MVC设计模式在网管系统中的应用与研究

马 争[1]，周 艳[1]，谢世波[2]

(1. 电子科技大学通信与信息工程学院 成都 610054; 2. 深圳中兴通讯股份有限公司成都研究所 成都 610041)

【摘要】在分析了MVC设计模式和MVC体系结构的设计原理基础上，将MVC设计模式运用于网管系统的开发设计实例中，给出了应用框架的逻辑结构。该设计方案使逻辑处理流程清晰，便于今后的扩充和维护，也可以方便地应用到电信网管系统的其他应用模块客户端设计中，带来更好的软件结构和代码重用。

关 键 词 模型－视图－控制器; 设计模式; 体系结构; 网管系统

中图分类号 TP31 文献标识码 A

Research and Applications of MVC Design Pattern in NMS

MA Zheng[1]，ZHOU Yan[1]，XIE Shi-bo[2]

(1. School of Communication and Information Engineering, UEST of China Chengdu 610054;

2. Technology Center Chengdu Institute of Shenzhen ZTE Corporation Chengdu 610041)

Abstract Based on the analysis of the principle of MVC design pattern and MVC architecture, this paper discussed the application of MVC design pattern in NMS system development. An example is given to discuss how to use patterns in designing NMS and the structure of framework is illustrated. This design can not only make the logic processing procedure clearer, but also make the later expansion and maintenance easier. It can be applied facilely to the design of other application module of client, and bring better software architecture and reusability of code..

Key words model-view-controller; design-pattern; architecture; network-management-system

设计模式(Design Pattern)是解决某个问题常用的方案。在项目开发过程中，经常会遇到一些相同或相近的问题，为避免每次都寻找新的解决方法，一些能够解决这些常见问题，且已被证实可行的方案被总结出来，构成一个统一的设计模式资源库[1-2]。模型－视图－控制器(Model-View-Controller，MVC)设计模式就是设计模式资源库中的一种。

在开发一个企业级应用项目时，必须面对运行在各种设备上的客户端，如PDA，WAP浏览器以及运行在桌面上的浏览器，所以需要开发不同的应用程序来处理来自不同客户端的请求。采用MVC设计模式可以分离数据访问和数据表现，开发一个有伸缩性的、便于扩展的控制器来维护整个流程，这样可以有效地防止数据的重复访问。

1 MVC思想

1.1 MVC设计模式

MVC是一种划分系统功能的方法，它将一个系统划分为3个部分[3-6]：

收稿日期：2004 – 11 – 10
作者简介：马 争(1957 –)，男，教授，主要从事信号处理、图像处理、计算机网络、网络通信技术方面的研究.

(1) 模型(Model)：是一组表示应用系统商业逻辑的对象。它封装数据源和所有基于这些数据的操作。在一个组件中，Model表示组件的状态和操作方法。

(2) 视图(View)：是一种向用户表达信息的具体方式。它封装的是对数据源Model的一种显示。一个模型可以对应多个视图，而一个视图理论上也可以与不同的模型关联起来。

(3) 控制器(Control)：是应用系统处理具体流程和导向的核心部分。封装的是外界作用于模型的操作。通常，这些操作会转发到模型上，并调用模型中相应的一个或者多个方法。

一般控制器(Controller)在Model和View之间起沟通作用，用于处理用户在View上的输入，并转发给Model。这样Model和View两者之间可以松散耦合，甚至可以在完全不知道彼此的情况下，由Controller实现连接。

MVC的关键是商业模型的设计与实现(模型)可以独立于应用系统的结构设计与实现(控制)以及界面的设计与实现(视图)。如图1所示，在MVC结构中，模型描述应用数据以及用于改变数据的方法，视图用于描述传递数据给用户的界面，控制器则将用户的行为翻译为对模型的相对操作。模型的操作引起的数据变化将反映在视图中[5]。

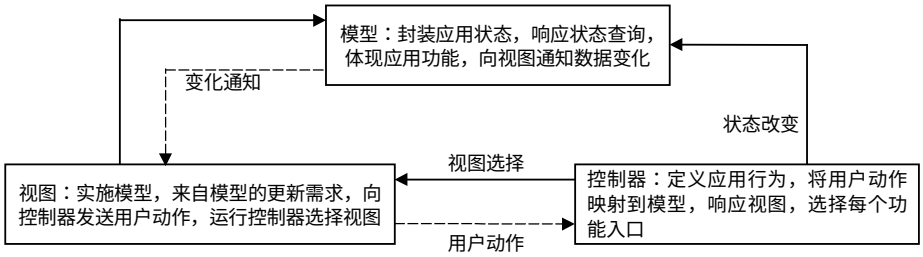

图1　MVC设计模式的结构

使用MVC的好处在于，一方面，通过分离数据和其表现，使得添加或者删除一个用户视图变得很容易，甚至可以在程序执行时动态地进行。同时，Model和View能够单独开发，增加了程序的可维护性、可扩展性，并使测试变得更为容易。另一方面，将控制逻辑和表现界面分离，使程序能够在运行时根据工作流、用户习惯或者模型状态来动态选择不同的用户界面。

