

Oracle11g新特性:更加灵活的分区策略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_ti2020/644/2021_2022_Oracle11g_E6_c102_644994.htm "mkhghigh"> 前不久，曾经接手一个性能调优

案例：这是一个报表系统，其基础数据主要存储于三张表中。表的大小已经很大了，最大一张接近100G。在生成报表时需要长时间才能返回结果，一些online查询甚至经常timeout。表中存储的是2万多个公司的数据，报表的生成也是以公司为单位的，因此，这一调优方案的思路比较明确：将表按公司分区。但是，这中间却存在一些麻烦：每个公司的数据并不是均衡的。其中近200家公司属于VIP用户，他们的数据量最大，每个公司差不多是十几万到几十万的数据量，其总量占了全部数据的30%左右.而其它非VIP用户的数据基本上每个都在1万以内。而我们的主要目标就是要优先保证VIP用户获取到最佳的性能(由于其数据量，当前最大的性能问题恰恰就出在这些VIP用户上)。因此，我们提出了2中分区方案：基于Company Id的Hash分区. 基于Company Id的List分区. 但是，这两种方案各有优缺点：对于Hash分区，分区的大小更加均衡，因而性能也更加均衡。但是，可能出现一些无法控制的极端现象：Hash分区仅仅是对Company Id使用Hash函数进行分组，它能做到每个分区分配基本相当数量的Company Id，但是每个Company Id对应的数据量并不考虑在内，因此可能出现某些分区集中的都是VIP数据或者都是非VIP数据，造成分区过大或过小.另外一个缺点就是我们很难直接干预某个公司的性能。例如，可能有某个非VIP用户成为了VIP用户，其数据量激增，它又正好处于一个大的分区上，这时，我们很

难将其从这个分区剥离出来，除非它所在分区正好出在一个即将分裂的分区上。对于List分区，VIP用户的性能能够得到保证。我们可以将每个VIP用户单独存储在一个分区上，但是，不可能将非VIP用户单独存储开(不仅增加维护难度，且增加整个表的大小)，只能将非VIP用户存储在几个分区上。但是这样还是造成DDL语句非常复杂，并且非VIP的分区很大(每个都在10G左右，而VIP分区最大才200M)。由于List分区更加接近我们的优化目的，最终还是采用了List分区。其实，期间我们曾经考虑过使用复合分区。在10g中(我们的生产库是10g)，仅支持2种复合分区：Range-List和Range-Hash。我们的解决方案是：为表增加一个数字类型的ID字段，VIP用户对应的数字大于100,000，非VIP用户的ID小于100,000。每个VIP用户被单独放置在一个Range分区中，所有非VIP用户被放置在一个Range分区中，然后再对非VIP分区通过Hash划分子分区。这样，即能保证VIP用户的性能，也能均衡非VIP用户的性能。但是，由于这种方案需要增加一个非业务的字段，以及其它一些原因，最终被否决了。到11g中，Oracle的分区策略更加灵活了。首先，11g支持更多方式的组合分区，除10g支持的两种之外，还支持Range-Range、List-Range、List-List、List-Hash的组合分区策略。对于我们上述这个案例，就可以通过List-Hash的组合分区来解决。以下就是一个List-Hash分区的演示：

```
SQL> analyze table par_test compute statistics.
Table analyzed.
SQL> select partition_name, subpartition_name, num_rows, blocks from dba_tab_subpartitions
2 where table_name = PAR_TEST.
PARTITION_NAME SUBPARTITION_NAME NUM_ROWS BLOCKS
```

----- P2 SYS_SUBP154 26604 434 DEF

SYS_SUBP158 4529 70 DEF SYS_SUBP157 2783 45 DEF

SYS_SUBP156 2422 39 DEF SYS_SUBP155 2854 47 P1

SYS_SUBP153 29770 437 6 rows 0selected. 顺便再提一下11g新增
的其他分区策略。 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