

基于 UML 的 Web 应用程序系统分析与设计

王 晟¹ 卢晓莉²

(1、厦门大学研究生院 2、厦门大学自动化系,福建 厦门 361005)

【摘要】 开发 Web 应用程序的中心很少关注开发的过程。本文给出利用 UML 为 Web 应用程序建模,并将此方法应用于厦门大学研究生选课系统的分析、设计和建模过程。这种程序的开发方法不但降低了系统复杂性,还为开发人员提供交流的工具,同时提高了系统重用性和可维护性,提高了系统开发的效率。

【关键词】 UML 统一建模语言 Web 程序设计

一、引言

面向对象分析与设计 (Object - Oriented Analysis and Design, 简称 OOA &D 或 OOAD 方法)的发展促成了统一建模语言 UML 的产生。目前,UML 是一种先进实用的标准建模语言,它可以为软件系统的产出建立可视化模型,是系统的结构变得直观,易于理解,同时 UML 符号具有定义良好的语义,不会引起歧义,这就使得用 UML 建立的模型不但利于交流,还有利于对软件的维护。另外,UML 还定义了开发软件系统过程中所做的所有重要的分析、设计和实现决策的规格说明,这就意味着用 UML 建立的模型是准确的、无歧义的。

随着网络技术的发展,Web 应用程序的应用越来越广泛。Web 应用程序与传统应用程序相比,具有重要的优势。但是到目前为止 Web 应用程序开发的中心一直是工具,开发人员往往忽略了极其重要的系统分析和设计过程。如今,Web 应用程序变得越来越复杂,为 Web 应用程序建模的重要性日益突出,UML 逐渐被应用于 Web 应用程序的分析与设计过程。

本文讨论了 UML 在 Web 应用程序设计中,尤其是瘦 Web 客户端模式的应用程序中的应用,并且利用 UML 针对厦门大学研究生选课系统进行面向对象的建模,按照面向对象分析与设计的一般过程对系统进行需求分析。

二、基于 UML 的 Web 应用程序建模

Web 应用程序是从 Web 站点或 Web 系统发展而来的,它扩充了 Web 站点,使用户可以调用其业务逻辑,改变服务器中业务的状态。Web 应用程序至少包括了三个重要组件:客户端浏览器、Web 服务器和应用程序服务器,另外还可能包含数据库服务器。Web 应用程序的体系结构模式描述了软件系统的基本结构组织机制。其体系结构模式一般分为三种:瘦 Web 客户端模式、胖 Web 客户端模式和 Web 发送 (Delivery) 模式。一般的基于 Internet 的 Web 应用程序均采用瘦 Web 客户端模式。瘦 Web 客户端模式的基本架构如图 1 所示。

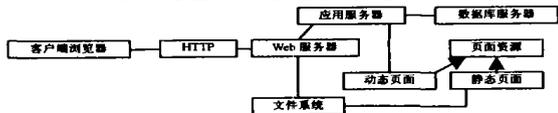


图 1 瘦 Web 客户端模式的基本架构

瘦 Web 客户端模式的重要组件主要位于 Web 服务器上,所有的业务逻辑均在 Web 服务器上执行,客户端的计算能力极其有限。

UML 主要由图和原型组成,从不同的视角为系统的架构建模,由此系统的不同视图。这些视图包括用例图、类图、对象图、时序图、协作图、活动图、状态图、组件图和配置图。每一种视图都是由一个或多个图组成的。一般建模时用例图、类图和对象图描述系统的静态结构,建立系统的静态结构模型,用时序图、协作图、活动图和状态图描述系统的动态行为,建立系

统的动态行为模型;用组件图和配置图描述系统的物理模型。在进行系统分析与设计时,一般先给出系统的需求,进行系统的需求分析。需求分析的目标是识别系统参与者、系统用例,得到系统的用例图,并用系统的用例事件流说明系统的业务流程。接着进一步分析系统需求,确定类以及类与类之间的关系,确定它们的静态结构和动态行为,最终应用类图和对象图描述系统的静态结构,用时序图、协作图、活动图和状态图描述系统的动态行为。系统分析与设计的最后,利用组件图和配置图构建系统的物理模型。

在利用 UML 为 Web 应用程序建模时,因为 UML 的符号表示是有限的,因此大量使用 UML 的扩充机制,如原型、标值、约束等。利用这些扩充机制来定义代表 Web 技术组件的 UML 符号,分别用来代表服务器页面、客户端页面、Java Applet 等 Web 技术组件。图 2 列出的就是 Rational Rose 中的分别代表服务器页面、客户端页面、HTML 表格、COM 对象和 Applet 的 UML 符号原型。

