

基于 OWC 的动态统计图表的设计与实现

张成才, 常 静, 叶 伟

(郑州大学 水利与环境学院, 河南 郑州 450001)

摘 要: 在使用 ASP.NET 进行 Web 开发时, 经常需要将各种统计数据或数据库以图形或图表的方式表达出来, 以便进行决策和分析。Microsoft Office 中所提供的 OWC 图表组件及其所支持的编程接口集合, 为 Web 环境下统计图表的开发提供了一种十分有效的方法。对 Office 自带的 OWC 组件进行了全面研究和剖析, 结合笔者在项目开发中的实际经验, 总结了一套在 ASP.NET 中调用 OWC 组件, 使用 C# 编程语言实现各种动态统计图、折线趋势图的绘制方法和技巧; 并将其运用到防洪工程规划地理信息系统中, 实现了对水库水位和库容关系以及水库水位和泄量关系的实时分析。

关键词: ASP.NET; OWC; C#; 统计图

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2009)10-0247-03

Design and Achievement of Dynamic Statistical Chart Based on OWC

ZHANG Cheng-cai, CHANG Jing, YE Wei

(School of Water Conservancy and Environment Engineering, Zhengzhou University, Zhengzhou 450001, China)

Abstract: For the requirements of decision and analysis, it often shows a variety of statistic data or database in the ways of figure or chart when developing Web through ASP.NET. Depending on the practical experience during this project development, a set of methods and skills is adopted when transferring OWC in ASP.NET and drawing sorts of statistical chart and polyline trend chart with C#. The methods also apply to the GIS of engineering plan. As a result, 24-hour analysis of the relationship between reservoir storage capacity and reservoir water level and between storage capacity and divulgence capacity were achieved.

Key words: ASP.NET; OWC; C#; statistical chart

0 引 言

图表是人们认识客观事物, 了解相关知识的重要辅助工具。它形象直观地反映了事物的发展变化、演变趋势或者是事物之间的相互关系^[1]。尤其是动态图表可以对系统行为的某些方面做出更为详细、清晰的描述, 以便让观测者根据图表的结果对所关心的事物做出前期的预测。随着网络和图形图像的广泛发展, 人们希望在实时查看数据或者实现数据共享的同时, 能够以不同的方式来呈现和分析数据, 如排序方式、分类方式、图表方式等等^[2]。但是, 通常在进行 Web 开发时首选的 ASP.NET 编程框架本身是不支持图表功能, 只能借助第三方控件进行开发。文中总结了一套在 ASP.NET 中调用微软组件 OWC (Microsoft Office Web Component) 实现各种曲线图、柱状图、饼状图、雷

达图等统计图表的方法。

1 OWC 组件

1.1 OWC 简介

微机组件 OWC 是微软公司针对 Web 应用而在 Office 2000 中开发的一套在线分析处理 (OLAP) 组件, 它是一个组件对象模型 (COM) 控件集合。Office XP 对应 OWC10 版本, Office 2003 对应 OWC11 版本。它是随着 Microsoft Office 一起发行的一套组件, 用于在 Web 上发布电子表格、图表和数据库, 以及在网查看已发布的组件和“数据访问页”^[3]。OWC 主要包含 4 个组件: 电子数据表格、图表、数据透视表和数据源 (Data Source)^[1]。通过在 ASP 页面调用 OWC 组件, 可以轻易地实现各种曲线图、柱状图、饼状图、雷达图等统计图表的绘制, 大大提高开发效率。

1.2 OWC 图表对象模型及关键编程元素

图 1^[4] 为图表对象模型及关键编程元素示意图。由图可见, 图表对象模型中的顶层是 ChartSpace (图表

收稿日期: 2009-01-13; 修回日期: 2009-04-24

基金项目: 水利部重点实验室开放课题 (2007005)

