

一种支持动态群组的工作流建模方法^{*})

何 伟 王海洋

(山东大学计算机科学与技术学院 济南 250100)

摘 要 传统的工作流产品从建模到运行对业务流程中复杂的群组特性难以提供有效的支持。在通用工作流模型的基础上,对流程模型的表达能力和工作流运行机制进行了扩展,设计并实现了一个支持动态群组活动和群组协同的工作流管理系统。同传统工作流模型相比,在描述群组活动的能力上有所增强,使建模更简单。本文介绍了支持群组活动的建模方法和关键的运行机制,然后通过示例与传统的建模方法进行了比较。

关键词 工作流模型,工作流管理系统,动态群组,群组活动,业务流程建模

A Dynamic Group-supported Workflow Modeling Method

HE Wei WANG Hai-Yang

(School of Computer Science and Technology, Shandong University, Jinan 250100)

Abstract Traditional workflow products have difficulties in supporting the complex features of group activities in business processes. Through extending traditional workflow model in activity modeling method and execution mechanism, a dynamic group-supported workflow management system is designed and implemented. Compared with traditional workflow modeling method, this method has more capabilities in describing group activities and makes modeling simpler. This paper introduces the modeling method for group activity and the key execution mechanism in the workflow management system.

Keywords Workflow model, Workflow management system, Group activity, Synchronized collaboration

1 介绍

在许多企业的业务流程中,业务活动的“群组”特性非常鲜明。其中,群组的动态性或成员不确定性是一个较为普遍的需求:在某些流程中需要由多个潜在的、不明确的成员以投票、协商等方式确定流程的走向,即为基于角色和决策权的动态群组。因此,在很多企业中用于管理业务流程的工作流管理系统必须具备传统产品的功能之外,还应该在半结构化的业务流程、基于多角色的动态群组、群组成员之间协作等方面提供足够的支持。

当前的工作流管理系统从建模到实现的各层次上,普遍未提供对群组特性真正、灵活、有效的支持^[3]。通常的做法是在建模时将群组活动分解为多个单人交互活动,但是参与此类活动的成员之间往往存在着大量的交互和反复,特别是面临参与成员无法预先确定的动态群组时,流程模型需要频繁的修改和调整以适应实际情况的变化,这对企业的众多业务流程规划和管理而言无疑是一个灾难。

根据某些企业中由动态群组引出的特殊需求,以工作流参考模型^[1]为基础并予以扩展,我们设计并实现了一个支持动态群组的工作流管理系统。主要工作包括:

- 扩展通用工作流模型,使其在建模上支持动态群组,为工作流模型中“活动”这一类关键建模元素构建了扩展的层次体系;

- 基于 J2EE 架构设计并初步实现了支持群组的轻量级工作流管理系统,较好地解决了某些企业中由动态群组带来的问题。

本文其余部分组织如下:第 2 部分总结和回顾了工作流柔性建模和群组支持方面的工作;第 3 部分介绍了支持群组的工作流建模方法和核心机制;第 4 部分给出了工作流建模实例,分别用传统的建模方法和扩展的方法予以实现,并进行比较;最后,对本文进行了总结,并对下一步的工作进行了展望。

2 相关工作

WFMC 于 1995 年提出了工作流参考模型^[1],文献中给出了通用工作流管理系统的主要功能架构。在群组工作方面曾提及“工作流管理系统提供支持群件应用的、规范的、可控的过程框架已经成为逐渐增长的需求”,但未做进一步说明,文[4]提出了一种提高系统柔性的工作流建模方法,该方法扩展了传统的活动网络模型,通过协调活动结点支持群组协调工作,但未进一步讨论协调机制,对动态群组也无法支持。现有的工作流技术、产品允许流程实例中的每个步骤由群组共同完成,但是只能由其中的一个成员与工作流管理系统进行交互,以实现任务的接收和反馈,因此,从工作流管理系统的角度上并没有真正的群组概念。例如,IBM Lotus Domino Workflow 明确地支持某个任务由群组成员完成,但是必须指定某一个成员作为活动所有者(activity owner),负责向 Lotus Domino Workflow 系统反馈任务的完成^[5],事实上这个特殊的成员在工作流管理系统和其他成员之间扮演了中介的角色。

