

一种柔性审批流程模型^①

江日念, 许 锐, 乔德新, 申端明

(中国石油勘探开发研究院, 北京 100083)

摘 要: 为了满足大部分信息系统中审批流程灵活多变的需求, 在抽取一些审批业务流程的共性后, 建立了一个支持回退、跳转、流程变更等功能的通用审批流程模型. 通过将流程模版和流程实例分离, 使得模型能应对变更; 通过将流程模版和流程实例拼接, 使之能够展示流程的轨迹和流转动向; 通过配置模型的各个构成要素, 使之能够满足不同业务的个性化审批需求, 从而具备一定的柔性. 最后, 以科研管理信息系统中项目立项审批为例, 验证了该模型的柔性.

关键词: 柔性审批流程; 回退; 跳转; 流程变更; 流程轨迹; 可配置

A Flexible Approval Process Model

JIANG Ri-Nian, XU Kun, QIAO De-Xin, SHEN Duan-Ming

(Research Institute of Petroleum Exploration&Development, Beijing 100083)

Abstract: In order to satisfy the requirement of more flexible approval process which is demanded in most information systems, this paper extracted some features in common and established a universal approval process model that supports functions including rollback, skip and alteration. Mechanism of the model works as the following: through separating the templates and processing instances it makes the model available when alterations occur; through jointing the templates and processing instances, it is able to display the trajectory and circulation trend of the process; through configuration of constituent elements, it is able to meet individualized approval needs for different businesses, which means a kind of flexibility. Finally, the author presented an example of project approval in Scientific Research Management Information System to demonstrate that the model is flexible.

Key words: flexible approval process; rollback; skip; process alteration; process trajectory; configurable

1 引言

审批流程是以审批人为中心, 将电子形式的表单、文档、图像等信息在有关人员之间进行分发, 以便得到不同人的审阅、审核和审批等处理^[1]. 大部分信息系统中的审批业务具备一定的共性, 但个性化需求更多, 且这些需求变更的随意性很强, 传统的审批流程难以满足. 因为传统的审批流程更多的是关注“如何辅助开发者更容易的让流程运行完成”, 而不是关注“记录流程运行的历史和轨迹”^[2]; 传统的审批流程认为“回退”、“跳转”也是业务流程定义的一部分, 是通过分支和条件流转的设计来解决的, 而不能通过配

置实现; 传统的审批流程在设计阶段时, 流程已经固化下来了, 在运行阶段变更流程相当困难. 即传统的审批流程不易满足以下三方面需求:

(1) 流程的可见性. 任何审批人都可以看到整个流程运行的历史和轨迹.

(2) 流程“回退”、“跳转”、“终止”的可配置性. 在当前的审批节点中, 审批人拒绝时, 应该回退给哪个审批节点是可配置的; 在当前的审批节点中, 审批人同意时, 审批人是否可跳转流程是可配置的, “跳转”到哪个流程节点是可选择的, 是否可终止流程同样也是可配置的.

^① 收稿时间:2013-08-29;收到修改稿时间:2013-09-27

(3) 审批流程的动态变更的适应性. 增加、删除一个审批节点时, 不会影响已审批业务历史轨迹的正常显示和正在审批中业务的正常流转.

针对以上传统审批流程无法解决的问题, 本文设计和实现了一种柔性审批模型, 以满足绝大多数业务审批的需求.

2 审批流程模型的设计

一般的审批流程流转过程为: 填报人填写业务单据后, 提交给相应的岗位的人员进行审批. 该岗位下的审批人接收到审批通知后, 对审批申请执行审批^[3], 之后根据预先的流程定义, 决定下一步流程的流转, 直至流程终结. 下面将从模型构成要素、模型审批过程以及模型的设计三方面描述一个具备柔性的审批流程模型.

2.1 模型构成要素

该模型同一般审批模型类似, 同样具备两大要素: 审批业务对象和审批流程定义.

2.1.1 审批业务对象

审批的业务对象是指在审批过程中流转的信息, 这个信息可以是表单、数据列表, 也可以是文档、图像等等. 在功能确定的组织结构中, 审批业务对象按照一定的路径在各职能部门之间流转, 完成审批任务^[4]. 在不同的审批节点中的审批人, 对审批业务对象可以有不同的权限, 即看到的审批业务对象可以是不一样的. 这种不一样体现在两个方面:

(1) 对审批业务对象的关注点不一样. 即不同节点的审批人只看到了自己要关注的审批业务对象部分.

