第一图书网, tushu007.com

<<Windows用户态程序高效排错>>

图书基本信息

书名：<<Windows用户态程序高效排错>>

13位ISBN编号：9787121051937

10位ISBN编号：7121051931

出版时间：2007-12

出版时间：电子工业出版社

作者：熊力

页数：236

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

内容概要

本书是一本介绍Windows系统上的用户态程序排错方法和技巧的书。
本书分为4个章节，先介绍最重要的、通用的思考方法，以便制定排错步骤；再介绍对排错有帮助的知识点和工具；并介绍了.NET Framework（CLR）的相关知识和调试技巧；最后一章针对常见的几大类问题进行了总结。
本书案例丰富，对现实中的实际问题进行了研究，并和读者一起分析解决办法；本书的写作思路为先给出问题描述，然后提供线索，再进行分析，让读者在阅读中也进行思考，以提高实际解决问题的能力。
本书适合希望学习排错、调试知识的软件开发、测试人员，希望深入学习Windows系统上用户态程序的排错知识的软件开发、测试人员。

作者简介

熊力，2004年开始在上海微软技术支持中心担任技术支持工程师。
他所在的小组负责帮助企业客户解决开发领域的技术难题。
作者专注于.NET Framework、 C/C++、COM和Web开发，现任微软中国研发集团服务器与开发工具事业部测试工程师。

