

多线程在 IOCP 服务器测试中的研究和应用^①

Study and Application of Multi-Thread in IOCP Server Test

段战钦 彭楚武 袁 峰 (湖南大学 电气与信息工程学院 湖南 长沙 410082)

摘 要: IOCP(I/O Completion Port)又称 I/O 完成端口,是一种在 Windows 平台上比较成熟的、伸缩性最好的 I/O 模型,采用这种模型对于管理为数众多的套接字可以达到最佳的系统性能。Web 服务器就是典型的例子。在本文中主要讲述如何利用多线程技术对 IOCP 服务器进行性能测试,并提供了一个成功实例。

关键词: IOCP 套接字 多线程

随着网络技术的发展,对服务器性能的要求也越来越高。在 Windows 平台上开发为大量套接字 I/O 请求提供服务的高性能服务器多采用 IOCP 模型。如 Web 服务器和网络游戏服务器^[1]。要达到 IOCP 服务器的高性能,性能测试必不可少。针对服务器应用的性能测试工具有很多,但是能满足对 IOCP 服务器测试要求并且有完美表现的只有惠普公司的 LoadRunner。LoadRunner 优点众多,但是操作复杂,价格昂贵^[2]。鉴于此,通过研究发现利用多线程技术模拟多个用户或终端来实现 IOCP 服务器的性能测试,成本低,并且容易实现,在资金不足或者需要降低成本的情况下,是一个很好的选择。

1 IOCP机制

IOCP(I/O Completion Port)又称 I/O 完成端口,是 WIN 平台下同时管理数千个连接的最为高效的网络模型,完成端口机制针对操作系统内部进行了优化,在 Windows NT 和 Windows 2000 上,使用完成端口的重叠 I/O 机制能够真正扩大系统的响应规模,这是高性能服务器管理数千套接字的核心技术。完成端口模型如图 1 所示:

完成端口模型要求创建一个 Win32 完成端口对象来对重叠 I/O 请求进行管理,并通过创建一定数量的工作线程(Work Thread),来为已经完成的重叠 I/O 请求提供服务。其实,可以把完成端口堪称系统维护

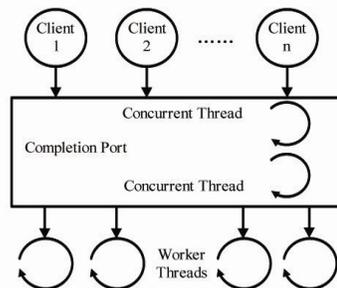


图 1 完成端口模型

的一个队列,操作系统把重叠 I/O 操作完成的事件通知放入该队列,由于是“操作完成”的事件通知,故取名为完成端口。一个完成端口被创建以后,可以和多个文件句柄进行关联(文件句柄可以是真正的文件句柄,也可以是 Socket 句柄或命名管道),并在关联后的句柄上进行重叠 I/O 操作。当 I/O 操作完成后,一个重叠 I/O 完成的事件通知就会被排在此端口的完成队列上,此时,某个工作者线程将会被唤醒来完成端口服务,执行特定的处理工作。一般来说,一个应用程序可以创建多个工作者线程来处理完成端口上的通知事件,工作线程的数量依赖于程序的具体需要^[3]。

2 多线程技术在测试中应用

IOCP 服务器的最大优点是管理为数众多的套接字,为这些套接字 I/O 请求提供服务,而测试服务器处理众多套接字请求的能力就成为测试服务器中非常

^① 基金项目:教育部博士点新教师项目(20070532054)

收稿时间:2008-09-18

重要的一项。

在这里我们利用多线程技术来完成对 IOCP 服务器的性能测试。所谓多线程,就是通过系统调度使几个具有不同功能的程序流(即线程)同时并行地运行。多线程是在程序内部实现多任务的能力,程序可由若干单独执行的线程构成。利用多线程可实现任务内的各子任务并行运行,提高事务处理效率。

2.1 测试程序的介绍

Win32 提供了一系列的 API 函数来完成线程的创建、挂起、恢复、终止以及通信等工作。笔者是在 Borland C++ Builder 环境下实现多线程的,因此使用了 VCL 类库提供的 TThread 类,其实在 TThread 类中封装的有 Windows 中关于线程机制的 Windows API 函数。测试模型如图 2 所示:

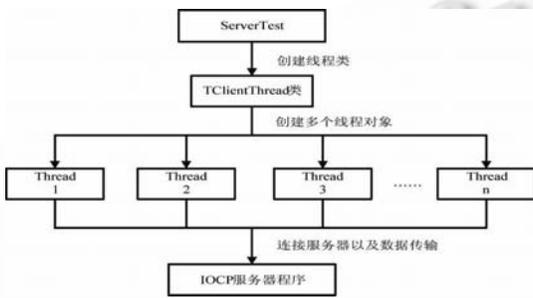


图 2 测试模型

测试程序的实现步骤:首先在测试程序中创建一个 TThread 类的派生类 TClientThread,然后在线程类中创建一个 TClientSocket 对象,用它用来模拟用户或者终端,TClientSocket 本身带有 Socket 即套接字,套接字可以用来向服务器发出连接请求以及数据传输的请求。在套接字连接服务器之前,首先要设置需要连接的服务器 IP 以及端口等属性,连上以后将 TWInSocketStream 和套接字关联,通过 TWInSocketStream 把数据发送到服务器。关于异常处理,当没连接上的时候要关闭套接字,终止线程;当通信异常中断时,关闭套接字,终止线程。在每次关闭套接字后线程都将修改相关信息,并显示在主界面,然后返回线程函数即终止此线程(采用线程函数返回的方式终止线程可以确保线程资源被正确的清除)。线程类的构造函数中包含了一些参数,这些参数用来标识线程数量、线程的状态、通信状态等,在正常以及异常通信时,通过获取这些参数值可以在程序的主界面上显示线程和通信的情况。析构函数里需要释放 TClientSocket 等动态分配了内存的对象或变量,以防止内存的泄露。在主线程中可以通过

GlobalMemoryStatus 函数和 NtQuerySystemInformation 函数来分别实时的获取内存和 CPU 的使用情况,以此来掌握测试程序的性能[4,5]。

对于多线程程序来说当没有过多的线程间通信时,会降低编程的复杂度,同时 Windows 操作系统的运行性能也最好,在本测试程序中基本不存在线程间通信。关于实现千以上的套接字连接请求,默认情况下,一个线程的栈要预留 1MB 的内存空间,而一个进程中可用的内存只有 2GB,理论上一个进程最多可以创建 2048 个线程。但是考虑到线程中套接字要连接服务器并且需要通信,还有响应速度的要求,通过实验当创建小于 200 个线程时,操作系统性能比较稳定。如果要实现千以上的套接字连接请求,可以通过多台电脑同时运行测试程序。部分源代码如下:

```

//套接字的设置
TstClientSocket = new TClientSocket(NULL);
TstClientSocket->ClientType = ctBlocking; //
阻塞方式
//需要访问的服务器 IP 地址
TstClientSocket->Address = "129.71.71.71";
TstClientSocket->Port = 8080; //需要访问的服务器端口
TstClientSocket->Active = true; //发出连接请求
//数据传输
while(!Terminated && TstClientSocket->Active)
{
    packageStream->Write(buffer, 50);
    Sleep(10000); } //循环周期发送数据
packageStream->Free(); //释放
//获取内存信息的函数
VOID GlobalMemoryStatus(LPMEMORYSTATUS
lpBuffer);
//获取 CPU 信息的函数
NTSTATUS WINAPI NtQuerySystemInformation(
__in SYSTEM_INFORMATION_CLASS
SystemInformationClass,
__inout PVOID SystemInformation,
__in ULONG SystemInformationLength,
__out_opt PULONG ReturnLength);
  
```

2.2 测试内容

性能测试对服务器来说很重要。在这里利用多线程技术虚拟用户来测试服务器,主要进行负载测试、压力测试、强度测试、并发(用户)测试、大数据量测试和可靠性测试。这些测试侧重点不同,并且相互联系。

例如, 压力测试侧重压力大小, 而负载测试强调压力持续的时间; 同时也都是对系统施加了一定的压力来测试。因此在实际测试中绝不能割裂它们的内部联系, 应分析它们之间的关系, 以一种高效的方式来规划与设计性能测试。