(2) 对审批业务对象的操作权限不一样. 有些审批节点的审批人仅仅只能对审批对象有查阅的权限, 而有些审批节点的审批人有对审批对象有修改的权限.

以上两种审批业务对象的个性化需求均可通过流程配置实现. 此外, 在每个审批业务对象实体中, 都有一个属性叫做“审批状态”. 这个属性用来标识审批业务对象在审批流程中的状态, 它有临时保存、审批中、已审批、已撤回四种状态. 每次流程提交时, 一旦涉及到审批业务对象状态变更, 就去更新这个属性, 为后续查看、统计审批业务对象的审批进度提供便利.

2.1.2 审批流程定义

审批流程定义是审批流中最重要的部分, 它预先

确定了业务对象在审批流中将走过哪些审批节点. 审批流程定义包含三要素: 审批节点、审批角色(岗位)、审批人. 定义的审批流程包含若干个审批节点, 每个审批节点对应唯一的审批角色, 该审批角色下配置了若干个审批人. 当一个具体的业务对象开始在流程中流转时, 就形成了一个流程实例, 每个审批节点对应的审批角色下的审批人便确定了. 每个审批人在提交流程时, 流程实例将记录下该步骤相关的所有信息, 如前序审批节点信息、当前节点信息、后续节点信息以及审批的业务对象信息. 将流程实例信息和流程定义信息隔离, 这样一旦审批流程定义发生变更, 不会影响审批中、已审批、已撤回的业务对象的正常审批.

在某些实际情况中, 有些流程节点的审批人具有较大权限, 可以选择跳过中间一些审批节点, 甚至直接终结审批流程. 这种“异常”流程是真实存在的需求, 但因其人为干预性很强, 使得在流程定义中无法根据条件分支等满足此类需求. 这就要求在某些流程节点中, 该流程的流转过程应该是可配置的, 即考虑到柔性化需求^[5,6], 则可以对审批流程进行柔性化定制和更改^[7].

2.2 模型审批过程

在该模型中, 流程基本的流转过程如下: 审批人填报表单或上传文档后, 将根据审批对象的业务类型和填报人的组织机构共同确定审批对象的审批流程. 将审批对象提交给相应角色后, 该角色下的审批人员根据配置, 有查看或者编辑审批业务对象的权限, 同时每个审批节点有两种选择: “同意”和“拒绝”. 当审批人同意该审批业务对象时, 审批的业务对象可继续往下流转, 根据流程的配置, 审批业务对象可以按序流转、跳转、甚至被终结审批. 当审批人拒绝时, 审批的业务对象必须回退, 根据流程配置, 该审批业务对象可回退给填报人, 亦可回退此节点以上任意一个审批节点. 在每一步审批中, 是否必须填写审批意见, 也是通过流程配置实现的.

2.3 模型设计

该模型设计的要点如下:

(1) 审批历史轨迹的显示

在填报人填报界面和审批人审批界面, 不只显示审批业务对象信息, 同时也显示流程的历史轨迹和后续尚未完成的流程节点信息. 这样填报人和审批人可清楚地知道整个业务对象流转的情况.

(2) 流程支持“回退”、“跳转”、“终止”操作.

通过配置流程节点属性,系统加载审批流程,审批人选择“拒绝”后,该流程将回退,默认将回退到填报人,即流程的第一步,此时也意味着流程撤回.如果流程定义中配置了回退节点的 ID,则流程将退回到指定的流程节点中.审批人选择“同意”后,流程可根据配置或按序流转、或跳转、或终结.

(3) 流程定义的动态变更.

审批流程定义中为了应对流程变更对现有流程实例的影响,将流程实例信息和流程定义信息隔离.这样在流程定义中插入、删除节点时,流程实例将自适应,因为此时的流程定义部分实际上由流程实例和后续未完成的流程“拼凑”而成.

下面以插入、删除节点为例,说明该模型对变更的适应性.其中 I 代表流程实例节点, N 代表流程定义节点, I_i 是 I_{i-1} 的后续节点,与 N_i 一样,都是流程实例和流程定义的衔接节点.

