



OCP 认证培训教材

数据泵



腾科 ORACLE 教学部



数据泵导入导出技术的结构

在数据泵导出导入技术中，涉及导出实用程序expdp和导入实用程序impdp，当启动数据泵导入或导出程序时，在数据库服务器端启动相应的服务器进程，完成数据的导入及导出任务，所以也称数据泵技术是基于oracle数据库服务器的，导入及导出的数据文件也保存在数据库服务器端。

expdp程序启动数据库服务器端的服务器进程，服务器进程完成数据的备份并将备份文件写入数据库服务器端磁盘中，导出的备份文件在导入时只能数据泵的导入实用程序impdp完成。

使用数据泵技术的优点：

- 1、可以只处理某些对象，或者不处理某些对象，或只处理某些对象下满足条件的数据
- 2、不实际导出的情况下，估计整个导出工作需要占用的磁盘空间
- 3、只导出元数据（如表结构），不导出实际数据。
- 4、可以进行并行操作

数据泵导入导出的目录对象

数据泵作业在数据库服务器上创建所有的备份文件，要求数据泵必须使用目录对象，以防止用户误操作数据库服务器上特定目录下的操作系统文件。目录对象对应于操作系统上的一个指定目录。

如果当前用户是DBA用户，可以使用默认的目录对象而不必再创建数据泵操作的工作目录。此时数据泵作业会将备份文件、日志文件以及SQL文件存储在该目录下，查看默认目录：

```
SQL> col directory_name for a15
SQL> col owner for a23
SQL> col directory_path for a50
SQL> select * from dba_directories where directory_name='DATA_PUMP_DIR';
```

不具备目录对象的数据泵作业错误

```
[oracle@oracle Desktop]$ expdp scott/scott
Export: Release 11.2.0.4.0 - Production on Fri Jan 17 17:04:18 2014
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.
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-39002: invalid operation
ORA-39070: Unable to open the log file.
ORA-39145: directory object parameter must be specified and non-null
```

如果需要自己创建目录对象，需要具有create any directory权限，首先向scott用户授权create any directory，然后创建属于scott用户的数据泵目录对象。

```
SQL> grant create any directory to scott;
SQL> conn scott/scott
SQL> create directory scott_dump_dir as '/home/oracle';
```

下面我们在数据库服务器上创建数据泵导入导出目录，采用sys用户登录



```
[oracle@oracle ~]$ sqlplus / as sysdba
```

```
SQL> startup
```

```
SQL> create directory pump_dir as '/u01/app/oracle/wl';
```

在目录创建后，必须给导入导出的用户赋予目录的读写权限

```
SQL> grant read,write on directory pump_dir to scott;
```

授权成功后，在使用数据泵导入或导出scott用户的数据时，就可以使用该目录作为存储和恢复目录了。

expdp参数

attach

用于在客户会话与已存在导出作业之间建立关联，语法如下：

```
attach=[schema_name.]job_name, schema_name
```

用于指定方案名，job_name指定导出作业名，如果使用attach选项，在命令行除了连接字符串和attach选项外，不能指定任何其它选项。

```
expdp scott/tiger attach=scott.export_job
```

content

用于指定要导出的内容，默认是all，语法如下：

```
CONTENT={ALL | DATA_ONLY | METADATA_ONLY},
```

当设置content选项为all时，将导出对象定义及其所有数据，当设置该选项为data_only时，只能导出对象数据；当设置该选项为metadata_only时，只导出对象定义。

```
expdp scott/tiger directory=dump dumpfile=a.dmp content=metadata_only
```

directory

用于指定转储文件和日志文件所在位置。语法如下：directory=directory_object，directory_object用于指定目录对象名称。目录对象是使用create directory语句建立的对象，而不是os目录。

```
expdp scott/tiger directory=dump dumpfile=a.dmp
```

dumpfile

指定转储文件的名称，默认名称为expda.dmp，语法如下：

```
dumpfile=[directory_object:]file_name[,...], directory_object指定目录对象名，file_name指定转储文件名。
```

```
expdp scott/tiger directory=dump1 dumpfile=dump2:a.dmp
```

estimate

用于指定估算被导出表所占用磁盘空间的方法，默认值为blocks，语法如下：

Estimate={blocks|statistics}，设置blocks时，oracle会按照目标对象占用的数据块个数乘以数据块尺寸估算对象占用的空间，设置statistics时，oracle会根据最近的统计值估算



