

oracle认证:浅谈Oracle中的三种Join方法Oracle认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/583/2021_2022_oracle_E8_AE_A4_E8_c102_583136.htm 基本概念 Nested loop join：Outer table中的每一行与inner table中的相应记录join，类似一个嵌套的循环。 Sort merge join：将两个表排序，然后再进行join。 Hash join：将两个表中较小的一个在内存中构造一个Hash表（对Join Key），扫描另一个表，同样对Join Key进行Hash后探测是否可以join，找出与之匹配的行。 一张小表被hash在内存中。因为数据量小，所以这张小表的大多数数据已经驻入在内存中，剩下的少量数据被放置在临时表空间中；每读取大表的一条记录，就和小表中内存中的数据进行比较，如果符合，则立即输出数据（也就是说没有读取临时表空间中的小表的数据）。而如果大表的数据与小表中临时表空间的数据相符合，则不直接输出，而是也被存储临时表空间中。当大表的所有数据都读取完毕，将临时表空间中的数据以其输出。如果小表的数据量足够小（小于hash area size），那所有数据就都在内存中了，可以避免对临时表空间的读写。如果是并行环境下，前面中的第2步就变成如下了：每读取一条大表的记录，和内存中小表的数据比较，如果符合先做join，而不直接输出，直到整张大表数据读取完毕。如果内存足够，Join好的数据就保存在内存中。否则，就保存在临时表空间中。 适用范围 Nested loop join: 适用于outer table(有的地方叫Master table)的记录集比较少（<10000）而且inner table（有的地方叫Detail table）索引选择性较好的情况下（inner table要有index）。 inner table被outer table驱动，outer table返回的

每一行都要在inner table中检索到与之匹配的行。当然也可以用ORDERED 提示来改变CBO默认的驱动表，使用USE_NL(table_name1 table_name2)可是强制CBO 执行嵌套循环连接。 $\text{cost} = \text{outer access cost} (\text{inner access cost} * \text{outer cardinality})$ Sort merge join：用在数据没有索引但是已经排序的情况下。通常情况下hash join的效果都比Sort merge join要好，然而如果行源已经被排过序，在执行排序合并连接时不需要再排序了，这时Sort merge join的性能会优于hash join。可以使用USE_MERGE(table_name1 table_name2)来强制使用Sort merge join。 $\text{cost} = (\text{outer access cost} * \# \text{ of hash partitions}) \text{ inner access cost}$ Hash join：适用于两个表的数据量差别很大。但需要注意的是：如果HASH表太大，无法一次构造在内存中，则分成若干个partition，写入磁盘的temporary segment，则会多一个I/O的代价，会降低效率，此时需要有较大的temporary segment从而尽量提高I/O的性能。可以用USE_HASH(table_name1 table_name2)提示来强制使用散列连接。如果使用散列连HASH_AREA_SIZE 初始化参数必须足够的大，如果是9i，Oracle建议使用SQL工作区自动管理，设置WORKAREA_SIZE_POLICY 为AUTO，然后调整PGA_AGGREGATE_TARGET 即可。也可以使用HASH_JOIN_ENABLED=FALSE（默认为TRUE）强制不使用hash join。 $\text{cost} = (\text{outer access cost} * \# \text{ of hash partitions}) \text{ inner access cost}$ 效率比较 Hash join的主要资源消耗在于CPU（在内存中创建临时的hash表，并进行hash计算），而merge join的资源消耗主要在于磁盘I/O（扫描表或索引）。在并行系统中，hash join对CPU的消耗更加明显。所以在CPU紧张时，最好

限制使用hash join。在绝大多数情况下，hash join效率比其他join方式效率更高：在Sort-Merge Join(SMJ)，两张表的数据都需要先做排序，然后做merge。因此效率相对最差；Nested-Loop Join(NL)效率比SMJ更高。特别是当驱动表的数据量很大（集的势高）时。这样可以并行扫描内表。Hash join效率最高，因为只要对两张表扫描一次。更多优质资料尽在百考试题论坛 百考试题在线题库 oracle/认证更多详细资料 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