**[SQL优化及注意事项](https://www.cnblogs.com/barrywxx/p/4342989.html)**

1、 把数据、日志、索引放到不同的I/O设备上，增加读取速度。数据量（尺寸）越大，提高I/O越重要。  
  
2、 纵向、横向分割表，减少表的尺寸，如：可以把大数据量的字段拆分表。  
  
3、 根据查询条件，建立索引，优化索引、优化访问方式，限制结果集的数据量。注意填充因子要适当（最好是使用默认值0）。索引应该尽量小，尽量使用字节数小的列建索引，不要对有限的几个值的列建单一索引。  
  
4、 用OR的字句可以分解成多个查询，并且通过UNION链接多个查询。它们的速度只与是否使用索引有关，如果查询需要用到联合索引，用UNION all执行的效率更高。  
  
5、 在查询SELECT语句中用WHERE子句限制返回的行数，避免表扫描。如果返回不必要的数据，则浪费了服务器的I/O资源，加重了网络的负担，降低了性能。如果表很大，在表扫描期间将表锁住，禁止其他的联结访问表，后果很严重。  
  
6、 注意使用DISTINCT，在没有必要时不要用，它同UNION一样会使查询变慢。  
  
7、 在IN后面值的列表中，将出现最频繁的值放在最前面，出现最少的放在最后面，减少判断的次数。  
  
8、 一般在GROUP BY和HAVING子句之前就能剔除多余的行，所以尽量不要用它们来做剔除行的工作，也就是说尽可能在WHERE中过滤数据。  
  
9、 尽量将数据的处理工作放在服务器上，减少网络的开销，如使用存储过程。存储过程是编译、优化过，并且被组织到一个执行规划里，且存储在数据库中的SQL语句（存储过程是数据库服务器端的一段程序），是控制流语言的集合，速度当然快。  
  
10、 不要在一句话里再三地使用相同的函数，浪费资源，将结果放在变量里再调用更快。  
  
11、 针对大量只读查询操作进行优化的方法：  
1) 数据量小的数据，可以考虑不存储在数据库中，而是通过程序常量的方式解决。  
2) 需要存储在数据库中的数据，可以考虑采用物化视图（索引视图）。当DBA在视图上创建索引时，这个视图就被物化（执行）了，并且结果集被永久地保存在唯一索引中，保存方式与一个有聚簇索引的表的保存方式相同。物化视图减除了为引用视图的查询动态建立结果集的开销，优化人员可以在查询中使用视图索引，而不需要在FROM子句中直接指定视图。  
3) 数据存储时可以考虑适当的数据冗余，以减少数据库表之间的链接操作，提高查询效率。  
4) 针对数据的特点，采取特定的索引类型。例如，位图索引等。  
  
12、 对于SQL语句书写时的一些建议：  
1) 写语句时能够确定数据库对象所有者的，尽可能把所有者带上，如：  
  
SELECT \* FROM dbo.Users  
  
2) 存储过程中，参数定义最好放在最前面，尽可能一次定义，如：  
DECLARE @USER\_ID INT  
  
,@USER\_NAME VARCHAR(50)  
  
,@PASSWORD VARCHAR(50)  
  
  
3) 为参数赋值时，尽可能一次赋值，如：  
SELECT @USER\_ID = 1001  
  
,@USER\_NAME = 'xiaojun.liu'  
  
  
4) 尽量少用游标