

基于 MDA 的软件建模方法研究

Research on Software Modeling Approach Based on MDA

杜 选 (嘉兴学院 数学与信息工程学院 浙江 嘉兴 314001)

摘 要: 模型驱动架构(MDA)是一个新的软件开发框架,它溶入了软件工程领域的新思想、新方法和新技术。本文研究了基于 MDA 的软件建模方法,探讨了模型在整个软件生命周期中起的关键作用,提出了一种从平台无关模型(PIM)到平台相关模型(PSM)的转换方法。

关键词: 模型驱动架构 平台无关模型 平台相关模型 软件建模

1 引言

按照软件工程学对软件生命周期的划分,一般的软件开发过程包括需求分析、总体设计、详细设计、编码、测试及软件维护等主要阶段。在软件开发过程中,需要多种人员的参与,需要大量的人与人之间的交流,这就需要有一整套的软件工程理论与技术体系作指导。

模型驱动架构(Model Driven Architecture, MDA)是对象管理组织(OMG)提出的先进的软件开发方法学,它通过提供一组规范,来解决软件系统生命周期中与集成相关的互操作和可重用问题,包括从建模到系统设计、组件构建、组合、集成、分发、管理以及进一步的发展,为利用和融合现有的一些成熟标准奠定了基础。在 MDA 中,模型不仅仅是描绘系统,辅助沟通的工具,而是软件开发的核心和主干,这就为软件开发提供了一种崭新的思路。MDA 由于将应用系统的行为逻辑从特定的支撑环境和平台中抽象出来,并采用广泛支持的可视化建模语言进行描述,在更大程度上实现了系统的互操作和可重用,提高了开发效率;并且由于对模型的构建及转换提供了一致的标准和规范,提高了模型构建的标准化和自动化的程度。

2 MDA 的系统开发

2.1 MDA 概述

MDA 是一个开放的、中立于软件供应商的架构,它广泛地支持不同的应用领域和技术平台,能够成为应用领域和具体技术平台之间的杠杆。MDA 技术的

核心概念均是 OMG 的一系列标准:统一建模语言(Unified Modeling Language, UML),元对象设施(Meta Object Facility, MOF),XML 元数据交换(XML Metadata Interchange, XMI),公共数据仓库元模型(Common Warehouse Meta-model, CWM)^[3]。MDA 通过对 OMG 中不同的标准采用一个统一的元模型来将它们融合到一个统一的视角,支持软件设计和模型的可视化、存储和交换。MDA 是以构建机器可读的系统模型作为系统开发的驱动力,这些模型以独立于实现的技术开发、以标准化的方式储存。因此这些模型可以被重复访问,并被自动转化为纲要、代码框架、测试工具、集成化代码及各种平台的部署描述^[4]。在 MDA 中,建模语言被用作为一种开发语言,而不仅仅是设计语言。

MDA 定义了一种系统描述方法,将系统功能描述从特定技术平台的功能实现描述中分离出来^[5]。MDA 的核心思想是采用标准的建模语言,从现实系统中抽象出与技术无关、与领域相关的业务模型,针对不同实现技术制定多个映射规则,然后通过这些映射规则及自动化工具将业务模型转换成领域相关、平台相关的应用模型,最后,利用工具将其自动转换成代码,进行集成和测试。

2.2 MDA 系统设计和开发过程

基于 MDA 的系统设计和开发过程是以系统建模作为驱动力的,模型是开发过程中的关键。MDA 将软件系统的模型分离为平台无关模型(Platform Independent Model, PIM)和平台相关模型(Platform Specific Model, PSM),同时又能通过转换规则将它们统一起

来,以这样的方式试图去摆脱需求变更所带来的困境。平台无关模型 PIM 是对系统高层次的抽象,其中不包括任何与实现技术相关的信息;平台相关模型 PSM 是跟特定平台相关的模型;它们各自的改变都可以是相互独立的,不会造成商业逻辑和实现技术的紧密耦合,同时 MDA 又可以通过转换来弥补它们之间的鸿沟。MDA 开发模式的生命周期与传统的方式看起来区别并不大。它们都具有相同的开发阶段,主要的区别就是各个阶段的设计工件是不相同的,MDA 的设计工件是正式的精确模型,它们能够被机器所理解。

