**概述：**

　　闪回技术是Oracle强大数据库备份恢复机制的一部分，在数据库发生逻辑错误的时候，闪回技术能提供快速且最小损失的恢复（多数闪回功能都能在数据库联机状态下完成）。需要注意的是，**闪回技术旨在快速恢复逻辑错误，对于物理损坏或是介质丢失的错误，闪回技术就回天乏术了**，还是得借助于Oracle一些高级的备份恢复工具如RAMN去完成（这才是Oracle强大备份恢复机制的精髓所在啊）

**撤销段（UNDO SEGMENT）**

　　在讲闪回技术前，需要先了解Oracle中一个逻辑结构--**撤销段**。因为大部分闪回技术都需要依赖撤销段中的撤销数据。撤销数据是反转DML语句结果所需的信息，只要某个事务修改了数据，那么更新前的原有数据就会被写入一个撤销段。（事务回滚也会用到撤销段中的数据）。事务启动时，Oracle 会为其分配一个撤销段，事务和撤销段存在多对一的关系，即一个事务只能对应一个撤销段，多个事务可以共享一个撤销段（不过在数据库正常运行时一般不会发生这种情况）。

**闪回技术**

　　Oracle提供了四种可供使用的闪回技术（闪回查询，闪回删除，闪回归档，闪回数据库），每种都有不同的底层体系结构支撑，但其实这四种不同的闪回技术部分功能是有重叠的，使用时也需要根据实际场景合理选择最合适的闪回功能。

**闪回查询(Flashback Query)**

　　　　a.**基本闪回查询**

　　　　功能描述：可以查询过去某个时间段的数据库状态。

　　　　工作原理：Oracle 会提取所需要的撤销数据（前提是撤销是可用的，即撤销数据还没被覆盖）进行回滚，但这种回滚是临时的，仅针对当前session可见。

SQL> select \* from dept as of timestamp to\_timestamp('2016-09-10 11:00:00','yyyy-mm-dd hh24:mi:ss');

　　　　b.**闪回表**

　　　　功能描述：可将某个表回退到过去某个时间点

　　　　工作原理：同样，Oracle会先去查询撤销段，提取过去某个时间点之后的所有变更，构造反转这些变更的SQL语句进行回退,闪回操作是一个单独的事务，所以若由于撤销数据过期之类的原因导致无法闪回，整个操作会回滚，不会存在不一致的状态。

　　　　步骤：

　　　　1.启用表闪回首先要在表上支持行移动（在数据字典中设置标识来标识该操作可能会改变行ID，即同一条数据闪回成功后主键都一样，但行ID其实已经发生变化了）

SQL> alter table emp enable row movement;

　　　　2.闪回表操作

SQL> flashback table dept to timestamp to\_timestamp('2016-09-10 11:00:00','yyyy-mm-dd hh24:mi:ss');

　　　　闪回表可能会失败，有可能有以下几种情况：

*违反了数据库约束，*比如用户不小心删除了子表中的数据，现在想利用闪回表技术进行回退，恰好在这中间，父表中与该数据对应的那条记录也被删除了，在这种情况下，由于违反了外键约束，导致闪回表操作失败了；

*撤销数据失效*，比如用于支撑闪回操作的撤销数据被覆盖了，这种情况闪回表操作自然会失败；

*闪回不能跨越DDL*，即在闪回点和当前点之间，表结构有过变更，这种情况闪回操作也会失败。

*注意：上述闪回功能都是基于撤销数据的，而撤销数据是会被重写的（Expired会被重写，Active不会被重写），所以，在需要使用这几种闪回功能去恢复数据的时候（确切地说，是需要使用基于撤销数据的闪回功能时），最短时间发现错误，第一时间执行闪回操作，才能最大程度地保证闪回功能的成功。*

**闪回删除（Flashback Drop）**

　　功能描述：闪回删除可以轻松将一个已经被Drop的表还原回来。相应的索引，数据库约束也会被还原（除了外键约束）

　　原理描述：Drop命令其实是Rename命令，早期的Oracle版本（10g之前），闪回删除意味着从数据字典中删除了该表的所有引用，虽然表中数据可能还存在，但已成了孤魂野鬼，没法进行恢复了，10g版本之后，Drop命令则仅仅是一个Rename操作，所以恢复就很容易了。

　　闪回删除操作执行命令很简单

SQL> flashback table emp to before

　　如果要还原的表名在当前系统中已经被占用，也可以在闪回删除的时候对表重命名

SQL> flashback table emp to before drop rename to emp\_new

　　也可以通过回收站查看当前用户那些表被删除了，每个用户都有一个回收站，这个回收站是个逻辑结构，它不是一块独立的存储空间，它存在在当前表空间内，所以如果有别的操作需要空间，比如现在需要创建一张表，没有足够空间可用，回收站中的数据就会被清理，这也是导致闪回删除失败的原因。

