

# 基于 J2EE 架构的 Web 报表系统

## WEB Report Form System based on J2EE Architecture

李智芳 (广州中山大学电子与通信工程系 510275)

蒋芝山 (广州市奥格科技有限公司 510663)

**摘要:**在对企业工作流程与企业报表的特点进行了全面分析的基础上设计了 XQuery 规范的 WEB 报表系统。系统采用了 J2EE 架构设计,利用 XQuery 标准建立了一种 WEB 报表/分析框架,处理统一的 XML 数据源,实现了在全异数据源之间进行查询,并充分利用了 Java 开源项目 IReport 的报表模板图形化制作工具。报表系统提供了各种不同的报表文件形式,可以将 XML 报表转换为 PDF、RTF 以及 HTML 等格式。

**关键词:**J2EE XQuery WEB 报表

### 1 引言

对于许多 B/S 和 C/S 应用系统,用户需要大量的统计分析报表,特别是需要远程查询,显然基于浏览器的方式提供 Web 报表是最方便的。国内报表系统着重于数据的定义、抽取、分析、描述及报表的生成等方面,报表定义的标准化方面考虑的较少,一些解决方案虽然使用基于模板的方法生成报表,但却在模板中使用专有标签,未能以大家都接受的国际标准来定义报表模板。国外同类产品功能较完善,但不太适合我国的实际应用,大多数已有的解决方案使用 XSL 文件来生成报表,但在这种解决方案中存在许多不足之处。随着使用 XML 存储、交换及表示的信息量的增加,智能地查询 XML 数据源并对此数据生成报表的能力变

言的缺乏意味着生成 XML 数据报表仍然很复杂。XQuery 规范提供了用于处理 XML 数据的功能强大且使用方便的语言。为了解决这些问题,开发了 J2EE 架构下 XQuery 规范的 WEB 报表系统。采用了 J2EE 设计架构,通过 WEB 浏览器方式实现数据报表和图形报表的生成、查询和打印。在对报表的结构与数据的描述中,采用了最先进的 XQuery 技术,实现了基于 XQuery 规范的报表/分析框架。该框架所创建的报表不仅能对 XML 数据进行转换,同时还包括对数据进行查询、提取及分析。

### 2 J2EE 体系架构

WEB 报表系统基于业界领先的 J2EE 技术构建,采用 MVC (Model - View - Controller) 设计模式,基于 WEB 的三层体系架构,具有极高的稳定性、安全性及可扩展性、可伸缩性、可维护性。

数据存储采用了 DAO (Data Access Object) 模式和单例模式,这使系统独立于具体的存储策略。体系结构如图 1 所示。MVC 模式由三大部分组成:Model 业务层:应用系统的业务逻辑层,分调度层的 JavaBean 及业务逻辑的 EJB,采用了代理模式使用 SessionBean 实现。View 表示层:数据的可视化展现及与

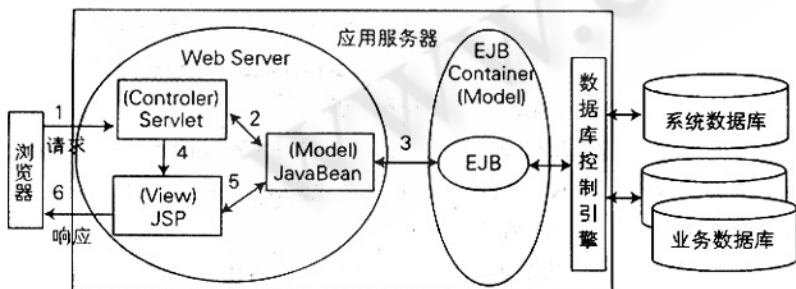


图 1 WEB 报表系统体系架构图

得愈发重要。虽然使用 XML 数据表示带来了极大的灵活性、可扩展性及开放性,但至今为止高水平查询语

用户的交互,使用基于 MVC 结构的 Struts web framework 实现。Control 控制层:接受用户的输入,通知 Model 处理业务逻辑及控制页面流转。

参照 WEB 报表系统的体系结构示意图,对系统响应用户的页面请求处理过程说明如下:

① 控制器 Servlet 接收来自客户端的 Http Request 请求,并转换为 Event 事件;

② 控制器 Servlet 根据相应的 Event 事件调用 Model 层中的 JavaBean 开始进行业务逻辑处理;