作者简介: 张成才 (1964-), 男, 博士生导师, 研究方向为水文信息化技术。

工作区)对象,它是图表容器的顶层,其中包含了多个对象或对象集合,其中的 ChCharts(图表集)为 ChChart(图表)对象集合,每个 ChChart 对象表示一个图表。图表对象中又包含了 ChAxes(坐标轴)、ChInterior(内部格式)、ChLengend(图例)、ChTitle(标题)、ChPlotArea(绘图区)、ChAeriesCollection(系列)等对象或集合对象。

一个图表工作区可以包含多个图表。当要使用图表时,首先要用 New 关键字新建 ChartSpace 对象^[5],然后通过相关对象的相关方法和属性在图表工作区中添加图表,设置图表类型,为图表绑定数据源,设置标题、标签、图例、轴刻度等等。ChartSpace 对象支持多种图表类型,通过设置图表类型属性值可以创建不同类型的图表,满足不同用户的需求。

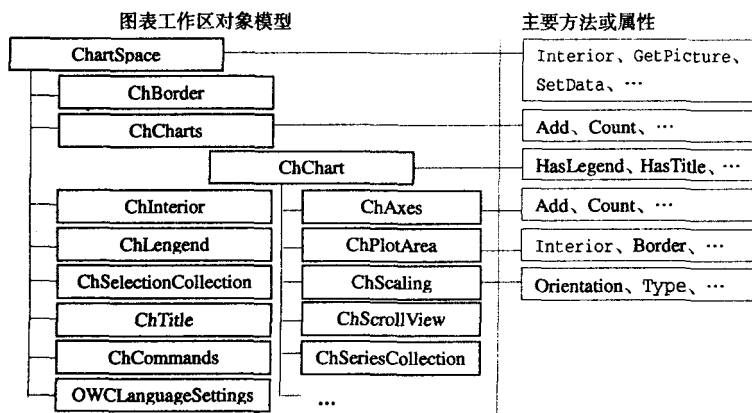


图 1 图表对象模型及关键编程元素示意图

2 OWC 组件在 ASP.NET 中的应用

2.1 基本原理及实现步骤

通过使用 OWC 组件,可以在 Web 服务器上根据数据库产生的记录集或用户提交的信息产生各种复杂图表,同时还能将生成的图表输出为网上常用图像格式的图像文件。输出的图像文件可以以一种二进制数据的格式发送到客户端浏览器,也可以根据需把图片以文件格式存放在服务器临时文件夹下^[6]。因为图表是作为图片格式被下载到客户端的,所以并不需要客户端下载任何 COM 组件,也不需要客户端安装 MS Office 软件,只需要服务器上装有 OWC 组件即可浏览这些图表,而每个客户存取的是一个二进制的数流,输出的图像文件只是一个临时的过渡品,每个用户都可以根据提交的信息获得一个属于自己的图表图像,各个客户互不影响,不存在彼此之间的图表文件交叉覆盖^[7]。但在“背后”,图形文件实际上是由服务器上 ASP.NET 回应客户请求时动态生成的,因此,这种技术对客户端没有特殊的要求,只要能够显示图形就可

以了。

利用 OWC 组件创建统计图的实现步骤如图 2 所示。

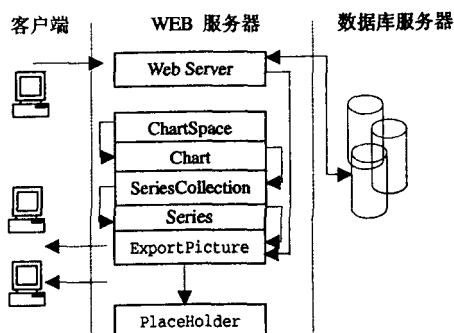


图 2 实现步骤图

2.2 应用实例

应用实例以某水库的库容、水位和泄量数据为基础,绘制出了该水库库容与水位、库容与泄量的关系曲线图。

(1) 在 Visual Studio 2005 中创建一个新的 ASP.NET Web 应用程序解决方案。并添加一个页面“ChartDrawing.aspx”。

