

Web 信息系统中上传文件的管理

黎 昂^{1,2}, 陈青青^{1,2}, 刘 康¹

(1.湖南工学院现代教育技术中心 湖南 衡阳 421002 2.中南大学信息科学与工程学院 湖南 长沙 410083)

摘 要】 在 web 信息系统开发中,经常需要上传文件。目前,大多数基于 Web 的应用系统都采用第三方组件或 FSO 对象将文件上传到服务器中,但由于上传的文件和信息系统中的数据库相互独立,所以,当系统长期应用或升级时会存在很多无用文件,它们既占用空间又不利于管理。本文介绍一种 Web 系统中基于数据库管理上传文件的方式,能很好的解决这个问题。

关键词】 文件上传,Web,数据库

一、引言

随着 Internet/Intranet 的普及,办公管理工作正逐步进入无纸化阶段,应用程序的服务模式也逐渐从客户/服务器模式向浏览器/服务器模式转变。文件存取是办公系统的基本需求,因此,基于 Web 的应用系统一般要求具有文件上传下载的功能。在 web 系统开发中,经常会遇到文件上传这样的问题,如图片的上传、Word 文件的上传、Flash 的上传等。

二、文件上传的一般方法

在互联网应用的初期,ftp 是文件上传的标准方法,使用 ftp 上传文件一般有两种方式:一种是管理员需要管理大量的用户名与密码;另一种是允许匿名上传。而允许匿名上传,无法保证上传的文件不被别人删除,并且使用 ftp 也无法实现智能化管理,如要求仅能上传某一类型的文件,或者不同类型的文件上传到相应的文件夹下,就无法做到。随着网络的发展,B/S 模式的普及,需要使用更方便、更容易为大众掌握的文件上传方法,而基于 Web 方式的文件上传能满足上面提出的所有要求。

采用 Web 方式,文件上传到服务器之后,可以采取两种存储方式:一是将文件的内容及相关信息直接保存到数据库中;二是文件直接存储方式,即把文件直接存放到某个目录中。显然,采用第一种方式的系统固然有其优点,但在整个处理过程中,对客户上传的数据流分离^[1]、上传后的查看等方面并不是太方便,而第二种方式却能很好地避免第一种方式引起的问题。目前,大多数基于 Web 的应用系统都采用第三方组件或 FSO 对象将文件上传到服务器中,但长期的应用或系统升级时,却发现问题的也随之而来,如:无用文件的管理、信息系统中数据库与上传文件的兼容性问题。为解决此问题,本文介绍一种 Web 系统中基于数据库管理上传文件的方式。

三、上传文件的管理

1、数据库设计

在 Web 信息系统中,一般采用前台(Asp、Jsp、Php 等技术)+后台数据库(SqlServer、MySql、Access 等)的模式。而重要的数据信息,如信息内容等,一般存储在数据库中。其表主要部分设计形式一般如下:

字段名称	类型	备注
id	Int(4)	信息编号、主键
biaoti	Char(30)	标题
neirong	Ntext(16)	信息内容
time	Datetime(4)	信息发布时间

当编辑某条信息的过程中,如果需要上传文件,将文件直接上传到服务器,然后对它们进行管理一种较好的方式是将文件相关信息,如文件名、上传时间等,也存放到数据库中。由于编辑中的信息并不一定最终发布,因此首先将上传文件的记录添加到 TempFile 表中,当确定发布时再将其添加到 File 表中。其表可设计如下:

TempFile 表(存储上传的临时文件)

字段名称	类型	备注
Client_Name	Char(50)	上传前的客户端名称
Server_Name	Char(30)	上传后的服务器端名称

File 表(存储相应信息中上传的文件)

字段名称	类型	备注
contentId	Int(4)	信息编号、外键
Client_Name	Char(50)	上传前的客户端名称
Server_Name	Char(30)	上传后的服务器端名称

2、系统功能分析及其实现

Web 信息系统一般包括添加信息、删除信息、修改信息、查询信息等基本功能,它们主要依靠数据库的一些基本操作,但某条信息需要上传文件时,主要需要以下功能:

(1)上传:当添加或修改信息的时候,可以利用组件或 FSO 对象上传需要的文件到服务器目录中,并将相应的信息存储到 TempFile 表中。其实现代码如下:

```
Set upl = Server.CreateObject("SoftArtisans.FileUp")
upl.SaveAs Server.mappath(SavePath & strFileName)
call openConn()
set rs=server.createobject("adodb.recordset")
sql="insert into tempUpFile (ClientName,ServerName) values ('"&upl.Form("
FileName").ShortFilename&"','"&SavePath & strFileName&"')
rs.open sql,conn,1,3
call CloseConn()
```

(2)删除:当添加信息时,可删除已上传的文件并将添加的文件信息记录从 TempFile 表中删除;当修改信息时,可删除已上传的文件并将添加的文件信息记录从 File 表或 TempFile 表中删除(如下图);当删除带有上传文件的信息时,要将已上传文件和 File 表中的文件信息记录删除。其实现代码如下:

已上传的文件:

删除文件:

```
Set fso = Server.CreateObject("Scripting.FileSystemObject")
strRubbishFile=server.MapPath(".")&RubbishFileName
if fso.FileExists(strRubbishFile) then
fso.DeleteFile(strRubbishFile)
end if
删除数据库中文件记录
call openConn()
set rs=server.createobject("adodb.recordset")
sql="delete from tempUpFile where serverName='"&RubbishFileName&"'"
rs.open sql,conn,1,3
sql="delete from UpFile where serverName='"&RubbishFileName&"'"
rs.open sql,conn,1,3
call CloseConn()
删除信息记录
sqlDel="select a.id,b.servername from Article a,upfile b where a.id=b.articleid
and a.id='"&ArticleID&"' and a.adminId='"&session("adminId")&"' and a.dept_id='"&
&session("dept_id")&"'"
Set rsDel= Server.CreateObject("ADODB.Recordset") (下转第 107 页)
```

档,即利用 XML Schema 为异构数据源提供一种统一的数据转换结构的描述与约束。各企业或各企业应用系统间,就是通过该信息交换集成平台,将可通过数据泵生成标准结构 XML 数据文件的各异构数据源联系起来,并实现数据的集成和共享。

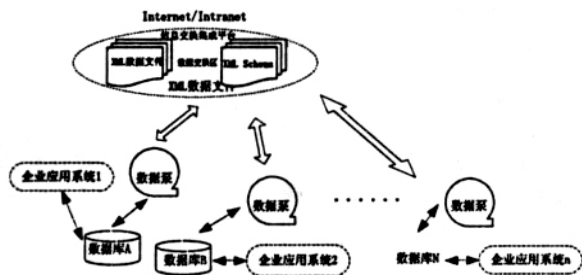


图2 应用体系结构

(1) 数据泵的设定

在上图中看到数据库到集成平台的传输中间是通过数据泵来实现的,这里可将数据泵设计成应用信息系统(如 ERP、SCM、CRM) 外挂的一个独立程序,也可作为应用系统的对外接口组件。其主要功能是定义本地应用系统数据库中有哪些数据表与信息集成平台给出的 XML Schema 对定义各数据表中的字段同其对应 Schema 中的标签间的映射关系;完成应用信息系统中数据到 XML 文件数据或 XML 文件数据到应用信息系统中数据的转换,即对于本地数据库来说,一种数据导出与导入的过程。

(2) 应用实现技术

异构数据库间信息交互的过程如图3所示。先给出各交换信息的 Schema 标准结构定义,然后确定应用系统数据到标准 XML 数据文件的转换映射模式,接着实现数据的转换,将文档上传到信息交换集成平台。

6. 结束语

基于 XML 的异构数据库之间数据交互的方法,不但可以为供应链上各企业提供了一条基于网络的畅通的信息通道,而且

能将企业电子商务系统与内部已有的各信息系统无缝集成,从而把客户和市场的需要纳入企业集成化供应链管理信息系统,实现开放式的企业运作模式,保证内外信息流的融合与畅通,增强企业的信息自动化处理能力和对市场扰动的鲁棒性。文中基于 XML 的 EAI 可以为企业提供一个廉价、有效的信息交换通道,解决了企业应用集成(EAI)中数据的平滑的、自动的、实时的通信,增强了企业及企业联盟内的信息交流,从而提高了整个企业及企业联盟的市场竞争力。