在流程模版或流程实例的动态修改方面,人们也做了很多工作^[6,7]。但是在通用的工作流产品中尚未有成熟的应

^{*})基金项目:国家自然科学基金(项目编号 60673130)。

用,用于解决企业中由于群组特性带来的流程频繁调整问题,需要花费较大的代价。

3 群组活动的建模方法

WFMC 提出了通用工作流模型中所包含的术语和符号^[2]。我们的模型主要以通用工作流模型建模元素为基础,通过对若干企业中群组工作的特点进行分析和抽象,引入了几种新的活动类型,为工作流模型中“活动”这一类关键建模元素构建了扩展的层次体系。

3.1 群组活动模型

在一般的工作流模型中,活动(Activity)是一个基本的概念,通常被分为交互活动(manual activity)和自动活动(automated activity)^[1]。我们对以上基本的活动元素进行扩展,从交互活动派生出单人活动和群组活动,其中群组活动又分为两种,分别用来表达确定群组交互活动和不确定成员的动态群组交互活动。图 1 表示了扩展工作流模型中活动的层次结构。

活动(Activity)、自动活动(automated activity)和交互活动(manual activity)是工作流模型中基本的元素,本文不再详细解释。以下对图中新引入的若干活动进行解释。

空活动(Dummy Activity):是自动活动的一种,不是一个语义上独立的工作单元,不包含具体的业务逻辑,仅用于描述其他活动之间的逻辑关系,比如活动之间的同步或分支。空活动可用一个元组表示:(活动标识,前驱活动集合,激活当前活动的逻辑关系,后继活动集合,结束当前活动的逻辑关系),其中为使模型简单和便于理解,激活和结束空活动的条件要求单一的逻辑关系,比如与、或关系。

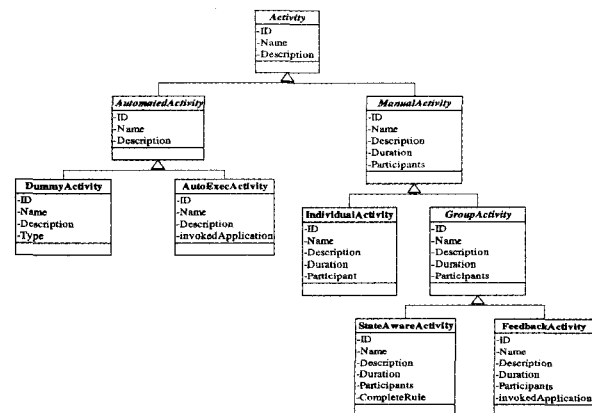


图 1 扩展的活动模型层次

自动执行的活动(Automatically Executed Activity):是自动活动的一种,包含需要自动完成的业务逻辑,需要绑定某个外部资源的业务对象,以便工作流管理系统能够进行调用。目前,绑定到此类活动的业务对象要求是 Web 服务或 EJB 组件。

单人交互活动(Individual Activity):能够并且仅能被一个成员接收、完成的活动。单人活动可作为多人活动的一种特例,为了使不同活动的表达更清晰,并且提高工作流引擎的工作效率,我们保留了此类活动。

群组活动(Group Activity):群组交互活动是一种多角色、多成员参与的交互活动,允许每个成员接收任务、向工作流管理系统报告自己的完成状态。

状态感知活动(State Aware Activity):用来表达基于多

种角色的、成员不明确的动态群组活动的元素。状态感知活动具有较为复杂的属性,包括此活动参与者所属的角色、每种角色的数量、每种角色的决策权、活动完成阈值等,活动拥有了这些属性,工作流引擎就可以随时感知当前活动实例的状态,而不必依赖于所有成员的工作状态,降低了任务在某个成员那里阻塞的可能性。此类活动实力的状态感知算法将在本文后面介绍。

群组反馈活动(Feedback Activity):是一种多角色、多成员参与的交互活动,要求所有成员各自接收任务,并全部反馈任务的完成状态。此类活动通常用来表达需要群组成员同步/异步协作的活动,必须绑定外部资源的业务对象。传统的工作流模型在表达此类活动时,往往分解为几个并行或串行的活动;此建模元素的引入使得流程模型简化,更符合现实情况。

