

# 无插件网页快照报表连续显示技术<sup>①</sup>

沈疆海, 杨 起

(长江大学 计算机科学学院, 荆州 434023)

**摘 要:** 报表作为一种信息组织和分析的有利手段, 广泛应用在各行各业, 是企业信息系统的重要组成部分之一。报表应用系统的开发是基于 Web 结构的信息化系统开发的重点之一, 尤其是在客户端, 用户在浏览报表时, 目前基本上都是通过上下翻页来查看, 这样使得报表浏览方式呆板, 不够直观。针对这个问题提出了无插件 Web 报表连续显示的技术并予以实现。该技术同时结合 XML 和 XSLT 技术的优势。详细说明了该技术的总体设计, 最后介绍了该套方案在实际项目中的应用情况, 证实了其可用性和易用性。

**关键词:** 报表; Web; 无插件; 连续显示; XML; XSLT

## Webpage Snapshot Report Continuous Display Technology without Plug-Ins

SHEN Jiang-Hai, YANG Qi

(College of Computer Science, Yangtze University of China, Jingzhou 434023, China)

**Abstract:** As a favorable method of information organization and analysis, report form is widely used in industries. It is an important part of enterprise information system. One of the important of Web-based information system is the report forms application system. Especially in the client side, users browse report forms mainly through flipping and down pages. This way makes report forms browsing dull and not intuitive. To solve this problem, this paper puts forward and realizes the technology that web snapshot displays report forms continuously without plug-ins. The technology also combines the advantage of XML and XSLT technology. In this paper, the overall design of this technology is explained in detail. The successful application of this technology in practical projects proves that it is both usable and accessibility.

**Keywords:** report forms; Web; without plug-ins; continuous display; XML; XSLT

报表作为信息提取和显示的重要方式, 已经得到越来越多的使用。传统的报表主要采用 C/S 结构, 开发主要是通过程序员编写大量复杂代码实现, 开发难度大、周期长、可维护性差。随着网络的快速发展, 报表应用向 Web 转移, 基于 B/S 方式的 Web 报表已经成为主流, 可以做到让客户可以随时随地使用 IE 浏览器去访问客户所需的各种数据和资源, 然而目前的报表系统发展还不够成熟, 功能特性上还不够完善。对于报表应用系统, 大体可以分成三个部分: 报表模板的生成部分、与数据库的交互部分、报表在客户端的

显示部分<sup>[1]</sup>。基于 Web 结构的信息化系统开发的重点之一就是报表部分, 用户在浏览报表时, 目前基本上都是通过上下翻页来查看, 这样使得报表浏览方式呆板, 不够直观。本文将给出一种无插件网页快照报表连续显示的技术解决方案, 实现报表的快速定位和滚动浏览, 已成功应用于地质分析化验应用系统的项目中。该方案采用 XML 和 XSLT 技术相结合, 通过 XML 技术实现报表的数据来源描述和配置, XSLT 负责 XML 文档转换, 通过 Jscript 技术实现事件的响应, 最终使客户端成为完全意义上的瘦客户端, 这种方式是

① 基金项目:湖北省自然科学基金(2009CDA140)

收稿时间:2010-06-29;收到修改稿时间:2010-07-31

最理想的，同时又是安全的。

## 1 技术分析

### 1.1 可扩展标记语言(XML)

XML(Extensible Markup Language, 可扩展置标语言)是由 W3C(World Wide Web Consortium, 互联网联合组织)于 1998 年 2 月发布的一种标准<sup>[2]</sup>, 是 SGML (standard Generalized Markup Language, 标准通用置标语言)的一个简化子集。XML 实际上是一种定义语言, 即所谓的元语言。使用者可以定义大量的标记来描述文件中的任何数据元素, 从而突破了 HTML 固定标记集合的约束, 使文件的内容更丰富、更复杂并组成一个完整的信息体系。XML 以纯文本的形式描述数据, 易于读写, 适合在不同的应用程序间交换数据, 实现数据共享, 并且速度快, 效率高。XML 具有开放标准、平台无关的特性, 适合于 Web 应用模型的发展, 已经成为新一代 Internet 通用语言。它具有以下几个特点: 简单性、可扩展性、互操作性、开放性。

