

以太网动态仿真实验系统设计

王建军, 朱承学, 赵晋琴, 罗平平

(湖南第一师范学院 信息科学与工程系, 湖南 长沙 410205)

摘要: TCP/IP 协议簇是网络数据通信中使用的主要协议。网络协议仿真系统为教学提供了一个良好的实验平台, 然而目前所用的协议仿真实验系统都是离散的、静态的, 即单独模拟一个协议, 且协议中的数据表示是静态的, 不能随着输入数据的变化而动态变化。为了帮助学生更好地理解以太网中数据通信过程, 有必要开发出一个既能使协议之间相互协作, 又能动态实现数据的变化, 使实验效果逼真实际的仿真实验系统。文中主要描述了以太网动态仿真实验系统混合体系结构的设计和实现, 满足了系统的设计需要, 提高了系统效率。

关键词: 以太网; 网络协议; 仿真; 协作; 动态

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2010)07-0125-04

Design for Ethernet Dynamic Simulation Experiment System

WANG Jian-jun, ZHU Cheng-xue, ZHAO Jin-qin, LUO Ping-ping

(Information Science and Technology Department, Hunan First Normal University, Changsha 410205, China)

Abstract: TCP/IP is the most protocol in network data traffic. A good experiment system is presented for teaching by network protocol simulation system, but network protocol simulation system is discrete and static now, one protocol is simulated singleness. The data can't dynamic variety with input data because the data is static in the protocol. A simulation experiment system must be implemented for the students to understand data traffic in Ethernet, which can collaboration, and dynamic. The design and implementation of the Ethernet hybrid architecture are described in this paper for the dynamic simulation system, meeting the needs of the system, improving the efficiency of the system.

Key words: Ethernet; network protocol; simulation; collaboration; dynamic

0 引言

TCP/IP 协议簇是计算机网络课程的主要教学内容, 也是网络数据通信中使用的主要协议。协议仿真实验教学是帮助学生理解网络协议的重要手段^[1,2]。

目前的协议仿真实验是离散的、静态的, 即单独模拟某个协议, 不能与其它协议合作, 且协议中的数据表示是静态的, 不能随着输入的数据的变化而动态变化, 实验效果单调、生硬。

具有代表性的网络仿真软件有: 国外的 NS2^[3], Boson Netsim^[4], 国内的吉林大学的“网络协议仿真教学系统”^[5], 解放军理工大学的“Psender”^[6]系统等。

NS2 能够演示协议的工作过程, 但是, 无数据内容的动态变化和协议之间的合作。Boson Netsim 主要用于仿真网络工程实验, 帮助学生熟悉网络设备和协议的使用, 不能演示协议的工作流程, 以及协议之间的相互协作过程, 其效果仅类似于一个实际的网络工程实验。

“网络协议仿真教学系统”实现了跨机器之间通信, 但是数据的传递全部由人工操作, 实验环境复杂, 为了更好地模拟协议, 忽略了实验环境与实际应用场景的一致性。

“Psender”系统借用了著作《计算机网络: 自顶向下方法与 Internet 特色》提供的一些在线实验案例, 实现了数据发送方和接收方之间的自动通信和速度匹配, 但是, 忽略了数据通信的中间过程, 如, 不能演示交换机的工作流程, 也是离散的。

因此, 已有的仿真实验是离散的、静态的。而网络通信是一个连续的过程, 需要协议之间相互协作, 而且数据也需要动态变化。

为了真实地模拟网络通信过程, 很有必要开发一

收稿日期: 2009-11-17; 修回日期: 2010-02-24

基金项目: 全国教育科学“十一五”规划 2009 重点课题 (DCA090321); 湖南省教育科学“十一五”规划青年专项 (XJK08QXJ002); 湖南省青年骨干教师培养基金 (2009-43); 湖南省教育厅大学生创新实验 (2008-307)

作者简介: 王建军 (1969-), 男, 湖南衡阳人, 副教授, 博士, 研究方向为计算机网络通信; 朱承学, 副教授, 研究方向为计算机软件; 赵晋琴, 高级工程师, 研究方向为计算机网络。

个系统,既能使协议之间相互协作,又能动态实现数据的变化,使系统能逼真展现出实际的网络通信的运行过程。

以太网动态仿真实验系统使用程序动态模拟以太网中的数据封装、发送、转发和接收过程,并且能够观察数据内容的变化,真实展现数据封装、ARP 表查询和 MAC 地址表查询的工作流程。