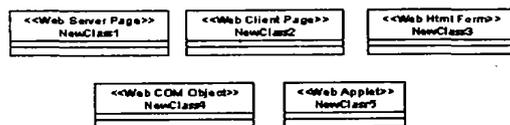


图 2 Web 组件的 UML 符号原型

三、研究生选课系统的建模

研究生选课系统是厦门大学研究生院网上办公系统的一个主要模块。该系统为采用 ASP 编写 Web 应用程序,后台数据库采用 SQL Server2000 关系数据库。研究生选课系统的基本结构模式属于瘦 Web 客户端模式,因此该系统对网页的使用设置了最严格的约束,规定了每个网页只能含有当前 HTML 所规定的结构元素。在选课系统应用程序中,用户只与客户端页面交互,服务器只与服务器资源交互,所以客户端页面和服务器页面应包含在时序图中。在实际设计系统模型时,可以直接将系统分析时得到边界类直接转换为客户端页面,将控制对象类直接转换为服务器页面。

3.1. 研究生选课系统需求分析

系统需求分析是系统分析和设计的第一步,也是关键的一步。UML 是采用用例驱动的分析方法,其分析需求的主要任务是准确地分析系统的功能需求,识别出系统的参与者 (Actor) 和用例 (Use Case),建立用例图 (Use Case Diagrams),为系统分析和设计的后续工作打好基础。

3.1.1. 系统功能需求

研究生选课系统主要功能就是提供研究生网上选修课程和授课教师网上维护课程信息及获取学生选课信息的功能。对研究生选课系统的域描述如下:

在研究生选课系统中,首先应为每个在读研究生、授课教师 (也可是各院系的研究生教学秘书) 和系统管理员建立一个

账户,其中在读研究生账户存储研究生的个人信息(包括学号、姓名、系别、专业、登录密码)和选课科目,授课教师账户存储授课教师的个人信息(包括姓名、系别、登录密码)及授课科目,系统管理员账户存储帐户名和密码。研究生可以网上注册并维护个人信息、查询课程信息和授课教师简介、选择要修课程;授课教师可以网上注册并维护个人信息、维护课程信息、获取学生花名册;系统管理员可以注册并维护个人信息、创建课程目录、维护课程信息、维护学生信息和维护教师信息,系统管理员具有最高的权限。每门课程的选课过程都必须遵循一定的规则,该规则存储在课程数据库的规则数据库里,学生选课时系统会自动验证此规则,若规则不符合则学生不能选择该课程。

3.1.2. 识别参与者

通过对系统功能需求的分析,可以确定系统中三个参与者:Student(在读研究生)、Lecturer(授课教师)、Administrator(系统管理员)。

各参与者的描述如下:

(1) Student

描述:研究生可以网上注册并维护个人信息、查询课程情况、查询授课教师简介和选择课程。

示例:在读研究生。

(2) Lecturer

描述:授课教师注册并维护个人信息、维护课程信息、获取学生花名册。

示例:在职教师,研究生教学秘书。

(3) Administrator

描述:系统管理员可以注册并维护个人信息、创建课程目录、维护课程信息、维护学生信息、维护教师信息。

示例:系统管理员。

3.1.3. 识别用例

在前述识别参与者的基础上,对系统需求进一步分析,可以确定系统中有如下用例存在:

- (1) Maintain Student Individual Info(维护在读研究生个人信息):本用例提供了维护在读研究生个人信息的功能。
- (2) Query Course Info(查询课程信息):本用例提供了查询课程信息的功能。
- (3) Query Lecturer Info(查询授课教师简介):本用例提供了查询授课教师简介的功能。
- (4) Select Course To Learn(选择课程):本用例提供了在读研究生选课的功能。
- (5) Maintain Lecturer Individual Info(维护授课教师个人信息):本用例提供了维护授课教师个人信息的功能。
- (6) Request Course Roster(获取选课研究生花名册):本用例提供了获取选课研究生花名册的功能。
- (7) Maintain Administrator Individual Info(维护系统管理员个人信息):本用例提供了维护系统管理员个人信息的功能。
- (8) Maintain Student Info(维护研究生信息):本用例提供了维护研究生信息的功能。
- (9) Maintain Lecture Info(维护授课教师信息):本用例提供了维护授课教师信息的功能。
- (10) Create Course Catalog(创建课程目录):本用例提供了创建课程目录的功能。
- (11) Maintain Course Info(维护课程信息):本用例提供了维护课程信息的功能。
- (12) Log In(登录):本用例描述了用户如何登录进入软件系统。

