目录

[1、 常用配置 2](#_Toc514662052)

[1.1、 监听和服务配置 2](#_Toc514662053)

[1.2、 查看和设置Oracle编码 3](#_Toc514662054)

[1.2.1、 字符集简介 3](#_Toc514662055)

[1.2.2、 查看oracle数据库的编码 3](#_Toc514662056)

[1.2.3、 查看oracle客户端编码 3](#_Toc514662057)

[1.2.4、 修改数据库编码 4](#_Toc514662058)

[1.3、 Oracle逻辑备份/还原的三种模式 5](#_Toc514662059)

[1.3.1、 EXP备份 5](#_Toc514662060)

[1.3.2、 IMP还原 5](#_Toc514662061)

[1.4、 Oracle 11g R2中exp不能导出空表的解决方法 6](#_Toc514662062)

[2、 常见命令简介 6](#_Toc514662063)

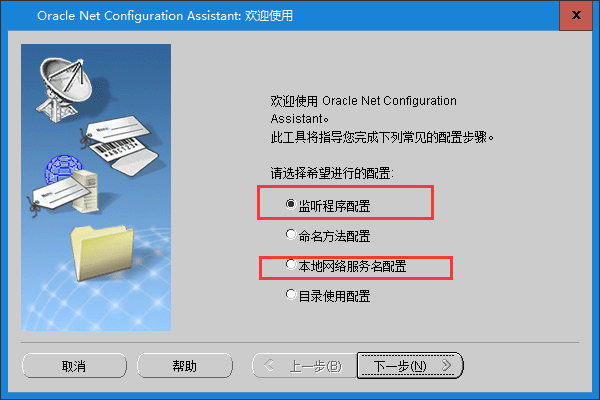
[2.1.1、 startup,startup mount,startup nomount之间的区别 6](#_Toc514662064)

# 常用配置

## 监听和服务配置

1. 先监听程序配置，后本地网络服务名配置





## 查看和设置Oracle编码

### 字符集简介

1. ORACLE数据库有国家字符集（national character set）与数据库字符集(database character set)之分。两者都是在创建数据库时需要设置的。国家字符集主要是用于NCHAR、NVARCHAR、NCLOB类型的字段数据，而数据库字符集使用很广泛，它用于：CHAR、VARCHAR、CLOB、LONG类型的字段数据；
2. ORACLE支持的Unicode字符集有以下几种，下面的列表给出了字符集的名称、对应的数据库版本范围、采用的Unicode的版本。
3. AL24UTFFSS：是ORACLE第一种支持Unicode的字符集，从7.2版本开始使用，但是它支持的Unicode版本为1.1，因此从9i开始就不支持此字符集了。
4. UTF8 : 是ORACLE从ORACLE8开始使用的属于UTF-8编码的字符集，从ORACLE8.0到ORACLE8.16，Unicode版本为2.1，而ORACLE817到10g，采用的Unicode标 准为3.0
5. UTFE ：用于EBCDIC码平台上的数据库Unicode字符集。因此它属于专用系统使用的字符集，其它属性与UTF8基本相同。
6. AL32UTF8 ：是从ORACLE9开始使用的属于UTF-8编码的字符集，与UTF8相比，它采用的Unicode版本更新，在10g版本中使用的是Unicode 4.01标准，而UTF8因 为兼容性的考虑，在10g版本中用的是Unicode 3.0标准。
7. AL16UTF16：是ORACLE第一种采用UTF-16编码方式的字符集，从ORACLE9开始使用，是作为缺省的国家字符集使用，它不能被用作数据库的字符集。这是因为数 据库的字符集决定了SQL与PL/SQL源码的编码方式，对于UTF－16这种使用固定的两个字节来表示英文字母的编码方案来说，确实不适于用作数据库 的字符集，ORACLE目前采用的数据库字符集都是基于ASCII或EBCDID作为子集的编码方案。
8. US7ASCII，表示区域是US，用7个比特位表示一个字符，标准的字符集名称为ASCII。

