基于组件的软件开发方法在 C^3I 系统仿真中的应用研究

舒 振, 张耀鸿, 罗雪山
(国防科学技术大学 管理科学与工程系, 湖南 长沙 410073)

摘 要: 介绍了组件化的软件开发方法的有关概念,分析了它的特征,比较了它与面向对象技术的关系,并讨论了将这种组件化的方法运用到 C^3I 系统仿真中,同时建立了一个基于组件的 C^3I 系统仿真环境。

关键词: 组件; 软件开发; C^3I 系统; 仿真环境
中图法分类号: TP311.52 **文献标识码**: A **文章编号**: 1001-3695(2004)10-0182-03

Application and Research of C^3I System Simulation Using the Method of Software Development Based on Component

SHU Zhen, ZHANG Yao-hong, LUO Xue-shan
(Dept. of Management Science & Engineering, National University of Defense Technology, Changsha Hunan 410073, China)

Abstract: First this paper introduce the concepts and analyze the characters of the method of software development based on component, compare this method with object oriented programming method, then discuss using this method into C^3I system simulation, and develop a component based C^3I system simulation environment.

Key words: Component; Software Development; C^3I System; Simulation Environment

C^3I 系统是一类庞大复杂的系统,包含多种技术因素和不确定性。对于一个大型的 C^3I 系统,究竟如何确定它的需求,如何协调并确定合适的作战技术指标,如何缩短研制周期,如何评估多种可选的系统方案、测试和评价性能指标,如何评估它的作战效能和平时使用效益以及改进已建系统,找出瓶颈,使系统不断进化,这些都是 C^3I 系统设计、开发和建设过程中必须面对的问题,已成为 C^3I 系统建设中广为关注的焦点。

仿真技术为这些问题的解决提供了一条可行的途径。在设计、开发 C^3I 系统之前,建立一个仿真系统,对于研究 C^3I 系统的动态特性,解决系统中的一些关键问题,揭示各种方案的内在能力和薄弱环节,并根据一定的原则和指标选择最优方案,不仅是可行的,也是必要的。现实问题的需求牵引,包括多媒体技术和灵境技术在内的当代技术的推动,都把 C^3I 系统仿真提到极其重要的地位。

但是随着被仿真系统的日益复杂性,仿真应用开发的工作量也成倍增加,完全从头开发一个仿真系统的工作效率越来越低,模型与仿真结果的可靠性与准确性也更难以得到保证。在这样的情况下,面向对象的思想已经难以适应这种分布式软件模型,于是组件化设计思想得到了迅速的发展。通过组件化的设计方式,将已有的各种仿真应用组装起来建立大型仿真系统,是克服复杂大型系统仿真面临困难的有效途径。仿真重用对于提高仿真应用开发的效率与质量,有着非常重要的意义。

1 基于组件的软件开发方法概述

1.1 组件的基本概念

组件不是一个新概念,在许多相对成熟的工程领域中,组件的概念是十分广泛的。但是由于组件自身固有的特性,目前人们对组件这一概念还没有一个统一的定义。下面是关于组件的一些具有代表性的观点:

(1) 组件是软件的基本量子。它具有一定的功能,可插用,同时又是可维护的。

(2) 组件是软件开发中一个可替换的软件单元,它封装了设计决策,并作为一个大单元的一部分与其他组件组合起来。

(3) 组件是具有特定功能的、能够跨越进程的边界,实现网络、语言、应用程序、开发工具和操作系统"即插即用"的独立的对象。

(4) 组件是指任何可被分离出来、具有标准化的、可重用的公共接口的软件。

1.2 基本组件开发的概念与特征[3,4]

基于组件的开发是从第一代(二层)和第二代(三层)的客户/服务器发展起来的,它能改变软件开发过程中的被动局面,可以使通过组件组装成应用程序的梦想成为现实,这将使软件产业发生革命性变化。

基于组件的开发是一种利用可重用的软件组件构建应用程序的技术。这些组件由三部分组成。与结构化软件开发技术相比,基于组件的开发有一些显著的特征:

(1) 真正的软件重用和高度的互操作性;

(2) 接口的可靠性;

(3) 可扩充服务;

(4) 具有强有力的基础设施;

(5) 具有构建和胶合组件的工具。

1.3 组件开发与面向对象的关系[4]

