

PowerBuilder 在数据库开发中建立多通信数据区

王竹云 (浙江省计算技术研究所 310006)

摘要:本文对怎样处理 PowerBuilder 在 SYBASE 数据库开发应用中多个通信数据工作区建立问题作简要的阐述,并给出了源程序。

美国著名的数据库应用开发工具生产厂家 PowerSoft 公司推出的功能强、性能优异的开发工具 PowerBuilder,是一种完全按照客户/服务器(Client/Server)体系结构研制的开发系统,是一种面向对象的数据库应用开发工具。

PowerBuilder 通过对不同数据库采用不同接口的形式,可同时支持多种关系数据库。

PowerBuilder 自身随安装生成的 WATCOM 数据库,同 FoxPro 一样,一个通信数据工作区可以反复地使用,即在该工作区中进行新的表操作,原先在该工作区的表操作自动失效。而 SYBASE 数据库的通信工作区是独占的,即当某表使用某一工作区后,在该表的操作未执行提交语句 COMMIT 或回退语句 ROLLBACK 前,该工作区一直被该表的操作独占,尚若该通信数据工作区未释放前又有其他表使用该工作区,则将导致程序死循环错误,并且不支持 WATCOM 所支持的当前光标操作。而我们在单机上或在初期开发应用时(模拟调试阶段)往往只顾及开发工具本身的基本功能对应用程序的实现,未能过多地考虑使用服务器上的 SYBASE 数据库将会存在着什么样的问题,而当单机上开发的应用要同服务器上的 SYBASE 联网或在初期开发的应用程序正式在网上运行时,结果发现在 WATCOM 数据库上运行通过的应用程序现在在网上还不能完全使用起来。

下面是 PowerBuilder 4.0 在 WATCOM 数据库中实现的一个统计报表统计的一部分,其主要内容包括:首先删除临时统计表 f0100201 中所有记录;根据用户输入的用户编号的范围截取前 4 位(小区编号)插入到临时表 f0100201 中的相应字段;定位查找小区编号与小区名称对照表 f01001a,将找到的小区名称填入到当前光标指定的记录对应字段中。

当在 SYBASE 数据库中已建立了与 WATCOM 数据库中同样的表结构及输入有关表的数据,将库连接到

SYBASE 数据库,运行单机上开发的应用程序时,系统将两处出现错误:其一,在光标名为 C₃ 取小区名称后(fetch c3 into :f01001a01, :f01001a03;),进行状态代码判断处(do while sqlca.sglcode = 0)死循环;其二,在 update f0100201 set f010020102 = :f010020102 where current of c2; //修改小区名处显示系统两行警告性提示:

```
[0052]:Database Warning C0054:Database Command not supported for this DBMS
```

```
[0052]:Database Warning C0038:incorrect syntax near 'of'
```

即对于第 52 行,数据库操作命令,当前光标定位该数据库管理系统不支持。如何避免上述错误,其方法是通过建立多个通信数据工作区来完成:

1. 在全局变量(Global Variables)中定义两个事务对象变量 transaction sqlca1 和 transaction sqlca2。

2. 在库应用程序中增加两个通信工作区,即创建二个新的事务对象,我们创建 SQLCA1 = CREATE TRANSACTION 和 SQLCA2 = CREATE TRANSACTION 事务对象,与原先缺省的事务对象 SQLCA 共计三个事务对象构成。并与数据库建立连接关系,connect using sqlca1 和 connect using sqlca2,详见库应用程序清单。

3. 并对有关通信工作区进行适当的分配使用,(例如,缺省的工作区 sqlca 及创建的 sqlca1、sqlca2。)同时将当前光标定位语句修改为使用通信工作区及加上修改操作的定位条件:

```
update f0100201 set f010020102 = :f010020102
where current of c2; //修改小区名
```

改为:

```
update f0100201 set f010020102 = :f010020102
where f010020101 = :f010020101 using
sqlca2; //修改小区名
```

