### 什么是索引？

oracle中的索引是一种供服务器在表中快速查找一个行的[数据库结构](http://baike.baidu.com/subview/8787195/8748135.htm" \t "http://baike.baidu.com/_blank)。合理使用索引能够大大提高数据库的运行效率。

### 索引类型

### 索引通常可以分为：分区和非分区索引、常规B树索引、位图（bitmap）索引、翻转（reverse）索引等。其中，B树索引在Oracle中是一个通用索引。在创建索引时它就是默认的索引类型。我们这次研究的目标就是B树索引。B树索引可以是一个列的(简单)索引，也可以是组合/复合(多个列)的索引。B树索引最多可以包括32列。

### 创建索引优缺点：

1. 为什么要创建索引呢（优点）？  
   创建索引可以大大提高系统的性能。  
   第一，   通过创建唯一性索引，可以保证**[数据库](http://lib.csdn.net/base/mysql" \o "MySQL知识库" \t "http://blog.csdn.net/dinglang_2009/article/details/_blank)**表中每一行数据的唯一性。  
   第二，   可以大大加快数据的检索速度，这也是创建索引的最主要的原因。  
   第三，   可以加速表和表之间的连接，特别是在实现数据的参考完整性方面特别有意义。  
   第四，   在使用分组和排序子句进行数据检索时，同样可以显著减少查询中分组和排序的时间。  
   第五，   通过使用索引，可以在查询的过程中，使用优化隐藏器，提高系统的性能。  
      
   二、建立方向索引的不利因素（缺点）  
   也许会有人要问：增加索引有如此多的优点，为什么不对表中的每一个列创建一个索引呢？这种想法固然有其合理性，然而也有其片面性。虽然，索引有许多优点，但是，为表中的每一个列都增加索引，是非常不明智的。这是因为，增加索引也有许多不利的一个方面。  
      
   第一，   创建索引和维护索引要耗费时间，这种时间随着数据量的增加而增加。  
   第二，   索引需要占物理空间，除了数据表占数据空间之外，每一个索引还要占一定的物理空间，如果要建立聚簇索引，那么需要的空间就会更大。  
   第三，   当对表中的数据进行增加、删除和修改的时候，索引也要动态的维护，这样就降低了数据的维护速度。

### **索引注意原则：**

索引是建立在数据库表中的某些列的上面。因此，在创建索引的时候，应该仔细考虑在哪些列上可以创建索引，在哪些列上不能创建索引。  
一般来说，应该在这些列上创建索引。  
第一，   在经常需要搜索的列上，可以加快搜索的速度；  
第二，   在作为主键的列上，强制该列的唯一性和组织表中数据的排列结构；  
第三，   在经常用在连接的列上，这些列主要是一些外键，可以加快连接的速度；  
第四，   在经常需要根据范围进行搜索的列上创建索引，因为索引已经排序，其指定的范围是连续的；  
第六，   在经常使用在WHERE子句中的列上面创建索引，加快条件的判断速度。

第七：在表中插入数据后创建索引。在表中插入数据后，创建索引效率将更高。如果在装载数据之前创建索引，那么插入每行时oracle都必须更改索引。  
第八：索引正确的表和列。如果经常检索包含大量数据的表中小于15%的行，就需要创建索引。为了改善多个表的相互关系，常常使用索引列进行关系连接。

第九：主键和唯一关键字所在的列自动具有索引，但应该在与之关联的表中的[外部关键字](http://baike.baidu.com/subview/703558/703558.htm" \t "http://baike.baidu.com/_blank)所在的列上创建索引。  
　　第十、合理安排索引列。在createindex语句中，列的排序会影响查询的性能，通常将最常用的列放在前面。创建一个索引来提高多列的查询效率时，应该清楚地了解这个多列的索引对什么列的存取有效，对什么列的存取无效。

　例如：在A,B,C三列上创建索引  
　　A有效　AB有效 ABC有效  
　　第十一、限制表中索引的数量。尽管表可以有任意数量的索引，可是索引越多，在修改表中的数据时对索引做出相应更改的工作量也越大，效率也就越低。同样，目前不用的索引应该及时删除。  
　　第十二、指定索引[数据块](http://baike.baidu.com/subview/702806/702806.htm" \t "http://baike.baidu.com/_blank)空间的使用。创建索引时，索引的数据块是用表中现存的值填充的，直到达到PCTFREE为止。如果打算将许多行插入到被索引的表中，PCTFREE就应设置得大一点，不能给索引指定PCTUSED。  
　　第十三、根据索引大小设置存储参数。创建索引之前应先估计索引的大小，以便更好地促进规划和管理磁盘空间。单个索引项的最大值大约是数据块大小的一半。

