

面向服务的网络管理系统设计

徐昆¹, 肖刚², 戴浩²

(1. 中国人民解放军理工大学指挥自动化学院, 南京 210007; 2. 中国电子设备系统工程公司研究所, 北京 100141)

摘要: 为满足网络管理系统扁平化的需求, 提出一种面向服务的综合网络管理系统。设计包含用户服务运行平台、用户门户、服务设计环境和服务管理系统的面向服务网络管理系统集成框架, 引入网络管理接口仿真, 以实现网络管理系统的开发和性能评估, 利用网络管理仿真环境为上层系统提供实时的网络信息。实验结果表明, 该系统能有效完成网络管理任务。

关键词: 面向服务; 接口仿真; 网络拓扑; 管理信息库

Design of Service-oriented Network Management System

XU Kun¹, XIAO Gang², DAI Hao²

(1. Institute of Command Automation, PLA University of Science and Technology, Nanjing 210007, China;

2. Institute of Chinese Electronic Equipment System Engineering Company, Beijing 100141, China)

【Abstract】 In order to adapt the demand of flat network management, this paper presents a Service-oriented Architecture(SOA) network management system. The paper designs service-orient network management system integration framework which consists of customer service operating platform, user portal, service design environment, and service management system, introduces the conception of network management interface simulation in order to achieve network management system development and performance evaluation, and uses the network simulation environment to provide real-time network information for the upper network management system to verify the function of the system and shorten the system development cycle. Experimental results show that the system can efficiently complete the network management tasks.

【Key words】 service-oriented; interface simulation; network topology; Management Information Base(MIB)

DOI: 10.3969/j.issn.1000-3428.2011.24.022

1 概述

随着计算机网络的规模以及复杂程度的不断增长, 传统的层级式管理模式已经不能很好地适应当前的情况。主要问题有: 管理层次过多, 延误问题处理时间; 高层管理系统需要处理大量的网络事件, 难以确定问题的主次顺序; 上层管理系统只掌握宏观抽象的情况, 不能掌握细节情况, 因此无法准确判断问题根源。网络管理扁平化需要有网络管理接口建模能力好、异构管理系统集成能力强、集成复杂度低、重组灵活行好的网络管理体系结构。传统基于 CORBA^[1]的网络集成方法无法满足所有以上要求, 而近年来受到广泛关注和研究的面向服务系统体系结构能够很好地适应。

面向服务体系结构(Service-oriented Architecture, SOA)^[2]可以根据需求, 并且通过网络对松耦合粗粒度的应用组件进行分布式部署、组合和使用。文献[3-5]融合 SOA 思想提出一种面向服务的综合网络管理系统体系结构的原型系统, 重点解决原型系统开发过程中所需网络仿真环境的问题。

由于原型系统开发过程中需要有网络环境与之相配合, 但是单纯为网管开发而组建一个网络环境既不经济也很难实现, 因此需要有一个仿真的网络环境配合系统的开发。但是传统的 OPNET^[6]、GloMosim^[7]、NS2^[8]等网络仿真都是基于离散事件驱动的串行仿真, 是依靠事件来推进, 事件只在可数仿真时间点出现, 因此无法做到实时的为网管系统提供仿真数据的目的。基于以上问题, 本文提出网络管理接口仿真的概念和方法, 为开发面向服务的综合网络管理原形系统提供有效的支撑。

2 面向服务网管体系结构

面向服务的综合网络管理系统集成框架由用户服务运行平台、用户门户、服务设计环境和服务管理系统组成:

(1) 服务运行平台是系统中管理能力服务和用户管理能力服务运行所依赖的分布式软硬件环境, 为网管用户提供服务, 为管理能力服务和用户管理业务设计者提供依据, 响应管理能力服务管理维护者的操作指令。

(2) 用户门户是指面向服务的综合网络管理系统用户侧多样的客户端以及支撑客户端的支撑的软硬件环境, 其向服务设计环境提交业务需求, 并使用服务环境提供的管理能力服务向服务管理系统提供故障申告和业务撤销命令。

