

workflow 管理平台的设计及其在自助服务中的应用^①

唐华云, 郭 芬, 桑英硕

(中央国债登记结算有限责任公司 软件部, 北京 100032)

摘 要: 随着信息技术的普及, workflow 技术在信息、制造和软件领域得到了广泛的应用, 越来越多的企业引入了 workflow 引擎和 workflow 管理平台来管理日常的业务流程. 但目前大部分 workflow 管理平台都有自己的业务背景并且对系统环境、接口有自己特定的要求, 在不同企业之间移植会碰见水土不服的情况. 本文针对银行间债券市场债券账户开户及信息披露的业务特点, 基于 J2EE 和 SOA 架构, 根据 WfMC 发布的 workflow 管理平台参考模型, 自主设计并开发了一套 workflow 管理平台, 并在上面成功实施相关的自助服务流程. 这是 workflow 技术首次在银行间市场开户和信息披露等方面的应用, 它较好的提高了银行间市场自助开户以及信息披露等工作的效率. 从性能测试的结果来看, 平台的并发性和稳定性也能满足目前业务需求.

关键词: workflow; workflow 管理平台; 自助服务

Design of a Workflow Management Platform and its Application on Self-Service

TANG Hua-Yun, GUO Fen, SANG Ying-Shuo

(Software Department, China Central Depository & Clearing Co., Ltd, Beijing 100032, China)

Abstract: With the popularization of information technology, workflow technology has been widely used in information, manufacturing and software fields. More and more enterprises introduce workflow engine and workflow management platform into the daily processes management. But each workflow management platform has its own requirements on business background, system environment and interfaces, which leads to the difficulty on the transplant between different companies. In this paper, referring to the reference model of workflow management platform, we developed a workflow management platform based on its business characteristics, J2EE and SOA architecture. The successful application of self-service greatly improves the operational efficiency and service levels. It is better to improve the efficiency of the inter-bank market, such as self opening accounts and information disclosure. Performance test results show that the concurrency and stability of the platform can meet the general business requirements.

Key words: workflow; workflow management platform; self service

1 引言

workflow 的概念最早起源于制造业中的业务流程, 这些业务流程通常是企业从多年的生产实践中提炼出来的^[1], 具有明确定义的任务和流程步骤. 在 90 年代前, 大多数企业都是完全以人力的方式执行这些业务流程, 效率低下. 因此, 人们希望能够出现一种能够让计算机使能的技术, 让生产和日常办公中的部分业务流程实现无纸化、自动化的流转^[2]. workflow 是针对工作中固有程序的常规活动而提出的一个概念. 通过将活动分解

成定义好的任务、角色、规则和过程来执行和监控, 达到提高生产组织水平和工作效率的目的. workflow 技术为企业更好地实现经营目标提供了先进的手段^[3].

在过去的 20 多年里, 随着信息技术的普及和发展, workflow 技术在信息、制造、软件等领域得到了广泛的应用^[4], 越来越多的信息系统采用显式定义好的 workflow 模型来驱动, 很多公司的 OA、ERP、CRM 系统就是其中非常典型和成功的案例. 虽然信息技术进入金融行业已经几十年了, 网上银行和在线交易等系统也

① 收稿时间:2015-11-11;收到修改稿时间:2015-12-31 [doi:10.15888/j.cnki.csa.005316]

早就和人们的日常生活密不可分,但是金融机构的业务流程(如银行间市场债券账户开户、债券信息披露等)依然是通过工作人员面对面的交互、纸质材料的传输来完成的,比较大的限制了工作效率的提高以及成本的控制。以银行间债券市场传统开户流程为例,它是一项需要纸质材料的反复邮寄、甚至客户多次往返的金融业务,客户从提交材料到开户成功一般耗时3天以上,难以满足现在的业务需求。为了提高企业竞争力、优化企业服务质量,我们具有较为迫切需求去开发一套面向流程信息系统。目前我们系统环境较为复杂,该系统应能够与各种架构的业务系统,实现交互。为此,本文设计并介绍了一个自主设计和开发的基于J2EE SOA体系的工作流管理平台(以下简称该平台为“平台”),并在平台上部署了自助开户、信息披露业务流程,较大的提高了银行间债券市场开户和债券信息披露的效率。