3.2 动态群组活动的状态感知算法

每一个群组活动实例都会基于角色指定多个成员来接收和反馈完成,对于动态群组活动来说,参与任务的成员不是预先确定的,任何成员对于任务的接收和反馈也不是必需的。工作流管理系统可以随时根据迄今为止的反馈情况确定当前活动实例应转换的状态,从而根据所属的流程模版和流程实例确定下一步的动作。

在我们的工作流模型中,当某个动态群组活动实例的任何一位成员反馈完成后,工作流管理系统会立即使用以下算法确定当前活动的状态(是否已经结束)。

Algorithm: stateAwareness

```

Input: Process instance PID, //当前流程实例
      Activity instance AID, //当前动态群组活动实例
      The member MID, //当前成员
      The task TID which is assigned to MID //当前任务
Output: the state of AID //当前活动实例的状态(结束/未结束)
Begin
  Step 1. 将已经接收当前任务 TID 的所有成员划分为 2 个集合: 已经反馈完成的成员集合 Sc, 和尚未反馈的成员集合 Su;
  Step 2. If MID ∈ Su then {
    Su = Su - {MID};
    Sc = Sc + {MID};
    Revoke task TID from member MID;
  } //将当前成员从 Su 中删除,加入 Sc, 并收回当前任务
  Else //当前成员已经反馈完成
    Return the current state of AID. Algorithm ends.
  Step 3. If AID is an instance of "Group Feedback Activity" then //需要每个成员反馈的群组活动
    If Su = {} then
      Return "complete" state. Algorithm ends.
    Else
      Return "uncompleted" state. Algorithm ends.
  /* 以下步骤针对动态群组的状态感知活动 */
  Step 4. Group the elements of Sc according to roles into m sets;
    R1, R2, ..., Rm, Respectively with role r1, r2, ..., rm, m (= sizeof(Sc)), Sc = R1 ∪ R2 ∪ ... ∪ Rm; //按照所属角色将 Sc 中的成员分成 m 个集合
  Step 5. For each group ri with corresponding role ri;
    从集合 ri 中随机选取 n 个元素, n 为在当前活动中角色 ri 必须参与的成员数量(在活动属性中指定), 从 1 到 n 分别为每个成员赋值;
  Step 6. Set DC = 0; //活动实例 AID 的完成度量
    For each element with a number in Sc with its role r //被 step 5 赋过值的成员
    {
      DC = DC + tif(AID, r); // tif(AID, r) 为角色 r 中每个成员的任务影响因子(task influencing factor, 在活动 AID 的属性中指定)
    }
  Step 7. If DC < completeLevel(AID) then //活动 AID 的完成度量阈值, 在活动属性中指定
    Return "uncompleted" state. Algorithm ends.
  Step 8. For each element m in Su; //对每个尚未反馈的成员
    Revoke task TID from member m; //从所有尚未反馈的成员处收回任务
  Step 9. Return "complete" state. Algorithm ends.
End.

```

对于需要每个成员反馈的群组活动,算法 step 2 要求必

须所有分派的人员全部反馈完成信息,当前活动实例才能视为完成;对于动态群组的状态感知活动,算法根据每个成员所属角色对活动实例的影响程度,在部分成员反馈完成消息的情况下,判定活动实例能否完成,如果能够完成, workflow引擎从所有尚未反馈的成员处收回已分派的任务。因此,两种群组活动实例的完成状态能够保证是正确的。

假定当前活动实例所分配的成员数量为 m , 所涉及的角色数量为 $r(r \leq m)$, 该算法的时间复杂度为 $O(m * r)$ 。

当前活动实例转为“完成”状态后, workflow引擎将从所有尚未反馈的成员收回当前任务的分派, 并根据当前流程的定义及对 workflow 相关数据的检索、运算, 确定某个能到达的后继活动的所有前驱活动是否都已经完成, 据此确定是否激活某个能到达的后继活动。这一策略与大多数传统的工作流管理系统类似。

4 基于动态群组的建模示例

本节以电力企业中一类普遍存在的业务流程为例, 分别采用支持群组活动的模型元素和传统的建模元素对流程建模, 并予以比较。

在一个高压用电工程中, 电力企业的客户服务部门接收到客户的申请后, 首先组织相关人员进行现场勘查, 然后将调查结果提交给一个虚拟的决策委员会, 由一定数量和比例的决策者、电力工程专家、计量专家等以投票的方式决定接收或拒绝此项工程。需要引起注意的是, 决策委员会的成员无法预先确定, 但需要一个最小的、受角色比例限制的成员集合。

