

基站全生命周期管理平台的关键技术研究

王竹霞

(西安邮电大学 通信与信息工程学院, 陕西 西安 710061)

摘要:针对基站管理过程中过程复杂、数据繁多和责任界定困难等问题,设计开发能对基站管理进行环节管控和流程监控,扩展性好、安全性高的管理平台具有重要的现实意义和应用价值。基于基站全生命周期管理思想,确定基站全生命周期管理平台能实现我的待办、项目管理、统计分析、合同管理、材料管理和系统后台等主要功能。平台采用基础服务与个性化定制服务分离的开发模式,在技术架构中采用轻量级工作流引擎来管理和监控任务的执行。通过对基于 B/S 模式的基站全生命周期管理平台的关键技术研究,采用 Java EE 平台设计业务管理工作流,利用 Ajax 技术、Spring 框架和 Hibernate 框架设计并开发管理平台,并以项目管理模块为例说明具体实现过程。通过对平台的测试和试运行表明,该平台达到了预期的设计目标。

关键词:基站管理;全生命周期;工作流;B/S 模式;项目管理

中图分类号:TN925

文献标识码:A

文章编号:1673-629X(2019)01-0163-05

doi:10.3969/j.issn.1673-629X.2019.01.034

Research on Key Technology of Base Station Life Cycle Management Platform

WANG Zhu-xia

(School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China)

Abstract:In view of complex process, numerous data and difficult definition of responsibility in the process of base station management, it is meaningful and practical to design and develop a management platform with well scalability and high security which can control the link and monitor the process of base station management. Based on the whole life cycle management thought of base station, it is determined that the whole life cycle management platform of base station can realize to-do list, project management, statistical analysis, contract management, material management and system background. The platform adopts the development mode of separating basic services and personalized custom services. Through the research on the key technologies of the base station whole life cycle management platform, the business management workflow is designed by Java EE, the management platform is designed and developed by Ajax technology, Spring and Hibernate framework, and the specific implementation is illustrated by taking the project management module as an example. Through the test and trial operation of the platform, it is shown that the platform has reached the expected design goal.

Key words:base station management; whole life cycle; workflow; B/S mode; project management

0 引言

随着通信行业的迅猛发展,2014 年移动电话用户总数达到 12.86 亿,2015 年 4G 用户数超过 1 亿。伴随着用户数的增加,各大运营商加快了移动通信网络的建设和优化,仅 2014 年新建基站 98.8 万个,基站总数达到 339.7 万个。面对数量如此庞大的基站,通信运营商在基站管理过程中面临着过程复杂、基站数据繁多和监管责任界定困难等问题,这些问题将会导致

基站管理难度加大、管理效率降低,最终将影响运营商的发展。

通信运营商位于信息化发展的前沿,在通信工程项目管理平台建设方面取得了显著的成就。张迪等针对通信工程项目的管理责任不清、信息交流不畅、管理体制落后等问题,基于 B/S 系统架构模式,以 ASP.net 为软件开发环境,利用 SQL2005 作为系统数据库,设计和实现了通信工程项目管理信息系统^[1]。华豹等基

收稿日期:2018-02-28

修回日期:2018-06-06

网络出版时间:2018-11-15

基金项目:陕西省教育科学研究计划项目(16JK1704);陕西省科技工业攻关项目(2016GY-113)

作者简介:王竹霞(1978-),女,硕士,工程师,研究方向为网络优化和大数据分析与应用。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20181114.1556.038.html>