(2) 打开“项目”菜单,选择“添加引用”。在“添加引用”的对话框中,选择“Microsoft Office Web Components 11.0”,然后“确定”。

(3) 配置 Web.config 文件^[8],添加如下代码:

```
<connectionStrings>
<add name="fhDataConnectionString" connectionString="Data
Source=.; Initial Catalog=fhData; Persist Security Info=True;
User ID=admin; Password=zzfh" providerName="System.Data.
SqlClient"/>
</connectionStrings>
```

(4) 在“ChartDrawing.aspx”前台添加一个 Placeholder 控件。

(5) 在后台添加如下命名空间:

```
using System.Data.SqlClient; //添加数据操作引用
using Microsoft.Office.Interop; //添加 Office 组件引用
using OWC11 = Microsoft.Office.Interop.Owc11; //为了引用方便
```

(6) 后台代码如下:

```
//连接数据库,获得数据源
string ConnectString = ConfigurationManager.ConnectionStrings
["fhDataConnectionString"].ConnectionString;
SqlConnection con = new SqlConnection(ConnectString);
string sql = "select 库容,水位,泄量 from 郭家咀水库水位库容泄
量关系";
SqlDataAdapter sdb = new SqlDataAdapter(sql, con);
```

```

DataSet ds = new DataSet();
sdb.Fill(ds);
//存放库容
string[] KuRong = new string[70];
//存放水位
string[] ShuiWei = new string[70];
//存放泄量
string[] XieLiang = new string[70];
//为数组赋值
for (int i=0; i<ds.Tables[0].Rows.Count; i++)
{
    ShuiWei[i] = ds.Tables[0].Rows[i][0].ToString();
    KuRong[i] = ds.Tables[0].Rows[i][1].ToString();
    XieLiang[i] = ds.Tables[0].Rows[i][2].ToString();
}
//为 x、y、轴指定特定字符串,以便显示数据
string strXdata = String.Empty;
foreach (string strData in ShuiWei)
{
    strXdata += strData + "\t";
}
.....
//创建 chartSpace 对象来放置图表
OWC11.ChartSpace laySpace = new OWC11.ChartSpaceClass();
//设置绘图空间标题,可以根据实际需要设置标题的样式
laySpace.HasChartSpaceTitle = true;
laySpace.ChartSpaceTitle.Caption = "库容与水位、泄量关系曲线图";
laySpace.AllowFiltering = true;
laySpace.AllowGrouping = true;
.....
//在 chartSpace 对象中添加图表,可以根据实际需要设置图表区域样式
OWC11.ChChart insertchart = laySpace.Charts.Add(0);
.....
//指定绘制图表的类型。类型可以通过
OWC11.chartchartTypeEnum 枚举值得到
insertchart.Type = OWC11.ChartChart-
TypeEnum.chChartTypeLineMarkers;
//带节点的折线图
//根据实际需要设置图例、图表标题的样
式、坐标轴图示说明、次要网格线的样式等
.....
//添加一个 series 系列,根据实际需要设置
图形的样式,例如本文中设置了曲线节点的
样式、线形图的连线的颜色、数据标签等
insertchart.SeriesCollection.Add(0);
insertchart.SeriesCollection[0].DataLa-
belsCollection.Add();
.....

```

```

//给定 series 系列的名字
insertchart.SeriesCollection[0].SetData(OWC11.ChartDimen-
sionsEnum.chDimSeriesNames, + (int)OWC11.ChartSpecial-
DataSourcesEnum.chDataLiteral, "库容与水位关系曲线");
//给定分类
insertchart.SeriesCollection[0].SetData(OWC11.ChartDimen-
sionsEnum.chDimCategories, + (int)OWC11.ChartSpecial-
DataSourcesEnum.chDataLiteral, strXdata);
//给定值
insertchart.SeriesCollection[0].SetData(OWC11.ChartDimen-
sionsEnum.chDimValues, (int)OWC11.ChartSpecial-
DataSourcesEnum.chDataLiteral, strYdata);
//OWC 支持在同一张图表中显示两条以上的曲线,只需继续添
加、设置 series 系列
.....

