

基于WEB+EXCEL模式实现报表采集系统的敏捷方法^①

叶扬

(福建省教育管理信息中心, 福州 350003)

摘要: 开发一个报表采集应用系统往往周期长、难度大。提出基于 Web+Excel 模式快速敏捷地实现报表采集系统的开发与布署。文中介绍了该模式的实现思路和设计方法, 并与其他设计模式进行了比较。

关键词: WEB; EXCEL; 报表采集; 敏捷方法

Based on WEB+EXCEL Model to Achieve Statements Collection System Agile Methods

YE Yang

(Educational Management Information Center of Fujian Province, Fujian 350003 China)

Abstract: Develop a statement collection application system often cycle is long, is difficult. This paper based on Web + Excel model realize rapid agility of statements collection system development and deployed. This paper introduces the realization of the model and the way of thinking, and the design method with other design patterns are compared.

Key words: WEB; EXCEL; Statements collecting; agile methods

1 引言

报表采集是很多政府部门经常面临的需求, 如各类数据报表、调查表等。报表通过采集体系从基层单位收集到汇总单位, 采集体系可大致分为直报类型和分级汇总类型。直报类型是指基层单位直接向汇总单位上报报表, 报表直接由汇总单位审核接收; 分级汇总类型则是指基层单位的上报报表通过一个树型网络, 经过各级审核、汇总、上报到上一级单位, 并最终汇总到采集单位, 如以全国教育事业统计报表为例, 国家、省、地、县各级教育行政部门及各类学校即组成一个树型网络, 作为基层的学校级报表经过县级、地级、省级逐级审核、汇总、上报, 最终在国家级汇总成为全国的综表。

直报类型的报表采集体系结构扁平、耗时短、效率高且能够得到基层单位的第一手信息, 往往是很多采集单位优先选择的类型, 特别是在面对紧急、突发性的采集任务的时候。然而, 在较短时间内从众多基层单位采集电子报表是一件困难的事情。如今, 随着互联网应用的深入人心, 通过网络上报报表已是快速

采集报表的必然选择, 然而当报表采集数据指标量较大、报表页数较多时, 通过 Web 页面直接填报方式存在开发周期长、填报时需实时在线、所有基层单位具备上网条件、网络带宽要求高、服务器负载大等缺点。而本文讨论的 Web 站点与定制的 Excel 报表文件相结合的报表采集系统设计模式, 即 Web+Excel 模式, 能较好的实现直报类型的报表采集, 并且能够快速敏捷的开发与布署, 特别适合突发的、时限要求紧的、大面积采集报表的任务需求。

