轻量级工作流管理系统架构设计

丁苍峰
(延安大学 计算中心,陕西 延安 716000)

摘 要:为了克服传统工作流管理系统的缺点,满足企业流程的自动化和提高企业的市场竞争力,分析了当前工作流管理系统普遍存在的问题和轻量级工作流管理系统架构设计的必要性;搭建了由客户层、表示逻辑层、持久层和数据库服务层组成的多层次工作流管理系统的整体架构,设计了可以根据需要灵活添加和裁减组件的轻量级工作流管理系统核心架构,详细地描述了该架构各个组件及功能,并且基于该架构的系统可以根据具体需求灵活地定制不同领域的工作流管理系统中。

关键词:架构;轻量级;工作流管理系统;组件
中图分类号:TP303 **文献标识码**:A **文章编号**:1673-629X(2011)06-0036-04

Design of Lightweight Workflow Management System Framework

DING Cang-feng
(Compute Center of Yan'an University, Yan'an 716000, China)

Abstract: In order to overcome the deficiency of traditional workflow management system and meet enterprise process automation and enhance competitive advantage of enterprise in the market, analyse current workflow management system which commonly exists problems and it's necessary to design lightweight system framework, builds overall architecture which is composed of multilayer of lightweight workflow management, design core framework of lightweight workflow management, also elaborate various components and functions of framework. According to specific needs, the core framework allows flexible addition and reduction of various components to expand the workflow management function, also flexibly customs to different areas of workflow management system.

Key words: framework; lightweight; workflow management system; component

0 引 言

工作流从提出至今,经过几十年的发展,取得一定的成果,但在实际应用中,应用的企业还是较少,应用的范围窄,效果不理想[1]。这主要是目前与工作流管理系统的全面性、集成性有关,也与企业应用的状况有关,最重要的原因是目前工作流产品功能齐全,结构庞大,系统复杂,标准化程度差,灵活性差,实现复杂,难以在不同系统之间进行交互,集成的效果不理想,不能很好将SCM、CRM、ERP和OA等具体业务流程处理系统紧密地联系在一起[2],多数产品的开发由于没有清楚地理解用户的需求,而不能满足用户的迫切需要,也不能灵活地集成到企业具体系统中。

针对目前工作流产品涌现的诸多问题,文中提出一个轻量级工作流管理系统架构,并利用该架构来搭建工作流管理系统整体架构 miniFlow,可以说,文中所设计的轻量级工作流管理系统架构,为开发者在以后开发工作流产品时指出了一个新的捷径。工作流管理系统 miniFlow 以组件为基础而设计,以最小功能集合为目的,从而可以方便集成到其它业务系统中,可以灵活地根据需要添减组件。并且,miniFlow 采用整合的 MVC 架构,采用 B/S[3] 架构实现,更好地满足企业应用集成和电子业务应用的需求,提升了企业效率,提高了企业的市场竞争力。

1 轻量级系统架构设计思想

当前工作流管理系统存在一些问题,不能根据用户的实际需求定制具体的应用开发,也不能柔韧性地集成到具体业务系统中[4]。与现有的提供完善的工作流管理系统不同,文中提出了轻量级的工作流管理系统 miniFlow 的架构设计思想,有如下三种:(1)根据具体需要灵活地选择功能。(2)针对不同应用进行不同集成裁减。(3)利用组件灵活裁减的特性进行新的特征、功能添加和扩展。

"轻量级"是设计工作流管理系统 miniFlow 的核心基础,其目的就是只提供满足工作流管理系统概念

收稿日期:2010-11-18;**修回日期**:2011-03-21
基金项目:陕西省教学改革项目(09BY37)
作者简介:丁苍峰(1978-),男,河南唐河人,硕士,讲师,CCF 会员,研究方向为工作流、嵌入式技术。

