

基于关系模式和 Web 的工作流管理系统的研究

董铁城, 何彪

(华南理工大学 计算机科学与工程学院, 广东 广州 510640)

摘要:随着信息化发展的不断深入, 传统的信息管理系统已经不能很好地满足企业的需求。传统的信息管理系统是面向功能的, 在企业的业务发生变化时很难快速地做出调整。而工作流管理系统则是面向流程的, 可以针对企业的流程做出相应调整。讨论了工作流管理系统对比传统信息管理系统所具有的优势, 在此基础上提出了一个基于关系模式, 面向 Web 的工作流管理系统的设计方案。给出了系统的设计思想和总体框架。并重点描述了工作流管理系统的数据库设计和系统实现的一些关键问题。

关键词:工作流引擎; 工作流模型; 活动; 任务; 关系模式

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2008)05-0040-04

Research of Workflow Management System Based on Relational Schema and Web

DONG Tie-cheng, HE Biao

(Computer Science and Engineering College, South China Univ. of Tech., Guangzhou 510640, China)

Abstract: With the development of the informationization, traditional management information system can't meet the need of the company. Traditional management information system is object to function and the workflow management system is object to process, where the process changes the workflow system can suit the process easily but the other can't. Talks about the advantage of workflow management system to the traditional management information system and presents a workflow management system which is on top of the Web and based on relational schema. The design philosophy and general framework are given. Also emphasize the design of the database and the critical point of the system.

Key words: workflow engine; workflow model; activity; task; relational model

0 引言

工作流技术起源 20 世纪 70 年代的办公自动化, 但是由于当时计算机硬件和网络水平的限制没有得到广泛的应用。进入 20 世纪 90 年代硬件和网络条件已经非常成熟, 工作流技术也得到迅速的发展。1993 年 8 月, 工作流技术标准化的工业组织——工作流管理联盟(WFMC)成立, 制定了一系列有关工作流的框架和标准, 标志着工作流技术进入了一个相对成熟的阶段。

现在工作流技术已不仅仅局限于企业中的公文流转、表格的审批等简单的流程, 而是应用于许多的企业的关键业务中。所谓企业的关键业务是指企业赖以生存的业务流程, 如保险公司的关键业务就包括保险索

赔业务; 贸易公司的主要业务就包括货物的买卖过程等。采用工作流技术来实施企业的信息化, 构建企业的信息系统, 可以提高企业的工作效率, 提高企业的管理水平。目前企业内部的业务流程繁多而且复杂, 工作流技术因为其灵活性和适应性受到很多企业的青睐。工作流技术已经成为构建企业信息系统的的发展趋势。

1 工作流技术和传统信息系统的一些讨论

1.1 工作流技术与传统信息系统的比较

从业务的角度来看, 企业或者政府部门的组成元素就是业务过程。工作流就是业务过程的抽象表示, 过程包含一组活动(Activity)和这些活动之间的关系, 也就是这些活动之间的先后执行顺序和依赖关系^[1]。用计算机可以理解的方式对业务过程进行建模就得到业务过程的工作流模型。在业务的概念上引入案例(Case), 案例就是业务流程的一次具体的执行, 同一个

收稿日期: 2007-08-24

作者简介: 董铁城(1982-), 男, 河北沧州人, 硕士研究生, 研究方向为工作流技术、网络软件编程环境; 何彪, 副教授, 主要从事网络软件编程环境的研究。

业务流程可能拥有多个具有不同结果的流程。

任务是一种操作或者变换,是组成业务流程的基本单位,只有确定任务才可以构建工作流。任务具有原子性,即一个任务必须完全执行,如果不能完全执行,则必须回滚(rollback)。任务是对于业务过程来说的,在案例中一个任务具体的执行叫做活动。

