

一种基于 Web 的层次编码处理模块设计

杨振东, 何宏, 刘光霆

(后勤工程学院 基础部, 重庆 401311)

摘要: 针对目前大多数 Web 应用系统对层次编码处理相对简单导致用户操作不便的问题, 开发了一个基于 Web 的层次编码处理模块, 详细介绍了其设计与实现思想。其操作简便, 易实现, 对开发 B/S 结构的管理信息系统在处理层次编码时, 提供了一个很好的解决方案。

关键词: 层次编码; Web; 应用系统; Ajax

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2011)07-0013-03

A design of level encoding processing module based on Web

Yang Zhendong, He Hong, Liu Guangting

(Dept. of Basic, Logistical Engineering University, Chongqing 401311, China)

Abstract: Because the level encoding process in the most current Web application is relatively simple, it is inconvenience for user. The author developed a level encoding processing module based on Web. The article introduces its ideas of design and implementation which is simple and easy to achieve. It provides a good solution to the development of management information system based on web while designing the level encoding processing module.

Key words: level encoding; Web application system; Ajax

在管理信息系统(MIS)中,经常涉及到对一些基本信息的管理。为了有效地组织、管理这些信息,开发者往往通过数据编码来确保信息的检索、分类、统计和便于控制,以提高信息的处理效率和利用价值^[1]。编码设计及其管理是信息系统中不可缺少的重要组成部分。在实际工作中,经常遇到一些数据之间具有从属或上下级关系的情况,例如机构编制、物资分类信息等都具有明显的上下级和从属关系。这类数据的编码就是一种层次编码。相比较传统的基于 C/S 的应用系统,基于 B/S 的 Web 应用系统灵活、易维护、成本低、使用方便、容易被用户接受,已成为企业应用的首选^[2]。本文以一种典型的层次编码为例,设计并实现了一个基于 Web 的简洁美观、功能强大的单位编码帮助模块。该模块界面如图 1 所示。

1 单位编码帮助模块功能

如图 1 所示界面,左边显示上级单位的编码和名称,并用分隔符分开,右边显示左边单位的直供单位或下级单位的编码和名称。下部是“返回上级”、“下级单位”、“确定”、“取消”按钮。选中左边某单位,然后单击



图 1 单位编码帮助模块一

“下级单位”按钮,界面显示如图 2 所示。

左边自动显示所选单位的下级单位,右边则是左边默认选中单位的下级单位,这样逐层展示每层的单位编码和名称,层次结构清晰,便于用户查找。

2 模块实现的主要技术

用户在操作某张页面时,可能多个地方需调用帮助模块,如果每次都提交后台处理,则在服务器上的程序处理页面并返回数据期间,用户只能等待。这样的 Web 应用程序因用户得不到立即的反应,操作习惯和思路都

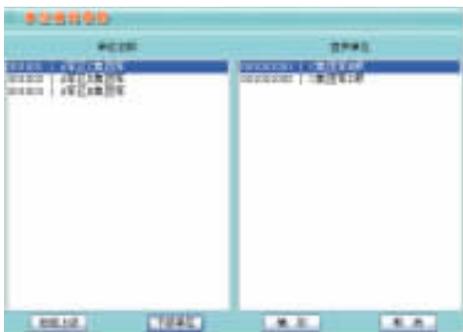


图2 单位编码帮助模块二

会在每一次提交-等待中被打断,交互性比较差,达不到桌面应用程序的效果。因而设计时主要考虑两个问题:是利用 Ajax 技术异步调用数据,还是直接把数据下载到客户端进行处理;如何按层次展示数据。Ajax 使用 XMLHttpRequest 对象与 Web 服务器进行异步数据交换^[3]。但当数据量比较大时,异步获取数据需较长的等待时间,因此可以把数据一次性下载到客户端再进行分层处理。

2.1 服务器端代码

首先,在数据库中建立一个 Dwxx 表,用 dwbm 和 dwmc 两个字段对应存储单位编码与单位名称,以 Java 平台为例,首先可以新建一个 POJO 类 Dwxx:

```
public class Dwxx implements java.io.Serializable {
    private String dwbm;
    private String dwmc;
    public void setDwbm(String dwbm){
        this.dwbm = dwbm;
    }
    public String getDwbm() {
        return dwbm;
    }
    .....
}
```

然后,定义一个 DwxxService 类来组装数据,以 ArrayList 的形式返回客户端进行处理:

```
public class DwxxService {
    //获取所有单位信息
    public ArrayList getDwxx () throws SQLException {
        .....
        ArrayList list = new ArrayList();
        String sql ="select dwbm,dwmc from sys_dwxx order
            by dwbm";
        rs=mydb.executeQuery(sql);
        while (rs.next()) {
            Dwxx dwxx = new Dwxx ();dwxx.setDwxxDwbm
                (rs.getString(" dwbm"));
            dwxx.setDwxxDwmc(rs.getString(" dwmc"));list.add
                (dwxx);
        }
    }
}
```

```
.....
    return list;
}
}
```

