

实时同步协同虚拟实验平台

孔德凤, 韩秀玲, 陈光, 任卓君

(东华大学信息科学与技术学院, 上海 201620)

摘要:针对协同虚拟实验平台中多人协同实验时组员之间无法实时同步实验操作的问题,提出将实验操作全部配置化的设计思想,并且在.NET平台下,基于UDP协议,利用自行设计的连线算法,结合Socket技术、委托技术及C#编程语言,实现了协同实验时同组成员实时同步实验操作的功能。实践结果表明,基于此功能的平台能够让同组成员共同构建实验拓扑,实时同步组员实验操作,查看设备实时的配置信息,从而满足了协同虚拟实验平台的实时同步多人操作的功能需求,学生可以同时操作同一项实验,实时同步实验过程,最终同步完成实验,很大程度上提高了虚拟实验的效率。

关键词:协同虚拟实验;实时同步;操作配置化;连线算法

中图分类号:TP 393.09 **文献标志码:**A

文章编号:1006-7167(2015)02-0117-05



Real-time and Synchronous Collaborative Virtual Experiment Platform

KONG De-feng, HAN Xiu-ling, CHENG Guang, REN Zhuo-jun

(College of Information Sciences and Technology, Donghua University, Shanghai 201620, China)

Abstract: When it comes to the issue that experimental operation can not be real-time synchronization in collaborative virtual experiment platform, the paper proposes the design ideas that make experimental operation configured. The design applies a connectivity algorithm, Socket technology, delegate technology and C# which based on .Net platform, and achieves a real-time synchronization of experimental operation among team members. It turns out that the platform makes joint building of experimental topology and real-time synchronization operation in one team, and meets the functional needs of real-time virtual experiment platform.

Key words: virtual collaborative experiment; real-time synchronization; operation configuration; connectivity algorithms

0 引言

虚拟实验室是一种基于VR虚拟现实技术、Web技术构建的网络化的开放式虚拟教学系统,是解决目

前高校实验资源短缺、经费紧张等问题的有效措施^[1-2]。而建立在虚拟实验及远程协作理论与技术基础上的多人协作虚拟实验室,是人们正在研究的一种新的虚拟实验模式,它能够为用户提供一个共享协作式的教学、学习和实验环境^[3]。

本课题组根据高校“计算机网络”教学实际和本学院的实验教学需求,基于现实教学环境下的通信网络实验室,研发了B/S架构的远程虚拟实验平台,用户只要通过浏览器,如IE, Firefox等,就可以不受时间、空间限制,随时随地进行各种网络实验^[4]。然而该系统只具备单人虚拟实验室,而无允许协同操作的多人实验室,使得设备配置重复率高,实验效率低下,

收稿日期:2014-07-25

基金项目:上海市教育科学2011年度重点项目(A11119)

作者简介:孔德凤(1990-),女,山东临沂人,硕士,研究方向:计算机网络。Tel.:18817842331; E-mail: kdf156@126.com

通信作者:陈光(1957-),男,广东汕头人,教授,硕士生导师,研究方向:无线移动通信,电磁场理论。

Tel.: 13651990193; E-mail: gchen@dhu.edu.cn

且系统不适用于一些大型复杂的实验。

针对这一问题,本文基于自行提出的实验操作配置化的设计思想,利用自行设计的连线算法,结合UDP协议,Socket网络技术、代理技术及加锁法+消息队列的并发控制方法^[5-6]开发了一个多人协同虚拟实验平台,此实验平台具有多人共同操作实验,同时构建实验拓扑,实时同步组员实验操作,查看设备实时配置信息等功能,大大提高了虚拟实验的效率。

1 实时同步协同虚拟实验平台的特点

本实验平台前期开发的“单人在线虚拟组网实验平台”中一个虚拟实验过程的网络拓扑截图^[7]如图1所示。

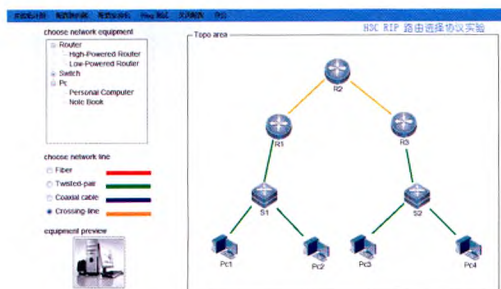


图1 单人虚拟实验过程的一个截图

图1是学生通过浏览器访问网站,进行虚拟实验时的操作界面,实验过程中学生不能多人协作完成实验,只能单人按照实验指导书的要求组建该网络拓扑并进行相应配置,单人配置的设备操作重复率较高,而且不适于大型实验的完成。

