



1 引言

目前,湖北全省的电厂、变电站的所有这样的发电、输变电数据每隔15分钟采集一次,送往调度局的数据库服务器,调度局对这些数据进行分析,再根据计划等各种情况向电厂、电站发出调度指令来完成调度任务。

为了使省电力各部门内部能够实时得到这些数据,我们运用ASP技术将数据发布于Internet上,用户在任何地方只要有一台浏览器就可以发送数据请求,在网上得到实时的数据报表。



2 系统实现

根据用户提供的条件从数据库中提取需要的数据,再将这些数据处理、求和后格式化为HTML语言生成报表形式发布在网上。下面,我们首先把系统所用的数据库介绍一下,然后将把典型报表、定制报表的实现分开来详细介绍。

2.1 数据库说明

本系统使用了Oracle数据库,该数据库是供整个省局使用的,有数百个数据表,数据量非常庞大。由于本系统只与电站的几种数据查询发布有关,所以只用到了其中的几个表格,但数据量也非常大。其中,

(1) info-field表是与站名信息有关的,它的结构见表1:

其中,BM代表“编码”字段,STATION代表“站名”字段,MC代表“数据点名称”,

湖北省电力高度信息发布系统设计与实现



殷勇 娄卫华 黄河潮 (武汉大学机械工程系 430072)

摘要:本文运用ASP技术实现了电网数据的网上实时发布,用户提取自己需要的数据,同时在页面中对数据进行统计运算,使用户不必再对数据库中的原始数据进行二次操作就可以看到统计结果;允许用户进行报表的定制,对不同数据类型进行各种组合,大大提高了系统的灵活性,还运用客户端语言DHTML提高了系统的可操作性。

关键词:ASP ADO 数据库 定制报表 DHTML

TABLENAME代表该数据类型存放的数据表, FIELDNAME代表该数据在表中的字段名, TYPE代表数据类型, JC代表在显示报表时的简称,每个数据点都有唯一一个编码号,如上面一行代表武昌电厂110KV2#母线电压编码为010019,它存放于rb-dy表中的wcdc2字段下,简称为“武昌110KV2”。

(2) info-virtual-field表是与典型报表中的求和运算有关的,它的结构见表2:

其中,NO是要显示的数据编号, LB为“类别”字段, MC为在报表中要显示的名称, CALC下的字符串即为需要求和的各个数据点的编码以加减公式表示出来。

(3) rb-dy表是与各数据点的电压值有关的,它的结构见表3:

其中, RQ代表“日期”字段,以1998年7月1日为0,之前每天减一,之后每天加1; SJ代表“时间”字段,0:00为1,隔15分钟加1,全天共96。WCDC1、WCDC2为数据点,

后面还有数十个数据点,这里略去。

(4) rb-ygcl、rb-wgcl、rb-dqfh表分别与有功出力、无功出力、地区负荷有关,结果与rb-dy类似,这里略去。

2.2 打开数据库

本系统的难点在于“典型报表”与“定制报表”这两种报表的实现,这将在下面两节中探讨。这里主要介绍一下数据库的情况。其实对数据库的提取操作并不难,主要就是利用ADO对象和SQL进行数据提取。例如,“定制报表”中,我们用

```
<% set cn=server.CreateObject("ADODB.connection")
cn.ConnectionString=" dsn=oracle"
cn.Open%>
```

来打开一个数据库连接, oracle是我们在控制面板中ODBC数据源中建立的一个DSN(数据源名称),再用

```
<% set rs=server.CreateObject("ADODB.recordset")
rs.Open "select jc,fieldname,tablename,type from info-field",cn%>
```

来打开一个Recordset,这里是从info-field表中得到jc,fieldname,tablename,type字段的值,再用

```
<%set rb-dy=server.CreateObject("ADODB.recordset")
rb-dy.Open "select rq,sj" & fieldstring (i) & "from rb-dy where rq=" & nday & " order by sj",cn%>
```

打开rb-dy电压表得到数据放在记录集rb-



表1

BM	010019
STATION	武昌电厂
MC	110KV2# 母线电压
TABLENAME	rb-dy
FIELDNAME	wcdc2
TYPE	电压
JC	武昌110KV2

 表 2

NO	17
LB	有功出力
MC	武昌
CALC	' 010001' + ' 010003' + ' 010023' + ' 010026'

dy 中 (其中 nday 是需要观看的日期, fieldstring 是各个站点在电压数据表中的字段名的字符串之和, 这两个值都是使用 ASP 在服务器端用 VBScript 求出的, 这些细节将在后面讨论)。

