

完成端口在视频会议服务器性能测试中的应用

汤国军 王翠荣 赵煜辉 张岩峰
(东北大学信息科学与工程学院, 辽宁 沈阳 110004)

摘要: 视频会议服务器的性能测试需要模拟大量的客户端和真实的流量,本文提出了一种基于 Windows 完成端口模拟客户端来产生真实流量,从而进行视频会议服务器性能测试的方法,并描述了设计与实现过程,最后通过理论分析和实验验证,表明该方法在视频会议服务器性能测试中的准确性和低开销性。

关键词: 视频会议; 服务器; 性能测试; 完成端口

中图分类号: TP393

引言

在视频会议系统中,服务器是整个系统的核心,它关系到整个系统性能的优劣。为揭示系统中潜在的性能问题,需要对服务器进行合理的性能测试。

目前,性能测试主要是通过自动化的测试工具模拟多种正常、峰值以及异常负载条件来对系统的各项性能指标进行测试^[1]。在视频会议服务器的性能测试中,同样也需要模拟这几种负载条件下的流量。

本文就视频会议服务器的性能测试中,如何产生大量模拟客户端,如何模拟真实的流量,进行了讨论。文中采用理论分析和实际测试相结合的方法。首先,在理论上,采用了一种理论分析方法对服务器的性能进行分析预测。其次,在实际测试上,采用了一种基于 Windows 完成端口 (IOCP) 的测试模型,并对该模型进行了设计与实现。

1 相关研究

目前,用于服务器性能测试的商用软件主要有 Mercury Interactive 的 LoadRunner, Rational 的 Teamtest, Compuware 的 QALoad 以及 Empirix 的 e-Load^[2-3]。还有一个颇受欢迎的网络性能测试工具是 NetPerf^[4]。NetPerf 主要针对基于 TCP 或 UDP

的传输,其测试结果所反映的是一个系统能够以多快的速度向另外一个系统发送数据,以及另外一个系统能够以多快的速度接收数据。

虽然这些工具在测试自动化方面都很出色^[5],但它们有的侧重于 web 应用,有的不能较真实地模拟操作情景,总之它们不适用于视频会议服务器的性能测试。而且,现有的测试工具所采用的技术都是利用线程来模拟客户端^[6],而当线程达到一定数量的时候,线程的上下文切换就会成为一个很大的开销,这不但限制了在低端机器上可以产生的模拟客户端的数量,而且还会影响测试结果的准确性。

Windows 的完成端口 (IOCP) 可以解决上述问题。完成端口是一种 Windows 核心对象,是服务请求和服务线程之间的一个高效的异步对象机制。其工作原理^[7]如图 1 所示。

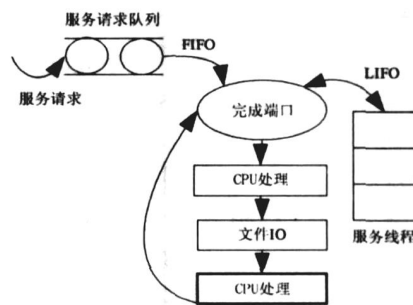


图 1 完成端口原理

Fig. 1 Principle of IOCP

在完成端口上有几个服务线程,它们为到达完成端口的服务请求提供服务。当有服务请求到达时,如果有可用的服务线程,则激活该线程。如果没有服务线程,则服务请求进入请求队列,该队列采用先进先出 (FIFO) 的策略,这样可以保证这些请求公

收稿日期: 2007-05-17

基金项目: 国家自然科学基金 (60273078); 河北省科技厅博士基金项目 (55470160-3)

第一作者: 男, 1983 年生, 硕士生

E-mail: towerness@gmail.com

平地得到服务。一般地,线程的个数等于 CPU 的个数。当有多个线程时,完成端口采用后进先出(LIFO)来管理这些线程,这可以显著降低线程上下文切换所造成的开销。

使用完成端口解决了模拟客户端数量受限的问题,但在视频会议服务器的性能测试中,还需要能模拟客户端的操作,以产生较真实的数据流。本文采用的测试模型中,根据视频会议系统的具体情况,在测试的开始阶段,为每一个模拟客户端定义了一些操作,客户端启动之后,自动执行这些操作,这样就可以较真实地模拟客户端的数据流。