总的来说工作流管理信息系统和工作流技术的区别在于传统管理信息系统是面向功能的,而工作流管理系统是面向流程的。正如文献[2]中所说,工作流技术是把业务流程作为计算模型的技术。传统的管理信息系统不能对企业的业务流程进行管理或者是把一些简单的流程硬编码到系统中。这样,一旦企业的业务流程发生改变,则信息系统也要随着进行大的变化甚至重新构建企业的信息系统。工作流管理系统能对企业的业务流程进行管理,把核心放在对企业流程的处理,如果业务流程发生改变,仅需要改变对业务流程的建模,而不是从根本上对信息系统进行修改。

从政府和企业信息化的发展趋势来看工作流技术也有很大的优点。当前电子政务的流程处理已经不限于单个部门之间,而是要考虑到上下级和兄弟部门之间的信息传递。而企业的信息化过程中,业务流程也从原来的企业内部扩展到两个企业甚至多个企业之间流转。对于传统的信息系统来说,由于不同信息系统之间功能和接口的差异很难完成上述功能。但是工作流系统是把流程作为处理对象,只要给出相应的流程定义,则很容易把流程扩展到不同的部门或是企业。

1.2 选择通用工作流产品还是自主开发工作流系统

目前的软件产品都朝着大型化和复杂化的方向发展。目前市场上也存在很多通用的工作流产品,这些产品集中了企业应用中通用的一些功能,而且一般说来功能多、体系庞大。对于一个特定的过程这些通用软件不能直接应用,必须设定它的参数。使用通用工作流系统的优势在于减少信息系统的开发时间,其缺点是通用的工作流管理系统不一定能满足用户一些特殊的需求,而且对于一些企业来说其业务过程可能比较简单,不需要这些工作流管理系统中的大部分功能。

许多通用的工作流管产品从数据存储到运行环境都有一套自己独立的结构,虽然有的厂商宣称产品可用于第三方应用集成,可是如果真正做到这些,系统开发人员要面对很大的困难。如果根据自身情况来开发工作流管理系统就不会存在上述问题,工作流系统和企业信息管理系统无缝集成,而且可以根据自身情况控制工作流引擎的规模,对于一些不需要的功能可以省略,一个轻量级的工作流引擎不会占用太多的开发时间。

2 工作流系统的框架及数据库设计

2.1 工作流系统总体架构

系统根据工作流管理联盟中的参考模型,并采用B/S三层架构开发。从上到下依次为表示层、业务逻辑层和数据存储层。系统开发采用J2EE技术,具有跨平台的特点,不依赖固定的硬件和操作系统。将工作流引擎通过实现其接口来部署在EJB容器中,使得不必考虑复杂的底层实现,如事务、线程、分布式等,该系统的架构如图1所示。

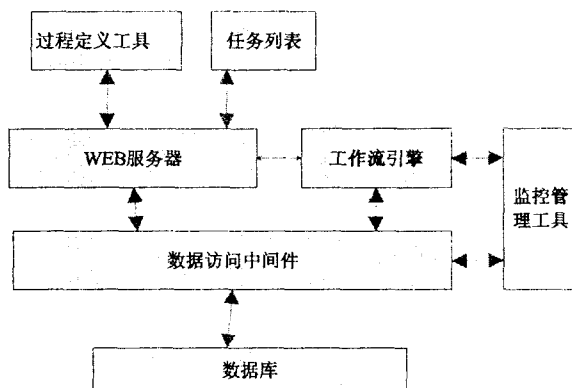


图1 工作流管理系统体系结构图

表示层提供了图形化的流程定义工具和用户任务列表,用户可以通过Web浏览器与工作流系统交互,不需要安装特定的客户端工具,这样增加了系统的灵活性同时也易于维护。业务逻辑层是系统的核心,工作流系统的大部分功能都在这一层实现了,包括推动整个工作流运行的工作流引擎和工作流的管理和监控工具。业务逻辑层通过数据访问中间件对数据存储层进行访问。工作流管理器是业务逻辑层的核心,它负责对工作流模型进行解析和初始化,对工作流案例进行监控,并对工作流状态和数据进行保持和维护。工作流引擎根据工作流模型的定义负责活动的跳转,推动案例的发展。数据存储层存储工作流系统的机构信息、控制信息、工作日志信息和应用数据。

