

不同数据源系统之间的数据同步和整合^①

刘浩

(中国江苏网 江苏中江网传媒股份有限公司, 南京 210005)

摘要: 信息系统实施过程中, 由于需要其他系统基础数据支持, 因此根据项目的实际情况同步数据。以一个数据整合项目为例, 提供了一个不同数据库之间的数据自动导入和同步的解决方案。

关键词: 数据库; 数据导入; SQLSERVER; ORACLE

Data Between Different Source Data System Synchronization and Integration

LIU Hao

(JSChina, Nanjing 210005, China)

Abstract: During info system implementation process, because of the need to support the basic data from other systems, according to the actual situation, we need the data of synchronization. Based on a data integration project as an example, we provide a different database of import and automatic data synchronization solution.

Key words: Web information extraction; knowledge presentation; data intensive web pages; ontology-based keyword library

1 引言

在项目中, 由于客户所使用的系统经常是由几家不同的公司提供的产品, 而客户希望这几个产品使用共同的一套基础数据, 协同运行。这就需要在信息系统实施部署的过程中, 对共同使用的基础数据进行定时自动同步, 以达到数据库层次的系统整合。在整合的过程中, 提供数据源的是 SQLSERVER 数据库, 最终使用数据的是 ORACLE 数据库。

2 项目实施的难点

2.1 系统之间的逻辑关系

两套系统各自具有不同的数据库结构和数据关系, 了解两套系统中基础数据的逻辑对应关系是此次数据同步整合的前提。

2.2 SQLSERVER 和 ORACLE 存储过程的设计和编写

虽然同为关系型数据库, 但是 SQLSERVER 和 ORACLE 数据库还是有不同之处, 在存储过程的编写方面, 差异非常明显, 特别是游标的调用。因此存储过程的设计和编写是数据同步整合的关键。

当查询返回结果超过一行时, 就需要一个显式游标, 在游标定义中 SELECT 语句中不一定是表, 也可以是视图, 也可以从多个表或视图中选择的列, 甚至可以使用*来选择所有的列, 游标允许对结果集, 如查询结果, 进行单独的操作, 如逐行处理。

在插入存在自增长字段数据的过程中, SQLSERVER 提供了一个方法可以在插入数据的同时取回生成的自增长的值, 以此用于其他关联表的处理, 由于此处理在同一个事务中, 并在处理过程中将此表锁住, 因此可以确定取回的自增长的值是刚插入的记录生成的。但是在 ORACLE 中却没有相应的方法用于处理类似的问题。通过先插入记录, 再以 max 的方式取回自增长的值, 在频繁调用过程中会造成取值偏差, 影响同步数据的正确性, 破坏了数据完整性约束, 造成同步失败。现在将插入记录和取自增长的值放在同一个事务中执行, 通过存储过程实现返回值。

2.3 数据完整性约束控制

由于数据源的数据本身的存储问题, 例如在 SQLSERVER 中使用 char 类型字段存储数据, 导致数

^① 收稿时间:2011-05-12;收到修改稿时间:2011-06-21

据长度不足时,用空格在字段后补足的情况,同步后的数据出现空格的问题。ORACLE10g 支持 SQL 正则表达式和表达式过滤的功能,可以在数据库中完成字符串的过滤和处理,保证了同步数据的正常使用。

在同步的过程中,数据源提供的数据并非百分百符合学习平台数据要求,缺少必要的关联关系,主外键关系,甚至会有重复数据和垃圾数据,在同步的过程需要对数据源数据进行逐条筛选。对已有的数据,进行部分更新。插入表中不存在的记录,并根据关联性对多个表进行插入操作,将这一系列操作放在一个事务中,一旦一条记录插入失败,则整个回滚。由于一些不可知的原因,从上一个数据库中无法正常提取数据,或提取的数据有部分缺失,因此通过筛选,发现不符合完整性约束的数据,不能执行,等待以后正常同步时,再做处理。

2.4 ODBC 数据源

在数据同步的时候,需要在不同数据库之间进行数据操作,而且数据是从 MSSQLSERVER 向 ORACLE 进行传送,微软公司开放服务结构中有关数据库的一个组成部分,即开放数据库互连(Open Database Connectivity, ODBC),它建立了一组规范,并提供了一组对数据库访问的标准 API(应用程序编程接口),这些 API 独立于不同厂商的数据库系统,也独立于具体的编程语言。一个基于 ODBC 的应用程序对数据库的操作不依赖任何数据库系统,所有的数据库操作由对应的数据库系统的 ODBC 驱动程序完成。因此能以统一的方式处理所有的数据库。正适合本项目的跨数据库操作。