在识别出参与者和用例后,建立用例图,用以描述参与者、

用例之间的关系。

系统的用例图如图 3 所示。

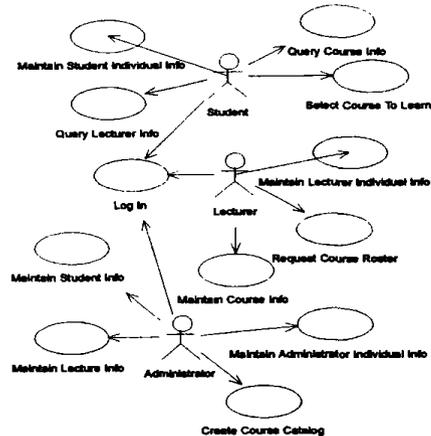


图 3 研究生选课系统的用例图

3.2. 系统静态结构模型

研究生选课系统的静态结构模型此处用类图描述。系统包含的类图包括实体类图 and 用户界面类图等。本文以实体类图为例,介绍如何用 UML 建立系统静态结构模型。

根据研究生选课系统需求分析,识别出系统中存在的实体如下:

研究生(Student); 授课教师(Lecturer); 系统管理员(Administrator); 帐户信息(Log Info); 课程目录(Course Catalog); 课程信息(Course Info); 研究生信息(Student Info); 教师信息(Lecturer Info); 选课研究生花名册(Course Roster);

根据已识别的系统中存在的对象,可以识别出系统中存在的类,如下所示:

类 ClassStudent; 类 ClassLecturer; 类 ClassAdministrator; 类 LogInfo; 类 CourseCatalog; 类 CourseInfo; 类 StudentInfo; 类 LecturerInfo; 类 CourseRoster; 类 Persistent(数据库文件中的持久对象)

各个类所具有的私有属性、公共操作以及类之间的关系详见类图,即图 4。

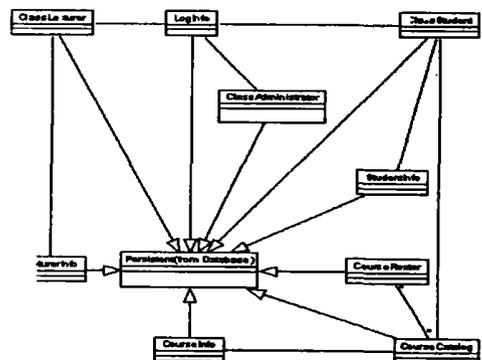


图 4 系统实体类图

3.3. 系统动态行为模型

UML 语言建模时一般采用时序图描述用例的主要场景,采用状态图描述了对象的动态行为。研究生选课系统用例的主要场景为研究生网上选择要修的课。描述系统用例的主要场景的时序图如图 5 所示。时序图起始于参与者 Student(在读研究生)发送消息“Select Course”(选课)给客户端页面(SelectCourseToLearn Datalog),本质上,这个消息是一个请求 Course Catalog(课程目录)页的命令。

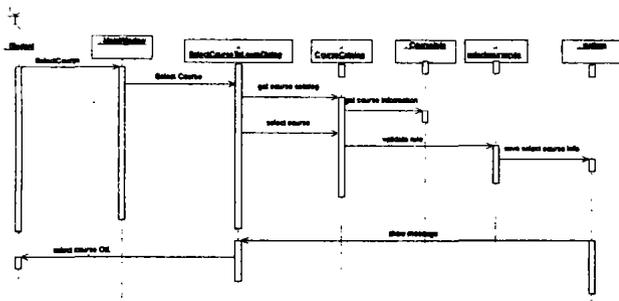


图 5 “选课”时序图

四、基于 UML 的 Web 应用程序建模的关键点

利用 UML 对 Web 应用程序建模时,应注意区别 Web 应用程序与以往传统程序设计的不同。以瘦 Web 客户端应用程序建模为例。

瘦客户端模式的 Web 应用程序中,服务器页面常常由 Web 服务器载入。尽管服务器页面可以实现某些业务逻辑,但是由于这些业务逻辑被重用于网页处理,因此服务器页面不是放置这些业务逻辑的最优位置。这时就应当考虑将这些业务逻辑放置在已被编译过的服务器组件中,这样非 Web 应用程序系统也可以重用他们,从而提高了系统的可重用性。另外,在瘦客户端模式下,服务器负责协调服务器端的业务对象活动,建立用户界面并将其发送给客户端浏览器,这样就常常导致服务器页面过载。鉴于此,在系统分析和设计时,可以考虑将域的细节放在服务器页面外减轻服务器页面的责任并使服务器页面易于维护,这个设计思想可以通过分离业务逻辑协调与用户界面建立的方法来实现。

