



**ParnassusData**<sup>TM</sup>



Oracle export/import传统导出导入工具与 expdp/impdp数据泵必知必会





```
macleandeMacBook-Pro:~ maclean$ whoami  
maclean
```

## 刘相兵(Maclean Liu):

- dbrecover for Oracle/MySQL数据恢复软件作者
- shoug上海Oracle用户组主席
- 前Oracle公司资深技术顾问
- an Oracle ACE Director
- 10g & 11g OCM
- askmaclean.com



**Parnassus**  
诗檀

**ORACLE** Silver Partner



# 搞清楚啥是export/import传统工具 主体是 exp 和 imp 这2个可执行文件 在命令行里执行，不是在sqlplus或plsqldveloper里！！

```
[oracle@ocp ~]$ exp help=y
Export: Release 11.2.0.4.0 - Production on Wed May 20 12:23:20 2020
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

You can let Export prompt you for parameters by entering the EXP
command followed by your username/password:

Example: EXP SCOTT/TIGER

Or, you can control how Export runs by entering the EXP command followed
by various arguments. To specify parameters, you use keywords:

Format: EXP KEYWORD=value or KEYWORD=(value1,value2,...,valueN)
Example: EXP SCOTT/TIGER GRANTS=Y TABLES=(EMP,DEPT,MGR)
or TABLES=(T1:P1,T1:P2), if T1 is partitioned table
```

```
[oracle@ocp ~]$ imp help=y
Import: Release 11.2.0.4.0 - Production on Wed May 20 12:23:40 2020
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

You can let Import prompt you for parameters by entering the IMP
command followed by your username/password:

Example: IMP SCOTT/TIGER

Or, you can control how Import runs by entering the IMP command followed
by various arguments. To specify parameters, you use keywords:

Format: IMP KEYWORD=value or KEYWORD=(value1,value2,...,valueN)
Example: IMP SCOTT/TIGER IGNORE=Y TABLES=(EMP,DEPT) FULL=N
or TABLES=(T1:P1,T1:P2), if T1 is partitioned table
```



# 搞清楚啥是expdp/impdp数据泵工具

## 主体是 expdp 和 impdp 这2个可执行文件

### 在命令行里执行，不是在sqlplus或plsqldeveloper里！！

```
[oracle@ocp ~]$ expdp help=y
Export: Release 11.2.0.4.0 - Production on Wed May 20 12:24:13 2020
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

The Data Pump export utility provides a mechanism for transferring data objects
between Oracle databases. The utility is invoked with the following command:

Example: expdp scott/tiger DIRECTORY=dmpdir DUMPFILE=scott.dmp

You can control how Export runs by entering the 'expdp' command followed
by various parameters. To specify parameters, you use keywords:

Format: expdp KEYWORD=value or KEYWORD=(value1,value2,...,valueN)
Example: expdp scott/tiger DUMPFILE=scott.dmp DIRECTORY=dmpdir SCHEMAS=scott
or TABLES=(T1:P1,T1:P2), if T1 is partitioned table
```

```
[oracle@ocp ~]$ impdp help=y
Import: Release 11.2.0.4.0 - Production on Wed May 20 12:24:41 2020
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

The Data Pump Import utility provides a mechanism for transferring data objects
between Oracle databases. The utility is invoked with the following command:

Example: impdp scott/tiger DIRECTORY=dmpdir DUMPFILE=scott.dmp

You can control how Import runs by entering the 'impdp' command followed
by various parameters. To specify parameters, you use keywords:

Format: impdp KEYWORD=value or KEYWORD=(value1,value2,...,valueN)
Example: impdp scott/tiger DIRECTORY=dmpdir DUMPFILE=scott.dmp
```



## 对初学者最容易搞错的概念

- `export/import`传统工具和数据泵工具组`expdp/impdp`所生成的文件格式是不兼容的！！
- 虽然一般导出数据的时候大部分人给这2种工具生成的导出文件的扩展名都是`dmp`
- 但是`exp`生成的`dmp`和`expdp`生成的`dmp`，格式完全不一样！
- 文件后缀扩展名是不影响文件实际使用的！
  