文章组织结构如下:

第一节,文章简单阐述了工作流和工作流技术的起源和概念,并且对文章的组织结构进行了概括。

第二节,文章从业务背景、平台整体架构、工作流建模、工作流引擎、工作流监控和统计及权限管理和文件系统6个方面阐述了工作流管理平台的设计和实现思路。

第三节,文章先分析了使用自助开户服务之前,客户在公司开立账户的流程,然后再描述了自助开户服务在“平台”的部署以及和外界系统交互方式。

第四节,介绍并分析了“平台”的性能测试结果。

第五节,概括性的总结了本文的主要内容和实效。

2 工作流管理平台的设计与实现

2.1 业务背景

工作流管理平台是一个能够实现流程定义、管理并且能够根据计算机描述的工作流逻辑驱动工作流的平台或系统^[5]。工作流管理平台主要作用是协调活动并且通过顺序执行工作任务、调用合适的人力以及其他与这些活动相关的信息化资源去自动化执行业务流程^[6]。随着工作流技术在ERP、OA等领域的广泛使用,目前学术界和产业界都沉淀了不少成熟工作流管理平台(例如JBPM、Activiti等),但这些工作流管理平台在移植到别的公司系统环境或业务场景过程中,常常会出现水土不服的情况。所以很多时候,企业在引入工

作流管理平台时,会采用自主开发的方式发展自己的工作流管理平台。

总体而言,工作流管理平台业务背景的特殊性主要有以下几点:

1) 要能够较好的支持人工节点。银行间市场的开户业务是典型的流程型业务操作,需要跨各职能部门之间进行公文的流转协助,人工参与节点较多,效率较低,通过工作流管理平台可实现多部门之间的协同在线审批。

2) 要能够与多种异构的业务系统进行良好的交互。工作流平台作为相对独立的技术基础平台,需要与诸多业务系统之间进行信息交互,不同系统环境不一,差异性大,需要工作流平台从设计上具有高内聚,低耦合的特性,对外提供统一的流程服务接口。

3) 平台应可记录业务流程各个节点的运行时间和等待时间等方面日志,为以后流程优化以及有关考核提供依据。

4) 具备必要的权限管理功能。工作流管理平台的不同用户只能驱动被授权的业务流程。

5) 面向流程的文件系统。工作流管理平台中不少业务流程是文件审批类型,为了方便不同部门、人员在同一个业务流程的不同任务节点上都能够准确使用相关文件,工作流管理平台需要相应的文件传输与存储系统。

为此,本文立足公司和银行间市场的实际需求,设计并开发了一套轻量级的基于J2EE SOA体系架构的工作流管理平台。

2.2 工作流管理平台架构

1994年,工作流管理大会(WFMC)发布了它的工作流管理平台参考模型,这一参考模型已经成为目前几乎所有工作流管理平台的架构规范^[7]。“平台”在设计时参考该模型将平台分为3部分内容:web控制台、工作流后台、工作流数据库。Web控制台是工作流平台的前端展示界面,在上面用户能够对业务流程进行配置和管理,还能够对正在运行的工作流进行干预并查看平台中流程实例进行相关统计信息。工作流后台是“平台”的核心组成部分,它的功能有:导出和解析标准BPMN2.0格式的业务流程;与Web控制台和其他系统进行交互实现业务流程的设计、监控、统计;完成业务流程的调度;根据客户管理系统(UAP),对业务流程进行权限控制。工作流数据库包含三部分,按它

们的作用分为 workflow 配置库、workflow 运行库、workflow 历史库, 其中 workflow 配置库保存定义好的业务流程数据, workflow 运行库保存流程实例上下文信息, workflow 历史库保存已经结束的实例的运行信息。