软件体系结构是系统开发的基础,是后续工作的前导。文中主要从系统体系结构角度,对系统的设计进行描述,具体包括体系结构设计、层次模块设计、对象设计和数据通信机制等。

1 软件体系结构

在 1992 年, Dewayne Perry 和 Alex Wolf 提出软件体系结构这一术语后,软件体系结构就正式作为一门学科被人们重视^[7]。随着研究的深入,许多专家学者从不同角度和不同侧面对软件体系结构进行了刻画。但是最终目的都是描述软件局部和总体计算部件的构成,以及这些部件之间的相互作用关系。

较为典型描述能实现简化设计和能被反复利用的定义是 2005 年 Kruchten 提出的软件体系结构^[7]。它包含有四个角度,它们从不同方面对系统进行描述:概念角度描述系统的主要构件及它们之间的关系;模块角度包含功能分解与层次结构;运行角度描述了一个系统的动态结构;代码角度描述了各种代码和库函数在开发环境中的组织。

为了满足以太网动态仿真实验系统的需求和简化设计模式,有必要先整体设计一个软件模型,再对各部分分开完成设计。

1.1 几种常用的软件体系结构模型

●管道和过滤器模型 在管道和过滤器类型中,每一个组件(过滤器)有一组输入和输出。一个组件在它的输入端读取数据流,并且在它的输出端产生数据流。连接器对数据流的服务犹如一个管道,它把从一个过滤器中输出的数据流作为另一个过滤器的输入数据流,传送给另一个过滤器,所以又把连接器叫做管道。管道和过滤器系统的优点:使软件具有隐蔽性和高内聚、低耦合的特点;系统容易维护和扩充等^[8]。

●层次模型 层次模型组织成一个层次结构,每一层为上层服务,并作为下层客户。这些层次结构是单独抽象出来的。层次模型的优点:支持抽象程度递增系统;层的重用^[9]。

●面向对象模型 基于数据抽象和面向对象结构是现今比较流行的一种系统体系结构设计方法。数据表示与它们相关联的原始操作被封装为一个抽象数据

类型。这种模型中的组件就是对象。面向对象模型的优点:对象独立;数据隐藏^[9]。

1.2 设计思想和目标

结合前面讨论的几种设计模型的基本思想,把它们灵活应用到以太网动态仿真实验系统体系结构设计中。本系统是管道-过滤器模型和层次模型的综合,在局部设计时采用面向对象模型。设计的目标是为了将来系统的功能扩充、维护和修改、可移植,方便用户理解和使用。为了实现这些目标,确定了如下几个设计原则:

* 对象独立。每个对象负责处理它自己的资源需求,比如数据封装和其它所需要的资源的使用。例如,IP 数据报封装的时候只需要知道该对象的实例和 UDP 数据封装提供的接口,并不需要知道 UDP 报文处理的细节。

* 模块耦合。每个模块处理自己相应的资源,并把结果返回给下一个请求资源的模块。比如转发模块接受到数据报文,检查匹配以后,将数据报文返回给请求的拆分模块。

* 分层设计。该体系表示了单独模块的多层设计,每层的功能构建于下层提供的服务之上,并隐藏下层服务的细节。

* 并行化设计。为了实现系统的高性能,程序需要实现并行化功能。譬如调用 MAC 查询程序的时候同时运行接受用户数据程序,节约系统资源。

* 可扩展设计。考虑到将来系统需求的不断增长,体系结构中的模块或者算法应实现可扩展设计。

2 体系结构设计

2.1 系统整体体系结构

以太网动态仿真实验系统主要目标是设计并实现一个分布交互式系统,在一个局域网内(3 台 PC 组成),各个机器能相互通信,模拟局域网数据通信、封装等流程。整体系统体系结构采取基于管道-过滤器模型,各个结点使用层次模型和对象模型。三种模型结合的方式,可以满足系统设计的要求,使实现简单化。具体设计方案如下。

在一个以太网的模拟工作环境中(具有发送主机,模拟交换机,接收主机),使用程序动态模拟以太网中的数据封装、发送、转发和接收过程,并且能够观察数据内容的变化,真实展现数据封装、ARP 表查询和 MAC 地址表查询的工作流程,使用程序模拟发送主机中数据封装时的数据填充,使用程序模拟发送主机中 ARP 表的查找和学习过程,使用程序模拟交换机中“MAC 地址表”查找和学习过程,帮助理解以太网数据

通信各协议协调工作机制。

拓扑结构如图 1 所示。

其主要工作流程如下:

- * 在 PC1 中,要求程序能动态实现数据封装、ARP 表查询,负责把数据发送给 PC2,并触发交换机(PC2)中的仿真程序运行;