2.5 同步数据的自动执行

在数据同步的过程中,每一个存储过程的调用都是有严格的先后次序的,为实现数据同步的自动调用,必须保证在自动调用的过程中严格按照先后次序执行存储过程,在 SQLSERVER 中可以通过可视化的界面中设置作业,安排作业的每个步骤的先后次序来实现。在 ORACLE 中,需要借助任务队列管理器在指定的时间去执行同步存储过程,在同步中对每一个步骤应留出足够的时间,由于两个不同数据库之间无法传递和获取对方数据库的作业信息,因此无法知晓对方数据库的作业是否已经完成,或调用失败,因此在两个数据库作业之间要留有充分的缓冲时间。

3 项目实施

在信息系统中,用户信息等数据是每个平台的基础数据,客户由于业务需要经常同时使用不同公司的产品,而这些基础数据却大致相同。但是却需要在每个系统中都维护一套基础数据,并以人工的方式保持这些数据的同步,造成管理效率的低下。因此以一个系统的基础数据为主,其他系统同步此基础数据,并保证定期更新,以解决此弊端。本文以学校教务和学习平台的整合为例,介绍数据同步在实际项目中的应用。

3.1 背景介绍

教务系统部署的比较早,学校日常的教学信息,教学计划等都存储在教务系统中。学习平台的实施,是为了给学校的教学提供一个新的教学辅助方式。在新老校区之间,架设一个平台,使教学能够跨越时间、空间的限制,缩短教师和学生交流的成本和时间。但是要真正使用的学习平台,需要学校提供教师和学生的信息以及教学计划,学生选课情况等基本数据。由于数据量巨大,而且会有一定的变动,因此通过数据导入的方式,从教务中将数据导入到学习平台中,并定期对学习平台的数据进行同步更新,实现基础数据的同步使用。

3.2 实施构想

这两个系统,所使用的数据库和系统环境差别很大。教务使用的是 windows 平台,SQLSERVER2000 数据库。学习平台使用的是 Linux 系统(RedHat),ORACLE10g 数据库。通过一个中转服务器,将教务中的基础数据导入学习平台中,并实现数据的同步更新。



图 1 数据导入流

3.3 教务数据

教务系统的数据部署在 SQLSERVER2000 上,基于系统安全性的考虑,针对学习平台需要的基础数据建立了几张视图,用于数据导出。每个视图涉及的信

息分别是课程基础信息（课程编码和课程名称），教师信息（教师编号，教师名称和邮件地址），学生信息（学生学号，姓名，班级编号和入学年份），教学计划（年度，级别，课程编码，合班标志，教师编号和课程开始时间）其中年度，级别，课程编码和合班标志为关键字，学生所学课程。

在整个数据导入同步的过程中，对于教务数据的删除，并不立即进行同步删除，由于基础数据在学习平台中会产生很多学习记录，涉及多个表，一旦由外界强行删除，会破坏学习平台数据库中的数据完整性，导致大量的垃圾数据，因此对学习平台的基础数据表只进行添加和修改操作，不删除基础数据。一切删除操作在平台管理界面进行。

3.4 中转服务器

中转服务器的系统环境为 windows2000sp4，数据库使用 SQLSERVER2000 sp4。中转服务器的主要职责是 1)从教务数据库中将学习平台需要的基础数据全部导入到中转服务器中。2)将导入的教务数据进行初步处理。3)将 ORACLE 中接收基础数据的辅助表中的数据清空。4)把进行初步处理过的数据导入到 ORACLE 数据库的辅助表中。其中辅助表作为将要导入的数据临时存储表，经过筛选和过滤后导入正式表中。

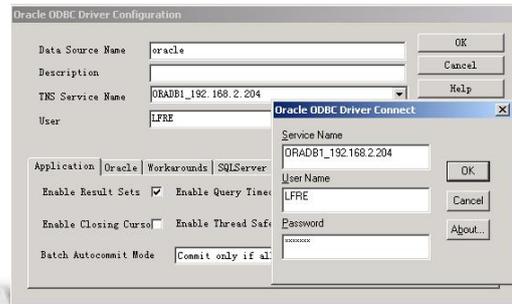
由于同是 SQLSERVER2000 数据库，教务和中转服务器的数据导入，使用的是 SQLSERVER2000 自带的数据库导入/导出工具，并将此调用保存成本地包，为以后定时导入数据时使用。