实际上,一个网络实验的完成,特别是大型复杂实验的完成需要多个组员的共同协作完成,协同功能实现的重点便是每个组员实验操作的实时同步。所谓“实验操作的实时同步”指的是:任何一个组员的实验操作结果显示到自己平台的同时也会显示到其他组员的平台上,同组成员同时完成一项实验,任何一个组员都可以查看任何一个设备当前实时的配置信息。根据本文提出的方法可以很好地解决这一问题,同组学生通过“多人在线虚拟组网实验平台”可以在一定的范围内实时同步的共同完成一项实验。

2 技术创新点

本文设计的多人协同虚拟实验室是以“计算机网络”课程为背景,利用了Windows Forms的控件技术、C#开发语言、Visual Studio 2008开发工具以及SQL SERVER数据库进行虚拟实验程序设计,并在实验操作配置化的创新思想下,利用前人未在虚拟实验中使用过的Socket网络技术、代理技术结合自行设计的连线算法来实现,具有多人同步构建实验拓扑,实时同步实验操作,查看设备实时配置信息,提高工作效率等优

点。设计完成后将窗体控件生成的动态链接库(.dll)文件嵌入.NET网页中,完成以上操作之后,学生便可通过浏览器轻松地访问网页,进行虚拟实验操作^[7]。

2.1 实验操作配置化

为实现同组成员实验操作及结果的实时同步,提出实验操作配置化的设计思想,即实验中所有的实验操作包括构建实验拓扑都是对于实验设备的配置,实验操作的结果也全部来自于实验设备的配置操作。

实验操作配置化的优势是:与真实“计算机网络”的实验过程和目的相吻合,真实的实验过程中,为了防止实验设备端口的损坏,实验拓扑的构建是通过对设备进行配置而不是通过插拔连接线实现的,此方式既保护了设备,也让学生学习到了有关交换机配置的知识。同时,由于配置信息易于实时传输,为多人同步搭建实验拓扑这一问题的解决提供了基础。

2.2 实时同步配置信息

为了实现多人虚拟实验中实验操作的实时同步,在实验操作配置化的思想下,实时同步配置信息成为了实时同步实验的关键。本文基于UDP协议,使用Socket通讯技术将组员配置的虚拟实验设备信息广播到其他组员对应的虚拟实验设备上,并利用Delegate技术唤醒主窗体平台将接收到配置信息,实时地同步到每个组员的操作平台上,从而使每个组员的虚拟设备上都有了最新的实验配置信息,即实现了配置信息的实时同步。

2.2.1 基于UDP协议的Socket技术

Socket技术是开发局域网通讯工具常使用的技术,在C#中Socket类用于连接管理,实现通信端套接字接口,同时它还定义了绑定、连接网络端点以及传输数据的各种方法,提供处理端点连接传输等细节所需要的功能^[8-11]。

本文使用Socket技术,基于UDP协议,将配置信息当做消息发送给同局域网的其他组员,从而实现配置信息的实时同步,此功能分为发送和接受两部分,具体实现如下:

(1) 发送配置信息。组员在虚拟设备配置框中输入配置信息,便触发了textconf文本框的KeyPress事件,每次键盘输入enter键,代表组员此句配置已经完成,这时便会调用senddata函数,使用Socket网络技术将此句配置发送给本广播组的所有成员;

(2) 接受配置信息。Platform窗体一运行后便启动了Socket网络技术中的监听线程,监听端口是否有发送来的配置信息,若有,便接收,处理后利用Delegate技术添加到操作平台相应的textcheck上,继续监听;若没有,继续监听。