然后, 将得到的这些数据嵌在 HTML 语句中, 作为 TABLE 的一个 TD 发出, 显示出报表。例如,

```
<td align=cnrter bgcolor=" #DFF3FF"
width=<%=width1%>><%=rb-dy
(fieldname0(t).value%></td>
```

其中, width1 是已求出的 td 宽度, fieldname0(t) 就是对应各个站点在电压数据表中的字段名的字符串, t 是一个循环变量, 所有的 fieldname0 之和就是我们上面提到的 fieldstring。

2.3 典型报表的实现

由于本系统实质上是完成日报表显示功能, 那么为了从数据库中正确提取得到用户指定的某年月日的数据 (有效日期为 98 年 1 月 1 日至现在), 系统首先要能够让用户任意选择日期以实现更大的灵活性; 其次, 由于数据库日期 (RQ) 字段的数据是以整型表示的, 即 1998 年 7 月 1 日为 0, 1998 年 7 月 2 日为 1, 7 月 3 日为 2, 而 6 月 30 日为 -1, 6 月 29 日为 -2 等等, 以此类推, 我们就要将这些 0、1、2 转换为具体日期。

为了实现这样的功能, 我们用 HTML 中的 Select 下拉框来让用户进行日期的选择。要得到正确的日期, 只要使用这样的函数即可:

```
request('lday'))
```

```
getdate=formatdatetime (getdate,
vbLongDate)
end function%>
```

但是, 为了转换成数据库中的形式 (2000 年 5 月 16 日在数据库中为 655), 我们又在 HTML 中嵌入了这样的函数,

```
<%function num2date (num1)
num1=cint (num1)
num2date=dateadd ('d',num1,' 1998-7-1')
end function
function date2num (date1)
date2num=datediff ('d',' 1998-7-1',
date1)
end function%>
```

其中, num2date 将数字转换为日期, date2num 反之。

在得到日期以后, 下面最重要的任务就是得到要显示在报表中的各站名字段, 以及需要求和的表达式, 如武昌的有功出力计算公式为 '010001' + ' 010003' + ' 010023' + ' 010026', 也就是说, "武昌" 字段的值是四个数据点的和, 这四个数据点在数据库中的编码为 0 1 0 0 1, 0 1 0 0 3,

```
<%function
getdate()
getdate=dateserial
(Request
('lyear'), re-
quest
('lmonth'),
```

010023, 010026, 那么, 要得到武昌的有功出力必须根据编码在数据库中找到这四个数据点, 再求和即可。

2.4 定制报表的实现

由于考虑到用户可能需要查看某个具体站点 (如武昌电厂、阳逻电厂等)、甚至某个数据点的值 (如武昌电厂 110KV1 号母线电压、110KV2 号母线电压, 1 号、2 号、3 号机有功出力等)。对于这点, 我们是这样实现的: 提供两种方案让用户选择, 一是通过列出站点, 让用户先选出他所要看的站点名, 再选出该站点下的数据点, 二是列出四种数据类型, 即电压、地区负荷、有功出力、无功出力, 用户选出类型后, 系统显示所有站点下的该种数据类型, 用户再进行选择。

下面以 "根据类型定制报表" 进行说明。

在定制报表界面中, 我们用三个多选 SELECT 框存放各种选项, 第一个作为数据类型框, 第二个作为数据点框, 第三个存放用户已选择的数据点。

将四种数据类型值放入数据类型框中是这样实现的:

```
<tr width=100% align=center><td
`height=90%>
<select id=lx size=5 style=" position:
relative;width=230;height=100;"
onclick=" show(lx.value)">
<%while not rs-infocfield1.EOF%>
<%typearray (selectnum)=rs-infocfield1
```

 表 3

RQ	655
SJ	1
WCDC2	116
WGDC1	117
...	...

```
(`type`).value%>
<option value=<%=rs-infocfield1 (`type`)
%>><%=rs-infocfield1 (`type`)%></
option>
<%rs-infocfield1.MoveNext%>
<%selectnum=selectnum+1%>
<%wend%>
<%rs-infocfield1.MoveFirst%>
</select>
</td></tr>
```

