

# 山西省邮政综合服务平台代收付太保保费系统的实现

田丽娟<sup>1,2</sup>

(1.太原理工大学计算机学院,山西太原,030024;2.山西省邮政信息技术局,山西太原,030001)

**摘要:**介绍了邮政综合服务平台代收付太保保费系统建设项目的背景以及业务流程,概述了系统业务功能的具体实现。

**关键词:**邮政综合服务平台;代收付保费系统;业务流程;业务功能

**中图分类号:**F616.3

**文献标识码:**A

代收付太保保费系统是属于邮政综合服务平台上运行的子业务之一,属于邮政综合服务平台的一部分,也可以称作是一个独立的子系统。

## 1 项目背景

基于邮政储蓄基础上的各种代理业务已经纷纷上线运行,并且为邮政储蓄的发展提供了有力的支撑。现在新的代理业务仍在不断地研究开发中,邮政信息化的建设正在加快发展步伐,中间业务成为邮政储蓄新的利润增长点。

现在很多保险公司推出了多种针对个人的保险业务,需要定期向用户收取保费或者支付赔偿,由于用户分散、覆盖面积大,上门收费用很不方便,于是很多保险公司就与各大银行合作,通过银行进行批量的代收付业务。邮政储蓄银行凭借着网点覆盖面大、遍布全国各大城市以及农村的特点,在各大银行的中间业务竞争中具有绝对的优势。

中国太平洋人寿保险股份有限公司山西分公司(以下简称太保)与山西邮储为了实现双方业务的共同发展、资源共享、优势互补,太保决定利用邮储计算机联网系统批量代扣个人寿险首、续期保费及代付保险金。

## 2 代收付太保保费业务整体流程

代收付太保保费业务整体流程基本上就是太保公司将与公司有购买保险产品或者需要进行理赔的客户的客户信息、交易信息等要素按照预先约定的格式文件提供给邮政储蓄银行各地市中心,由综合服务平台进行处理。综合服务平台负责身份的校验,与储蓄核心主机联网进行扣款、付款以及交易的加密和统计分析。综合服务平台按照太保提供的需求付、扣款信息反馈太保,具体流程见图1。

## 3 代收付太保保费系统的设计和实现

由于太保业务地市中心集中的特点,代收付太保保费系统采用业务数据地市中心统一加载、全省数据集中存储、统一处理的设计思想。利用实时的数据传输,将全省的代收付数据上传到省中心,在省中心统一存储。

代收付太保保费系统的开发工作主要包括省中心后台业

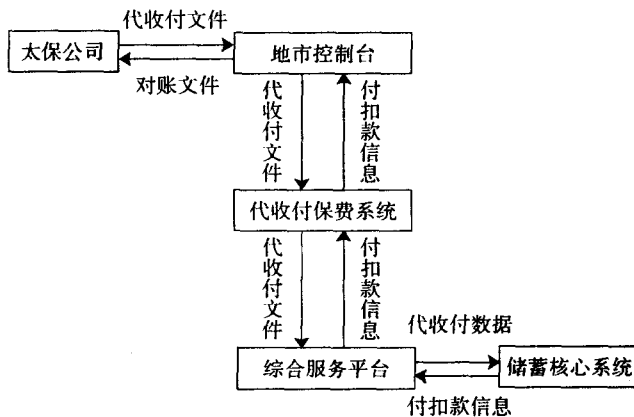


图1 代收付太保保费业务整体流程

务、会计开发以及地市控制台的开发。山西省邮政综合服务平台后台开发环境采用上海华腾公司研发的TSA+平台,由于此次开发的系统与已有系统完全不同,需要在原有平台上进行二次开发。

### 3.1 控制台操作员登录模块的实现

业务控制台的操作对象是省中心、市县操作员,出于安全的考虑,操作员使用控制台前必须进行登录,并且系统将根据操作员的所属机构编号、用户代码及用户口令对操作员进行验证以及权限的限定,确保不同权限的用户登录后,可见的只是他有操作权限的菜单,而其余的菜单是会被隐藏的。

操作员的登录操作模块是借助struts来实现的,是一个典型的MVC Model2模式。

### 3.2 控制台操作菜单模块的实现

操作员登录控制台后,系统将会根据他们各自的权限显示不同的操作菜单,并且拥有所有以上权限并可启用注销其他操作员的管理员权限。

#### 3.2.1 操作菜单模块中的类简介

(1)MenuTree:表示整个菜单的树型结构,它通过一个xml文件来初始化菜单结构,记录根菜单(rootMenu),并将所有子菜单Put到一个静态的LinkedHashMap结构中,它可以通过getRootMenu()方法返回rootMenu,也提供getChildren()方法返回一个Iterator来遍历整个menutree。

