外连接在 PL/SQL 数据迁移程序中的应用

Application of Outer Join in Data Migration

宋　鹏[1]　赵球红[2]　(1. 西安工程大学电子信息学院 陕西 西安 710048;
2. 西安电子科技大学软件学院 陕西 西安 710071)

摘　要: 针对数据迁移中 PL/SQL 程序要求具有较高的执行速度,分析了迁移程序中游标的查询语句,提出应用外连接技术优化查询语句能减少游标的嵌套层次,达到提高数据迁移程序执行速度的目的。数据迁移实践证明此方法能显著提高海量数据迁移程序的执行速度。

关键词: 数据迁移 游标 PL/SQL

伴随着企业管理系统的更新换代,旧系统中积累的大量历史数据必须迁移到新系统中去,这些历史数据都是新系统正常运行所必须的[1-2]。数据迁移,就是将这些历史数据进行清洗、转换,并装载到新系统中的过程。简单的数据迁移任务,可以使用专业数据迁移工具,例如 Ascential Software 公司的 DataStage,迁移速度很快,但是购买费用较高。复杂的数据迁移任务,一般需要用自主开发的数据迁移程序进行数据迁移。本文以 PL/SQL 程序作为 Oracle 数据库之间数据迁移的工具,论述了外连接在迁移程序中的应用,达到了提高数据迁移速度的目的。

1　编写 PL/SQL 程序进行数据迁移

对于 Oracle 数据库之间的数据迁移任务,可以通过编写 PL/SQL 程序实现,因为 PL/SQL 语言是 Oracle 数据库的专用语言,数据变换处理比较方便,程序的执行速度较快。目标数据库中每一个表对应一个程序包(package),包中包含一个或几个过程和函数,具体处理步骤如图 1 所示。每一个程序包中都包括完整的 ETL(Extract – Transform – Load,抽取、转换、装载)过程,其中数据的抽取在游标的定义中实现,数据的变换在主过程中实现,变换完数据的插入由插入过程实现,独立重复的处理功能由函数实现。

数据的抽取在游标的定义中实现,根据变换要求,可以从源数据库的多个表中抽取数据,游标的定义需要精心设计,好的游标定义就代表好的数据抽取方案,

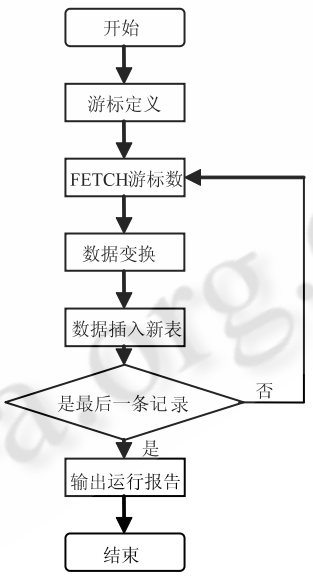

图 1　PL/SQL 程序数据迁移执行流程

会对下面的数据变换处理带来较大的便利,并会大幅度的提高 PL/SQL 程序的执行速率。数据迁移实践表明,通过仔细研究数据变换的业务要求,在游标的定义中增加合适的外连接可以大幅度的提高数据迁移程序的执行速度。

数据的变换在主过程中实现,FETCH 游标得到一组数据,根据新旧数据对应关系对该组数据进行变换,得到目标数据库中表的一条记录,把该条记录插入目标数据库的表中,下来判断该条记录是否为该表的最后一条记录,若是,则输出运行报告后结束程序。若不