SQL> SHOW RECYCLEBIN;

　　彻底删除表，闪回删除也无能为力

SQL> DROP TABLE EMP PURGE;

　　清空回收站

SQL> PURGE RECYCLEBIN;

*注意：闪回删除只针对Drop命令，注意区分truncate操作和drop操作，truncate称为表截断，会清空表中数据（调节Oracle高水位线实现），表结构不受影响，速度很快，弊端是此过程不会产生任何撤销数据或是重做日志，如果误删，恢复异常麻烦，要慎重使用。而Drop则会删除数据+表结构，闪回删除仅针对Drop操作。*

**闪回数据归档（Flashback Data Archive ）**

　　　功能描述：闪回数据归档可使表具有回退到过去任何时间点的能力，前面提到的闪回查询，闪回表都会受限于撤销数据是否失效，如果撤销数据被覆盖重写了，闪回操作自然会失败，闪回删除则受限于表空间是否有足够可用空间，而闪回数据归档，则没有这些限制。

　　　创建闪回归档

　　　1.创建一个用户闪回数据归档的表空间，当然，也可以使用已经存在的表空间。

　SQL> create tablespace test\_tb datafile 'test.dbf' size 20m;

　　　2.创建一个保留时间为2年的闪回归档

SQL> create flashback archive test\_fa tablespace test\_tb retention **2** year;

　　  为scott用户下的emp表启用闪回归档

　　　1.赋予用户归档的权限

　SQL> grant flashback archive on test\_fa to scott;

　　  2.连接用户

SQL> conn scott/tiger;

　　  3.为emp表启用闪回归档

SQL> alter table emp flashback archive test\_fa;

　　至此，emp表就拥有了可以查询或回退到过去2年任意时间点的能力！

**闪回数据库（Flashback Database）**

　　　功能描述：闪回数据库可将整个数据库回退到过去某个时间点，闪回表是某张表的时空穿梭，闪回数据库则是整个数据库的时空穿梭。当然，闪回点之后的所有工作就丢失了，其实就相当于数据库的不完整恢复，所以只能以resetlogs模式打开数据库。闪回数据库会造成停机时间，当然相比于传统备份恢复机制，恢复过程会快很多。

　　　工作原理：闪回数据库不使用撤销数据，使用另外一种机制来保留回退所需要的恢复数据，当启用闪回数据库，发生变化的数据块会不断从数据库缓冲区缓存中复制到闪回缓冲区，然后，称为恢复写入器（Recovery Writer）的后台进程会将这些数据刷新到磁盘中的闪回日志文件中。闪回的过程，则是一个 提取闪回日志-->将块映像复制回数据文件 的过程。

　　　配置闪回数据库（闪回数据库要求数据库为归档模式）

　　　　1.指定闪回恢复区，也就是存放闪回日志的位置，但闪回恢复区不仅仅是为了存放闪回日志，Oracle的很多备份恢复技术都用到这个区域，比如控制文件的自动备份等都会存放到此区域。

SQL> alter system set db\_recovery\_file\_dest ='/flash\_recovery\_area';

　　　　 2.指定恢复区大小

SQL> alter system set db\_recovery\_file\_dest\_size=4G;

　　　　3.指定闪回日志保留时间为2小时，即通过闪回操作，可以将数据库回退到前两小时内的任意时间点

SQL> alter system set db\_flashback\_retention\_target=**120**;

　　　　4.有序关闭数据库--mount模式下启用闪回数据库--打开数据库

[复制代码](javascript:void(0);)

SQL> shutdown immediate;

　　　SQL> startup mount;

　　　SQL> alter database flashback on;

　　　SQL> alter database open;

[复制代码](javascript:void(0);)

　　至此，闪回数据库配置完成！

　　　使用闪回数据库功能

[复制代码](javascript:void(0);)

SQL> shutdown immediate;

　　　SQL> startup mount;

　　　SQL> flashback database to timestamp sysdate-**60**/**1440**;

　　　SQL> alter database open resetlogs;

[复制代码](javascript:void(0);)

**总结**

本文列举了四类闪回技术，其中，闪回查询，包括基本闪回查询，闪回表等技术都依赖于撤销数据（还有一类闪回技术为闪回事务，可以对指定事务进行闪回操作，原理类似，借助于撤销数据来构建用于反转事务的SQL语句），依赖于撤销数据，则自然受限于撤销数据的保留时间，可能会由于撤销数据被覆写而导致闪回失败。闪回删除，则是由于10g版本后对表的删除仅表现为一个rename操作，引入回收站的概念，但此回收站仅是当前表空间的一块逻辑划分，所以会受限于当前表空间的可用空间的限制；闪回归档可提供查询或回退到过去任意时间点的功能，闪回数据库则是一中更极端的数据库恢复功能，相当于不完整恢复，依赖于闪回日志。

转自https://www.cnblogs.com/j839035067/p/6582711.html