对象占用的空间。

```
expdp scott/tiger tables=emp estimate=statistics directory=dump dumpfile=a.dmp
```

estimate_only

用于指定是否估算出作业所占用的磁盘空间，默认n，语法如下：estimate_only={y|n}，设置为y时，导出作业只估算对象所占用的磁盘空间，而不会执行导出操作，设置为n时，导出作业不仅估算对象所占用的磁盘空间，且会执行导出操作。

```
expdp scott/tiger estimate_only=y nologfile=y
```

exclude

用于指定执行导出操作时要排除的对象类型或相关对象，语法如下：

exclude=object_type[:name_clause][,...]，object_type指定要排除的对象类型，name_clause指定要排除的具体对象，exclude和include不能同时使用。

```
expdp scott/tiger directory=dump dumpfile=a.dmp exclude=view
```

filesize

用于指定导出文件的最大尺寸，默认值是0(表示文件尺寸无限制)语法如下：

filesize=integer[B|K|M|G]

```
expdp scott/tiger directory=dump dumpfile=hr_3M.dmp filesize=3M
```

flashback_time

指定导出特定时间点的表数据。语法如下：

flashback_time="to_timestamp(time_value)" time_value用于指定日期时间值，

flashback_time和flashback_scn不能同时用。

```
expdp scott/tiger directory=dump dumpfile=a.dmp
```

```
flashback_time="to_timestamp('25-08-2009 14:34:00', 'dd-mm-yyyy  
hh24:mi:ss')"
```

flashback_scn

用于指定导出特定scn时刻的表数据。语法如下：flashback_scn=scn_value

```
expdp scott/tiger directory=dump dumpfile=a.dmp flashback_scn=385823
```

full

指定数据库模式导出，默认主n，语法如下：full={y|n}，设置为y时，表示执行数据库导出。

```
expdp scott/tiger directory=dump dumpfile=full.dmp full=y
```

include



指定导出时要包含的对象类型及相关对象。语法如下：

include=object_type[:name_clause][,...], object_type指定要导出的对象类型，name_clause指定要导出的对象名。

```
expdp scott/tiger directory=dump dumpfile=a.dmp include=table
```

job_name

指定导出作业的名称，语法如下：job_name=jobname_string, jobname_string用于指定导出作业的名称。

```
expdp scott/tiger directory=dump dumpfile=a.dmp job_name=wangli
```

logfile

指定导出日志文件的名称，默认名称为export.log, 语法如下：

LOGFILE=[directory_object:]file_name。directory_object用于指定目录对象名称，file_name用于指定导出日志文件名。

```
expdp scott/tiger directory=dump dumpfile=a.dmp logfile=a.log
```

network_link

指定数据库链名，如果要将远程数据库对象导出到本地例程的转储文件中，必须设置些选项。语法如下：NETWORK_LINK=source_database_link。source_database_link用于指定数据库链名。

```
expdp scott/tiger directory=dump dumpfile=a.dmp network_like=orcl
```

nologfile

用于指定禁止发生导出日志文件，默认值n, 语法如下：nologfile={y|n} 设置为y时，导出操作不会生成日志文件。

```
expdp scott/tiger dumpfile=dump:a.dmp nologfile=y
```

parallel

用于指定执行导出操作的并行进程个数，默认是1, 语法如下：parallel=integer Integer用于指定并行进程个数。通过执行并行导出操作，可以加快导出速度。

```
Expdp scott/tiger directory=dump dumpfile=a.dmp parallel=3
```

parfile

指定导出参数文件的名称，语法如下：PARFILE=[directory_path]file_name。

directory_path指定参数文件所在目录，file_name指定参数文件名。参数文件a.txt示例如下：tables=dept,emp

Directory=dump

Dumpfile=tab.dmp

参数文件不能包含parfile选项。参数文件里不指定directory_path



```
Expdp scott/tiger parfile=a.txt
```