插入节点: 在流程中插入节点 N 时,有如下三种情况.

a) 插入节点在衔接点之前

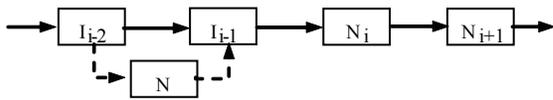


图 1 衔接点 N_i 之前插入衔接点

由图 1 可知,流程定义中在衔接点之前插入流程节点,该流程依然按照旧的流程定义规则执行,不影响流程实例的正常运行.

b) 插入节点在衔接点之后

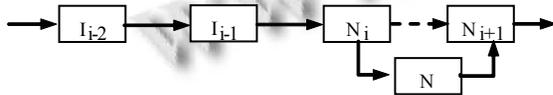


图 2 衔接点 N_i 之后插入衔接点

由图 2 可知,在衔接点之后插入节点 N,因为流程实例尚未运行此处,所以在此位置插入节点 N,后续流程会按照新的流程定义规则运行.

删除节点: 在流程定义中删除节点时,同样有以下三种情况:

a) 删除的节点在衔接点之前

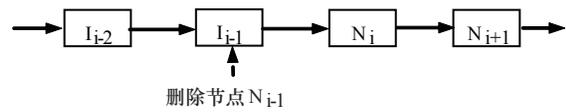


图 3 在衔接点 N_i 之前删除节点 I_{i-1}

由图 3 可知,删除节点 N_{i-1} 不影响流程实例运行,它依然按照旧的流程规则运行,因为流程实例已经记录了节点 I_{i-1} .

b) 删除的节点为衔接点 N_i

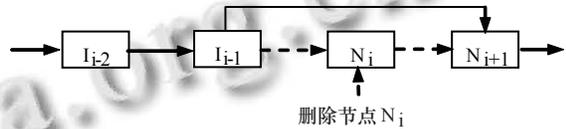


图 4 删除衔接点 N_i

因为在流程实例中,当前流程节点 I_{i-1} 会去流程定义中寻找衔接点 N_i ,流程初始化时会因找不到该流程节点引发异常.为了避免该异常,应在加载流程实例时,先更新流程实例,使得当前节点的下一节点指向衔接点的下一节点,如图 4.若衔接点为最后一个节点,则该终止该流程,并将业务状态的审批状态调整为“已审批”.所有该流程实例会按照新的流程规则运行.

c) 删除的节点在衔接点之后

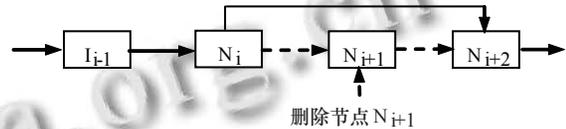


图 5 在衔接点 N_i 之后删除衔接点 N_{i+1}

如图 5 所示,同删除衔接点结论一样,只不过不用更新流程实例的下一个节点指向.

(4) 审批人的确定

每个审批节点下的审批人都是通过授权分配的,一个流程节点至少有一个审批人.审批人的确定根据审批角色的不同而不同,对于本部门的审批角色,审批节点将取出本部门相关的审批人.对于上级部门的审批人,审批节点将取出所有审批人,对于具有跨部门的角色,由于具体审批人预先难以固定,故由当前审批人手动选择,如项目负责人.至于审批角色的部门属性,则是通过在流程定义中配置流程节点的相关属性来实现的.

在流程实例运行时, 当前审批人提交审批流程至下一审批节点时, 必须确定下一环节审批人. 当下一环节审批人唯一时, 在此审批人的系统首页生成该业务对象的审批待办; 当下一环节审批人不唯一, 即选择了多个审批人时, 则系统采用“抢办”的策略: 在这些审批人系统首页均生成该业务对象审批待办, 如果其中一人已办理, 则其他审批人的待办消失, 不可办理.

(5) 退回后的流程的流转

此处流程的退回, 可能指退回到任意审批节点, 也可能指退回到填报人, 即流程撤回. 无论流程退回到哪个节点, 此时流程将在当前退回的节点 Iback 和流程的衔接点 Ni 之间重新流转, 流转过程中遵循两个原则:

- a) 谁审批过, 谁负责重新审批. 审批节点的审批人是唯一的, 是流程退回前办理审批业务的审批人.
- b) 流转过程中不支持“跳转”. 流程退回后, 审批人或者填报人可能会修改审批业务对象, 基于后续审批人对业务负责的考虑, 不允许审批人越级选择审批节点, 必须按照流程实例的历史轨迹按序走完流程.