MDA 的系统开发首先使用平台无关的可视化建模语言(UML)来搭建平台无关模型,每个系统仅需建立一次 PIM;然后根据特定平台和实现语言的映射规则,将 PIM 与特定平台相结合,利用标准的转换规则将其转换成平台相关模型 PSM,最终生成应用程序代码和测试框架^[6]。在将 PIM 模型转换为 PSM 模型时,不同的底层实现平台其相应的 PSM 模型是不同的,它决定了模型的映射方式。编码是从 PSM 模型到代码的映射,通常用绑定与 PSM 模型的程序来实现,这种程序就是元程序(MetaPrograms)。

MDA 开发过程中,由于模型是建立在统一的技术框架下的,因此不同阶段工件的转换过程是自动化的。这也就意味着在 MDA 开发过程中,大量成熟、广泛的自动化转换工具的研制与开发也是系统开发工作重要的组成部分。

3 平台无关模型(PIM)

基于 MDA 的开发中,通过平台无关模型(PIM)完整描述业务功能,业务建模独立对业务行为和领域元素建模,包括对模型的理解、设计、创建、维护和修改。MDA 软件开发是应用各阶段模型来驱动软件的生成,关注系统应用的本身,而不是将中间件平台作为系统开发的中心。

3.1 业务模型描述实体

业务建模的过程就是抽取固定的业务模式,通过形象的方式描述业务模式。对业务过程的描述需要定义一系列的基本概念和术语,描述模型的组成,实现业务过程的建模。具体来说,在描述模型时使用以下的具体实体。

(1) 角色:角色描述了业务过程中参与操作的人

员,蕴含了角色对其特定的信息的操作能力。

(2) 功能包:功能包将功能不同的业务过程分离开,降低模块间耦合度,反映业务过程的目标。

(3) 业务过程:这个过程模型反映了一个业务处理过程,它是按照一定方式由角色和活动用例组合而成。一个功能包内通常包含一个或多个业务过程。

(4) 活动用例:活动用例相对于业务活动中的事务,主要反映某个角色完成一件具体的事件。通过角色参与的活动用例的组合,就可以用来描述一个业务过程。

(5) 活动状态:活动状态描述了完成某个特定活动用例所采取的步骤。通过描述活动状态可以反映完成相似活动的细微区别。

(6) 转换条件和数据:根据相关数据和转换条件进行业务或者活动状态的推进,它是执行任务推进的依据。转换条件对应业务过程中的业务规则和操作顺序,数据为业务规则提供数值上的描述。

3.2 业务层建模过程

对于整个系统来说,通过功能包这一层的模型,就可以确定系统实现的范围。根据上述的描述实体,对系统进行建模过程如下:

(1) 按照组织情况划部门的功能包。功能包描述了要达到的目的或者目标。功能包内通常包括了部门的一个或多个业务功能,一个业务功能用一个业务过程来进行描述,体现了业务功能的目的和作用。

(2) 定义一个业务流程。一个业务流程就是一个用箭头连接起来的一组活动用例和角色的组合。它反映了业务的执行过程(如用户管理过程等)、角色产生活动操作,并且,一个活动用例只与一个角色发生关系。在描述业务过程的时候就包括:处理的事件(活动用例)、事件之间的先后关系(用箭头的指向描述)和参与事件的角色(人物和活动用例之间通过直线连接)。

(3) 每个活动用例对应一个事务(如用户设置等),这些事务还可以继续细分成不同的活动状态。一个活动用例就是一个用箭头和判断条件连接起来的一组活动状态的组合。每个活动用例都是由活动状态组成的,活动用例至少有三个活动状态:开始状态、活动状态和结束状态。所有活动用例的都是通过活动状态的组合进行描述,所以,最终系统就是由许多活动状态

的组合进行描述。

4 平台相关模型(PSM)

业务层建模结束之后,需要将业务层模型反映到实际的应用中去。本文将 J2EE 平台为例,对平台层模型进行描述。

4.1 J2EE 平台层模型

J2EE 平台采用 MVC (Model View Controller) 模型 - 视图 - 控制器模式来开发企业级应用程序。典型的 J2EE 模型包括: 客户层、表示层、业务逻辑层和数据层。