1.2　MVC体系结构

在J2EE中，MVC已经不全作为设计模式，而是作为体系结构模式的一个应用程序。作为体系结构MVC的各个部件不再是单纯的类或者接口，而是应用程序的一个组成部分。

在J2EE Blueprint中，SUN推荐了一种基于MVC的J2EE程序的模式。对于企业级的分布式应用程序，它需要能够支持多种形式的用户接口。如，网上商店需要一个超文本标记语言(HyperText Markup Language，HTML)的界面来与网上的客户联系，提供无线应用协议(Wireless Markup Language，WML)的界面给无线用户。管理者可能需要传统的基于Swing的应用程序来进行管理。对于商业伙伴，基于可扩展标记语言(Extensible Markup Language，XML)的Web服务可能更为方便。MVC是这样一个问题的有效解决方法。通过以控制和显示逻辑分离为核心的数据存取功能，形成一个Model，能够让多种视图来共享。

另外，J2EE中有几个核心的技术，Java伺服器网页(Java Server Pages，JSP)，JavaBean，伺服器(Servlet)，企业JavaBeans(Enterprise JavaBeans，EJB)，会话Bean(SessionBean)，实体Bean(EntityBean)也体现了MVC的思想[3]。JSP能够生成HTML，WML甚至XML它对应于Web应用程序中的View部分。EJB作为数据库与应用程序的中介，提供了对数据的封装。一般EntityBean封装的是数据，而SessionBean封装的是对数据的操作。这两个部分合并起来，对应于Web应用程序的Model部分[4]。在技术上，JSP能够直接对EJB进行存取，但这并不是好办法，因为会混淆程序中的显示逻辑和控制逻辑，使JSP的重用性能降低。通常的解决方法是通过JavaBean或Servlet作为中介的控制逻辑，对EJB所封装的数据进行存取。这时，JavaBean或Servlet将对应于Web引用程序中的Controller部分，有时这两者也可以混合使用。

由此可见，MVC模式已经实现了与J2EE框架的结合，它已被提升到一个体系结构模式的高度，具有了更加广泛的含义。

2 MVC思想在网管系统中的运用

MVC的思想除了应用于整个体系结构的设计外，还在EJB数据库访问，客户端数据组织等细节设计部分都得到广泛的运用。下面将以网管系统客户端主视图的开发程序为例，说明MVC思想在数据组织和呈现中的运用。由于MVC是一种面向对象的设计模式，本文以统一建模语言(Unified Modeling Language，UML)建模的方式进行分析说明：该网管系统的客户端主视图主要负责显示被管网元基本信息，包括网元的拓扑树，网元拓扑图、被管网元的告警信息和告警统计。主视图接收来自服务端的拓扑信息和告警信息，通过简单的数据组织和处理将数据呈现出来。通过分析可将主视图的功能实现如图2所示的模型。