面向对象是在传统开发方法基础上发展而来,具有许多很

收稿日期: 2003-06-25; 修返日期: 2003-11-19

好的特性, 如信息隐藏使得调用者不需要了解实现细节, 而且使得对一个对象内部的修改不会影响其他对象。系统的可维护性很好, 易于修改。但是面向对象技术还有许多不足, 其中最显著的一点是一个对象若要与另一个对象交互, 必须了解另一个对象的标志。而且若一个对象改变了标志, 所有显式调用该对象的其他对象都要改变。面向对象的技术在使用对象时, 会有数据不一致的副作用, 例如 A 使用对象 B, C 也使用 B, C 作用于 B 的结果导致影响 A, 反之亦然。

基于组件的设计是在面向对象的软件开发方法的基础上逐渐发展起来的一种新技术。对于基于组件的设计和面向对象技术之间的关系有以下几种观点:

(1) 基于组件的设计抽象了许多面向象技术的实现概念, 是将面向对象技术应用于系统设计级上的一种自然的延伸。

(2) 基于组件的设计是面向对象技术的一个简单化的版本, 它注重封装性, 但忽略了继承性和多态性。

(3) 基于组件的设计是构造系统的体系结构级的方法, 并且组件可以采用面向对象的方法很方便地实现组件。

这三种观点各有所长, 都不全面, 但对我们理解基于组件的设计和面向对象技术之间的关系却很有意义。基于组件的设计和面向对象技术之间的关系如表 1 所示。

表 1　基于组件的设计和面向对象技术之间的关系表

	基于组件的设计	面向对象技术
目标	应用程序间的更大范围的组件的重用	应用程序内的更大行为的重用
基本关系	请求/响应的关系	行为的继承关系
重点	组件接口不变	可扩展的类的层次关系
最终目的	应用程序具有定义良好的接口, 可以很容易地替换其中的组件	应用程序通过对象、类的层次关系的扩展来改变行为

2　基于组件的 C^3I 系统仿真环境

2.1　基于组件的开发在 C^3I 系统仿真中的应用

C^3I 系统具有指挥层次多、频域范围广、系统复杂等特点, 对 C^3I 系统的建模和仿真也是一项极为复杂的系统工程。因此在 C^3I 系统建模仿真过程中, 应尽可能地利用现有的研究成果, 将已有的各种仿真模型组装起来建立仿真系统, 是克服复杂大型系统仿真困难的有效途径。

基于组件的开发就是将软件开发看成一个组装过程, 其对传统的软件开发产生了重大的影响, 大大提高了软件的开发效率, 于是我们考虑将组件化的开发方法引入到 C^3I 系统的建模仿真中。

基于组件的 C^3I 系统建模仿真方法可以看成是一个以组装为主的开发过程, 即按照 C^3I 系统领域确定系统所需的可复用仿真模型组件, 根据仿真系统的特殊要求对构架和组件进行特化, 并开发系统的专用组件, 然后将领域构架、领域组件和系统专用组件组装成一个完整的系统。其中, 仿真领域分析通过收集 C^3I 系统仿真领域的相关领域知识、领域专家经验和现有系统的技术资料, 产生领域构架和对该领域可复用仿真组件的需求; 仿真组件开发实现可复用的仿真模型组件的开发; 仿真组件测试对开发的仿真模型组件进行功能测试; 仿真组件提交根据一定的检验标准和管理制度, 把可复用组件提交到仿真模型组件库; 仿真系统开发从具体仿真应用的问题域和用户需求出发, 开发专用组件, 与领域构架和组件一起进行系统组装, 最终产生仿真系统。

基于组件的 C^3I 系统建模仿真能够提供许多优势, 这是其他建模仿真所没有的:

(1) 减少复杂性。开发仿真系统将集中在一个或多个小的组件上面, 从而减少了问题域, 降低了管理风险。

(2) 为仿真系统并行开发提供了机会。工程边界定义了稳定的组件定义, 从而支持仿真系统并行开发。

(3) 提高了灵活性。组件满足一定的需求, 能够重用, 支持新的仿真系统开发。

(4) 减少仿真系统测试。组件方便了单元测试, 支持过程测试。

鉴于基于组件的 C^3I 系统建模仿真方法的种种好处以及实际工作的需要, 我们开发建立了基于组件的 C^3I 系统建模仿真环境, 在环境中建立了 C^3I 系统仿真组件模型规范。该规范定义了仿真模型和仿真环境的接口协议, 使满足协议的模型能够无缝地嵌入到仿真环境中。只要模型接口保持不变, 模型可以方便地升级和修改, 实现了仿真模型的 "即插即用", 大大提高了 C^3I 系统的开发效率。