书籍目录

第1章　比工具、技巧和经验都重要的是你的思考——从四个风格迥异的案例说起1.1　绝望的性能问题：ADO.NET 2.0竟然比1.0要慢1.1.1　问题描述1.1.2　悲观和绝望1.1.3　换位思考1.1.4　排错1.1.5　结论和收获1.1.6　题外话和相关讨论Safehandle的更多讨论平衡、取舍、双赢和RFC 1925Profiler的下载地址和相关资源1.2　不可思议：一个API同时打开了两个文件1.2.1　问题描述1.2.2　第一印象1.2.3　深入分析1.2.4　革命尚未成功1.2.5　结论1.2.6　题外话和相关讨论MSDN是最值得信赖的吗你敢说CPU坏了DWORD和文件长度程序输出0xcdcdcdcd,想到了什么1.3　简单的问题最棘手：稀疏平常的ASP.NET Session Lost问题1.3.1　问题描述1.3.2　制定策略1.3.3　具体操作和结论1.3.4　题外话和相关讨论排查session lost的经验1.4　本可以做得更好：SharePoint中文界面变英文1.4.1　问题描述1.4.2　排错步骤1.4.3　错过的线索第2章　汇编、异常、内存、同步和调试器——重要的知识点和神兵利器2.1　排错的工具：调试器Windbg2.1.1　调试器的功能：检查代码和资料，保存dump文件，断点控制程序的执行2.1.2　符号文件（Symbol file），把二进制和源代码对应起来2.1.3　一个简单的上手程序2.1.4　用Internet Explorer来操练调试器的基本命令vertarget检查进程概况！
peb 显示Process Environment Blocklmvm 检查模块的加载信息.reload / ！
sym 加载符号文件lmf 列出当前进程中加载的所有模块r,d,e 寄存器，内存的检查和修改！
address显示内存页信息S 搜索内存！
runaway 检查线程的CPU消耗~ 切换目标线程k,kb,kp,kv,kn 检查call stacku 反汇编x 查找符号的二进制地址dds 对应二进制地址的符号2.1.5　检查程序资料的小例子.frame 在栈中切换以便检查局部变量dt 格式化显示资料2.1.6　用Windbg控制程序进行实时调试（Live Debug）Wt Watch and Trace, 跟踪执行的强大命令断点和条件断点（condition breakpoint），高效地控制观测目标伪寄存器,帮助保存调试的中间信息Step Out的实现2.1.7　远程调试（Remote debug）2.1.8　如何通过Windbg命令行让中文魔兽争霸运行在英文系统上2.1.9　Dump文件2.1.10　CDB、NTSD和重定向到Kernel Debugging2.1.11　Debugger Extension，扩展Windbg的功能2.2　读懂机器的语言：汇编，CPU执行指令的最小单元2.2.1　需要用汇编来排错的常见情况案例分析：用汇编读懂VC编译器的优化问题描述我的分析案例分析：VC2003 编译器的bug、debug模式正常，release模式会崩溃例子程序跟踪汇编指令来分析案例分析：臭名昭著的DLL Hell如何导致ASP.NET出现Server Unavailable2.2.2　题外话和相关讨论Release比 Debug快吗2.3　理解操作系统对程序的反馈：异常（Exception）和通知（Debug Event）2.3.1　异常（Exception）的方方面面和一篇字字珠玑的文章案例分析：如何让C++像C#一样打印出函数调用栈（callstack）2.3.2　Adplus，抓取dump的方便工具案例分析：华生医生（Dr. Watson）在什么情况下不能记录Dump文件问题描述背景知识问题分析新的做法问题解决了，可是为什么华生医生（Dr. Watson）抓不到dump呢2.3.3　通知（Debug Event）是操作系统跟调试器交流的一种方法案例分析：VB6的版本问题2.3.4　题外话和相关讨论错过第一现场后还从dump中分析出线索吗Adplus，天天都用的工具未处理异常发生后的主动退出如何调试UnhandledExceptionFilter2.4　平坦内存空间中的层次结构：Heap和Stack2.4.1　Heap是对平坦空间的高效管理和利用2.4.2　PageHeap，调试Heap问题的工具简单例子的多种情况Heap上的内存泄漏和内存碎片2.4.3　Stack overrun/corruption2.4.4　题外话和相关讨论PageHeap的/unaligned参数Heap trace，系统帮你记录下每次Heap的操作为何才分配了300MB内存，就报告Out of memory2.5　找准排查问题的对应层次2.5.1　从C运行库看层次2.5.2　简单的_CRTDBG_MAP_ALLOC定义就可以让内存泄漏无可遁形2.5.3　BSTR Cache，建立在Heap之上的COM字符串内存管理2.5.4　题外话和相关讨论CRT Debug Heap一定对Debug有帮助吗C++中new操作符的尴尬2.6　理清多个线程对资源的竞争：同步和锁2.6.1　句柄泄漏、死锁和线程争用，三个典型问题句柄泄漏（Handle Leak）死锁（Deadlock）线程争用 （contention）2.6.2　Windbg中的对应排错！
handle 检查句柄信息！
htrace 检查操作句柄的历史记录！
cs 列出CriticalSection的详细信息排查CriticalSection leak（ Orphan CriticalSection）Invalid handle exception案例分析：ArrayList.Add的时候发生IndexOutOfRangeException问题描述这个异常不简单具体操作结论2.7　调试和设计2.7.1　一位热心朋友的提问案例分析：反被聪明误第3章　.NET Framework的原理