于项目流的全过程,利用流行的 IT 技术 J2EE、Linux 和 XML 开发完成了通信工程项目管理系统,该系统能有效地减少工作人员工作量,提高工作效率^[2]。陈中祥等分析了通信工程项目进度管理的基本内容和一般步骤,给出了基于 GIS 的工程进度管理系统的设计和实现方法,该系统具有强大的空间可视化功能,能对通信工程项目的进度情况进行高效的管控和调整^[3]。针对通信企业在通信工程项目管理中存在的流程复杂、历史数据多、时间跨度大和管理效率低等问题,兰勇超以 J2EE 作为系统的开发技术^[4-5],利用 MVC 模式进行系统架构设计,开发的通信工程项目管理系统能有效地管理企业的资源,简化业务流程,提高了业务处理效率,系统的财务报表生成功能,提高工程项目管理中财务统计的效率和准确性。在分析通信工程行业的特点和规律的基础上,给出了通信工程项目全生命周期的概念,研究了通信工程项目全生命周期中各阶段的流程和任务,为通信工程项目全生命周期的管理提供了对策和建议^[6]。基于基站全生命周期管理思想,研究了基于 B/S 模式的基站全生命周期管理平台关键技术,采用 Java EE 平台设计并开发了管理平台。

1 基站全生命周期管理

1.1 产品全生命周期管理

20 世纪 80 年代中叶,美国国防部提出了一项名叫计算机辅助后勤支援(CALS)战略性计划,开启了产品全生命周期管理研究的征程。产品全生命周期管理(PLM)作为一种先进的企业管理理念,是从产品的生产开始,直到产品的报废整个过程加以管理。全生命周期管理理念有助于对过程中的各个环节加以监控,也有利于对整个过程进行管控。

随着计算机技术应用的深入,通过设计开发产品全生命周期管理系统(PLMS)能使企业有效地实施 PLM 理念,提升产品管理的效率和信息化水平。沈建新等研究了基于 STEP 标准的产品全生命周期管理系统框架,该系统采用四层的体系结构^[7],由 CORBA 技术和 Web 技术加以实现。另外,许承东等基于 J2EE 框架,研究了 PLMS 的关键技术^[8],提出了具有六层的体系结构,在该系统的实现方面,PLMS 的关键技术可以由 CORBA 技术和 DCOM 技术实现。

1.2 基站全生命周期管理

基站全生命周期^[9-11]是指基站从需求分析、规划设计、建设、运行、维护和优化的整个过程。通常情况下,基站全生命周期管理(base station life cycle management,BSLCM)包括基站建设管理、基站运行管理、基站维护管理和基站优化管理等。基站的建设管理主要是指基站建设前期的对设计、规划的管理,基站的运

行管理是指建设过程中的项目管理、合同管理与材料管理,基站维护管理主要是指对基站实施过程参与人员和遗留问题的管理,基站优化管理主要是指依据基站项目实施过程中的统计数据提醒或调整项目的实施过程。

2 平台设计与关键技术

2.1 设计思想和目标

平台的开发设计以基站全生命管理为主线,采用基站管理的多视角多层次业务模型,达到动态扩展为目标进行平台的设计开发。该平台的设计目标是:

(1)以基站全生命周期管理为业务目标。

该平台以实现基站从需求分析、规划设计、建设、运行、维护和优化的生命周期的管理为目标。具体能实现基站全生命周期各环节的管理和控制,实现基站建设的流程监控,实现基站全生命周期中数据的共享,实现全生命周期各阶段业务管理。

(2)采用多视角多层次的业务模型。

在基站全生命周期管理过程中,设计的人员很多,为了将基站管理过程中的所有参与者纳入系统的管理范围,设计时采用多视角多层次业务模型,有助于相关人员明确业务界限。

(3)采用标准套件和个性化定制开发模式,使平台有良好的扩展性。

为了满足客户的需求,基站全生命周期管理平台采取以标准套件+个性化定制的开发模式。采用动态扩展平台,将二次开发和独立定制开发应用相隔离,以实现快速个性化定制开发,提高二次开发的效率,使平台具有良好的扩展性能。

通过对平台需求的分析,对用户的需求进行归纳整合,基站全生命周期管理平台由我的待办、项目管理、统计分析、合同管理、材料管理、遗留问题和系统后台七个模块构成,平台的总体功能架构如图 1 所示。

