Oracle 数据库字符集问题分析及解决方法

The Analyse and Solution on CharacterSets Problem in Oracle Database

戴常英　张孝天　（中国石油大学 计算机与通讯工程学院 东营 257061）

摘要：在分析了数据迁移过程中的字符集变换、出现字符乱码现象的原因的基础上，介绍了如何查询数据库及客户端的字符集。并对数据迁移中可能出现的问题，给出具体的解决方法。

关键词：Oracle 数据库 字符集 迁移

Oracle 数据库是目前比较流行的数据库平台之一，但是 Oracle 数据库在实际应用过程中，常常会因字符集问题而导致数据库内汉字信息变成乱码或在数据迁移时导致数据损失或迁移失败，给用户带来很多不便。本文就该问题及其解决方法进行了详细的介绍。

1　Oracle 数据库的字符集问题

Oracle 数据库为适应不同语言文字的显示而设定了字符集。字符集不仅在服务器端存在，而且客户端也必须注册字符集。要在客户端正确显示 Oracle 数据库汉字信息，必须使服务器端的字符集与客户端的字符集一致；另外字符集不一致，在进行数据迁移操作时，就会进行字符集的转换，可能会造成不必要的数据损失，导致数据库的数据导出、导入失败。

Oracle 的导入导出（Import/export）工具是我们常用的一个数据迁移及转化工具，因其导出文件具有平台无关性，所以在跨平台迁移中，最为常用。在数据导出导入过程中主要涉及以下字符集：

（1）源数据库字符集；

（2）Export 过程中用户会话字符集；

（3）Import 过程中用户会话字符集；

（4）目标数据库字符集。

从示意图中可以看出，在数据迁移时候有四处涉及到字符集问题，而这四处字符集的不一致恰恰是导致 Oracle 进行字符集转换的原因。在 Export 过程中，如果字符集（1）和（2）不同会发生字符集转换，（1）的 ID 号将会在导出的二进制格式 dmp 文件的头部几个字节中存储，当文件导入时，将会检查 dmp 文件使用的字符集设置，如果（2）不同于（3）设置，字符集将根据（3）的设置进行转换，如果必要，在数据插入数据库之前会进行（3）到（4）的转换。

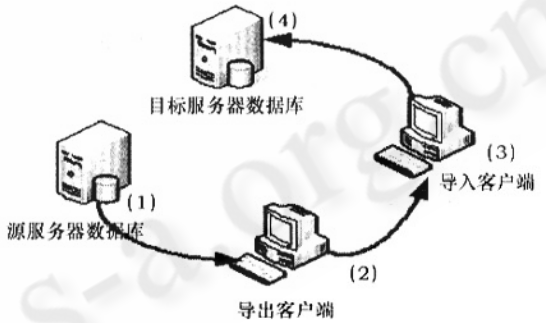

图 1　数据迁移示意图

通常情况下，在数据导出时，最好把客户端字符集设置和数据库服务器端相同，这样可以避免在导出时发生不必要的数据转换，如字符集不一致，进行字符集转换时，可能会导致迁移失败或造成数据的损失。

影响 Oracle 数据库字符集最重要的参数是 NLS_LANG 参数。它的格式如下[1]：

NLS_LANG = < Language > _ < Territory > . < Clients Characterset >，其中：LANGUAGE 指定 Oracle 消息使用的语言和日期中月份和日显示；TERRITORY 指定货币和数字格式、地区和计算星期及日期的习惯；CHAR-

ACTERSET 指定客户端应用程序使用的字符集。因此，真正影响数据库字符集的是第三部分。

2 查询 Oracle 数据库的字符集参数

2.1 查询 Oracle server 端的字符集

数据库服务器字符集：select * from nls_database_parameters，其来源于 props $，是表示数据库的字符集。

2.2 查询 Oracle clicte 端的字符集

客户端字符集：select * from nls_instance_parameters，其来源于 v $ parameter，表示客户端的字符集的设置，可能是参数文件，环境变量或者是注册表。在 WINDOWS 平台下，可以打开注册表，在"HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE/HOME0"中查询环境变量 NLS_LANG 的值。

2.3 查询会话字符集

会话字符集环境 select * from nls_session_parameters，其来源于 v $ nls_parameters，表示会话自己的设置，可以是会话的环境变量或者是 alter session 完成，如果会话没有特殊的设置，将与客户端的字符集相一致。

客户端的字符集要求与服务器一致，才能正确显示数据库的非 Ascii 字符。如果多个设置存在的时候，alter session > 环境变量 > 注册表 > 参数文件字符集要求一致，但是语言设置却可以不同，语言设置建议用英文。如字符集是 ZHS16GBK，则 NLS_LANG 可以是 AMERICAN_AMERICA. ZHS16GBK。

3 Oracle 数据库字符集问题的解决方法

Oracle 数据库的字符集有互相的包容关系，如 US7ASCII 就是 ZHSI6CBK 的子集，从 US7ASCII 到 ZHSI6CBK 不会有数据解释上的问题，也不会有数据丢失。在所有的字符集中 UTF8 应该是最大，因为它基于 UNICODE，双字节保存字符（也因此在存储空间上占用更多）。根据 Oracle 数据库的官方说明，字符集的转换从子集到超集受支持，反之不行。如果两种字符集之间根本没有子集和超集的关系，那么字符集的转换是不受 Oracle 数据库支持的。对数据库 server 端错误的修改字符集将会导致很多不可测的后果，可能会严重影响数据库的正常运行，所以在修改之前一定要确认两种字符集之间是否存在子集和超集的关系。特别说明，一般最常用的两种字符集 ZHS16GBK 和 ZHS16CGB231280 之间不存在子集和超集关系，因此理论上讲这两种字符集之间的相互转换是不受支持的。

