

高等院校目标考核信息化框架研究

欧阳宏基¹, 宋笑雪¹, 李红^{1,2}

(1. 咸阳师范学院 计算机学院, 陕西 咸阳 712000;

2. 西安电子科技大学 智能感知与图像理解教育部重点实验室, 陕西 西安 710071)

摘要: 开放、稳定和可扩展的目标考核信息化框架能够为高等院校目标考核工作提供详细、可共享的支撑数据, 有助于提高目标考核工作效率。针对目前高等院校目标考核工作中存在的数据分散、数据共享不及时、工作效率低下等问题, 在详细分析目标考核工作业务流程的基础上, 基于 MVC 设计模式, 整合 Struts2、Spring、Hibernate 和 Easy UI 框架, 按照 Java EE 分层开发模型设计了高等院校目标考核信息化框架, 包括基础数据、教学信息管理、指标管理、报表管理、系统维护和信息查询等重要模块。详细阐述了框架的结构和实现的关键技术。该信息化框架实现了目标考核网络化管理, 提高了考核工作中数据管理的精度, 达到了职能部门与二级学院之间数据及时更新与共享的目的, 提高了高校的业务管理水平。

关键词: 目标考核; MVC 设计模式; Struts2; Spring; Hibernate

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2018)10-0182-06

doi: 10.3969/j.issn.1673-629X.2018.10.038

Research on Information Framework of Target Assessment for College and University

OUYANG Hong-ji¹, SONG Xiao-xue¹, LI Hong^{1,2}

(1. School of Computer, Xianyang Normal University, Xianyang 712000, China;

2. Key Lab of Intelligent Perception and Image Understanding of MOE, Xidian University, Xi'an 710071, China)

Abstract: The open, stable and extensible information framework for target assessment can provide detailed and sharable support data for the target assessment of colleges and universities, which can help to improve the efficiency of target assessment. Aimed at the problems of data dispersion, untimely data sharing and low work efficiency in the target assessment work of colleges and universities, on the basis of detailed analysis of the target assessment work business process, based on the MVC design model, integrated Struts2, Spring, Hibernate and Easy UI framework, we design an information framework of target assessment in colleges and universities by the hierarchical development model of Java EE, including some import modules such as basic data, teaching information management, index management, report management, system maintenance and information inquiry. The structure and key technologies of the framework are described in detail. The framework implements the target assessment network management, improves the accuracy of data management in the assessment work, achieve timely updating and sharing of data between functional departments and secondary colleges, and improves the business management in colleges and universities.

Key words: target assessment; MVC design pattern; Struts2; Spring; Hibernate

0 引言

随着国内高等院校发展程度的提高和管理复杂度的增加, 目标考核信息化程度也要随之提高。目标考核工作所需要的数据来自多个职能部门、多个教学院系, 具有信息分散、逐步产生等特点。传统手工处理方

式存在考核数据处理任务繁重、更新不及时、共享程度低等缺点, 从而导致考核结果不透明、缺乏公正性, 无法反映某个部门的真实工作情况^[1]。目标考核过程要经历任务下达、任务分解、数据输入、数据审核、数据汇总、考评结果分析等多个环节, 目标任务数据、任务完

收稿日期: 2017-10-10

修回日期: 2018-02-26

网络出版时间: 2018-05-16

基金项目: 国家自然科学基金(61573267); 陕西省教育专项科研计划项目(17JK0828); 咸阳师范学院“青年骨干教师”培养项目(XSYGG201615)

作者简介: 欧阳宏基(1982-), 男, 讲师, 研究方向为软件工程、Java EE 应用; 宋笑雪, 博士, 教授, 研究方向为软件工程; 李红, 博士, 副教授, 研究方向为图像处理、软件工程。