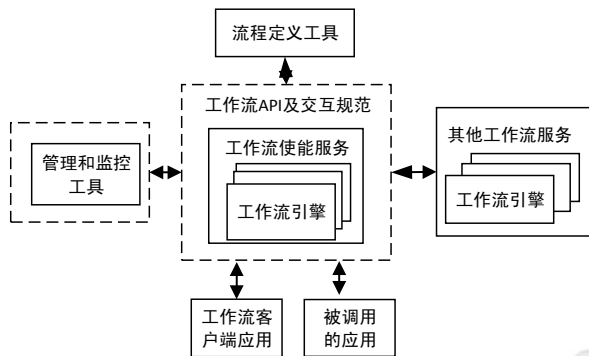


图 1 workflow 管理平台参考模型

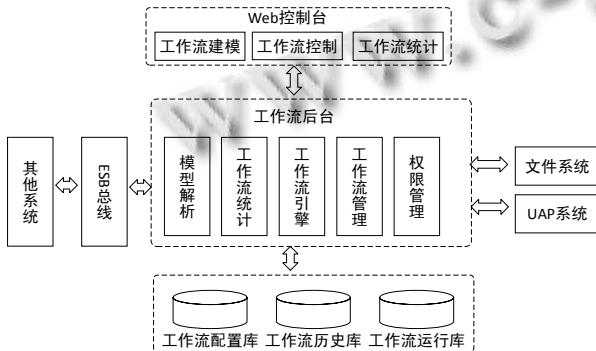


图 2 workflow 管理平台架构

在“平台”的技术架构上, 可以分为三个层次, 分别是数据持久层、应用服务层、展现及接口层。MyBatis 比起 Hibernate 要轻快很多, 上手容易, 也更加灵活, 非常适合 workflow 平台的开发特点。在应用服务层, 也就是服务的具体实现层面, “平台”也是以精简适用为原则, 拒绝过度引入开源产品, 以 J2EE SOA 为基础, 采用面向服务的设计思路, 所有服务都继承共同的接口, 采用反射技术来完成调用。展现层实现了 WEB 控制台的功能, “平台”根据需要引入了 easyUI、Draw2D 两个开源框架, 控制台大部分页面都是“无刷新”形式, 所以引入一个好的 Ajax 框架是必须的, 另外一方面 workflow 的可视化配置及进度查看界面, 通过引入 Draw.js 绘图框架实现。接口采用封装 jar 文件的形式, 其内部与“平台”的通讯协议使用 JSON 定义, 不依赖过多的开源项目, 保持“平台”的简单稳定易维护的特点。

2.3 workflow 建模

workflow 建模是指用计算机能够理解的语言表述人

们日常生活中的业务流程, 也被称为业务过程建模。目前国际上主流的业务过程建模方式是采用 BPMN2.0(Business Process Modeling Notation)规范建模, BPMN2.0 建模规范相比于 EPC 和 Petri-net 等建模规范, 具有更多的符号集合, 能够适应更多的业务场景, 而且具备更好的可读性, 能够非常好的被业务人员理解^[7]。目前主流的开源 workflow 平台都已经能够非常好的支持 BPMN2.0 规范建模(例如 JBPM 和 Activiti)。

展现层	Jquery、easyUI、CSS、HTML5等 统一请求响应处理
控制层	Servlet统一接入 统一页面分发 统一会话管理 统一报文转换
服务总线	ESB总线
应用层	HTTP入口 业务系统管理 模型管理 工作流实例管理 工作流任务管理 工作流统计
持久层	MyBatis
数据层	DB2

图 3 workflow 管理平台分层结构

任务节点类型有 7 种: (1)开始节点, 开始节点表示一个 workflow 流程实例的开始, 一个 workflow 流程只有一个开始节点。每一个流程实例都从开始节点开始, workflow 平台会在这个时候为流程实例完成所有初始化操作。(2)结束节点, 结束节点表示一个 workflow 流程的结束, 一个 workflow 流程有可能有多个结束节点。当流程实例运行到在结束节点时, workflow 平台会从 workflow 运行库中删除流程实例信息, 并且保存到 workflow 历史库中, 最后回收分配资源。(3)人工节点, 人工节点是指需要跟用户进行交互的节点。当流程实例运行到一个人工节点时, workflow 平台会暂停流程实例直到用户完成相关操作并且通知 workflow 平台重新启动流程实例。(4)作业节点, 作业节点是指调用外部系统完成某个作业的任务节点。当流程实例运行到一个作业节点时, 流程实例会向外部系统发送一条指令, 然后继续向下一个任务节点运行。(5)脚本节点, 脚本节点是不需要

与用户和外部系统进行交互的任务节点. 当流程实例运行到一个脚本节点时, workflow平台根据脚本节点配置调用相应的应用程序, 并且根据应用程序的返回值决定下一步运行的任务节点. (6)并行节点, 并行节点是一类流程控制节点. 当流程实例运行到并行节点时, 所有子任务节点都将被触发. (7)互斥节点, 互斥节点也是一类流程控制节点. 当流程实例运行到互斥节点时, 所有子任务节点中只有一个子任务节点会被触发.