所要求的最小功能集合[5],在核心功能的基础上,利用组件"可插可拔"的特点,在针对具体的业务需求时可以灵活地扩展工作流管理系统的其它功能,从而实现了既轻量级又灵活的设计特性。其实,工作流管理的轻量级就是从用灵活、最小的成本和仅满足具体需要的设计原则出发,不去追究目前工作流产品既复杂又难集成的思想,仅仅实现了工作流管理系统中必不可少的功能及特点,设计 miniFlow 系统时就仅仅设想了如何定义业务流程、如何解析业务流程、如何实现业务流程的实例化、如何分配业务流程的任务以及如何控制业务流程的流转等功能。在 R. Tagg 的论文"轻量级工作流服务器的初步设想"中使用了"轻量级"这一概念[6],但是他的论文叙述的则重点是怎样构造一个"瘦客户端"。而文中设计的轻量级工作流 miniFlow 系统架构侧重点就是充分支持了工作流技术的小型内核,可以耦合到具体不同应用的 MIS 系统中,达到灵活应用的目的。

2　整合 MVC 的轻量级的 miniFlow 设计

miniFlow 是基于整合 MVC 架构的 Web 环境的系统,整个系统参照了 J2EE[7] 架构,采用了 Brower/ActionServlet 控制器/BO 业务逻辑层/Hibernate DAO 层/DBMS 数据库服务层多层的框架结构。轻量级是设计 miniFlow 系统所一直围绕的核心任务,为此,利用目前很流行的 Struts2[8]、Hibernate 开源框架[9]和 Ajax[10]异步请求/响应技术来设计,主要采用了组件能拔能插的特性来容易地扩展工作流管理系统的其它功能。文中利用这些技术设计的 miniFlow 系统整体架构如图 1 所示。

图 1 清楚地描述了 MVC 架构基础上的工作流管理系统和应用进行集成的多层模型,分为客户端应用层、表示逻辑层、业务逻辑层、持久层和数据库服务层。

1. 数据库服务层为工作流管理系统和业务系统通过持久层提供数据。

2. 持久层用 Hibernate 进行数据库的完全封装,使软件开发者真正对数据库的操作是面向对象的操作,完全实现了面向对象的设计思想,并且在该层进行了对通用公共方法的封装,使整个系统都可以调用该公共的方法。在进行对方法封装时,也设置了开关控制等一些操作,不仅提高了整个开发系统的效率,大大节约了项目的成本,也使系统开发的代码易于维护和便于以后的升级。

3. 业务逻辑层实现各种业务逻辑功能,是 miniFlow 系统的核心层。主要由工作流管理系统和企业具体业务系统(MIS、ERP 和 SCM 等)两部分组成。工作流引擎是工作流管理系统的核心,其中引擎是通过引擎调度中心进行控制,该中心可以根据业务逻辑对工作流引擎进行调用所对应的组件,组件分别由转发控制组件、任务指派组件、引擎调度规则组件等一些可以由调度中心进行调度的组件构成;引擎是驱动流程流动的主要部件,工作流引擎主要用来负责解释工作流流程定义、创建并初始化流程实例、控制流程流动的路径、记录流程运行状态等工作。