③ Model 层中的 JavaBean 实现中间调度,调用业务逻辑层中的 EJB 组件实现业务逻辑,EJB 组件可通过数据库控制引擎实现对业务数据的存取,其中“数据库控制引擎”封装了一切有关数据库的操作细节;

④ 控制器 Servlet 根据业务处理结果,解析路径 URL,调用相应的 JSP 页面;

⑤ JSP 页面调用业务逻辑层中的相应方法取得数据;

⑥ JSP 根据数据,生成 HTML 页面,返回浏览器,进行页面表示。

这种处理(MVC 模式)过程,具有如下主要优点:

① 将业务逻辑、数据展现和控制逻辑三者分开,使得程序设计的过程更清晰;

② 简化了程序扩展时的副作用,从而提高了可扩展性;

③ 将应用程序的业务逻辑与数据表现分开,提高了可复用性;

④ 使得应用程序的发布更容易,因为 MVC 模式中各组件的分界线就是很自然的分发接口点;

⑤ 提高了系统灵活性,因为业务逻辑、控制逻辑和数据展现等部分都可以设计为“可接插”组件。

### 3 使用 XQuery 规范的 WEB 报表框架

#### 3.1 XQuery WEB 报表框架

XQuery 是“XML Query”的缩写。XQuery 旨在为所有类型的 XML 数据源提供标准查询机制。它是 W3C 为使用简单和统一的语法来访问分布式的信息存储系统例如内容仓库,文件,关系型数据库,和对象仓库而制订的 XML 标准。XQuery 使用 XML 结构的优势能表达跨越多种类型数据查询的能力[ ]。“XML 数据”应用程序对于不同的 XML 格式都可用。报表内容

及结构在“XQuery 报表模板”中定义。“报表生成器”对 XML 数据运行报表模板来生成 XHTML 格式的报表。该报表还能生成为其他格式,如 PDF 或 RTF。对于以非 XML 格式存储数据的应用程序,一个附加组件“XML 生成器”能够为报表创建数据的 XML 表达。如图 2 为 XQuery 规范的 WEB 报表系统框架图。

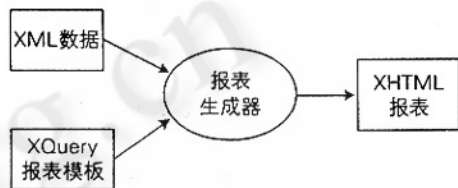


图 2 XQuery 报表框架

#### 3.2 XQuery 报表模板

报表模板是具有内嵌 XQuery 的 HTML 文档。该方法与动态脚本语言(例如 JSP 或 ASP)非常相似。框架的 HTML 部分定义了总体报表布局及格式信息。内嵌的 XQuery 表达式定义了应当从 XML 工程中查询的动态内容。HTML 与 XQuery 能通过两种方式结合。第一,我们可以将 XQuery 表达式嵌入到 HTML 标签中。第二,可以在 XQuery 中嵌入 HTML。这样就为报表模板的构建提供了高度灵活性。

在 XQuery 语言中的一个查询包含了一个或几个查询表达式。XQuery 支持条件表达式、元素构造器和 FOR、LET、WHERE、RETURN (FLWR) 表达式,表达式包括了一些运算符、函数调用、量化、类型检查和路径表达式。一些 XQuery 表达式求出简单节点的值或原子节点的值。查询也能求出这两种节点的次序和简单值。

XQuery 的查询结果是 XML 查询数据模型的实例。它使用了 XML schema 类型并且提供编译时的静态类型和运行时的动态类型,同时也支持原始的和衍生的类型。

一个 FLWR 表达式创建一些绑定,应用一个谓词逻辑,并且产生一个结果集。FOR 用于创建对选定节点的绑定,而 LET 用于创建对单个节点的绑定,FOR 子句可以嵌套使用,这在 XQuery 反复对结果序列进行循环处理时非常有用。

#### 3.3 报表生成器

报表生成器为 XQuery 的执行提供了执行环境。它将 XML 数据及用于报表生成的用户输入绑定到

XQuery 处理器。它同时对 XQuery 处理器的输出进行处理,并且可选地将报表转换为 PDF、RTF、Excel 以及 HTML 格式如图 3。

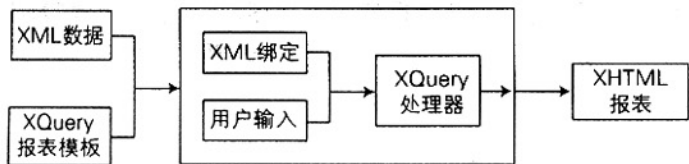


图 3 报表生成器

XQuery 以 XML 为基础,并且具有类似于 SQL 的强大功能,但易于使用的语法。这使得创建模板变得很容易。使用 Report templates 能够缩短开发周期并且使应用程序和报表彼此独立。这就缩短了报表创建的整体周转时间,并且提供了更高的灵活性。