网络出版地址: <http://cnki.net/kcms/detail/61.1450.TP.20180515.1651.040.html>

成的支撑数据、汇总数据以及考核结果数据是整个目标考核管理的核心数据,数据的管理工作能否到位直接影响整个考核过程的正确性和公平性。文中采用基于 B/S 的分层开发模式,整合 Easy UI、Struts2、Spring 和 Hibernate 轻量级开源框架,构建了高等院校目标考核管理信息化框架,实现了目标考核工作的数字化管理,提高了考核工作的透明度、公平性和公正性,为学校领导层作决策提供了数据支撑。

1 目标考核工作流程

通过对部分高校进行调研,得出目标考核工作的基本流程,如图 1 所示。

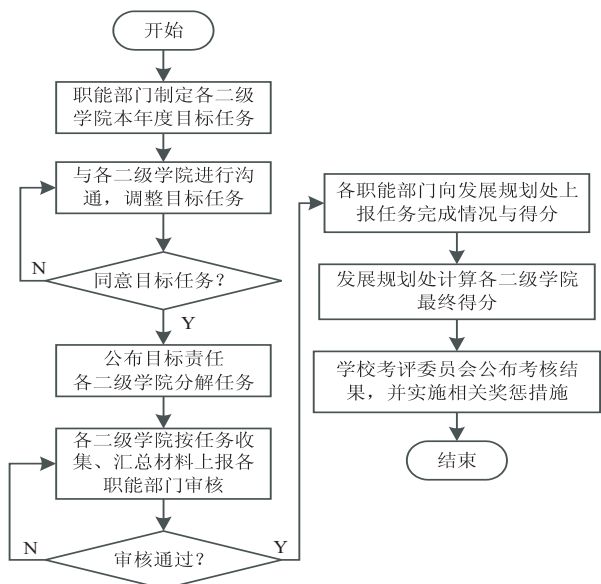


图 1 目标考核工作流程

首先在年初由发展规划处牵头,组织教务处、科技处、学科办、学生处、招生就业处、人事处、党委宣传部、党委组织部等学校职能部门给各二级学院制定目标责任书(征求意见稿),目标责任包括教育教学、科学研究、师资队伍、学生工作、党建工作等 5 大部分。在与各二级学院充分沟通、协调的基础上,确定目标责任书的正式版并向全校公布,各二级学院将目标责任按照本院职工岗位类型分解到人。到年底,学校启动考核工作,各二级学院要求每位职工上报任务完成情况并提交相关支撑材料,二级学校按照目标类别汇总资料并提交相关职能部门审核,审核通过后会计算每个责任类别的得分。各职能部门将审核后的得分数据汇总上报发展规划处,发展规划处将每个学院各目标类的得分按照相关规则进行换算,得到二级学院本年度最终目标考核得分。学校召开目标考核办公会,对各二级学院实施相应的奖惩措施,并对本年度的目标责任完成情况进行具体分析,并为来年制定目标任务提供重要的参

