

基于 .NET 平台的复杂报表管理系统的结构设计

'Structure Design of Complex Report Forms Management System Base on .NET Platform

陈斌 彭达洲 胥布工 黄健平 (广州华南理工大学 自动化科学与工程学院 510640)

摘要:对于可实现完全自定义、多维结构,基于工作流的,操作上可以一级提交多级审批的一类复杂报表管理系统,提出了在 .NET 平台下采用面向对象的开发模式,以及运用 ASP.NET、C# 等实现技术,并且给出了系统核心的数据结构设计和程序结构设计。

关键词:复杂报表 .NET 平台 结构设计 工作流

1 引言

近年来基于 B/S 模式下的 Web 报表管理在悄然走俏,同时,现代企业下的报表需求呈现出更复杂(多维),自定义更灵活,管理更简便的趋势。如何实现 B/S 方式下的 Web 报表管理是许多用户或开发商都会遇到的问题。而作为新时代 Web 开发的宠儿——ASP.NET 提供了目前最先进的 Web 开发平台,它采用了微软最新的 DOT NET 框架,是 DOT NET 框架的核心元素。

2 系统需求分析及解决方案

2.1 需求分析

本系统的需求所涉及的功能和性能方面比较全面,在此只列举部分较为核心的需求,如下:

(1) 报表可完全自定义,即可方便的定义新的报表,删除不用的报表定义,并且定义好的报表在系统运行时动态生成;

(2) 报表的多维性,在 2 维的基础上可以延伸到 3 维和 4 维,即行和列的定义均可再归类细分,使得报表更灵活,适应现代企事业单位复杂报表的要求;

(3) 报表具有两种提交模式,一种是直接在系统中在线填写数据提交,另一种是离线编辑再通过上传文件进行提交,相应具体的模式在报表定义时即确定。

(4) 报表具有工作流特性,提交的报表可根据设定好的路径进行多级审批,即实现一级提交,多级审批;

(5) 每类报表的审批路径可通过报表授权模式实

现对审批路径的完全自定义;

(6) 用户对报表的操作通过报表授权的模式进行,可以灵活的指定某个用户可以操作哪些表格,对每个表格具有哪些操作权限;

(7) 对于大型集团企业,可适应跨国、跨地域或跨门类的报表提交,即同一报表可隶属于不同的分公司/门类;

(8) 报表数据可系统自动进行汇总、分类和生成图表(柱状图、饼状图等),并可产生分析结果;(9) 报表具有导入导出数据、打印等功能;

(10) 系统易于维护和再次修改开发,移植性要好,具有良好的扩展接口;代码量要少且运行效率高;

2.2 解决方案

基于以上需求分析,提出以下解决方案:

2.2.1 实现技术

表 1 技术方案

	方案
开发平台	采用微软最新 .NET 平台
系统模式	基于 B/S 构建
开发技术	ASP.NET, ADO.NET
程序语言	C#
开发模式	面向对象
辅助技术	JavaScript, Flash 等
数据库选型	SQL Sever 2000
界面模式	采用日历导航