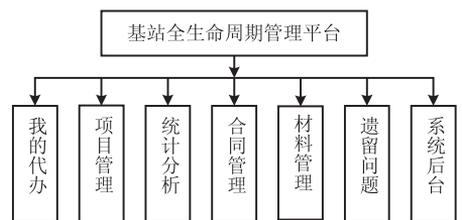


图 1 基站全生命周期管理平台总体功能架构

我的待办模块由我的待办、我的参与、我的开启子模块构成。项目管理模块由项目管理和基站管理子模块构成。项目管理子模块又包括项目列表、新建项目、转计划内和项目维护模块,基站管理子模块又包括基站查询、新建基站和基站维护模块。统计分析模块由基站数量统计、基站造价统计、承接单位、偏离度查询

和超期天数查询子模块构成。合同管理模块由合同管理、订单合同管理和租赁协议管理子模块构成。合同管理子模块又包括合同查询、合同维护、合同付款、合同审计和合同录入模块,订单合同管理子模块又包括土建订单查询、外电订单查询、土建第三次付款、外电第三次付款、土建合同录入、外电合同录入、土建合同维护和外电合同维护模块,租赁协议管理子模块又包括租赁协议查询和租赁协议付款模块。材料管理模块由材料领用、材料使用、材料归还、材料平衡、材料结算、清帐查询、材料维护和材料入库子模块构成。系统后台模块由组织机构、权限管理、流程管理子模块构成。组织机构子模块又包括部门管理、职位管理、职位类型、员工查询和员工维护模块,权限管理子模块又包括角色管理和权限分配模块,流程管理子模块又包括流程设计和流程监控模块。

2.2 平台关键技术

随着 Internet 技术的快速发展,B/S 模式成为 Web 应用中的架构模式,该模式实现了一次性到位的开发,能达到不同的人员、在不同的地方、以不同的连接方式对公共数据库访问和操作的目的是。因此,基于 B/S 模式的平台有助于实现基站全生命周期的管理。

在基站管理过程中,数据比较庞大,而 SQL Server^[12]是一个全面的关系数据库管理平台,具有应用方便、伸缩性强及集成化程度高的优势。它集成了商业智能工具为用户提供企业级的数据管理服务,可以对任何数据进行组织管理,具有可信任、高效及智能化的特点。因此,在平台数据库中采用 SQL Server 进行开发设计。为了使基于 B/S 模式的基站全生命周期管理平台独立于 Web 服务器软件,Internet 应用程序更小、更快、更友好,平台的开发采用创建交互式动态网页的开发技术 (asynchronous Javascript and XML, Ajax)。JavaScript 的对象 XMLHttpRequest 是 Ajax 的核心^[13],它支持异步请求,通过 JavaScript 向服务器提出请求并处理响应,但它不会阻塞用户。通过 XMLHttpRequest, Ajax 在浏览器不重新加载页面的情况下,实现与 Web 服务器的数据交换。Ajax 在浏览器与 Web 服务器之间使用异步数据传输方式,这样就可使网页从服务器请求少量的信息,而不是整个页面。在平台开发过程中,采用轻量级的 Java 开发框架 Spring, Spring 采用分层架构的形式。在 Java 语言下,采用 Hibernate 对象关系映射框架,以建立 Java 类与基于 SQL 的数据库表的映射关系,并具有获取数据和查询数据的功能,使得 Java 软件开发人员能随心所欲地利用对象编程思想对数据库加以操纵^[14]。

2.3 平台技术架构

为了实现对基站管理各环节和整个过程的管

理与控制,在流程开发方面采用 workflow 技术,依据 WFMC 发布的工作流系统模型,开发设计适合基站全生命周期管理的工作流引擎,基站全生命周期管理平台的技术架构如图 2 所示。

