

OPNET Modeler

仿真建模大解密

高
端
编
辑

OPNET Modeler

理论与实践兼备，基本概念与实际操作俱全

全面系统地梳理Modeler的相关知识，讲解由浅入深，内容翔实透彻

富于实用性和启发性的复合型精编实例，是各阶段使用者不可多得的参考资料



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

OPNET Modeler

仿真建模大解密

高 嵩 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

OPNET Modeler 是一款网络仿真软件,主要用于网络设计,能够满足大型的、复杂的网络仿真需要,帮助设计人员设计和分析网络、网络设备和通信协议。本书着重介绍 OPNET 产品系列的技术核心——网络建模,以及相应软件 Modeler 的基本概念与使用技巧,即从仿真理论基础开始,逐步展开介绍 Modeler 建模的基本概念,分级梳理 Modeler 的实用技巧,探讨常用模型资源的使用,并在其中穿插讲解相关的技术内幕,最后给出实用性与启发性兼备的复合型精编实例。本书讲解由浅入深,内容详实透彻,是各阶段使用者不可多得的参考资料。

本书可供使用 OPNET 进行网络仿真、网络协议研究开发、网络规划、网络性能优化等工作的技术人员阅读参考,也可作为高等院校通信、网络及相关专业的研究生和高年级本科生的学习参考用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

OPNET Modeler 仿真建模大解密/高嵩编著. —北京:电子工业出版社, 2010.9

ISBN 978-7-121-11766-4

I. ①O… II. ①高… III. ①计算机网络—计算机仿真—应用软件, OPNET Modeler IV. ①TP393.01

中国版本图书馆 CIP 数据核字(2010)第 173724 号

责任编辑: 窦 昊 特约编辑: 牛雪峰

印 刷: 北京京师印务有限公司

装 订:

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 27 字数: 691 千字

印 次: 2010 年 9 月第 1 次印刷

印 数: 4 000 册 定价: 59.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

通信网络的数字化已经进行了二十多年,国内也进行了十几年。在这个过程中,通信网络正在逐步计算机化,其结构和功能越来越复杂,技术更新也越来越快,使得对通信网络的研究和开发也因此变得越来越困难,这就要求革新传统的开发方式,需要研发方式将完全基于实物转变为尽量采用仿真手段辅助。

仿真作为一种越来越重要的科学方法,除了可以用于验证设计方案、测试产品、减少研发成本外,对生产流程的科学化也有很大的帮助。然而,通信网络仿真是一种涵盖计算机和通信两个领域的技术,需要结合两个领域的知识才能运用得当。

OPNET 自问世以来,其排名就一直位居网络仿真软件的前列,现在 OPNET 已经具有魔术般令人眩目的功能,但由于其精巧复杂的结构,使用起来让人如坠泥沼,因而学习和掌握 OPNET 并不是能在短期内能完成的。其复杂程度可以用“令人筋疲力尽”来形容,为此,很多研究人员都迫切需要一本对 OPNET 相关知识和使用技巧进行全面、综合介绍的书籍。

为此,作者编写了《OPNET Modeler 仿真建模大解密》,本书力求帮助读者在循序渐进地了解背景知识和软件操作的同时,达到知其然又知其所以然的境界,其中,第 1 章对 OPNET 及其同类产品的特点做了简要的介绍;第 2 章引入了系统仿真与网络仿真的基础知识,并从理论层面上分析了 Modeler 的建模方法;第 3 章对 Modeler 建模概念进行了全面的描述,包括网络、节点、进程域的基本概念;建模中常用的模型衍生、继承与 C++ 中概念的区别;以及运行仿真的相关知识,模型文件变成可执行程序的流程,与一般 C/C++ 代码编译过程的关系;第 4 章对建模概念对应的操作进行了系统的梳理,包括各种属性和数据结构的定义、各域的建模操作、仿真执行的各种途径和可选参数,并在适当的地方深入介绍了它们的执行效果及其原理;第 5 章介绍了一些特殊的建模技巧,这些技巧够对提高仿真结果的可信度、可比性提供帮助;第 6 章介绍了常用资源的使用技巧,包括应用、应用需求、IP、IP QoS、IP 多播、MANET 等标准模块;OPNET 模型支持库 (OMS);以及一些内建的数据结构和算法,这些资源的熟练使用无疑会帮助读者提高仿真建模的效率和能力;第 7 章介绍了使用 VC6、VS.net、CDB、ODB 等手段调试仿真程序的方法,通过动画提供可视调试信息的办法;第 8 章介绍了编程接口的使用,通过这些接口,用户可以对 Modeler 进行二次开发,实现自动生成仿真模型、定制界面等目标;第 9 章介绍了作者精心编写的 6 个实例,包括高级编程技巧的跳频电台建模、界面定制的 ETS 应用实例、利用 Modeler 为后台仿真引擎的 ESA 应用实例、Modeler 与 Matlab 联合仿真的实例、可复用的无线动画类实例,这些实例在着重表现一个主题的同时,还展示了 3~8 种技巧,实际上相当于 20~30 个实例。