**同样，对于有些列不应该创建索引。一般来说，不应该创建索引的的这些列具有下列特点：**

第一，   对于那些在查询中很少使用或者参考的列不应该创建索引。这是因为，既然这些列很少使用到，因此有索引或者无索引，并不能提高查询速度。相反，由于增加了索引，反而降低了系统的维护速度和增大了空间需求。  
第二，    对于那些只有很少数据值的列也不应该增加索引。这是因为，由于这些列的取值很少，例如人事表的性别列，在查询的结果中，结果集的数据行占了表中数据行的很大比例，即需要在表中搜索的数据行的比例很大。增加索引，并不能明显加快检索速度。  
第三，    对于那些定义为text, image和bit数据类型的列不应该增加索引。这是因为，这些列的数据量要么相当大，要么取值很少。  
第 四，    当修改性能远远大于检索性能时，不应该创建索引。这是因为，修改性能和检索性能是互相矛盾的。当增加索引时，会提高检索性能，但是会降低修改性能。当减少索引时，会提高修改性能，降低检索性能。因此，当修改性能远远大于检索性能时，不应该创建索引。

注：**容易引起oracle索引失效的原因很多：**

**（1）在索引列上使用函数。如SUBSTR,DECODE,INSTR等，对索引列进行运算.需要建立函数索引就可以解决了。**

**（2）新建的表还没来得及生成统计信息，分析一下就好了**

**（3）基于cost的成本分析，访问的表过小，使用全表扫描的消耗小于使用索引。**

**（4）使用<>、not in 、not exist，对于这三种情况大多数情况下认为结果集很大，一般大于5%-15%就不走索引而走FTS。**

**（5）单独的>、<。**

**（6）like "%\_" 百分号在前。**

**（7）单独引用复合索引里非第一位置的索引列。**

**（8）字符型字段为数字时在where条件里不添加引号。**

**（9）当变量采用的是times变量，而表的字段采用的是date变量时.或相反情况。**

**（10）索引失效，可以考虑重建索引，rebuild online。**

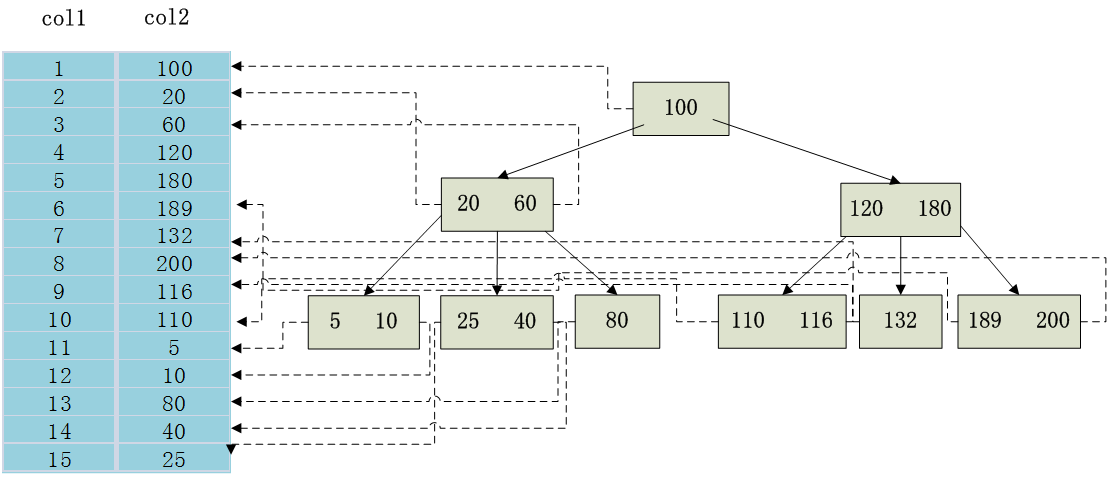
**（11）B-tree索引 is null不会走,is not null会走,位图索引 is null,is not null  都会走、 联合索引 is not null 只要在建立的索引列（不分先后）都会走。**

### **B-tree索引**

是一种平衡的m-way树查找形式

### 查询原理：

  数据库索引，是数据库管理系统中一个排序的数据结构，以协助快速查询、更新数据库表中数据。索引的实现通常使用B\_TREE。B\_TREE索引加速了数据访问，因为存储引擎不会再去扫描整张表得到需要的数据；相反，它从根节点开始，根节点保存了子节点的指针，存储引擎会根据指针快速寻找数据。



        上图显示了一种索引方式。左边是数据库中的数据表，有col1和col2两个字段，一共有15条记录；右边是以col2列为索引列的B\_TREE索引，每个节点包含索引的键值和对应数据表地址的指针，这样就可以都过B\_TREE在一定的时间复杂度内获取相应的数据，这样明显地加快了检索的速度。

### 深入研究参考文档：

见深入研究参考文档

建表需求文档

场景：模拟一个天猫网店的后台数据，忽略掉用户信息

建立数据库和需要的Table：

1.product 产品表

id

name

price

pnum 库存

category 分类

2.order 订单表

id 订单id

money 订单价格

order\_time 订单时间

receiveInfo 收货信息

paystate 支付状态

3.orderitem 订单详情

order\_id 订单id

product\_id 商品id

buy\_num 购买数量

总的商品数量大概在200W左右，商品的订单大概在5W左右，订单详情至少10W条数据