和SOS调试——剖析CLR程序和CLR本身3.1　MetaData、JIT、GC和Exception的关键点3.1.1　MetaData（元资料）和引擎初始化3.1.2　JIT动态编译3.1.3　GC 内存管理3.1.4　Exception Handling异常处理3.2　用Windbg探索CLR的实现3.2.1　开源的CLR实现：Rotor3.2.2　对一个Hello world的WinForm程序庖丁解牛mscoree！
_CorExeMain CLR引擎的入口EEStartupHelper 重要的引擎初始化函数mscorwks！
SystemDomain::ExecuteMainMethod 执行托管代码的入口CallDescr /MakeJitWorker Jit引擎发动的地方NtUserWaitMessage 托管程序完成加载gc_heap::allocate_more_space/ GCHeap::GarbageCollect 通过GC管理内存的分配和　　释放AppDomain，ThreadPool，Exception，StackWalk，Security都是有趣的话题3.3　通过SOS快捷方便地调试托管程序3.3.1　CLR让托管程序的调试变得非常简单3.3.2　SOS的命令介绍3.4　用简单的程序演示SOS的常见操作3.4.1　.load SOS 加载SOS到Windbg3.4.2　！
dumpheap 统计托管内存使用信息3.4.3　！
do 显示托管对象的详细信息3.4.4　！
gcroot 查找托管对象的引用关系案例分析：ASP.NET High CPU和更多的CLR命令演示！
threads查看托管线程！
tp查看线程池和CPU占用率！
SyncBlk查看托管线程的lock！
ip2md 映像内存地址到托管函数名！
savemodule 保存模块到本地以便用reflector分析著名的blog:If broken it is，fix it you should3.5　题外话和相关讨论3.5.1　ReleaseCOMObject 释放COM对象时候的两难困境3.5.2　PInvoke应该Pin住内存防止崩溃3.5.3　Pin住内存又会导致内存碎片3.5.4　臭名昭著的mixed DLL loading deadlock3.5.5　有趣且有用的练习和更多的资料第4章　崩溃，性能和资源泄漏——分享一些经验4.1　排错开始前的准备工作4.1.1　用正确的态度对待问题4.1.2　用简单的提问缩小排错的范围4.1.3　通过MPS REPORT获取系统的详细信息4.1.4　通过简单的Dump分析获取基本信息4.2　崩溃（Crash）4.2.1　崩溃的万千种不同死相4.2.2　准确获取DumpAdplus：最容易上手的dump脚本华生医生（dr Watson）通过Image File Execution Options让调试器随目标程序一起启动COM+和ASP.NET的dump获取需要特殊配置4.2.3　crash dump中需要重点关注的信息案例分析：VC程序的崩溃问题描述MessageBox 嵌套调用从源代码中发现的疑点从This指针找崩溃的根源结论4.2.4　小结和更多的资源4.2.5　题外话和相关讨论HeapCorruptionStackCorruption4.3　性能（Performance）4.3.1　“你真牛，不如你再给我缩短10秒吧！
”不是想要多快就能调到多快4.3.2　性能调优的步骤，CPU利用率是关键4.3.3　无所不知的性能监视器使用性能监视器的基本步骤重要的计数器案例分析：博客园的性能问题案例分析：堵塞在SqlCommand.ExecuteReader上就一定在等sql吗问题背景案例分析：堵塞在Assembly.Load上的deadlock问题背景案例分析：196个线程织成的一张网问题背景小结4.3.4　用Profiler精确定位性能瓶颈案例分析：DataTable中foreach和for loop性能差了50%问题背景4.3.5　题外话和相关讨论Task manager跟performance monitor的差别性能监视器的超级用法C++跟C#到底谁快没有profiler怎么办4.4　资源泄漏（Resource Leak）4.4.1　资源泄漏分轻重缓急4.4.2　内存泄漏排错的基本步骤泄漏了什么，谁分配的，为什么无法释放定位泄漏内存的类型和增长趋势区分managed heap leak和native leak案例分析：IE7的内存泄漏问题描述重现问题和基本分析用传统的Pageheap+UMDH找到问题根源方便强大的IIS Diagnostics工具结论分析IIS Diag4.4.3　托管内存泄漏案例分析：object chain让排错简单明了问题背景案例分析：一个bt的案例碎片的其他原因4.4.4　句柄泄漏（Handle Leak）4.4.5　题外话和相关讨论GDI LeakDesktop heap issue更多的资源

编辑推荐

《Windows用户态程序高效排错》案例丰富，对现实中的实际问题进行了研究，并和读者一起分析解决办法；《Windows用户态程序高效排错》的写作思路为先给出问题描述，然后提供线索，再进行分析，让读者在阅读中也进行思考，以提高实际解决问题的能力。
《Windows用户态程序高效排错》适合希望学习排错、调试知识的软件开发、测试人员，希望深入学习Windows系统上用户态程序的排错知识的软件开发、测试人员。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com