2.2　仿真环境基本结构

仿真环境提供一种图形化的编辑环境, 将基本的模型组合成一个仿真工程项目。仿真环境可对模型和仿真任务进行配置, 对仿真过程进行控制和管理。仿真环境的基本结构如图 1 所示。

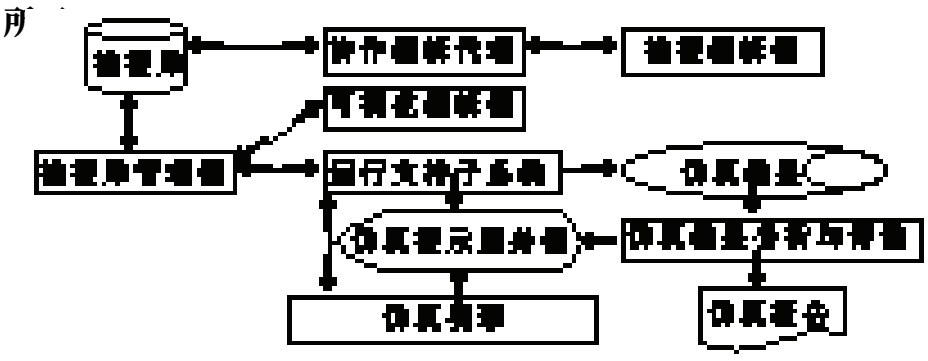

图 1　仿真环境的结构

仿真环境由以下部分组成:

(1) 模型编辑器;

(2) 可视化编辑器;

(3) 模型库管理器;

(4) 仿真实验管理器;

(5) 仿真引擎;

(6) 仿真运行支持子系统。

模型编辑器用于建立仿真系统的描述模型。模型可以采用 Petri 模型、排队模型等。模型的开发语言可以采用 VC, VB 等高级语言。

可视化编辑器是集图形和文本编辑于一体的可视化集成编辑环境, 用于将基本的模型组件集成到仿真环境中, 并对模型进行初始化配置。

仿真模型库存放着基本模型组件。模型库管理器负责类库的维护, 包括类库中类的删除、加入、拷贝等, 并可以帮助用户在模型库中加入新的模型, 删除废弃的模型, 对模型的存储进行管理。

仿真实验管理器实现仿真环境的配置和管理。在仿真运行中, 仿真实验管理子系统是用户与仿真器交互的中介。仿真实验管理器可以对仿真实验进行配置, 并可接受用户的命令,

控制仿真进程。

仿真引擎采用时钟同步算法推进全局时钟,模型的时钟必须与全局时钟进行协调。时钟管理器询问每个模型的局部时钟,仲裁后形成仿真环境的全局时钟,并向模型发出时钟推进命令。

仿真运行运行子系统实现模型间的信息交互和信息共享。

2.3 仿真环境的模型要求

仿真环境中的模型包括 Petri 模型、排队模型等其他离散事件模型。这里采用了 Microsoft 的 COM 技术,将一个离散事件仿真模型对应于 COM 中的组件对象模型。COM 是一套为组件架构设置标准的文档形式的规范,具有良好的可重用性和可扩展性及平台无关性。关于 COM 的技术细节可参考文献[6]。

模型的基本数据结构可以描述为: M = < P, M, E, Port >。其中, P 表示属性集, M 表示方法集, E 表示事件集, Port 表示端口集。

{
属性: 属性 1, 属性 2, 属性 3…
方法: 方法 1, 方法 2, 方法 3…
事件: 事件 1, 事件 2, 事件 3…
端口: 端口 1, 端口 2, 端口 3…
}

属性是描述模型各方面的性质。方法是用于刻画对象的行为。可以被外界访问的方法称为接口。事件是用于引起模型状态发生变化的行为。端口是描述模型与环境间进行信息交互的界面。端口分为输出端口和输入端口。每个端口对应一个消息类型。当一个模型的输出端口的消息类型与另一个模型的输入端口的消息类型相同时,这两个模型的端口存在一个连接关系。每个模型通过仿真环境实现信息交换,仿真环境接收到消息类型为 "Exam"的消息时,寻找仿真环境中的模型,其输入端口的类型为 "Exam",将消息发往该模型。

如图 2 所示,模型 1 的输出端口 1 具有类型 s1,模型 2 的输入端口 2 具有类型 s1,则模型 1 的输出端口 1 和模型 2 的输入端口 2 之间存在一个连接关系。