3 审批流程模型的实现

下面以某机构的科研管理信息系统中项目立项审批为例, 描述了该审批模型的实现过程, 并验证该审批模型柔性.

3.1 业务场景描述

项目立项审批业务场景如下: 项目填报人填写该项目信息后, 提交给依托项目负责人审批, 由于该依托项目负责人不定, 需要填报人手动选择审批人. 该项目的依托项目负责人若审批通过, 则提交给所长审批, 否则退回给项目填报人. 所长若审批通过, 则流程提交给上级部门科研项目室的业务负责人审批, 由于某些特殊原因, 科研处项目室的审批人具有“跳转”和“终止”流程的权限, 即审批人同意该项目审批通过时, 审批人可将流程提交到后续的任意节点, 甚至终止流程. 科研处项目室的下一审批节点为科研处处长, 若经过该流程时, 且科研处处长同意流程通过时, 进入主管领导审批环节, 否则退回到依托项目负责人. 在主管领导的审批环节中, 若领导同意审批通过时, 流程终结, 否则流程退回到依托项目负责人并继续流转, 直至终结. 项目审批流转情况如图 6.

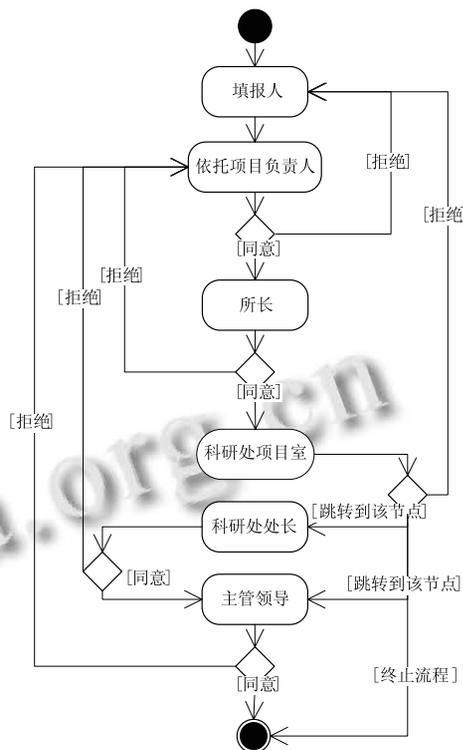


图 6 项目立项审批流转示意图

此外, 根据业务部门需求, 填报人和审批人可以随时查阅审批进度和流程的历史轨迹.

3.2 模型实现

根据 2.1 业务场景的描述, 可知该项目的立项审批流程必须实现以下七点, 使得实现的模型具备柔性, 以随时应对未来需求变更造成对系统的影响.

(1) 回退节点的可配置性. 如在 2.1 业务场景描述中, 科研处项目室审批拒绝后退回到依托项目负责人, 若需退回到其他节点, 须通过配置实现.

(2) 审批角色部门属性可配置性. 在该立项审批流程中, 审批角色部门有三种属性: 与填报人同一部门, 如所长; 跨部门, 如依托项目负责人; 上级部门, 如科研处项目室. 这三种属性角色下审批人的确定应有不同的策略. 如依托项目负责人不定, 所以必须依靠填报人手动选择若干个依托项目负责人.

(3) 跳转节点和终结流程可配置性. 某些流程节点有较大权限, 能够选择跳过后续某些流程节点, 或终止流程. 如科研处项目室就具备了 this 权限, 类似此种情况应该通过配置实现.