本书是作者多年研究经验的总结,相信能给读者一定的启发和帮助,但由于作者水平有限,错漏之处在所难免,恳请读者对书中的缺点和错误给予批评指正。可以通过 E-mail: opnet_uncover@tom.com 与作者取得联系。

高 嵩

目 录

第 1 章	OPNET 及同类产品简介	(1)
1.1	OPNET 公司及其产品	(1)
1.2	Berkley 大学的 NS-2	(2)
1.3	UCLA 大学的 GloMoSim	(3)
1.4	QualNet	(4)
第 2 章	系统仿真及网络仿真基础	(5)
2.1	系统仿真	(5)
2.2	仿真系统的本质	(6)
2.3	模型与建模	(6)
2.4	网络仿真建模方法	(7)
2.4.1	事件调度	(8)
2.4.2	活动扫描	(9)
2.4.3	进程交互	(10)
2.5	网络仿真系统软件运行的驱动力	(12)
2.6	随机数及随机数种子	(13)
第 3 章	Modeler 建模基本概念	(15)
3.1	概念的起点	(15)
3.2	共性概念	(16)
3.2.1	对象与模型	(16)
3.2.2	属性	(17)
3.2.3	属性接口	(19)
3.2.4	统计量	(22)
3.2.5	数据包	(23)
3.2.6	ICI	(23)
3.2.7	分布与随机变量	(24)
3.3	网络域概念	(24)
3.3.1	子网	(24)
3.3.2	通信节点	(25)
3.3.3	通信链路	(26)
3.3.4	探针	(30)
3.3.5	坐标系统与移动性	(34)
3.3.6	轨迹	(36)
3.3.7	卫星轨道	(37)
3.3.8	业务	(38)

3.3.9	失效与恢复	(39)
3.4	节点域概念	(40)
3.4.1	处理器模块	(40)
3.4.2	队列模块	(41)
3.4.3	外部系统模块	(42)
3.4.4	发信机模块	(42)
3.4.5	收信机模块	(43)
3.4.6	天线模块	(43)
3.4.7	包流	(45)
3.4.8	统计线	(46)
3.4.9	逻辑关联	(46)
3.4.10	接口定义	(47)
3.5	进程域概念	(48)
3.5.1	Proto-C	(48)
3.5.2	动态进程	(54)
3.5.3	接口定义	(59)
3.5.4	事件	(60)
3.6	外部系统域概念	(64)
3.7	模型的衍生与继承	(64)
3.8	Modeler 运行概念	(66)
3.8.1	工程的组成	(66)
3.8.2	仿真运行	(70)
3.8.3	仿真程序与模型文件	(70)
3.8.4	进程模型与节点模型	(73)
3.8.5	偏好	(74)
第4章	基本建模技巧	(77)
4.1	共性操作	(77)
4.1.1	模型文件的新建、打开及保存	(77)
4.1.2	模型的派生	(78)
4.1.3	模型属性的定义	(79)
4.1.4	局部统计量的定义与探针的使用	(83)
4.1.5	数据包格式定义	(85)
4.1.6	ICI 格式定义	(86)
4.1.7	声明外部文件引用	(87)
4.1.8	填写自述	(87)
4.1.9	随机变量的定义	(88)
4.2	定制网络模型	(98)
4.2.1	创建空白网络模型	(98)
4.2.2	修改度量单位和边界地图	(100)
4.2.3	使用对象拼盘	(102)