```

若需要显示双坐标轴,可写入以下代码:

```

insertchart.SeriesCollection[1].Ungroup(true);
OWC11.ChAxis axXieliangAxis = insertchart.Axes.Add(in-
sertchart.SeriesCollection[1].get_Scalings(Microsoft.Office.In-
terop.Owc11.ChartDimensionsEnum.chDimValues));
axXieliangAxis.Position = Microsoft.Office.Interop.Owc11.Char-
tAxisPositionEnum.chAxisPositionRight;
axXieliangAxis.HasMajorGridlines = false;
.....
string strAbsolutePath = (Server.MapPath(".")) + "\\Show-
Data.gif";
laySpace.ExportPicture(strAbsolutePath, "GIF", 800, 600);
//创建 GIF 文件的相对路径
string strRelativePath = ".\\ShowData.gif";
//创建 GIF 文件的相对路径
string strImageTag = "<IMG SRC='" + strRelativePath + "'>";

```

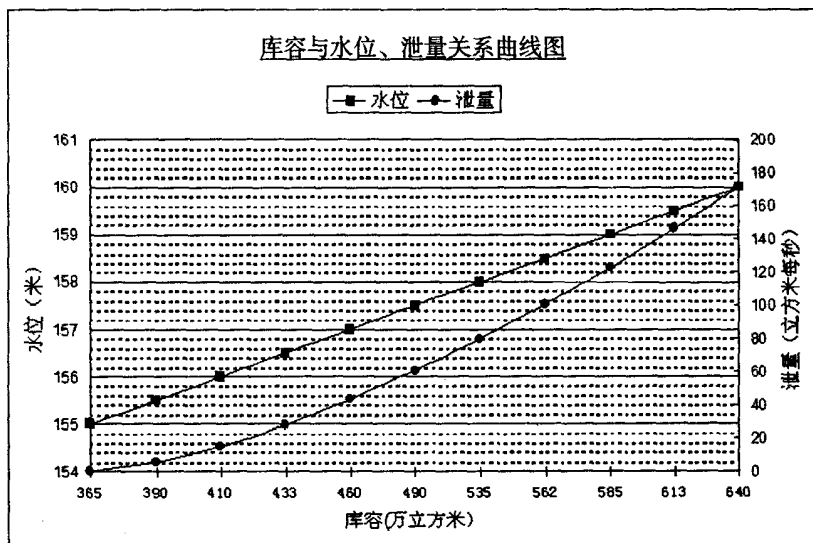


图 3 郭家咀水库库容与水位、泄量关系曲线图

(下转封三)

医疗信息子系统直接产生的数据。

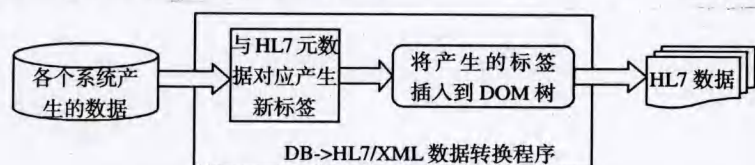


图5 接收数据并存入医疗信息整合数据库

(2)通过建立 DOM 对象,新建一棵 DOM 空树,这棵空树是要产生的 XML DOM 树的初始化形态。

(3)根据 HL7/XML 的元数据定义,将使用者输入的数据和元数据对应,使其完全符合 HL7/XML 标准,将不同含义的数据加上不同的标签,这样就可以生成新的 XML DOM 节点,这一个节点也就是所建立的医疗信息整合后的一个最小数据单元。

(4)将新建立的 XML DOM 节点插入到第 2 步所建立的树中去。

(5)重复(2)、(3)、(4)的过程,直到所有数据被插入到 DOM 树中。至此,已经在内存中建立起来一个医疗信息整合的 DOM 树,最后将这棵树输出,保存进医疗信息整合数据库中。