是则继续 FETCH 游标数据,一直循环下去,直到处理完该表的最后一条记录,输出运行报告,结束程序。

2 外连接技术

数据查询中外连接可以显示某个表的所有记录,包括不符合约束条件的记录。SQL 语言提供了两个关键字:LEFT OUTER JOIN 与 RIGHT OUTER JOIN。LEFT OUTER JOIN 关键字使外连接显示 LEFT OUTER JOIN 左侧的表包含的所有记录。而 RIGHT OUTER JOIN 关键字则使外连接显示其右边的表所包含的所有记录[3-5]。为了方便举例说明,下面给出 Employee 表与 Project 表的定义:

```
CREATE TABLE Employee ( ---员工信息表
id varchar2(5) not null, ---员工编号
name varchar2(10) not null, ---员工姓名
pno varchar2(10), ---员工参与项目的编号
PRIMARY KEY (id));
CREATE TABLE Project ( ---项目信息表
pnumber varchar2(10) not null, ---项目编号
pname varchar2(20) not null, --- 项目名称
PRIMARY KEY (pnumber));
```

2.1 LEFT OUTER JOIN

要显示 FROM 子句中先命名的表的所有记录,可以使用 LEFT OUTER JOIN。例如,要显示员工的信息及其参与项目的信息,可以对 Employee 表与 Project 表进行连接。同时,用户可能希望显示没有参与项目的员工的信息。这时就可以使用外连接进行查询,使用 LEFT OUTER JOIN 方法如下:

```
SELECT name,id,pnumber,pname
FROM Employee LEFT OUTER JOIN Project
ON pno = pnumber
```

LEFT OUTER JOIN 告诉系统,无论 Employee 表中的记录是否有一个可以与 Project 表中记录连接的值,都将在结果中包含 Employee 表中的所有记录,对于没有连接值的记录其对应的 Project 表中的列显示为空值。

在不同的数据库产品中,外连接的实现可能会有所不同。例如在 ORACLE 数据库中在不需要显示所有记录的表的后面使用一个加号来实现。

上面的书写方式可以写为:

```
SELECT name,id,pnumber,pname
FROM Employee,Project
WHERE pno = pnumber(+);
```

2.2 RIGHT OUTER JOIN

RIGHT OUTER JOIN 与 LEFT OUTER JOIN 的作用正好相反,将显示关键字右边的表包含的所有记录。例如:

```
SELECT name,id,pnumber,pname
FROM Employee RIGHT OUTER JOIN Project
ON pno = pnumber
```

这个查询将显示所有 Project 表中的记录。RIGHT OUTER JOIN 告诉系统,无论 Project 表中的记录是否有一个可以与 Employee 表中记录连接的值,都将在结果中包含 Project 表中的所有记录,对于没有连接值的记录其对应的 Employee 表中的列显示为空值。

3 利用外连接改进游标的定义

为了举例说明的方便,下面给出 N01R (A 表),N01R0(B 表),KUSNKNR(C 表)3 个表的定义:

```
CREATE TABLE N01R (
KG_CD varchar2(20) not null, --- A 表的企业编号
TIKJGYCD varchar2(20) not null, ---A 表的企业识别编号
PRIMARY KEY (KG_CD));
CREATE TABLE N01R0 (
KG_CD varchar2(20) not null, --- B 表的企业编号
TIKJGYCD varchar2(20) not null, ---B 表的企业识别编号
PRIMARY KEY (KG_CD));
CREATE TABLE KUSNKNR (
KGCD varchar2(20) not null, --- C 表的企业编号
KUSNKNRNO varchar2(20) not null, ---C 表的企业更新管理编号
PRIMARY KEY (KGCD));
```

某个数据迁移程序中定义了下面两个游标:

```
CURSOR1
SELECT A. KG_CD KG_CD
FROM N01R A, N01R0 B
WHERE A. KG_CD = B. KG_CD
```

```
AND A. TIKJGYCD = B. TIKJGYCD
AND A. TIKJGYCD IN ('98','58');
CURSOR2
SELECT C. KUSNKNRNO
FROM KUSNKNR C
WHERE C. KGCD = CURSOR1. KG_CD;
```