4.2.4	搭建拓扑	(106)
4.2.5	定义路径	(120)
4.2.6	定制链路模型	(126)
4.3	定制节点模型	(133)
4.3.1	使用设备创建器自动生成节点模型	(133)
4.3.2	手动搭建(定义固定、移动或卫星节点, 使用 Queue)	(136)
4.4	定制进程模型	(153)
4.4.1	描述状态跳转图	(154)
4.4.2	状态变量定义	(156)
4.4.3	临时变量定义	(158)
4.4.4	函数代码块的使用	(159)
4.4.5	诊断块的使用	(160)
4.4.6	终止块的使用	(161)
4.4.7	头部块的使用	(162)
4.4.8	接口定义	(163)
4.4.9	全局统计量定义与探针使用	(164)
4.4.10	全局属性定义	(165)
4.4.11	声明子进程	(165)
4.4.12	基本编程技巧	(166)
4.5	定制外部系统定义模型	(173)
4.6	想定的编辑与管理	(174)
4.6.1	创建想定	(174)
4.6.2	编辑、复制与删除想定	(177)
4.6.3	编辑仿真集对象	(177)
4.6.4	想定成分导入与导出	(180)
4.7	运行仿真	(181)
4.7.1	GUI 操作基本步骤	(181)
4.7.2	命令行操作基本步骤	(184)
4.7.3	两套操作的共同点	(184)
4.8	统计结果的显示与分析	(185)
4.8.1	向量结果的显示与分析	(185)
4.8.2	标量结果的显示与分析	(188)
4.9	配置 Modeler——偏好的设定	(189)
4.9.1	通用的配置方法	(189)
4.9.2	目录的设置	(191)
4.9.3	颜色的设置	(194)
4.9.4	图标库设置	(195)
4.9.5	编译和链接参数设置	(196)
4.9.6	管道阶段相关设置	(198)

第5章 特殊建模技巧	(200)
5.1 使用地形数据	(200)
5.2 无线区域定义	(200)
5.3 定义失效与恢复	(203)
5.3.1 静态设置	(203)
5.3.2 动态设置	(204)
第6章 高效建模技巧	(207)
6.1 使用标准模型库	(207)
6.1.1 应用层模型	(208)
6.1.2 应用需求	(228)
6.1.3 RPG 模型	(229)
6.1.4 IP 模型	(232)
6.1.5 IP QoS 模型	(263)
6.1.6 IP 多播模型	(265)
6.1.7 MANET 模型	(267)
6.2 使用内建的数据结构和算法	(280)
6.2.1 拓扑图	(281)
6.2.2 Dijkstra 算法	(281)
6.2.3 哈希表	(281)
6.2.4 IP 地址	(282)
6.2.5 队列	(282)
6.2.6 映射	(282)
6.2.7 字符串使用操作	(283)
6.2.8 向量	(283)
6.3 使用 OMS 库	(283)
6.3.1 自动编址	(284)
6.3.2 进程注册	(286)
6.3.3 Buffer 管理	(288)
6.3.4 全局数据定义	(291)
6.3.5 概率分布函数库	(292)
第7章 调试与动画	(294)
7.1 不同编译器所需配置	(295)
7.1.1 VC6.0 环境变量设置	(295)
7.1.2 VS.net 相关环境变量	(296)
7.1.3 R6034 错误	(297)
7.2 编译错误信息	(297)
7.3 异常提示与警告信息	(299)
7.4 使用 ODB	(300)
7.4.1 GUI 界面	(301)
7.4.2 控制台命令	(304)

7.5	C/C++调试器	(314)
7.5.1	VC6.0	(315)
7.5.2	VS.net 调试	(316)
7.6	CDB	(318)
7.6.1	附着操作	(318)
7.6.2	断点设置	(319)
7.6.3	执行控制	(320)
7.6.4	变量查看	(320)
7.8	动画的运用	(320)
7.8.1	自动动画	(321)
7.8.2	自定义动画	(323)
第 8 章	Modeler 的编程接口	(330)
8.1	模型的外部访问 (EMA)	(330)
8.1.1	EMA 层次结构	(331)
8.1.2	<ACV>结构	(332)
8.1.3	EMA 函数	(334)
8.1.4	EMA 程序结构与流程	(335)
8.1.5	EMA 应用编译和使用	(347)
8.2	ETS 与界面控制	(347)
8.2.1	界面配置信息的组织结构	(348)
8.2.2	编辑器配置文件结构	(348)
8.2.3	操作列表文件结构	(349)
8.2.4	对话框文件	(350)
8.2.5	共享库的制作和使用	(351)
第 9 章	综合应用实例	(353)
9.1	跳频电台建模	(353)
9.1.1	实例的示范作用	(353)
9.1.2	实例介绍	(353)
9.1.3	实例编制	(354)
9.1.4	实例运行效果	(361)
9.2	ETS 应用实例——无线可达性标注	(362)
9.2.1	实例的示范作用	(362)
9.2.2	实例介绍	(362)
9.2.3	实例编制	(362)
9.2.4	实例运行效果	(364)
9.3	ESA 应用实例——与外部仿真器融合	(364)
9.3.1	实例的示范作用	(364)
9.3.2	实例介绍	(365)
9.3.3	实例编制	(365)
9.3.4	实例运行效果	(372)