- 倡议：
  - `Expdp`生成的文件后缀扩展名使用`oraexpdp`
  - `Exp`生成的文件后缀扩展名使用`oraexport`





## export/import简单使用

- 从oracle 11.2.0.4导出pd用户到oracle 10.2.0.5中，则需要使用10.2.0.5服务器上的exp/imp

在10.2.0.5服务器上执行：

```
exp system/oracle@SOURCE owner=pd buffer=9000000 file=/目录/pd.oraexport log=pd_exp.log
```

```
imp system/oracle@TARGET file=/目录/pd.oraexport full=y buffer=9000000 log=pd_imp.log
```

如果导入时想将pd用户的数据导入到pda用户则：

```
imp system/oracle@TARGET file=/目录/pd.oraexport fromuser=pd touser=pda buffer=9000000 log=pd_imp.log commit=y
```





## export/import简单使用

建议在imp前执行的:

- 创建对应的用户, 授予必要的权限

- 创建对应的表空间

原则上export/import传统工具组不支持转换表空间等信息, 例如原来导出时表创建在ABC表空间上, 则导入时也需要创建ABC表空间, 否则imp建表会报错。

当目标库中要导入的表已经存在时, imp将不导入数据; 如果需要强制导入数据到已经存在的表中, 那么需要使用ignore=y选项

```
imp system/oracle@TARGET file=/目录/pd.oraexport fromuser=pd touser=pda buffer=9000000 log=pd_imp.log commit=y ignore=y
```



# expdp/impdp简单使用

数据库泵工具组与export/import最大的2个区别:

1. 要在数据库创建directory目录对象，所有impdp/expdp的文件都在这目录下
2. 所有的导入导出文件都发生在数据库服务器上！expdp/impdp可以通过用户pc上的expdp/impdp工具发起，但实际操作都发生在数据库服务器上。expdp/impdp是纯服务器端的技术！！而export/import是server-client模型的。



## expdp/impdp简单使用

以下从oracle 11.2.0.4导出pd用户到oracle 11.2.0.1中的pda用户，并转换表空间ABC为EFG

在oracle 11.2.0.4源头库中创建一个目录，在sqlplus或plsqldeveloper或sqldeveloper里执行：

Create directory s01 as '源头服务器端的一个目录'; //这是一条SQL命令!!!! 注意这个目录在服务器操作系统上必须已经存在

在命令行执行：

```
expdp dumpfile=pd.oraexpdp directory=s01 schemas=pd logfile=pd_expdp.log version=11.2.0.1
```

以上执行完成后，在目标数据库11.2.0.1中也要创建一个目录，并将上述生成的pd.oraexpdp 拷贝到目标数据库所在服务器上的该目录中，在目标库的sqlplus或plsqldeveloper或sqldeveloper里执行：

Create directory d01 as '目标服务器端的一个目录'; //这是一条SQL命令!!!! 注意这个目录在服务器操作系统上必须已经存在

在命令行执行：

```
impdp dumpfile=pd.oraexpdp directory=d01 full=y remap_schema=pd:pda remap_tablespace=ABC:EFG
```



## expdp/impdp简单使用

REMAP\_SCHEMA 将一个方案中的对象加载到另一个方案

REMAP\_TABLESPACE将表空间对象重新映射到另一个表空间

注意这2个参数是较常用的，一定要会用。否则收了别人的expdp文件，自己连简单的转换都做不了，岂不尴尬？



## expdp/impdp简单使用

当导入数据时目标库中已经存在对应的数据表时怎么办？

impdp存在TABLE\_EXISTS\_ACTION选项:导入对象已存在时执行的操作。有效的关键字为: APPEND, REPLACE, [SKIP] 和 TRUNCATE。

APPEND直接追加数据到已经存在的表

REPLACE代表先DROP掉原表后重建该表并插入数据

TRUNCATE代表先TRUNCATE掉原表后重建该表并插入数据

SKIP代表对这类表不做任何操做，默认为SKIP

实际使用不建议任何人用REPLACE和TRUNCATE选项！！太多人因为这2个选项把数据删掉又覆盖掉了！！

对于确实有覆盖需求的，建议自己手动做drop等操做。



## expdp/impdp简单使用

expdp/impdp可以高效地排除某些对象，会指定某些对象。export/import不具备这些功能：

仅导出表对象

```
expdp scott/tiger include=table schemas=scott directory=s01 dumpfile=scott_tables.oraexpdp
```