在中转服务器的第二步操作中将教学计划和学生的学习计划中的关键字：年度，级别，课程编码和合班标志字段合并成一个字段。这样可以方便两个表的关联查询，大大降低查询的资源消耗，大幅度提高查询的效率，减少了 ORACLE 数据库端同步的时间和负载。

在中转服务器的第三步和第四步操作中，由于数据库不同，需要借助 ODBC 数据源来实现数据清空和数据导入。

首先必须用 ODBC 管理器注册一个数据源，管理器根据数据源提供的数据库位置、数据库类型及 ODBC 驱动程序等信息，建立起 ODBC 与具体数据库的联系。这样，只要应用程序将数据源名提供给 ODBC，ODBC 就能建立起与相应数据库的连接。在 ODBC 中，ODBC API 不能直接访问数据库，必须通

过驱动程序管理器与数据库交换信息。驱动程序管理器负责将应用程序对 ODBC API 的调用传递给正确的驱动程序，而驱动程序在执行完相应的操作后，将结果通过驱动程序管理器返回给应用程序。



ODBC 数据源设置

学习平台登陆时，使用的是 MD5 加密码进行帐号效验，所以在数据初步处理中，需要对用户的密码的明码做加密处理，将明码转换成相对应的 MD5 加密码。因此在此数据库中编写一个 MD5 加密码生成函数，用于明码的转换。

3.5 ORACLE 同步

当数据从中转服务器的数据库上导入到 ORACLE 的辅助表中，整个同步工作进入最关键的部分：ORACLE 数据同步。

由于基础数据表中有关键字和对应的外键关系，一旦出现数据重复，将无法导入，因此在导入数据的过程中，对辅助表中的数据进行进一步的筛选和区分。

全部用 ORACLE 中的存储过程实现数据的筛选导入。在导入数据过程中，对目标表中没有的数据进行 insert 操作，对目标表中已有的数据进行 update 操作，且只更新部分字段。以下是课程信息数据导入的存储过程：

```
course_info_imp.
```

```
as
```

```
cursor cur1 is  申请一个游标，用于定位。
```

从辅助表中将课程编号和课程名称读出来，保存到自定义的字段 l_course_id 和 l_course_name 中。

```
select kcbm,kcmc from temp_course_info;
```

```
l_course_id varchar2(20);
```

```
l_course_name nvarchar2(50);
```

```
l_count number(4);
```

开始按照读出的记录进行循环

```

begin
open curl;
loop
fetch curl into l_course_id,l_course_name;
exit when curl%notfound;
判断此课程编号是否已经存在, 不存在则插入信息, 存在则更新课程名称。
select count(course_id)into l_count from course_
info where course_id=l_course_id;
if l_count=0 then
insert into course_info(course_id,course_name)
values (l_course_id,l_course_name);
else
update course_info set course_name=l_course_name
where course_id = l_course_id;
end if;
end loop;
close curl;
end ;

```

3.6 作业的设置

完成所有涉及数据同步的存储过程和调用后, 需要设定同步的时间和先后顺序, 保证同步的正常进行。第一步: 在中转服务器上的数据库中设置作业。将每个调度按照执行的先后次序加入调度中, 在作业中设置执行时间, 由于学期初期, 修改频繁, 所以将执行时间设置为每天凌晨 0 点开始。

第二步: 在 ORACLE 数据库中设置定时调度。

首先添加一个存储过程, 用于调用所有导入数据的存储过程 IMPORT。

as

```

begin
course_info_imp;
.....
end;
再建立一个定时自动调用的存储过程, 设置在
ORACLE 上的同步过程在每天凌晨 2 点开始执行
AS JOB
BINARY_INTEGER;
BEGIN
DBMS_JOB.SUBMIT(JOB,'IMPORT;',sysdate,'trunc
c(SYSDATE)+1/24+2',FALSE);
commit;
end;

```

4 项目方案实施效果

同步的过程都是在数据库中执行, 因此执行的效率很高, 特别是大大提高了筛选数据的效率。

通过本次数据同步整合, 实现了教务和学习平台的基础数据的共享和同步更新的功能, 成功实现了教务管理平台和平台的数据对接, 大大减轻了学习平台基础数据建立和维护的工作量。方便学校对教学信息的管理。实现了两个系统的协同工作。

参考文献

- 1 oracle sql 性能优化.http://www.orasos.com.
- 2 王海亮.精通 Oracle.10g.PL,SQL 编程.北京:中国水利水电出版社,2004.
- 3 Kyte T.Oracle 专家高级编程.北京:清华大学出版社,2002.
- 4 邹建.中文版 SQL Server2000 开发与管理应用实例.北京:人民邮电出版社,2005.