**内核对象：**

内核对象的一般特性，而不是那个内核对象的特性！！！！

首先，内核对象是一个比较抽象的概念！！

1. 什么是内核对象？

* 令牌对象token
* 事件对象Event
* 文件对象File
* 文件映射对象Mapping\_File
* 线程对象Thread
* 时钟对象Timer
* 线程池对象ThreadPool
* I/o完成端口对象，Completion port
* 工作对象Job
* 邮槽对象mailslot
* 互斥对象Mutex
* 管道对象pipe
* 进程对象process
* 信号灯对象semaphore

1. 每个内核对象，其实就是一块内存块，这个内存块位于操作系统的内核地址空间。因此，应用程序不能直接操作内核对象，需要用系统给定的函数来操作。不同内核对象的创建使用不同的函数。内核对象的结构：公用部分（安全描述符，计数）和个性部分。
2. 内核对象用一个句柄来标识。
3. 内核对象的安全描述符和计数的作用。

HANDLE CreateFileMapping(

HANDLE hFile,

PSECURITY\_ATTRIBUTES psa,

DWORD flProtect,

DWORD dwMaximumSizeHigh,

DWORD dwMaximumSizeLow,

PCTSTR pszName);

HBITMAP CreateBitmap(

\_\_in int nWidth,

\_\_in int nHeight,

\_\_in UINT cPlanes,

\_\_in UINT cBitsPerPel,

\_\_in const VOID \*lpvBits);

typedef struct \_SECURITY\_ATTRIBUTES {

DWORD nLength;

LPVOID lpSecurityDescriptor;//安全描述符

BOOL bInheritHandle;

} SECURITY\_ATTRIBUTES;

1. 如何区分内核对象和用户对象，也就是应用层对象和内核对象。看创建对象的函数，如果创建对象的函数有安全描述符参数，那么这个函数创建的对象就是内核对象。例如画笔，窗口，画刷，位图。CreateBitmap
2. 进程内核对象列表。
3. 如何创建内核对象：几个例子：

HANDLE CreateThread(

PSECURITY\_ATTRIBUTES psa,

size\_t dwStackSize,

LPTHREAD\_START\_ROUTINE pfnStartAddress,

PVOID pvParam,

DWORD dwCreationFlags,

PDWORD pdwThreadId);

HANDLE CreateFile(

PCTSTR pszFileName,

DWORD dwDesiredAccess,

DWORD dwShareMode,

PSECURITY\_ATTRIBUTES psa,

DWORD dwCreationDisposition,

DWORD dwFlagsAndAttributes,

HANDLE hTemplateFile);

HANDLE CreateFileMapping(

HANDLE hFile,

PSECURITY\_ATTRIBUTES psa,

DWORD flProtect,

DWORD dwMaximumSizeHigh,

DWORD dwMaximumSizeLow,

PCTSTR pszName);

HANDLE CreateSemaphore(

PSECURITY\_ATTRIBUTES psa,

LONG lInitialCount,

LONG lMaximumCount,

PCTSTR pszName);

1. 关闭内核对象CloseHandle(HANDLE)；如果忘了关有什么后果。用explorer process查看系统中的进程，以及进程中的内核对象。
2. 继承选项的含义。

SECURITY\_ATTRIBUTES sa;

sa.nLength = sizeof(sa);

sa.lpSecurityDescriptor = NULL;

sa.bInheritHandle = TRUE; // Make the returned handle inheritable.

1. 修改内核对象的信息：

BOOL SetHandleInformation(

HANDLE hObject,

DWORD dwMask,

DWORD dwFlags);

|  |  |
| --- | --- |
| HANDLE\_FLAG\_INHERIT  0x00000001 | If this flag is set, a child process created with the *bInheritHandles* parameter of [**CreateProcess**](http://127.0.0.1:47873/help/1-2512/ms.help?method=page&id=3EF0A5B2-4D71-4C17-8188-76A4025287FC&product=VS&productVersion=100&topicVersion=85&locale=ZH-CN&topicLocale=EN-US) set to TRUE will inherit the object handle. |
| HANDLE\_FLAG\_PROTECT\_FROM\_CLOSE  0x00000002 | If this flag is set, calling the [**CloseHandle**](http://127.0.0.1:47873/help/1-2512/ms.help?method=page&id=9B84891D-62CA-4DDC-97B7-C4C79482ABD9&product=VS&productVersion=100&topicVersion=85&locale=ZH-CN&topicLocale=EN-US) function will not close the object handle. |

SetHandleInformation(handle, HANDLE\_FLAG\_INHERIT,true);

GetHandleInformatioin(handle, &dw);

If(dw&& HANDLE\_FLAG\_INHERIT)

{

该对象可以被继承

}

1. 命名内核对象，这个主要是为了进程间共享内核对象，但这样很难保证，命名不重复！！！创建内核对象的函数中，如果有pszName参数，说明这个内核对象可以被命名，也就是可以创建命名内核对象！命名内核对象，可以在不同进程中使用！！！
2. 终端服务命名空间！

私有命名空间：边界描述符：Boundary Descriptor