(1) 客户层是浏览器,用于作为客户端显示应用程序。

(2) 表示层负责处理用户接口及与用户交互相关的组件,对应 J2EE 平台 PSM 的 Web 模型。Web 模型用 UMLProfile 创建,用于生成表示层代码的一些信息。在 Web 模型中可以使用框架提供的页面描述工具,使用库 (Library)、模板 (Template)、层叠式样式表 CSS (Cascading Style Sheets) 等技术,使得页面美观、条理清晰和便于维护。

(3) EJB 体系结构是封装业务逻辑的服务器端的组件模型,对应 J2EE 平台 PSM 的 EJB 模型。EJB 模型是处理事务、安全、持久性、可测量性的中间层模型,它是在代码之上的抽象层次,这部分的建模已经和代码层关系紧密了。EJB 模型用 UMLProfile 创建,包含有 UML 中的类、关联等,还专门为 EJB 定义了一些构造型 (Stereotype)。

(4) 数据层包括各种异构数据库 (如: MSSQL、MySQL、Oracle) 和现有的各种遗留系统数据,对应 J2EE 平台 PSM 的 DBMS 数据库模型。UML 对象类图能够更好地用于对数据库建模。

4.2 平台层模型对业务层模型的映射

基于 MDA 软件开发方法的一个主要内容就是从 PIM 到 PSM 的映射,也就是从分析模型到实现模型的转换。当确定了平台,平台就限定了最终实现中的很多内容。例如,对于 J2EE 平台,显示层是使用 JSP 进行描述,使用 EJB 描述业务逻辑,EntityBeans 处理数据。因此,业务层模型和最终的代码已经相当接近了。

4.2.1 Web 端的 PIM 到 PSM

Web 端的 PIM 到 PSM 的映射生成的是基于 Struts 的应用,符合 MVC 架构。Struts 是一个 Frame - work,

具有相当的可靠性和稳定性,同时它也是一个灵活的软件架构,能够支持良好的重用和扩展。

4.2.2 业务层的 PIM 到 PSM

业务层的 PIM 模型到 PSM 模型的映射是 MDA 映射中最关键的部分。该转换过程分为 4 步:

(1) 为业务逻辑中的每个类生成对应的 Session-Bean 子系统和 EntityBean 子系统。SessionBean 用于和客户端的交互,EntityBean 用于和后台数据库的交互。

(2) 精化每个 EJB 子系统,建立 Bean 所需的接口和类。

(3) 建立角色和 Bean 之间的访问关系,用于描述 EJB 的安全策略。在 J2EE 平台中使用如下机制来保护 Bean 资源,实现 Bean 的安全性:首先定义系统中的角色;然后为每个方法指定哪些角色可以调用它;最后在程序中可以用 EJBContext 接口的 getCaller - Principal 和 is Caller In Role 方法来实现安全控制。该机制实际上是一个访问控制列表 (Access Control List)。

(4) 生成整个业务逻辑的组件图。每个子系统 (EnterpriseBean) 在实现时都对应一个 JAR 包,它包含各个类和接口的 .class 文件以及配置描述信息。

4.2.3 后台数据库的 PIM 到 PSM

对数据库建模,首先考虑业务逻辑层中哪些类的数据是要存放在数据库中的,从 UML 类模型到 RDB (关系数据库) 模式设计的映射需考虑 RDB 模型的局限性,然后,按以下规则设计数据库。

(1) 映射实体类 (EntityClasses) 实体类到关系表的映射必须遵循表的第一范式,列必须是原子的。

(2) 映射关联 (Associations) 和聚合 (Aggregations) 将关联映射到 RDB 涉及到表之间参照完整性约束的使用。任何“一对一”或“一对多”的关联都可以通过在一个表中插入一个外键来匹配另一个表中的主键来表达。关联映射的主要原理也适用于聚合映射。

(3) 映射类属 (Generalizations) 将类属关系的层次映射为 RDB 设计模型的方法是:将每个类映射到一个表;将整个类层次映射到一个超类 (Superclass) 表;将每个具体类 (ConcreteClass) 映射到一个表;将每个不相交的具体类映射到一个表。

5 结语

本文研究基于 MDA 的应用,(下转第 117 页)