如果在数据迁移过程中涉及到的字符集都一致，就不会出现字符集转换过程，如果不一致，只有当子集向集级转换时候才能保证字符集的正常转换。下面介绍几种在数据迁移过程中，经常会遇到的主要问题及解决办法。

3.1 服务器指定字符集与客户端字符集不同，而与加载数据字符集一致。

Oracle 数据库字符集通常是在创建时确定，一旦存储用户数据后就不要再修改了，因为其数据都是使用该字符集进行存储的，改换其他字符集之后，原有数据就不能够正确表示了。

因此通过查看服务器端字符集，根据服务器端的 Oracle 数据库字符集对客户端进行设置。配置方法有两种：其一是在安装 Oracle 的客户端软件时指定。另外一种是通过更改客户端的操作系统的注册表，在 WINDOWS 平台下，可以打开注册表，通过修改"HKEY_LOCAL_MACHINE \SOFTWARE \ORACLE \HOME0"中环境变量 NLS_LANG 的值，来实现客户端与服务器字符集的统一。

3.2 服务器指定字符集与客户端字符集相同，与加载数据字符集不一致

问题一般发生在 Oracle 版本升级或重新安装系统时，选择了与原来服务器端不同的字符集，而恢复加载的备份数据仍是按原字符集卸出的场合，以及加载从其它使用不同字符集的 Oracle 数据库卸出的数据的情况。这两种情况中，不管服务器端和客户端字符集是否一致都无法显示汉字。

这种情况下面提供四种解决方法：

（1）服务器端重新安装 Oracle

在重新安装 Oracle 时选择与原卸出数据一致的字符集，加载原卸出的数据。这种情况仅仅使用于空库和具有同一种字符集的数据。

(2) 强行修改服务器端 Oracle 当前字符集(以 ZHS16GBK 为例)

在用 imp 命令加载数据前,先对当前 Oracle 数据库字符集修改,使得服务器的字符集与加载的数据字符集一致。这是最简单的转换字符集的方式,但并不总是有效,只能在新字符集是旧字符集严格超集的情况下使用这种方式转换。

```
C:/ >SQLPLUS/NOLOG
SQL >CONN AS SYSDBA;
SQL > ALTER SYSTEM ENABLE RESTRICTED SESSION;
SQL > ALTER SYSTEM SET JOB_QUEUE
_PROCESSES =0;
SQL > ALTER SYSTEM SET AQ_TM_PROCESSES =0;
SQL > ALTER DATABASE OPEN;
SQL > ALTER DATABASE CHARACTER SET
ZHS16GBK;
```

修改后的情况同 3.1 情况相同。

(3) 修改 dmp 文件字符集[2]

导出文件(dmp 文件)的第 2、3 字节记录了字符集信息,因此直接修改 dmp 文件的第 2、3 字节的内容。例如服务器的字符集为 ZHS16GBK。

使用 2 进制文件编辑工具,如 uedit32。打开导出的 dmp 文件,获取 2、3 字节的内容,如 00 01,先把它转换为 10 进制数为 1,使用函数 NLS_CHARSET_NAME 即可获得该字符集:

```
SQL > select nls_charset_name(1) from dual;
NLS_CHARSET_NAME(1)
- - - - - - - - - - - -
US7ASCII
```

可以知道该 dmp 文件的字符集为 US7ASCII。下一步是把该 dmp 文件的字符集换成 ZHS16GBK,则需要用 NLS_CHARSET_ID 获取该字符集的编号:

```
SQL > select nls_charset_id('zhs16gbk') from du-
al;
NLS_CHARSET_ID('ZHS16GBK')
- - - - - - - - - - - - - - -
852
```

把 852 换成 16 进制数为 354,把 2、3 字节的 00 01 换成 03 54,即完成了把该 dmp 文件字符集从 US7ASCII 到 ZHS16GBK 的转化,这样,再把该 dmp 文件导入到 ZHS16GBK 字符集的数据库就可以了。

(4) 通过数据转换,绕过字符集

将数据加载到具有相同字符集的服务器上,然后用转换工具卸出为 access 或别的数据库,再用转换工具转入到目的数据库中,这样就避免了 Oracle 字符集的困扰。

3.3 服务器指定字符集与客户字符集不同,与输入数据字符集不一致

这种情况可通过修改客户端字符集,使客户端与服务器端字符集一致后,再按照 3.2 情况处理。

必须指出的是进行字符集转换时候应该非常谨慎,在进行任何可能对数据库结构发生改变的操作之前,先做有效的备份,防止在字符集转换过程中带来的损失。

4 结论

要正确理解 Oracle 字符集的转换过程的内在机理,可以使我们在实际工作中,避免不必要的麻烦和数据损失。针对在实际工作中的具体字符集合问题,本文提出几种解决方法,具有一定的参考价值和借鉴意义。

参考文献

1 Loney K. Theriau M Oracle9i DBA 手册[M], 机械工业出版社 ,2002。

2 冯春培 、盖国强等, Oracle 数据库 DBA 专题技术精粹[M] ,冶金工业出版社, 2004。