

oracle10gflashbacktable PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/465/2021_2022_oracle10gf_c67_465179.htm flashback table与9i的flashback query相似，利

用undo信息来恢复一个或者一些表到现在以前的一个时间点（一个快照）。Undo相关参数如下，需要确保AUM与足够的retention值。 SQL> show parameter undo

```
NAME TYPE VALUE-----
```

```
undo_management string AUTO undo_retention integer 900
```

```
_tablespace string UNDOTBS1
```

首先要说明的是，flashback table不等于flashback query，所谓query，仅仅是查询以前的一个快照点而已，并不改变当前表的状态，而 flashback table不一样，将改变当前表及附属对象一起回到以前的时间点。其实9i

的flashback query在10g中也有了新的变化，10g中可以简单的利用以下语句实现flashback query，而不再需要象9i那样需要调用 DBMS_FLASHBACK包。 SELECT * FROM TABLENAME

```
AS OF TIMESTAMP TO_TIMESTAMP(2003-04-04 09:30:00,
YYYY-MM-DD HH:MI:SS) WHERE .....
```

10g的flashback table有如下特性 在线操作 恢复到指定时间点或者SCN的任何数据. 自动恢复相关属性，如索引，触发器等 满足分布式的一致性 满足数据一致性，所有相关对象将自动一致 语法为：

```
TABLE tablename TO TIMESTAMP (JUL-07-2003, 02:33:00)
```

```
TABLE employee TO SCN 123456. FLASHBACK TABLE t1 TO
TIMESTAMP 2003-03-03 12:05:00 ENABLE TRIGGERS.
```

中ENABLE TRIGGERS表示触发器恢复之后为enable状态，而默认为disable状态。 注意：如果需要flashback一个表，需要保

证 需要有flashback any table的系统权限或者是该表的flashback对象权限。 需要有该表的SELECT, INSERT, DELETE, ALTER 权限 必须保证该表ROW MOVEMENT 下面，我们用一个详细的例子来说明这个过程：

```
16:16:51 SQL> create user flash
identified by flash. User created.
16:17:04 SQL> grant
connect,resource to flash. Grant succeeded.
16:17:19 SQL> connect
flash/flash. Connected.
16:26:35 SQL> create table t1 as 0select *
from all_objects. Table created.
16:37:24 SQL> create table t2 as
0select * from t1. Table created.
16:37:35 SQL> 0select count(*)
from t1. COUNT(*)----- 38949
16:37:43 SQL> 0select
count(*) from t2. COUNT(*)----- 38949
16:38:06 SQL> create
index inx_test1 on T1 (object_name). Index created.
16:39:55 SQL>
create index inx_test2 on T1 (object_id). Index created.
16:40:47
SQL> 0select to_char(sysdate,yyyymmdd hh24:mi:ss) from dual.
TO_CHAR(SYSDATE,YY----- 2004-04-06
16:41:18
以上获得一个时间戳，假定我们要恢复该表到这个时间，那么以下对该表的操作都将被前滚。
```

```
16:41:18 SQL> 0drop index
inx_test1. Index 0dropped.
16:41:33 SQL> 0delete from t1. 38949
rows 0deleted.
16:41:46 SQL> commit. Commit complete.
16:41:49
SQL> truncate table t2. Table truncated.
在以上的操作中，我们0delete一个表，然后truncate一个表，下面，我们将来看看flashback table的效果，能恢复到什么程度
```

```
16:37:55 SQL>
ALTER TABLE t1 ENABLE ROW MOVEMENT. Table altered.
16:38:03 SQL> ALTER TABLE t2 ENABLE ROW MOVEMENT.
Table altered.
16:43:10 SQL> flashback table t1 TO TIMESTAMP
to_timestamp(2004-04-06 16:41:18,yyyymmdd hh24:mi:ss).
```

```
Flashback complete. 16:43:49 SQL> flashback table t2 TO
TIMESTAMP to_timestamp(2004-04-06 16:41:18,yyyy-mm-dd
hh24:mi:ss). flashback table t2 TO TIMESTAMP
to_timestamp(2004-04-06 16:41:18,yyyy-mm-dd hh24:mi:ss)*
ERROR at line 1: ORA-01466: unable to read data - table definition
has changed 我们可以发现，执行delete操作的表是可以恢复
的，而执行truncate操作的表是不可以恢复的，这正好也说明
了flashback table利用undo的结论。看看我们的结果：SQL>
select count(*) from t1. COUNT(*)----- 38949 SQL>
select count(*) from t2. COUNT(*)----- 0 SQL>
select
t.index_name from user_indexes t where t.table_name=T1.
INDEX_NAME----- INX_TEST2 还可以
看到，对于drop的索引，也是没有办法恢复的，因为drop
并不记录undo，所以所谓索引的恢复，仅仅是相关索引树的
改变而不能找回删除掉的索引。 100Test 下载频道开通，各类
考试题目直接下载。详细请访问 www.100test.com
```