2.2.2 功能模块划分

(1) 系统登录模块

提供用户登录界面,并识别用户类型、权限等信息。

(2) 报表管理模块

主要完成报表的增加、自定义、修改、删除、授权等功能。

• 报表定义子模块——完成新增报表的完全定义,定义报表的基本属性、提交模式、审批级数等。

• 报表编辑子模块——对既有报表进行修改、删除等工作。

• 报表授权子模块——对用户、对象(包括报表

名称和隶属)和操作类型三者关系进行关联,完成报表授权。

(3) 报表操作模块

完成报表的查看、录入提交和多级审批功能。

• 报表录入子模块——对新报表进行数据录入和提交,并存储到数据库。

• 报表审批子模块——对已提交数据进行第一级审批,对上一级审批通过数据进行次级审批。

• 报表查看子模块——对自己提交的数据,自己审批的数据进行时候查看;相应权限用户对多级审批通过完成的数据进行查看。

(4) 报表运算模块

主要完成已审批完成的数据进行自动汇总、生成图表等工作,另外提供数据的导入导出、打印等功能。

(5) 系统管理模块

完成系统用户的增删管理工作,以及系统权限的授予,提供用户可修改密码等。

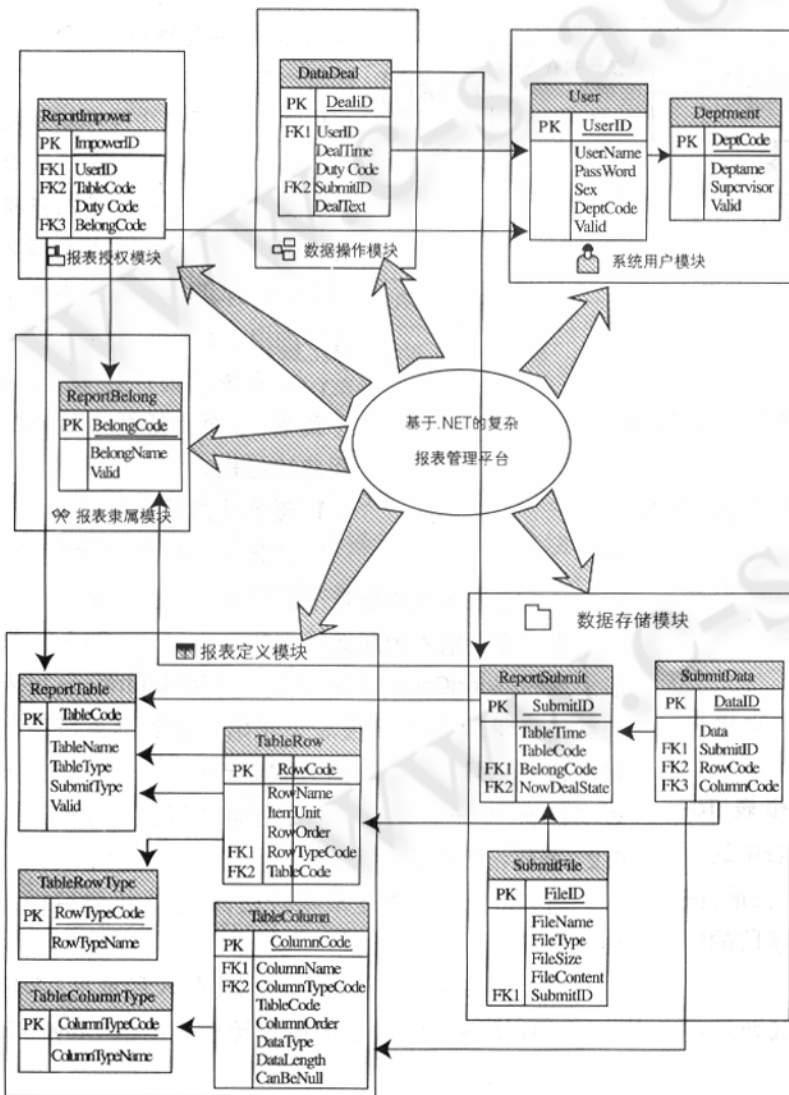


图 1 系统数据结构实体关系图

3 系统数据结构设计

3.1 数据结构图

由于本系统要求报表定义和操作的高度灵活性和报表复杂性(报表需具有 2~4 维结构,能够自定义每类报表的工作流程,实现报表的一级提交,多级审批),而且报表能够进行灵活的授权……这些需求都决定了系统数据结构设计的极端重要性,数据结构的是否合理,直接影响了系统的开发和最终实现。