2 需求分析

根据上一节的业务流程,制定高等院校目标考核信息化功能,如图 2 所示。

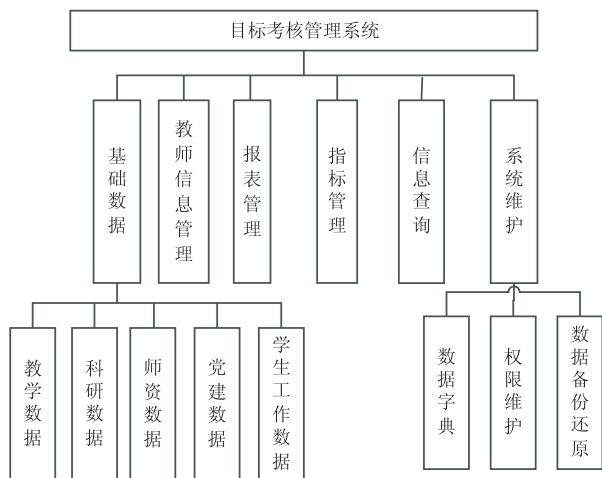


图 2 目标考核信息化功能模块

(1)基础数据维护:包括教育教学、科学研究、师资队伍、党建、学生工作等 5 个方面的基础数据。教师按照二级学院分配的任务,从教学任务、项目申请、论文、专利、著作、获奖等方面按照三级指标要求,维护本年度的完成信息。教学秘书从学科建设、实验室建设、实习基地、质量工程等方面维护集体指标方面的信息,监控本学院教师任务完成情况,并向职能部门上报汇总信息。学生办公室从党建、学生工作两个方面维护本年度的信息,并向职能部门上报。职能部门对二级学院上报的信息进行审核,按照三级指标的得分点计算生成目标得分,上报发展规划处。发展规划处汇总生成各二级学院最终考核成绩。

(2)教师信息管理:教师个人对自己基本信息的维护,包括职称变动、部门调动、岗位聘任结果、获得荣誉、密码修改等操作,都通过该模块实现。

(3)报表管理:各二级学院按照一级指标生成本学院年度任务完成情况报表;各职能部门按照各自分管的指标生成全校年度任务完成情况报表;发展规划处生成各二级学院年度考核结果报表。

(4)指标管理:根据目标责任划分指标,分为教育教学、科学研究、师资队伍、学生工作、党建工作 5 个一级指标,各一级指标下包括若干二级指标,各二级指标下又划分若干三级指标。该模块实现对指标的增、删、改、查。

(5)信息查询:二级学院领导查看本学院目标任务完成情况。职能部门按照分管的一级指标查看二级学院的完成情况。领导层以学院、工号、学位、职称、聘任岗位、指标等条件查看任务完成情况。

(6)系统维护:实现数据库的备份与还原、系统公告的发布、权限角色信息的管理、相关参数的配置等。

3 目标考核信息化框架设计

3.1 目标考核信息化框架结构

目标考核信息化框架总体基于 Java EE 平台,满足 MVC 设计模式^[2-3],服务器采用开源 Tomcat7.0,综合应用 Easy UI、Struts2、Spring 和 Hibernate 等开源框架进行搭建,是一个 5 层的分层结构,如图 3 所示。

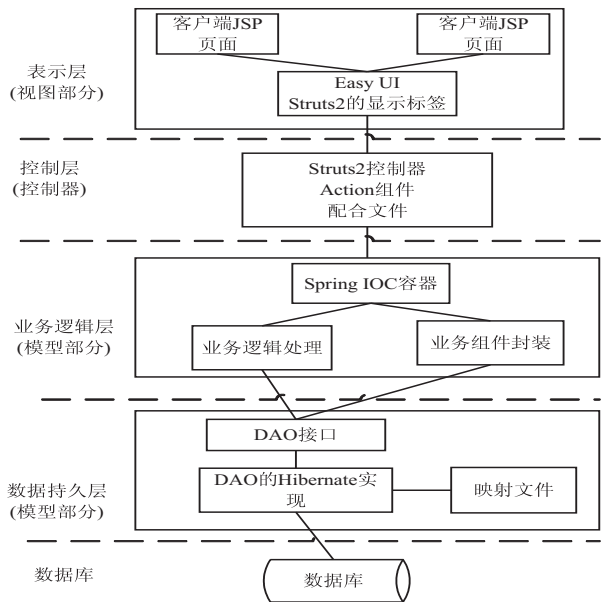


图 3 目标考核信息化框架结构

(1)表示层:负责构建交互界面,接收并转发用户请求到控制器,向用户显示响应请求的结果。该层主要由 JSP、CSS 样式、Div、Easy UI 和 Struts2 的标签构成。Easy UI 是轻量级的 JavaScript 框架,提供了丰富的插件,强大的选择器和良好的浏览器兼容性,能够简化 DOM 和 Ajax 操作^[4]。Struts2 标签通过与值栈数据交互,简化页面 Java 代码量,提高了开发效率^[5-6]。