XML 最大的特点在于它的可扩展性<sup>[3]</sup>。其次的优点是内容与形式的分离, XML 的标记为要表现的数据赋予了一定的含义, 用这种形式存储时, 数据非常简单明晰, 同时数据搜索可以简单高效的进行。XML 使用一个简单而又灵活的标准格式, 为基于 Web 的应用提供了一个描述数据和交换数据的有效手段。

### 1.2 XSLT 及工作原理

可扩展样式表语言转换<sup>[4]</sup>(Extensible Stylesheet Language Transformation, XSLT)是 W3C 正在制定的规范 XSL 中的一部分, 负责 XML 文档的转换。XSLT 用于将一个 XML 文档转换成另一个 XML 文档或另一类型的文档, 在转换过程中可对数据进行过滤、排序、计算等操作。XSLT 文档本身是一个格式良好的 XML 文档, 在表达文档转换时利用 Xpath 指定源文档中待转换的部分, 链接到需要转换的 XML 文档, 通过 XSLT 处理器实现对 XML 文档的转换并将转换结果显示出来。使用 XSLT 的优势在于<sup>[5,7]</sup>:

(1) 功能强大。XSLT 提供了条件判断、排序、变量定义、表达式求值、函数库支持等手段来实现各种变换的要求。

(2) 它可以很方便地把 XML 文件转换成 HTML 文件, 从而可以借助 HTML 在普及性和应用性等方面

原有的优势迅速方便地利用 XML+XSLT 来构建 Web 网站;

(3) XML 是独立于表现形式的保存“真正”数据的文件, 而 XSLT 可以告诉浏览器如何展示这些数据。

(4) XSLT 符合 XML 的语法, 可以利用 XML 各种良好的可扩展特性和工具。

(5) 它是 W3C 制定的标准, 通用性有保证, 使用者不必考虑工作运行的平台或是编程语言的不兼容性等。

## 2 方案设计与实现

### 2.1 报表系统总体结构

本系统设计目标是用户能够在友好的系统平台上完成报表的相关工作, 并能够快捷方便的浏览、打印报表。系统通过 HTML 语言定义报表模板、布局及数据显示格式等, 方便维护和修改。同时由于 XML 其独特的优点, 系统采用 XML 作为报表数据的传输载体, 通过 XML 对报表的数据来源进行描述和配置。对于报表应用系统, 大体可以分三个部分: 报表模板的生成模块、与数据库的交互模块、报表在客户端的展示模块。

其中, 报表模板生成模块负责定义报表的封面 (PageCover)、报表头 (PageHeader)、报表内容 (PageContent)、报表尾 (PageFooter) 的表现形式以及对项目模板的 XML 数据来源配置文件; 与数据库的交互模块负责根据一定的查询条件从数据库中提取数据, 将其组织成动态的 Web 文档; 报表客户端展示模块负责将动态生成的 Web 文档在客户端以比较理想的方式展现出来。其总体结构如图 1 所示:

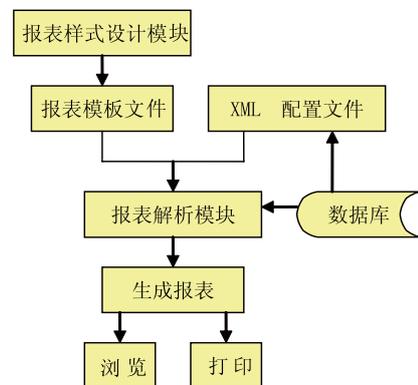


图 1 报表生成系统设计图

对图 1 中设计图分析如下:

(1) 用户通过浏览器发出报表请求, 交由业务逻辑进行参数组织等筹备工作;

(2) 服务器端, 报表解析模块读取 HTML 报表格式文件, 分析报表的各个组成部分, 加载 XML 数据来源配置文件, 自动解析报表的数据来源并填充到报告里面;