9.4	与 Matlab 联合仿真	(373)
9.4.1	实例的示范作用	(373)
9.4.2	实例介绍	(373)
9.4.3	实例编制	(374)
9.4.4	实例运行效果	(375)
9.5	TCP/IP 协议栈二次开发实例——嫁接 MAC 协议	(376)
9.5.1	实例的示范作用	(376)
9.5.2	实例介绍	(376)
9.5.3	实例编制	(376)
9.5.4	实例运行效果	(382)
9.6	无线动画类	(382)
9.6.1	实例的示范作用	(382)
9.6.2	实例介绍	(382)
9.6.3	实例编制	(383)
9.6.4	实际运行效果	(388)
附录 A	ip_rte_v4.h 的声明修改	(390)
附录 B	ip_dispatch 进程模型的改动	(391)
附录 C	manet_mgr 模型的代码改动	(393)
附录 D	ip_higher_layer_proto_reg_sup.h 中的代码改动	(394)
附录 E	ip_rte_support.ex.c 中的代码改动	(395)
附录 F	Radio.cpp 文件内容	(397)
附录 G	FH_Phy 进程模型函数块代码	(401)
附录 H	结合 Matlab 仿真实例管道过程代码内容	(406)
附录 I	Wireless_topo 类代码	(413)
附录 J	Node 类代码	(418)
附录 K	Anim.h 文件内容	(421)
	参考文献	(422)

第1章 OPNET 及同类产品简介

1.1 OPNET 公司及其产品

OPNET(OPTimized Network Engineering Tool)最早是 MIL-3 公司的核心软件产品。MIL-3 公司成立于 1986 年,创立者是三位麻省理工大学的教授,成立的目的是为美国军方开发网络及其应用的决策支持软件。与军方的合作成功后, MIL-3 公司逐渐建立了与民用领域的 Cisco、HP 等网络设备巨头的合作伙伴关系,为它们提供各自的设备模型、协议模型及技术支持服务,使 MIL-3 公司逐渐成为世界上、领先的网络仿真、建模、决策支持软件公司。后来以主营产品命名,称为 OPNET 公司, MIL-3 于 2000 年在纳斯达克上市(代码 9Q2358, 简称为 OPNT)。

现在, OPNET 公司的总部位于马里兰州贝瑟斯塔,大约拥有 550 名员工,持有 Cisco 的全球发布协议,是以提供网络管理和应用软件及相关服务为主营业务的著名软件企业。公司的业务主要是性能管理、网络操作、容量管理以及网络研发,其产品以能支持在当前和未来的操作条件下的网络分析、应用分析以及服务器性能分析而著称。公司的软件产品包括进行仿真和网络技术及通信协议分析所需的模型库;服务和支持,包括提供顾问服务、软件许可升级、技术支持和服务以及为没有维护协议的用户提供培训,这些培训包括指导客户使用其产品的课程。2007 年 10 月 19 日, OPNET 公司用 1 000 万美元完成了对另一家软件公司 Network Physics 的收购。后者提供的应用流监控软件 NetSensory 比 OPNET 的 ACE 具有更丰富的功能,收购后成为 OPNET 产品系列中的 ACE Live。

OPNET 网络仿真软件作为 OPNET 公司的核心系列软件产品,包括:

① ACE Analyst (ACE)。提供管理应用性能的高级分析技术。通常用于定位并解决产品应用中的问题,并保证应用在发布的同时能保证质量。ACE Live 能执行用户体验的监控,帮助分析方法发现应用性能问题的根源。