(2)控制层:该层提供控制逻辑,通过 Struts2 配置文件将请求/响应页面与 Action 组件关联起来。Action 组件负责调用下层的业务逻辑组件执行用户请求,解耦合了视图部分与模型部分。

(3)业务逻辑层:该层功能由 Spring 框架实现,包括 Service 接口和 Service 实现两部分。Service 接口向控制层提供调用业务逻辑处理的入口,JavaBean 组件负责 Service 实现并调用下层的接口完成持久化操作。业务层组件之间的关联关系通过 XML 文件配置。

(4)数据持久化层:由 DAO 模式与 Hibernate 框架实现,DAO 接口定义 POJO 对象的持久化逻辑并作为业务层访问的入口^[7],DAO 实现封装 Hibernate 通过映射文件完成对象与数据库之间的交互。

(5)数据库层:该层负责存储目标考核业务所需要长久保存的数据以及之间的关联关系,通过触发器完成相关的汇总、统计计算。综合考虑易操作性、稳定、安全、集群搭建、开源等特性,最终选择 MySQL 数

据库。

3.2 框架实现的关键技术

(1)以 Spring IOC 机制为核心的框架整合。

通过 3.1 节对框架平台结构的解析可知,Struts2、Spring、Hibernate 均是 MVC 模式某个部分的实现,必须将它们整合起来形成统一整体。整合的思路是以 Spring 为桥梁,向下将 Hibernate 的 SessionFactory 纳入 IOC 容器管理;向上将 Struts2 的 Action 纳入 IOC 容器管理^[8]。具体实施时采用 XML 文件+注解的方式进行框架整合和系统开发,其中基础配置部分采用 XML 文件,包括 Struts2、Spring、Hibernate 的核心配置文件以及持久化类的映射文件,这部分的内容一次配完后,以后很少变动。开发部分中的各层组件及其依赖关系采用注解,利用 @ Controller 定义 Action 组件,在其内部通过 @ Resource 定义 Action 所依赖的业务逻辑层组件。利用 @ Service 定义业务逻辑组件,在其内部通过 @ Resource 定义所依赖的持久层组件。利用 @ Repository 定义 DAO 组件。组件的设计变动较大,采用注解方式能够提高开发效率并具有一定的灵活性。

(2)ORM 技术。

ORM 是定义面向对象技术与关系型数据之间如何映射的一套规范,能够简化开发人员使用 SQL 和 JDBC 处理关系数据库的操作。Hibernate 是实现 ORM 规范的一个具体产品,通过提供 Configuration、SessionFactory、Session、Transaction 和 Query (或 Criteria) 5 个核心接口对持久化对象进行存储和事务控制^[9],交互过程如图 4 所示。

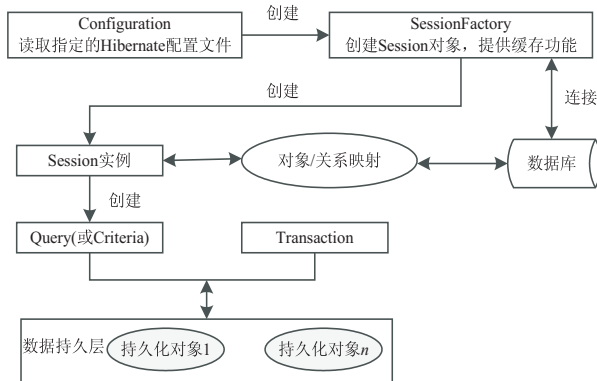


图 4 Hibernate 核心接口交互过程

首先,Configuration 接口读取配置文件启动 Hibernate 并创建 SessionFactory;然后,由 SessionFactory 实例负责初始化 Hibernate,建立与数据库的连接^[10]。SessionFactory 是重量级组件,在初始化时只创建一次常驻内存,用来创建 Session 对象。最后,Session 对象创建 Query (或 Criteria) 对象,在事务处理对象 Transaction 的协助下,通过映射文件实现持久化对象与数据库的交互。

