

Oracle8i中回滚段使用和ORA-1555 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/143/2021_2022_Oracle8i_E4_B8_c102_143587.htm

回滚段中保存的回滚数据有2个作用：一致读和回滚。回滚段是由连续block组成的区间extent组成.回滚段有顺序的循环的使用这些区间。当当前区间写满的时候，oracle移向下一个区间. 如一个回滚段有4个区间A,B,C,D.当区间C被写满的时候，oracle将写区间D,而当D写满的时候，oracle将尝试重新写区间A.这样循环的有顺序的使用区间。事务必须将回滚信息写到回滚段中；事务的当前新产生的回滚信息写在该回滚段的位置叫做回滚段的head. 而在该回滚段上最早发生的尚未提交的事务最早产生的回滚信息所在位置叫做回滚段的tail. 当前区间写满的时候，oracle移动head到下一个区间. 8i时每个事务只能使用一个回滚段。Oracle会根据回滚段workload，平均将事务分配给各个回滚段。在回滚段使用上的一些规则

1. 一个事务只能使用一个回滚段。
2. 多个事务可以共用一个区间。但Active的事务不能共用一个block.
3. 回滚段的current extent写满的时候，回滚段的Head不能够移动到回滚段tail所在的区间。
4. 区间总是被有顺序的循环的时候；当head移动的时候，不会跳跃区间；只能移动到下一个区间。
5. 如果head不能够使用下一个区间（如tail在下一个区间），将会分配一个新的区间extent，并将新区间extent插入到这个循环使用的extent圈中。这叫做回滚段的扩展。

ORA-1555 snapshot too old主要是在一致读和延迟块清除delay block cleanout的时候产生；[参考]一致读的步骤

1. Read the Data Block.
2. Read the Row Header.
3. Check the Lock Byte to

determine whether there's an ITL entry. 4. Read the ITL entry to determine the Transaction ID (Xid). 5. Read the Transaction Table using the Transaction ID. If the transaction has been committed and has a System Commit Number less than the query's System Change Number, 0 update the status of the block (block cleanout) and start over at step 1. 第5步细分 ---IF 在Transaction Table 中根据Transaction ID 找到transaction -----IF transaction 已经commit -----IF query scn > commit scn -----则接受该块，进行clean out，返回1 -----ELSE IF query scn -----则进行一致性读，从第6步向后执行 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