(3) 服务设计环境功能包括: 管理能力服务的设计; 用户管理业务的设计。

(4) 服务管理系统为系统维护提供必要的软硬件设施和软硬件管理工具。管理维护环境接收服务设计环境提交的设计方案, 并接收用户门户提出的故障申请和业务撤销命令, 从服务运行平台了解当前服务的运行态势, 向其下达各种指令, 是面向服务的综合网络管理系统中保持管理能力服务能够正常工作的、用户管理业务能够正常使用的必备支撑。

面向服务网络管理系统集成框架如图 1 所示。

作者简介: 徐昆(1982—), 男, 博士研究生, 主研方向: 网络管理, 信息系统; 肖刚, 高级工程师、博士; 戴浩, 研究员、中国工程院院士

收稿日期: 2011-07-12 **E-mail:** xukunpost@163.com

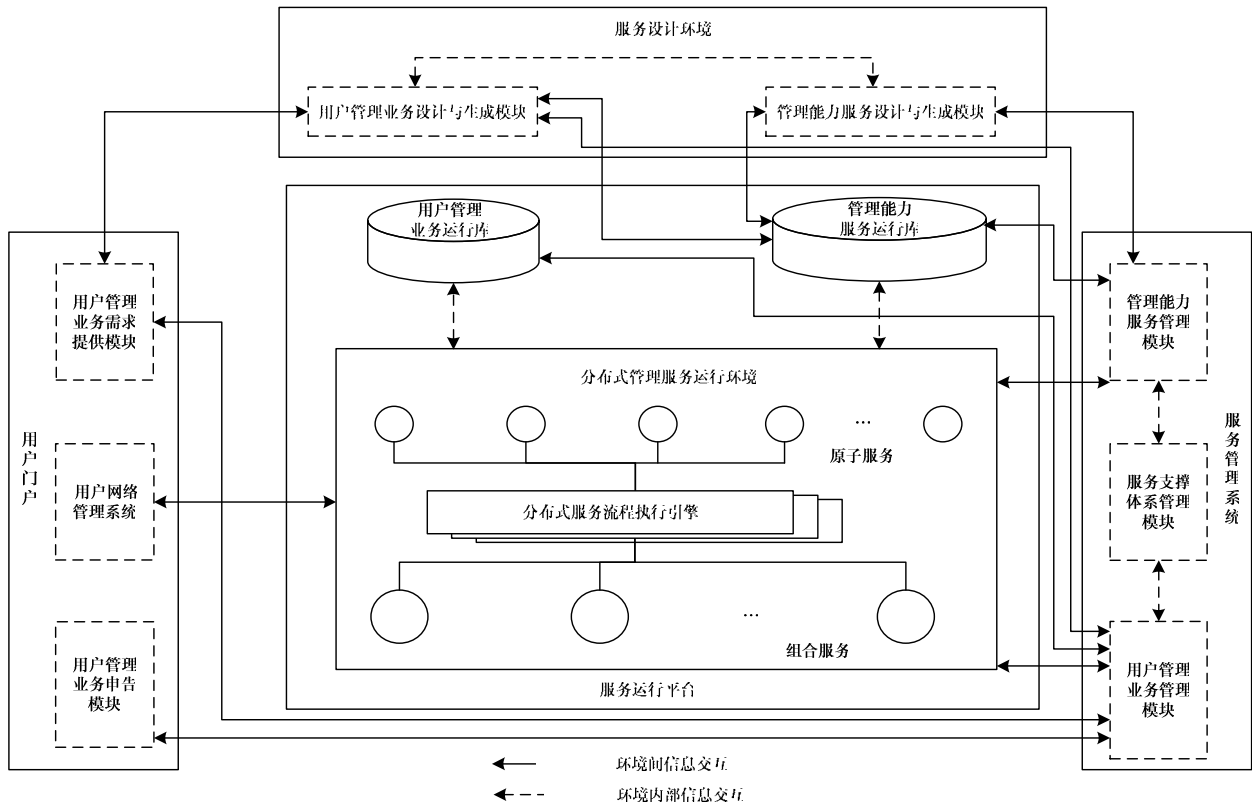


图1 面向服务网络管理系统集成框架

3 网络管理接口仿真系统设计

3.1 网络管理接口

网络管理接口仿真通过计算机仿真的形式在一台或几台计算机上，将一个虚拟的网络接口状态呈现给上层的网络管理系统。对于一个实际网络，由于无法预知网络拓扑、网络设备状态，因此无法准确地评估网管系统的性能。通过网管接口仿真，可以预先对仿真网络拓扑、状态和行为进行配置，通过比较预配置信息与实际管理信息之间的差距，实现对网管系统验证的功能。