(3) AOP 技术。

AOP 是一种关注点分离技术^[11], 强调将掺杂在核心业务逻辑中的辅助逻辑(例如事务处理、日志记录、访问控制等)进行分离, 形成单独的公共模块。通过反射与动态代理机制将公共模块与核心业务模块相融合^[12], 从而形成完整的系统。Struts2 和 Spring 提供了对 AOP 技术的支持, 目标考核信息化框架采用 Struts2 的 AOP 实现权限验证和日志记录, 采用 Spring 的 AOP 实现对 Hibernate 事务的控制。

以访问控制的设计为例, 说明 AOP 的应用。由于目标考核涉及众多不同角色的用户, 所以采用 RBAC 模型^[13]进行用户、角色的管理。分别定义用户类、角色类和权限类, 用户中包含角色, 角色中包含权限。权限类中包含一个 URL 属性, 代表某个可用资源的路径。用户访问系统的某个资源时, 在所拥有的权限集合中去检索这个资源的 URL, 如果找到表明用户可以访问这个资源, 否则用户不具备操作这个资源的权限并跳转到访问出错提示页面。

由于 Struts2 是通过拦截器提供 AOP 支持的, 所以将上述访问控制逻辑提取出来定义到拦截器中, 相关核心代码如下:

```
public class AuthorityInterceptor extends AbstractInterceptor
{
    //AOP 的切面逻辑
    public String intercept ( ActionInvocation invocation ) throws
Exception
    {
        //从 session 中获取登录用户信息
        User user = ( User ) ActionContext. getContext ( ). getSession
( ). get ( "user" );
        //获取所访问的 Action 对应的 URL
        String url = buildURL ( invocation );
        if ( user == null )
        {
            if ( "user/userAction_login". startsWith ( url ) )
            //如果是去登录, 就放行
            return invocation. invoke ( );
            else
            {
                HttpServletResponse response = ( HttpServletResponse ) Ac-
tionContext. getContext ( ). get ( org. apache. struts2. StrutsStatics.
HTTP_RESPONSE );
                response. sendRedirect ( "login.jsp" );
                //如果不是登录, 就转到登录页面
                return "loginUI" ;
            }
            //如果已登录, 就判断权限
            else
            {
                if ( user. hasAuthorityByUrl ( url ) )
                //如果有权限, 就放行
                return invocation. invoke ( );
                else
                万方数据
            }
        }
    }
}
```

```
//如果没有权限, 就转到提示页面
```

```
return "noAuthorityError" ;
```

```
}}}
```

通过上述源码得出, AOP 技术进一步解耦合了业务逻辑与辅助逻辑, 对辅助逻辑进行模块化设计, 减少了代码的编写量, 提高了复用率。

3.3 框架实现的流程

采用分层开发模式完成框架的实现, 该方法最大限度地降低了各层之间的耦合性, 实现了表示层、控制层、业务逻辑层、数据持久层、数据库各层之间的同步实施, 提高了开发效率, 降低了开发难度。框架具体实现流程如下:

(1) 按照领域建模的思想, 创建实体类及其之间的关联关系, 并在 Hibernate 的映射文件中进行描述。Hibernate 会将实体类映射成相关的数据表, 关联关系映射成表之间的外键。该系统设计的实体类一共有 52 个, 分别表示用户、职能部门账户、角色、权限、教材、专著、科研项目、科研论文、科研奖励、部门、岗位、课程、班级、综合指标、业绩指标、指标下达、指标完成和指标得分等。以指标管理模块的综合指标为例, 实体类及其联系如图 5 所示。IndexLevelOne 表示一级指标, IndexLevelTwo 表示二级指标, IndexLevelThree 表示三级指标。一级指标与二级指标之间是一对多关系, 二级指标与三级指标是一对多关系。AssignTask 表示目标任务, 由各职能部门给各二级学院下达目标任务时使用。TaskDetail 表示目标任务完成的具体信息, 由各二级学院教学秘书和教师角色按照三级指标提交任务完成信息时使用。TaskSubtotal 表示审核通过的任务信息, 审核通过后计算对应三级指标的实际得分和超额得分, 供按二级指标和一级指标统计某个学院得分时使用。