2 Web+Excel模式采集报表的实现思路

报表的数据采集应用系统必须能够实现录入、打印、校验及数据安全等基本功能。Web+Excel 模式通过以下三点实现这些基本功能。

① 设计 Excel 表样作为填写报表的前端即客户端, 实现报表的录入和打印功能。

② 设计 Web 页面作为接收报表的后端即服务器端, 提供 Excel 表样下载、上传并接收 Excel 文档数据, 提供数据校验功能及相关业务逻辑的处理功能。

^① 收稿时间: 2011-04-20;收到修改稿时间:2011-05-16

③ 登录 Web 页面时提供必要的身份认证, 保证上传数据的安全。

Web+Excel 模式的应用系统的活动图如图 1 所示。

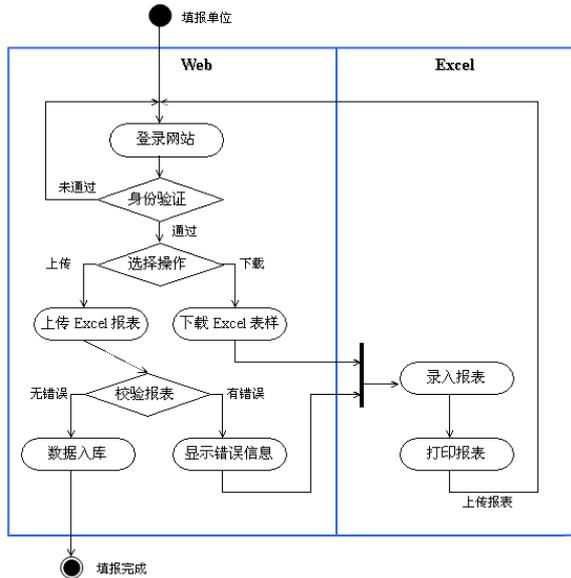


图 1 Web+Excel 模式的 UML 活动图

3 Excel表样的设计

在 Web+Excel 模式中, Excel 表样作为报表的客户端, 实现用户的录入和打印功能。由于 Excel 软件已经得到广泛的应用, 用户界面友好, 定制简便, 无需开发专门的打印功能, Web+Excel 模式实现的敏捷性, 即是得益于此。为了使录入的界面友好, 实现报表录入和打印的功能, 定制 Excel 表样十分重要, 以下是定制 Excel 表样的常用的几个基本技巧。

① 限定报表表样的修改。复选表样所有的填写栏所在的单元格, 选择菜单-格式>单元格>保护>锁定去勾, 重复上述步骤直到报表所有的填写栏均设成可修改, 然后点击菜单-工具>保护>保护工作表>设置密码。设置的效果就是试图在不可编辑区域(如表头等)修改内容和格式都会有警示窗口提示不可修改被保护的只读单元格。

② 限定输入数据的有效值。如要设置输入数据的数值范围为整数 0 至 100, 可选定该数据单元格, 选择菜单-数据有效性>设置>有效性条件-允许“整数”-数据“介于”-最小值“0”-最大值“100”>出错警告-设置提示信息。设置的效果就是在指定的单元格输入 0 至 100 以外的字符后, 便会弹出一个自定义提示信息的警告窗口阻止修改。

③ 实现级联下拉列表。如图 2 所示, 实现 E 列的级联下拉列表, 选定 E3 单元格>选择菜单-数据有效性>设置>有效性条件-允许“序列”-来源“=IF(D3=\$B\$3,\$B\$7:\$B\$8,IF(D3=\$B\$4,\$B\$11:\$B\$12))”, 最后利用拖曳填充, 将格式复制到 E 列的其他行。值得一提的是最终的样表中应当将 B 列隐藏, 这样就不会影响到录入界面和打印。



图 2 级联下拉列表示例

要实现更多定制的界面和功能, 或者在客户端即实现部分数据加工功能, 如数据校验等, 我们可以在表样 Excel 文件内嵌入 VBA 程序, 不过需要注意的是带有宏的 Excel 文件打开时会有安全提示, 且需更改 Excel 软件默认的安全性设置才可执行 VBA 程序。

4 Web页面的设计

在 Web+Excel 模式中, Web 站点作为服务器端, 提供用户身份认证、表样下载、接收上传数据、反馈数据校验信息, 及业务逻辑相关的数据处理功能, 其中的关键就在于如何接收 Excel 文件数据。以下以 ASP.NET 为例介绍 Web 站点如何接收 Excel 文件数据。

ADO.NET 专门提供了 System.Data.SqlClient 命名空间, 其中的 SqlConnection 类用于建立与 SQL Server 服务器的连接。典型的连接字符串如下: “Server = (local); User id= sa; pwd =; Database= master; ”, 其中 Server 为要连接的 SQL Server 实例的名称或网络地址, (local) 表示在本地机器上。Database 表示要连接的数据库名称。User id 和 pwd 为登录名和密码。与 Excel 文件的连接则采用 Microsoft. Jet.OLEDB.4.0 引擎, System.Data.OleDb 命名空间的数据访问类支持对 Microsoft. Jet. OLEDB. 4. 0 引擎的访问。Excel 表格与数据库类似, 也含有多个表格, 每个表格均以二维的形式显示数据。具体实现代码如下:

```
string xlsPath = Server.MapPath("~/myexcel.xls"); //上
```

传到服务器上的 Excel 文件路径

```
string connStr="Provider=Microsoft.Jet.OLEDB. 4.0;"
+ "Extended Properties=Excel 8.0;"
+ "data source="+ xlsPath;
string sql = "SELECT * FROM [Sheet1$]";
DataSet ds = new DataSet();
OleDbDataAdapter,da=new OleDbDataAdapter
```