4. 用户客户端主要由客户层和逻辑表现层提供,不同用户因角色不同而具有不同的权限,从而在客户端显示不同的页面操作信息,这些页面信息是由 Struts

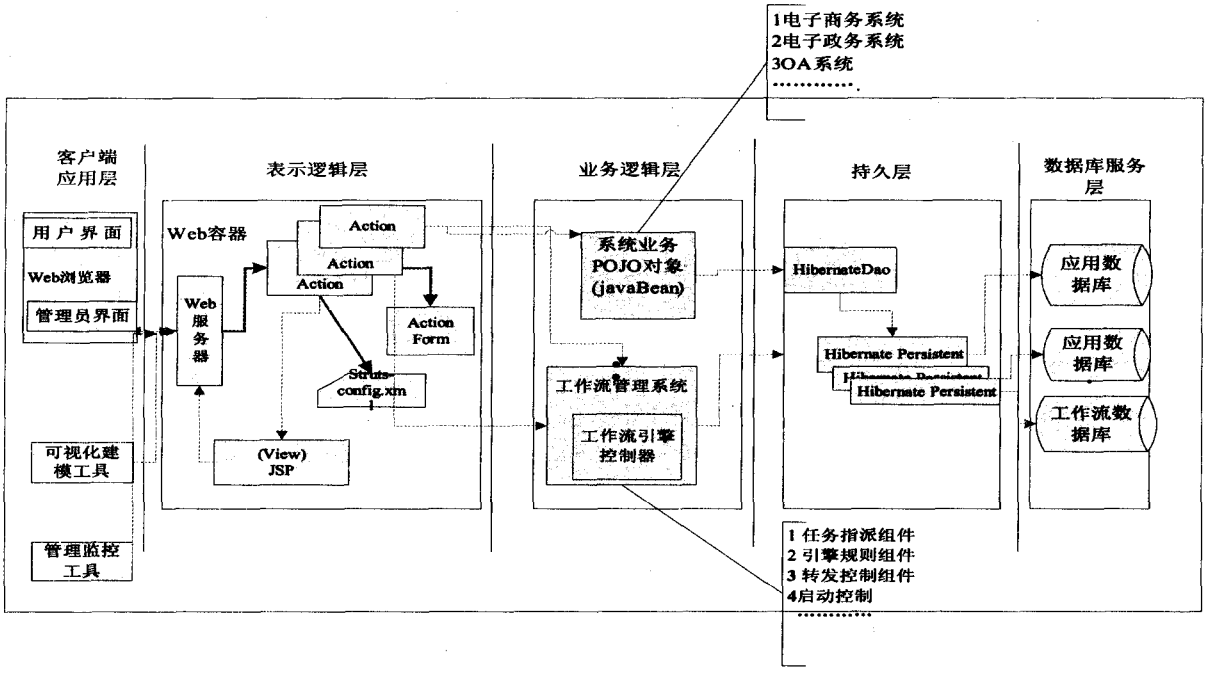

图 1　miniFlow 系统整体架构

的标签显示的,JSP 教本在 Struts 的 ActionServlet 控制器下通过 WEB 容器的 ActionForm 表单和业务逻辑层的工作流管理系统组件和系统业务组件进行交互工作,根据业务用户的不同需求来调用相应的工作流管理系统组件。

3 miniFlow 系统核心架构设计

目前几乎所有的工作流产品都是重量级的,从而导致所采用的架构也是重量级的。通过一些研究和总结发现,目前的工作流管理系统中的大多功能几乎不属于 Workflow Management System(WFMS)[11] 定义的概念范畴,从而导致了工作流管理系统的功能很难集成、定制和重构到其它业务系统中。在 Claus Johanners 的"一个通用的内核对可靠过程的支持"论文中论证了传统工作流管理系统架构设计的狭隘性[12]。文中在设计 miniFlow 系统架构时不要求实现工作流的所有功能,而仅是实现工作流管理系统的最小功能集合。目前大多数的企业的业务流程几乎都是各类行政申请流程、通知流程、财务审批流程、人事处理流程及客服流程等,这些流程一般都有严格的环节,不是随机化产生的,只要不背离这些过程化的流程环节就可以完整地完成整个业务。文中设计的轻量级的 miniFlow 系统架构就设计了满足以上所需的业务流程,不追求工作流管理系统功能的完备和复杂,以满足一般性业务为目的,为企事业快速开发出适合他们业务的工作流管理系统。工作流管理系统最小的功能集合仅仅提供的是解析业务流程定义和控制流传业务流程。文中设计轻量级 miniFlow 架构的核心就是提供了这两个基本的功能,从而可以使得 miniFlow 达到了符合轻量级的工作流管理系统的要求。miniFlow 系统核心架构如图 2 所示。

1. miniFlow 核心架构 miniFlow 架构核心由工作流引擎组件(包括流程解析和定义组件、流程转发控制组件、流程节点持久化组件及流程规则组件)构成。

2. miniFlow 系统引擎核心 在 miniFlow 架构核心的集成上,添加上权限分配和任务指派组件,从而构成了轻量级工作流管理系统 miniFlow 的引擎核心。

3. miniFlow 扩展组件 为了具有可视化流程定义工具和记载流程执行轨迹日志信息,为此扩展可视化的流程定义工具、任务列表和日志信息组件等。

4 miniFlow 架构核心组件

轻量级的 miniFlow 架构核心部分是流程定义和解析组件、控制转发组件、引擎规则组件的设计,引擎的核心是在 miniFlow 架构核心的基础增加了任务指派组件和权限分配组件,课题设计的这些组件提供了工作流运行的最基本功能。

为此确定了 miniFlow 引擎的设计原则:

1. 与业务弱耦合、可分离的原则 即业务模块不完全依赖工作流引擎驱动。

2. 基于关系数据、轻量级的原则 和传统的工作流引擎相比,本课题的工作流引擎设计不追究直接业务数据组态等重量级功能,达到既能满足业务流传需求,又能快速提供任务消息即可。

3. 软件模块平台同一的原则 采用新的 Ajax 技术,使用整合 Struts 和 Hibernate 的 MVC 设计模式思想,工作流建模程序以 B/S 方式提供给客户使用。

4.1 持久化的业务流程和活动

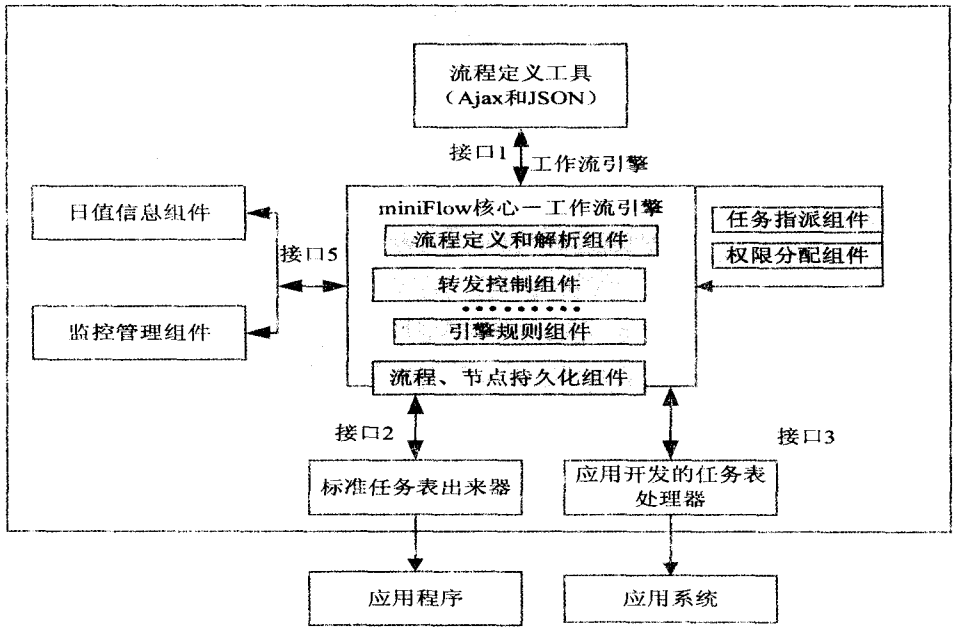

图 2 miniFlow 体系核心架构

工作流管理系统的可视化流程定义工具所定义的业务流程其实仅仅是逻辑上的业务流程,只是对具体业务流程的抽象化操作,并没有实例化到具体的业务流程上,也就是说通过可视化流程定义的业务流程只是抽象的,并不是实例,为此,要实现从抽象的流程到实例流程上来,需要对流程进行持久化的操作。由于可视化流程定义工具定义的每个流程是由若干逻

辑节点、连线及节点之间的关系构成进行描述的,对流程进行实例化时要给每个逻辑节点分配具体的任务、指派给具体的用户,流程活动持久化组件为逻辑流程的持久化、解析和流转控制提供了基础支持。这些功能由 miniFlow 核心架构中定义的流程、节点持久化组件来完成。

4.2 定义、解析和转发控制流程

工作流引擎作为工作流管理系统的核心部分,为由可视化工具定义的业务流程提供了解析和路径选择的解决方案。miniFlow 系统架构的引擎核心的一些组件就提供了一个流程定义的版本管理。在具体业务流程的流转时,部署在 miniFlow 架构的引擎控制器由调度中心根据启动动作来启动相应的业务流程。在流程的流转过程中,应用程序并不清楚流程的下一个任务是谁,它只需要完成当前任务,引擎控制器的调度中心会自动调用引擎规则组件就可以根据当前任务的流转给下一个任务,从而完整解析和流程转发功能,其对应工作如下:

1. 业务流程定义的解析工作。
2. 流程实例的创建和初始化工作。
3. 业务流程路径选择工作。
4. 待完成任务的任务指派工作。
5. 记录和跟踪具体业务流程实例的日志信息,进行实例的查看、统计等工作。