2.2.2 委托(Delegate)技术

委托是一个类,它定义了方法的类型,一旦为委托

分配了方法,委托将与该方法具有完全相同的行为。委托允许传递一个类的方法给另一个类的对象,使得另一个类的对象能够调用这个方法^[12-13]。

本文通过委托将监听子线程接收到的配置信息添加到主线程的配置信息文本框中,使组员可以同步查看所有设备的实时配置信息,以了解到整个实验的进度。部分主要代码如下^[14]:

```
delegate void del(string t); //声明一个委托
private void prif1(string t1)
{
    if (! InvokeRequired) //判断是否在同一线程中
    {
        textBox_check1.AppendText(t1);
    }
    else
    {
        del dele = new del(prif1);
        Invoke(dele, new object[] { t1 }); //执行唤醒操作
    }
}
```

2.3 连线算法

实验拓扑图的构建是实验的基础,目前协同实验平台中实验拓扑的构建是异步的,即组员轮流操作,导致组员被动,实验效率低下,协同的概念也没有完全体现出来。对此,本文利用自行设计的连线算法实时同步同组成员构建的实验拓扑。

多人协同虚拟实验平台的拓扑效果图,如图 2 所示。

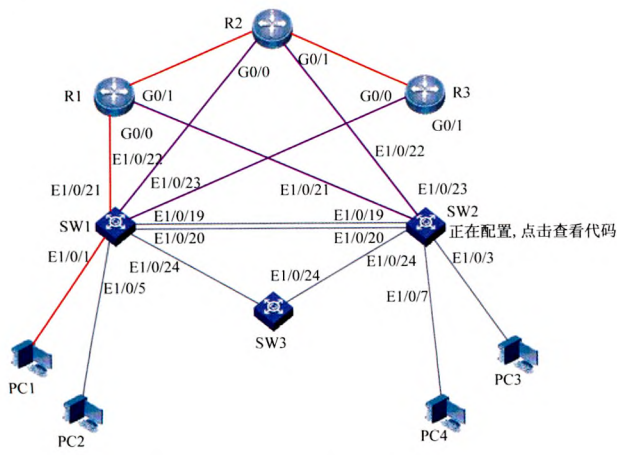


图 2 多人协同虚拟实验拓扑例图

图 2 中,实验拓扑图由本用户和远程端组员合作完成,交换机一(左 SW1)的连线是本用户配置交换机一形成的,交换机二(右 SW2)的连线是远程端组员配置交换机二形成的,两者的结果都可以实时地显示在实验平台上。

2.3.1 连线算法的流程

由于实际实验中用户搭建本实验组的网络拓扑是

通过划分 vlan 来实现的。在本文中将这一操作虚拟化的实现方法是:在用户配置设备划分好 vlan 后,利用实时同步配置信息技术相应的改变本组所有成员的 ConnectPoint 数组值,用 timerliner 计时器根据 ConnectPoint 数组的值实时的刷新连线,以获取最新的实验拓扑,从而实现与实际实验相同的实验操作,进一步提高虚拟实验室的仿真程度。

本文设计的连线算法依据的是划分 vlan 构建网络拓扑的方式,算法基本流程图如图 3 所示。

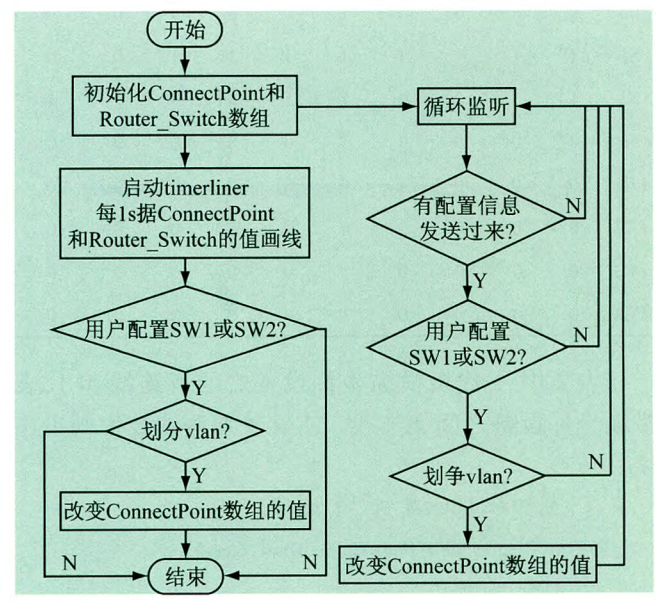


图 3 连线算法基本流程图

图 3 中,实验拓扑的共同构建主要分为两个方面,左侧流程是根据用户自己的操作步骤(包括选取并配置交换机)进行设备连线,右侧流程是根据接受到的远程端组员的交换机配置信息实时的进行设备连线。