② IT Guru Network Planner。提供可预测的网络容量规划和设计优化,也可提供网络配置更改的验证。

③ IT Guru Systems Planner。提供服务器的容量规划,包括从物理服务器到虚拟服务器的迁移规划。

④ IT Netcop。为企业提供集中、实时可见的网拓扑、业务和状态监视,它提供了一个综合视图,用以识别网络事件的影响,并帮助定位并解决出现的问题。

⑤ IT Sentinel。提供网络配置的完整性和安全性的自动连续审核,以及后应式的变更验证。

⑥ OPNET Panorama。提供管理应用性能的系统分析技术。和 ACE Analyst 类似,它也用于整个应用的生命周期,以保证应用发布并定位产品应用中的性能问题。

⑦ SLA Commander。提供对 Web 应用的响应时间的主动监视，以完善 ACE Live 的被动监视。

⑧ VNE Server includes。VNE Server includes 包括一批从网络设备获取拓扑、业务和其他信息的适配器，以及一批第三方的数据源。VNE Server 将从其他 OPNET 产品中收集数据的过程自动化了，这些产品包括 IT Guru Network Planner 以及 SP Guru Network Planner。VNE Server 的能力包含在 IT Sentinel 和 SP Sentinel 中。

⑨ SP Guru Network Planner。SP Guru Network Planner 是在 IT Guru Network Planner 基础上开发的，包含对服务提供商（SP）很有价值的分析方法，可用于规划、网络优化以及验证配置更改。

⑩ SP Guru Transport Planner。SP Guru Transport Planner 是一个用于光网络的网络规划产品，目标客户是网络设备生产商。

⑪ SP Netcop。SP Netcop 用一个综合视图提供集中、实时可见的网拓扑、业务和状态监视。

⑫ SP Sentinel。SP Sentinel 为服务提供商（SP）提供网络配置的完整性和安全性的自动连续审核，以及后应式的变更验证。

⑬ OPNET Modeler。OPNET Modeler 是一个网络建模和仿真产品，它使用户能够评估网络设备、通信技术、系统和协议在仿真设定的网络条件下将如何表现。公司还开发和销售了一批为应用和网络管理软件提供附加功能的软件模块。

⑭ 模型库。模型库被 OPNET 软件用于仿真和分析网络技术及通信协议，这些库是产生网络模型的基石。网络模型由构成实际网络的设备、计算机、链路对应的软件对象组成，这些对象的活动由设备、计算机、应用、通信协议、链路的模型控制。IT Guru Network Planner、IT Guru Systems Planner、IT Netcop、IT Sentinel、SP Guru Network Planner、SP Netcop、SP Sentinel 和 OPNET Modeler 都包含了网络技术和通信协议的库，如 TCP/IP、HTTP、OSPF、ATM、帧中继、IP-QoS、802.11 或 Wi-Fi、802.15.4 或 ZigBee、802.16 或 WiMAX，这些模型库被包含在基本产品中，其他的库则需要额外的费用。

目前全球有 1 400 多个组织，包括美国军方和许多著名的电信公司都在使用 OPNET 软件。OPNET 产品系列功能很强大，但能完整地拥有该产品的中国客户并不多，用得好的就更少了。如果能切实发挥各产品的强大功能，综合运用到网络规划、设备开发、协议制定等工程和科研领域，一定会给使用者带来不可估量的好处。

OPNET 是当前网络仿真领域最著名的主流产品，是目前世界上最先进的网络仿真开发和应用平台，近几年被第三方权威机构评选为“世界级网络仿真软件”第一名^[3]，但是市场上还是出现了几个强有力的竞争对手。

1.2 Berkley 大学的 NS-2

NS-2 即 NS 版本 2，是知名的、源代码公开的、免费的软件模拟平台。NS 起源于 1989 年的 REAL 网络模拟器。NS 的开发在 1955 年获得了 DARPA 的支持，通过 VINT 项目，由 LBL、Xerox、PARC、UCB 和 USC/ISI 合作进行。目前 NS 由 DARPA 的 SAMAN 项目和 NSF 的 CONSER 项目支持。NS 具有开放的结构和良好的可扩充性，已经从其研究者那里吸

