doi: 10.3969/j.issn.1006-1576.2010.08.015

基于 GL Studio 的分布式虚拟训练系统关键技术

朱敏[1]，陈立奎[2]，王宏伟[1]，范绍里[1]
（1. 海军航空工程学院 战略导弹工程系，山东 烟台 264001；
2. 中国人民解放军 92538 部队，辽宁 大连 116041）

摘要：为在 GL Studio 中实现分布式虚拟仪表的远程交互，对该虚拟训练系统的关键技术进行研究。在分析分布式虚拟训练系统运行环境和操作流程的基础上，给出系统的总体设计框架；设计出基于混合协议的网络通信方式，保证数据快速、可靠传输，采用多线程技术提高系统的响应速度，并使用信号量实现线程同步，避免了外部控制时间和资源的浪费；给出了在 GL Studio 中实现多线程网络通信的关键步骤。研究结果为虚拟训练系统由单机模式向分布式协同训练模式发展提供可借鉴的技术基础，具有一定工业应用前景。

关键词：GL studio；分布式；虚拟训练；Windows sockets；多线程
中图分类号：TP391 **文献标识码：**A

Key Technology of Distributed Virtual Training System Based on GL Studio

Zhu Min[1], Chen Likui[2], Wang Hongwei[1], Fan Shaoli[1]
(1. Dept. of Strategic Missile Engineering, Naval Aeronautical & Astronautical University, Yantai 264001, China;
2. No. 92538 Unit of PLA, Dalian 116041, China)

Abstract: In order to realize long-distance distributed interactive operation of virtual instrument in GL Studio, the key technology of this virtual training system is researched. The running environment and operation flow of virtual training system are analyzed, and the system overall design framework is given. Then, network communication based on hybrid protocol is designed to ensure data transfer fast and reliably. Multi-thread technology is used to improve the system response speed. Semaphore is adopted to synchronize threads, and the waste of external control time and resource is avoided. Finally, the key steps of multi-threaded and network communication in GL Studio are given. The research results provide referenced technical foundation for virtual training system which is concerting from single machine model to distributed collaborative training model, and has a certain industrial applications.

Keywords: GL studio; distribution; virtual training; windows sockets; multi-thread

0 引言

GL Studio 是一种专业的虚拟仪表仿真平台，以“所见即所得”方式建立实时、交互的三维图形，比直接使用 OpenGL 语言编程生成三维视景节省了大量的时间，而且也降低了对计算机硬件要求[1]。因此，GL Studio 现已广泛应用于飞机地勤训练[2]、飞机驾驶座舱训练[3-4]、导弹指挥仪训练[5]和装甲车辆训练[6]等领域中，这些虚拟训练系统都是基于单机操作的。

近年来，随着广域信息共享的需求日益增长，给虚拟训练领域带来了深刻的影响，虚拟训练系统已不再局限于单机操作，远程分布式虚拟训练的需求越来越迫切，但 GL Studio 中并没有提供远程交互功能，如何在 GL Studio 中实现分布式虚拟仪表的远程交互成为研究的关键问题。

故在 GL Studio 设计器生成 C++代码的基础上，集成开发 Windows Sockets、多线程等关键技术，以实现基于 GL Studio 的分布式虚拟仪表的远程交互。

1 系统设计

在分布式虚拟训练中，分布在各网际结点的虚拟仪表间的网络通信是其核心问题。由于网络通信面临着异构的物理环境、有限的网络带宽及很高的实时性要求等问题，故设计分布式虚拟训练系统必须要提供透明的网络通信支持，要支持仿真协议所规定的各种消息的实时发送接收，并补偿网络延迟的影响等。

在分布式虚拟训练中，各虚拟仪表完全自治，不存在一个中心服务器结点。虚拟训练的属性数据完全分布在各个结点上，各个结点负责对所绘制的虚拟仪表的人机交互，并在状态改变时，按标准协议向其它结点发送有关消息，其它结点在收到消息后，解释所收到的消息并根据消息内容作出相应的改变。虚拟仪表间远程通信的总体框架如图 1。

由图 1 可以看出，分布式虚拟训练系统中涉及

收稿日期：2010-02-24；修回日期：2010-04-19
基金项目：总装备部“十一五”预研基金资助（51328040107）
作者简介：朱敏（1978-），男，安徽人，博士，讲师，从事仿真训练、数字图像处理、电子对抗方面的研究。