(2) 在数据持久层定义实体类对应的 DAO 接口和 DAO 实现类, 在 DAO 实现类中封装 Hibernate API 完成数据库的交互。由于每个实体类都具有保存、删除、修改、按 id 查找、查找全部等共性持久化逻辑, 为了将这些共性操作合并, 同时减少持久化代码的冗余度, 设计了图 6 所示的持久层逻辑结构。其中 BaseDao 是采用泛型技术设计的所有 DAO 的父接口, 其中封装了所有实体类共有的持久化方法。泛型类 BaseDaoImp 是 BaseDao 的实现类, 通过关联的 SessionFactory 对象来获取 Hibernate 的 Session 对象实现相关的持久化逻辑; 在框架集成阶段已经将 SessionFactory 对象配置好了, 所以通过 @Resource 注解表明从 IOC 容器获取 SessionFactory。具体 DAO 接口从 BaseDao 继承, 具体 DAO 接口的实现类继承 BaseDaoImp 并添加 @Repository 注解。

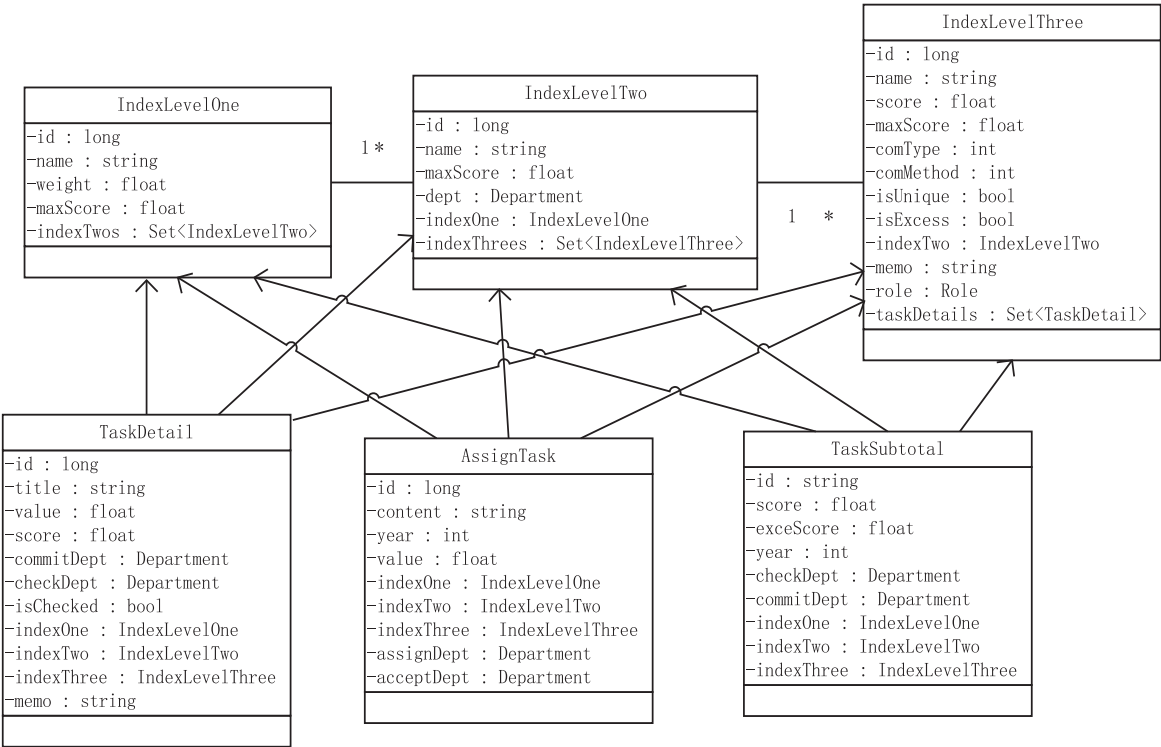


图 5 综合指标实体类关系

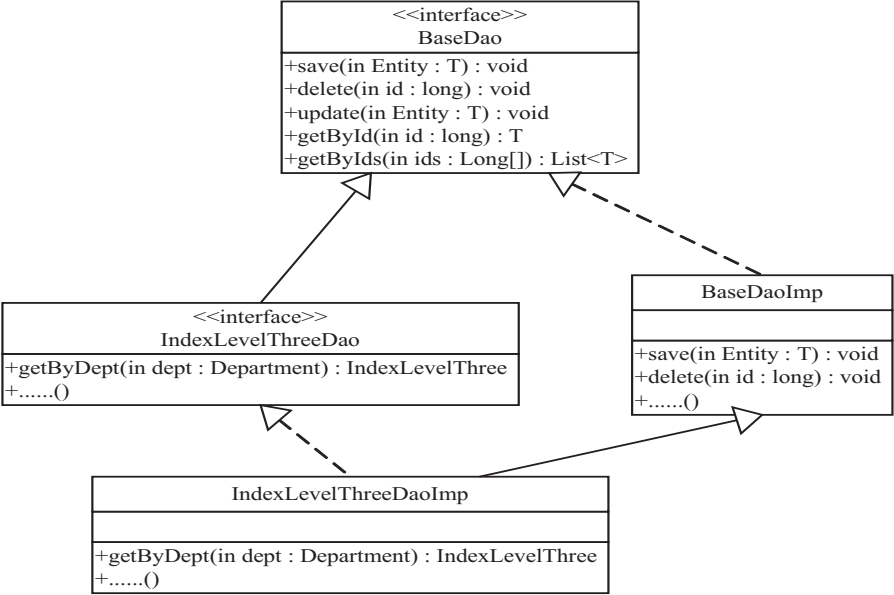


图 6 持久层逻辑结构

(3) 在业务逻辑层定义 Service 接口和对应的 Service 实现类,Service 实现类中通过注解声明所依赖的 DAO 接口。通过 Spring 的依赖注入在运行时创建 DAO 实现类对象,供 Service 调用。