(3) 将加载数据后生成的 Web 文档显示在 Web 浏览器上。

### 2.2 报表展示布局详细设计与实现

.NET 框架是微软提供的日臻成熟的组件技术<sup>[6]</sup>。与 COM 类似, 它允许使用多种语言, 但是采用了一种更加统一和一致的方式, 这个框架包含大量的类, 不仅提供了创建典型的图形用户界面应用程序所需要的功能, 而且还提供了创建可伸缩性的、基于 Web 的应用程序和服务所需要的功能。鉴于以上论述, 选择在 .NET 环境下进行架构。

本文讨论的报表展示界面的设计共分为三个部分: 工具栏区、报表缩略图显示区以及报表显示区, 界面布局如图 2 所示:

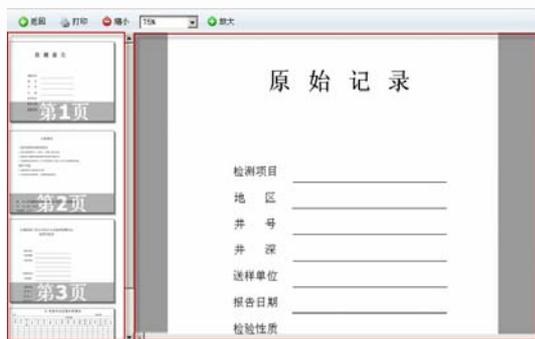


图 2 总体布局设计

其设计思想如下:

根据美国微软公司 Word 文档滚动浏览, 以及 PowerPoint 文档快速定位的优点, 实现检测项目原始记录和分析报告的滚动浏览和快速定位。

部分代码如下:

```
<tr>
//报表缩略图显示区设计
<td><div id="Thumb" >
<asp:PlaceHolder ID="PHThumb" runat="server">
```

```
</asp:PlaceHolder></div></td>
//报表显示区设计
<td>
<div id="Films">
<center>
<table id="ReportContent">
<asp:PlaceHolder ID="PHReportContent"
runat="server"> </asp:PlaceHolder></table>
</center></div></td>
</tr>
```

在报表缩略图显示区, 报表的每一页将以缩略图快照的形式显示, 当用户单击某一页时, 报表显示区将会呈现该页报表的大图。将报表已缩略图的形式显示, 主要是通过 WebPreview.cs 文档中定义的方法完成缩略图快照的获取, 并结合 JavaScript 技术, 最终实现报表在 IE 浏览器上展示, 其流程如图 3 所示:

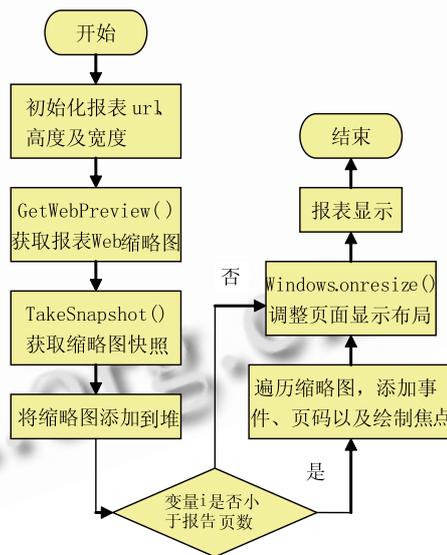


图 3 IE 浏览器上报表展示流程图

设计思路分析如下:

(1) 在报表缩略图显示区和报表显示区这两个部分, 分别添加了两个服务器端控件 Placeholder, 其 ID 分别为 PHThumb、PHReportContent, 然后通过循环将报表加载到相应的控件中(代码表示部分①②), 部分代码如下:

```
//定义变量指示报表的显示方向
string strPageOrientation = "";
```

```
//通过 for 语句循环将报表加载到对应控件中
for (int i = 1; i <= iPageCount; i++)
{
    .....
    //将报表缩略图快照加载到 ID 为 PHThumb 控件中
    this.PHThumb.Controls.Add(new
    LiteralControl(strThumbHtml)); //..... ①
    .....
    //将对应报表加载到 ID 为 PHReportContent 控件中
    this.PHReportContent.Controls.Add(new
    LiteralControl(strIFrameHtml)); //..... ②
}