的关键技术有：网间通信、多线程及线程间通信、GL Studio 生成代码和 C++融合问题。

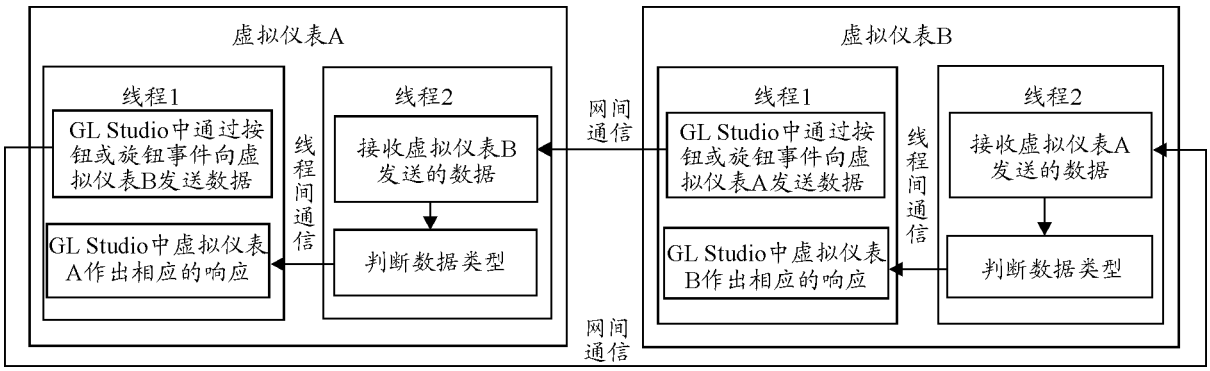

图 1　基于 GL Studio 分布式虚拟训练系统总体框架

2　关键技术

2.1　基于混合协议的网络通信

Windows Sockets 是 Microsoft Windows 的网络程序设计接口，以动态链接库的形式提供给用户使用[7]。Windows Sockets 基于 TCP/IP 协议，提供面向连接（TCP 协议）和面向无连接（UDP 协议）2 种通信方式。这 2 种通信方式各有优缺点，基于 TCP 协议的通信方式提供可靠的数据传输服务，数据无差错、无重复地发送，且按发送顺序接收，但当一个数据包发送不到目的地时会反复地重发，在网络出现异常时程序就会陷入停滞，在网络恢复正常后程序不具备“自愈”的能力；基于 UDP 协议的通信方式具有更高的性能，数据包以独立形式发送，但不提供无错保证，数据可能丢失或重复，并且接收顺序混乱。

在工程实践中，分布式虚拟训练系统的数据传输经常要跨越集线器、交换机、路由器等网络设备，经过电缆、光缆、微波甚至卫星等传输介质，被各种网络管理软件转发、过滤、检验、分配，此过程极其复杂，而且在网络上同时承载着企业经济管理系统、各种信息查询、甚至个人娱乐等大量数据传输。这给远程分布式虚拟仪表程序的可靠性、顽强性和通信速度带来极大的考验。为适应工业现场的需求，必须整合 TCP 协议和 UDP 协议的技术优势，开发实用的通信程序[8]。混合协议通信方式如图 2。

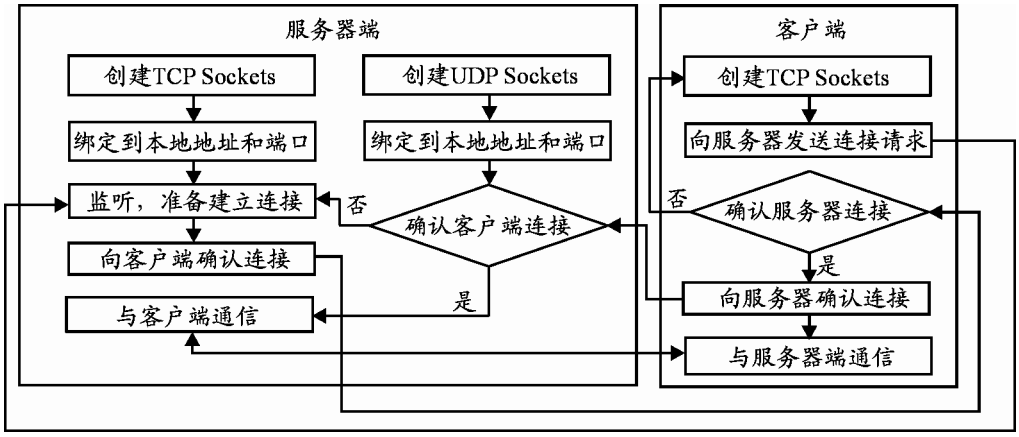

图 2　TCP/UDP 混合通信流程图

服务器端在网络上监听 TCP 连接的请求，连接建立后，即应用 TCP 协议向客户端发送数据。同时建立一个 UDP Sockets，接收来自客户端的确认信息。一旦网络出现故障，收不到确认信息，立即断开 TCP 连接，并重新进入 TCP 连接监听状态。

客户端首先建立 TCP 连接，然后应用 TCP 协议接收数据。如果正常接收数据，则每次向服务器端发送一个确认信息，如果收不到数据就断开原来 TCP 连接，重新建立一个 TCP 连接。

采用 TCP 协议与 UDP 协议混合编程可实现可靠、实时的网络通信，在网络正常时利用 TCP 协议的可靠性，保证数据正确传输；在网络异常时发挥 UDP 协议的灵活性，保证通信及时恢复。

2.2　多线程及线程间同步

在分布式虚拟仪表训练系统中，信号的实时传输是仪表准确响应的前提，只有仪表准确响应，才

能保证虚拟训练的真实性。若以一定的时钟间隔来实现信号的传输，时钟间隔能预先设定，但是在连续读取数据的同时，还要完成各种控制功能。在时钟控件回调函数连续执行过程中，程序对其它控件的响应会很迟钝，会有很大的延迟，有时会导致系统的死机，严重影响虚拟仪表训练的操作[9]。