- * PC2 模拟交换机,要求程序能够模拟交换机的数据接收过程、MAC 地址映射和学习过程以及数据发送过程,并触发 PC3 中的仿真程序运行;

- * 在 PC3 中,要求程序实现模拟数据的接收和解封装过程。

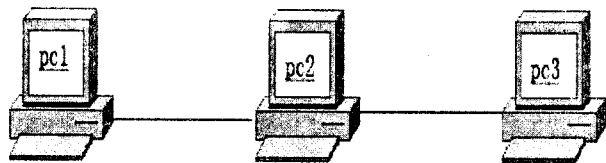


图 1 仿真平台拓扑结构

2.2 系统数据通信机制

20 世纪 80 年代初,美国加州大学伯克利分校的研究人员提出了称为 Socket(套接字)规范的通信标准。简单的说就是通信两方的一种约定,用套接字中的相关函数来完成通信过程^[8]。在 Socket 中,区分不同应用程序进程间的网络通信和连接,主要有 3 个参数:通信的目的 IP 地址、使用的传输层协议(TCP 或 UDP)和使用的端口号。通过这 3 个参数,应用层就可以和传输层通过套接字接口,区分来自不同应用程序进程或网络连接的通信,实现数据传输的并发服务^[8],这样就可以满足进程之间相互通信但又相互不影响的要求。

整个系统中,每台 PC 相当于一个过滤器,信息交互是通过管道传输。每个过滤器负责多个模块的数据筛选,传输服务相当于一个管道将信息进行交互。基于管道-过滤器模型的设计,大量数据对象可以被共享,单个计算机只保存一部分的应用数据,访问对象的存储、处理和通信负载被分布到多个计算机中。且应用由大量运行在独立计算机上的进程组成,进程之间的通信模式完全依赖于应用的需求。进程就相当于过滤器,应用需求就相当于服务管道。它们之间的通信方式可以采取 Socket 套接字的方式进行。这样,实现了数据的及时更新,保证了联机操作的可能性。

本系统主要采用 java 语言编写,在研究服务管道之间的通信机制时,采取 Socket 套接字的方式实现,可以快速实现机器间的信息交流。B、C 机器通过 Socket 与 A 连接。

在 java 语言中,实现 Socket 通信可以采取流操作

模型^[10]。首先,通信双方,“服务器”和“客户”,需要建立 Socket 连接,双方都有各自的 Socket 对象。Socket 对象包含有两个流,一个是输入流,表示流入的数据流,其作用是接受数据;另一个是输出流,表示流出的数据流,其作用是向外发送数据^[7]。结合本系统的特性,实施的方案是过滤器 A 作为服务器,B、C 作为客户机。

建立连接的时候,采用一个通信 B 机器和 C 机器都能控制的变量。首先,A 机器和 C 机器进行一对一的连接,实现数据通信。当且仅当满足条件之后,才能改变变量的值。此时,A 机器与 C 机器才进行通信。采用信号变量,使得 Socket 套接字能实现通信之间的同步控制,满足数据的及时更新。例如,在 java 语言中,可以定义一个私有变量(Private Socket Client)进行同步控制^[10]。

2.3 层次模型

对于单独的一个 PC,采用分层设计思想。在单独的 PC 中,所有的模块是按照层次方式耦合的。高层结构建立在低层提供的功能上。每一个层不是单独存在的,通过层之间的信息交互,实现数据的层层封装。

在过滤器 A 中,实现的是数据的封装。针对 TCP/IP 协议簇的特点,需要实现的是发送模块、UDP 报文封装模块、IP 报文封装模块、帧封装模块(ARP 程序模块)等一系列模块的设计。过滤器 A 中的层次模块设计如图 2 所示。

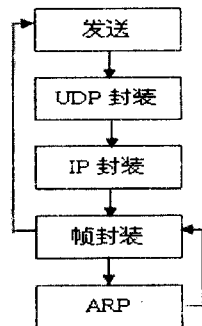


图 2 过滤器 A

在过滤器 B 中,实现的是模拟交换机的功能。该过滤器的主要职能是刷新 MAC 地址表和转发帧数据。图 3 是过滤器 B 的层次模块设计。



图 3 过滤器 B

在过滤器 C 中,实现数据报的接收和解封装(拆除)。该过滤器和 A 的工作流程刚好相反。实现的是接收模块、帧拆除模块、IP 报文拆除模块、UDP 报文拆

除模块、数据还原模块。图 4 是过滤器 C 的层次模块设计。

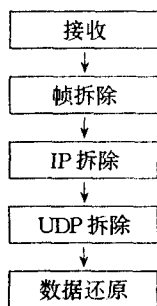


图 4 过滤器 C

2.4 对象模型

在局部设计时,采取面向对象的方法实现某个对象的设计。数据表示和它们相关联的操作被封装为一个抽象数据类型,这些数据和操作被分为公共部分和私有部分两种类型,信息封装可以保证数据的完整性和隐藏性。基于对象模型设计的体系结构可以增强系统的安全性。另外,由于对象之间的相互独立性,基于对象模型设计的系统有利于功能扩充和维护。

