## 原理说明

运用ora\_load.sh导出原数据库的表的数据库，然后在目标数据库建好表，再把导出的数据ftp到目标数据库库，执行batch\_load\_data.sh导入目标数据库。导出采用spool导出，然后处理导出数据，以便可以用sqlldr执行导入。

## 脚本代码

### #ora\_load.sh

#!/bin/ksh

USAGE="`basename $0` userid filename sql"

##############################################################

#######Determine whether the correct number of parameters#####

##############################################################

if [ $# -ne 3 ];then

echo $USAGE

exit 1;

fi

##############################################################

################Get the parameter valuess#####################

##############################################################

userid=$1

filename=$2

sql=$3

##############################################################

####################Clear temporary files#####################

##############################################################

Clear\_data()

{

rm -rf sql.temp $filename.temp1 $filename.temp;

}

##############################################################

#####################use spoll export data####################

##############################################################

echo "set colsep |;

set echo off;

set heading off;

set feedback off;

set linesize 9999;

set pagesize 0;

set termout off;

spool $filename.temp;

$sql;

spool off;">sql.temp

echo "@sql.temp"|sqlplus -S $userid

##############################################################

#####################Process and export data##################

##############################################################

cat $filename.temp |sed 's/ //g'|sed 's/^ \*//g'|sed 's/| \*/|/g'|sed 's/\*|/|/g'>$filename.temp1

offlineset=`cat $filename.temp1|sed -n 1p|grep -o '.$'|awk '{print $1}'`

if [ "${offlineset}" != "|" ];then

cat $filename.temp1 |sed 's/.$/&|/g'>$filename.unl

else

cat $filename.temp1>$filename.unl

fi

##############################################################

###########check whether the export was successful############

##############################################################

numrows=`cat $filename.unl|wc -l |awk '{print $1}'`

if [ $numrows -gt 0 ];then

echo "Load $numrows record succeed!"

Clear\_data

else

echo "Load Error!"

Clear\_data

fi

### #batch\_load\_data.sh

#!/bin/ksh

USAGE="`basename $0` user/password tablename filename sign[y/n:y=delete]"

##############################################################

#######Determine whether the correct number of parameters#####

##############################################################

if [ $# -ne 4 ];then

echo $USAGE

exit 1;

fi

##############################################################

################Get the parameter valuess#####################

##############################################################

userid=$1

tablename=$2

filename=$3

sign=$4

##############################################################

#####################Get table structure######################

##############################################################

echo "spool $tablename.temp;

desc $tablename;

spool off; "|sqlplus -S $userid >>/dev/null

##############################################################

##############Generate control files into tables##############

##############################################################

total\_line=`wc -l $tablename.temp|awk '{print $1}'`

end\_line=`echo "$total\_line-1"|bc`

cat $tablename.temp|

sed -n "3,$end\_line p"|

awk '{print $1,$NF}'|

awk 'BEGIN{

print "load data"

print "append"

print "into table '$tablename'"

print "fields terminated by '\''|'\''"

print "("

}

{

if($2~/DATE/)

print $1" date '\''yyyy-mm-dd HH24:mi:ss'\''"","

else

print $1 " CHAR " ","

} '|

sed '$s/,//'|sed '$a )'>>${tablename}.ctl

##############################################################

####################Clear temporary files#####################

##############################################################

clear\_temp\_file()

{

rm -rf ${tablename}.ctl ${tablename}.temp

}

##############################################################

###################delete import table data###################

##############################################################

delete\_table\_data()

{

echo "truncate table ${tablename};"|sqlplus -S $userid >>/dev/null

}

##############################################################

#########Determine whether the import was successful##########

##############################################################

check\_result()

{

total\_database=`cat ${tablename}.log |grep -i "successfully"|awk '{print $1}'|awk '{print $1}'`

total\_datafile=`wc -l ${filename}|awk '{print $1}'|awk '{print $1}'`

if [ ${total\_database} -eq ${total\_datafile} ];then

echo "Load ${tablename} record ${total\_datafile} succeed!"

rm -rf ${tablename}.log

else

echo "Load ${tablename} failed,please check ${tablename}.log!"

fi

}

if [ ${sign} = "y" -o ${sign} = "Y" ];then

delete\_table\_data

else

if [ ${sign} != "y" -o ${sign} != "Y" -o ${sign} = "n" -o ${sign} = "N" ];then

echo $USAGE

clear\_temp\_file

exit 1

fi

fi

##############################################################

#####################use sqlldr import data###################

##############################################################

#sqlldr $userid control=$tablename.ctl data=$filename direct=true parallel=true

sqlldr $userid control=$tablename.ctl data=$filename parallel=true >>/dev/null

clear\_temp\_file

check\_result

## 示例

### 导出

#./ora\_load.sh test/test123 test "select \* from test"

**参数说明**：

(1)test/test123为用户名/密码

(2)test为导出数据存储的文件名，导出后的文件就是test.unl

(3)"select \* from test"为sql语句，这个语句表导出test表的所有内容，sql语句可以加where条件

### 导入

#./batch\_load.sh test/test123 test test.unl y

**参数说明:**

(1)test/test123为用户名/密码

(2)test为导入的表名

(3)test.unl为sqlldr导入时候的数据文件

(4)y代表导入时先删除原来表的数据，如果为n或者N则不删除原表数据