2.2 客户端代码

客户端处理代码封装成模块且置入主界面(如 index.htm)中。用户登录系统进入主界面后自动下载单位编码并生成帮助系统隐藏在客户端,载入其他需调用单位编码帮助功能的页面时就不必重复去后台下载单位编码,而是把已生成的模块由隐藏变为可见,这样大大提高层次编码的处理速度。代码以通用的 JavaScript 代码为主,再辅以少量服务器端脚本。核心代码如下:

```
<SCRIPT LANGUAGE="JavaScript">
/** 单位信息以三个层次为例,用三个数组分别存储各层
数据,提高了处理效率,更多层次情况做类似处理。*/
var dwxxarr1 = new Array(); var dwxxarr2 = new Array();
var dwxxarr3 = new Array();
var listl=document.dwform.listl; var listr=document.dwform.
listr
// listl,listr 分别为左右边下拉列表
function dwxx(dwbm, dwmc){ //构造数据
    this.dwbm=dwbm;this.dwmc=dwmc;
}
function getDwxx(){<% //功能:接收单位编码和名称,根据
//编码长度分别存入到三个数组
    Dwxx service =new DwxxService ();Dwxx dwxx =null;
    ArrayList list=new ArrayList();
    String dwbm="",dwmc="";list=service.getDwxx ();
    for(int i=0;i<list.size();i++){
        dwxx =(Dwxx)list.get (i);dwbm =dwxx.getDwbm ();
        dwmc=dwxx.getDwmc();
        switch(dwbm.length()){ //假设单位编码三位为一级
            case 3:
                %> dwxxarr1 [dwxxarr1.length]=new dwxx ("<%=
                dwbm%>","<%=dwmc%>"); <% break;
                .....case 6 与 case 9 做类似处理
            }%>
        }
    }
}
function nextdw(){ //功能:处理下级单位
    var selectedvalue= listl.options[listl.selectedIndex].value
    switch (selectedvalue.length){
        case 3:
            removeOptions(document.dwform.listl);
            //调用清除 option 选项函数
            for(i=0;i<dwxxarr2.length;i++){
                if (dwxxarr2 [i].dwbm.substring (0,3) ==
                selectedvalue)
                    listl[listl.length]=new Option(dwxxarr2[i].
```

```

dwbm+"|"+dwxxarr2[i].dwmc, dwxxarr2[i].dwbm);
    listl.options[0].selected=true;
    addOptions(listl.options[listl.selectedIndex].value);
        //调用增加 option 选项函数
    ..... //case 6 与 case 9 做类似处理
}
}
function priordw(){ //功能:处理上级单位
    var selectedvalue= listl.options[listl.selectedIndex].value
    switch (selectedvalue.length){
        case 6:
            removeOptions(listl);
                //调用清除 option 选项函数
            for(i=0; i<dwxxarr1.length; i++){
                listl[listl.length]=new Option(dwxxarr1[i].
                    dwbm+"|"+dwxxarr1[i].dwmc, dwxxarr1[i].dwbm);
                if (dwxxarr1 [i].dwbm ==selectedvalue.substring
                    (0, 3))
                    listl.options[listl.length-1].selected=true;
            }
            addOptions(listl.options[listl.selectedIndex].value);
            ..... //case 9 做类似处理
        }
    }
}
</script>

```

参照以上设计思路,可以实现如图3所示的单位编码维护模块。由于篇幅所限,就不再给出处理代码。

编码处理是构建优秀、完善的管理信息系统的重要功能,其效率的高低直接关系到整个系统的性能。通过以上设计,完成了一个高效的层次编码处理模块。该模块结构简单,稍做修改就可以在.NET平台下实现,相对于大部分系统只列出一个层次编码的下拉框来说,更符



图3 单位编码维护模块

合用户使用习惯,突出特点是对大数据量层次编码的处理也非常迅速,目前已在多个大型管理信息系统中采用,受到用户一致好评。此外,本文所提出的开发思路对处理其他各种复杂的层次编码也起着良好的借鉴作用,完全可以运用于供货商编码、地区编码、油品编码等处理。

参考文献

- [1] 康业娜,许勇,魏士伟.物流信息系统编码模型研究[J].微计算机信息,2007,36(2):59-60.
- [2] 郭晓峰,姚世军,尹祖伟.基于.NET的Web应用框架的设计与应用[J].计算机工程与设计,2008,29(2):454-455.
- [3] 季强,田薇,陈远.基于Ajax技术的WebOS系统NixOS的构建[J].计算机工程与设计,2008,29(5):1198-1199.

(收稿日期:2010-11-07)

作者简介:

杨振东,男,1981年生,硕士,讲师,主要研究方向:软件工程、网络及数据库技术。

何宏,男,1965年生,教授,主要研究方向:系统工程、网络及数据库技术。