

文章编号: 1005-8451 (2007) 02-0043-04

ASP.NET 中的 Web 打印方法研究

田 颖, 何 鹏

(西南交通大学 信息科学与技术学院, 成都 610031)

摘要: Web 打印是在开发 B/S 结构应用程序时经常遇到的一项比较复杂的技术。主要针对 ASP.NET 下的 Web 打印问题提出 4 种解决方法, Web 打印中还有很多细节问题需要进一步的探讨。

关键词: Web 打印; ASP.NET; 样式表; XML; Crystal Report

中图分类号: TP311.52 **文献标识码:** A

Research on methods of Web print in ASP.NET

TIAN Ying, HE Peng

(School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: Web print was a complicate technology when developing application program under B/S structure. 4 methods were put forward to resolve problem of Web Print under ASP.NET. The advantage and disadvantage of every method was also described. Many other detail problems of Web Print needed to be researched on.

Key words: Web printing; ASP.NET; CSS; XML; Crystal Report

Web 打印是很多 B/S 结构软件项目必不可少的功能, 也是开发人员所面临一个共同的难题。相对于 Windows 桌面应用程序来讲, Web 应用程序的打

收稿日期: 2006-07-22

作者简介: 田 颖, 在读硕士研究生; 何 鹏, 助教。

印有种种限制, 技术人员在项目开发过程中经常会遇到用户这样或那样的需求。在开发 ASP.NET 下的 Web 应用程序时, 由于只能采用浏览器作为用户界面进行交互, 所以不能精确地控制客户端的打印机。因此, 针对如何在客户端实现 ASP.NET 下的 Web 打

End If

```
Id=ws.PickListStrings( PICKLIST_CUSTOM,  
False,db.server,curpath & "org.nsf","cldname","提示  
","请选择负责人",2)
```

doc.fzr=Id(0)

步骤 2: 负责人审阅文档, 填写意见, 选择文
档签字。

doc.fzrqz=s.CommonUserName

步骤 3: 返送签人, 负责人审阅完毕后, 将文
档返回给经办人。

doc.fzr=""

步骤 4: 经办人根据实际需要, 选择是否传给
签发人, 其操作和实现原理与传负责人类似, 只是
传给签发人前必须判断负责人签字否。

(2) 流程跟踪: 在表单上设置流程跟踪, 详细
记录文档每一步流转的操作人员, 操作时间, 操作
名称。每一条信息都是系统计算出来的, 用户无法
修改。这样既实现了对文档的监控, 在一定程度上
也保证了工作人员及时、认真地完成工作。

3 结束语

本文以基于 Lotus Notes/Domino 平台的成都铁
路局人事管理信息系统中领导干部任免为例, 运用
Lotus Notes/Domino 的安全技术实现了任免过程中
的安全控制; 实现了关联数据库间的互操作; 通过
模板和 OLE 技术提高了用户的工作效率, 解决了
Lotus 生成报表薄弱的困难; 通过文件流转实现了
不同部门、不同级别人员的协同办公, 在满足了用
户需求的同时用户的工作效率也大大地提高了。

参考文献:

- [1] Randal I A.Tamura. Lotus Notes 和 Domino Server 4.6 技术大全 [M]. 王建华, 高杏生, 刘孜群, 等. 北京: 机械工业出版社, 1998.
- [2] 罗 琳, 陈怡之, 陈 斌, 等. Notes 编程疑难详解 [M]. 北京: 人民邮电出版社, 2000: 281-296.
- [3] 刘福江, 郭 艳. 基于 OLE 自动化技术实现 Excel 报表到
Notes 文档的转换 [J]. 编程与应用起步, 2002 (8): 17-19.

印这一问题，本文提出4种解决方法：

- (1) 直接利用IE和样式表(CSS)实现Web打印；
- (2) 导出到excel或者word中实现Web打印；
- (3) 利用.NET组件实现Web打印；
- (4) 利用Crystal Report for Visual Studio .NET实现Web打印。

1 直接利用IE和CSS实现Web打印

1.1 调用window.print方法实现打印

在页面上添加“打印”按钮，然后增加如下代码：

```
<input type="button" onclick="window.print()" value="print">
```

这段代码很简单，是通过Javascript脚本语言调用window.print()方法来实现打印。但是，如果采用这种方式来打印页面，页面上不需要的元素也会被打印处理，页头和页尾的格式也不好控制，并且无法实现“打印设置”和“打印预览”功能。可以使用下面的方法来实现这些功能。

1.2 调用IE内置的WebBrowser控件实现打印

WebBrowser控件是IE内置的浏览器控件，它是ActiveX控件，无需用户下载，可以通过调用WebBrowser控件的ExecWB方法实现打印，代码如下：

```
// 加入ID为WebBrowser1的WebBrowser控件
<object ID='WebBrowser1' WIDTH=0 HEIGHT=0
CLASSID='CLSID:8856F961-340A-11D0-A96B-00C04FD705A2' VIEWASTEXT></object>
```

