

工作流在企业环境微生物检测中的应用^①

张 健, 张云华

(浙江理工大学 信息学院, 杭州 310018)

通讯作者: 张 健, E-mail: 280828919@qq.com

摘 要: 在当前企业环境微生物检测过程中, 操作人员手工处理数据很繁琐, 同时也会带来各种偶然误差, 在检测项目流程中, 纸质形式的数据流转存在着很多不合理、不规范等问题, 尤其各部门操作人员之间无法实现项目信息的共享, 致使客户和操作人员都无法及时准确地了解各项项目的进度和结果. 由此, 结合 JAVA 语言框架成熟和稳定的优点, 开发一项基于工作流的微生物检测系统, 用于规范流程, 信息共享. 该系统首先设计了一套符合企业微生物检测流程的工作流, 然后使用 Bootstrap 框架设计前端页面, 对检测关键数据进行对应的限定, 最后通过结合 Struts 和 Spring 框架, 开发后台业务程序. 整套系统逻辑严密, 结构稳定, 操作人员可以容易地获取所有检测项目信息, 极大地提高了微生物检测的工作效率和准确率, 带来了经济效益.

关键词: 工作流; 企业环境微生物检测流程; 信息共享; Struts 框架; Spring 框架

引用格式: 张健, 张云华. 工作流在企业环境微生物检测中的应用. 计算机系统应用, 2018, 27(3): 84-89. <http://www.c-s-a.org.cn/1003-3254/6263.html>

Application of Workflow in Microbial Detection of Enterprise Environment

ZHANG Jian, ZHANG Yun-Hua

(Information Institute, Zhejiang Sci-Tech University, Hangzhou 310018, China)

Abstract: In the current enterprise environment microbiological detection process, it is cumbersome for the operators to manually process the data, which also bring a variety of accidental errors. In the testing process, the paper form of data flow has many problems like unreasonable, non-standard, and other issues. In particular, the sharing of project information cannot be achieved between departmental operators, so that customers and operators fail to timely and accurately access to all projects progress and results. Thus, by combining JAVA language framework mature and stable advantages, this study develops a workflow-based microbiological detection system for regulating processes and information sharing. First, it designs a set of workflows that conform to the enterprise microbiological testing process in this system. Then, it uses the Bootstrap framework to design front-end pages and limit the key data. Finally, by combining Struts and the Spring framework, it develops background business processes. The whole system is rigorous and stable in structure, and the operators can easily obtain all projects' information, which greatly improves the efficiency and accuracy of microbiological detection and brings economic benefits.

Key words: workflow; enterprise environment microbiological detection process; information sharing; Struts framework; Spring framework

1 引言

工作流技术起源于上个世纪 70 年代中期办公自

动化领域的研究工作, 通过将工作分解定义良好的任务角色, 按一定的规则和过程来执行这些任务并对它

① 收稿时间: 2017-06-23; 修改时间: 2017-07-10; 采用时间: 2017-07-20; csa 在线出版时间: 2018-02-09

们进行监控来提高办事效率降低生产成本,提高企业生产经营管理水平和企业竞争力^[1]. Activiti 是一个业务流程管理 (BPM) 和工作流系统,适用于开发人员和系统管理员.其核心是超快速、稳定的 BPMN2 流程引擎.它易于与 Spring 框架集成使用.

JAVA 语言 Spring 框架是一个轻量级的控制反转 (IoC) 和面向切面 (AOP) 的容器框架.低侵入式设计,代码污染极低,同时独立于各种应用服务器 Spring 的 DI 机制降低了业务对象替换的复杂性,提高了组件之间的解耦; Spring^[2] 的 AOP 支持允许将一些通用任务如安全、事务、日志等进行集中式管理,从而提供了更好的复用; Spring 的 ORM 和 DAO 提供了与第三方持久层框架的良好整合,并简化了底层的数据库访问; Spring 并不强制应用完全依赖于 Spring,开发者可自由选用 Spring 框架的部分或全部.