XQuery 是独立于系统的标准,对于所有类型的 XML 数据都可用。大多数的 XML 数据库厂商、工具供应商甚至一些关系型数据库厂商都已经对 XQuery 进行了研发,或者宣布对 XQuery 进行支持的计划。因此,基于报表的 XQuery 模板是极有希望的解决方案。

## 4 报表系统核心模块

### 4.1 核心模块组成

报表系统的核心模块由以下三部分组成:报表设计器、报表生成器引擎、XML 数据生成器。其中报表设计器是利用 JAVA 开源产品 iReport 和 JFreeReport 经功能扩展后得到的。核心模块结构如图 4 所示。

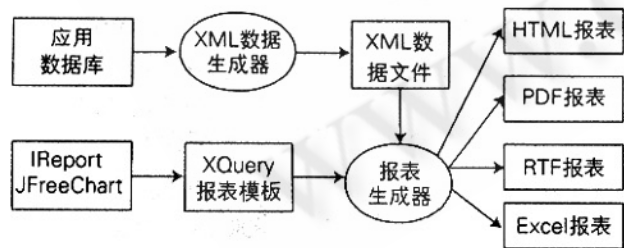


图 4 核心模块结构图

### 4.2 报表设计器

报表设计器用以对 Web 报表模板进行设计,包括 Web 报表版面的设计、数据源定义、数据重组(过滤、计

算等)的函数设计、Web 报表被引用参数的定义等。报表设计器的目的是通过图形界面向导设计出需要的报表模板文件。本系统的报表设计器是在 Java 开源软件 iReport 和 JFreeChart 的基础上扩展而来的。在扩展了 iReport 和 JFreeChart 的部分软件接口的同时也解决了开源软件对中文支持不足的问题。

iReport 是一个 JAVA 开源报表引擎 Jasperreport 的可视化开发工具,类似于 CrystalReport 的开源报表系统,最主要能帮助我们在设计时把数据和表示分开,帮助我们实现 Web 报表的分页打印,分组合计,变量的加减处理,输出的格式化,还有导出 Excel、PDF 等。

Jasperreport 的原始输入就是一个 .xml 文档,然后编译成 jasper 文件(以 .jasper 结尾),真正要使用的是编译以后的 .jasper 文件。

iReport 就是一个用来产生 Jasperreport 需要的 XML 报表模板文件的可视化开发工具,通过扩展 iReport 软件用它来生成 WEB 报表系统需要的 XML 报表模板文件,最关键的扩展就是改写 iReport 软件包中的 it.businesslogic.iReport.Report 类,因为这个类文件是定义报表模板文件的,其中最关键的是改写 Report 类中的 saveXMLFile 方法,因为这个方法是最后生成 XML 报表模板文件的方法,将它输出为本系统需要的模板文件格式即可(即 HTML 中嵌入 Xquery 语法)。

iReport 可以连接各种标准的数据库作为报表的数据源,同时也支持自定义数据源。由于在本系统中的数据源用统一的 XML 数据表示,因而在利用 iReport 设计报表模板文件时应利用 XML 文件数据源。在处理复杂的多数据源时,比如从多个 XML 数据文件中提取数据整合报表时,可以考虑使用自定义数据源。简单的报表,当然用不着自定义数据源。但查询复杂时,把查询分开执行,每个查询只负责向 VO(值对象 Bean)赋值若干个列,程序就会清晰很多。而且,相近的查询还可以进一步封装在一个 Reporter 类里,提供给报表内或者报表之间重用。

JFreeChart 是开放源代码站点 SourceForge.net 上的一个 JAVA 项目,它主要用来各种各样的图表,这些图表包括:饼图、柱状图(普通柱状图以及堆栈柱状图)、线图、区域图、分布图、混合图、甘特图以及一些仪表盘等等。JFreeChart 已经嵌入了 iReport 软件。JFreeChart