3.2 网络管理接口仿真与网络仿真的比较

从概念上讲，网络仿真是一种利用数学建模和统计分析的方法模拟网络行为，从而获取网络设计和优化所需要的网络性能数据地仿真技术。网络管理接口仿真的重点是，向上层网管系统呈现一个与实际网络一致的网络管理接口，通过网络管理接口仿真，有效地为网络管理系统的开发、性能评估提供高效的手段。从仿真运行上讲，网络仿真在实现上通常是基于离散事件驱动的串行仿真，其依靠事件来推进，且该事件只在可数仿真时间点出现，按事件出现的时间执行与事件相关的操作。以最常见排队系统为例，将系统中出现的时间按时间顺序存放于时间列表：设 t_i 为第 i 个用户到达时间(事件 A_i)； c_i 为第 i 个用户离去时间(事件 D_i)； e_i 为事件间隔时间，仿真时间为 $\sum e_i$ 。基于事件推进仿真如图2所示。

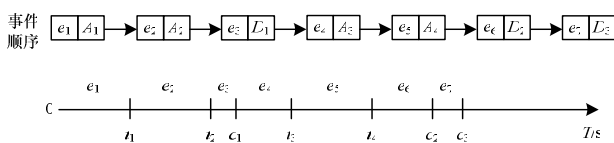


图2 基于事件推进仿真

事件驱动的仿真实现步骤如下：

(1)将事件插入列表时按序插入。

(2)若事件发生时间 $\sum_{i-1} e_j \leq t \leq \sum_i e_j$ ，那么时间间隔 $e_i = t - \sum_{i-1} e_j$ ， $e_{i+1} = \sum_i e_j - t$ 。

(3)一个事件的处理可能导致另一个事件的出现，如果有新事件产生按照产生时间插入到队列中。

(4)每处理一个事件，仿真时钟推进 e_i 。

可以看到，基于事件驱动的仿真时钟是依靠处理事件来推进的，仿真时钟的快慢取决于事件 i 中的间隔时间 e_i 的大小，而仿真执行的快慢取决于事件队列的长短与事件的复杂程度，因此仿真时钟与实际时钟通常不同步。

而网络管理系统需要的是一个能够实时地为网管系统提供接口信息的虚拟网络，其要求仿真时间与系统时间保持同步，而基于事件驱动的网络仿真无法做到实时的为网管系统提供仿真数据，为保证实时地向网管系统提供数据，网络管理接口仿真应该采用基于时间驱动的仿真，按固定时长考察系统状态及行为，并分析记录相关结果。

基于时间推进仿真如图3所示，在时间 t_i 处理事件 A_i ，另外要求仿真时间 t 与时间 T 保持一致就可以保证数据的仿真实时性。

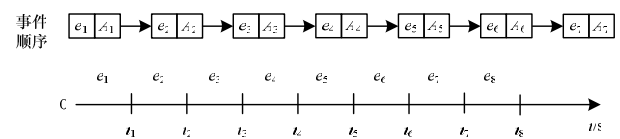


图3 基于时间推进仿真

3.3 网络管理接口仿真系统结构

网络管理接口仿真系统分为虚拟设备、虚拟网络、网络管理系统3个层次。虚拟的网络设备包括设备所支持的管理信息库(Management Information Base, MIB)、虚拟代理、设备信

