

1 数据库备份

数据库备份分为逻辑备份和物理备份。逻辑备份用于备份独立对象，如表或者模式；而物理备份则备份文件系统级别的对象粒度，如构成表空间的物理文件。无论数据库处于联机模式还是脱机模式，均可执行数据库的物理备份。

1.1 逻辑备份和物理备份

逻辑备份导出数据库的结构以及数据，这些结构包括表的定义、触发器和存储过程等数据库对象，当使用数据泵技术或EXP/IMP技术时，实现的是逻辑备份。

物理备份是将数据库的数据文件、控制文件和归档日志文件的重要文件复制到操作系统的其他磁盘，此时的文件保持原文件的类型。

1.2 脱机备份和联机备份

脱机备份指在数据库关闭情况下的备份，也称为冷备份；而联机备份是数据库运行时进行的数据库备份，也叫做热备份。采用联机备份还是脱机备份依赖于业务的需求，对于7*24运行的数据库显然不能使用脱机备份，但是联机备份相对复杂，必须考虑数据库的归档模式，以及设计合理的联机备份方案。

执行物理备份时，需要复制构成数据库的文件，这些备份也称为“文件系统备份”，因为它们涉及使用操作系统文件备份命令。Oracle支持两种不同的物理文件备份，分为脱机备份和联机备份（也分别称为冷备份和热备份）。

1.2.1 脱机备份/冷备份

通过SHUTDOWN命令的NORMAL、IMMEDIATE或者TRANSACTIONAL选项使数据库正常关闭，将执行一致的脱机备份，备份以下文件：

- 所有数据文件
- 所有控制文件
- 所有归档重做日志文件
- init.ora文件或者服务器参数文件SPFILE

- 文本格式文件，如密码文件和tnsnames.ora

如果在数据库发生异常终止后执行脱机备份，将会出现不一致情形，而且在恢复期间可能需要做更多的工作。

1.2.2 联机备份/热备份

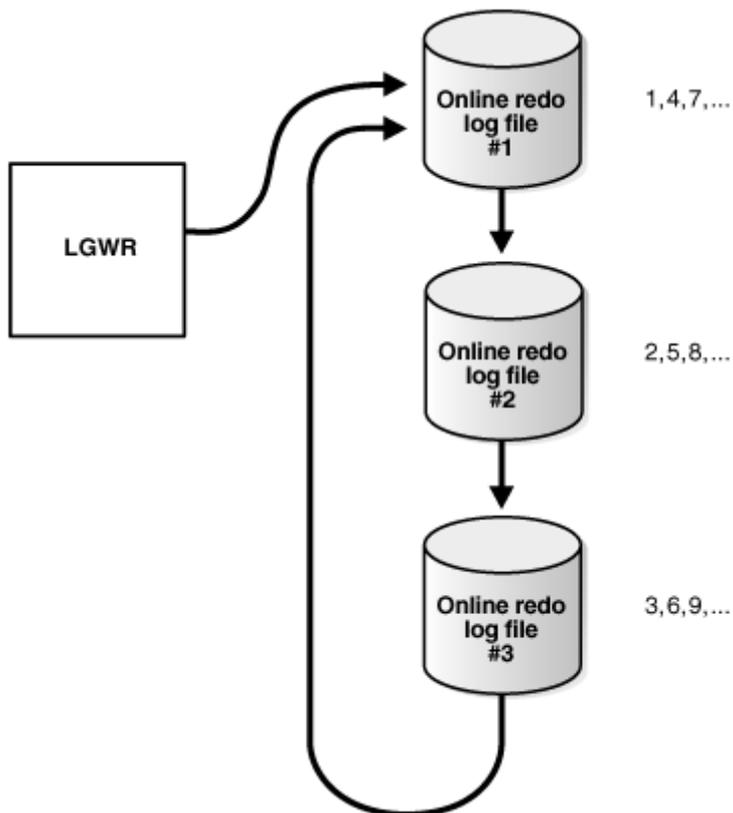
数据库在联机状态下进行的备份，联机备份过程是个很强大的功能，原因有两点，首先，它们提供了完全时间点恢复，第二，允许数据库在备份文件系统时仍然处于打开状态。通过数据库一直处于打开状态，也可以避免在关闭和重启数据库时清除数据库实例的系统全局区（SGA），由于避免了清除SGA内存，从而减少了重启数据库时需要执行的物理I/O数量，数据库性能将得到提高。