JAVA 语言^[3] Struts 框架具有组件的模块化、灵活性和重用性的优点,同时简化了基于 MVC 的 web 应用程序的开发. Struts 可以清晰地区分控制,事务逻辑

和外观,从而简化了开发应用程序的过程. Struts^[4] 提供的类使得开发工作更加简单,这些类包括:实现和执行程序事务逻辑的类自定义的标记库使得创建和验证 HTML 表单更加容易.

针对当前企业环境微生物检测中的问题,设计了一套基于工作流的企业环境微生物检测平台系统.

2 企业环境微生物检测平台

2.1 业务背景和目标

在当前企业环境微生物检测平台中,操作人员手工处理数据很繁琐,工作量繁重,尤其在检测过程中,数据流转不合理,不规范的问题,致使客户和操作人员都无法及时准确地获取检测项目的进度和结果,甚至可能会带来严重的经济损失.由此,设计开发一套逻辑严密,结构稳定的微生物监测系统,进而可以提高工作效率和准确率,带来经济效益,减少不必要的经济损失.

2.2 系统的业务流程

系统整体业务流程如图 1 所示.

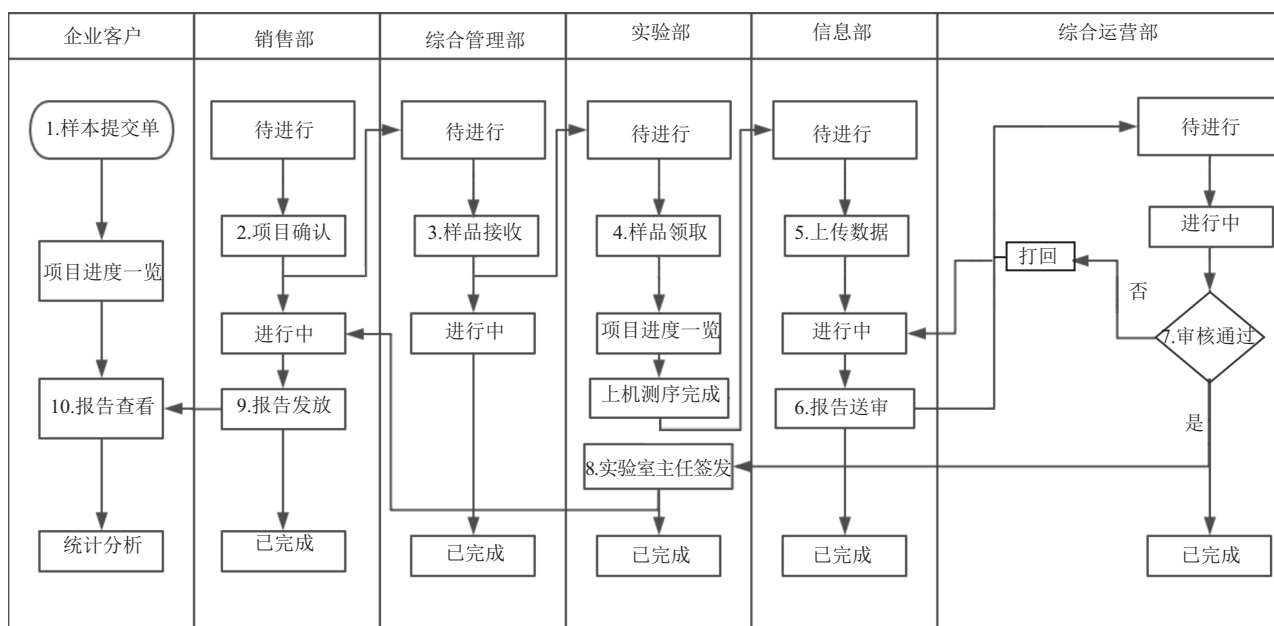


图 1 系统业务流程图

① 从客户提交订单后开始,同时产生一个唯一的流程编号,由工作流引擎将表单传给销售部门.② 销售部接受到订单任务后,检查订单,发现有误可以进行修改,无法修改时可以打回给客户,通知其重新修改订单提交.确认无误后,将订单传递到综合管理部门.③ 综合管理部门接收检查订单信息以后,发现问题同样可