Query

指定过滤导出数据的where条件，语法如下：query=[schema.][table_name:]query_lause
Schema用于指定方案名，table_name指定表名，query_lause指定条件限制子句，query选项不能与connect=metadata_only、estimate_only、transport_tablespace等选项同时使用。
expdp scott/tiger directory=dump dumpfile=a.dmp tables=emp query=' " where deptno=20" '

schemas

指定执行方案模式导出，默认为当前用户方案，语法如下：SCHEMAS=方案名称[, ...]
方案名称：用于指定方案名，用户可以导出自身方案，但如果要导出其它方案，须具有exp_full_database角色或DBA角色。

```
Expdp system/manager directory=dump dumpfile=a.dmp schemas=scott,system
```

status

指定显示导出作业进程的详细状态，默认为0，语法如下：STATUS=[整数]，整数用于指定显示导出作业状态的时间间隔（秒），指定了该选项后，每隔特定时间会显示作业完成的百分比。

```
expdp system/manager directory=dump dumpfile=a.dmp full=y status=30
```

tables

用于指定表模式导出。语法如下：

```
TABLES=[schema_name.]table_name[:partition_name][, ...]
```

schema_name指定方案名，table_name指定要导出的表名，partition_name用于指定要导出的分区名。用户可以直接导出其自身方案的表，但要导出其它方案的表，须具有exp_full_database角色或DBA角色。

```
expdp system/manager directory=dump dumpfile=a.dmp tables=scott.dept,scott.emp
```

tablespaces

指定要导出的表空间列表，语法如下：TABLESPACE=tablespace_name[, ...]

tablespace_name用于指定要导出的表空间。指定选项时，会导出该表空间上的所有表。

```
Expdp system/manager directory=dump dumpfile=a.dmp tablespaces=user01
```

transport_full_check

用于指定被搬移表空间和未搬移表演关联关系的检查方式，默认值是n，语法如下：

TRANSPORT_FULL_CHECK={Y|N} 设置为y时，导出作业会检查表空间之间的完整关联关系，如果表所在表空间或其索引所在表空间只有一个表空间被搬移，将显示错误信息，当设置该选



项为n时，导出作业只检查单端依赖，如果搬移索引所在的表空间但未搬移表所在表空间，将显示错误信息，如果搬移表所在表空间，未搬移索引所在表空间，则不会显示错误信息

```
expdp system/manager directory=dump dumpfile=b.dmp  
transport_tablespace=user01 transport_full_check=y
```

transport_tablespace

指定执行表空间模式导出，语法如下：TRANSPORT_TABLESPACES=Tablespace_name[, ...]

Tablespace_name用于指定要导出的表空间名称，导出表空间时，要求数据库用户必须有exp_full_database角色或DBA角色。

```
expdp system/manager directory=dump dumpfile=b.dmp transport_tablespace=user01
```

数据泵导出实例

使用expdp可以导出整个数据库、单个模式、特定的表或特定的表空间。

1、导出整个数据库，我们使用system用户登录数据库，限制备份的数据文件大小为230M，一旦备份数据文件满，则自动创建一个新的备份文件，使用%U来实现备份文件的自动创建，nologfile=y即不记录备份过程。

```
[oracle@oracle Desktop]$ expdp system/oracle dumpfile=pump_dir:mydb_%U.dat  
filesize=230M nologfile=y job_name=tom full=y
```