仅导出索引对象

```
expdp scott/tiger include=index schemas=scott directory=s01 dumpfile=scott_indexes.oraexpdp
```

仅排除某些表

```
expdp scott/tiger schemas=scott directory=s01 dumpfile=scott_tables.oraexpdp exclude=table:\"LIKE 'EMP%'\\"
```

```
expdp scott/tiger schemas=scott directory=s01 dumpfile=scott_tables.oraexpdp EXCLUDE= table:\"IN \\('NAME', 'ADDRESS', 'EMPLOYEE', 'DEPT')\"
```



## 历史

- Oracle 10g以前使用export/import传统工具组导入导出数据
- Oracle 10g以后推出了新的数据导入导出数据泵工具组expdp/impdp
- 以上两者的格式不互相兼容
- Oracle 10g以后oracle官方态度仅推荐使用数据泵工具组expdp/impdp
- oracle官方态度不会再改进export/import传统工具组
- 部分新型数据类型不支持使用export/import工具组导出



## 数据泵工具组expdp/impdp解决了以下问题：

- Oracle 10g以前使用export/import传统工具组导入导出数据
- Oracle 10g以后推出了新的数据导入导出数据泵工具组expdp/impdp
- 以上两者的格式不互相兼容！！
- expdp/impdp工具组对export/import传统工具组的改进如下：
  - expdp/impdp数据泵工具组的设计结构更合理，具体基本的检验。export/import传统工具组的原始设计包括二进制数据格式过于粗糙
  - expdp/impdp数据泵工具组的速度快好几倍
  - expdp/impdp数据泵工具组支持并行PARALLEL导入导出
  - expdp/impdp数据泵工具组在11g以后支持原生压缩
  - expdp/impdp数据泵工具组支持dblink
  - expdp/impdp数据泵工具组支持指定排除某些对象

对普通用户而言，没有特殊理由的话，都建议使用expdp/impdp。你的员工还在使用export/import传统工具组？让他别那么懒，学习下已经出来快20年的“新技术”！！







## 数据泵工具组expdp/impdp的兼容性

- 可以使用version参数指定expdp生成的文件的兼容性

### VERSION

要导出的对象版本。

有效的关键字值为: [COMPATIBLE], LATEST 或任何有效的数据库版本。

例如11.2.0.4的数据库要导出数据到10.2.0.1, 则在11.2.0.4数据库服务器上执行:

```
expdp directory=s01 dumpfile=pd.oraexpdp schemas=pd logfile=pd_exp.log version=10.2.0.1
```



## 数据泵工具组expdp/impdp常见使用错误

- 注意expdp/impdp是纯服务器端的技术。这意味着它导出文件只能导出在数据库所在服务器上，它导入的文件也必须放在要导入它数据库的服务器上。
- export/import传统工具组是不同的，是server-client模型的。传统工具组可以通过网络在你的pc个人电脑上用exp导出数据库服务器上的数据直接到你的笔记本上。也可以通过网络把你pc上的一个export文件用imp导入到数据库服务器上。

例如在你的pc上执行 `expdp system/oracle@orcl directory=s01 dumpfile=full.dmp full=y`  
以上命令会在orcl数据库所在服务器上检查是否有s01这个目录对象，如果有的话则在该目录下生成文件

同理在你的pc上执行 `impdp system/oracle@orcl directory=s01 dumpfile=full.dmp full=y`  
也是在orcl数据库所在服务器上检查是否有s01这个目录对象，如果有的话则对该目录s01下的full.dmp文件执行全导入操作



## 数据泵工具组expdp/impdp常见使用错误

一般expdp/impdp都是在数据库服务器上执行，所以一般不用system/oracle@orcl 这样远程连接，同时这样远程连接要求ORACLE LISTENER监听正常，服务正常注册到监听。如果没启动监听就无法正常工作了。