“数据点”框中的值是在服务器端得到并一次发送到客户端的。点击每种数据类型后就显示该类型对应的数据点。这种功能是通过HTML语言中的层(DIV)实现的,每种数据类型对应一个层,所有的层重叠在一起,需要哪种数据类型就将哪层显示出来,而把其他层隐藏起来。

用户点击数据类型后显示对应数据点的函数show()代码如下:

```
<script language=vbscript>
sub show (value1)
<%rs-infocfield1.movefirst
while notrs_infocfield1.EOF
response.write " document.all ("
response.write rs-infocfield1 (`type`),
value
response.write """).style.visibility=""
hidden""
response.write chr(10)
rs-infocfield1.movenext
wend
%>
document.all ("blank").style.visibility=""
hidden"
document.all( value1).style.visibility=""
visible"
end sub
```

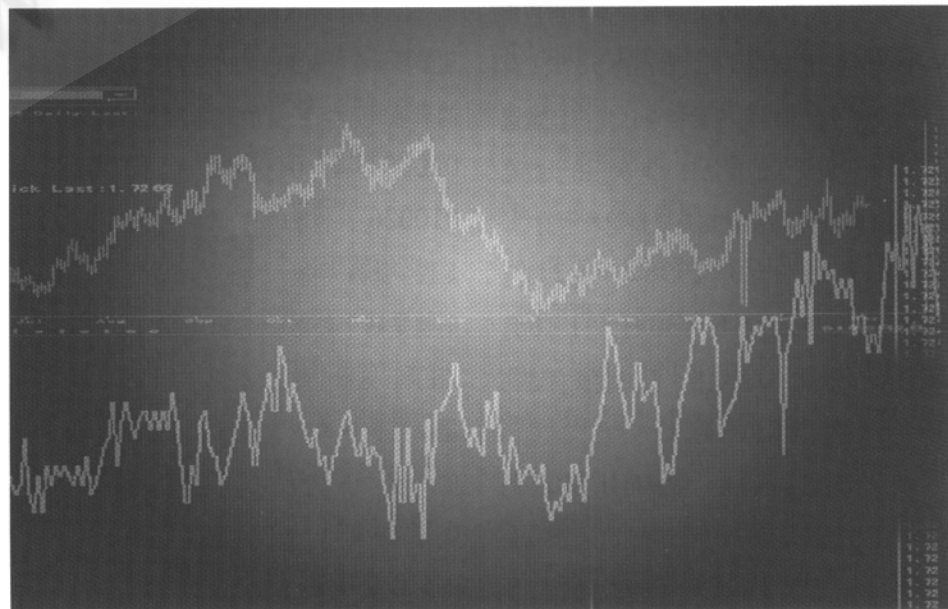
```
</script>
```

用户单击数据点将该数据点加入右边的“您已选择的数据”框中,函数addtoright()代码如下:

```
sub addtoright(value1)
i=document.all ("results").options.lenght
for k=0 to i-1
if document.all ("results").options(k).
value=value then
msgbox "对不起,您已经选择了这项数据",
vbExclamation,"提醒"
exit sub
end if
next
set opt=document.createElement
("OPTION")
opt.value=value1
opt.text=value1
form1 ("results").add opt
set opt=nothing
end sub
```

到此为止,用户完成了选择数据点的工作,下面只要点击“生成报表”即可。

“生成报表”功能是通过另一个ASP页面处理的,它的功能实现与“典型报表”基本相同,由于篇幅原因,这里从略。



3 几点说明

本系统的开发运用了多种技术,像上述的ASP、ADO、SQL语句,另外还大量使用了DHTML语言技术,从而更方便用户的使用。举两个例子:

(1) 在本系统使用的数据库中,1998年7月1日以前的每天数据是每小时采集一次,7月1日以后是每十五分钟采集一次,那么在用户选择每天24点还是96点时,系统在客户端就要进行判断并进行自动修改,而不能等到发送到服务器以后再由服务器来发出错误声明。而且,当用户选择了今天以后的日期时,系统也应该报错。

(2) 在显示报表时,如果数据点较多,报表横向长度会超出屏幕,此时要能够让报表头、报表脚在移动滚动条时始终保持在中间。这个功能也是通过DHTML实现的,首先将报表头、脚定义为层,然后在<body>中添加事件函数onscroll="scroll1()"

```
sub scroll1() `当scrollbar滚动时将报表头及
脚注固定不动
divhead.style.left=document.body.
scrollleft
divfoot.sytyle.left=document.body.
scrollleft
end sub
```