(2)Menu: 表示一个菜单项,它通过一个 JDOM 的 Element 实例来初始化自己的同时以递归的方式遍历它的所有子菜单项,并将它们 put 进 MenuTree 的静态成员 LinkedHashMap 中去。

(3)SimpleMenu: 是 Menu 类的简化版本,它同样表示一个菜单项,但只是完成初始化这个菜单项的属性值。

### 3.2.2 JDOM 在控制台中的应用

出于代码可读性、可维护性、可重用性等方面的考虑,将操作菜单的所有内容结构以规范的 XML 形式来描述,并保存在一个单独的 XML 文件 menutree.xml 文件里,其形式如下例所示:

```
<menu id="A" title="菜单" tip="" type="" target="" image="">
  <submenu>
    <menu id="AA" title="代收太保" tip="" type="" target="" image="" />
    <menu id="AB" title="会计清算" tip="" type="text" target="" image="">
      <submenu>
        <menu id="ABA" title="清算业务流水查询" ..... />
        <menu id="ABB" title="清算账户明细查询" ..... />
      </submenu>
    </menu>
  </submenu>
</menu>
```

可见 menutree.xml 形成的树型结构恰好形象地映射成操作菜单结构图。当控制台启动的时候,menutree.xml 文件就会被自动加载,这一工作是由一个名为 OnStartupServlet 的 servlet 来完成的,在 web.xml 文件中,将这个 servlet 配置成当 web 项目启动时自动执行:

```
<servlet>
  <servlet-name>onstartupservlet</servlet-name>
  <servlet-class>com.huateng.servlet.OnStartupServlet</servlet-class>
  <load-on-startup>2</load-on-startup>
</servlet>
```

在 OnStartupServlet 类被执行的时候[作为一个 servlet,其实是它的 init()方法被执行],通过定位 menutree.xml 文件的路径,将其加载,此时,JDOM 的存在,使得工作量大大减轻,代码也变得非常简洁明了。

### 3.3 后台服务器与控制台的协作

综合服务平台系统的后台服务平台运行在省中心机构,网点交易局只是通过网点客户端与省中心的后台服务平台进行通信。网点及省中心控制台通过网络通信,向后台服务平台发送严格约定格式的报文,后台服务进程收到报文后通过创建子进程来完成交易所需的事务处理,完成后杀死子进程,然后同样将成功或失败信息以报文形式发还给网点和控制台。

后台服务平台处理的事务大致分为联机交易和批处理交易。

#### 3.3.1 联机交易

联机交易是需要在线、及时、准确、迅速处理的交易。对于联机服务,处理的是由控制台发起的一个个有序的报文,通过后台服务器运行平台的翻译和解释,从接口报文中确定相对应的服务程序,然后通过调用相应服务程序的接口,将报文交给服务程序处理。处理完成后及时将成功或失败信息组织成有序的报文发回给控制台。

#### 3.3.2 批处理交易

批量服务处理的对象是大批量数据的操作,后台服务器提供了一套关于批量处理的机制,采用多线程、支持断点恢复的方式。通过控制台发送报文到后台调用发起一批处理,在本交易日结束时进行批处理,并在数据库表中记录批处理成功或出错状态,次日控制台通过交易触发从后台取回处理明细。

### 3.4 后台数据库的设计

后台数据库设计是系统设计的重要环节,设计的好坏直接关系到系统的质量、实现的难度、可维护性和工作效率。在这个项目中,由于系统在前期完成了部分代收付业务,现在的数据库表已经完全可以满足需求,不需要增加新的数据库表结构。

## 4 结语

日前,山西省邮政综合服务平台代收付太保保费系统运行稳定,并在后期的工作中体现出了良好的系统可扩展性以及维护性。代收付太保保费系统的实现为省邮政发展中间业务提供了一条切实可行的途径。

(责任编辑:张红)

第一作者简介:田丽娟,女,1982年6月生,2004年毕业于太原理工大学(本科),现为太原理工大学计算机科学与技术专业2006级在职研究生,助理工程师,山西省邮政信息技术局,山西省太原市五一东街2号,030001。

## Discussion on the Implementation of Pacific Life Insurance Company's Premium Collection and Payment System Based on Shanxi Postal Integrated Service Platform

TIAN Li-juan

**ABSTRACT:** This paper introduces the project background of the construction of Pacific Life Insurance Company's premium collection and payment system based on Shanxi Postal Integrated Service Platform and the operation flow of this system, and expounds the concrete implementation of operational functions of this system.

**KEY WORDS:** postal integrated service platform; premium collection and payment system; operation flow; operational function