基于类(或者对象)之间的调用可以实现层层封装的协调性。在 java 语言中,将处理数据的一些方法定义在某一个类中。在程序的主方法中,如果需要使用该方法的时候,可以通过类(或者对象)来实现方法的调用。

例如,单独 UDP 把报文封装程序定义为 UDP 的一个类,如果在程序主面板中需要显示 UDP 报文的校验和时,只需要调用 UDP.getCRC()即可。

面向对象模型设计中,需要保护的是数据。一是数据的完整性,另一个是数据的保密性。图 5 是针对面向对象模型设计的子系统。

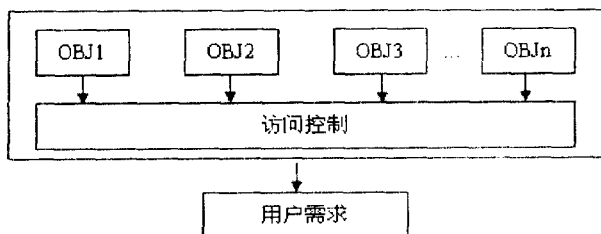


图 5 面向对象子系统

对数据的保护,是通过访问控制和外部接口实现的。例如,ARP 报文程序和 MAC 地址查询程序是通过访问控制来实现其同步的。只有当 MAC 地址查询完之后,ARP 才能得到 MAC 地址。访问控制保证了数据的完整性。用户只能通过接口访问数据。数据流是如何变化、周转的,这些对用户来说都是不可见的。譬如,在 UDP 报文程序中,用户输入数据之后,仅能得

到 UDP 报文数据,而看不到数据报是如何生成的。外部接口保证了数据的保密性。

2.5 综合模型

综上所述,结合三种模型的设计思想,整个系统的体系结构是一个混合模型。在该模型中,包含了三个过滤器,它负责处理所有的信息数据。而每个过滤器由多个对象模块组成。每个模块又由多个对象组成,每个对象拥有完成自己功能的方法和属性。比如,UDP 对象集成了产生一个数据报文所需要的数据和方法。每个对象有自己接口函数,是对象与他方交流信息的唯一通道。

3 结束语

在开发以太网动态仿真实验系统的过程中,系统体系结构设计是关键的一步。采用结合多种设计思想方法,能够缩短系统开发周期,使系统标准化、规范化。同时,系统的设计支持可重用性,也为研发不同的网络仿真平台提供了坚实的技术基础。这种基于过滤器—管道和层次模型结合的方式,满足了系统的动态仿真设计需要,实现了仿真系统的真实模拟,具有现实意义。

参考文献:

- [1] 王建强,吴辰文,李晓军. 基于 NS-2 的 Ad hoc 网络 QoS 仿真与研究[J]. 计算机技术与发展,2009,19(11):183-186.
- [2] 夏云,孙力娟,叶晓国,等. SCTP 协议分析与仿真研究[J]. 计算机技术与发展,2009,19(11):27-30.
- [3] The network simulator NS-2[EB/OL]. 2006-10. <http://www.isi.edu/nsnam/ns>.
- [4] 谢慧,聂峰. 基于 Boson Netsim 的计算机网络仿真实验教学研究[J]. 实验技术与管理,2007,24(24):89-91.
- [5] 蒋一川. 网络协议仿真教学系统的设计与实现[J]. 吉林大学学报,2007,9(9):85-94.
- [6] 陈鸣,常强林,岳振军. 计算机网络实验教程—从原理到实践[M]. 北京:机械工业出版社,2007.
- [7] Kruchten P. The tao of the software architect[EB/OL]. 2005-04. <http://www-128.ibm.com/developerworks/rational/library/4032.html>.
- [8] 套接字—百度百科[EB/OL]. 2008-01. <http://baike.baidu.com/view/538713.html?wtp=tt>.
- [9] Shaw M, Garlan D. 软件体系结构[M]. 北京:清华大学出版社,2007.
- [10] 吕凤翥,马皓. Java 语言程序设计[M]. 北京:清华大学出版社,2006.