下面提供了围绕用户、报表定义、报表隶属、数据存储、数据操作和报表授权这几个核心模块的数据结构,见图 1。

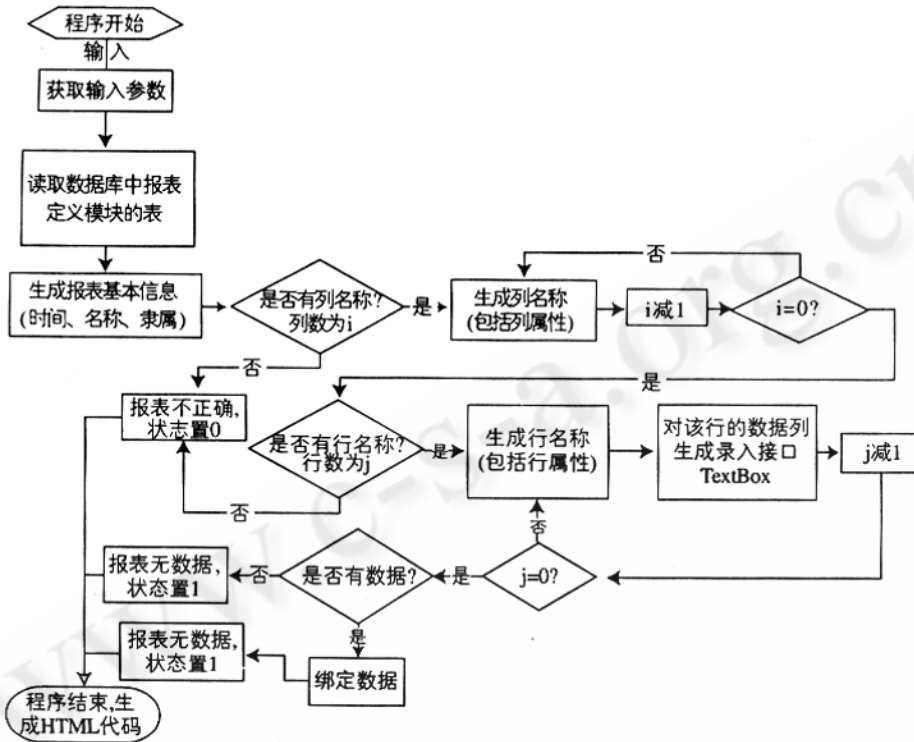
3.2 部分表和字段说明

① 多个表中的 Valid 字段:用于标识该记录的有效性,对于一些比较重要且与其他表主外键关联的记录,直接删除是不妥的,比较好的做法是执行删除操作时将该记录的 Valid 置 0(假设 1 为有效,0 为无效);

② 表 ReportTable 中的 TableType 字段: 标识报表类型, 一般为日报表、周报表、月报表、季度报表和年报表等;

⑧ 表 DataDeal: 记录用户的操作记录, 暂存、提交和审批;

⑨ 表 ReportImpower 中的 DutyCode 字段: 标识用户对某个隶属的某个表格的操作权限, 是提交、1 级审批还是 n 级审批;



报表状态说明: 0 报表不正确, 1 报表无数据, 2 报表有数据

图 2 动态生成报表程序流程图

③ 表 ReportTable 中的 SubmitType 字段: 标识提交类型, 本系统分为两种: 在线填报提交和离线编辑并上传文件提交;

④ 表 ReportTable 中的 DutyNum 字段: 标识审批级数;

⑤ 表 TableRow 中的 RowOrder 字段和表 TableColumn 中的 ColumnOrder 字段: 分别标识表中的行名称和列名称的排序号, 以保证动态生成的表格能按照定义时的行列顺序一致。通过改变这两个字段的值即可调整行列顺序;

⑥ 表 SubmitData 和 SubmitFile: 分别存储两种提交模式下的数据;

⑦ 表 ReportSubmit 中的 NowDealState 字段: 标识目前此记录的操作状态, 区分是暂存数据、提交数据、某一级审批中数据、完成数据还是被退回数据;

4 系统程序结构设计

本系统采用面向对象的开发技术, 在 .NET 框架下, 可以实现真正的用户界面与程序代码分离 (即“代码隐藏” CodeBehind 技术)。利用此特性, 我们可以把每一功能模块进行逻辑划分成程序块, 再把这个程序块封装成类 (.cs 文件), 然后再把所有的类文件编译成一个动态链接库 (.dll 文件)。当系统操作需要某个功能时, 只要调用动态链接库中的相应类即可。

本系统的程序块较多, 限于篇幅, 下面介绍两个核心的程序块结构。

4.1 动态生成报表程序

这是一个系统基本程序, 因为系统在很多操作场景下都需要动态生成报表。

本程序的输入为所选报表的名称 (TableCode)、报表隶属 (BelongCode) 和报表时间 (TableTime, 日报表为某天日期, 周月季年报表为起始日期到结束日期)。程序输出为所生成报表的 HTML 代码, 有三种状态: 报表不正确、报表正确但无数据、报表正确且有数据。