2.4 工作流引擎

对于所有 workflow 管理平台, workflow 引擎都是其底层核心组件. 如图 4 所示, 本文 workflow 引擎由“流程实例上下文工厂”、“初始化服务”、“调度服务”、“结束服务”和 7 种基本任务节点组成. 其中“流程实例上下文工厂”是 workflow 平台中的一个数据共享平台, 流程实例在运行时跨任务节点的数据变量以及流程实例的一些上下文信息都被保存在“流程实例上下文工厂”中. “初始化服务”主要在流程实例创建时, 为流程实例创建上下文或者从 workflow 运行库中加载历史流程实例, 恢复其流程实例上下文. “调度服务”负责在根据 workflow 模型图和流程实例的具体情况调度不同任务节点. “结束服务”主要在流程实例结束或者暂停时调用, 它将内存中流程实例上下文信息保存到数据库中. 根据 workflow 建模是定义的 7 种任务节点, 在 workflow 引擎中, 对应封装了 7 种任务节点, 同一种任务节点具有统一的程序调用接口.



图 4 工作流引擎架构

工作流引擎调度一个流程实例的流程如图 5 所示. 在业务流程刚启动时, 调度服务调用初始化服务为新的流程实例创建流程实例上下文. 接着判断下一个任务节点是否为结束节点. 如果是, 调度服务调用“结束服务”将流程实例上下文和一些流程实例执行信息保存到 workflow 历史库中, 将流程实例在 workflow 运行库中

的数据删除, 最后释放流程实例占用的资源; 如果不是, 则调用相应的任务节点执行相应的操作. 每当一个任务节点完成之后, “调度服务”判断下一个任务节点的种类, 从而决定流程实例下一步操作.

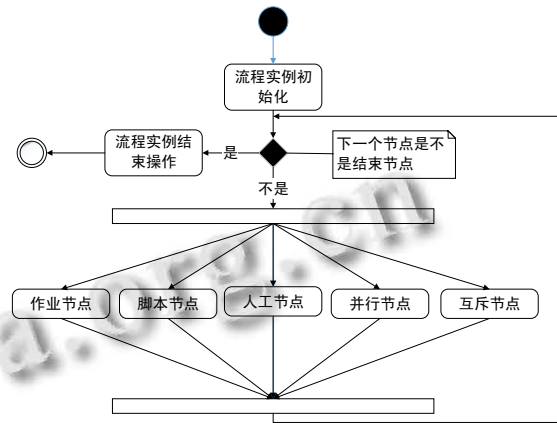


图 5 工作流引擎调度算法

为了更好的维护一个流程实例的完整生命周期, 为一个流程实例定义了如图 6 所示 5 种运行状态. 当流程实例刚创建时, 首先进入“初始化”状态, 由初始化服务为流程实例创建流程实例上下文. 当初始化完毕后, 流程实例进入“就绪”状态并且等待调度服务的调用. 当一个“就绪”状态的流程实例被调用时, 流程实例变为“活跃”状态. 当一个“活跃”状态的流程实例调用返回后, 流程实例继续变回“就绪”状态. “就绪”状态的流程实例可以被调度服务暂停, 从而变为“暂停”状态. 当一个流程实例被暂停时, 调度服务会将流程实例的上下文信息保存到 workflow 运行库中, 并且从内存中释放流程实例占用的资源; 当流程实例恢复时, 调度服务从 workflow 运行库中, 将流程实例上下文信息加载到“流程实例上下文工厂”, 并将流程实例变为“就绪”状态, 等待调度服务的调用. 当流程实例结束时, 会进入“结束”状态, 由结束服务负责清理流程实例占用的资源并且保存相关历史信息. 其中就绪的流程实例放置在一个队列中, 由工作流引擎顺序调用就绪的流程实例, 并且由工作流引擎进行状态的维护.

