**逻辑读在Oracle调优中有四个好处：**

**（1）逻辑读是受制于CPU能力的操作，因而，很好的反映了CPU的使用情况。**

**（2）逻辑读可能导致物理读，因而，通过减少逻辑读的数量，很可能会降低I/O操作次数。**

**（3）逻辑读是受制于串行的操作，既然经常要考虑多用户负载的优化，最小化逻辑读将有利于避免扩展性问题。**

**（4）逻辑读的数量可以通过SQL跟踪文件和动态性能视图在SQL语句以及执行计划级别获得。**

**下面就来详细的讲述下逻辑读相关的知识，以作为自己学习的一个总结。  
我们都知道,数据块是oracle最基本的读写单位,但用户所需要的数据,并不是整个块,而是块中的行,或列.当用户发出SQL语句时,此语句被解析执行完毕，就开始了数据的抓取阶段,在此阶段,服务器进程会先将行所在的数据块从数据文件中读入buffer cache,这个过程叫做物理读.物理读,每读取一个块,就算一次物理读.当块被送进buffer cache后,并不能立即将块传给用户,因为用户所需要的并不是整个块,而是块中的行.从buffer cache的块中读取行的过程,就是逻辑读.为了完成一次逻辑读,服务器进程先要在hash表中查找块所在的buffer cache 链.找到之后,需要在这个链上加一个cache buffer chains 闩,加闩成功之后,就在这个链中寻找指定的块,并在块上加一个pin锁.并释放cache buffer chains闩.然后就可以访问块中的行了.服务器进程不会将块中所有满足条件的行一次取出,而是根据你的抓取命令,每次取一定数量的行.这些行取出之后,会经由PGA传给客户端用户.行一旦从buffer cache中取出,会话要释放掉在块上所加的PIN.本次逻辑读就算结束.如果还要再抓取块中剩余的行,服务器进程要再次申请获得cache bufffer链闩.再次在块上加PIN.这就算是另外一次逻辑读咯.也就是说,服务器进程每申请一次cache buffer链闩,就是一次逻辑读.而每次逻辑读所读取的行的数量,可以在抓取命令中进行设置.  
逻辑读和Cache buffer chains闩关系密切，TOM曾有文章提到，进程每申请一次Cache buffer chains闩，就是一次逻辑读。但是，逻辑读并不等同于Cache buffer chains闩，每次逻辑读，在9i中至少需要获得两Cache buffer chains闩。逻辑读是指在Hash表中定位块的这个过程。**

分区

**-Hash分区（散列分区）**

    散列分区为通过指定分区编号来均匀分布数据的一种分区类型，因为通过在I/O设备上进行散列分区，使得这些分区大小一致。

一、好处  
  1、对于分区本生不需要定期的进行分区加入（范围分区和LIST分区需要定期的对新加入的值新建分区）  
  2、可以消除访问热点块及索引热点块，由于索引是排序后的结构，对于一列自增的列加入范围分区，可能对索引的高位块进行频繁的数据插入，导致频繁的写入和分裂  
     对于这样的索引如果加入散列分区索引即可消除。  
二、注意  
  1、分区不能太多，典型的大约1000个分区，那么在分区触发（谓词导致索引范围扫描）的并行访问操作时可能更慢，因为有非常多额外的分区维护操作（我曾经遇到过1个1844个HASH分区，导致的并行性能问题）  
  2、散列索引在维护方面可能和范围和LIST分区有一些区别。  
  详细见：

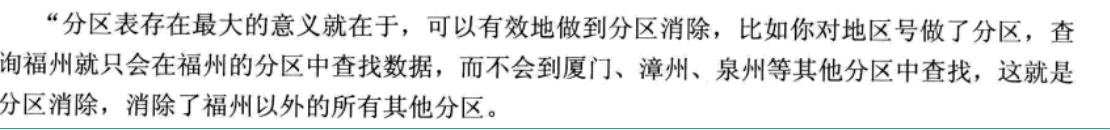
列表分区：

列表分区明确指定了根据某字段的某个具体值进行分区，而不是像范围分区那样根据字段的值范围来划分的。

范围分区：



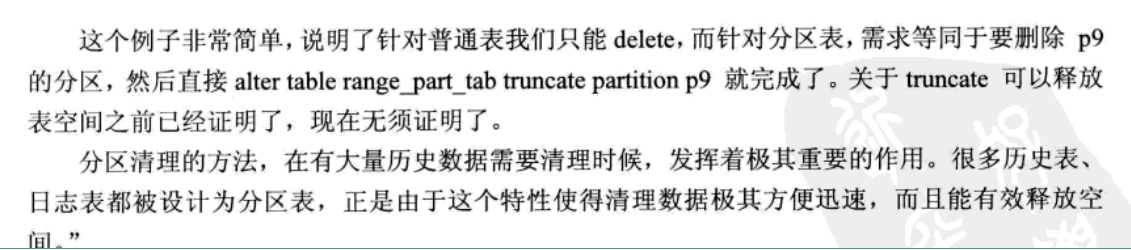
各种分区之间的优势

访问路径减少



分区truncate

delete无法释放空间，而truncate却有效释放空间，针对普通表，truncate往往不能轻易使用，delete针对某些条件的局部记录删除，而truncate显然不能戴上条件，无法做到局部删除





分区数据：

分区交换，可以实现普通表和分区表的某个分区之间数据的相互交换，他们之间的交换非常快，基本上瞬间完成，实际上在oracle内部的数据字典做的一些小动作而已，不过要两张表的字段都相同

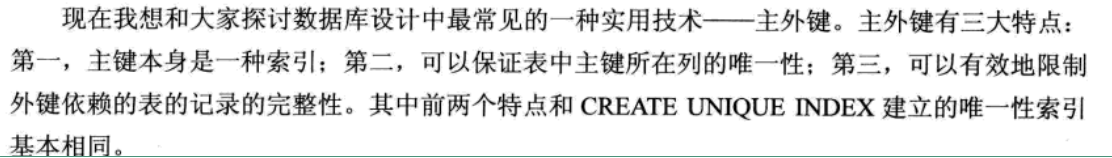


分区切割

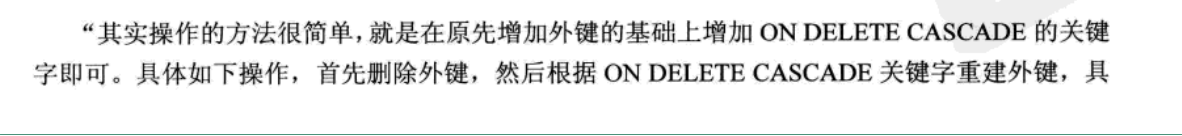


表设计：

外键 业务方面的







压缩索引

 oracle 索引压缩（key compression）是oracle 9i 中引入的一项新特性。该特性可以压缩索引或者索引组织表中的重复键值，从而节省存储空间。非分区的unique 索引和non-unique（至少两列）索引都能够被压缩。bitmap 索引不能够进行压缩。



分区索引

全局索引，局部索引



先全量，子查询筛选范围