在分布式虚拟训练系统中，以用户界面接口为主线程，以网络通信、数据处理、训练报告生成为次线程。很多情况下，需要多个线程互相协助来完成同一个任务。但线程很难从外部进行控制。利用线程同步技术可以使线程彼此交互，从而避免外部控制时间和资源的浪费。

目前，在编程中主要使用临界区、互斥对象和信号量 3 种方法实现线程同步。临界区是一种保证在某一时刻只有一个线程能访问数据的简便办法。互斥对象跟临界区很相似，但远比临界区复杂，因为使用互斥不仅能在同一应用程序不同线程中实现资源的安全共享，还可在不同应用程序的线程之间实现对资源的安全共享；除了使用临界区与互斥对象可以完成线程间的同步外，还可以使用信号量。使用信号量的优点是：信号允许多个线程同时使用共享资源。在分布式虚拟训练系统中，网络通信为插入线程，主线程、数据处理和训练报告生成为读取线程，插入线程不断将数据放入共享的缓冲，读取线程不断从缓冲取出数据。插入线程必须等读取线程取走数据后才能再放新数据（不覆盖数据），读取线程必须等插入线程放入新数据后才能去取（不重复）。并且网络通信为多对多关系，可能会出现多个线程访问共享资源，利用信号量实现线程间同步的具体实现方法见文献[10]。

2.3 GL Studio 中多线程网络通信的实现

GL Studio 生成的代码有 3 种类型，在分布式虚拟训练系统中使用 Visual C++的 MFC 编程，故重点介绍 GL Studio 生成的代码与 MFC 的混合编程实现多线程网络通信。

1) 建立应用程序，设计面板

使用 VC＋＋的 GLStudio Standalone AppWizard 建立一个 MFC 应用程序，编译运行生成 GL Studio 设计器，接着，在 GL Studio 编译环境中建立主面板对象并添加纹理，建立各种旋钮、转换开关等设备并设置相应属性，然后建立仪表面板对象、制作指针等并添加纹理。

2) GL Studio 代码生成

利用 GL Studio 代码生成器把 GL Studio 设计器创建的文件生成为 C＋＋和 OpenGL 源代码，并添加到分布式虚拟训练系统应用程序中。

3) VC＋＋中多线程网络通信功能完善

在程序的初始化函数 Initialize ()中创建网络通信、数据处理、训练报告生成等次线程，并编写相应的线程执行函数。其中，网络通信线程执行函数采用前述的 TCP 和 UDP 协议混合编程，并应作为全局函数。

在程序的主线程中，为仪表添加必要的变量、响应事件和属性，实现旋钮、转换开关等对仪表指针的控制，如触发网络通信事件，向远程虚拟仪表发送消息等。

4) 编译、连接、运行，生成可执行文件。

3 结术语

在虚拟仪表仿真平台 GL Studio 设计出仪表面板及各种旋钮、开关操作的基础上，结合 VC++，集成开发 Windows Socket 和多线程技术，为虚拟训练系统由单机模式向分布式协同训练模式发展提供可借鉴的技术基础，具有一定工业应用前景。

参考文献：

[1] Distributed Simulation Technology Inc. GL Studio Programmer's Guide Version 2.1[M]. Beijing: Electronic Industry Press, 2002.

[2] 徐勤超, 欧阳中辉, 朱锐. GL Studio 在飞机虚拟地勤训练中的应用[J]. 舰船电子工程, 2008, 28(6): 146-148.

[3] 高颖, 邵亚楠, 郑涛. GL Studio 在飞行座舱模拟器中的仿真研究[J]. 弹箭与制导学报, 2008, 28(1): 257-260.

[4] 袁梅, 赵仲. 虚拟座舱显示系统关键技术研究[J]. 飞机设计, 2007, 27(2): 59-63.

[5] 李彪, 唐金国, 李治庆. 某型导弹指挥仪训练仿真系统虚拟仪表集成方案研究[J]. 工业控制计算机, 2006, 19(11): 77-80.

[6] 谢永成, 魏宁, 李光升. 虚拟仪表在装甲车辆信息化改造中的应用[J]. 电子测量技术. 2008, 31(8): 67-71.

[7] 孙鑫, 余安萍. VC++深入详解[M]. 北京: 电子工业出版社, 2006: 529-532.

[8] 雷振山, 赵晨光. 虚拟仪器系统的网络技术研究与应用[J]. 国外电子测量技术, 2006, 25(5): 59-61.

[9] 王俊鸣, 张智军, 张安旭. 基于 LabWindows／CVl 的多线程技术的电磁兼容预测试系统设计与实现[J]. 弹箭与制导学报, 2008, 28(2): 311-314.

[10] David J. Kruglinskl, Scot Wingo, Goorge Shepherd. Programming Visual C++[M]. Beijing: Electronic Industry Press, 1998: 364-380.