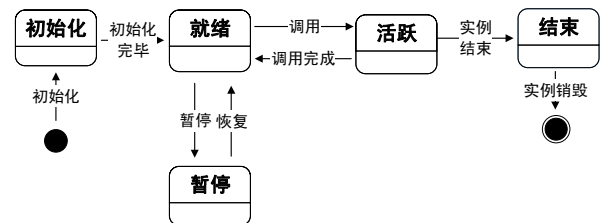


图 6 工作流引擎状态图

2.5 workflow 监控和统计

为了能够及时了解 workflow 平台上流程实例运行状况, workflow 平台在设计时通过 Web 控制台提供了图形化的 workflow 监控的功能, 并对外提供了 workflow 监控的接口. 具体而言, 用户可以通过 workflow 平台 Web 控制台或接口, 查询某一个流程实例的运行状况; 针对某一个 workflow 查询某一个任务节点的所有流程实例; 查询某一个用户最近使用 workflow 平台记录; 查询某一段时间内某个工作流的所有流程实例; 集中查询运行状态异常的工作流.

为了能够帮助 workflow 平台管理人员更好的管理和优化 workflow 平台, workflow 平台还提供了统计功能. 主要统计的要素有: 某个工作流的平均流转时间(指一个 workflow 模型的所有流程实例从启动到结束的平均运行时间); 某个工作流的平均流程实例深度(指某个工作流模型的所有流程实例从启动到结束平均执行的任务数目); 各个 workflow 每天流程实例的数目; 某个工作流每一个任务节点的平均耗时.

2.6 权限管理和文件系统

一个 workflow 的执行权限, 可以按 workflow 的任务节点划分. 能够驱动某一个任务节点的可以是一个部门组合或者一个公司员工的组合, 这些信息都是保存在 workflow 模型的配置文件里的. 某一个用户登陆 workflow 管理平台并想驱动具体业务流程时, workflow 管理平台从 UAP(客户授权管理系统)系统中获取他对应的部门等信息, 并对他的权限进行验证, 只让他驱动和浏览被授权的任务节点, workflow 流程的授权信息由平台管理员进行维护.

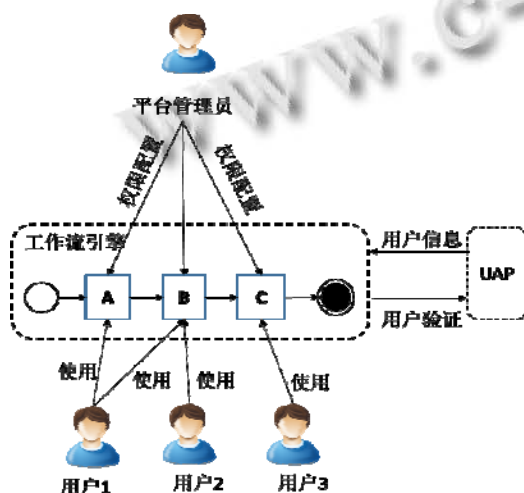


图7 workflow 管理平台权限管理

文件系统主要由一个文件存储数据表和本地硬盘组成. 文件存储数据表的结果如表 1 所示, 给流程实例编号和文件名添加唯一索引, 流程实例和文件名不能同时重复. 在一个流程实例中, 只需要根据文件名就可以相应的数据文件.

表1 文件存储数据表结构

序号	字段名	数据类型
1	文件 ID	整型
2	流程实例编号	字符串
3	文件名	字符串
4	存放地址	字符串

3 自助开户服务 workflow 的应用

3.1 传统开户流程

如图 8 所示, 在银行间债券市场传统开户流程中, 首先由客户服务相关部门接收客户邮寄过来的纸质材料, 并对其中内容进行审核, 审核通过则为客户开设债券相关账户; 在成功为客户开设债券账户后, 客户服务相关部门将纸质材料和客户债券账户信息转交给后台材料审核部门, 由他们对客户资料再次进行审核, 审核通过则为客户开设资金相关账户; 在两次审核中如有审核不通过的情况, 则中止流程并通知客户失败消息, 直至客户提交新的材料并完成整个开户流程. 在这样的一个开户流程里, 由于没有电子化流程化得系统辅助, 跨部门之间的协作都需要依赖于人工对客户纸质材料的串行审批, 工作效率较低, 对用户也带来诸多不便.