收了非常丰富的模块,包括从 UCB Daedalus 和 CMU Monarch 计划,以及从 SUN 微系统公司获得的无线代码。

REAL 模拟器是为了研究分组交换数据网络中的数量控制和拥塞控制方案的动态性而设计的,它提供给用户一种方法来描述这些网络并观察它们的行为。REAL 模拟器是用 C 语言编写的,它提供源代码以便感兴趣的用户可以根据它们自己的需要修改模拟器。

在此基础上, Lawrence Berkeley National Laboratory 的网络研究组开发了 NS 的第一个版本。NS-1 继承了 REAL 模拟器的工作,包括几种风格的 TCP (SACK、Tahoe、Reno) 和路由器调度算法等,并形成了一个可扩展的事件驱动模拟引擎。NS-1 所使用的模拟描述语言是工具命令语言 Tcl 的扩展。一个模拟由一个 Tcl 程序来描述,通过这样一个通用语言, NS 具有了很强大的模拟配置描述能力。

在不断改进 NS-1 的基础上, UC Berkeley 进行了三项重大改进后,发布了 NS-2。

- NS-2 重新定义了对象结构,对 NS-1 中复杂的结构作了简化和分解;
- 使用 MIT 的面向对象 Tcl (Otccl) 代替了 Tcl 作为模拟器配置的接口;
- 将 Otccl 解释器的接口代码和主模拟器分离^[4]。

NS-2 已经具备了相当强大的功能,也有众多的使用者,但它有以下的缺点:

- NS-2 的网络拓扑定义、配置业务源和汇聚点定义、统计信息收集、执行等重要的仿真设计活动全部是通过命令完成的,缺乏直观性;
- NS-2 的模型设计采用面向对象的方式进行,在复用性提高的同时增加了结构的复杂性,很多父类的行为必须通过阅读源代码才能获悉;
- NS-2 不是一个商业软件,其开发和投入有限,到目前为止还远未达到完善的程度,软件的 bug 还在不断地出现和更正,用户必须核实软件的 bug 对模拟结果是否造成影响;
- 众多的组织和个人参与了其构件库的扩展开发,在增加功能的同时,也带来了新的 bug;
- NS-2 的很多辅助工具并不是同一公司开发的,格式上不是很统一,说明手册也缺乏条理性和一致性。

1.3 UCLA 大学的 GloMoSim

GloMoSim 是由美国加州大学洛杉矶分校 (UCLA) 计算机系的 R. Bagrodia 教授等人开发的网络仿真平台项目。它最早是为 DARPA 的全球移动信息系统 (Global Mobile Information Systems, GloMo) 提供仿真服务的,所以不是开源项目,但允许在教育科研领域免费使用。在 2003 年该项目已终止开发,其源代码成为商业产品 QualNet 的基础。

GloMo 项目旨在为商业及军事应用开发、论证、转化无线移动通信技术,所以 GloMoSim 主要针对无线网络仿真,对有线网络仅提供很有限的支持。它基于 UCLA 计算机系研发的 PARSEC 语言/编译器构建离散事件仿真引擎,不同操作系统平台上的 PARSEC 都可以编译 GloMoSim,具有较强的跨平台能力。

GloMoSim 用 PARSEC 语言或 C 语言实现网络协议,用脚本进行配置。GloMoSim 仿真的主配置文件还可以引用别的配置文件完成业务配置、移动性配置等工作。

1.4 QualNet

QualNet 是 GloMoSim 的商业化延续, 其生产商是 Scalable Network Technologies 公司, 简称为 SNT (公司主页为 <http://www.scalable-networks.com>)。SNT 创建于 1998 年, 创始人是 UCLA 的 Rajive Bagrodia 教授和曾任美国空军 (USAF) 少将的 George Norwood。SNT 公司主要以大规模异构网络的管理、开发、仿真软件的销售和咨询业务为主。SNT 的客户包括美国政府部门、军方承包商以及许多世界 500 强公司, 主要产品 QualNet 于 2004 年 5 月通过美国国防部和商务部的批准, 对中国市场开放。

QualNet 的主要特点是:

- 采用标准 C 语言编译, 同时也支持 C++, 用户易于修改、调用、仿真自己的协议;
- 基于已经过验证的 PARSEC 并行仿真内核, 每个节点都可独立进行运算, 可以对节点达到上万个的网络进行仿真;
- 按照网络的七层架构采用模块化设计, 有利于用户直接选择想仿真的协议模块, 各个层之间采用标准的 API 接口;
- 优秀的算法设计, 仿真速度快, 并行仿真时的加速比高于其他的仿真器;
- 动画效果强大, 可查看单个功能的动画效果, 能做到仿真与动画同步;
- 仿真协议的移植性很强, 仿真协议与真实设备上的协议相似, 只需要进行简单的修改就可以下载到设备中去使用, 并且与 CPU 无关;
- 支持半实物仿真, 可以作为真实网络的一部分, 参与到网络的测试中。

但 QualNet 为了提高仿真运行速度, 对仿真细节做了很多省略, 所以更倾向于模拟, 而非严格意义的仿真。

第2章 系统仿真及网络仿真基础

2.1 系统仿真

系统是按照某些规律结合起来的互相作用、互相依存的所有实体的集合或总和。一个系统可以定义为如下集合结构：

$$S = (T, X, \Omega, Q, Y, \delta, \lambda)$$

- T 为时间基，描述系统变化的时间坐标， T 为整数则成为离散时间系统， T 为实数则成为连续时间系统。
- X 为输入集，代表外部环境对系统的作用。通常 X 被定义为 R_n ，其中 $n \in I^+$ ，即 X 代表 n 个实值的输入变量。
- Ω 为输入段集，用于描述某个时间间隔内的输入模式，是 (X, T) 的一个子集。
- Q 为内部状态集，是系统内部结构建模的核心。
- δ 为状态转移函数，定义系统内部状态是如何变化的，它是一个映射，即 $\delta: Q \times \Omega \rightarrow Q$ ，其含义是：若系统在 t_0 时刻处于状态 q ，并施加一个输入段 $\omega: \langle t_0, t_1 \rangle \rightarrow X$ ，则 $\delta(q, \omega)$ 表示系统在 t_1 时所处状态。
- Y 为输出集，系统通过它作用于环境。
- λ 为输出函数，它是映射 $\lambda: Q \times Y \times T \rightarrow Y$ 。输出函数给出了一个输出段集。

这是系统模型的一般描述。人们在长期地研究与应用中，还将模型形式加以分类，如表 2-1 所示^[1]。

表 2-1 模型分类

模型描述变量的轨迹	模型的时间	模型形式	变量范围	
			连续	离散
空间连续变化 空间不连续变化	连续时间	偏微分方程	√	
		常微分方程	√	
离散变换	离散时间	差分方程	√	√
		有限状态机		√
		马尔可夫链		√
	连续时间	活动扫描	√	√
		事件调度	√	√
		进程交互	√	√

系统仿真的定义主要有以下几个。

- G.W.Morgenthater 认为：仿真是指在实际并不存在的情况下对于系统活动本质的实现；
- Korn 认为：用能代表所研究的系统的模型进行实验；
- Spriet 认为：所有支持模型建立与模型分析的活动均为仿真活动；
- Or  n 认为：仿真是基于模型的活动，该定义被认为是现代仿真技术的一个重要概念。

实际上，随着科学技术的进步，特别是信息技术的迅速发展，仿真的技术含义不断地得到发展和完善，但无论哪种定义，仿真基于模型这一基本观点都是共同的。

现代仿真技术均是在计算机的支持下进行的，因此，系统仿真也成为计算机仿真。系统仿真的三个基本活动是系统建模、仿真建模和仿真试验，这三个活动将系统、模型、计算机（包括硬件和软件）三要素紧密地联系在一起了。

2.2 仿真系统的本质

仿真系统是指用于研究真实系统而建立的一整套模型，根据系统的形式化定义，可以把仿真系统看做一个代数系统，称为 M 代数，即

$$S_M = (A, \delta, \lambda)$$

式中， $A = X \cup T \cup Q \cup Y$ ， A 一定是一个非空集合。在通常情况下，由于 Ω 是无限集，因而 A 是无限集，因此该代数系统是个无限系统。

在构造仿真系统 S_{Sim} 时必须建立一个与之近似系统 $S_{\text{Sim}} = (A', \delta', \lambda')$ ，其中 $A' = X' \cup T' \cup Q' \cup Y'$ ，同时需要保证存在一个映射满足 $f: A \rightarrow A'$ 是近似双射。