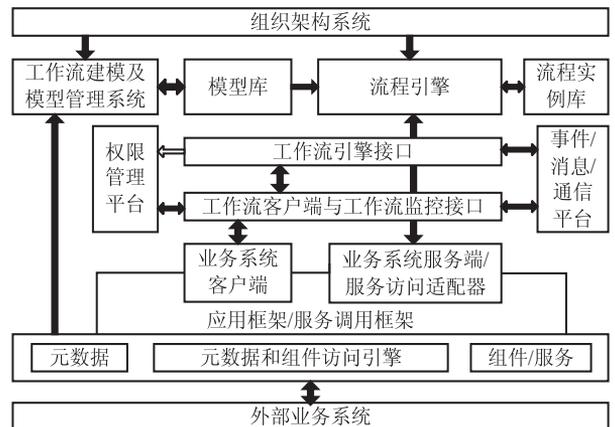


图 2 基站全生命周期管理平台的技术架构

在基站全生命周期管理平台开发技术方面,以基站全生命周期管理的业务流程为主线,基于 workflow 引擎技术实现基站管理业务过程的自动化和业务过程的重组^[15]。workflow 技术必须具备 3 个阶段:基站管理业务流程的确定通过映射来实现,业务流程的生成通过建模来实现,业务流程的跟踪和监控则通过管理来实现。基站全生命周期管理平台 workflow 运行环境由 workflow 客户端与 workflow 监控工具、workflow 引擎接口和 workflow 建模及模型管理系统等几部分组成。

2.4 设计与实现

基站全生命周期管理平台是基于 Java EE 开发设计的轻量级 workflow 引擎^[16]。平台的业务层实现基于 B/S 模式,平台的表示层则基于 Spring 框架,而业务数据的访问基于 Hibernate 框架。

2.4.1 详细设计

下面以项目管理模块为例说明详细设计。基站全生命周期管理平台的项目管理模块由项目管理和基站管理子模块构成。功能结构如图 3 所示。

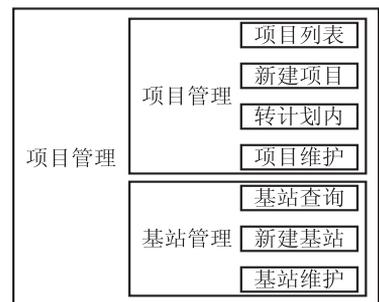


图 3 项目管理功能结构

项目管理子模块包括项目列表、新建项目、转计划内和项目维护模块。项目列表用于查看已有项目,可以根据项目编号、项目名称、立项年份、项目类型、项目

经理和专业归属等关键字进行查询。新建项目用于新建项目详细信息和相关资料的填写录入,支持新建项目相关文件的上传。转计划内可通过添加单项选项将已有的单项项目加入到某个大项项目当中,也可对某个项目的明细资料及其单项明细资料进行查看。项目维护支持对大项项目信息的查看、修改、CAPEX 更新和删除等操作。

基站管理子模块又包括基站查询、新建基站和基站维护模块,基站查询可以根据基站编号、基站名称、归属地和区域项目经理等关键字进行查询,并对某个基站的明细信息进行查询。新建基站用于新建基站信息的录入和基站建设流程的规划,支持流程查看以跟踪基站建设进度。基站维护用于对基站信息做相应的维护,支持基站明细信息的查看、修改和删除等操作。

2.4.2 数据库设计

在数据库设计方面,首先建立名为 unicomdb 的数据库,在管理平台的数据库中设计了基站信息表 a_m_bstation_info、项目信息表 a_m_project_info 等。下面给出项目信息表和基站信息表的字段信息,项目信息表(部分)见表 1,基站信息表(部分)见表 2。

表 1 项目信息表(部分)

Field	Type	Comment
Project_Id	varchar(32)	项目编号
Project_Name	varchar(32)	项目名称
Project_Time	varchar(32)	立项时间
Project_Type	varchar(32)	项目类型
...