3.2 自助开户服务业务流程

通过对传统开户流程的分析和梳理, 基于本文提出的 workflow 平台, 将开户流程重构如图 9 所示. 首先由客户经办员通过客户端向自助服务平台提交电子开户材料; 然后由客户服务相关部门经办员通过中心端对客户材料进行债券账户开户资格审核; 审核通过则由后台审核相关部门经办员通过中心端对客户材料进行资金账户开户资格审核, 否则通知客户并等待客户重新提交材料; 后台审核相关部门经办员审核通过则由自助服务平台调用客户管理系统完成开户操作. 在整个流程中, 客户经办员都可以通过客户端查看开户进程.

通过自助服务平台实现的自助开户服务, 取消了纸质材料邮寄和经办员手工录入客户信息等工作, 提高了开户工作效率, 同时也提升了公司的服务质量.

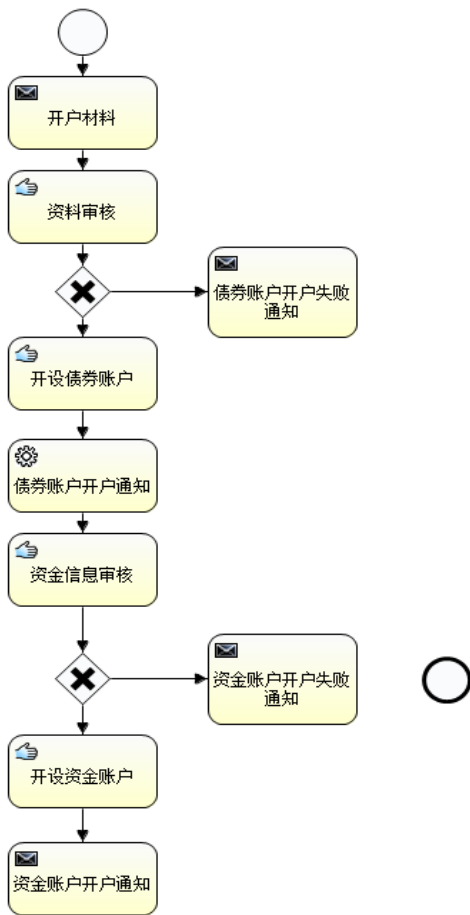


图8 传统开户业务流程

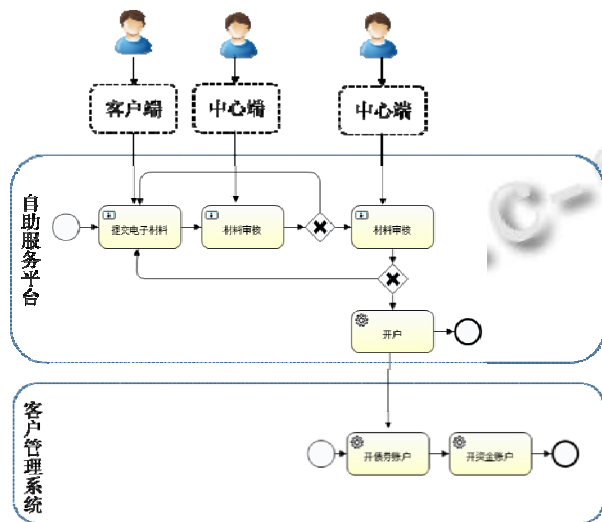


图9 自助开户业务流程

4 测试结果

在自助开户服务部署到“平台”上后，对“平台”所做的并发性测试结果如表 2 所示。在并发用户数为 50

时，启动工作流和重启工作流的响应时间在 3 秒以内，其他任务的响应时间也都在 2 秒以内，能够满足银行间债券市场的自助开户的需求。

表 2 并发测试结果

序号	任务名称	用户	RTT(s)	TPS	成功率(%)
1	启动工作流	50	3.082	15.971	100
2	删除工作流	50	1.366	13.289	100
3	参数修改	50	0.646	4.382	100
4	实例查询	50	1.097	43.715	100
5	任务查询	50	1.182	40.879	100
6	进度查询	50	0.553	15.948	100
7	接口调用查	50	0.302	84.434	100
8	重启工作流	50	2.197	1.397	100