定义完object以后，下面使用这个控件完成打印设置、预览及打印功能，代码如下：

```
<script language="javascript">
function doPageSet()
{
    // 打印设置
    WebBrowser1.ExecWB(8,1)
}
function doPreview()
{
    // 打印预览
    WebBrowser1.ExecWB(7,1)
```

```
}
function doPrint()
{
    // 打印
    WebBrowser1.ExecWB(6,1)
}
function doDirectPrint()
{
    // 直接打印
    WebBrowser1.ExecWB(6,6)
}</script>
```

调用上面的Javascript函数可以实现打印、打印设置和打印预览功能。客户端独立完成打印目标文档的生成，减轻了服务器端的负荷，但是打印的时候整个页面都会被打印，因此就会打印出一些不需要的对象，可以通过在样式表CSS中设置打印或显示样式来隐藏这些对象，代码如下：

```
<style> @media Print { .Noprn { DISPLAY: none } }</style>
```

该样式表将media的属性设置为print来控制打印输出样式，如果在页面上的“打印”按钮的属性中加上class="Noprn"，这样按钮就不会被打印出来。在调用window.print()方法实现打印时，也可以通过该样式表隐藏不需要的元素。

2 导出到excel或者word中实现Web打印

这种方法将Web打印转化成Windows单机版的打印，可适应性强，易于控制，以导出到excel为例，实现代码如下：

```
Response.ContentType = "application/vnd.ms-excel";
Response.AddHeader("Content-Disposition", "attachment;filename=test.xls");
// 如果输出为Word，修改为以下代码
// Response.ContentType = "application/ms-word"
// Response.AddHeader("Content-Disposition", "attachment;filename=test.doc")
StringBuildersb=newStringBuilder();
System.IO.StringWriter sw = new System.IO.StringWriter(sb);
System.Web.UI.HtmlTextWriter hw = new System.
```

```

Web.UI.HtmlTextWriter(sw);
    sb.Append("<html><body>");
    dgShow.RenderControl(hw);
    sb.Append("</body></html>");
    Response.Write(sb.ToString());
    Response.End();

```

导出成 Word 或 Excel 后用户可以自定义打印的内容和格式，开发过程简单易行，但难以生成打印预览，因此，易生成简单格式的 Word 或 Excel 表格。此方法有一个限制就是在服务器端或客户端使用时，要保证安装有 office 软件，在客户端使用时要求客户端在 IE 的安全设置上有一定的要求。

3 利用.NET 组件实现 Web 打印

如果要求精确控制打印效果，并且出于安全性考虑，不能直接连数据库时，可以采用.NET 编写的受控组件实现 Web 打印，优点：在.NET Common Language Runtime 的控制之下运行，不需要进行客户端注册，比 ActiveX 安全性高，代码编写方便，直接支持 XML 技术，具有很强大的打印控制功能并且与 IE 兼容性高。其原理是通过 XML 强大的自定义功能，自定义出所有需要的格式控制标签，在服务器端进行动态编码后通过 Web 服务器传到客户端，然后在客户端进行格式解析，根据服务器端定义的打印格式从客户端直接控制打印机实现 Web 打印。

由于不允许直接连接到数据库，因此，只能采用 XML 文件进行中间数据交换格式，.NET 内部已经集成了 XML 解析器，直接就可以通过使用.NET 类库调用。既可以做成桌面应用程序形式，通过远程调用；也可以嵌入到 IE 浏览器中，直接在网页中运行。

使用 XSL（扩展样式表语言）将 XML 转换成其他的文本格式，XSL 转换包括发现或者选择一个模式匹配，通过使用 XPath 选择一个结果集，然后对结果集中的每一项，为这些匹配定义结果输出。XSL 是一个功能强大的工具，可以把 XML 转换成任何想要的格式。代码如下：

```

// 创建新的XsITransform对象
XsITransform xslt = new XsITransform();
// 从XSL文件中导入样式表
xslt.Load(Server.MapPath("WorkersToHTML.

```

```

xsl"));
// 创建新的XPathDocument对象，并导入
// XML 文件
XPathDocument XDoc = new XPathDocument
(Server.MapPath("Workers.XML"));
// 创建新的XmlTextWriter对象用于导出
// HTML 格式文件
XmlWriter writer = new XmlTextWriter(Server.
MapPath("Workers.html"),
System.Text.Encoding.UTF8);
// 进行实际的XSLT转换操作
xslt.Transform(XDoc, null, writer);
// 操作完成后关闭XmlTextWriter对象
writer.Close();

```

采用服务器脚本动态生成 XML 文档时，发送内容类型应该设置为 text/xml（普通 html 页面为 text/html），字符编码应该为 UTF-8，否则会出现编码错误问题。这种打印方式适合于格式变化大，数据量小的应用，但是客户端需要安装.NET Framework 组件，如果文件较大 XML 的解析速度不是很理想，页面首次加载时会有明显的延时。

4 利用Crystal Report for Visual Studio .NET 实现 Web 打印

.NET 平台中集成有 Crystal Report，因此，在使用 Visual Studio .NET 开发应用程序时，不需要安装额外的软件。

Crystal Report for Visual Studio .NET 特点：

(1) 针对所有以.NET 为目标的程序语言提供高度的集成，可以在 Windows 应用程序或 ASP.NET Web 应用程序的项目中使用本身所专长的程序语言来设计报表。