2.2 数据库模型

2.2.1 为什么使用关系数据库

关系型的工作流管理系统的数据库模型全部通过关系结构来表达^[3],控制工作流引擎的各种逻辑程序也是通过关系数据库管理系统中的存储过程以及触发器来实现,同时事物的并发控制也通过数据库系统所提供的机制来实现。还有就是各种工作流信息的完整性和依赖性也可以通过工作流关系系统中的数据的完整性和依赖性来实现。

从技术角度来说,使用关系结构来表达工作流引擎中的数据模型可以降低工作流引擎开发过程中的技术难度和工作量。具体表现在:

(1)与 workflow 引擎相关的各种控制数据(包括业务活动的状态数据)可以存储在数据库系统中;

(2)与此相关的数据的完整性可以由数据库管理系统来维护;

(3)利用关系结构可以方便地定义 workflow 引擎中的各种数据格式和数据结构;

(4)可以方便地利用数据库管理系统提供的各种 DML 语句来操纵 workflow 引擎所需的各种数据。

2.2.2 企业资源模型和 workflow 信息模型

关系模式包括企业资源模型和 workflow 信息模型。在数据库中都是以表的形式存在。由于篇幅关系,下面只给出数据库中核心表的 ER 图,如图 2 所示。

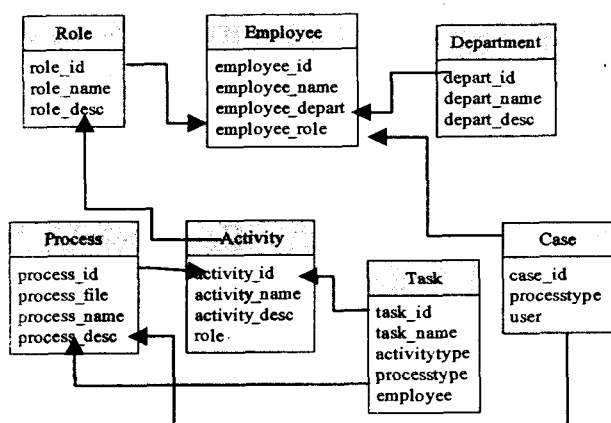


图 2 数据库 ER 图

企业内部的职员(Staff)、打印机都是资源。一组相同的资源又属于一个资源类,一个任务通常只需要一种资源类当中的资源来完成即可。

Role 表存储了根据职能特征划分出的资源类称为角色,它涉及到职能和资格。一个角色是一组具备一些明确技能的资源。如 Administrator(管理员), Programmer(程序员)等。

也可以按照资源在组织中的位置进行划分,按照这种定义,可以得到如下的资源类, Sales- Department(销售部)、Purchasing- Department(采购部)等。这些划分存储在 Department 表中。

按照上述分类对企业资源建模,表示成数据库中的表则有下列的表: Staff 表、Role 表、Department 表。对于小型机构而言这三张表已经可以,对于一些大的机构,可能会组建一些临时的团队(Team),这样属于一个部门的表有可能属于另一个团队,所以还要增加一个 Team 表。

workflow 信息主要包括:流程定义、活动、案例、任务。在 workflow 信息数据中存在以下表格: Process, Case, Task, Activity 表。

其核心数据是 Activity,因为其他的信息都要与活

动表格联系起来。Activity 存储在数据库中只能是一个编号或者地址,没有任何实际意义,所以活动必须同实体联系起来。实体可能是一个文件或者是符合某种规范的类。

Process 表存储流程的定义,每个流程都有唯一的 ID。完整的流程定义在数据库中表达也是非常困难的,因为除了流程中各个活动之外,还要保存活动之间的关系如并列、会聚等。这种关系的表达在数据库中难于表达出来。可以用 XML 语言来表达流程定义,并在 Process 表中存储 XML 流程定义文件的地址。

Case 表存储了流程的一次执行过程。员工每启动一次流程则表达案例的一次增加,每个案例都属于一种流程,而且有唯一的 ID,用来区分同种流程的案例。workflow 引擎在案例的执行过程中根据活动产生相应的任务,存储在任务列表中。同一案例中可能出现相同类型的任务,它们也通过唯一的 ID 区分。