表 2 基站信息表(部分)

Field	Type	Comment
DBID_	varchar(32)	主键 UUID
Base_Station_Id	varchar(32)	基站编号
Base_Station_Name	varchar(32)	基站名称
Attribution	varchar(32)	归属地
...

2.4.3 平台实现

项目维护支持对大项项目信息的查看、修改、CAPEX 更新和删除等操作。核心编码如下:

```
package com. xa. am. project;
public class ProjectAction extends BaseAction {
    @RequestMapping( value = "/project/modify. shtml" ,method =
    { RequestMethod. GET, RequestMethod. POST } )
    ...    万方数据
```

```
@RequestMapping( value = "/project/modifyProject. shtml" ,
method = RequestMethod. GET )
...
    @RequestMapping ( value = "/project/modifyProject. shtml" ,
method = RequestMethod. POST )
        public String saveModifyProject ( HttpServletRequest request ,
POAMProjectInfo projectInfo ) {
            ...
        }
    @RequestMapping ( value = "/project/capexUpdate. shtml" ,
method = RequestMethod. GET )
        public String setCapexUpdate ( String id , Model model ) {
            this. projectService. setCapexUpdate ( id , model ) ;
            return this. forward ( " capexUpdate. html" ) ;
        }
    @RequestMapping ( value = "/project/capexUpdate. shtml" ,
method = RequestMethod. POST )
        public String saveCapexUpdate ( String id , POAMProjectInfo-
projectInfo ) {
            this. projectService. saveCapexUpdate ( id , projectInfo ) ;
            return this. redirect ( "/"project/modify. shtml" ) ;
        }
    @RequestMapping ( value = "/project/deleteProject. shtml" ,
method =
    { RequestMethod. GET , RequestMethod. POST } )
        public String deleteProject ( String id ) {
            this. projectService. deleteProject ( id ) ;
            return this. redirect ( "/"project/modify. shtml" ) ;
        }
```

基站维护用于对基站信息做相应的维护,支持基站明细信息的查看、修改和删除操作。

核心编码如下:

```
package com. xa. am. station;
@ Controller
public class StationAction extends BaseAction {
    @RequestMapping( value = "/bstation/modifyStation. shtml" ,
method = RequestMethod. GET )
        public String modifyStationGet ( String id , Model model ) {
            this. stationService. modifyStation ( id , model ) ;
            return this. forward ( " modifyStation. html" ) ;
        }
    @RequestMapping( value = "/bstation/modifyStation. shtml" ,
method = RequestMethod. POST )
        public String modifyStationGet ( String id , POAMBstationInfo
bstationInfo , POAMLLatitudeInfo latitudeInfo ) {
            this. stationService. modifyStationUpdate ( id , bstationInfo , lati-
tudeInfo ) ;
            return this. redirect ( "/"bstation/modify. shtml" ) ;
        }
    @RequestMapping( value = "/bstation/delete. shtml" , method =
    { RequestMethod. GET , RequestMethod. POST } )
```

```
public String deleteStationGet( String id) {
this. stationService. delete( id);
return this. redirect( "/bstation/modify. shtml" );
}
```

3 平台验证

基站全生命周期管理平台开发完成后,对平台的正确性、安全性、完整性和实用性进行测试和验证。在对基站全生命周期管理平台测试时,采取按开发流程进行平台测试的方法,按照单元测试、模块测试和集成测试的过程对平台加以测试。在每个单元设计开发完成后,进行了单元测试,及时检查各个功能单元是否正确完成了规定的功能。以合同录入单元测试为例,重点测试了数据录入是否正确,合同附件能否上传等等。模块测试将经过测试的每个功能单元整合在一起构成不同的功能模块,检查功能模块内是否能正确完成规定的功能,信息交互是否正常。集成测试是对平台整体性能的测试,重点测试各模块之间的接口是否正常,这对平台的兼容性起到了关键的作用。在平台验证时,为了确保验证的全面性,在重点进行联网运行测试的基础上,也进行了大量的单机运行测试,以及时发现问题并加以处理。结合平台验证和试运行的情况可以看出,基站全生命周期管理平台提供的功能能满足客户需求,平台安全性、完整性较好,平台的可扩展性较高。当然平台也有一定的不足,比如有些功能模块存在重叠现象,平台在操作性上还有待完善等等。

