随笔- 170  文章- 0  评论- 85

[**windows资源管理（内核对象/GDI对象/user对象）**](http://www.cnblogs.com/kekec/archive/2011/07/04/2097702.html)

　　在使用c++进行windows编程时，程序员除了管理使用new/malloc动态在堆上分配出来的内存外，还需要对windows的**内核对象、GDI对象和user对象**进行管理，这些对象使用句柄来标识，通过操作这些句柄就使用不同的资源对象。和堆内存一样，程序员也需要管理这些对象资源，以免造成系统资源泄漏。

**句柄（HANDLE）**是WONDOWS用来标识被应用程序所建立或使用的对象的唯一整数，WINDOWS使用各种各样的句柄标识诸如应用程序实例，窗口，控制，位图，GDI对象等等。句柄实际上是一种指向某种资源的指针，但与指针又有所不同：指针对应着一个数据在内存中的地址，得到了指针就可以自由地修改该数据。Windows并不希望一般程序修改其内部数据结构，因为这样太不安全。

      【内核对象、GDI对象和user对象之间的区别】

　　内核对象是不属于进程的，是属于windows内核的。进程只有一个内核对象句柄表，用来存放所有的内核对象句柄。所以，多个进程可以同时使用一个内核对象 -- 只要有handle即可。

　　对于GDI对象和user对象，他们是一个进程内部拥有的东西，不会被多个进程共有。GDI对象与绘图相关，而user与交互相关。

------

　　内核对象的直接拥有者是操作系统内核，所有进程共享这些内核对象，因此要有一种机制保证内 核对象的正确构建、销毁，Windows采用引用计数的技术；内核对象维护着一个引用计数成员。一个进程创建了一个内核对象，对象的引用计数为1，如果该 对象又被另外的进程共享，每多一个进程，引用计数会加1，当一个进程调用CloseHandle函数后，引用计数会减1，如果引用计数变为0，操作系统会 销毁该内核对象。引用计数实现上跟COM计数的引用计数类似。内核对象使用完之后，要调用CloseHandle。这个函数的作用就是将该内核对象的引用计数 - 1，当这个计数为0的时候，该内核对象会变成signal状态，并被操作系统销毁。

　　GDI对象和user对象的销毁不需用调用CloseHandle。每一个GDI对象和user对象的销毁都有其对应的Destroy或Delete方法。

------

## 内核对象 ##

|  |  |
| --- | --- |
| **核心对象** | **产生方法** |
| 事件对象 | HANDLE CreateEvent(); |
| 文件对象 | HANDLE CreateFile(); |
| 文件映射对象 | HANDLE CreateFileMapping(); |
| I/O完成对象 | HANDLE CreateIoCompletionPort(); |
| 作业对象 | HANDLE CreateJobObject(); |
| 信箱对象 | HANDLE CreateMailslot(); |
| 互斥对象 | HANDLE CreateMutex(); |
| 管道对象 | BOOL CreatePipe(); |
| 进程对象 | BOOL CreateProcess(); |
| 信号量对象 | HANDLE CreateSemaphore(); |
| 线程对象 | HANDLE CreateThread(); |
| 等待计时器对象 | HANDLE CreateWaitableTimer(); |

## GDI对象 ##

|  |  |  |
| --- | --- | --- |
| **GDI对象** | **产生方法** | **销毁方法** |
| 位图（HBITMAP） | CreateBitmap,CreateBitmapIndirect,  CreateCompatibleBitmap,CreateDIBitmap,  CreateDIBSection,CreateDiscardableBitmap | DeleteObject |
| 画刷（HBRUSH） | CreateBrushIndirect,CreateDIBPatternBrush,  CreateDIBPatternBrushPt,CreateHatchBrush,  CreatePatternBrush,CreateSolidBrush | DeleteObject |
| 设备上下文（HDC） | CreateDC | DeleteDC,ReleaseDC |
| 字体（HFONT） | CreateFont,CreateFontIndirect | DeleteObject |
| 内存DC（HDC） | CreateCompatibleDC | DeleteDC |
| 调色板（HPALETTE） | CreatePalette | DeleteObject |
| 画笔（HPEN） | CreatePen,CreatePenIndirect | DeleteObject |
| 区域（HRGN） | CombineRgn,CreateEllipticRgn,  CreateEllipticRgnIndirect,CreatePolygonRgn,  CreatePolyPolygonRgn,CreateRectRgn,  CreateRectRgnIndirect,CreateRoundRectRgn, | DeleteObject |

## USER对象 ##

|  |  |  |
| --- | --- | --- |
| **user对象** | **产生方法** | **销毁方法** |
| 快捷键（HACCEL） | CreateAcceleratorTable | DestroyAcceleratorTable |
| Caret | CreateCaret | DestoryCaret |
| 光标（HCURSOR） | CreateCursor,LoadCursor,LoadImage | DestoryCursor |
| 钩子（HHOOK） | SetWindowsHookEx | UnhookWindowsHookEx |
| 图标（HICON） | CreateIconIndirect,LoadIcon,LoadImage | DestroyIcon |
| 菜单（HMENU） | CreateMenu,CreatePopupMenu  LoadMenu,LoadMenuIndirect | DeleteMenu |
| 窗体（HWND） | CreateWindow,CreateWindowEx,  CreateDialogParam,CreateMDIWindow | DestroyWindow |
| HDWP | BeginDeferWindowPos | EndDeferWindowPos |

-- 关于程序对资源的使用情况，可以从任务管理器上看到。