3 实例

笔者参与一个工程机械远程监控系统项目的开发, 该系统在原有智能监控装置的基础上进行扩展, 通过 GPRS 无线上网技术采集机载终端上传的工程机械工况数据。在此基础上实现远程监控工程机械的工况、车况、位置为大型工程项目的业主和车主提供诸如考勤、工作负荷、调度、违规操作等基本的管理信息, 了解机械位置及时发现被盗情况; 实现远程调试工程机械上的机载终端减少技术支持人员的出差费用、提供响应速度、降低调试工作量关系。系统服务器要求支持与不小于 1 千台终端进行通信, 每台机载终端 10 秒送一次工况数据, 每帧工况数据不带图片小于 1KB, 带图片小于 50K, 因此系统服务器采用了完成端口模型。系统的通信流程如图 3 所示:

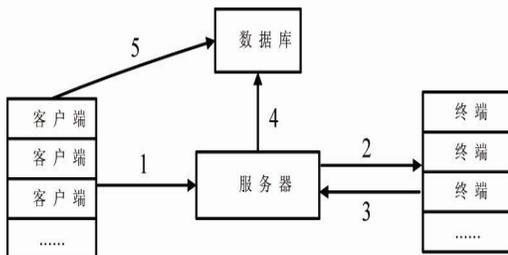


图 3 系统通信流程

系统通信流程的简单介绍:

- 1) 客户端向服务器发送 DTU(终端)数据请求命令。
 - 2) 当服务器验证了客户端的权限后, 将命令转发给终端, 若无权限, 会话终止。
 - 3) 终端获取命令后, 将相应的数据发送给服务器。服务器进行分帧操作, 并通过通信协议将相应的信息(如工况, GPS 数据等)存入内存。
 - 4) 服务器根据一定的存储策略将相应数据存储到数据库中。
 - 5) 客户端定时(通过定时器完成)查询数据库信息, 将数据进行刷新。
- 除了上面的步骤以外, 终端还会周期地向服务器发

送心跳包(确认终端在线)和工况数据, 并且在出现故障时自动向服务器发送故障信息、工况数据和车况数据。

对服务器程序的测试, 性能测试是重中之重了, 之前考虑过用 HP(Mercury)公司的工业标准级性能测试工具 LoadRunner, LoadRunner 是一种预测系统行为和性能的负载测试工具, 功能点: 创建虚拟用户, 创建真实的负载, 定位性能问题, 分析结果以精确定位问题所在, 重复测试保证系统发布的高性能等; 虚拟用户上限数量可以达到上万。所有的条件都符合项目的要求, 而且也是业界公认的最好的性能测试工具, 但是价格非常昂贵, 操作复杂, 对于承担项目开发的公司来说, 成本太高, 所以不适宜。通过查阅和研究, 发现利用多线程技术创建多个虚拟机载终端, 并通过虚拟终端向服务器发送连接请求以及传输数据, 同样可以完成对系统服务器的性能测试。测试的内容主要包括: 1) 系统服务器可以承载的终端连接数目; 2) 一定数量的终端同时周期的发送心跳包和工况数据; 3) 8 个小时的负载测试; 4) 当大数据量时系统服务器的承载能力; 5) 当出现坏连接时系统服务器的处理能力; 6) 当出现错误数据时服务器的处理能力。

4 结束语

通过对多线程技术的研究, 开发了一个测试 IOCP 服务器性能的程序, 并应用在某工程机械远程监控系统服务器性能测试中, 利用多线程技术创建多个虚拟机载终端, 创建真实负载, 不仅能完成监控系统服务器测试内容, 并且容易实现、成本低、效果良好。因此在资金不足或者需要降低成本的情况下, 本文所介绍的利用多线程技术测试 IOCP 服务器性能是一个很好的解决方案。

参考文献

- 1 王艳平, 张越. Windows 网络与通信程序设计. 上海: 人民邮电出版社, 2006: 67-72.
- 2 陈绍英, 刘建华, 金成姬. LoadRunner 性能测试实战. 北京: 电子工业出版社, 2007: 1-41.
- 3 Jones A. Windows 网络编程技术. 北京: 机械工业出版社, 2000: 198-206.
- 4 Richter J. Programming Applications for Microsoft Windows. 北京: 机械工业出版社, 2008: 126-299.
- 5 Beveridge J, Wiener R. 侯捷译. Win32 多线程程序设计. 武汉: 华中科技大学出版社, 2002: 397-411.