4 结束语

针对基站的管理水平和信息化程度不高,基站管理过程中仍然面临着过程复杂、数据繁多和责任界定困难等问题,研究了能对基站管理进行环节管控和过程监控,可扩展性强、安全性高的基站全生命周期管理平台。首先研究了基站全生命管理过程,进行了平台需求分析。其次对基站全生命管理平台进行了功能设计、技术设计和数据库设计。在此基础上,研究了基站全生命管理平台的关键技术,采用 Ajax 技术、Spring 框架、Hibernate 框架和 SQL Server 技术,在开发模式上采用基础服务与个性化定制服务分离的模式,在技术架构中采用工作流系统建模设计流程的工作流引擎,实现了基于 B/S 模式的基站全生命周期管理 Web 平台。从测试和试运行的情况来看,该平台达到了设

计的目标。

参考文献:

- [1] 张迪. 通信工程项目管理系统的设计与实现[D]. 成都: 电子科技大学, 2013.
- [2] 华豹. 通信工程项目管理系统的分析与实施[D]. 北京: 北京邮电大学, 2008.
- [3] 陈中祥, 罗宁辉, 杨洁, 等. 基于 GIS 的工程项目进度管理系统设计[J]. 武汉理工大学学报, 2003, 25(7): 81-83.
- [4] 兰勇超. 基于 J2EE 和 MVC 的工程项目管理系统的设计和实现[D]. 西安: 西安电子科技大学, 2007.
- [5] 王家骥, 于海霞. 基于 MVC 设计模式的 Web 应用框架研究[J]. 计算机与信息技术, 2006(11): 8-10.
- [6] 崔丽. 基于项目生命周期的通信工程项目管理[J]. 中国科技投资, 2012(30): 86-87.
- [7] 沈建新, 周儒荣. 产品全生命周期管理系统框架及关键技术研究[J]. 南京航空航天大学学报, 2003, 35(5): 565-571.
- [8] 许承东, 赵向领, 宋伟. 产品全生命周期管理系统的关键技术和系统层次结构[J]. 北京理工大学学报, 2006, 26: 9-12.
- [9] 孙维, 林振辉, 舒华英. 电信运营客户全生命周期管理方法探讨[J]. 移动通信, 2005, 29(2): 107-110.
- [10] 姚伟. 3G 基站建设与维护[M]. 北京: 机械工业出版社, 2012.
- [11] HUNKELER D, REBITZER G, JENSEN A A, et al. Life cycle management: bridging the gap between science and application[J]. International Journal of Life Cycle Assessment, 2001, 6(6): 384-390.
- [12] 魏江来. 科研管理系统数据库设计与实现[J]. 山西科技, 2009(3): 25-26.
- [13] MARCHETTO A, RICCA F, TONELLA P. A case study-based comparison of web testing techniques applied to AJAX web applications[J]. International Journal on Software Tools for Technology Transfer, 2008, 10(6): 477-492.
- [14] WIERINGA R, DANEVA M. Six strategies for generalizing software engineering theories[J]. Science of Computer Programming, 2015, 101: 136-152.
- [15] AALST W M, TER HOFSTED E A H. Workflow patterns put into context[J]. Software & Systems Modeling, 2012, 11(3): 319-323.
- [16] 于群英, 李媛, 杨文荣. 基于轻量级 J2EE 的网站群管理系统开发架构[J]. 计算机技术与发展, 2011, 21(4): 48-51.