息、相关数据库等部分。数据库中存储了 MIB 库中各个节点的值, 在仿真时用来给虚拟的代理提供数据; 虚拟代理所完成的工作与真实的网络设备中的代理基本一致, 主要是采集设备信息, 并与管理端进行通信, 将设备状态信息告知管理端, 不同的是真实设备中的代理是直接从中采集信息, 而虚拟代理是在其相应数据库中采集信息。网管仿真系统结构如图 4 所示。

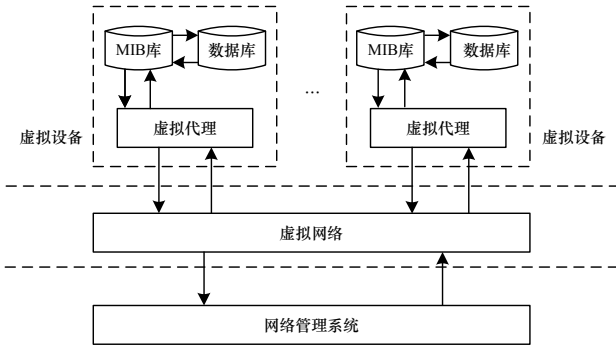


图 4 网管仿真系统结构

由图 4 可知, 虚拟网络由多个虚拟设备组成, 各个设备之间不存在物理上的连接, 只需要通过逻辑上的连接就可以将各个设备组成一个网络, 通常是采用在虚拟设备数据库中配置各个设备之间的连接关系, 以及路由表等信息, 来组织虚拟网络的。最上层的网络管理系统完成对网管接口内容的显示, 主要是用来对仿真网络进行验证、分析。

4 网络管理仿真方法

网管接口仿真流程由仿真系统搭建、仿真数据生成、仿真脚本编写、仿真结果验证 4 个部分组成。

4.1 系统搭建

在网络管理接口仿真系统中, 仿真工具采用的是 MIMIC Simulator Suite(MSS)^[9], 其在每个虚拟设备都有一个独立的 Agent, 可以对虚拟设备的 IP 地址作任意的配置, 对于预编译大量的 MIB 库, 可以根据需要配置不同设备的 Agent。支持 TCL(Tool Command Language)脚本语言, 通过运行脚本语言能够实现动态的对设备状态、参数、数据库进行修改, 从而达到仿真网络行为的目的。

4.2 仿真数据生成

仿真数据生成是网管接口仿真最为重要的部分, 根据仿真执行过程中数据是否发生变化, 将仿真数据分为静态数据和动态数据 2 大类。静态数据是指在仿真过程中数据值不发生改变或很少发生改变的数据; 动态数据是指在仿真过程中需要发生变化的数据。

4.2.1 静态数据生成

静态数据生成通过采集的方法来生成, 即分为未知 MIB 库的编译、采集参数的确定、采集脚本的处理 3 个主要部分。其中, 对采集脚本的处理需要解决一些网络设备会把 Internet 控制报文协议(Internet Control Message Protocol, ICMP)报文作为一种网络攻击手段而进行屏蔽, 在生成采集脚本的时候无法生成这些设备的脚本, 但是这些设备是能够支持简单网络管理协议(Simple Network Management Protocol, SNMP), 所以需要根据实际情况修改采集脚本, 以确保对全部网络设备数据的正确采集。

4.2.2 动态数据生成

相对静态数据来说, 动态数据的所占比例小但是仿真数据生成中的重点。动态仿真数据生成流程如图 5 所示, 首先

抽样采集一定长度数据, 分析其统计特性判断是否具有解析解, 如果存在解析解, 则按照其分布的概率密度函数生成动态的仿真数据; 如果不存在, 则根据仿真的需要实际取样采集数据。然后编写仿真脚本加载动态仿真数据。