4.3 组件可扩展性

为了满足用户的需求,大部分开发商或企业开发出具有上百个可选功能的工作流产品,然而这些功能都是以一种集成的方式提供的,针对具体用户需求时难以按需定制。而当前的工作流产品一般设计的架构都是重量级的且功能之多,难以集成和应用,从而理解整个架构非常困难,也使系统难以重用、定制和扩展。轻量级的引擎是 miniFlow 系统的核心,利用组件灵活的可插可拔的特点扩展了 miniFlow 的内建功能。扩展的 miniFlow 组件如下:

1. 流程定义工具提供可视化的界面。流程定义工具提供可视化的逻辑业务流程操作界面,方便流程定义管理员用户根据不同的业务需要来定义不同的业务流程,达到流程定义的目的,且流程定义工具是独立于 miniFlow 系统核心部分的扩展组件。

2. 工作列表。待处理工作表示当前用户要等待用户处理的工作,其任务状态为 WAITING,等待用户来处理;已提交工作表示当前用户处理了等待用户处理的工作,在用户处理完后就表示已提交工作了,其任务状态为除 WAITING 外的其它状态(RUNNING、CANCEL 和 SUSPENDED 等);已完成工作:表示当前工作已经处理完毕或由于别的原因取消工作的执行,其任务状态为 COMPLETED 或 FAILURE 或 CANCEL 等。

3. 日志信息。工作流管理系统跟踪记录业务流程中任务流转操作或状态。日志信息的组件是独立于 miniFlow 架构的核心部分,仅是架构的扩展部分。

这样设计轻量级的 miniFlow 架构,很容易被理解,也容易根据具体的需要而集成在不同的业务中。

5 结束语

文中分析了当前工作流管理系统普遍存在的问题,设计了轻量级工作流管理系统架构,miniFlow 架构主要由工作流引擎核心组件和扩展组件如流程定义工具、权限控制和任务指派、日志信息等组件组成。采用 miniFlow 架构进行开发的信息系统,可以提高企业运营效率、改善企业资源利用、提高企业运作的灵活性和适应性,为企业在激烈的市场竞争中赢得了更多的时间和机会,实践证明是切实可行的。

参考文献:

[1] 伍庆林,王加阳. 基于 J2EE 框架的工作流管理系统设计与实现[J]. 计算机技术与发展,2008,18(4):107-110.

[2] 牟　军,吕　立. 使用轻量级框架进行 J2EE 应用开发[J]. 小型微型计算机系统,2006,27(6):1149-1152.

[3] 姜晨临,蔡鸿明. 基于 Web 的分布式工作流系统的研究[J]. 计算机技术与发展,2008,18(6):43-46.

[4] Alonso G, Agrawal D, Abbadi E A, et al. Functionality and limitations of current workflow management systems [EB/OL]. 1997. http://www. almaden. ibm. com/cs/exotica/wfm-sys. ps.

[5] 何清法,李国杰,焦丽梅,等. 基于关系结构的轻量级工作流引擎[J]. 计算机研究与发展, 2001,38(2):129-137.

[6] Tagg R. Preliminary Design of a Lightweight Workflow Server [C]// In: 8th Australasian Conf on Information Systems. Australia:[s. n.], 1997.

[7] 王卫平,王松涛,王名茗. 一种基于 J2EE、Spring 和 Hibernate 的轻量级 EAI 构架[J]. 计算机系统应用,2005(11):38-41.

[8] 何成万,于秋惠. MVC 模型及软件框架 Struts 的研究[J]. 计算机工程,2002,28(6):271-274.

[9] 孙卫琴. 精通 Hibernate:Java 对象持久化技术详解[M]. 北京:电子工业出版社,2005.

[10] Asleson R, Schutt N T. Foundation of Ajax[M]. 金　陵,译. 北京:人民邮电出版社,2006.

[11] Bussler C. Enterprise-wide workflow management[J]. IEEE Concurrency,1999,7(3):32-43.

[12] Hagen C J. A Generic Kernel for Reliable Process Support [D]. Zurich, Switzerland: Swiss Federal Institute of Technology,2004.