

基于 activiti 多实例任务回退方案的研究与实现

张超,王萌,任女尔

(北京卡达克数据有限公司,天津 300393)

摘要:近年来, workflow 技术日益成熟,会签在流程系统中的出现也越来越频繁。各大流程引擎也对会签做了默认支持。Activiti 作为现今十分流行的开源流程引擎,通过多实例任务节点也对会签做了支持,但是对多实例的回退却没做支持。为解决上述问题,该文通过 Activiti 提供的服务接口,并结合一定的算法,实现多实例任务的回退。

关键词:工作流;Activiti;多实例;回退

中图分类号:TP311 文献标识码:A 文章编号:1009-3044(2018)19-0113-03

DOI:10.14004/j.cnki.ckt.2018.2445

伴随着信息化技术的高速发展,企业信息化也得以逐渐深入,普通的办公自动化系统已经无法满足日渐增多的企业流程需求。会签作为工作流程中的一种常见的场景,可以让多个人相互协同,共同完成某一个任务,因而在流程的信息传递上更加自动化,在流程的信息处理上更加高效,在任务的完成上更加直观。所以会签使用的时间成本更低,安全性、封闭性更高,相应的,效率也就更高。

Activiti 通过使用多实例任务节点,在业务流程中定义重复环节,从而实现对会签的支持。多实例任务节点通过给定的集合,对集合中的每一个元素执行一次任务,既可以顺序同步执行,也可以并发同时执行这些任务。然后通过设置完成多实例任务的条件,决定会签的后续走向。但是对于某些业务场景,多实例任务节点并不能满足需求。比如:会签中的某个参与者发现自己没有参与会签的资格,造成该现象的原因可能是在选择会签人员时的疏忽。

针对该业务场景,本文对 Activiti 的多实例任务进行了扩展和延伸,使得会签任务中的某一个子任务得以有修正的机会,从而降低出错率,避免整个会签任务的失败。

1 技术简介

1.1 Workflow

1993 年工作流管理联盟(Workflow Management Coalition, WfMC)成立,作为工作流管理的标准化组织,对工作流给出了定义:工作流是指一类能够完全自动执行的经营过程,根据一系列过程规则,将文档、信息或任务在不同的执行者之间进行传递与执行。通俗点讲,工作流是将一组任务以及任务相关的人组织起来以共同完成某一个业务过程。它定义了任务的顺序以及后续任务的出发条件,每个任务可以由一个人来完成,也可以由多个人协同完成。

工作流引擎指的是 Workflow 应该作为应用系统的一部分,并为之提供解决方案,这些方案包括:根据分工和条件的不同以及角色来决定信息传递路由、内容等级等核心解决方案。工作流引擎包含了流程图的绘制、流程节点的管理、流程流向的

控制以及流程实例的管理等。目前市面上常见的流程引擎主要包含:JBPM、Activiti、SWF 等。

1.2 Activiti

Activiti 项目是一个开源的 BPM 平台,提供了对 BPMN2.0 标准的支持。Activiti 是一种 BPM 引擎,它轻量,并且可嵌入。为了可以被广泛使用,Activiti 提供了宽松的 Apache 许可 2.0。同时,为了促进和 BPMN2.0 标准的匹配,该项目由 OMG 通过标准审定。

Activiti 项目的创始人之一 Tom Baeyens,原先是 JBoss jBPM 的项目架构师,因此,Activiti 是基于 JBPM4 开发的,同时支持流程虚拟机技术,并且对外提供了清洗的服务接口,可以很好地集成外部服务。关于 Activiti 的优势, Tom Baeyens 曾说:“Activiti 有非常大的影响力来改变目前 BPM 的生态。Activiti 的 Apache 授权,完整的功能,将使 Activiti 到达一个新的水平。Activiti 将推动业界的创新,因为 BPM 技术可以广泛而自由地被应用。通过实现这些想法以及开源社区的努力,也让 Activiti 成为事实上的 BPM 和 BPMN 标准执行”。SpringSource 的首席技术官 Adrian Coyler 也说道:“这是一个对 Spring 开发人员和 Java 社区总体的发展非常令人兴奋的事情,长期以来一直需要一个 Apache 许可的流程引擎,这对许多应用系统非常实用的需求。我们认为,Activiti 作为新的应用领域扩展到的 Java 和开源的发展,特别是在云架构上”。

2 系统架构

本课题涉及项目的软件技术架构使用 JAVA 语言,采用 B/S 架构,综合 SpringBoot、Hibernate、Shiro 等框架进行业务逻辑开发,采用 Activiti 工作流引擎处理业务流程,页面则采用 Html、Bootstrap、Ajax 开发。整体系统架构如下所示:

收稿日期:2018-05-25

作者简介:张超(1986—),男,河南孟州人,软件工程师,学士,主要研究方向为计算机、汽车信息化建设。

本栏目责任编辑:谢媛媛

■■■■■■■■■■ 软件设计开发 ■■■■■■■■■■ 113

```

rollbackTaskVariablesLocal. put("MULTI_INSTANCE_ROLLBACK_",true);
rollbackTaskVariablesLocal. put("INSTANCE_ROLLBACK_ID_",user21Task.getId());
taskService.saveTask(rollbackTask);
taskService.setVariablesLocal(rollbackTask.getId(),rollbackTaskVariablesLocal);
6) user1 重新分配任务
user1Task = taskService.createTaskQuery().taskAssignee("user1").singleResult();
Map<String, Object> user1TaskVariablesLocal = taskService.getVariablesLocal(user1Task.getId());
if(user1TaskVariablesLocal.containsKey("MULTI_INSTANCE_ROLLBACK_")){
    boolean isRollbackTask = (boolean)user1TaskVariablesLocal.get("MULTI_INSTANCE_ROLLBACK_");
    if(isRollbackTask){
        String originalTaskId = (String)user1TaskVariablesLocal.get("INSTANCE_ROLLBACK_ID_");
        if(StringUtils.isEmpty(originalTaskId)){
            throw new Exception("the multi instance rollback task has no original task");
        }
        taskService.setAssignee(originalTaskId,"user24");
    }
}
7) 新分配的用户查看待办并完成任务,代码如下:
Task user24Task = taskService.createTaskQuery().taskAssignee("user24").singleResult();
taskService.complete(originalTaskId);
historyService.deleteHistoricTaskInstance(originalTaskId);
8) 会签任务结束,流程继续

```

3 结论

本文利用 Activiti 技术并结合一定的算法实现了多实例会签任务的回退,为多实例任务分发错误的情况提供了解决方案。这个方案通过使用 Activiti 自身提供的 API,并结合任务的本地变量,从而扩展了 Activiti 的多实例任务,让分发错误的子

实例得以修正,同时还不影响其他正确的子实例。但是由于时间、精力、知识水平的制约,本设计也有需要提高的地方,例如由于使用了变量,导致变量表的快速增加,对查询性能存在影响。后续可以考虑将这些变量的内容单独存在一张表中。

参考文献:

- [1] 范玉顺. 工作流管理技术基础[M]. 北京:清华大学出版社, 2001.
- [2] 闫洪磊. Activiti 实战[M]. 北京:机械工业出版社, 2014.
- [3] 顾大明. 基于 JBPM 的工作流会签设计[J]. 民营科技, 2010 (12).
- [4] 罗海滨, 范玉顺, 吴澄. 工作流技术综述[J]. 软件学报, 2000, 11 (7): 899-907.
- [5] 梅峰, 于洋, 卫青延, 等. 基于开源工作流 Activiti 的企业协同信息平台研究[J]. 福建电脑, 2012 (11): 125-127.
- [6] 张利君, 马骏, 杨涛, 等. 基于工作流的动态任务调度研究与实现[J]. 计算机工程与设计, 2009 (10): 2533-2537.
- [7] 王安慧, 孙长银. 管理信息系统会签管理模块的设计与应用[J]. 三峡大学学报, 2001, 23 (5): 454-457.
- [8] 冯天佑, 梁立宇, 李成华, 等. 基于 Activiti 工作流引擎技术的电力行业业务流程管理系统[J]. 信息通信, 2013 (8): 57-59.
- [9] Chen W. Enterprise information management system based on J2EE and MVC mode[C]//Proceedings of the 9th International Symposium on Linear Drives for Industry Applications. 2014, 3: 427-433.
- [10] 薛华成. 管理信息系统[M]. 北京:清华大学出版社, 1993.
- [11] WFMC-TC-1025, 工作流过程定义接口——XML 过程定义语言[S].
- [12] 阿斯特. 作流管理: 模型、方法和系统[M]. 王建民, 译. 北京: 清华大学出版社, 2004.
- [13] 叶小路, 钱真坤. 基于 Activiti5 的水司 OA 系统设计与实现[J]. 软件导刊, 2013, 12 (6): 93-95.
- [14] Alexander Grosskopf, Gero Decker, Mathias Weske. The Process Business Process Modeling Using BPMN[M]. Meghan-Kiffer Press, 2009.