详见修改后应用程序清单。

```
//库应用程序清单
//This script will read all the database values from PB.INI
// and store them in SQLCA.
SQLCA.DBMS = ProfileString("PB.INI","Database","DBMS"," ")
SQLCA.Database = ProfileString("PB.INI","Database","DataBase"," ")
SQLCA.LogID = ProfileString("PB.INI","Database","LogID"," ")
SQLCA.LogPass = ProfileString("PB.INI","Database","LogPassword"," ")
SQLCA.ServerName = ProfileString("PB.INI","Database","ServerName"," ")
SQLCA.UserID = ProfileString("PB.INI","Database","UserID"," ")
SQLCA.DBPassword = ProfileString("PB.INI","Database","DatabasePassword"," ")
SQLCA.Lock = ProfileString("PB.INI","Database","Lock"," ")
SQLCA.DbParm = ProfileString("PB.INI","Database","DbParm"," ")
```

```
SQLCA1 = CREATE TRANSACTION
SQLCA1.DBMS = ProfileString("PB.INI","Database","DBMS"," ")
SQLCA1.Database = ProfileString("PB.INI","Database","DataBase"," ")
SQLCA1.LogID = ProfileString("PB.INI","Database","LogID"," ")
SQLCA1.LogPass = ProfileString("PB.INI","Database","LogPassword"," ")
SQLCA1.ServerName = ProfileString("PB.INI","Database","ServerName"," ")
SQLCA1.UserID = ProfileString("PB.INI","Database","UserID"," ")
SQLCA1.DBPassword = ProfileString("PB.INI","Database","DatabasePassword"," ")
SQLCA1.Lock = ProfileString("PB.INI","Database","Lock"," ")
SQLCA1.DbParm = ProfileString("PB.INI","Database","DbParm"," ")
```

```
SQLCA2 = CREATE TRANSACTION
SQLCA2.DBMS = ProfileString("PB.INI","Database","DBMS"," ")
SQLCA2.Database = ProfileString("PB.INI","Database","DataBase"," ")
SQLCA2.LogID = ProfileString("PB.INI","Database","LogID"," ")
SQLCA2.LogPass = ProfileString("PB.INI","Database","LogPassword"," ")
SQLCA2.ServerName = ProfileString("PB.INI","Database","ServerName"," ")
SQLCA2.UserID = ProfileString("PB.INI","Database","UserID"," ")
SQLCA2.DBPassword = ProfileString("PB.INI","Database","DatabasePassword"," ")
SQLCA2.Lock = ProfileString("PB.INI","Database","Lock"," ")
SQLCA2.DbParm = ProfileString("PB.INI","Database","DbParm"," ")
```

```
connect using sqlca;
connect using sqlca1;
connect using sqlca2;
open(face1-win)
```

```
//修改后应用程序清单
delete from f0100201; // 删除临时表
if sqlca.sqlcode = 0 then
    commit using sqlca;
else
    rollback using sqlca;
end if
char f0100201[10]
char xqbh[4]
char f010020101[4]
char yhbh1[10]
```

```
char yhbh2[10]
yhbh1 = sle-3.text // 获取起始用户编号
yhbh2 = sle-4.text // 获取终止用户编号
declare c1 cursor for
select f0100201 from f01002 where f0100201 > =
    :yhbh1 and f0100201 < = :yhbh2 using sqlca;
open c1;
fetch c1 into :f0100201;
do while sqlca.sqlcode = 0
    xqbh = f0100201
    insert into f0100201(f010020101)
    values(:xqbh) using sqlca1; // 插入小区编号
    if sqlca1.sqlcode = 0 then
        commit using sqlca1;
    else
        rollback using sqlca1;
    end if
    fetch c1 into :f0100201;
loop
close c1;
commit using sqlca;
```

```
char f010020101[4]
char f010020102[16]
declare c2 cursor for
select f010020101 from f0100201 using sqlca;
open c2;
fetch c2 into :f010020101;
do while sqlca.sqlcode = 0
    char f01001a01[4]
    char f01001a03[16]
    declare c3 cursor for
    select f01001a01, f01001a03 from f01001a
    where f01001a01 = :f010020101 using sqlca1;
    open c3;
    fetch c3 into :f01001a01, :f01001a03; // 获取小区名称
    do while sqlca1.sqlcode = 0
        if left(f01001a01, 4) = left(f010020101, 4) then
            f010020102 = f01001a03
        end if
        fetch c3 into :f01001a01, :f01001a03;
    loop
    close c3;
    update f0100201 set f010020102 = :f010020102
    where f010020101 = :f010020101 using sqlca2;
    // 修改小区名
    if sqlca2.sqlcode = 0 then
        commit using sqlca2;
    else
        rollback using sqlca2;
    end if
    fetch c2 into :f010020101;
loop
close c2;
messagebox("提示信息", "小区名称已替换完成");
commit using sqlca;
commit using sqlca1;
```

(来稿时间:1996年5月)