以进行打回,直至无误后,将订单样品进行接收,任务进程进入到实验部.④ 实验部获取到样品,可以进行对应实验并填写实验数据.实验完成后,任务流程到达信息部.⑤ 信息部上传实验汇总数据,外部程序获取到上传实验数据后,进行验证分析,得出结果.⑥ 将结果返回到服务器数据中,系统程序自动检测到结果数据后,

根据结果生成实验报告,信息部将所得实验结果发送到综合运营部门。⑦综合运营部门对实验报告进行审核,审核不通过,可以打回到信息部门,审核通过,可以将报告送到实验室主任。⑧实验室主任对报告进行最后确认,决定是否进行签发操作。⑨签发后,数据报告到销售部门,销售部将报告发放给客户。⑩最后客户接收到报告后流程结束。

在系统整个流程中,任何部门都可以随时查询到订单的进度情况,信息能够实现共享同时每一个订单都能在部门之间准确的流转。

2.3 系统建设方案

基于当前纸质化的微生物检测流程,建立一个环境微生物检测平台势必存在一些挑战和困难,主要包括以下几个方面。

1) 如何将项目流程进度与部门联系起来,并实时查看项目所处部门,获取项目进度;

2) 如何解决实验步骤的控制和动态修改;

3) 报告审核过程中,每一个样本都要生成报告,然后并上传到服务器,由于每个样本都存在大量数据,生成报告和上传需要都消耗大量时间,如何优化报告审核功能。

4) 如何控制不同用户在不同地点登录权限问题。

针对上述问题,主要的解决办法如下。

1) 首先将所有的项目状态规范定义并与相关部门进行关联,然后统计不同类型项目步骤总数,项目每完成一步或者打回时,项目状态立即发生改变,并计算当前状态处于项目哪一步骤,确定进度值。

2) 首先需要设计好所有类型项目的实验步骤页面并对其进行编码,针对不同的项目的实验步骤,可以进行排序以及调整,项目实验进行过程中,通过当前项目类型和项目当前实验步骤,从而可以得到下一步实验步骤编码和对应URL地址,通过URL地址就可定位到对应实验步骤页面。

3) 报告审核时需要服务器中其他程序进行审核判断的,所以需要将一个项目中的所有样本数据生成对应的样本报告并上传到FTP服务器。这需要消耗大量的时间,用户在此过程中无法进行其他操作,用户的体验很差。本系统在原有的基础上,设置一个循环定时器,定时的去获取上传进度,直至完成后,销毁定时器。之前的流程在后台自动进行,从而给用户很好操作体验。

4) 由于数据安全性的问题,除了销售部门成员和

客户,其他部门人员只能在公司内部进行登录,而销售部门人员由于业务的需求,不能经常在公司内部,此时需要设置权限认证用户进行登录。公司内网采用是用VPN (Virtual Private Network),即“虚拟专用网络”,是虚拟出来的企业内部专线。

为外网登录用户分配登录权限,当用户进行登录,首先判断当前网络是公司内网还是外部网络,若为内网,账号密码无误后,直接登录,而处于外网,账号密码无误后,判断是否有登录权限,若有权限,登录成功,若无权限,登录失败。

3 企业环境微生物检测平台实现

3.1 开发环境

Eclipse 是著名的跨平台的自由集成开发环境 (IDE),是一个开放源代码的、基于Java的可扩展开发平台。就其本身而言,它只是一个框架和一组服务,用于通过插件组件构建开发环境,因此环境微生物监测平台的设计采用Eclipse开发环境。

3.2 数据库设计

MySQL 数据库简称MySQL,是一款由瑞典MySQL AB 公司开发并且广泛应用于小型企业或者组织中的小型数据管理系统,MySQL 数据库因其体积小、速度快、总体拥有成本低等优点。