2 测试方法设计与实现

本文描述的测试主要针对视频会议服务器的最大并发连接数和数据吞吐能力这两方面的测试。其系统框架如图 2 所示。

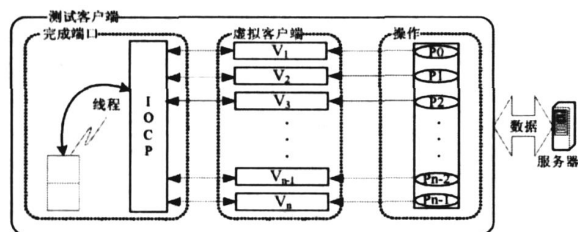


图 2 测试系统框架

Fig. 2 Architecture of testing system

其中, $V_1, V_2, V_3 \dots V_{n-1}, V_n$ 表示测试系统中的虚拟客户端, $P_0, P_1, P_2 \dots P_{n-2}, P_{n-1}$ 表示虚拟客户端的 n 个操作。

在视频会议系统中, TCP Socket 用于传输视频会议系统中电子白板操作和一些控制信息,而 UDP Socket 则负责收发音视频数据。因此,在本测试系统中,每一个模拟客户端都有一个 TCP Socket 和一个 UDP Socket,并且在操作中定义了一个操作序列。当系统启动后,模拟客户端按照定义好的操作序列,自动循环操作。另外,系统中采用了 IOCP,这样就可以不使用线程来模拟客户端,而只需将每个模拟客户端的 Socket 与完成端口建立关联即可。

为实现图 2 所示的测试系统,本文设计了 5 个类,如图 3 所示。它们对应于图 2 的各个部分。

本文给出的测试系统的时序图如图 4 所示。

(1) 启动系统,通过调用 CClientTest 的 StartIOCP 方法,而 StartIOCP 调用 CIOCP 类的 RunIOCP 方法;

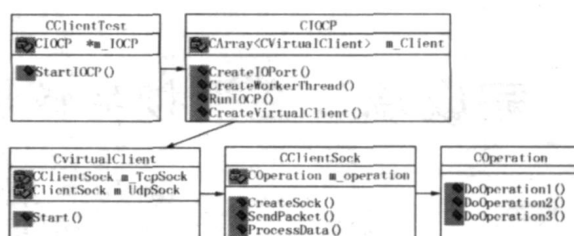


图 3 测试系统类图

Fig. 3 Class diagram of testing system

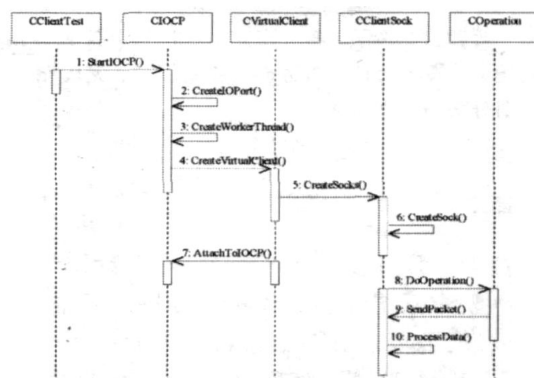


图 4 测试系统时序图

Fig. 4 Sequence diagram of testing system

(2) 调用 CIOCP 的 CreateIOPort 方法创建完成端口 m_IOCP;

(3) 调用 CIOCP 类的 CreateWorkerThread 方法在完成端口上创建工作者线程;

(4) 创建模拟客户端,客户端有两个 Socket, CVirtualClient 的 CreateSocks 方法分别创建;

(5) 通过调用 AttachToIOCP 方法,将系统中模拟客户端的 Socket 与完成端口关联;

(6) 模拟客户端执行 COperation 中的操作。

本系统的实现采用 C++ 语言在 VC 6.0 环境下完成,并且已应用到实际测试中。同以前使用线程模拟客户端的方式相比,测试效果更好,效率更高。

3 理论分析与实验验证

3.1 理论分析

定义以下几个常量

C:最大 I/O 吞吐量(Capacity),也就是每秒能处理的包数,单位,包数/秒(packets/s)。

S:会话(Session),每个模拟客户端对应一个会话,它包括 $\{P_0, P_1 \dots P_{N-1}\}$, P_i 是会话中的第 i 个包, N 是会话中的总操作数,也即总包数。