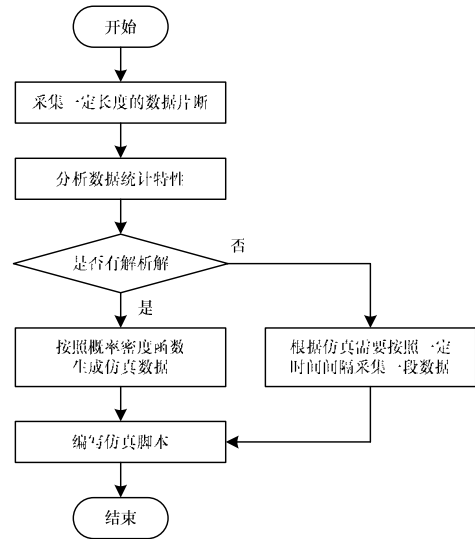


图 5 动态仿真数据生成流程

4.3 仿真参数配置

仿真参数是指对各个仿真设备的基本参数的配置, 主要包括 IP 地址、团体名、设备类型、端口号、PDU、时延、丢包率等参数。其中, 到设备的往返时延通过多次测量, 求其均值的方法来确定, 为简化模型, 认为路由是对称的, 因此将平均往返时延的一半作为到设备的端到端时延:

$$dealy = \frac{\sum_{x=1}^{x=n} averagedelay(x)}{2n}$$

其中, x 表示第 n 次采集的值。

丢包率通过计算重传次数占总传输报文数的百分比来确定, 具体如下:

$$errorrate = \frac{retransmits}{transmits + retransmits}$$

平均丢包率为:

$$averageerrorrate = \frac{\sum_{x=1}^{x=n} errorrate(x)}{n}$$

其中, x 表示第 n 次采集的值。

4.4 仿真的执行与验证

网络管理接口仿真是按照时间推进的, 由于仿真时间与实际时间严格同步, 因此要求仿真平台具有更强的处理能力。对仿真系统的验证分为存在性验证和有效性验证。存在性验证是指检查各个仿真的设备是否正确配置参数以及仿真数据是否加载; 有效性验证是指采用测试工具测试仿真数据的正确性。

5 原型系统设计与仿真

5.1 面向服务的网络管理原型系统设计

面向服务的网络管理原型系统以网络拓扑管理业务设计作为实例。使用 Eclipse3.5 和 Jgraph 组件开发管理能力设计系统, 系统采用 ISO(International Standard Organized)制定的 Z 语言定义描述管理服务的原语集, 并对原语集进行图形化表示, 实现图形拖拽的方式设计管理服务。管理维护系统采用 Web 服务分布式管理(Web Services for Distributed Management, WSDM)技术实现对管理服务的信息管理、运行监控、

策略和安全管理。在管理服务运行平台中,企业服务总线采用 ESB4.2 实现,流程引擎采用 JBPM4.3,服务运行容器和用户门户采用 Tomcat5.0 实现。

网络管理用户首先通过用户门户向系统提交网络管理需求;系统管理员根据用户需求以及当前系统已有能力,将用户需求转化为服务文档提交给管理维护系统;管理维护系统负责当前系统已有服务和业务的调度;管理服务运行平台接受管理维护系统的调度,通过流程引擎运行设计好的业务,并通过用户门户向网管用户提供服务接口,实现用户的管理需求。面向服务的网络管理原型系统运行流程如图 6 所示。

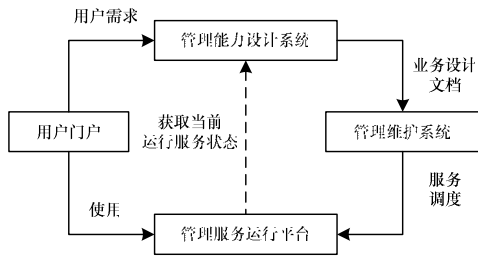


图 6 面向服务的网络管理原型系统运行流程

网络拓扑管理业务所依赖的管理能力包括拓扑搜索、拓扑呈现、拓扑融合。原型系统的预设场景为 2 个运营商各自管理自己的自治域拓扑(自治域 1、自治域 2)、2 个自治域物理上相互链接,但互相无法获取对方的拓扑管理信息。由于业务需求网管用户需要同时对 2 个网络的拓扑同时进行管理,传统的网络管理系统需要重新对 2 个自治域的拓扑管理系统进行修改,然后重新部署软件,才能实现对 2 个域的同时管理,由于需要重新开发和部署软件,因此周期会较长。联合拓扑管理业务流程如图 7 所示。