MySQL 数据库是基于Linux操作系统开发出来的数据库,众所周知Linux是开放源码的操作系统,所以MySQL数据库也是开放源码的免费数据库,这也是MySQL数据库的优点。基于以上的优点,本系统使用就是Mysql数据库。

3.3 系统设计

系统的设计采用的是MVC模式^[5]。MVC是一种开发模式,是很强大,应用很广泛的,是面向对象的基础的常见开发模式。

MVC定义:即Model-View-Controller,把一个应用的输入、处理、输出流程按照Model、View、Controller的方式进行分离,这样一个应用被分成3层,即模型层、视图层、控制层。

3.3.1 后台业务设计

系统中后台设计采用分层结构来实现,主要分为3层:行为(Action)层,业务逻辑层,数据交换层。

其中Action层使用的是Struts技术,ActionServlet是Struts中核心的控制器,所有的用户请求都必须通

过 ActionServlet 的处理, ActionServlet 通过配置文件把用户的请求发送到对应的 Action 控制器中. 所以, 所有从前端向后台发送的请求方法, 都是在 Action 控制器中进行编写, 而 Action 全部交由 Spring 去管理. 数据库中获取的数据要先回到 Action 层, 通过 HttpServletResponse 将数据返回到前端. 由于篇幅原因, Struts 配置文件部分代码如下:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE struts PUBLIC
"-//Apache Software Foundation//DTD Struts
Configuration 2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">
<struts>
  <constant name="struts.action.extension"
value="do" />
  <constant name="struts.enable.Dynamic
MethodInvocation" value="true" />
  <constant name="struts.devMode" value="true" />
  <include file="struts2/*.xml"></include>
  <!--解决乱码 -->
  <constant name="struts.i18n.encoding"
value="UTF-8"/>
  <!-- 指定允许上传的文件最大字节数. 默认值
是 2097152(2M)-->
  <constant name="struts.multipart.maxSize"
value="10701096"/>
  <!-- 设置上传文件的临时文件夹, 默认使用
```

```
javax.servlet.context.tempdir -->
```

```
<constant name="struts.multipart.saveDir "
value="d:/tmp"/>
</struts>
```

业务逻辑层和数据交换层使用的是 Spring 技术, Action 层的获取的到前台 Ajax 请求后, 会先到业务逻辑层进行对应处理, 同时事务管理是通过 Spring 中的 AOP 模块来控制, 它是将一个或多个操作放在同一个事务下, 最终统一进行提交或者回滚. 而数据交换层, 即与数据库的交互, 是通过 Spring DAO 中的 JdbcTemplate 模块来实现. Spring 的 IOC 特性可以很好的管理 Action 层, 业务逻辑 (Service) 层, 数据交互 (Dao) 层. 部分管理代码如下:

```
<!--管理某一 ShortcutAction-->
<bean id="shortcutAction" class="zstuSoft.
action.ShortcutAction"></bean>
<!--管理某一 shortcutService 和对应 ShortcutDao-->
<bean id="shortcutService" class="zstuSoft.
shortcut.bpo.ShortcutService"/>
<bean id="shortcutDao" class="zstuSoft.shortcut.
dao.ShortcutDao">
  <property name="dataSource" ref="data
Source"></property>
</bean>
```

整个系统基础数据的传输交互流程如图 2 所示. 由于本系统的分层设计, 使得各层内部的能独立变化, 而整体不受影响.

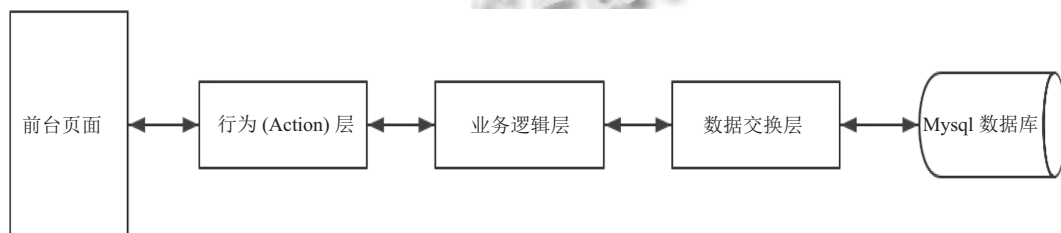


图 2 数据传输流程图

3.3.2 Activiti 工作流引擎接口的设计

Activiti 流程^[6,7]引擎重点关注在系统开发的易用性和轻量性上. 每一项 BPM^[8]业务功能 Activiti 流程引擎都以服务的形式提供给开发人员. 通过使用这些服务, 开发人员能够构建出功能丰富、轻便且高效的 BPM 应用程序. 图 3 是 Activiti 引擎的系统服务结构

图, 该图显示了引擎提供的所有功能组件.

本系统要实现流程的控制, 主要是使用了 RuntimeService 和 TaskService. 每当一个流程定义被启动一次之后, 都会生成一个相应的流程对象实例. RuntimeService 提供了启动流程、查询流程实例、设置获取流程实例变量等功能. 此外它还提供了对流程

部署, 流程定义和流程实例的存取服务.

在 Activiti 中业务流程定义中的每一个执行节点被称为一个 Task, 对流程中的数据存取, 状态变更等操作均需要在 Task 中完成. TaskService 提供了对用户 Task 相关的操作. 它提供了运行时任务查询、领取、完成、删除以及变量设置等功能.

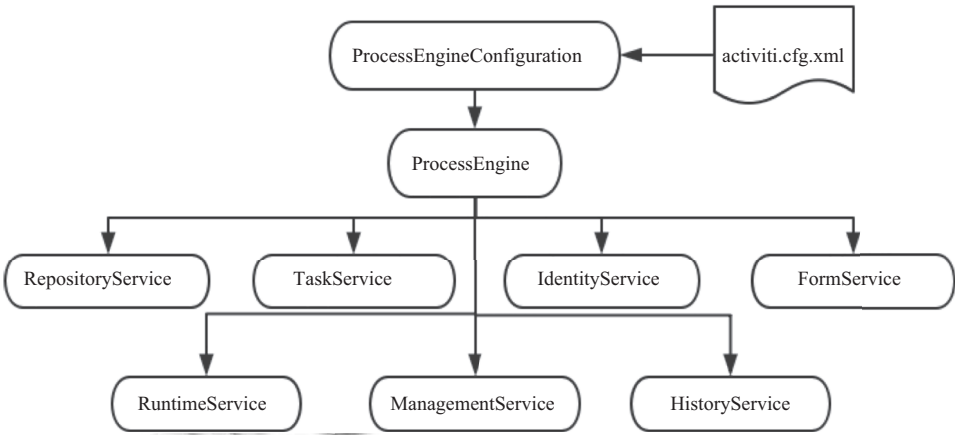


图 3 Activiti 引擎的系统服务结构图

本系统的工作流引擎接口放在业务逻辑层来进行处理, 这样带来的好处是, 系统在工作流中出现的异常时, Spring 中事务管理都可以方便的进行回滚, 从而避免因异常带来的错误. 下面简单介绍下任务流程开启和领取并完成个人任务代码:

```
//流程开启
runtimeService.startProcessInstanceByKey("taskKey", projectCode, variables);

Task task = taskService.createTaskQuery()
    .processInstanceBusinessKey(projectCode)
    .taskCandidateUser(code)
    .singleResult();
//领取个人任务
```