按照传统的工作流管理系统的任务分发机制, 必须指定某一个成员作为任务的接受者, 也就是说如果此任务被某个成员接收下来, 其他人将无法接收到同一个任务。如果为这一个环节建模, 只能基于确定数量的角色建模, 一个可能的模型如图 3 所示。此模型主要有以下缺点:

- 建模复杂;
- 灵活性较差, 当角色和数量发生变化时, 必须修改流程模版;
- 所有接收到任务的成员全部完成后才能进入下一步骤, 因此, 任务有可能在某个成员处阻塞。

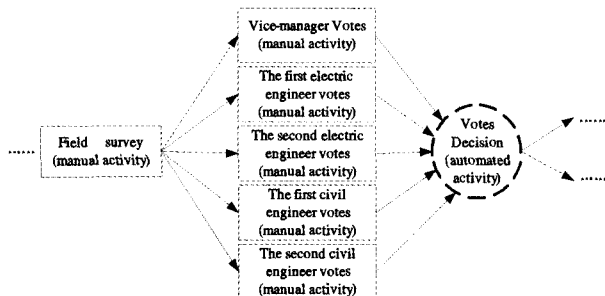


图 2 使用传统工作流模型描述动态群组活动

图 3 是采用“状态感知交互活动”的流程模型。 workflow管

理系统允许所有符合条件的成员接收此任务, 当任何一个成员反馈其完成状态时, workflow引擎将立即使用算法 stateAwareness 判定当前活动实例能否结束, 而不必依赖于所有成员, 降低了任务在某个成员那里阻塞的可能性。同时, 流程模型也更为简单、灵活。

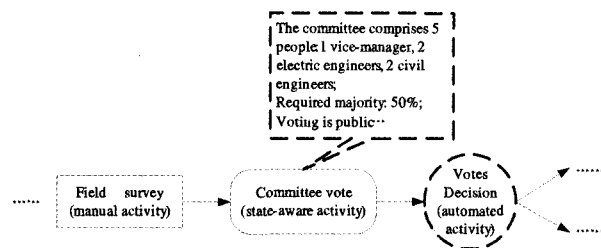


图 3 使用状态感知交互活动元素为流程建模

总结和展望 在企业业务流程中, 对群组活动的需求呈现出增长之势, 传统的工作流产品对复杂的群组特性通常难以提供有效的支持^[2]。在流程建模方面, 我们通过扩展传统的活动元素, 并引入新的群组交互活动, 为 workflow模型中“活动”这一类关键建模元素构建了扩展的层次体系。新的建模元素在群组活动方面的表达能力明显强于传统的建模方法。在此基础上, 设计并实现了一个支持动态群组活动的轻量级 workflow管理系统, 用来满足某些企业的业务流程中对群组活动的需求。

我们的下一步工作是继续完善 workflow管理系统, 使其在支持群组同步协同方面更加简单和灵活。

参考文献

- 1 Workflow Management Coalition. Workflow Reference Model, Document Number TC00-1003. The Workflow Reference Model (Issue 1.1). Document Number TC00-1003
- 2 Workflow Management Coalition. Workflow Terminology & Glossary (Issue 3.0). Document Number WFMC-TC-1011
- 3 van der Aalst W M P, Kumar A. A reference model for team-enabled workflow management systems [J]. Data & Knowledge Engineering, 2001, 38: 335~363
- 4 Fan Yushun, Wu Cheng. Research of a Workflow Modeling Method to Improve System Flexibility. Journal of Software, 2002, 13(4): 833~839
- 5 Nielsen S P, Easthope C, Gosselink P, Gutsze K, Roelle J. Using Lotus Domino Workflow 2.0. Redbook SG24-5963-00. IBM, Poughkeepsie, 2000
- 6 Reichert M, Dadam P. A Framework for Dynamic Changes in Workflow Management Systems. In: Proc. of DEXA'97, Toulouse, France, September 1997
- 7 Reichert M, Dadam P. ADEPT_flex Supporting Dynamic Changes of Workflows Without Losing Control, in Journal of Intelligent Information Systems, Special Issue on Workflow and Process Management, 1998, 10(2)