(4) 在控制层定义 Action 组件并在 Struts2 的 XML 文件中配置。在 Action 中通过 @Resource 注解声明所依赖的业务逻辑层对象,利用 json-lib API 封装 JSON 格式的数据,通过 Easy UI 的 Ajax 异步处理 JSON 数据,达到 Web 页面无刷新地实现数据管理及访问^[14-16] 万方数据

(5) 在表示层创建 JSP 页面,通过 taglib 指令引入 Struts2 标签库,通过 Script 标签引入 Easy UI 对应的 JS 文件。利用 CSS+Div 进行页面布局,通过 Easy UI 标签提供数据输入和显示。创建 JS 文件完成对应 Easy UI 控件的编程,并以 Ajax 方式发送请求给 Action。

3.4 框架应用效果

该框架投入使用以来顺利完成了咸阳师范学院 2016 年的目标考核工作。实现了考核任务下达、任务完成数据上传、任务审核、考核结果计算等工作的网络

化管理,在相关职能部门和各二级学院之间达到了任务透明、数据共享的目的。同时为学校其他工作提供了方便获取数据的途径,例如科技处就可以从该框架导出年度学校科研数据上报教育部;在教师职称评审工作中,教务处可以通过该框架快速审核教师教学工作量等等。由于采用 Java EE 轻量级框架并以分层模型为基础,所以当需求变更时,只需少量代码修改即可进行功能扩展。例如,2017 年的目标考核工作将学生、党建模块的任务下达和审核都交给了相关职能部门完成,所以框架只需在权限模块进行适当调整,即可满足用户需求。各高校的目标考核工作大致相同,主要区别在于考核具体指标不同,框架提供了指标管理模块,所以只需调整考核内容,即可轻松地移植到其他学校的考核工作中。