2、导出一个模式

导出scott模式，在例子中没有schema参数，但是默认导出登录数据库时的模式对象。

```
[oracle@oracle Desktop]$ expdp scott/scott  
dumpfile=scott_dump_dir:scottschema.dmp logfile=scott_dump_dir:scottschema.log  
或
```

```
[oracle@oracle Desktop]$ expdp scott/scott directory= scott_dump_dir  
dumpfile=scottschema.dmp logfile=scott_dump_dir:scottschema.log schemas=scott
```

3、导出特定的表，使用tables参数指定导入的表的列表，如果该表不属于登录的用户，但是登录用户有访问这些表的权限，则tables参数的表必须使用schema.tablename的方式。

```
[oracle@oracle Desktop]$ expdp system/oracle  
dumpfile=pump_dir:scott_tables_%U.dat tables=scott.emp,scott.dept nologfile=y  
job_name=only_scott
```

4、导出表空间

导出表空间使用tablespaces参数，如果有多个表空间需要导出，表空间名使用逗号隔开，使用parallel参数指定数据导出并行线程数量，与之对应使用替换变量%U来创建相应数量的备份数据文件，这样每个线程可以独立写一个备份数据文件，提高导出速度。

```
[oracle@oracle Desktop]$ expdp system/oracle dumpfile=pump_dir:users_tbs_%U.dmp  
tablespaces=users filesize=230M parallel=2 logfile=users_tbs.log  
job_name=exp_users_tbs
```

5、导出数据

使用content参数，可以指定导出表数据和元数据(应用参数all)，导出表行数据(对应参数



data_only) 或只导出元数据即表以及其他数据库对象的定义 (对应参数metadata_only)

```
[oracle@oracle Desktop]$ expdp system/oracle  
dumpfile=pump_dir:mydb_dataony_%U.dat filesize=230M job_name=larry full=y  
content=data_only logfile=pump_dir:mydb_exp_dataonly.log
```

6、使用参数文件

在使用expdp导出数据时，由于参数很多导致每次执行备份都输入一大串指令，这样不但繁琐且不易修改，数据泵技术允许使用参数文件，用户在参数文件中创建各种参数，保存该文件paraname.par文件，然后再执行导出时使用parfile指定参数文件的位置执行导出备份。

```
[oracle@oracle ~]$ vi paraname.par
```

```
directory=pump_dir  
dumpfile=para_data_only_%U.dmp  
content=data_only  
exclude=table:"in(' salgrade',' bonus')"  
logfile=para_data_only.log  
filesize=50M  
parallel=2  
job_name=para_data_only
```

```
[oracle@oracle ~]$ expdp scott/scott parfile=/home/oracle/paraname.par
```

7、估计空间导出文件的空间大小

estimate_only计算导出数据所需要的存储空间，在导出的数据大小不清楚时，事先知道备份文件的大小，可以提前分配磁盘空间，防止由于磁盘空间不足引起的expdp导出作业停止。

```
[oracle@oracle ~]$ expdp system/oracle full=y estimate_only=y estimate=statistics  
nologfile=y
```

使用STATISTICS方法计算system用户所有数据库对象的大小。最后给出一个总的估计结果。

impdp参数

attach

用于在客户会话与已存在导入作业之间建立关联，语法如下：

```
attach=[schema_name.]job_name, schema_name
```

用于指定方案名，job_name指定导出作业名，如果使用attach选项，在命令行除了连接字符串和attach选项外，不能指定任何其它选项。

```
impdp scott/tiger attach=import_job
```

content

用于指定要导入的内容，默认是all，语法如下：

```
CONTENT={ALL | DATA_ONLY | METADATA_ONLY},
```




当设置content选项为all时，将导入对象定义及其所有数据，当设置该选项为data_only时，只能导入对象数据；当设置该选项为metadata_only时，只导入对象定义。

```
impdp scott/tiger directory=dump dumpfile=a.dmp content=data_only tables=dept,emp
```