(2) 提供强大的报表设计工具 Crystal Report Designer，可以通过其定义报表的数据源，定位各个控件，轻易制作出各种复杂格式的报表。

(3) 通过报表查看器控件 (Crystal Report Viewer) 设置报表的格式和自定义导出选项等，将报表展现在 Web Form 网页上。

(4) 用户仅需要 Web 浏览器就可以查看报表。

.NET 中的 Crystal Report 具有两层结构：客户端和服务器端。客户端仅需使用浏览器即可得到装载于.aspx 页面中的报表。服务器端包括：

- a.Crystal Report Engine (CREngine.dll),它将Crystal Report转为传到.aspx页中的简洁明了的HTML；
- b.Crystal Report Designer (CRDesigner.dll),使用它设计标题、插入数据、公式、表格和次级报告等；
- c.Data Source,.rpt文件获取数据的方式可以选择制作Crystal Report以获取数据而无须手工写任何代码，也可以选择绑定dataset并将其传入报表文件；
- d.Crystal Report Viewer web form Control (CRWebFormViewer.dll),可以作为一个包含报表的容器。

Crystal Report for Visual Studio .NET的执行模式有两种: Pull Model (拉模型)和Push Model (推模型),执行任一种模式均可获取Crystal Report数据。在拉模型中,驱动程序将连接到数据库并根据需要将数据“拉”进来。使用这种模型时,与数据库的连接和为了获取数据而执行的 SQL 命令都同时由 Crystal Reports 本身处理,不需要开发人员编写代码。如果在运行时无须编写任何特殊代码,则使用拉模型。相反,推模型需要开发人员编写代码以连接到数据库,执行 SQL 命令以创建与报表中的字段匹配的记录集或数据集,并且将该对象传递给报表。该方法使您可以将连接共享置入应用程序中,并在 Crystal Reports 收到数据之前先将数据筛选出来。

用户可以将报表导出各种格式的文件,包括.pdf,.doc,.xls,.html,.rpt等。

下面是以推模式为例实现 Web 打印的步骤如下:

- a.在设计页面放置Crystal Report Viewer控件;
- b.建立数据库连接并创建新的数据集DataSet;
- c.执行 SQL 命令得到数据集,将对象传递给报表,代码如下:

```
string strSel = "Select * from SaleOfCountry";
SqlDataAdapter MyAdapter = new SqlDataAdapter(strSel, MyConn);
MyAdapter.Fill(ds, "SaleOfCountry");
ReportDoc.SetDataSource(ds);
Crv.ReportSource = ReportDoc;
d.将报表导出为.pdf 格式文件,代码如下:
CrystalDecisions.Shared.DiskFileDestinationOptions
DiskOpts = new;
CrystalDecisions.Shared.DiskFileDestinationOptions
```

```
());  
ReportDoc.ExportOptions.ExportDestinationType =  
CrystalDecisions.Shared.ExportDestinationType.  
DiskFile;  
string strFileName = Server.MapPath("Output");  
ReportDoc.ExportOptions.ExportFormatType =  
CrystalDecisions.Shared.ExportFormatType.  
PortableDocFormat;  
DiskOpts.DiskFileName = strFileName + ".pdf";
```

尽管Crystal Report Viewer控件内置了zooming、page navigation 等选项,但是它并没有内置的打印选项。需要使用浏览器的打印选项才可以。

Crystal Report for Visual Studio .NET未注册版本只有前 30 项功能可用。在此之后的 save option 将不能使用。为了使用全部的功能,将不得不为之付费。

Crystal Report for Visual Studio .NET适合于大型的报表系统,使用这种发式进行打印非常灵活,但是客户端需要安装组件,部署不是太方便。

5 结束语

Web 打印是很多 B/S 结构应用系统不可或缺的功能,而且是一项比较复杂的技术,本文主要针对 ASP .NET 下的 Web 打印问题提出了 4 种解决方法,并指出了每种方法的优缺点,在开发过程中可以根据用户的需求作选择。Web 打印中还有很多细节问题需要进一步的探讨。

参考文献:

- [1] 陈宗兴,杜国梁. ASP.NET 网络系统程序开发[M]. 北京:中国水利水电出版社, 2004.
- [2] Richard Leinecker. ASP.NET实用全书[M]. 齐 岷,张 力. 北京: 电子工业出版社, 2003, 1.
- [3] 卢 彦. 利用XML 实现通用WEB 报表打印[ER/OL]. <http://www.microsoft.com/china/community/program/originalarticles/techdoc/xmlwebreport.mspx>, 2006-03-02
- [4] Mickey Williams. Visual C#.NET技术内幕[M]. 北京: 清华大学出版社, 2004.
- [5] 章立民. 用实例学Crystal Report for Visual Studio.NET[M]. 北京: 电子工业出版社, 2004.
- [6] 李小延, 王焕彬. 在VB中实现报表打印的两种方法[J]. 铁路计算机应用, 2005 (9).