利用impdp 和 db link 完成oracle 10g迁移到 oracle 11g

查看源库mysteel用户默认表空间 ：

select USERNAME,DEFAULT_TABLESPACE from dba_users where USERNAME like 'MYSTEEL';

查看源库mysteel schema默认表空间users的大小 ：

select TABLESPACE_NAME,BYTES/1024/1024 from dba_data_files where TABLESPACE_NAME like 'USERS';

查看数据库字符集 ：

select * from nls_database_parameters;

创建好所需表空间MYSTEELNET，建好用户mysteelnet (for 100.130)

create tablespace MYSTEELNET datafile '/oradata/userbehavior/mysteelnet01.dbf' size 10G autoextend on maxsize 32G;

alter tablespace MYSTEELNET add datafile '/oradata/userbehavior/mysteelnet02.dbf' size 10G autoextend on maxsize 32G;

alter tablespace MYSTEELNET add datafile '/oradata/userbehavior/mysteelnet03.dbf' size 10G autoextend on maxsize 32G;

create user mysteelnet identified by asdefg default tablespace mysteelnet quota unlimited on mysteelnet;

SYS@MYSTEEL> select grantee,granted_role from dba_role_privs where grantee='MYSTEEL'

  2 ;

```
GRANTEE                      GRANTED_ROLE
---------------------------- ----------------------------
MYSTEEL                      DBA
SYS@MYSTEEL>
SYS@MYSTEEL> select grantee,privilege from dba_sys_privs where grantee='MYSTEEL';
GRANTEE                      PRIVILEGE
---------------------------- ---------------------------------------
MYSTEEL                      UNLIMITED TABLESPACE
SYS@MYSTEEL>
```

grant dba to mysteelnet;

grant connect, resource to mysteelnet;

grant unlimited tablespace to mysteelnet;

创建好所需表空间MYSTEELXXX，建好用户mysteelxxx (for 100.111)

如果网络可以互通的话，则直接使用network_link=database_link的方式迁移，这同时还减少了数据的落地时间，而且还可以跨版本：

在目标库创建db link

create public database link to10g113 connect to mysteel identified by asdefg using 'mysteelnet';

（需要在目标库的tnsnames.ora里配置mysteelnet别名）

```
MYSTEELNET =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.100.130)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MYSTEEL)
    )
  )
```

测试db link是否能用

select * from ENGMAR_MARKET@to10g130 where rownum < 11;

查看个用户所有对象占用多少硬盘空间 ：

SELECT SUBSTRB('ABCDEFG',5,4.2) "Substring with bytes"

    FROM DUAL;

使用db link, impdp进行数据迁移 ：

impdp directory=dirdp logfile=imp113mysteel.log network_link=to10g130 remap_tablespace=users:mysteelnet

remap_schema=mysteel:mysteelnet parallel=16

create tablespace nettest datafile '/oradata/userbehavior/nettest.dbf' size 100M autoextend on maxsize 2G;

impdp directory=dirdp logfile=imp113nettest.log network_link=to10g130 parallel=16

[oracle@ibm-userbehavior-db01 ~]$ impdp directory=dirdp logfile=imp113nettest.log network_link=to10g130 parallel=16

Import: Release 11.2.0.4.0 - Production on Fri Jul 28 19:43:38 2017

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

Username: / as sysdba

Connected to: Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production

With the Partitioning, OLAP, Data Mining and Real Application Testing options

ORA-39006: internal error

ORA-39113: Unable to determine database version

ORA-04052: error occurred when looking up remote object SYS.DBMS_UTILITY@TO10G130

ORA-00604: error occurred at recursive SQL level 3

ORA-06544: PL/SQL: internal error, arguments: [55916], [], [], [], [], [], [], []

ORA-06553: PLS-801: internal error [55916]