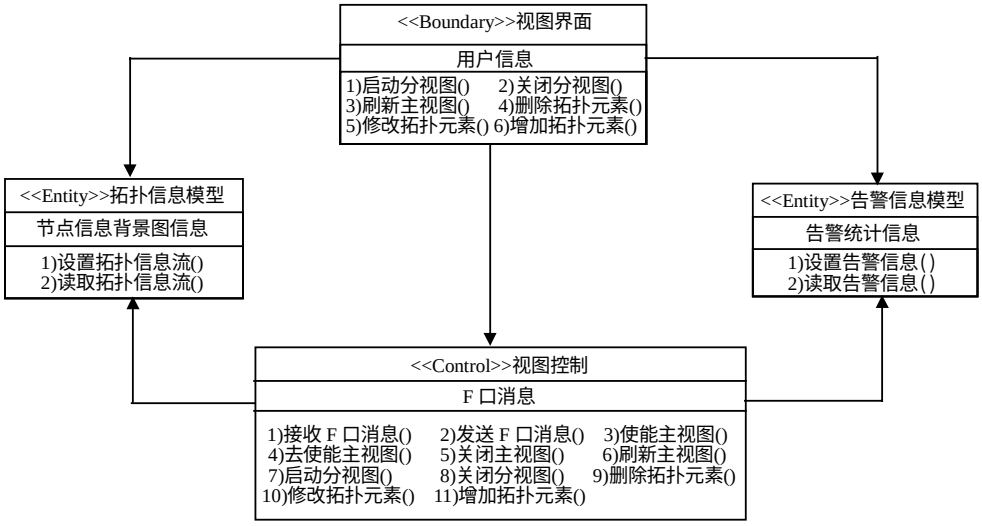

图2 网管系统主视图框架分析模型

图2中，视图界面是整个系统设计的边界类，它负责将数据以不同的形式呈现出来，同时接收来自用户的请求，完成系统和用户的互动操作；告警信息模型和拓扑信息模型是系统设计的实体类，它们构成了主视图的数据模型；视图控制是系统设计的控制类，它完成了整个逻辑控制和逻辑计算的功能，负责对来自视图界面的请求做出应答，更新数据模型。该模型采用MVC设计思想，将控制逻辑、表现界面和数据模型分离，增强了程序的可维护性和可扩展性。同时，主视图的设计充分发挥了MVC思想的优势，完成了一个数据模型到多个表现组件的对应。具体体现在图像用户接口(Graphical User Interface，GUI)界面上的拓扑图和拓扑树，它们共享了同一个拓扑数据源，是同一数据模型在GUI界面的不同表现形式。

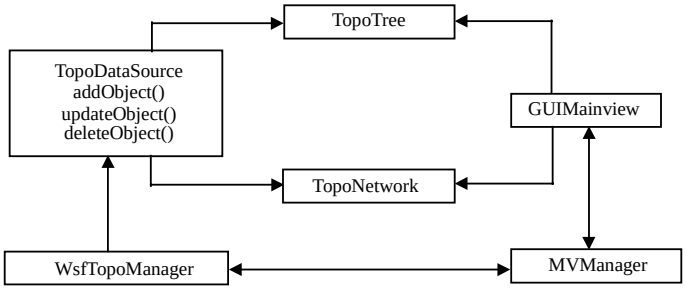

图3 网管系统主视图框架设计模型

在如图3所示的主视图设计模型中，TopoTree和TopoNetwork都是GUI界面上的一个组件(View)，它们分别实现网元信息以树型结构显示和以网络关系图结构显示的功能。与其关联的都是同一个TopoDatasource

(Model)，由它负责显示数据的增加、删除和更新。当服务端数据有所变化时通过信息上报通知客户端，MVManager(Controller) 负责侦听这种上报信息，并将信息转发给 WsfTopoManager(Controllor)。WsfTopo-Manager对上报数据进行处理，解析出符合模型结构的数据信息，并发送到TopoDatasource中。这样的设计使得数据模型完全从界面显示中独立出来，如果想增加其他形式的显示，也只需对GUI设计添加新的组件，而不必再修改数据模型和控制类，降低了耦合度，提高了开发效率。

由此可见，MVC思想在项目的分析和设计中起到了重要的作用，它帮助将控制逻辑、表现界面和数据模型分离，使系统功能扩充和代码维护变得简单起来。

3　结 束 语

MVC最早运用于Smalltalk。其成功的应用使之逐渐被人们接受，并被抽象成为一种设计模式。随着与SUN各项技术的融合，MVC模式逐步扩展成为一种架构的思想，尤其在以Java的Servlet、JSP和JavaBean动态网页为基础的电子商务开发技术上得到了广泛地应用。而对于当前综合网管系统的开发趋势来说，系统结构的复杂度、代码之间的耦合度、代码的易维护性、应用框架的可重用性、EJB组件的可重用性等问题成为系统设计的重点。MVC思想中数据和控制分离的思想正好为解决这些问题提供了方案，它将逐渐成为项目设计和开发的主流思想。

参　考　文　献

[1] Erich G. Design patterns elements of reusable object-oriented software[M]. 北京: 机械工业出版社, 2002

[2] Vinoski S. CORBA: integrating diverse applications within distributed heterogeneous environments[J]. IEEE Communications Magazine, 1997, 14(2): 1-12

[3] Deepak A. Core J2EE patterns[M]. 北京: 机械工业出版社, 2002

[4] Alan S, Jim T 著. 设计模式精解：面向对象设计的新视角[M]. 透　明 译. 北京: 清华大学出版社, 2002

[5] 邓玉龙. MVC设计模式在电子商务系统中的设计与应用[J]. 南京邮电学院学报, 2002, 22(2): 80-83

[6] Grant L. Component-based enterprise frameworks[J]. Communications of the ACM, 2000, 43(10): 24-26

编　辑　漆　蓉

·科研成果介绍·

微波泄漏检测仪

微波泄漏检测仪是用于测量电子设备特别是微波设备电磁波泄漏的一种专门仪器，是近场测量仪，具有灵敏度高，响应快，全方位探头，数字显示，可交直流供电，便携式，最大值锁定等特点。它使测试仪器不仅能测连续波泄漏，同时能测量脉冲泄漏信号，满足各种辐射源电磁波泄漏的测量要求。

其主要技术指标：(1) 三维全共心立体探头；(2) 分辩率：0.1 μW/cm^2；(3) 工作频率：2 450 MHz± 50 MHz；(4) 校准点：1 mW/cm^2；(5) 校准误差：不大于1.0 dB；(6) 测量不确定度：不大于2.0 dB；(7) 量程：2 mW/cm^2。

·渠　涌·