Task 表存储分派给每个员工的任务,如果一个活动需要人工参与则在表中加入一个分派给某个员工的任务项。

2.3 workflow 系统的应用过程

从与 workflow 交互的几类工作人员的角色说明 workflow 系统的运行情况。workflow 系统中存在下面角色:

(1) workflow 设计者。workflow 设计者使用过程定义工具对 workflow 进行建模。

(2) 管理员。管理员使用运行管理工具。对 workflow 运行进行管理,例如增加员工信息,授权,对 workflow 设计者定义的工作流模型进行激活等。

(3) 雇员。雇员是 workflow 系统的真正使用者,他们通过任务列表与 workflow 交互完成需要人工参与的活动。每个雇员都有自己的任务列表。

首先是过程定义人员通过图形化的建模工具对业务流程进行建模,每个活动是一个 XML 元素,用 XPDL 文件格式保存流程定义。然后是管理人员对 workflow 模型进行激活并授权,这样相应的雇员就可以在任务列表中启动一个 workflow。雇员通过任务列表即可启动一个 workflow,workflow 管理系统对 workflow 模型进行解析并实例化之后把控制权交给 workflow 引擎,workflow 引擎推动流程发展。如果活动需要人工参与,则在相应雇员的任务列表中产生一个任务,雇员通过完成任务来参与到 workflow 当中。如图 3 所示。

3 workflow 管理系统的核心技术

3.1 过程定义工具

过程定义语言就是把业务过程表达成计算机可以理解的模型。WfMC 制定了一系列的标准,其中

XPDL 语言应用最广泛,许多 workflow 引擎支持的 workflow 模型都是使用经过一些改动的 XPDL 语言定义的。XPDL 是基于 XML 语言的,具有结构化的特点,可以通过嵌套的形式把复杂的业务过程表示出来。workflow 中的业务节点通过元素来表示,元素的属性就表示业务节点的信息。如:

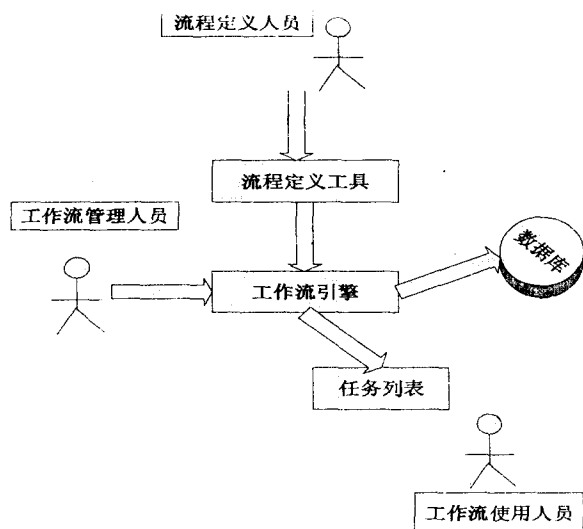


图3 工作流管理系统的用户

```

<ACTIVITY name=activity1 type=parallel>
<ACTIVITY name= activity2 property1=property2= />
<ACTIVITY name= activity3 property1= property2= />
</ACTIVITY>
  
```

上述语言片断就表达了一个并行活动 activity1 包含两个活动 activity2 和 activity3,并且表达了每个活动的属性。

3.2 工作流引擎的调度算法

工作流引擎在工作流系统中的位置相当于操作系统中的内核,工作流引擎的调度算法直接影响到了工作流系统的运行的稳定性和速度。活动是工作流引擎执行的实体。

堆栈方式:在初试化任何一个活动以后,就把这个活动放在堆栈之中,然后遍历堆栈之中的活动,进行执行。这种方式的调度其实把一些判断机制交到外围过程。引擎只是要获得一个活动实体然后激活它即可。用户启动一个 workflow 案例,进行初试化,把工作流模型中的第一个活动实例化之后放到堆栈中。引擎执行完一个活动之后,会实例化下一个活动,进行执行,这样引擎总是优先执行当前的 workflow 案例,只有在当前 workflow 的活动没办法很快完成(比如需要人工参与,或是等待其他资源)的情况下,才会挂起当前的案例,选择下个案例进行执行。