directory

用于指定转储文件所在位置。语法如下：directory=directory_object，directory_object用于指定目录对象名称。目录对象是使用create directory语句建立的对象，而不是os目录。

```
impdp scott/tiger directory=dump dumpfile=a.dmp tables=emp
```

dumpfile

指定转储文件的名称，默认名称为expda.dmp，语法如下：

dumpfile=[directory_object:]file_name[,...], directory_object指定目录对象名，file_name指定转储文件名。

```
impdp scott/tiger directory=dump dumpfile=a.dmp tables=emp
```

estimate

用于指定估算执行网络导入操作时要生成的数据量，默认值为blocks，语法如下：

Estimate={blocks|statistics}，设置blocks时，oracle会根据数据块个数乘以数据块尺寸估算要生成的数据量，设置statistics时，oracle会根据统计值估算对要生成的数据量。

```
impdp scott/tiger tables=emp estimate=statistics directory=dump dumpfile=a.dmp
```

exclude

用于指定执行导入操作时要过滤的对象类型或特定对象，语法如下：

exclude=object_type[:name_clause][,...], object_type指定对象类型，name_clause指定对象名。

```
impdp scott/tiger directory=dump dumpfile=a.dmp exclude=cluster
```

flashback_time

指定导入特定时间点的表数据。语法如下：flashback_time="

to_timestamp(time_value)" time_value用于指定日期时间值，flashback_time和flashback_scn不能同时用。

```
impdp scott/tiger directory=dump flashback_time=" to_timestamp(' 25-08-2009 14:34:00' , ' dd-mm-yyyy hh24:mi:ss' )" )"
```

flashback_scn

用于指定导入特定scn时刻的表数据。语法如下：flashback_scn=scn_value

```
impdp scott/tiger directory=dump flashback_scn=385823
```



full

指定是否要导入转储文件的全部内容，默认y, 语法如下：full={y|n}, 设置为y时，表示所有内容。

```
impdp scott/tiger directory=dump dumpfile=full.dmp full=y
```

include

指定导入时要包含的对象类型及相关对象。语法如下：

include=object_type[:name_clause][,...], object_type指定要导入的对象类型，name_clause指定要导入的对象名。

```
impdp scott/tiger directory=dump dumpfile=a.dmp include=view
```

job_name

指定导入操作的作业名称，语法如下：job_name=jobname_string, jobname_string用于指定导出作业的名称。

```
impdp scott/tiger directory=dump dumpfile=a.dmp job_name=wangli
```

logfile

指定导入日志文件的名称，默认名称为export.log, 语法如下：

LOGFILE=[directory_object:]file_name。directory_object用于指定目录对象名称，file_name用于指定导出日志文件名。

```
impdp scott/tiger directory=dump dumpfile=a.dmp logfile=a.log
```

network_link

指定数据库链名, 如果要将远程数据库对象导入到本地例程的转储文件中，必须设置些选项。语法如下：NETWORK_LINK=source_database_link。source_database_link用于指定数据库链名。

```
impdp scott/tiger directory=dump tables=emp network_like=orcl
```

nologfile

用于指定禁止生成入日志文件，默认值n, 语法如下：nologfile={y|n} 设置为y时。

```
impdp scott/tiger dumpfile=a.dmp nologfile=y
```

parallel

用于指定执行并行导入操作，默认是1, 语法如下：parallel=integer

Integer用于指定并行进程个数。通过执行并行导出操作，可以加快导出速度。

```
impdp scott/tiger directory=dump dumpfile=a.dmp parallel=3
```



parfile

指定导入参数文件的名称，语法如下：PARFILE=[directory_path]file_name。

directory_path指定参数文件所在目录，file_name指定参数文件名。参数文件a.txt示例如下：tables=dept,emp

Directory=dump

Dumpfile=tab.dmp

参数文件不能包含parfile选项。参数文件里不指定directory_path

```
impdp scott/tiger parfile=a.txt
```

query

指定过滤导入数据的where条件，语法如下：query=[schema.][table_name:]query_laue

Schema用于指定方案名，table_name指定表名，query_laue指定条件限制子句，query选项不能与connect=metadata_only、estimate_only、transport_tablespace等选项同时使用。

```
impdp scott/tiger directory=dump dumpfile=tab.dmp tables=emp query='where deptno=20'
```

schemas

指定执行方案模式导入，默认为当前用户方案，语法如下：SCHEMAS=方案名称[, ...]