### 查看oracle数据库的编码

1. 其来源于props$，这是表示数据库的字符集。

|  |
| --- |
| select \* from nls\_database\_parameters where parameter ='NLS\_CHARACTERSET'; |

### 查看oracle客户端编码

1. 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会话字符集环境。

|  |
| --- |
| select \* from nls\_instance\_parameters where parameter='NLS\_LANGUAGE'; |

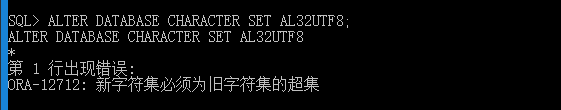
### 修改数据库编码

1. 首先以sysdba的身份登录上去 conn /as sysdba
2. 关闭数据库shutdown immediate;
3. 以mount打来数据库，startup mount
4. 设置session

|  |
| --- |
| ALTER SYSTEM ENABLE RESTRICTED SESSION;  ALTER SYSTEM SET JOB\_QUEUE\_PROCESSES=0;  ALTER SYSTEM SET AQ\_TM\_PROCESSES=0; |

1. 启动数据库 alter database open;
2. 修改字符集，假设修改字符集为ZHS16GBK

|  |
| --- |
| ALTER DATABASE CHARACTER SET ZHS16GBK; |



这会可能会报错，提示我们的字符集：新字符集必须为旧字符集的超集，这时我们可以跳过超集的检查做更改：

|  |
| --- |
| ALTER DATABASE character set INTERNAL\_USE ZHS16GBK; |

这条语句就可以了，TERNAL\_USE提供的帮助就会使oracle绕过了子集与超集的验证，这条语句和上面的语句内部操作时完全相同的。

1. 关闭，重新启动

|  |
| --- |
| shutdown immediate;  startup |

当然字符集最好不要轻易修改，因为这会对数据库的数据有直接的影响，如果是生产环境的话，可能会造成不可估计得损失。

## Oracle逻辑备份/还原的三种模式

### EXP备份

1. 完全(整个数据库)

|  |
| --- |
| #数据库级备份(BPMSVG用户所在数据库，ALIBAO为实例名)  exp BPMSVG/1234@ALIBAO file=C:\Users\walkw\Desktop\oracle\backup\bpmsvg20180419\bpmsvg\_database.dmp full=y |

1. 用户(用户及表)

|  |
| --- |
| #用户级备份(BPMSVG用户及用户的表)  exp BPMSVG/1234@ALIBAO file=C:\Users\walkw\Desktop\oracle\backup\bpmsvg20180419\bpmsvg\_user.dmp owner=(BPMSVG) |

1. 表(指定的表)

|  |
| --- |
| #表备份(例如 table1，table2)  exp BPMSVG/1234@ALIBAO file=C:\Users\walkw\Desktop\oracle\backup\bpmsvg20180419\bpmsvg\_tables.dmp tables=(table1,table2) |

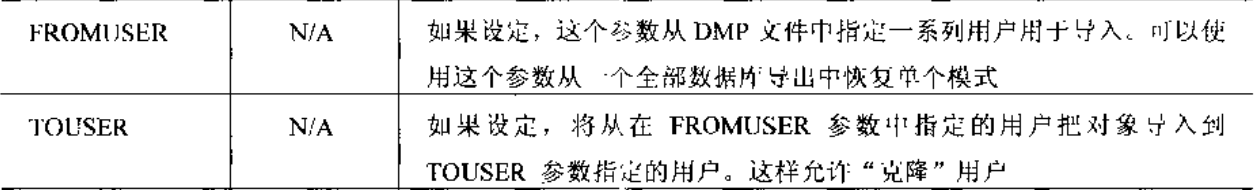
### IMP还原

1. 完全(整个数据库)

|  |
| --- |
| #imp 数据库用户名/密码@数据库名称 file=文件路径加文件名 full=y ignore=y;  imp bpm/Sys123456@127.0.0.1:1521/bpm file=dmp文件路径加文件名full=y ignore=y; |