系统设计时应识别出网页、重要的协作和责任,然后再逐一设计各个页面。重点要考虑服务器页面、链接(Links)、表单(Forms)和框架(Frames)等元素的设计。

在研究生选课系统设计时采用的是微软的活动服务器页面(Active Server Pages,简称 ASP)技术实现服务器。本质上,ASP是过程的,引擎从头至尾顺序处理页面,执行所遇到的可执行语句,使得整个页面成为一个 main() 函数。这样,为 <

Web Server Page > 类定义的任何属性都会映射到在页面开始定义的一个变量 Session,这个变量的作用于是整个页面,可被页面中定义的所有函数访问。为服务器页面定义的操作将映射到 ASP 文件中的函数。ASP 文件中函数外的代码被认为是页面的入口函数(main() 函数)的一部分。ISAPI、NSAPI、Java Servlets 为服务器页面定义了更像对象的接口。

在 Web 应用程序中,可以用多种方式来表达网页之间的链接关系。当客户端页面链接到另一个页面时,本质上是链接到网页组件,而不是客户端页面或服务器页面抽象,所以本质从客户端页面到另一个客户端页面的链接与从客户端页面到 < Builds > (建立)客户端页面的服务器页面的链接是一样的。“Link”既可以链接到客户端页也可以链接到服务器页。要描述支持应用程序的特定流的网页结构关系的图可能只包含客户端页,所以“Link”只能链接到客户端页。

Web 应用程序建模时,< Web Html Form > 对象只存在于客户端页的上下文中,是标准输入元素的集合,它接受来自用户的输入并提交给服务器页处理。表单(Forms)总是把信息提交给服务器页,服务器页可以是 ASP 页、CGI 脚本、或者 ISAPI/ NSAPI DLL。表单的属性即为它的输入域。

Web 应用程序建模时,最难建模的 Web 元素之一是框架(Frames)。框架使得 Web 设计者浏览器窗口分成多个矩形子区域,用来显示不同的 Web 页面。在 Internet 上,框架通常被用来将窗口分为导航格和内容格。框架集(Frameset)是一种特殊的网页,它将视区分成多个格,每个格都含有自己的网页。框架集可以定义任意多行和列来分割它的显示区,每个格是一个目标(Target)。框架集中的目标是其他客户端页可以为之请求网页的命名框架。

五、小结

随着 Internet 的迅猛发展,Web 应用程序因其优势被越来越广泛应用。但目前开发 Web 应用程序的中心很少关注开发的过程。利用 UML 为 Web 应用程序建模,按照面向对象分析与设计的一般过程对系统进行需求分析可以有效的进行 Web 应用程序的设计。实践证明,将此方法应用于厦门大学研究生选课系统的分析、设计和建模过程,不但降低了系统开发的复杂性,还为开发人员提供交流的工具,同时提高了系统重用性和可维护性,提高了系统开发的效率。

参考文献

- [1] 冀振燕. UML 系统分析设计与应用案例. 人民邮电出版社, 2003
- [2] 郑怀强, 吕振肃. 基于 UML 的系统需求分析——E- motor 客户关系管理系统设计. 计算机应用研究, 2002 年第 12 期 114 页 - 115 页

(上接第 60 页)

得到签名信息,然后再用我给他的密钥可以验证此真彩图的真伪。

3 结论

该方法较好地结合了数字签名和数字水印技术各自的优

参考文献

- [1] 胡彦, 陈昭明. MATLAB 在数字水印中的应用[J]. 计算机工程, 29(7): 184 - 186
- [2] D. R. 斯廷森. 密码学 - 理论和实践[M]. 国防科学技术保密通信重点实验室, 1997
- [3] 汪小帆, 戴跃伟, 茅耀斌. 信息隐藏技术 - 方法与应用[M]. 机械工业出版社, 2001

势,实现了对多媒体信息双重保护的功能,提高了论证精度。不过所使用的 DSA 签名法仅提供数据签名,不提供数据加密功能,且该水印方案抵抗攻击的能力不是太强。进一步提高水印的抗攻击能力和改进 DSA 算法将是今后研究的方向。