windows运行一个.exe的过程

来自于《windows核心编程》

构建dll：

链接时，将<crtdll.c>（我本地在C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\crt\src）的 _DllMainCRTStartup 嵌入到dll映像中。

每当系统第一次将一个dll映射到进程地址空间或程序创建一个线程时，调用。

构建exe：

所有.obj 链接成.exe文件，其中包括所有二进制代码，全局、静态变量，各种段（如导入段，未初始化段）。

将运行库的启动函数嵌入exe。

1，运行.exe（双击，右键运行，命令行等）

操作系统为进程创建虚拟地址空间，并将.exe映射到基地址上，此时进程，主线程创建。

系统加载程序（估计是开机就启动的一个.exe）会检查.exe文件头（PE）。

文件头包含着一些exe级别的信息，构建exe时由链接器嵌入，比如“这是个控制台程序还是窗口程序”。

解析导入段，导入段包括所有需要的dll名称（不包括动态加载），dll函数地址，dll依赖的dll等。

2，操作系统调用C/C++运行库实现的运行时启动函数，比如wWinMainCRTStartup()。C/C++运行库安装VS都会有。

启动函数所做的事情：

> 1）获取进程命令行指针
>
> 2）进程环境变量指针
>
> 3）初始化运行库的全局变量，包含stdlib.h就可以访问。比如unsigned int _osver; 操作系统的版本号，unsigned int __argv命令行参数数组等等。
>
> 4）初始化内存分配。malloc，堆等。
>
> 5）调用所有全局和静态C++类对象构造函数。

初始化之后，执行我们开发人员写的 main或者WinMain函数，然后走我们的代码。

走完之后，启动函数调用exit()函数，退出进程。

exit做的事情：

1）调用_onexit()执行注册的函数

2）C++析构

3）如果需要生成内存泄漏报告，就去生成

4）调操作系统ExitProcess()函数

给一个由VS2010向导创建的MFC单文档程序的调试过程：

1，按下F5，相当于双击了.exe。

系统创建进程，主线程，在<crtdll.c>中的_DllMainCRTStartup打断点，因为如果exe的导入段有程序需要的dll，dll又写了DllMain函数，就会以DLL_PROCESS_ATTACH调用。通过hDllHandle参数的地址可以知道是哪个dll。

具体它做了什么事情，书上说是初始化了全局和静态变量及其它初始化工作。

2，所有dll加载完之后，调用运行库的启动函数。

在<crtexe.c>（我本地在C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\crt\src）的 mainret = WinMain(//...)打断点，然后就走到AfxWinMain了。

3，走自己的代码流程，消息循环等。如果你要关闭程序，先在<crtexe.c>的exit函数打断点，它就走这了。