1. 用户(用户及表)

|  |
| --- |
| #假设导出的用户BPMSVG，导入的用户为BPMVML  IMP BPMVML/1234@ALIBAO FILE=C:\Users\walkw\Desktop\oracle\backup\bpmsvg20180419\bpmsvg\_user.dmp tablespaces=BPM\_UIEX FROMUSER=BPMSVG TOUSER=BPMVML |



1. 表(指定的表)

|  |
| --- |
| #将bpmsvg\_tables.dmp中的表table1，table2 导入  IMP BPMVML/1234@ALIBAO FILE=C:\Users\walkw\Desktop\oracle\backup\bpmsvg20180419\bpmsvg\_tables.dmp tablespaces=BPM\_UIEX tables=(table1,table2) |

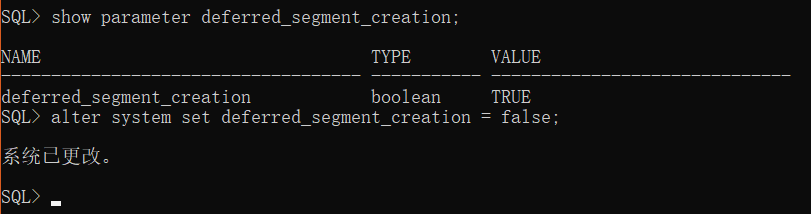
## Oracle 11g R2中exp不能导出空表的解决方法

1. Oracle 11g Realease 2(11.2.0.1)中，在导出空表的时候，会报EXP-00011: XX表不存在的错误。原因是在建表的时候，不会分配存储空间，只有在第一次insert数据的时候，才会分配存储空间。
2. 当导出的空表数量较少的时候，可以写一条数据，然后删除或rollback掉。
3. deferred\_segment\_creation参数用于控制在表结构创建的时候是否延迟创建/分配存储空间，直到第一次插入数据为止。查看该参数的语句为：

|  |
| --- |
| show parameter deferred\_segment\_creation; |

1. 设置在建表的时候自动分配segment，即设置deferred\_segment\_creation为false.

|  |
| --- |
| alter system set deferred\_segment\_creation = false; |



1. 强制给之前的空表分配表空间的方法：alter table <tablename> allocate extent 批量修改如下：

|  |
| --- |
| --查询所有未分配空间的表并拼凑成分配表空间语句  select 'alter table ' ||table\_name || ' allocate extent;' from user\_tables where num\_rows =0;  --导出查询结果，再执行SQL语句 |

# 常见命令简介

### startup,startup mount,startup nomount之间的区别

1. STARTUP NOMOUNT选项：（读初始化参数文件，启动实例）

STARTUP NOMOUNT选项启动实例，但不安装数据库。当数据库以这个模式启动时，参数文件被读取；后台进程和内存结构被启动；但它们不被附加或与数据库的磁盘结构进行通信。当实例处于这个状态时，数据库是不可使用的。

1. STARTUP MOUNT选项：（打开控制文件）

STARTUP MOUNT选项执行STARTUP NOMOUNT选项的所有工作，但另外附加数据库结构并与这些结构进行交互。这时Oracle从它用来查找和附加到主要数据库结构的控制文件中获得信息。当处于这个模式时，可以执行一些管理型任务，比如恢复。

1. STARTUP OPEN选项：（打开数据文件，日志文件）

如果STARTUP命令行上没有指定任何模式，STARTUP OPEN选项就是默认的启动模式。STARTUP OPEN选项执行STARTUP NOMOUNT和STARTUP MOUNT选项的所有步骤。这个选项把数据库变成对所有用户都时可用的。

1. STARTUP FORCE选项：

如果在用正常方式启动数据库时遇到了困难，可以使用STARTUP FORCE选项。STARTUP FORCE选项首先异常关闭数据库，然后重新启动它。

1. STARTUP RESTRICT选项：

STARTUP RESTRICT选项启动数据库并把它置入OPEN模式，但只给拥有RESTRICTED SESSION权限的用户赋予访问权。