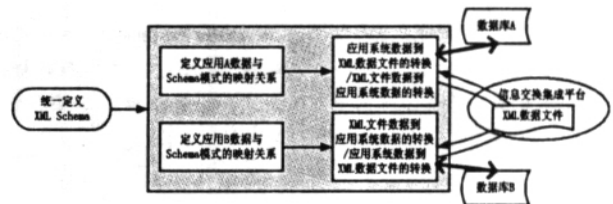


图3 异构数据交换集成流程

参考文献:

1. May Roth. 从数据管理到信息集成:自然的演变[M]. 电子工业出版社, 2002, 123- 130.
2. 林春梅, 金鑫, 何跃, 陈家训. 基于 web 服务的企业应用集成[J]. Journal of Donghua University, 2004, 125- 132.
3. Lear A.C. XML Seen as Integral to Application Integration [J]. IT Professional, 1999, 12- 16.
4. AMT. 成功实施 EAI 的十把钥匙[M]. 国家新闻出版社, 2003, 214- 216.
5. Julie Gable. Enterprise [J]. Information Management Journal, Lemexa: Mar/Apr 2002, Vol. 36, Iss 2; Page(s): 48- 51.
6. Chelsea Valentine, Lucinda Divested Tittel. 毛选, 魏海萍等译. XML Schema 数据库编程[M]. 北京: 电子工业出版社, 2002.
7. 钟永光, 黄立平, 徐锡荣. 基于 EAI 构建企业外部服务平台[J]. 管理现代化, 2002(1): 15- 17.
8. 张博, 杨丽君. 企业应用集成[M]. 北京: 机械工业出版社, 2003, 354- 360.

(上接第 144 页)

```
rsDel.open sqlDel, conn, 1, 3
do while not rsDel.eof
    call DelFiles(trim(rsDel("servername"))) & ""
rsDel.movenext
loop
```

(3) 清理无用上传文件: 当添加或修改信息时, 可能进行了文件上传的操作, 但并未最终完成信息的编辑, 这些已上传文件便成为无用的上传文件, 他们既占用空间, 又不利于管理, 所以要进行清理。由于这些文件信息的记录都会保存在 tempUpFile 表中, 因此, 可以从表中查询初相应信息将已上传文件物理删除并将记录从 tempUpFile 表中删除。其实现代码如下:

```
set rs1=server.createobject("adodb.recordset")
sql="select serverName from tempUpFile"
rs.open sql, conn, 1, 3
do while not rs.eof
    call DelFiles(trim(rs("servername"))) & ""
rs.movenext
loop
sub DelFiles(strUploadFiles) '删除文件及文件信息
if strUploadFiles="" then exit sub
Set fso = CreateObject("Scripting.FileSystemObject")
if fso.FileExists(server.MapPath("") & strUploadFiles) then
```

```
fso.DeleteFile(server.MapPath("") & strUploadFiles)
set rs1=server.createobject("adodb.recordset")
sql="delete from tempUpFile where servername='"&strUploadFiles&"'"
rs1.open sql, conn, 1, 3
end if
end sub
```

四、结论

目前, 程序员开发 Web 信息系统的技术在不断进步, 但广大用户在使用过程中的却会不断的出现问题。我们的系统采用该方式对文件进行管理后, 更好的将信息系统和上传文件结合了起来, 在长期的应用中, 效果良好。

参考文献:

1. 麦范金等. 基于 asp 和数据库技术的文件存取处理技术[J]. 桂林学院学报, 2005, 25(3): 353- 355.
2. 王新房等. 基于浏览器的文件上传技术研究[J]. 计算机应用, 2000, 11: 57- 59.
3. 邓文渊, 陈惠贞, 陈俊荣. ASP 与网络数据库技术[M]. 中国铁道出版社, 2005.
4. 赵增敏等著. SQL Server 2000 案例教程. 北京: 电子工业出版社, 2005.