```
taskService.claim(task.getId(), code);
//完成个人任务
if(variables == null){
    taskService.complete(task.getId());
} else {
    //设置一些参数
    taskService.complete(task.getId(), variables);
}
```

客户通过页面点击提交订单按钮, 调用流程开启代码, 启动一次任务流程, 企业操作人员通过依次领取个人任务, 完成任务, 直至任务流程最终完成, 客户获得检测报告.

图 4 是根据系统业务流程设计的工作流程图.

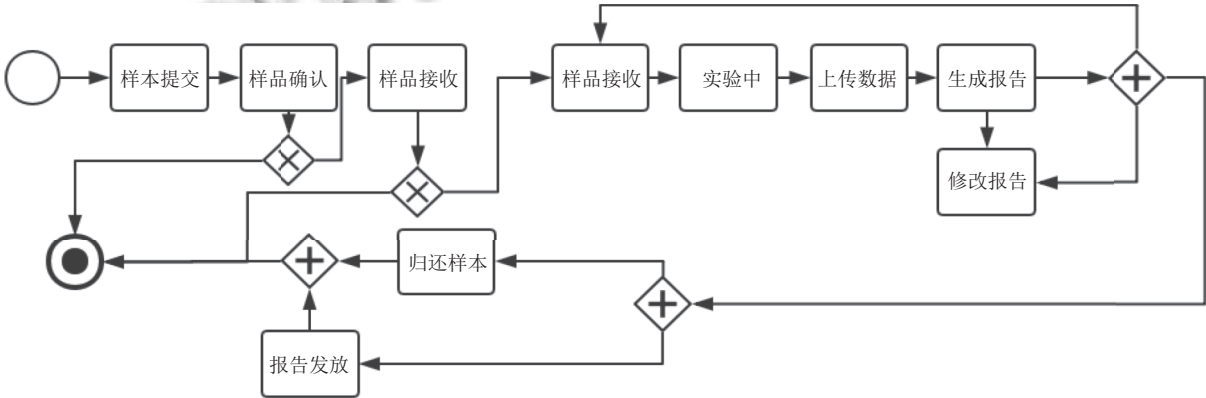


图 4 企业微生物检测系统工作流程图

4 结果分析

未引入本系统时,由于整个流程是通过纸质方式来传输的,人工完成一个微生物检测项目流程所需时间平均为10–15天,而当通过引入本系统后,作者测试了50个项目,每个项目100个样本,完成一个项目流程时间缩短为1–2天。

5 结束语

本文设计的一种基于工作流的企业环境微生物检测系统,将Spring和Struts框架与工作流很好的整合为一个整体,运用在了企业环境微生物检测中,使得信息能够共享,极大方便了企业客户项目进度的查询和获取,同时也大大的提高了企业工作效率,减少了手工带来的误差和不便,方便了管理,提高了经济效益。

参考文献

- 1 范玉顺. 工作流管理技术基础. 北京: 清华大学出版社, 2001.
- 2 王幸福. 基于Spring和Struts的轻量级J2EE通用框架的研究与应用[硕士学位论文]. 武汉: 武汉理工大学, 2007.
- 3 Eckel B. Thinking in Java. Upper Saddle River, NJ, USA: Prentice Hall, 2006.
- 4 林琴, 谭骏珊. 基于Struts框架的Web报表展示的设计与实现. 计算机系统应用, 2006, 15(11): 25–28. [doi: [10.3969/j.issn.1003-3254.2006.11.007](https://doi.org/10.3969/j.issn.1003-3254.2006.11.007)]
- 5 易可可, 陈志刚. 基于MVC模式的Web OA系统设计与研究. 计算机工程与应用, 2005, 41(4): 112–115.
- 6 唐华云, 郭芬, 桑英硕. 工作流管理平台的设计及其在自助服务中的应用. 计算机系统应用, 2016, 25(7): 89–95. [doi: [10.15888/j.cnki.csa.005316](https://doi.org/10.15888/j.cnki.csa.005316)]
- 7 汤幸江, 姚明海. 工作流系统在离散制造业的应用. 计算机系统应用, 2010, 19(8): 146–151.
- 8 王鹏. 一种轻量级BPM引擎功能扩展模型的研究[硕士学位论文]. 哈尔滨: 哈尔滨工程大学, 2015.