方案名称：用于指定方案名，用户可以导入自身方案，但如果要导入其它方案，须具有exp_full_daabase角色或DBA角色。

```
impdp system/manager directory=dump dumpfile=tab.dmp schemas=scott
```

status

指定显示导入作业的详细状态，默认为0，语法如下：STATUS=[整数]，整数用于指定显示导出作业状态的时间间隔（秒），指定了该选项后，每隔特定时间会显示作业完成的百分比。

```
impdp system/manager directory=dump dumpfile=full.dmp full=y status=10
```

tables

用于指定表模式导入。语法如下：

TABLES=[schema_name.]table_name[:partition_name][, ...]

schema_name指定方案名，table_name指定要导入的表名，partition_name用于指定要导入的分区名。

```
impdpd system/manager directory=dump dumpfile=tab.dmp tables=emp
```

tablespaces

指定执行表空间模式导入，语法如下：TABLESPACE=tablespace_name[, ...]

tablespace_name用于指定要导入的表空间。



```
impdp system/manager directory=dump dumpfile=a.dmp tablespaces=user01
```

transport_full_check

用于指定被搬移表空间和未搬移表演关联关系的检查方式，默认值是n，语法如下：

TRANSPORT_FULL_CHECK={Y|N} 设置为y时，导入作业会检查表空间之间的完整关联关系，如果表所在表空间或其索引所在表空间只有一个表空间被搬移，将显示错误信息，当设置该选项为n时，导入作业只检查单端依赖，如果搬移索引所在的表空间，但未搬移表所在表空间，将显示错误信息，如果搬移表所在表空间，未搬移索引所在表空间，则不会显示错误信息

```
impdp system/manager directory=dump dumpfile=b.dmp  
transport_tablespaces=user01 transport_full_check=y  
transport_datafiles='/u01/app/oracle/tbs6.dbf'
```

transport_tablespaces

指定执行表空间模式导入，语法如下：TRANSPORT_TABLESPACES=Tablespace_name[, ...]

Tablespace_name用于指定表空间名称，导入表空间时，要求数据库用户必须有imp_full_database角色或DBA角色。

```
impdp system/manager directory=dump dumpfile=b.dmp transport_tablespaces=user01
```

transport_datafiles

指定搬移表空间时要被导入到目标数据库的数据文件

Table_exists_action

指定当表存在时，导入作业要执行的操作。默认是skip，语法如下：

```
table_exists_action={skip|append|truncate|replace}
```

skip_unusable_indexes

指定导入时是否跳过不可使用的索引，默认是n，语法如下：skip_unusable_indexes={y|n}

reuse_datafiles

指定建立表空间时是否覆盖已存在的数据文件，默认n，语法如下：Reuse_datafiles={y|n}

remap_tablespace

用于将源表空间的所有对象导入到目标表空间中，语法如下：

remap_tablespace=source_tablespace:target_tablespace。source_tablespace指定源表空间名称，target_tablespace指定目标表空间名称。



remap_schema

用于将源方案的所有对象装载到目标方案中，语法如下：

```
remap_schema= source_schema:target_ schema
```

remap_datafile

将源数据文件名转变为目标数据文件名，语法如下：

```
remap_ datafile = source_ datafile:target_ datafile
```

数据泵导入实例

使用数据泵导入impdp可以导入基于使用数据泵导出的备份文件，可以导入整个数据库，指定的表空间，指定的表或指定的数据库对象类型，如索引、函数、存储过程和触发器等。