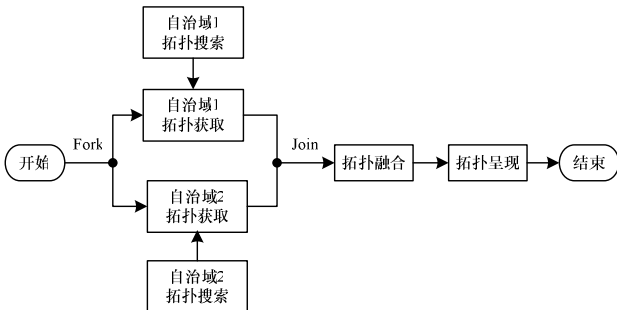


图 7 联合拓扑管理业务流程

由图 7 可知,在本文原型系统中,2 个自治域通过拓扑搜索和拓扑呈现 2 个服务管理各自域的拓扑管理,同样无法获取对方的拓扑管理信息。然而当需要同时管理时,不需要对 2 个域的网管系统进行任何改造,只需要设计一个新的业务,增加一个拓扑融合服务,就能够实现对原有 2 个自治域的拓扑管理,同时只需要开发一个拓扑融合服务并部署,由于系统具有完善的部署运行环境,因此部署基本时间可以忽略,只需要开发一个服务即可,总的开发周期比传统网管系统要短得多,并且不影响原有的服务运行。

5.2 系统仿真验证

仿真环境是为原型系统提供一个虚拟的测试网络。将虚拟网络分为 2 个自治域:自治域 1 的 IP 字段为 194.0.0.1~194.255.255.255;自治域 2 的 IP 字段区间为 198.0.0.1~

198.255.255.255。通过虚拟多台设备的代理对其 IP 进行分配,并对路由表进行配置,使这些设备形成 2 个相互连接的自治域。

面向服务的网络管理原型系统,通过面向服务的网络拓扑管理原型系统对 2 个自治域进行综合管理,采用拓扑融合算法^[10]实现对已有拓扑的拼接管理,系统的用户接口如图 8 所示。



图 8 面向服务的网络管理原型系统用户接口

6 结束语

本文提出面向服务的网络管理系统结构。网络仿真环境能够为上层网管系统提供实时的网络信息,能较好支撑原型系统开发,缩短系统开发的周期,并为系统结果的呈现提供一种便捷、高效的方式。今后将进一步完善仿真数据,重点在各个仿真设备数据的关联性方面开展研究,为网管系统的开发提供更真实可信的虚拟环境。

参考文献

- [1] 宋光磊,张 玥. 基于 CORBA 技术的网管系统集成[J]. 计算机工程, 2002, 28(3): 151-153.
- [2] OASIS. Reference Model for Service Oriented Architectures(Draft 1.0)[EB/OL]. (2010-11-21). <http://www.oasis-open.org/committees/download.php/16587/wd-soa-rm-cd1ED.pdf>.
- [3] Paul A. Service Orientation: Writing Strategies and Best Practice[M]. [S. l.]: Cambridge University Press, 2006.
- [4] Nicolai M J. SOA 实践指南[M]. 程 桦,译. 北京: 电子工业出版社, 2008.
- [5] 王卫星,王晨光. 基于 SOA 的企业信息系统集成框架[J]. 计算机工程, 2010, 36(18): 29-31.
- [6] 王文博,张金文. OPNET Modeler 与网络仿真[M]. 北京: 人民邮电出版社, 2003.
- [7] 高振国. GloMoSim 网络仿真——从入门到精通[M]. 哈尔滨: 哈尔滨工业大学出版社, 2007.
- [8] 徐雷鸣,庞 博,赵 耀. NS 与网络模拟[M]. 北京: 人民邮电出版社, 2003.
- [9] Gambit Communications. MIMIC Overview: Online Web Documentation[EB/OL] (2010-11-21). <http://www.gambitcomm.com/products/mainmmic.htm>.
- [10] 全云鹏,肖 刚. 子网拓扑融合技术研究[J]. 计算机应用, 2009, 29(s2): 423-427.

编辑 刘 冰