游标 1 的功能是抽取 A 表的企业编号(KG_CD),条件为:A 表的企业编号等于 B 表的企业编号、A 表的识别编号等于 B 表的识别编号、A 表的识别编号为 98 或 58。游标 2 的功能是抽取 C 表的更新管理编号(KUSNKNRNO),条件为:C 表的企业编号等于游标 1 的企业编号。数据处理过程为:游标 1 和游标 2 依次打开,当 C 表中的企业编号在游标 1 中存在,那么输出 C 表的更新管理编号并做进一步的变换处理,若不存在,输出游标 1 的企业编号并做进一步的处理。

这种办法数据抽出后,在变换部分容易实现。但因为两个游标嵌套,外层游标每取得一条纪录,内层游标都要对 C 表从头到尾查询一遍,这种重复查询取决于外层游标所取得的纪录条数,所以在大量数据处理时,程序效率很低,不能满足客户的需求。

在满足业务要求的前提下,利用外连接把两个游标合并成一个游标如下:

```
CURSOR3
SELECT A. KG_CD KG_CD,
C. KGCD KGCD,
C. KUSNKNRNO KUSNKNRNO
FROM N01R A, N01R0 B, KUSNKNR C
WHERE A. KG_CD = B. KG_CD
AND A. TIKJGYCD = B. TIKJGYCD
AND B. KG_CD = C. KGCD( + )
AND A. TIKJGYCD IN ('98','58');
```

游标 3 的功能是对满足条件的 A 表中的企业编号全部输出,如 C 表中没有对应的企业编号,则 C 表的企业编号和 C 表更新管理编号项目以空值输出。

对游标 3 抽取的数据进行变换处理,可以完成游标 1 和游标 2 嵌套循环所完成的同样的数据迁移任务。不同的是:应用两个游标嵌套循环抽取数据,后续数据变换处理简单,结构清晰,但程序执行速度较慢;应用一个游标抽取数据,后续数据变换处理复杂,但程序执行速度较快。外连接的条件书写比较难于理解,但是,为了满足客户在速度上的需求,必须对业务需求进行深入理解后,设计单一的游标来完成。

4 不同查询方法的测试比较

为了比较两种查询方法对数据迁移程序执行速度的影响,做了以下测试:N01R 表、N01R0 表中各有 5 万条数据,KUSNKNR 表中有 4.6 万条数据,应用 Oracle 数据库的执行命令 set timing on,在 SQL * Plus 运行环境中运行,可以使执行的 procedure 所用的时间表示出来,结果如图 2 所示。

图中 zhao_test_ikou 是两个游标的查询过程,zhao_test_hebing 是合并后的一个游标的查询过程,其中两个游标用时 2.03 秒,一个游标用时 0.08 秒。从测试结果中明显可以看出,在对同样的 5 万条记录只做查询,执行改进了游标的 procedure 所花费的时间明显减少。得出结论:采用减少游标的设计方案能够明显提高数据迁移程序的执行速度。

图 2　数据迁移不同查询方法所用时间的比较

5 结束语

本文首先介绍了数据迁移的基本概念,给出了编写 PL/SQL 程序进行数据迁移的流程。针对如何提高数据迁移程序执行速度的问题,提出应用外连接技术改进游标定义中的查询语句的解决方案,进一步给出了改进实例并进行了测试。需要注意的是,除了外连接技术外,给源数据库中的表加合适的索引也可以提高数据迁移程序的执行速度。(下转第 127 页)

(上接第 123 页)

参考文献

1 杜娟,何正国. 从 SQL SERVER2000 向 ORACLE 9i 迁移的技术实现方案. 中国水运(理论版),2006(8):104 – 105.

2 李永良. 数据迁移在新旧系统中切换. 中国计算机用户,2003,35:45 – 46.

3 连育英. SQL 多表外连接查询. 科技情报开发与经济, 2005,(6): 240 – 242.

4 王平勤, 董付国, 周翔凤. SQL 外连接查询在系统开发中的应用. 电脑开发与应用, 2008,(3): 57 – 58.

5 史嘉为. SQL 语言中连接查询和嵌套查询. 电脑知识与技术, 2003,29: 23 – 24.