ORA-02063: preceding 2 lines from TO10G130

ORA-39097: Data Pump job encountered unexpected error -4052

[oracle@ibm-userbehavior-db01 ~]$

好像遇到了bug

参考 ： http://blog.csdn.net/tonyzhou_cn/article/details/9343067

涉及三方面的字符集，

1. oracel server端的字符集；

2. oracle client端的字符集；

3. dmp文件的字符集。

在做数据导入的时候，需要这三个字符集都一致才能正确导入。

1. 查询oracle server端的字符集

select userenv('language') from dual;

2. 如何查询dmp文件的字符集

select nls_charset_name(to_number('0354','xxxx')) from dual;

如果dmp文件很大，比如有2G以上(这也是最常见的情况)，用文本编辑器打开很慢或者完全打不开，可以用以下命令(在unix主机上)：

cat exp.dmp |od -x|head -1|awk '{print $2 $3}'|cut -c 3-6

然后用上述SQL也可以得到它对应的字符集。

3. 查询oracle client端的字符集

在unix平台下，就是环境变量NLS_LANG。

$echo $NLS_LANG

AMERICAN_AMERICA.ZHS16GBK

SYS@MYSTEEL> select * from nls_database_parameters;

PARAMETER	VALUE
NLS_LANGUAGE	AMERICAN
NLS_TERRITORY	AMERICA
NLS_CURRENCY	$
NLS_ISO_CURRENCY	AMERICA
NLS_NUMERIC_CHARACTERS	.,
NLS_CHARACTERSET	ZHS16GBK
NLS_CALENDAR	GREGORIAN
NLS_DATE_FORMAT	DD-MON-RR
NLS_DATE_LANGUAGE	AMERICAN
NLS_SORT	BINARY
NLS_TIME_FORMAT	HH.MI.SSXFF AM
NLS_TIMESTAMP_FORMAT	DD-MON-RR HH.MI.SSXFF AM
NLS_TIME_TZ_FORMAT	HH.MI.SSXFF AM TZR
NLS_TIMESTAMP_TZ_FORMAT	DD-MON-RR HH.MI.SSXFF AM TZR
NLS_DUAL_CURRENCY	$
NLS_COMP	BINARY

```
NLS_LENGTH_SEMANTICS             BYTE
NLS_NCHAR_CONV_EXCP              FALSE
NLS_NCHAR_CHARACTERSET          AL16UTF16
NLS_RDBMS_VERSION               10.2.0.1.0


20 rows selected.

SYS@MYSTEEL>
```

NLS_CHARACTERSET是数据库字符集，NLS_NCHAR_CHARACTERSET是国家字符集

Oracle中有两大类字符型数据，VARCHAR2是按照数据库字符集来存储数据。而NVARCHAR2是按照国家字符集存储数据的。同样，CHAR和NCHAR也一样，一是数据库字符符，一是国家字符集。

字符集不同，二进制码的组合就不同。

NLS_CHARACTERSET是db中定义的char类型的编码方式

NLS_NCHAR_CHARACTERSET是db中定义的nchar类型的编码方式

源库NLS_CHARACTERSET和目标库NLS_CHARACTERSET相同，但NLS_NCHAR_CHARACTERSET不同。

查看mysteel schema下所有表是否含NVARCHAR2类型的列 ：

```
select object_name, object_type, owner from dba_objects where owner='MYSTEEL' and object_type='TABLE';
```

desc 查到的每个表

确保所有表都不含NVARCHAR2类型的列之后才放心导出

通过db link远程访问含大对象的表报错 ：

```
SYS@userbehavior> select * from MEMBER_RELATION_TEST@to10g130;
ERROR:
ORA-22992: cannot use LOB locators selected from remote tables
no rows selected
SYS@userbehavior> host oerr ora 22992
22992, 00000, "cannot use LOB locators selected from remote tables"
// *Cause:  A remote LOB column cannot be referenced.
// *Action:  Remove references to LOBs in remote tables.
SYS@userbehavior>
```