DLL(Dynamic Link Library)　DSO(Dynamic Shared Objects)

- 动态链接库(.dll、.ocx等)
- 静态链接库(.lib)

- 动态共享对象(.so)
- 静态库(.a)

- 文件格式：
- **Windows： PE(Portable Executable)**
- **Linux： ELF(Executable Linkable Format)**

可执行文件、目标文件、**.dll、.lib、.so、.a**

DLL(Dynamic Link Library)

- 首先看看我们的可执行程序依赖了什么?
- 实用工具：dumpbin.exe
- 在自己的机器上查找在什么位置
- Dumpbin /imports file.exe
- Dumpbin –exports kernel32.dll

Windows与DLL

Windows系统中大量采用了DLL机制，甚至windows内核结构在很大程度上都依赖DLL机制。

Windows平台上大型软件的升级是通过DLL实现的。比如Office、IE、VS系列等。

Service Packs软件升级包

DLL优点

- 节省内存和减少交换操作。很多进程可以同时使用一个 DLL，在内存中共享该 DLL 的一个副本。相反，对于每个用静态链接库生成的应用程序，Windows 必须在内存中加载库代码的一个副本。
- 易于升级，提供售后支持。当 DLL 中的函数发生更改时，只要函数的参数和返回值没有更改，就不需重新编译或重新链接使用它们的应用程序。相反，静态链接的对象代码要求在函数更改时重新链接应用程序。
- 支持多语言程序。只要程序遵循函数的调用约定，用不同编程语言编写的程序就可以调用相同的 DLL 函数。

比如

使用VB或C#编写应用程序的界面，业务逻辑使用C++或C

隐藏代码的实现

DLL缺点

- 应用程序不是独立的
- 意味着发布自己的应用程序时，必须同时发布DLL

DLL的创建和使用

一、命令行下的创建和使用

二、VS开发环境下的创建和使用

一、创建一个DLL

新建Math.c 程序；

注意使用指定符号：__declspec(dllexport)

cl /LDd Math.c 创建Debug版DLL；

cl /LD Math.c 创建release版DLL；

dumpbin /EXPORTS Math.dll 查看DLL的导出符号

二、使用DLL创建应用程序

新建TestMath.c程序；

注意使用指定符号： __declspec(dllimport)

cl /c TestMath.c

使用编译器只编译TestMath.c

link TestMath.obj Math.lib

使用链接器将TestMath.obj 和Math.lib链接一起生成可执行文件TestMath.exe

运行TestMath.exe时，需要Math.dll

命令行下示例

- 创建自己的DLL
- 使用DLL创建自己的应用程序

RAV,Relative Address Virtual

相对虚拟地址，指明了在什么地方找到符号。

演示lib文件的作用（链接过程需要）

演示dll文件的作用（运行发布时需要）

VS/VC开发环境下示例

1.创建DLL工程

2.创建测试DLL工程

演示隐式链接：两种方法

3.创建测试工程

演示显示链接

隐式链接和显示链接

隐式链接：加载时动态链接（常用的方法）

在加载时动态链接中，应用程序像调用本地函数一样对导出的 DLL 函数进行调用。要使用加载时动态链接，需要在编译和链接应用程序时提供头文件 (.h) 和导入库文件 (.lib)。

