

支持脚本嵌入的 Web workflows 的研究与实现

卢良进 郑 民 吴海洪 (义乌工商学院 现代教育技术处 浙江 义乌 322000)

摘要: 分析了预编译 workflow 与 Web 表单 workflow 模型的优缺点, 提出了两者相结合的支持在活动中嵌入解释性脚本的 Web workflow 模型, 实现了活动业务逻辑与 workflow 系统的分离, 使用户可以在不修改 workflow 基础设施的情况下, 实现对流程活动的用户界面、业务逻辑及活动流向的完全自定义。实践证明, 该模型具有较高的可扩展性和可重用性。

关键词: workflow; 脚本; Web; 表单; 用户自定义

Research and Implementation of Web Workflow System Supporting Embedded Script

LU Liang-Jin, ZHENG Min, WU Hai-Hong

(Department of Modern Education Technology, Yiwu Industrial & Commercial College, Yiwu 322000, China)

Abstract: This paper analyzes the merits and shortcomings of precompiled workflow and Web form-based workflow model. A Web-based workflow model supporting script embedded into activity is proposed, which combines the merits of precompiled with the form-based workflow. It separates the business logic of activity from the workflow engine, and supports the sole user-definition of interface, business logic and activity transition. Practice shows this model has better expandability and reusability.

Keywords: workflow; script; Web; form; user-defined

1 引言

近年来, workflow 系统得到了广泛应用^[1]。根据 workflow 活动的业务逻辑及界面是否允许在建模期和运行期更改, 主要分为两种类型: 一种是经过编译才能执行的工作流, 例如: 微软的 Windows Workflow Foundation^[2], 这种模式对 workflow 中活动的业务逻辑和界面进行硬编码, 只支持在运行期进行活动重组, 当增减活动或修改了活动的业务逻辑及执行界面后, 必须重新编译系统。然而在实际环境中, 变更时常发生, 这给开发人员和用户都带来很大麻烦; 另一种是以 Web 表单作为 workflow 的执行界面^[3,4], 由于 HTML 本身的解释性和灵活性特征, 使得用户不仅可以动态的增减活动, 而且可以自定义流程活动的界面, 这种模式的缺点在于表单只能被动的接受和保存用户的输入数据, 却不能对用户提交的数据加以适当的处理, 因此只适用于公文流转类 workflow。

本文结合两种模式的优点, 通过引入解释性脚本

引擎, 使用户不需要经过编译就可以在流程活动中嵌入脚本, 把 workflow 建模期的部分工作转移到运行期, 使活动和活动转移关系两者在建模期和运行期都可以动态修改, 从而大大提高了系统的灵活性和可扩展性。

2 系统模型

系统采用基于 B/S 模式的三层架构, 分别是展现层、业务层和数据层。如图 1 所示。

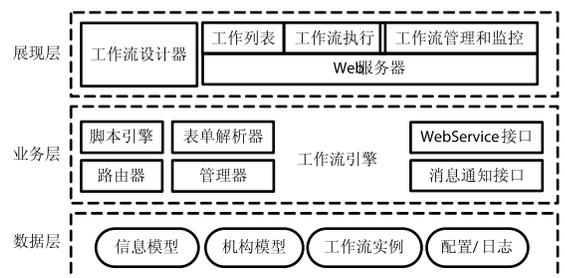


图 1 系统体系结构示意图

(1) 展现层

展现层是用户与系统的接口。 workflow 设计器用于定义 workflow 的模板,对 workflow 包含的活动及活动间的转移关系进行设置与维护。 workflow 列表列出了当前用户需要执行的流程活动; workflow 执行模块用于活动执行时与用户用户进行交互,包括执行界面显示、数据输入和保存,脚本的执行,以及活动的跳转; workflow 管理监控模块用于维护组织机构信息、基于角色的权限分配及 workflow 创建、激活、挂起和终止等操作。

(2) 业务层

业务层的核心是 workflow 引擎,包括脚本引擎、表单解析器、路由器和管理器等组成,脚本引擎负责嵌入流程活动中脚本的解释执行;表单解析器负责流程执行是 Web 表单的显示、提交数据保存、外部应用的调用等工作;路由器负责流程活动的条件跳转;而管理器负责流程实例的创建和状态转换等工作。

(3) 数据层

数据层为业务层提供数据服务。其中信息模型包含了 workflow 和流程活动的定义;机构模型定义了部门、角色、用户和权限等信息; workflow 实例库用于保存 workflow 运行期的相关数据;数据层还包括流程执行日志和配置等信息。