3.3 活动的表示方式

上文已经提及活动在数据库中的存储只是一个编

号,如果没有跟实体联系起来是没有任何意义的^[4]。实体是 workflow 管理系统执行的基本单位。活动的表示是 workflow 管理系统架构中的首要问题。传统的的实体也称为元 workflow,元 workflow 与具体的业务流程无关,它是从功能中抽象出来的供 workflow 管理系统调用的模块。表现形式为信息系统中的一些功能性模块。

在此 workflow 管理系统中,采用了类来表示活动。定义一个 Activity 抽象类,Activity 类定义了活动所需要基本的数据和方法。所有的实体类都从此类继承。workflow 引擎在确定要执行哪个活动之后,根据活动 ID 查找到相应的实体类并实例化此类的一个对象。通过对象的执行方法即可获得完成该活动的执行。

3.4 人工活动与工作流引擎之间的信息交换

workflow 在推进整个 workflow 定义的过程中,必须能够为前后两个活动传递一定的信息^[5]。这些信息可能是前个活动的处理结构或者是后个活动所必须的参数。可以采用 XML 文件来传递这些参数。例如有一个员工查找的活动,它的主要功能是根据引擎给出的条件查找适合的员工。这些活动需要人工的参与进行。当引擎要查找一个程序员,年龄在 30 岁以下时,它给出查找的条件:

```
<Person role=programmer age= 30></Person>
```

则活动经过查找返回结果给工作流引擎的 XML 文件为:

```
<Person role= programmer age= 30>
```

```
<Result> Andy</Result>
```

```
<Result> Ken</Result>
```

```
</Person>
```

上述文件说明活动找到两个符合要求的员工 Andy 和 Ken。

4 结束语

随着 WIMC 的成立,标志着 workflow 管理系统进入了一个相对成熟的阶段,但是目前市场上的各种 workflow 产品还是没有有一个统一的标准。目前的主流产品都是基于关系数据库。这是因为关系型数据库是目前最成熟、应用最广泛的数据库。还有就是上文提到的关系型数据库对于 workflow 管理系统的一些独特优势。随着 Web 技术的发展,workflow 管理系统也倾向于采用 B/S 结构来替代传统的 C/S 结构。

文中提出的基于 Web 和关系型数据库的 workflow 管理系统遵循了 WIMC 提出的一系列标准,该系统具有跨平台性和可扩展性。该管理系统主要作为企业内部的管理信息系统,文中没有涉及到 B2B 的流程传递

(下转第 46 页)