```

(2) 在 " WebPreview.cs " 文档中定义的 GetWebPreview() 方法获取 Web 预览图, 并通过 TakeSnapshot() 方法来获取其快照;

(3) 完成页面的加载。

(4) 遍历缩略图, 添加事件。为获取的缩略图添加页码并为当前的缩略图绘制焦点;

(5) 调整页面显示布局, 最终显示页面如图 4 所示:

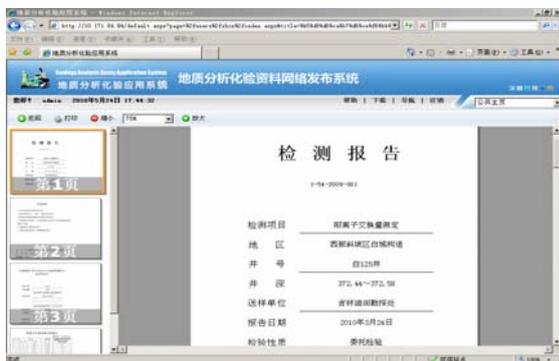


图 4 报表在 IE 浏览器上展示图

(6) 在图 4 中, 单击放大或缩小按钮, 系统就会调用 ReportViewer.js 中 PageZoomIn() 和 PageZoomOut() 两个方法实现报表的放大、缩小功能, 方便用户浏览; 同时用户单击打印按钮, 系统调用打印模板, 实现报表的单页或多页打印。

### 3 方案的实施情况

目前, 此套无插件网页快照报表连续显示技术已经在油田信息化建设项目中得到实施, 并取得了良好

的效果, 较好的实现报表在 IE 浏览器上的展示, 使得用户在浏览报表时, 无需在 IE 客户端安装任何插件或控件, 另外在设计报表时, 通过 HTML 语言绘制报表的版面、布局及数据的显示格式; 报表的数据来源描述和配置采用 XML 技术, 真正实现报表格式与数据源的分离, 并且通过 HTML 语言绘制报表模板, 很好地解决了报表样式复杂, 形式多变的问题, 满足了用户对报表定制的需求。当需求发生变化时, 只需在页面上修改样式信息, 不再需要重新编写程序, 进行重复性的劳动, 减轻了程序员的负担, 提高了工作效率, 也解决了修改和维护的困难。

### 4 总结

本文分析了报表的特点以及相关报表应用系统后, 提出了一种无插件网页快照报表连续显示技术。文中介绍了该项技术的总体框架, 并详细探讨了其设计与实现思路; 同时, 该课题的研究, 吸收了美国微软公司 Word 文档滚动浏览, 以及 PowerPoint 文档快速定位的优点, 实现检测项目原始记录和分析报告的滚动浏览和快速定位。实践证明, 该项技术应用, 可大大提高系统开发效率, 同时解决了不必要的大量重复工作和查询浏览上不便等现实问题, 提高定期统计报表生成的效率, 实现报表数据的有权限共享, 为辅助领导进行决策提供科技支持, 从而促进油田的信息化建设。

### 参考文献

- 1 方伟, 左春, 孙玉芳. 基于 XML 的数据模型在报表模块上的应用. 计算机工程及应用, 2003, 30: 197-200.
- 2 World Wide Web Consortium. Extensible Markup Language (XML) [1999-11]. <http://www.w3.org/TR/xslt>
- 3 耿祥义, 张跃平. XML 实用教程. 北京: 人民邮电出版社, 2009.
- 4 Holzner S. 闻道工作室译. XSLT 技术内幕. 北京: 机械工业出版社, 2002.
- 5 李铮, 尤枫, 赵恒永. 基于 XML 和 XSLT 的 Web 报表解决方案的研究与实现. 计算机工程与设计, 2006, 27(5): 867-871.
- 6 李兰友, 杨晓光. Visual C#.NET 程序设计. 北京: 清华大学出版社, 2004. 154-200.
- 7 李力鸿, 邵敏, 郑震坤. XML 文档信息的几种转换方法分析与应用实例. 计算机科学, 2003, 30(2): 40-44.