2.3.2 连线算法流程的实现

针对图 3 连线算法的基本流程,下面详细的介绍每个阶段的实现技术。

(1) 左侧流程。首先是 Router_Switch 数组和 ConnectPoint 数组初始化,其中 Router_Switch 用来存放网络设备名,ConnectPoint 数组用来存放设备间的连线情况,Router_Switch 数组初始化的值如表 1 所示。

表 1 Router_Switch 数组初始化

	<i>i</i>				
	0	1	2	3	4
<i>j</i>	lr1	lr2	lr3	ts1	ts2
<i>i</i>	5	6	7	8	9
<i>j</i>	ts3	pc1	pc2	pc3	pc4

表 1 中 *i* 代表 Router_Switch 数组的索引值,*j* 代表设备名称 picturebox_ *j* 中的 *j* 变量。

由于线两端连接的是网络设备,所以 ConnectPoint 数组的横纵索引值应为设备名称,本文为了方便实现,

将横纵索引值设为设备名称在 Router_Switch 中对应的索引值。ConnectPoint 数组初始化依据实际实验室网络设备的拓扑,为了清晰表示设备连线关系,表 2 中第一行和第一列的值是 ConnectPoint 数组横纵索引对应于同索引号的 Router_Switch 数组值。

表 2 ConnectPoint 数组初始化

	R1	R2	R3	S1	S2	S3	PC1	PC2	PC3	PC4
R1	0	0	0	1	1	0	0	0	0	0
R2	0	0	0	1	1	0	0	0	0	0
R3	0	0	0	1	1	0	0	0	0	0
S1	1	1	1	1	1	1	1	1	0	0
S2	1	1	1	1	1	1	0	0	1	1
S3	0	0	0	1	1	0	0	0	0	0
PC1	0	0	0	1	0	0	0	0	0	0
PC2	0	0	0	1	0	0	0	0	0	0
PC3	0	0	0	0	1	0	0	0	0	0
PC4	0	0	0	0	1	0	0	0	0	0

表 2 中,1 代表横纵坐标设备之间有连线,0 代表横纵坐标设备之间无连线。本文使用画线技术画出连线。

① 画线的具体实现^[15]为:

```
Pen myPen = new Pen( Color. DimGray, 1 );//
```

② 画笔的颜色

```
Graphics g = this. CreateGraphics();//
```

③ 初始画一幅图

```
g. DrawLine( myPen, Router_Switch[ m ]. Location. X + 20, Router_Switch[ m ]. Location. Y + 20, Router_Switch[ n ]. Location. X + 20, Router_Switch[ n ]. Location. Y + 20 );//
```

根据设备位置画线,在用户开始实验时 timerliner 计时器便开始启动,根据 ConnectPoint 数组和 Router_Switch 数组的值实时的刷新连线,此时用户便有了一个初始化的实验界面。其中 timerliner 的 Interval 属性值设为 1,即每隔 1s 执行一次 tinerliner_Tick 事件。ConnectPoint[m, n] 的值不同时会有不同的颜色粗细的线代表现在设备连接的状态,ConnectPoint[m, n] 值的不同含义如下:① 值为 1 代表设备之间有双绞线,但并没连通;② 值为 2 代表设备之间的连线已经连通;③ 值为 3 代表设备之间存在虚拟的连线,即可以通过两个及两个以上的交换机连通,见图 4。

在整个实验初始化后用户配置 vlan 构建自己的网络拓扑,此时会根据配置信息响应改变 ConnectPoint 数组的值,进而改变了实验设备之间的连线状态。实际的网络设备划分 vlan 必须在端口打开之后进行,因此虚拟实验中使用公共静态数组 undo 来保存需要打开的端口的状态,0 代表未打开不可配置 vlan,1 代表打开可以进行vlan的配置。用户配置设备构建实验