1、导入整个数据库

导入整个数据库至少需要二个参数，一个是full, 设置full=y说明是导入全库，一个是dumpfile, 说明要导入的备份文件的目录和名称，job_name参数，允许切换到交换模式，允许终止或重启导入会话

```
[oracle@oracle oracle]$ mkdir /u01/app/oracle/wl/
```

```
[oracle@oracle ~]$ sqlplus / as sysdba
```

```
SQL> startup
```

```
SQL> create directory pump_dir as '/u01/app/oracle/wl';
```

```
SQL> select * from dba_directories where directory_name='PUMP_DIR';
```

导出：

```
[oracle@oracle Desktop]$ expdp system/oracle dumpfile=pump_dir:full_db_%U.dat  
filesize=230M nologfile=y job_name=tom full=y
```

```
SQL> drop user scott cascade;
```

导入：

```
[oracle@oracle Desktop]$ impdp system/oracle dumpfile=pump_dir:full_db_%U.dat  
logfile= pump_dir:myfulldb.log parallel=3 job_name=my_fullldb_impdp full=y
```

2、导入表空间

使用impdp导入特定的表空间时，需要有备份表空间文件，需要使用tablespaces参数说明要导入的表空间名，此时实际上是导入表空间中的所有数据库对象，由于表空间中有表对象，需要使用table_exists_action来告诉impdp怎么做，可以使用replace重建表，truncate删除当前表中的数据，然后使用备份文件中的表数据进行加载，但会跳过所有相关元数据。

```
[oracle@oracle Desktop]$ expdp system/oracle  
dumpfile=pump_dir:users_tbs_%U.dmp tablespaces=users filesize=230M  
parallel=2 logfile=users_tbs.log job_name=exp_users_tbs
```

```
SQL> drop table scott.emp;
```

```
SQL> drop table scott.dept;
```

```
[oracle@oracle Desktop]$ impdp system/oracle  
dumpfile=pump_dir:users_tbs_%U.dmp logfile= pump_dir:users.log
```



```
tablespaces=users table_exists_action=replace
```

3、导入指定的表

使用impdp导入特定的表使用tables参数，该参数的后，是要导入表的对象列表，如果有多个表，使用逗号分隔。

```
[oracle@oracle Desktop]$ expdp system/oracle
dumpfile=pump_dir:scott_tables_%U.dat tables=scott.emp,scott.dept
nologfile=y job_name=only_scott
SQL> drop table scott.emp;
SQL> grant read,write on directory pump_dir to scott;
[oracle@oracle Desktop]$ impdp scott/scott
dumpfile=pump_dir:scott_tables_%U.dat nologfile=y tables=emp
table_exists_action=replace
```

如将表dept,emp 导入到scott 方案中

```
impdp scott/tiger directory=dump_scott dumpfile=tab.dmp tables=dept,emp
如将表dept 和emp 从scott 方案导入到system 方案中，对于方案的转移，必须使用remap_shcema 参数
impdp system/manage directory=dump_scott dumpfile=tab.dmp
tables=scott.dept,scott.emp remap_schema=scott:system
```

4、导入指定的数据库对象

导入指定的数据库对象使用include参数，恢复scott用户的所有表和触发器对象，对于已存在的表则重新加载数据

```
[oracle@oracle wl]$ expdp system/oracle dumpfile=pump_dir:full_db_%U.dat
filesize=230M nologfile=y job_name=tom full=y
```

```
[oracle@oracle wl]$ impdp scott/scott dumpfile=pump_dir:full_db_%U.dat
nologfile=y include=table,trigger table_exists_action=replace
```

导入方案

如将dump_scott 目录下的schema.dmp 导入到scott 方案中

```
impdp scott/tiger directory=dump_scott dumpfile=schema.dmp schemas=scott
```

如将scott 方案中的所有对象转移到system 方案中

```
impdp system/redhat directory=dump_scott dumpfile=schema.dmp
schemas=scott remap_schema=scott:system
```

表空间迁移