

优化MySQL数据库性能的八大“妙手”计算机等级考试 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__E4_BC_98_E5_8C_96MySQ_c98_644437.htm

本文探讨了提高MySQL数据库性能的思路，并从8个方面给出了具体的解决方法。

- 1、选取最适用的字段属性 MySQL可以很好的支持大数据量的存取，但是一般说来，数据库中的表越小，在它上面执行的查询也就会越快。因此，在创建表的时候，为了获得更好的性能，我们可以将表中字段的宽度设得尽可能小。例如，在定义邮政编码这个字段时，如果将其设置为CHAR(255),显然给数据库增加了不必要的空间，甚至使用VARCHAR这种类型也是多余的，因为CHAR(6)就可以很好的完成任务了。同样的，如果可以的话，我们应该使用MEDIUMINT而不是BIGINT来定义整型字段。另外一个提高效率的方法是在可能的情况下，应该尽量把字段设置为NOT NULL，这样在将来执行查询的时候，数据库不用去比较NULL值。对于某些文本字段，例如“省份”或者“性别”，我们可以将它们定义为ENUM类型。因为在MySQL中，ENUM类型被当作数值型数据来处理，而数值型数据被处理起来的速度要比文本类型快得多。这样，我们又可以提高数据库的性能。
- 2、使用连接（JOIN）来代替子查询(Sub-Queries) MySQL从4.1开始支持SQL的子查询。这个技术可以使用SELECT语句来创建一个单列的查询结果，然后把这个结果作为过滤条件用在另一个查询中。例如，我们要将客户基本信息表中没有任何订单的客户删除掉，就可以利用子查询先从销售信息表中将所有发出订单的客户ID取出来，然后将结果传递给主查询，如下所示：

DELETE FROM customerinfo WHERE CustomerID NOT in (SELECT CustomerID FROM salesinfo) 使用子查询可以一次性的完成很多逻辑上需要多个步骤才能完成的SQL操作，同时也可以避免事务或者表锁死，并且写起来也很容易。但是，有些情况下，子查询可以被更有效率的连接（JOIN）.. 替代。例如，假设我们要将所有没有订单记录的用户取出来，可以用下面这个查询完成：SELECT * FROM customerinfo WHERE CustomerID NOT in (SELECT CustomerID FROM salesinfo) 如果使用连接（JOIN）.. 来完成这个查询工作，速度将会快很多。尤其是当salesinfo表中对CustomerID建有索引的话，性能将会更好，查询如下：SELECT * FROM customerinfo LEFT JOIN salesinfo ON customerinfo.CustomerID=salesinfo. CustomerID WHERE salesinfo.CustomerID IS NULL 连接（JOIN）.. 之所以更有效率一些，是因为MySQL不需要在内存中创建临时表来完成这个逻辑上的需要两个步骤的查询工作。

3、使用联合(UNION)来代替手动创建的临时表 MySQL 从 4.0 的版本开始支持 UNION 查询，它可以把需要使用临时表的两条或更多的 SELECT 查询合并的一个查询中。在客户端的查询会话结束的时候，临时表会被自动删除，从而保证数据库整齐、高效。使用 UNION 来创建查询的时候，我们只需要用 UNION 作为关键字把多个 SELECT 语句连接起来就可以了，要注意的是所有 SELECT 语句中的字段数目要想同。下面的例子就演示了一个使用 UNION 的查询。SELECT Name, Phone FROM client UNION SELECT Name, BirthDate FROM author UNION SELECT Name, Supplier FROM product

4、事务 尽管我们可以

使用子查询 (Sub-Queries)、连接 (JOIN) 和联合 (UNION) 来创建各种各样的查询，但不是所有的数据库操作都可以只用一条或少数几条SQL语句就可以完成的。更多的时候是需要用到一系列的语句来完成某种工作。但是在这种情况下，当这个语句块中的某一条语句运行出错的时候，整个语句块的操作就会变得不确定起来。设想一下，要把某个数据同时插入两个相关联的表中，可能会出现这样的情况：第一个表中成功更新后，数据库突然出现意外状况，造成第二个表中的操作没有完成，这样，就会造成数据的不完整，甚至会破坏数据库中的数据。要避免这种情况，就应该使用事务，它的作用是：要么语句块中每条语句都操作成功，要么都失败。换句话说，就是可以保持数据库中数据的一致性和完整性。事物以BEGIN 关键字开始，COMMIT关键字结束。在这之间的一条SQL操作失败，那么，ROLLBACK命令就可以把数据库恢复到BEGIN开始之前的状态。 BEGIN. INSERT INTO salesinfo SET CustomerID=14. UPDATE inventory SET Quantity=11 WHERE item=book. COMMIT. 事务的另一个重要作用是当多个用户同时使用相同的数据源时，它可以利用锁定数据库的方法来为用户提供一种安全的访问方式，这样可以保证用户的操作不被其它的用户所干扰。 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