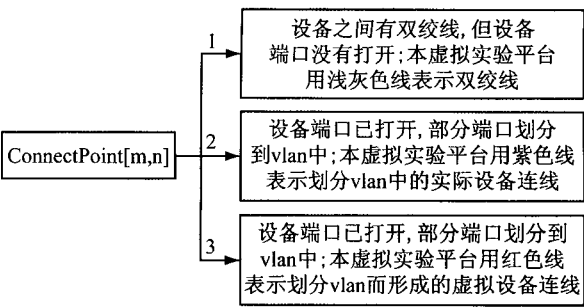


图 4 ConnectPoint[m, n] 值的含义

拓扑的核心代码^[16]如下:

```
string a;
a = ( textBox_conf4. GetLineFromCharIndex( textBox_conf4. SelectionStart ) ). ToString( );
int i = int. Parse( a );
if ( e. KeyChar == 13 )
{
    if ( textBox_conf4. Lines[ i ]. Length < 21 )
    {
        switch ( textBox_conf4. Lines[ i ] )
        {
            ...
            case "[ SW2-vlan3 ] port E1/0/21 to E1/0/22" :
                if ( undo[ 5 ] == 1 && undo[ 6 ] == 1 )
                {
                    textBox_conf5. AppendText( "\n" );
                    textBox_conf5. AppendText( "[ SW2-vlan3 ]" );
                    ConnectPoint[ 0, 1 ] = 3;
                    ConnectPoint[ 4, 0 ] = 2;
                    ConnectPoint[ 4, 1 ] = 2;
                }
            else
            {
                textBox_conf5. AppendText( "\n" );
                textBox_conf5. AppendText( " The port is closed!" );
                textBox_conf5. AppendText( "\n" );
                textBox_conf5. AppendText( "[ SW2-vlan3 ]" );
            }
        }
        e. Handled = true;
        break;
    }
}
```

(2) 右侧流程。在实验开始后平台便利用 Socket 技术获取其他成员的远程配置信息,并根据配置信息以左侧流程相同的逻辑改变接受方的 ConnectPoint 数组值,timerliner 计时器根据 ConnectPoint 数组的值实时刷新连线,进而将发送方构建的实验拓扑实时的同步到接受方的平台上,从而实现多人协同实验的效果。

通过左右两个流程实现同组组员共同构建实验拓扑,并且每个组员都可实时同步到最新的实验拓扑图。

2.4 多人协同实验的实现实例

以上详细的介绍了实时同步协同虚拟实验的实现过程,下面以“华为协同 rip 路由选择协议实验”作为演示实例来验证设计的有效性。

首先用户 A 注册登录,选择“rip 路由选择协议实验”并选择创建 rip 实验组。

用户 B 注册登录,选择加入 rip 实验组,进入实验平台后 A、B 分别选择不同的 SW(交换机)进行实验, A 选择 SW1,B 选择 SW2,然后进行预先计划好的实验拓扑构建,搭建好实验拓扑之后便选择不同路由器并对其进行配置直至完成实验。用户 A 的实验中间过程如图 5 所示。