4 结束语

在分析高等院校目标考核管理业务流程的基础上,结合 MVC 设计模式与分层架构,提出了一种目标考核信息化框架设计方案。该方案提高了考核过程中目标任务数据管理的精度,达到了职能部门与各二级学院之间数据及时更新与共享的目的,满足了目标考核过程中的业务需求。该框架实施运行以来,得到了高校用户较好的满意度。下一步的研究方向是在该框架基础上考虑与高校的其他管理信息系统进行整合,在不同系统之间消除数据冗余;在考核对象方面,加入二级学院内部按教师岗位类别进行任务考核。

参考文献:

- [1] 张宇. 中国高校教师绩效考核研究[J]. 中国经贸,2015(11):61-62.
- [2] 刘红霞,陆文迪. 改进的 MVC 设计模式的研究与应用[J]. 计算机工程与科学,2015,37(9):1688-1691.
- [3] 黄美林,马建华,李东. 基于 SSH 框架与泛型的通用分页方法设计与实现[J]. 计算机技术与发展,2012,22(1):

67-71.

- [4] 郭庆燕,张敏,杨贤栋. JQuery Ajax 异步处理 JSON 数据实现气象图片的显示[J]. 计算机应用与软件,2016,33(6):20-22.
- [5] 欧阳宏基,葛萌. 基于 S2SH 框架的煤炭企业生产统计管理系统[J]. 计算技术与自动化,2015,34(3):118-122.
- [6] 贺建峰,曹守军,杜珂珂,等. 基于 SSH 与 EXTJS 框架的计量服务平台设计与实现[J]. 哈尔滨理工大学学报,2016,21(1):27-30.
- [7] HONG Wenqiang, CHE Chao, ZHANG Qiang, et al. Hibernate combined TableViewer in the application of medicines inventory management system[J]. Computer Aided Drafting, Design and Manufacturing, 2014, 24(2):48-53.
- [8] 李杰. 基于 ORM 的轻量级数据持久化技术研究及应用[J]. 计算机科学,2010,37(9):190-193.
- [9] WU Peng, YIN Ke. Application research on a persistent technique based on hibernate[C]//Proceedings of international conference on computer design and applications. [s. l.]: IEEE,2010:629-631.
- [10] 陈正举. 基于 HIBERNATE 的数据库访问优化[J]. 计算机应用与软件,2012,29(7):144-149.
- [11] 王书怀,邢建春,李决龙,等. AOP 技术在管理信息系统中的应用研究[J]. 计算机应用与软件,2012,29(6):189-195.
- [12] WASHIZAKI H, KUBO A, MIZUMACHI T, et al. AOJS: aspect-oriented JavaScript programming framework for web development[C]//Proceedings of the 8th workshop on aspect, components and patterns for infrastructure software. [s. l.]: [s. n.], 2009:31-35.
- [13] 安沛,王春玲. OA 系统中 RBAC 扩展模型的研究与实现[J]. 西安工程大学学报,2015,29(1):78-83.
- [14] 李琦,姚龙. 基于 REST 架构的湖泊环境监测物联网平台[J]. 计算机工程,2016,42(11):27-31.
- [15] 孙光明,王硕. 基于 JSON 的 Ajax 数据通信快速算法[J]. 计算机应用与软件,2015,32(1):263-266.
- [16] SERRANO N, AROZTEGI J P. Ajax frameworks in interactive Webapps[J]. IEEE Software,2007,24(5):12-14.