对“平台”中自助服务进行 12 小时的疲劳测试，共调用接口 4437438 次，成功率达到 100%，未发生超时，宕机等异常情况，应用服务器 CPU 利用率在 60% 以下，系统能基本保持稳定运行。

表 3 疲劳测试结果

序号	接口	RTT	TPS	成功调用次数	通过率(%)
1	启动工作流	0.716	1.272	57363	100
2	重启工作流	1.38	0.097	4386	100
3	修改工作流	0.593	0.371	33432	100
4	删除工作流	0.673	0.675	60914	100
5	节点跳转	0.946	0.068	21490	100
6	下一个节点查	0.128	29.123	1312960	100
7	接口调用查询	0.068	46.467	2094483	100
8	参与工作流查	2.593	1.181	82020	100
9	任务查询	1.796s	3.507	423476	100
10	进度查询	0.092	7.703	346914	100

在测试环境下，进行自助开户测试， workflow 管理平台统计功能得到的结果如表 4 所示，从统计结果可以看出自助开户服务目前的平均流转时间是 140min(即开户流程从开始到结束平均耗时 140min)，平均流程实例深度是 6.5(即开户流程从开始到结束平均经过了 6.5 步)，在所有工作流任务中，“材料审核”(2) 是耗时最长的，提高材料审核的效率将有助于加快自助开户服务的平均流转时间。

表 4 自助开户服务 workflow 统计信息

序号	统计指标	统计结果
1	平均流转时间	140 min
2	平均流程实例深度	6.4

3	“提交电子材料” Pending	0
4	“材料审核”(1)Pending	40 min
5	“材料审核”(2)Pending	50 min
6	“开户“ Pending	20 min
7	“开债券账户” Pending	10 min
8	“开资金账户“ pending	13 min

5 结语

workflow 技术虽然起源于制造业, 但随着信息技术的发展, 已经在 OA 系统、ERP、CRM 等系统中取得了广泛而且成功的应用. 这些信息系统都表明, workflow 管理系统能够非常好的支撑具有明确业务流程的信息系统, 并且能够帮助这些信息系统实现流程优化和一定程度上的业务统计分析. 本文基于 workflow 技术的基本原理和 BPMN2.0 建模规范, 设计了一个基于 J2EE 和 SOA 架构的能够支持 7 种任务节点的 workflow 管理平台. 该 workflow 平台虽然设计较为简单, 但已经能够支撑很多流程相对固定的业务系统, 并且在“中债登”的自助开户和信息披露等自助服务中取得了较为成功的应用. 在 workflow 管理平台中部署了自助开户服务后, 原本至少需要 3 天的开户流程, 缩短至几个小时以内. 从性能测试结果上看, 平台能稳定的支持一定并发量的用户访问, 能够满足目前银行间市场的

业务发展需要. 通过引入 workflow 技术平台, 银行间市场自助开户的效率得以有效提升.

参考文献

- 1 Georgakopoulos D, Hornick M, Sheth A. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and parallel Databases*, 1995, 3(2): 119-153.
- 2 罗海滨, 范玉顺, 吴澄. workflow 技术综述. *软件学报*, 2000, 11(7): 899-907.
- 3 范玉顺, 吴澄. workflow 管理技术研究及产品现状及发展趋势. *计算机集成制造系统*, 2000, 6(1): 1-7.
- 4 廉成哲, 李欣, 李歆. workflow 技术概述及应用研究. *科技广场*, 2012, (5): 66-69.
- 5 Workflow MC. The workflow reference model [WfMC1003]. WfMC TC00-1003. 1994.
- 6 Meilin S, Guangxin Y, Yong X, et al. Workflow management systems: A survey. 1998 *International Conference on Communication Technology Proceedings*, 1998. ICCT'98. IEEE. 1998, 2. 6.
- 7 White SA. Introduction to BPMN. IBM Cooperation, 2004, 2(0).