这样做时，链接器将向系统提供加载 DLL 所需的信息，并在加载时解析导出的 DLL 函数的位置。

隐式链接和显示链接

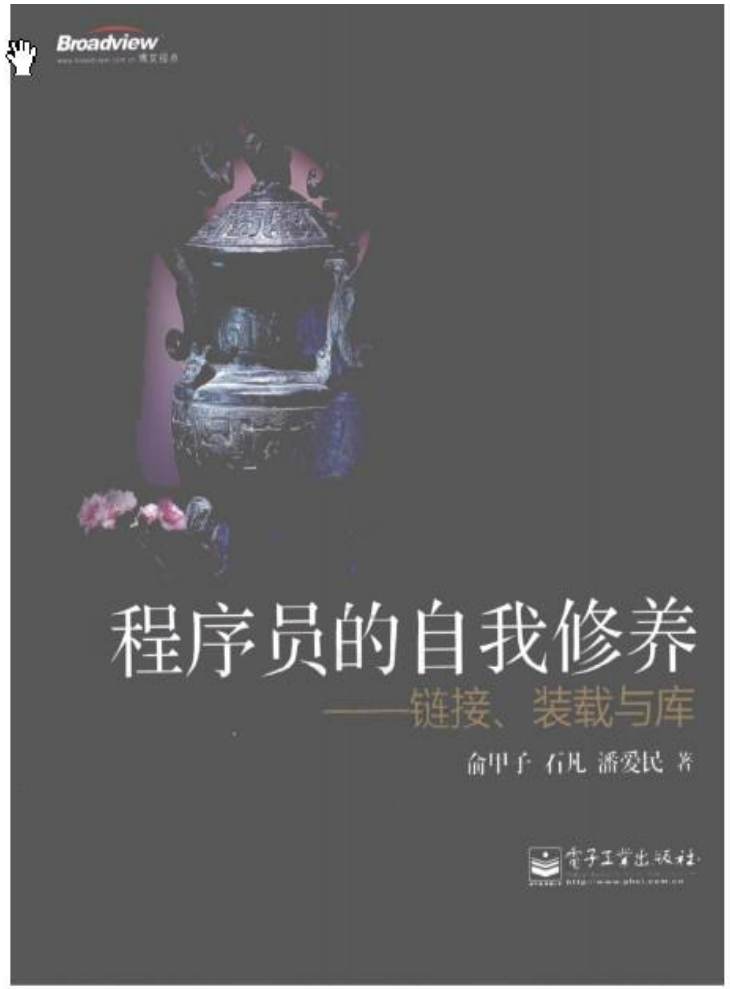
显示链接：运行时动态链接

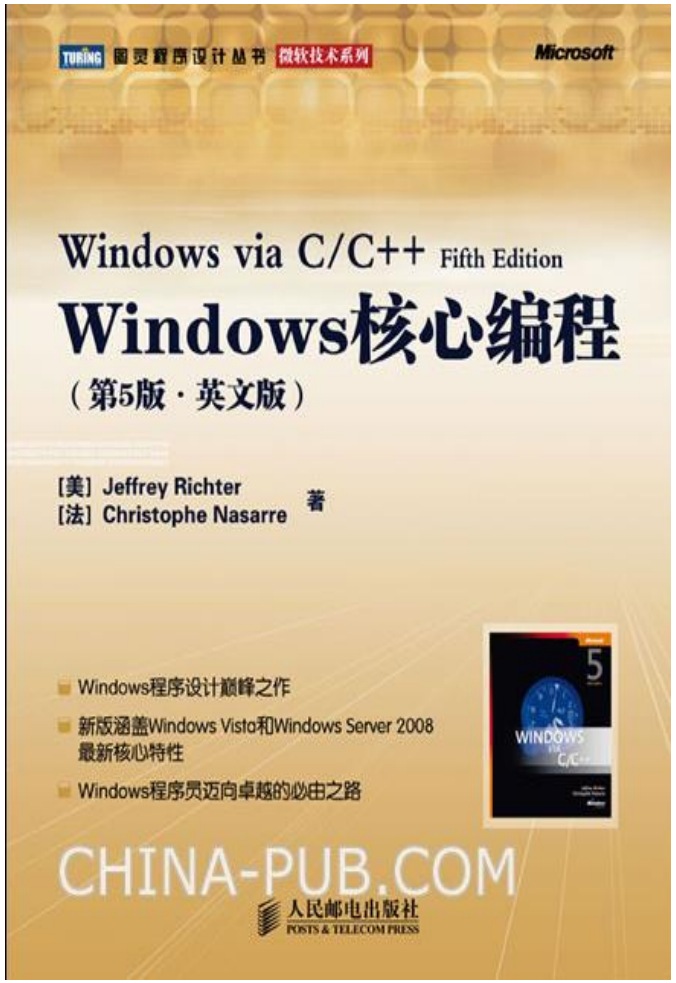
在运行时动态链接中，应用程序调用 LoadLibrary 函数或 LoadLibraryEx 函数以在运行时加载 DLL。成功加载 DLL 后，可以使用 GetProcAddress 函数获得要调用的导出的 DLL 函数的地址。在使用运行时动态链接时，无需使用导入库文件。

Windows 用来定位 DLL 的搜索路径

- 当前进程的可执行模块所在的目录。
- 当前目录。
- Windows 系统目录。GetSystemDirectory 函数检索此目录的路径。
- Windows 目录。GetWindowsDirectory 函数检索此目录的路径。
- PATH 环境变量中列出的目录

参考书目


Broadview
程序员的自我修养
——链接、装载与库
俞甲子 石凡 潘爱民 著
电子工业出版社


TURING 图灵程序设计丛书 微软技术系列
Microsoft
Windows via C/C++ Fifth Edition
Windows核心编程
（第5版·英文版）
[美] Jeffrey Richter
[法] Christophe Nasarre 著
Windows程序设计巅峰之作
新版涵盖Windows Vista和Windows Server 2008最新核心特性
Windows程序员迈向卓越的必由之路
CHINA-PUB.COM
人民邮电出版社
POSTS & TELECOM PRESS