可以在数据库打开时备份以下文件：

- 所有数据文件
- 所有归档重做日志文件
- 使用ALTER DATABASE BACKUP CONTROLFILE命令备份的一个控制文件
- SPFILE

每当备份整个数据库或者SYSTEM表空间时，RMAN将自动备份控制文件和SPFILE。

Oracle以循环方式写入联机重做日志文件：在填满第一个日志文件后，它开始写入第二个日志文件，在填满第二个日志文件后，它开始写入第三个日志文件，以此类推。在填满最后一个联机重做日志文件后，LGWR后台进程开始覆盖第一个重做日志文件的内容。如下图：



当Oracle以ARCHIVELOG模式运行时，ARCn后台进程会在覆盖每个重做日志文件前进行归档，从而形成归档的重做日志文件，通常将其写入磁盘文件或者磁带设备中，用来恢复数据。大多数生产数据库，特别是OLTP生产数据库，必须以ARCHIVELOG模式运行，为了使用RMAN，数据库必须处于ARCHIVELOG模式。

数据字典：

```

1 SELECT * FROM v$instance;
2 SELECT * FROM v$database;
3 SELECT * FROM Db_Data_Files;
4 SELECT * FROM v$datafile;
5 SELECT * FROM v$tempfile;
6 SELECT * FROM v$logfile;
7 SELECT * FROM v$controlfile;

```

1.3 一致备份和非一致备份

先解释恢复的概念，恢复（Recover）是因为数据文件和控制文件中的系统SCN不一致，恢复进程必须使用归档日志文件和联机重做日志文件的数据更新数据文件中的内容，也就是将重做日志文件中用户提交的数据重新写入数据文件。一致性，Oracle为每个事务设置了一个唯一的SCN，当每次事务提交时都自动增加SCN号，这个号码永远是唯一的，当DBWR写进程运行时，将触发一个检查点事件，将数据库缓冲区中所有已经提交的数据写入磁盘，

并使得所有数据文件和控制文件中的SCN一致。而当LGWR将数据库缓冲区中变化的数据写入重做日志文件时，对于用户提交了数据的事务的SCN将记录在控制文件中，注意此时的数据文件中的SCN没有变化，这叫做不一致状态。

所以一致备份和不一致备份的区别就是是否需要恢复。要实现一致备份可以关闭数据库使用脱机备份的方式，也可以使数据库处于Mount状态，使用RMAN实现。在7*24小时运行的数据库中，不一致备份是唯一的选择，它并不代表这样的备份是不可靠的，只是在数据恢复时需要一个“一致”的过程，只要数据库处于归档模式，且重做日志归档文件没有损坏就可以使用不一致备份实现数据库的完全恢复，不会造成数据的丢失。

2 数据库恢复

2.1 实例恢复

数据库实例包括内存结构和后台进程。实例故障是指实例突然失败造成的数据库故障，如使用SHUTDOWN ABORT关闭数据库，此时启动数据库会使用当前的数据文件和重做日志文件执行实例恢复。实例恢复由数据库自动完成，不需要用户干预。

实例恢复由两个步骤，前滚 (ROLL FORWARD) 和回滚 (ROLLBACK)。前滚是将重做日志文件中记录的用户提交和未提交的事务涉及的数据写入数据文件；回滚是将未提交事务涉及的修改通过UNDO中的记录退回到之前的状态，以保证事务的原子性。

2.2 介质恢复

介质恢复是由于磁盘损坏或者数据文件损坏而需要的数据库恢复方式，需要用户干预来执行整个介质恢复过程。介质恢复必须要有备份的数据文件，如果需要完全恢复，还需要自备份以来的归档重做日志文件和当前的重做日志文件。

介质恢复需要两个步骤，即还原 (RESTORE) 和恢复 (RECOVER)。还原是将备份的数据文件复制到相应的目录，如果目录的位置改变，则需要使用ALTER DATABASE RENAME DATAFILE指令告诉数据库新的数据文件的位置；恢复是指使用归档重做日志文件和当前的重做日志文件恢复数据库到最新的状态，即进行实例恢复。

2.3 完全和不完全恢复

完全恢复是指当数据库发生故障后，通过备份将数据库恢复到最新的状态，不丢失任何数据。需要数据库有数据文件的备份，而且有备份以来的全部的归档重做日志和当前的重做日志，这样才不会丢失数据，可以不同粒度的实现完全恢复，如数据库级、表空间级和数据文件级；

不完全恢复不能把数据库恢复到最新状态，会有数据丢失。