(4) 对审批业务对象的权限可配置性. 基于每个审批节点对项目的关注点不一样, 每个流程节点对看到审

批对象应不一样。此外有些节点能修改项目立项信息,而有些节点只有查看权限,如填报人和依托项目负责人有编辑权限,而其他审批节点的审批人则只有查看权限。

(5) 审批意见填写与否的可配置性。在每个审批节点,审批人在审批通过或审批拒绝时是否应该强制填写处理意见,须通过配置实现。默认情况下,拒绝时必须填写处理意见,同意时则可以填写处理意见。

(6) 流程变更的适应性。为了增强系统的稳定性,应对流程定义中增加、删除节点时对已审批和审批中的项目的影响,流程应由流程实例和流程定义“拼凑”起来,走过的节点将不受流程变更的影响。

(7) 流程的可见性。为了提高系统的易用性,让审批人能查看整个流程进度和流程所走的历史轨迹,在审批人的审批界面将显示整个流程信息,包括流程的历史轨迹和后续尚未走完的流程信息。

以上需求使得流程实现时必须考虑到流程的可配置性、流程变更的适应性、流程的可见性。基于这三原则后实现的项目立项审批流程如图 7。审批人在

序号	环节名称	处理人	处理结果	审核意见	处理时间
1	项目申报	张三	已填报		2013-05-14 10:30:36
2	依托项目负责人	张三	已填报		2013-05-14 10:38:13
		李四	未通过	预算金额不对	2013-05-14 10:37:48
		李四	通过		2013-05-14 10:38:45
3	所长	赵五	通过	同意	2013-05-14 10:39:22
4	科研处项目室	周六、吴七			
5	科研处长	郑八、王九、许十			
6	主管领导	董十一			

下一环节:	<input checked="" type="checkbox"/> 董十一 <input type="checkbox"/> 主管领导 <input type="checkbox"/> 科研处长 <input type="checkbox"/> 主管领导 <input type="checkbox"/> 终止流程	<input checked="" type="radio"/> 通过 <input type="radio"/> 拒绝
处理结果:		
处理意见:		

图 7 项目立项审批实现效果(审批流程部分)

前台界面可方便地查阅流程进度和流程历史轨迹,其中项目申报人在第一次提交流程后,依托项目审批人以“预算金额不对”退回。项目填报人修改预算金额后再次提交通过,直至流转到科研处项目室,此节点具

有跳转、终止流程的权限,即审批人可以选择审批的下一节点。每次处理系统都将记录审批人的处理时间,以使用户查阅、跟踪流程进度。

4 总结

本文描述了一个柔性审批流程模型的设计和实现过程。该审批模型具有流程可配置性、流程变更的适应性和流程可见性的特点,它能够弥补传统的审批流程灵活性的不足,有效地支持人工调整流程,在实现电子审批的同时,充分发挥了审批人的主观能动性。该自定义审批模型能应对各种“异常”的审批流程,这些“异常”情况,需要通过人机交互来动态决定审批流程的实际路径^[8]。该审批模型已经应用于某机构科研管理信息系统中,取得了不错的应用效果。

参考文献

- 张霞,陆剑江.面向角色的动态审批流程的研究.计算机应用与软件,2011,28(1):193-195.
- 胡奇.jBPM4 工作流应用开发指南.北京:电子工业出版社,2010,306.
- 王正方,南琳,王作鹏,孙兆华.企业信息系统中的业务信息审批模型.计算机工程,2008,34(9):257-259.
- 刘晓冰,吕强,薛冬娟,邱立鹏.基于特征映射的自定义审批系统的设计与实现.计算机集成制造系统,2008,14(5):970-976.
- 周建涛,史美林,叶新铭.柔性工作流技术研究的现状与趋势.计算机集成制造系统,2005,11(11):1501-1510.
- 韩月娟,赵雷,吕强,杨季文.一个基于 SOFM 的柔性化审批应用方案的设计.计算机工程,2006,32(21):275-276
- 蒋建彬.基于 SOA 的审批流程管理系统的研究和实现[学位论文].成都:电子科技大学,2010.
- 何青.一个基于 TBAC 的审批业务工作流模型.山东大学学报(工学版),2006,36(4):121-124.

(上接第 6 页)

- Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems, 1998, 30(1): 107-117.
- Vavilapalli VK, Murthy AC, Douglas C, et al. Apache hadoop YARN: Yet another resource negotiator. Proc. of the Fourth ACM Symposium on Cloud Computing. ACM, 2013.

- Hindman B, Konwinski A, Zaharia M, et al. Mesos: A platform for fine-grained resource sharing in the Data Center. Proc. of the 8th USENIX Conference on Networked Systems Design and Implementation. USENIX Association. 2011. 22-22.
- Corona Home Page. <https://github.com/facebook/hadoop-20/tree/master/src/contrib/corona/>.