I :产生模拟客户端的时间间隔,单位,秒(s)。

O :开销(Overhead),这里主要是指线程上下文切换的开销,单位,秒(s)。

模拟客户端的操作过程如图 5 所示,每个会话模拟一个客户端,每个会话由 $P_0, P_1 \dots P_{N-1}$ 这 N 个操作组成,系统每隔时间 I 启动一个会话。会话启动后,循环执行这 N 个操作。之所以设置时间间隔是为了消除客户端同服务器建立连接的时间对测试结果造成的影响^[8]。

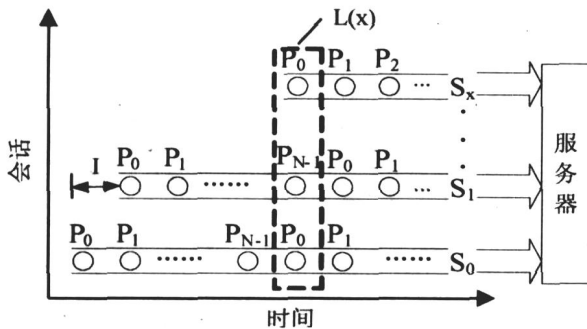


图 5 模拟客户端操作模型

Fig. 5 Model of virtual client operation

服务器性能测试应该由理论分析,实验验证以及针对测试结果做相应的调整这三部分组成。有了理论上的分析结果,就可以根据这个结果进行相关的测试,再由测试结果对服务器做相应的调整,这样才可以更好地达到性能测试的目的。下面给出一种从理论上计算服务器某些性能指标的方法^[8]。

服务器的最大 I/O 吞吐量 C 是已知的。假设当前的会话数为 x ,如图 5 所示, $L(x)$ 为某时刻服务器的负载。 $nL(x)$ 为某时刻到达服务器的数据包个数,服务器处理 $L(x)$ 所需的时间计为 $T(x)$ 。则有以下 3 个公式

$$nL(x) = x \tag{1}$$

$$T(x) = nL(x) (1/C + O) = x(1/C + O) \tag{2}$$

$$f_{SLA}(S) = \sum_{i=0}^{N-1} f_{SLA}(P_i), \text{ 其中 } f_{SLA}(P_i) \text{ 为处理 } P_i \text{ 所需要的时间。}$$

处理负载 $L(x)$ 所需的时间 $T(x)$ 为 $x(1/C + O)$,由于使用了完成端口,线程上下文切换造成的开销 O 可以忽略不计,则 $T(x)$ 减少为 x/C ,这样就降低了由于上下文切换造成的额外开销。

当有 x 个会话,每个会话有 $P_0, P_1 \dots P_{N-1}$ 共 N 个操作,那么处理一个会话需要的时间为 $NT(x)$ 。根据服务等级协议(Service Level Agree-

ment, SLA),这个时间不大于 $f_{SLA}(S)$,则可得式(4)

$$Nx/C < f_{SLA}(S) \tag{4}$$

其中, N 和 C 都是固定的值,而从 $f_{SLA}(S)$ 可以从 SLA 协议得知。从这个式子可以求得理论上服务器能承载的最大客户端数 x 。有了这个值,测试时可以逐步逼近客户端数的理论值,从而使服务器性能测试更准确。

3.2 实验验证

(1) 验证环境

两台 Pentium(R) 4CPU 1.70 GHZ,512 M 内存台式机。一台用于模拟客户端,一台用作服务器。

(2) 验证结果及分析

客户端饥饿数测试结果如图 6 所示。在使用线程模拟客户端时,随着客户端数的增加,得不到服务的客户端数也相应的增长。相反,使用完成端口时,由于完成端口处理服务采用先进先出^[7](FIFO)的方式,因此没有出现客户端得不到服务的情况。