在数据库服务器上执行，直接指定ORACLE\_SID即可

```
set ORACLE_SID=orcl //在Windows上  
export ORACLE_SID=orcl //在Linux/Unix上
```

```
expdp system/oracle directory=s01 dumpfile=full.dmp full=y
```



## 常见使用错误

```
expdp system/oracle directory=s01 dumpfile=full.dmp full=y;
```

```
exp system/oracle file=full.dmp buffer=9000000 full=y;
```

这些都是命令行程序，不是SQL脚本。最后不需要加分号！！分号可能引起转义



# 常见使用错误IMP-00009 IMP-00037

```
IMP-00009 abnormal end of export file
// *Cause: This is usually caused by an export file generated by an aborted
// export session.
// *Action: If the export file was generated by an aborted export session,
// retry the export followed by the import. Else, report this
// as a bug and submit the export file that causes this error.
```

```
IMP-00030 failed to create file %s for write
// *Cause: Import was unable to create the specified file with write enabled
// *Action: Check the file name and file system for the source of the error
```

```
IMP-00037 Character set marker unknown
// *Cause: export file is corrupted.
// *Action: Try to obtain an uncorrupted version of the export file.
```

```
IMP-00009: 导出文件异常结束
IMP-00020: 用于列缓冲区大小 (7) 的长整数列过大
IMP-00018: 上一个表的部分导入已完成: 已导入 行
```

```
IMP-00037: 未知字符集标记
IMP-00000: 未成功终止导入
```

IMP-00009: 导出文件异常结束 IMP-00037 未知字符集标记

以上错误一般说明**export**导出的文件存在损坏，要么重新导出。如已丢失原库数据，诗檀软件可以提供收费恢复服务。



# 常见使用错误 ORA-31619

ORA-39001: invalid argument value

ORA-39000: bad dump file specification

ORA-31619: invalid dump file "\*.dmp"

ORA-39001: 参数值无效

ORA-39000: 转储文件说明错误

ORA-31619: 转储文件 "\*.dmp" 无效

可能的原因:

- expdp导出时的版本和impdp不匹配，应当指定对应的version
- expdp导出的文件损坏

如果expdp导出的文件存在损坏，要么重新导出。如已丢失原库数据，诗檀软件可以提供收费恢复服务。



# OraDumpReader

OraDumpReader产品是诗檀软件公司开发的export和expdp导出文件读取工具，可直接读取dump文件中的数据。对于dump文件损坏和被部分加密等情况均可恢复其中数据。

	REFCODE	MARCCODE	PROPCODE	BARCODE	PRICE	DISPRICE	SEARCHCODE	ANNEX	BRANCHID
1	88	80		C0003814	68.00	68.00	1565.44/195-2		1
2	88	80		C0003814	68.00	68.00	1565.44/195-2		1
3	717141	2053730		C0821494	32	32	1313.45/564		1
4	881563	3208435		C0882973	39.8	39.8	1267/2232		1
5	883738	2056934		C0861885	39.8	39.8	1313.45/719(2)		1
6	753584	7047751		C0772915	35	35	1313.45/356		1
7	776449	7047829		C0773149	46.8	46.8	1267/1810-4		1
8	648786	2047120		C0678388	25	25	1247.57/3401		1
9	715964	3018971		C0817110	38	38	1266/330		1
1	425050	2030347		C0480051	20.5	20.5	1516.44/8		1
1	668658	6007136		C0699715	29.9	29.9	1712.45/561-8		2
1	780970	7048104		C0773973	39	39	1313.43/3-1		1
1	332960	2008726		C0450151	38	38	1247.57/560		1
1	779726	7048133		C0774059	35	35	1313.45/364		1
1	820796	3101120		C0815170	29.8	29.8	1267.1/663		1
1	3470	29075		C0046008	2.5	2.5	1247.57/3146		2
1	712969	3200300		C0817843	27.8	27.8	1222.844/29		1
1	880915	2056569		C0860013	45	45	H319.32/5		2



专注于数据

**PARNASSUSDATA**

软件，方案，服务供应商



**ParnassusData**<sup>TM</sup>