对于任何 $x \in X$ ， $t_0, t_1 \in T$ ， $q \in Q$ ， $y \in Y$ ，都有

$$f(\delta(t_0, t_1, x, q)) = \delta'(f(t_0), f(t_1), f(x), f(q))$$

$$f(\lambda(t_0, t_1, y, q)) = \lambda'(f(t_0), f(t_1), f(y), f(q))$$

可见，仿真系统与真实系统之间存在同构关系，仿真系统的本质是构造同构映射，只有这样，对仿真系统的研究成果才能被推导到原系统中。

因此，仿真建模的核心就成为了寻找同构映射 f 。不同的构造方法就是不同的 f ， f 的不同之处在于对系统的七大主要元素有着不同的处理原则。

2.3 模型与建模

一般认为，模型是指用特定方式对客观事物（或实体）及其联系的描述，是实体的抽象。美国国防部对模型的定义是，对一个系统、实体、现象或过程的物理、数学或其他逻辑表示。我国大百科全书出版社出版的《大百科全书》就把模型定义为按照科学研究的目的，用物质形式或思维形式对原型客体本质关系的再现。通过对模型的研究可获得关于原型客体的知识。

总结这些观点，我们可以看到，模型的用途是重现、解释、研究规律、分析预测，甚至交流沟通，所以它必须具备四个要素：描述语言、内容、推演规则、构建模型的目的。从这个角度来看，模型就不局限于科学研究的领域了，例如小说也可以称为模型，因为小说以表

达作者的观点为构造目的，以自然语言为描述语言，以故事情节为内容，以日常逻辑、历史背景、生活常识等为推演规则。

另外，一个已有模型也是可以用来构建更复杂的新模型，这时前者就成为后者语言的一部分，此时两个模型的语言和推演规则必须相同。比如，小说中用到的成语、典故、习语、谚语也是一种模型，这种模型的构建目的是为了生动简捷、形象鲜明地表达一个完整的意思，这个意思就是它的内容，而其语言、推理规则与小说是相同的。

当然，更经典的模型是数学模型、物理模型、UML 模型等，下面对上述几种模型进行简单的比较，如表 2-2 所示。

表 2-2 模型比较

	实 例	描述语言	推演规则	内 容	构建目的
自然语言模型	小说	词汇、文字 (自然语言)	语法、修辞（一般逻辑 和生活常识）	情节、主题	表达主观观点或描述客观情况
	成语			一个完整的含义	生动简洁、形象鲜明地描述
数学模型	万有引力定律	数学公式	数学规则	引力的数学描述	解释引力的规律
物理模型	用电容电感振荡回路模拟弹簧振子	实物、物理现象	器件性能+器件参数+器件相互作用关系	弹簧振子阻尼振荡规律	用电子器件模拟机械规律
UML 模型	用 UML 描述的软件模型	UML	UML 元素与高级语言代码间的映射关系+高级语言语法	软件的逻辑结构和实现方式	在开发小组成员间达成共识、了解全貌、发现问题

所谓建模，就是为重现、解释、研究规律、分析预测，甚至交流沟通而建立模型的行为，一般通过组合运用已有的模型来构建，完全全新的模型是不可理解的，比如希腊文字在发现……碑之前，一直都未能破解，知道……发现……之后才得以破解，而玛雅文字的破解则到现在还是个难题。

为了研究、分析、设计和实现一个系统，需要进行试验。可以直接在真实的系统上进行，也可以先构造模型，通过对模型的试验来代替或部分代替对真实系统的试验。传统上大多采用第一种方法，随着科学技术的发展，尽管第一种方法在某些情况下仍然是必不可少的，但第二种方法日益成为人们更为常用的方法，主要原因在于：

- 系统还处于设计阶段，真实的系统尚未建立，人们需要更准确地了解未来系统的性能，这只能通过对模型的试验来进行；
- 在真实的系统上进行试验可能会破坏系统或发生故障，例如，对于一个处于运行状态的化工或电力系统进行没有把握的试验将会冒巨大的风险；
- 需要进行多次试验时，难以保证每次试验的条件相同，因而无法准确地判断试验结果的优劣，另外，试验时间太长或费用昂贵。

2.4 网络仿真建模方法

通信网络系统简称为网络，也是一种系统，系统的变量通常是状态。这种系统的状态变化可能发生在连续时间轴上的任意一点，因此属于连续时间系统。但这种系统的状态变化只发生在一个离散的集合内，所以该系统的变量变化轨迹是离散的，变量变化范围也是离散