基本不需扩展即可使用,只是为了和自己的 WEB 报表系统的整合方便,做了一些调用接口方便的封装。

### 4.3 报表生成器引擎

Web 报表生成器引擎用以在 Web 服务器端解析报表内容,接受并解析应用系统传递来的控制参数,挖掘并分析处理 XML 报表数据,形成完整的 Web 报表 DHTML 文件传递给浏览端。报表生成器引擎的输入是报表模板 XML 文件和报表数据 XML 文件和可选的 CSS 样式文件,输出是 HTML 格式的 WEB 报表文件,可以直接在浏览器中显示。报表生成器也可以根据条件不同生成 RTF、PDF、EXCEL 等不同格式的报表文件。关键是 XQuery 语法解析和报表格式文件的生成。

报表生成器在对于 XQuery 语法处理得到数据后依据报表模板的要求生成相应格式的报表文件。生成不同格式的报表实际就是通过对报表模板的解析结合报表数据文件和报表样式文件将报表表格打印成相应格式的文件的过程。此 WEB 报表系统对于报表规定了以下概念模型。

报表的各部分都是以表格来组成的;报表 = 报表头部表格 + 报表主体 + 报表尾部表格;报表主体 = 报表数据表格 + 报表列头表格;表格是表行的纵向集合;表行是表单元的横向集合;表单元简称为一个单元格由数据和格式组成。

典型的两类中国式报表:

① 普通报表——数据按行排列,列头描述每列数据的含义。数据可以分组统计(统计方法例如:求和、求平均值)。

② 交叉报表——行头和列头分别指出原始数据的某个属性(字段),与行头同行并与列头同列的交叉部分表示符合列头和行头条件的某个统计值(统计方法可以定义,例如:求和、求平均值)。

### 4.4 XML 数据生成器

XML 数据生成器用以从不同的异构数据源中提取和生成统一的 XML 格式报表数据。由于报表数据保存于 XML 文件中,因而很容易实现报表数据的审核和数字签名功能。本系统的关键是从统一的 XML 数据中查询统计以得到需要的报表,因而将不同的数据用统一的 XML 文件形式表示和解析各种 XML 数据文件是十分重要的。

XML 数据生成器就是在处理报表之前将来自于不

同数据源的数据统一转化为一致的 XML 格式数据,这也是为了响应项目要求的数据标准化的需求,当数据被转化为统一格式的 XML 数据后,数据在各个单位之间的数据交换和数据通过 Internet 传递都成为了一种简单而方便的事情。

XML 的应用似乎是无穷无尽的,利用 XML 来完成数据的表示和交换(针对 XML 的简单 API(SAX)和文档对象模型(DOM)语法解析,不同的文档类型定义(DTDs)和概要(schemas))。在 J2EE 架构中恰好数据表示和交换功能是 EJB 组件模型中持久化服务(persistence services)的一部分。修改甚至写入某个 XML 文档而经常被读入到某个对象模型中。作为例子,假定我们正处理多种类型的媒体(图片、视频、文本文档等等),并且用下面这个简单的 XML DTD 来描述这些媒体的元数据:

以下是一个基于上述媒体 DTD 的 XML 文档,描述了与某个课程讲座相关的内容:

如果要把这些数据加载到 Java 应用中,我们可以从当前众多的 Java 语言 XML 解析器中选用一个,通过它将 XML 数据装入一个 DOM 文档,最后遍历文档,将所有这些数据转换到我们应用系统的对象模型中。

## 5 结束语

目前本系统在报表样式的丰富性上还有很多需扩展的地方,尚不能利用报表模板图形设计器灵活定义各种复杂报表,如嵌套报表,复杂的交叉报表等,此时只有通过手工定义报表模板来实现。但足以满足日常工作的需要。报表系统直观可视化、快速方便、容易使用,支持多平台的应用,能够对报表轻松实现格式化和自定义。系统提供标准的 API 和接口,可以和用户现有的 Java 程序无缝结合,减少用户的重复投资。

### 参考文献

- 1 阎宏著,Java 与模式,电子工业出版社,2002,10。
- 2 冯大鹰、晋玉星,通用报表的 xml 描述,计算机应用与软件,2005,8。
- 3 王华麟、杨育普,使用 XML 构建协作式 WEB 解决方案,计算机工程,2000,9。
- 4 杨芳、陶世群,基于 xml 异构数据库系统的集成设计,计算机工程与设计,2005,7。