图 2 模型之间的信息交换

模型的开发规范依据 COM 规范。能够被仿真环境访问的模型的方法称为模型的接口。为实现仿真环境对模型和管理以及模型之间的信息交互,要求模型必须具有以下的接口以及在接口中实现相应的功能:

(1) 仿真初始化

long SimInit(BSTR strInitFunc)

(2) 仿真开始

long SimStart(double SimClock)

(3) 仿真暂停

long SimPause(double SimClock)

(4) 仿真继续

long SimResume(double SimClock)

(5) 仿真停止

long SimStop(double SimClock)

(6) 推进时钟

long SimAdvance(double SimClock)

(7) 询问时钟

double NextEventClock( )

(8) 发送消息

long SendToken(BSTR portName, BSTR TokenData)

2.4 应用示例

我们已经应用这个仿真环境对军师级防空、海上舰艇编队对抗和卫星支持下二炮部队封锁机场等多个典型的 C^3I 系统进行了仿真,分析了天基信息、指挥控制信息对作战效能的影响,并且取得了很好的效果。图 3 是该环境运行时系统的信息流图。

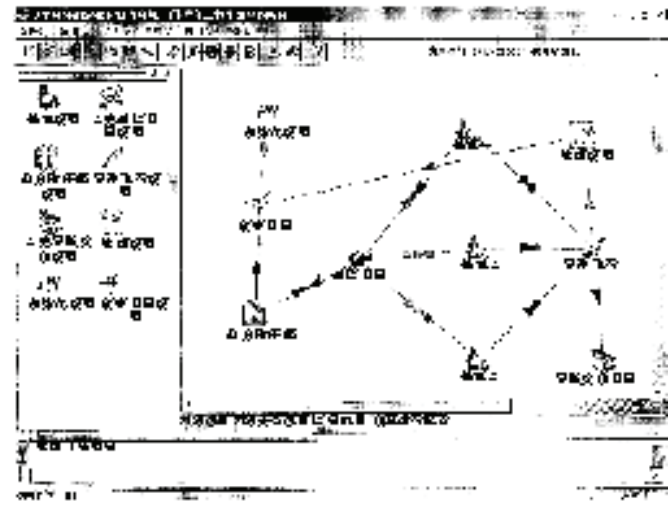

图 3 仿真运行时系统的信息流图

3 结束语

我们在将基于组件的设计思想贯彻到 C^3I 系统仿真中的实践证明,虽然在采用基于组件的设计技术的初期,要编写基本的程序组件,花费的精力、时间要多一些,但这些组件在后来的项目中得到重用,可以大大提高软件开发效率,大幅度降低系统的开发费用;并且当一个系统需要进行修改时,只仅仅改变有关的组件而不影响系统的其他部分。同时,用组件可以很好地实现并行开发和实现重用机制,大大提高了建模与仿真的效率,开发的仿真环境为定性和定量分析 C^3I 系统提供了有效的手段,对 C^3I 系统的分析与设计有指导作用。

参考文献:

[1] hillip Martin. The Rapid Modelling System: A Component-based Approach to the Simulation of Tactics[C]. Proceeding of the 2001 Winter Simulation Conference, Arlington, VA, USA, 2001. 655-662.

[2] Gilbert Chen, Boleslaw K Szymanski. Component-Oriented Simulation Architecture: Toward Interoperability and Interchangeability[C]. Proceeding of the 2001 Winter Simulation Conference, Arlington, VA, USA, 2001. 495-501.

[3] 周敬华, 姜东胜, 李克清. 基于组件的软件开发方法及应用[J]. 计算机应用研究, 1999, 16(10): 74-75.

[4] 左爱群, 黄水松. 基于组件的软件开发方法研究[J]. 计算机应用, 1998, (11): 4-7.

[5] 郭健强, 龚杰民. 构件软件与面向对象技术[J]. 西安电子科技大学学报, 1998, (12): 719-724.

[6] 潘爱民. COM 原理与应用[M]. 北京: 清华大学出版社, 1997.

作者简介:

舒振(1977-), 男, 江西人, 助教, 硕士, 研究方向为 C^3I 系统建模与仿真; 张耀鸿(1971-), 男, 四川人, 副教授, 博士, 研究方向为 C^3I 系统建模与仿真、作战模型与模拟等; 罗雪山(1965-), 男, 湖南人, 教授, 博士生导师, 主要研究方向为 C^3I 系统理论和总体技术研究。