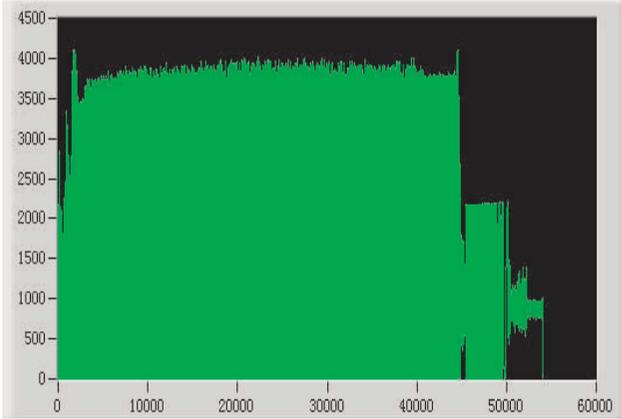


图3 CWGraph 控件显示信号时域波形

```
CNiMath : : AutoPowerSpectrum ( dwRead0 , spectrum0 ,
df , dt ); // 计算频谱
m_spectrumGraph0. PlotY ( spectrum0 , 0 , df ); //
显示信号频谱
```

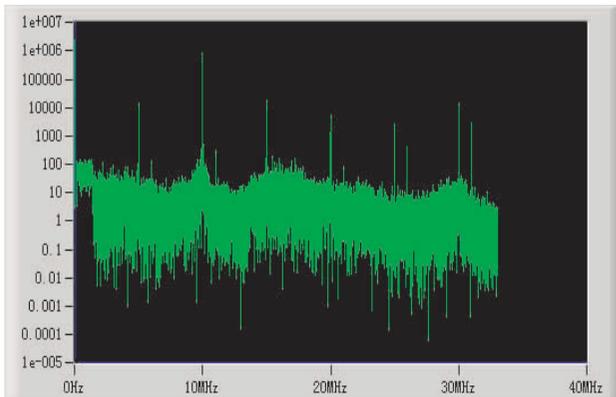


图4 CWGraph 控件显示信号频谱另外

对信号的其他处理和显示方法类同,调用 LabWindows/CVI 的信号处理函数处理和 CWGraph 控件显示处理结果,这里不一一列举了。

5 结论

系统采用目前普遍采用的信号处理系统的开发方式,即使用 USB2.0 作为通信接口将硬件和各种计算机相连,加上相应的软件,实现数字化信号的采集、控制、测试、分析,研究开发了具有 Windows 风格的信号处理系统。该系统具有良好的用户界面,并且在实际

应用中验证了系统的可行性。采用 VC++ 开发工具和 LabWindows/CVI 相结合的开发方法,具有较强的灵活性和可扩充性,在实际应用中可以得到不断的改善和发展。

参考文献

- 1 杨智君,马晓庆,吴昭春. Windows 操作系统中的 USB 2.0 体系结构分析. 电子测试,2008. 20-23.
- 2 朱中锐,蔡志明,郭岩. 基于 LabWindows/CVI 数据采集系统. 电子测量技术,2007. 105-106.
- 3 王红伟,舒大文,朱琦琦. 基于 Visual C++ 6.0 的信号处理系统研究. 微计算机应用,2005. 372-374.
- 4 王奕,王凯. 基于 USB2.0 的数据采集系统的设计与实现. 电子工程师,2002. 15-17.
- 5 刘改梅,韩慧莲. 基于 LABVIEW 的 USB 无线通信接口的设计. 计算机技术与应用,2007. 121-123.

(上接第 82 页)

以平台独立模型 PIM 和平台相关模型 PSM 的变换为驱动,实现从 PIM 到 J2EE 平台上的 PSM 自动转换。对软件开发而言,MDA 的软件开发进行对象的建模,并能生成大部分的代码,为应用程序的互操作性和可移植性提供了全面与结构化的解决方案。

参考文献

- 1 董建武. MDA: 新一代软件互操作体系结构. 计算机工程,2003, (2).
- 2 林炜,夏宽理. 基于 MDA 的模型转换方法研究. 计算机工程与应用,2005, (2).
- 3 Kleppe A, Warner J, Bast W. 鲍志云译. 解析 MDA. 北京:人民邮电出版社,2004.
- 4 徐晓钟,谢康林. 基于 MDA 方法学软件开发方式的原理与实现. 微机发展,2004, (4).
- 5 张小华,韩永生,余军合. 模型驱动体系综述. 计算机工程, 2004, (2).
- 6 黎才茂,关丽霞. MDA: 模式驱动体系的软件建模. 现代计算机, 2006, (7).