4 结束语

医疗信息化建设是目前医疗卫生行业发展的一个总趋势,而医疗信息的有机整合又是其中最核心的部分,只有这样才能真正消除原有医院之间及各医院内部各医疗系统间信息的相对封闭性及传输滞后的状

态,实现信息的高度共享,最终服务于医患人员。

借助于智能 Agent 技术,并将其应用到医疗信息系统中是一种全新的尝试,相信也会为今后医疗卫生行业的信息化建设提供一种新的思路。

参考文献:

- [1] 姚伟华,朱旭阳. 医疗信息整合研究概述[J]. 第一军医大学学报, 2003,23(12):1334-1336.
- [2] 陆波,李伟鹏. 基于 HL7 Engine 的医疗信息网络整合方案[J]. 医疗卫生装备, 2005,26(2):34-37.
- [3] America National Standards Institute. Health Level Seven Version 3[S]. Michigan:Health Level Seven Inc,2003.
- [4] 刘毅,张月琳. 基于 Agent 的邮件过滤与个性化分类系统设计[J]. 计算机技术与发展,2009,19(2):66-67.
- [5] 何炎祥,陈萃萌. Agent 和多 Agent 系统的设计与应用[M]. 武汉:武汉大学出版社,2001.
- [6] 张春飞,李万龙. 基于多 Agent 的智能教学系统模型[J]. 河北科技大学学报,2008(1):48-52.
- [7] Hoffmeyer J A, Vogler L E, Mastrangelo J F, et al. A new HF channel model and its implementation in a real-time simulator[C] // Proc. IEEE Fifth International Conference on HF Radio Systems and Techniques. Edinburgh, UK: [s. n.], 1991:173-177.
- [8] 张文君,胡淑涛,张磊. 基于 XML 的医疗机构信息交换系统[J]. 上海大学学报:自然科学版, 2006,12(4):419-422.

(上接第 249 页)

```
Placeholder1.Controls.Clear();
```

```
Placeholder1.Controls.Add(new LiteralControl(strImageTag));
```

(7) 输出结果。

结果如图 3 所示。可以看到库容与水位、泄量的关系,以便于决策层分析和制定具体的应对方案。

根据用户的实际需求,通过更改 OWC 组件相关类里面的方法和属性,就可以输出不同类型的统计图表,例如柱状图、饼状图等等。

3 结束语

通过调用组件 OWC 可以根据用户的实际需要输出不同类型的图形,灵活多变,执行效率高,大大提高了开发效率,避免了直接输出动态图表到客户端时有可能带来的一系列问题,克服了由于 .NET Framework SDK 中没有提供制作数据图表的相关类而使得在 Web 环境下绘制各种动态统计图、趋势图复杂、麻烦的缺点,增强了系统的统计功能,有利于快速集成和开发。

参考文献:

- [1] 张成才,田文文,崔雅博. TeeChart 控件实现河南省航运水资源量动态统计图[J]. 计算机技术与发展,2009,19(2):224-226.
- [2] Microsoft. OWCVB11.chm[M]. 美国:微软公司,2002.
- [3] Stearns D. Microsoft Office 2000 Web Components 编程技术内幕[M]. 希望图书创作室译. 北京:北京希望电子出版社,2000.
- [4] 金栋林. OWC 图表组件及其在 WEB 环境中的开发应用[J]. 沈阳航空工业学院学报,2006,23(4):58-59.
- [5] 刘焯,季石磊. C# 编程及应用程序开发教程[M]. 北京:清华大学出版社,2007.
- [6] 朱冠宇,熊伟. 基于 ASP.NET 的图像动态显示技术[J]. 微机发展(现更名:计算机技术与发展),2004,14(7):49-51.
- [7] 倪峥. 利用 GDI+ 和 ASP.NET 实现 Web 图形的动态生成[J]. 信息技术,2005(11):81-83.
- [8] Smacchina P. Practical .NET2 and C#2[M]. 北京:人民邮电出版社,2008.