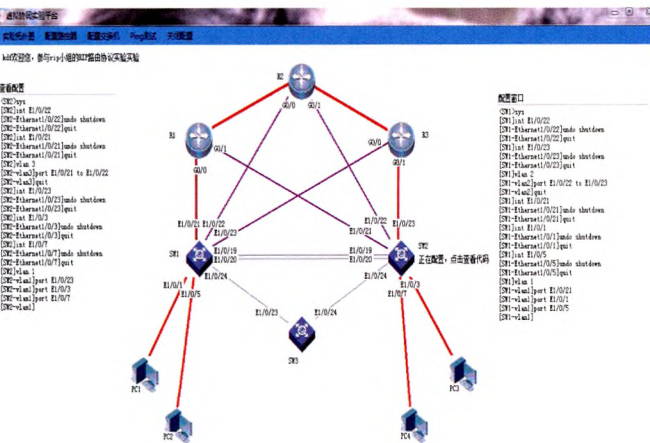


图 5 实验过程图

图 5 中实验拓扑是 A、B 共同搭建的,右侧“配置窗口”中是用户 A 对 SW1 的配置信息,左侧“查看配置”窗口用来显示远程用户的配置信息,默认值是空。由于用户 B 此时正在配置交换机(SW2),所以对于用户 A 来说交换机 SW2 是不可配置的,点击提示信息“正在配置,请点击查看代码”即可在左侧“查看配置”窗口中查看用户 B 的当前配置信息。中间的拓扑图是用户 A、B 共同完成的拓扑图,灰色线代表设备间有双绞线但是端口未打开,紫色线代表设备间有双绞线端口已打开,红色线代表设备间比较直观的虚连接。此结果表明,本协同虚拟实验平台允许多人共同完成一个实验,不仅实现了组员同步构建本组的实验拓扑,且在这个过程中每个组员的平台都具有最新的协同实验状态,每个组员都可以查看设备的实时配置信息,大大提高了实验效率。

3 结 语

本文研究并开发了实时同步协同虚拟实验平台。实例验证,该平台可以满足多人实时同步完成同一实验的需求,提高了大型和复杂实验的工作效率。

参考文献 (References) :

[1] 三维仿真. 虚拟实验室[EB/OL]. (2012-09-18) [2012-09-18]. <http://baike.baidu.com/link?url=v6uyyLcStN0b9tLCOllygdCrFCeJQn9toy1VS8CbJXdLV7OFrwGMS04q7ikrLKSo02AQuK0Hf5DQ-LUN79ANnIK>

[2] 张伟欣. 虚拟计算机网络组网平台的设计及虚拟现实技术的应用[D]. 上海: 东华大学, 2008.

[3] 冯 琦. 基于互联网的协同实验的研究[D]. 西安: 西安电子科技大学, 2006.

[4] 潘 森, 韩秀玲, 陈 光. 在线虚拟网络实验系统命令行提示技术的研究与实现[J]. 计算机应用与软件, 2013, 30(9): 327-331.

[5] 范 志. 实时协同编辑系统中并发控制算法的研究[D]. 兰州: 兰州理工大学, 2009.

[6] 刘新福, 王光彩, 代雯君, 等. 集中式 CSCW 环境中对实时共享对象的并发控制算法[J]. 小型微型计算机系统, 2002, 23(6): 667-672.

[7] 戴维焯, 韩秀玲, 陈 光. 利用动态控件技术实现自由组网拓扑构建[J]. 计算机应用与软件, 2013, 30(7): 119-123.

[8] 马艳芳. 协同虚拟实验的研究与设计[D]. 北京: 北京邮电大学, 2011.

[9] 王继刚, 顾国昌, 徐立峰, 等. 可靠 UDP 数据传输协议的研究与设计[J]. 计算机工程与应用, 2006, 42(15): 34-37.

[10] 黄海芳, 宋筱媛, 许 聪. 基于 UDP 广播的局域网聊天室设计[J]. 福建电脑, 2008, 24(4): 70-74.

[11] Sunev. 基于 C 的 UDP 协议的同步实现[EB/OL]. (2012-08-10) [2012-08-10]. <http://www.cnblogs.com/sunev/archive/2012/08/08/2627247.html>

[12] 王小科. C#开发实战宝典[M]. 北京: 清华大学出版社, 2010.

[13] 我已经飘过. C#委托 [EB/OL]. (2014-05-20). <http://baike.baidu.com/view/1935199.htm?fr=aladdin>

[14] Wei-Meng Lee. C# 2008 Programmer's Reference [M]. New Jersey: Wiley Publishing, Inc, 2009.

[15] Jesse Liberty, Donald Xie. Programming C# 3.0 5th Edition [M]. Massachusetts: O'Reilly Media, Inc, 2009.

[16] 张 青, 郭亚萍. C#编程思想与实践[M]. 北京: 冶金工业出版社, 2002.

· 名人名言 ·

如果你问一个善于溜冰的人怎样获得成功时,他会告诉你:“跌倒了,爬起来。”这就是成功。

——牛顿