(sql,connStr);

da.Fill(ds);//将 Excel 表格中的 [Sheet1] 表填充到 DataSet 对象

```
DataTable dt = ds.Tables[0];
```

DataTable questionnaire=new DataTable();//对 DataSet 中的数据进行操作举例

```
questionnaire.Columns.Add("Name");
questionnaire.Columns.Add("Value");
```

for (int i = 1; i < dt.Rows.Count; i++)//实际应用中可根据报表表头调整 i 的起始值

```
{
DataRow dr = questionnaire.NewRow();
dr["Name"] = dt.Rows[i][0].ToString();
dr["Value"] = dt.Rows[i][1].ToString();
questionnaire.Rows.Add(dr);
}输出, 绑定到 DataGrid 控件, 显示 Excel 表内容
GridView1.DataSource = questionnaire;
```

5 与其他报表采集系统设计模式的比较

常见的报表采集系统设计模式有单机版模式、B/S 架构模式和 C/S 架构模式。单机版报表采集系统是一种传统的应用模式, 各级单位依靠单机版软件上报的格式文件传递数据, 系统不提供网络传输功能; B/S 架构的报表采集系统提供一个 Web 站点, 用户通过网页浏览器登录该站点后在线填报数据; C/S 架构的报表采集系统为用户提供一个客户端软件, 提供用户端所有功能并可通过网络向服务器上传数据。如表 1 所示, 列出了 Web+Excel 模式与这三种常见的模式的比较。

6 结语

Web+Excel 模式的本质上也是一种 C/S 架构模式, 只不过客户端无需开发, 而是大家熟悉的 Excel 软件, 由此节约了开发的成本、安装的麻烦甚至培训的成本。而面对报表分级汇总体系, 该模式中 Excel 样表虽然也能在基表上报层级中发挥独特的作用, 如免界面开

发、免客户端安装等, 但由于业务逻辑的和数据处理的复杂, 使得该模式的敏捷实现的优势被抵消, 因此该模式最具优势的应用案例应是: 采用直报类型的报表采集体系, 报表采集指标量大、报表页数多、打印要求高, 而需求是突发的、上报时限要求紧、采集面又十分广, 要求系统能够实现快速敏捷的开发与布署。出了 Web+Excel 模式与这三种常见的模式的比较。

表 1 常见报表采集系统设计模式比较表

Web+Excel 模式的优缺点	优点	缺点
与单机版模式和 C/S 架构模式相比较	1.支持在线提交数据(相对于单机版)。 2.无需开发客户端界面, 数据处理及业务逻辑均在服务器端实现, 开发成本低。 3.不需安装客户端, 避免陷于因各用户系统环境差异造成的复杂情况。 4.由于 Excel 表样不包含代码, 出错可能性小, 避免了客户端需频繁打补丁的情况。 5.提供上报等操作说明后, 几乎不需要对基层单位进行培训。	1.相较于 B/S 架构模式, 先下载后上传的操作方式不如直接在线填报方式方便和直观。 2.在不嵌入 VBA 程序的情况下, Excel 表样的界面元素较少, 可能无法实现复杂的界面要求。 3.如果报表要求采用分级汇总体系, Excel 表样只能在基层上报
与 B/S 架构模式相比较	1.不完全依赖于网络, 录入数据和打印报表可以离线操作, 对带宽要求低, 只要能够满足下载和上传的要求即可。 2.无需开发复杂的 Web 报表界面和打印功能。 3.没有在线人数要求, 服务器负载低。	且相应需在服务器端增加大量的分级汇总的业务逻辑和数据处理功能, 将使得该模式敏捷实现的优势被抵消。

参考文献

- 1 Bullen S, Bovey R, Green J. Excel 专业开发.北京:电子工业出版社,2007.
- 2 厉毅,虞建东.动态网页中对 Excel 表格的转换操.计算机与现代化,2005,(9).
- 3 旭日,薛慧君.一种通用报表的设计和实现.现代电子技术, 2010,(18).
- 4 张永涛.基于 Excel 控件和 SQLServer 的自定义报表系统.电信技术,2010,(9).
- 5 彭江平,王湘群.基于 Excel 与 VBA 的通用报表系统的设计.计算机系统应用,2003,12(1):56-58.