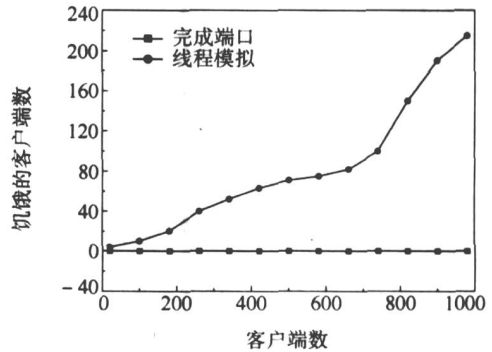


图 6 客户端饥饿状况对比

Fig. 6 Comparison of client-side starvation

单位时间线程上下文切换次数测试结果如图 7 所示。从图 7 可以看到,使用线程模拟客户端时,线

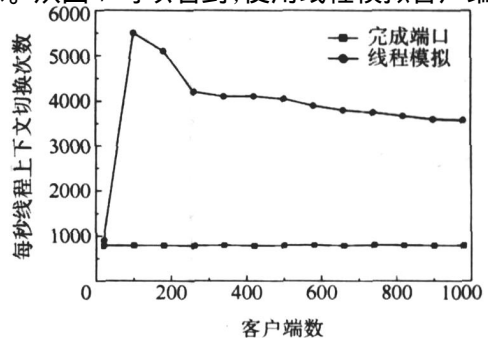


图 7 线程每秒上下文切换次数比较

Fig. 7 Comparison of the number of context switches per second

程上下文频繁切换,导致了大量的额外开销。而使用完成端口时,显著减少了线程上下文切换次数,而且线程的上下文切换次数基本和操作系统正常运行时的次数相当。这就说明了本文所使用的方法可以显著减少线程上下文切换,使用这个方法,可以在低端机器上产生大量的模拟客户端。

4 结论

本文针对视频会议系统的特点,提出了一种基于完成端口产生模拟客户端进行视频会议服务器性能测试的方法,并对该方法进行了设计实现,最后通过实验验证了该法的可行性。由实验结果得知,在视频会议服务器性能测试中,基于完成端口产生模拟客户端的方法与使用线程模拟客户端的方法相比,开销更小,准确度更高,而且在低端的机器上也可以模拟更多的客户端。本文实现的方法已应用于秦皇岛东大软件视频会议系统的测试中。

参考文献:

- [1] 周应超, 苗彦超, 郝敏, 等. 视频服务器性能测试研究[J]. 计算机工程, 2004, 30(14): 133 - 135.
- [2] HAVELUND K, STOLLER S D, SHMUEL. Benchmark and Framework for Encouraging Research on Multi-Threaded Testing Tools. IEEE CNF 2003[C]// Parallel and Distributed Processing Symposium, Proceedings International, 2003; 22 - 26.
- [3] AMIR K. Mercury Interactive Corporation. Use of a Single Thread to Support Multiple Network Connections for Server Load Testing[P]. US, 6754791. June 2004.
- [4] 唐海娜, 李俊. 网络性能监测技术综述[J]. 计算机应用研究, 2004, 21(8): 10 - 13.
- [5] TRETMAINS J, BRINKSMA E, TOR X. Automated Model Based Testing[C] 1st European Conference on Model-Driven Software Engineering. Nurnberg, Germany, December 2003.
- [6] CHEN Y K, LOCKARD E N, DURASOFF M D, et al. Method, System, and Product for Assessing a Server Application Performance [P]. US, 5812780. September 1998.
- [7] JIM B, POBERT W. Win32 多线程程序设计[M]. 侯捷, 译. 武汉:华中科技大学出版社, 2002:149 - 190.
- [8] KIM G. A method of generating massive virtual clients and model-based performance test, quality software[C] Fifth International Conference, 2005: 250 - 254.

Application of IOCP mechanism in performance testing of video conference server

TANG GuoJun WANG CuiRong ZHAO YuHui ZHANG YanFeng

(College of Information Science and Engineering, Northeastern University, Shenyang Liaoning 110004, China)

Abstract: Performance testing of a Video Conference Server needs massive virtual clients and realistic traffic. This paper proposes a method to generate massive virtual clients and realistic traffic for performance testing of Video Conference Server. The approach exploits the Windows I/O Completion Port (IOCP). It's designing and implementation details are described in this paper. Analysis and experimental results prove that this approach is accurate and cost-effective in testing performance of Video Conference Server.

Key words: video conference; server; performance test; IOCP