3 系统实现

3.1 工作流信息模型

一个 workflow 定义为一个四元组 $W=(A,T,P,s)$,其中 A 是一个非空的流程活动集; $T=\{(a_i,a_j|a_i,a_j \in A)\}$ 是流程活动的转移关系集合, (A,T) 代表一个有向图;流程参数集 P 用于保存 workflow 实例的相关属性,由名称值对组成; s 是 workflow 起始活动。

workflow 活动 a_i 具有两个相关属性: WebForm 和 Script,其中 WebForm 就当前活动对应的表单内容,它定义了活动的执行界面和与用户交互的数据; Script 是表单提交后需要执行的脚本。

活动间关系 t_i 具有两个相关属性 Condition 和 DependFlag,其中 Condition 表示在什么条件下前驱活动 a_i 向后继活动 a_j 转移; DependFlag 是个布尔值,表示当某个后继活动有多个前驱时,后继对前驱有依赖关系,也就是说,必须在前驱活动完成后,其后继活动才能启动。

3.2 工作流定义

workflow 定义由 workflow 设计器完成,由于 Web 图形界面的可操作性不高,且 workflow 设计器一般由管理员使用,因此 workflow 设计器采用 C/S 模式实现。 workflow 定义主要完成以下几个方面的工作:

(1) 活动的添加和删除

由于流程活动与 workflow 系统分离,活动成为一种描述性信息。在图形操作界面上,用户可以自己绘制或删除一个活动,同时可以赋予活动一定的业务处理能力。

(2) 活动的界面定义和业务逻辑定义

用户通过输入活动的 WebForm 和 Script 属性来修改其界面定义和业务逻辑定义。 WebForm 属性完全兼容 HTML,用户可以通过其他方便的设计工具来设计表单,再将 HTML 代码填入 WebForm 属性中。添加新表单后,系统会自动提取输入框控件的名称,将该名称与当前活动的名称共同组成 workflow 参数名称,并加入到 workflow 定义的参数集 P 中。

在流程执行过程中,用户提交表单后, workflow 引擎会自动调用 Script 属性中定义的脚本来完成用户指定的业务功能。相对于编译型语言,脚本语言更为灵活,用户可以方便的修改脚本而不需重新编译。实现脚本解析引擎的方式主要有两种:一是利用现有的脚本引擎,如 Microsoft 的 ActiveX Script 引擎;二是自行开发解析引擎^[5]。自行开发脚本引擎需要花费较大的开发成本,但可以实现自己特定的需求。这里我们自主开发了一个轻量级的脚本引擎,其语法规则与 VBScript 类似,但对其功能进行了削减和扩展。主要支持以下几个方面的功能:

简单的算术、逻辑和赋值运算,字符串常规操作;

可以通过参数名称直接读取 workflow 实例的参数值。

支持对数据库的基本操作。在脚本中可以直接调用 DBQuery(sql)和 DBUpdate(sql)函数来查询和更新数据库,具体操作由 workflow 引擎来完成,用户不必关心数据库的连接建立和释放等问题。

支持 Webservice 的调用。通过调用 Webservice(wsdl, params)直接跟外部应用进行交互^[6]。其中参数 wsdl 和 params 分别是 Web 服务的描述文件 wsdl 文件的地址和调用时参数的名称值对字符串。

WebForm 属性也支持脚本,用户可以像 ASP/JSP 一样在 HTML 中嵌入脚本,通过脚本来控制表单内容与样式的上下文定制,例如:某种角色的用户不能对工作流的某个数据项作修改时,可通过脚本对其进行控制。用户在输入 WebForm 及 Script 属性时,可通过系统提供的测试功能来获取脚本的错误提示,以指导脚本的排错。

(3) 活动转移关系定义

根据 WfMC 对工作流模型的定义^[7],一个工作流引擎至少支持 6 种活动的流向控制:顺序、或分支、与分支、或连接、与连接、异或分支等。系统通过活动间多对多的条件关联来实现上述转移模式。在工作流设计器上,两个活动的连线可以定义 Condition 和 DependFlag 属性,Condition 属性可以是数值或表达式,系统根据 Condition 的计算结果来决定是否生成后续活动的实例。在具有多个后续活动的情况下,用户可以根据具体情况设计 Condition 来实现或分支、与分支及异或分支等转移关系,通过 DependFlag 的设置可以实现与连接和或连接等转移关系。

3.3 工作流执行

工作流的执行通常是外部条件触发的,系统根据工作流的定义生成工作流实例,并由工作流引擎启动工作流的起始活动。由工作流引擎对后续活动的执行进行引导。每个活动具有一个当前状态标识,活动的状态转换过程是:初始、激活、执行、挂起和完成等。状态切换过程如图 2 所示。