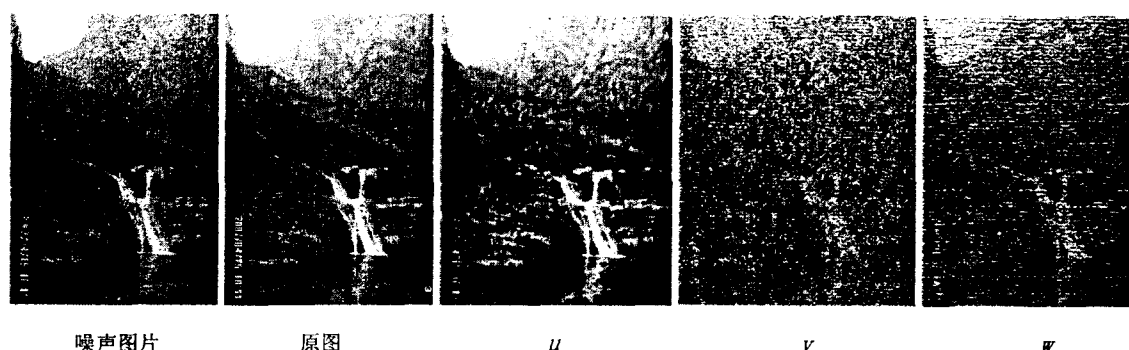


图 1 实验结果 I

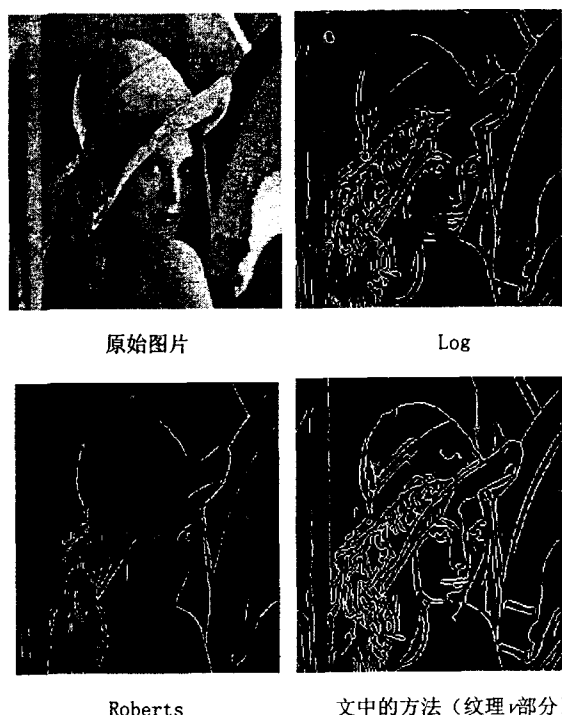


图 2 实验结果 II

F. Aujol, G. Aubert, L. Blanc-Feraud 等人提出的图像分割模型,对于处理含噪声多的彩色图片有着很好的效果,但对于噪声少、目标背景单调的图片进行处理时,效果并不突出。可见,图像的分割没有一种万能的分割模型,对于具体的分割目标还需注意选择合适的模型。

(上接第 43 页)

和管理,这是 workflow 管理系统研究的一个重要方面。也是文中进一步研究的重点。

参考文献:

- [1] der Aalst W, van Hee K. Workflow Management - Models, Methods, and Systems [M]. Cambridge, Massachusetts: MIT Press, 2002.
- [2] WfMC. The workflow reference model [R]. [s. l.]: WfMC

参考文献:

- [1] 冈萨雷斯. 数字图像处理 [M]. 第 2 版. 北京: 电子工业出版社, 2003.
- [2] 黄庆明, 张田文, 潘少静. 基于色彩学习的彩色图像分割方法 [J]. 计算机研究与发展, 1995, 32(9): 60-64.
- [3] Chan T F, Kang S H, Shen J. Total variation denoising and enhancement of color images based on the CB and HSV color models [J]. J. Visual Comm. and Image Rep, 2001, 12(4): 422-435.
- [4] Meyer Y. Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean [C] // Lewis J B. Memorial Lectures, Vol. 22 of University Lecture Series, AMS. Providence: [s. n.], 2001: 78-89.
- [5] Blomgren P V. Total Variation Methods for Restoration of Vector Valued Images [D]. [s. l.]: Dept. of Math., UCLA, 1998.
- [6] Blomgren P V, Chan T F. Color TV: Total Variation Methods for Restoration of Vector Valued Images [J]. IEEE Trans. Image Processing, 1998, 7: 304-309.
- [7] Aujol J - F, Aubert G, Blanc - Feraud L, et al. Image decomposition into a bounded variation component and an oscillating component [J]. Journal of Mathematical Imaging and Vision, 2005, 22: 71-88.
- [8] Aujol J - F, Aubert G, Blanc - Feraud L, et al. Decomposing an image: Application to SAR images [C] // in: Scale - Space '03, Vol. 2695 of Lecture Notes in Computer Science. [s. l.]: [s. n.], 2003: 297-312.

TC00-1033 Workflow Management Colition, 1995: 28-35.

- [3] 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.
- [4] 秦巧. 基于 .net 的工作流系统的设计和实现 [D]. 成都: 电子科技大学, 2006.
- [5] 范永全. 基于关系模式的工作流关系系统的设计和实现 [D]. 成都: 成都理工大学, 2002.