4.2 操作报表程序

当完成上述动态生成报表后, 即进入报表待操作模式。

报表操作分为三个状态: 查看、录入和审批 (多级审批中具体是哪级审批由报表授权表中的 DutyCode 字段标识)。

此程序判断环节较多, 主要是判断报表状态、用户的操作状态、用户对于此报表的有效权限。根据每一个判断的不同, 程序走向也将不同。另外本程序过程

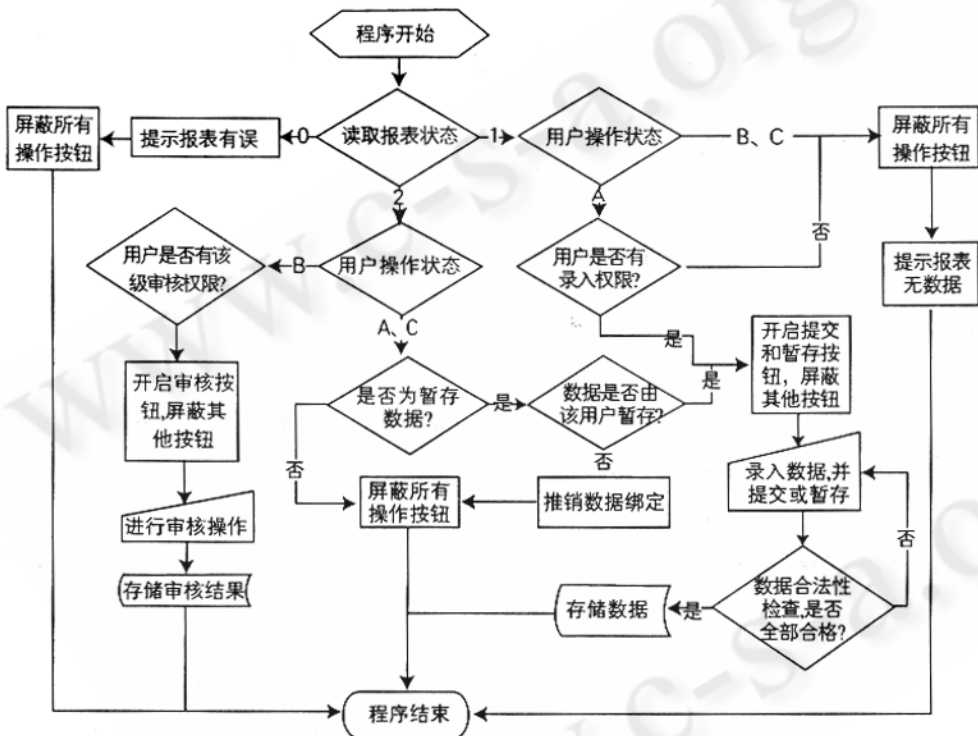
中涉及用户录入数据这一环节,当然,录入数据分为填写数据和上传文件两种情况。

5 结束语

基于 .NET 平台的复杂报表管理系统,实现了对一类复杂报表的最大限度的灵活自定义、报表网上工作流程转、报表数据综合处理等。它的面向对象的开发模式和最前沿的开发工具 ASP.NET,保证了其良好的可扩展性、移植性和操作性。更加适应当今网络化办公及数据管理的趋势和大型企业事业单位的需求。

参考文献

- 1 黄忠成 编著,深入剖析 ASP.NET 组建设计,北京电子工业出版社,2004.5。
- 2 嘉木工作室 编著,C#编程思想,北京机械工业出版社,2003.3。
- 3 杨正洪 等编著,SQL Server 2000 关系数据库系统管理和开发指南,北京机械工业出版社,2001.9。



报表状态说明: 0报表不正确, 1报表无数据, 2报表有数据

操作状态说明: A录入, B审核, C查看

图 3 操作报表程序流程图

除了这两个核心程序块以外,还有报表定义程序块、报表授权程序块、报表数据汇总计算程序块、报表数据导入导出程序块等等。在此就不一一介绍了。