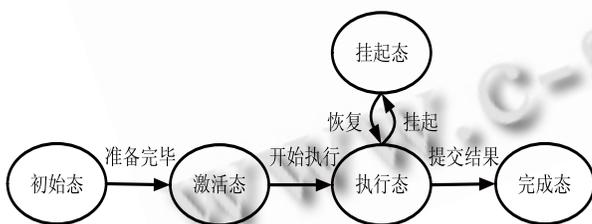


图 2 活动状态转换图

(1) 当前活动完成后,工作流引擎根据活动的转移关系决定创建当前活动的哪些后续活动,活动刚创建时处于初始状态。

(2) 由系统检查活动间关系的 DependFlag 属性,若具有依赖关系的前驱活动都已完成,则活动进入激活状态,根据权限定义将活动分配给某个用户,将活动加入到用户的工作列表中,并通过系统提醒接口(包括系

统内消息、手机短信或 Email 等方式)通知用户。

(3) 用户在工作列表中打开活动执行界面后,活动切换到执行状态,由工作流引擎的表单解析器负责将 WebForm 的内容呈现给用户。

(4) 用户提交表单后,工作流引擎将提交数据逐个填入工作流实例的对应参数值中,供后续过程引用,同时当前活动的状态切换到完成状态,重复前面的过程直到整个工作流结束。

(5) 工作流在执行过程中随时可能被挂起,挂起操作取消后恢复到执行态。

3.4 工作流管理和监控

如果在工作流执行过程中修改流程的定义,可能会引起数据的不一致甚至流程无法正常执行^[8],系统采用版本的概念来支持工作流定义的动态变更。在修改工作流定义后,若不想应用于当前已启动的工作流,则可将新的定义生成新的版本,在下次生成工作流实例时,自动启用新的工作流定义。

工作流在执行过程中可能出现某个活动不存在执行人员(如某个员工离职)或某个活动不能按时完成而影响整个流程的完成时间的情况。为提高系统的灵活性,系统设置了超时和外出管理策略,在某个流程活动激活后,若没有执行人员则立即告警,或者在超时时间后未执行,系统也会自动向管理员告警。用户外出或无法及时处理工作流时,可将自己设为外出状态,系统在选择执行用户时不会将其考虑在内。为支持流程的重组优化,系统会记录每个活动从初始到完成所需要的时间,管理员可通过对比找出效率低下环节,从而进行改进。

4 结语

工作流的运行过程是一个不断完善的过程,本文提出的支持脚本嵌入的工作流系统继承了预编译工作流和基于表单的工作流系统的优点,使活动执行界面、活动业务逻辑及活动流向都可以在建模期和运行期动态修改,使系统具有更好的可扩展性和可配置性。该系统目前应用于某印刷企业的订单处理流程中,系统展示了良好的方便性和适应性。但现在脚本的功能还只局限在基本的计算语句、数据库查询更新和 WebService 调用语句中,在进一步的研究工作中,我们将丰富脚本功能,并使之在事务、安全性上提供更好的支持。

(下转第 196 页)

参考文献

- 1 范玉顺. workflow 管理技术基础. 北京: 清华大学出版社, 2001. 5 - 22.
- 2 Cubo J, Salaun G, Canal C. Relating Model-Based Adaptation and Implementation Platforms: A Case Study with WF/.NET 3.0. Proceedings of 12th International Workshop on Component-Oriented Programming(WCOP), 2007. 9 - 13.
- 3 张洪山, 殷人昆, 张素琴. 基于 WEB 的工作流引擎设计. 计算机工程, 2004, 30(4): 87 - 89.
- 4 路春光, 孟丽丽, 等. 基于 WEB 的柔性工作流引擎的设计. 微计算机信息, 2006, (5): 21 - 25.
- 5 Li N, Ma Z. The Application of Script Language in Advanced Control Platform. Proc. of the 25th Chinese Control Conference, 2006. 1996 - 2000.
- 6 Leitner P, Rosenberg F, Dustdar S. Daios: Efficient Dynamic Web Service Invocation. IEEE Internet Computing, 2009, 13(3): 72 - 80.
- 7 Hollingsworth D. Workflow Management Coalition. The Workflow Reference Model. <http://www.wfmc.org/standards/docs/tc003v11.pdf>. 1995.
- 8 Aalst W, JABLONSKI S. Dealing with workflow change: Identification of issues and solutions. International Journal of Computer System, Science and Engineering, 2000, 15(5): 267 - 276.