENTRY DllMain(HMODULE hModule,
    DWORD ul_reason_for_call,
    LPVOID lpReserved
)
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

```

- (2) 将“应用程序”修改为“动态库”。



- (3) 编译生成解决方案后，可以看到工程目录下已生成.dll 文件和.lib 文件。

共享 查看

此电脑 > 文档 > Visual Studio 2012 > Projects > MyDll > Debug				
名称	修改日期	类型	大小	
MyDll.dll	2023/11/29 14:59	应用程序扩展	30 KB	
MyDll.exp	2023/11/29 14:59	Exports Library ...	1 KB	
MyDll.lib	2023/11/29 14:59	Incremental Link...	183 KB	
MyDll.lib	2023/11/29 14:59	Object File Library	2 KB	
MyDll.pdb	2023/11/29 14:59	Program Debug...	347 KB	

注意：静态链接库有两种，这里的静态库是配合动态库使用的那种。

- (4) 创建一个测试工程，并将生成的 MyDll.dll、MyDll.lib 与 dll.h 拷贝到工程目

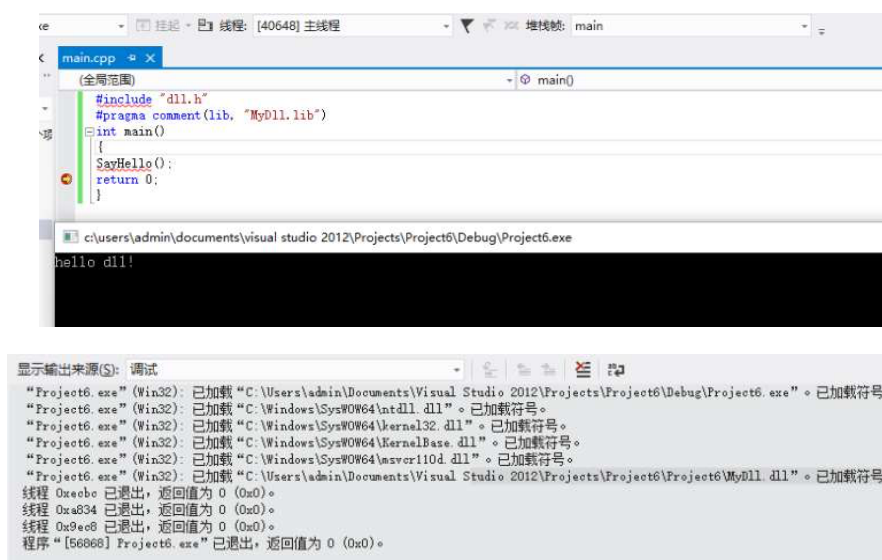
录下。写一个 main.cpp 进行测试

```
main.cpp
#include <iostream>
#include "dll.h"
#pragma comment(lib, "MyDll.lib")
int main()
{
    SayHello();
    return 0;
}

dll.h文件改为
#pragma once
#define WIN32_LEAN_AND_MEAN
// Windows 头文件
#include <windows.h>

_declspec(dllexport) void SayHello();//这里是dllimport
```

(5) 编译运行可以看到输出及程序运行过程中加载了 MyDll.dll



以上展示的是从空白工程创建静态库与动态库的方法，实际上，VS 提供了快速创建的模板，关于该方法的使用，可上网搜索到具体过程。

