### 索引优化

一、存储引擎以不同方式使用B-Tree索引

1.MyISAM使用前缀压缩技术使索引更小，但InnoDB则按照原数据格式存储。

2.MyISAM索引通过物理位置引用被索引的行，InnoDB则根据主键引用被索引的行

二、使用索引

索引的优点

建立索引的目的是加快对表中记录的查找或排序！

① 建立索引的列可以保证行的唯一性，生成唯一的rowId

② 建立索引可以有效缩短数据的检索时间

③ 建立索引可以加快表与表之间的连接

④ 为用来排序或者是分组的字段添加索引可以加快分组和排序顺序

4. 索引的缺点

① 创建索引和维护索引需要时间成本，这个成本随着数据量的增加而加大

② 创建索引和维护索引需要空间成本，每一条索引都要占据数据库的物理存储空间，数据量越大，占用空间也越大（数据表占据的是数据库的数据空间）

③ 会降低表的增删改的效率，因为每次增删改索引需要进行动态维护，导致时间变长

索引的分类和创建和修改删除等命令

1. 基本索引类型

① 普通索引（单列索引）

② 复合索引（组合索引）

③ 唯一索引

④ 主键索引

⑤ 全文索引

2. 创建的语句

CREATE TABLE table\_name[col\_name data type]

[unique|fulltext][index|key][index\_name](col\_name[length])[asc|desc]

unique|fulltext为可选参数，分别表示唯一索引、全文索引

index和key为同义词，两者作用相同，用来指定创建索引

col\_name为需要创建索引的字段列，该列必须从数据表中该定义的多个列中选择

index\_name指定索引的名称，为可选参数，如果不指定，默认col\_name为索引值

length为可选参数，表示索引的长度，只有字符串类型的字段才能指定索引长度

asc或desc指定升序或降序的索引值存储

① 普通索引（单列索引）

普通索引（单列索引）：单列索引是最基本的索引，它没有任何限制。

② 复合索引（组合索引）

复合索引：复合索引是在多个字段上创建的索引。复合索引遵守“最左前缀”原则，即在查询条件中使用了复合索引的第一个字段，索引才会被使用。因此，在复合索引中索引列的顺序至关重要。

③ 唯一索引

唯一索引：唯一索引和普通索引类似，主要的区别在于，唯一索引限制列的值必须唯一，但允许存在空值（只允许存在一条空值）。

前缀索引

Alter table tableName add key (cloum(num)) num为前缀长度

前缀索引能使索更小，更快，但无法使用前缀索引做ORDER BY和GROUP BY

多列索引

给独立列建立多个单列索引，虽然MySQL5.0以后引入索引合并策略，但仍是个糟糕的选择。

当出现对多个索引操作时，通常意味着需要一个包含相关列的多列索引，而不是独立的单列索引，效果还不如直接UNION

根据选择性和基数的经验法则选择合适的索引列顺序

维护索引和表

1. 找到并修复损坏的表

check table 命令检查 repair table修复 （不是所有引擎都支持）不支持可以通过不做任何操作的alter重建表如：alter table innodb\_tbl enging=innodb

1. 维护准确的索引统计信息

ANALYZE RABLE更新生成统计信息

3、减少碎片

Optimize table 整理数据 不支持的存储引擎 通过alter重建表更新

### 查询性能优化

优化数据访问

1. 是否向数据库请求了不需要的数据 如select \*
2. MySQL是否在扫描额外的记录

从响应时间、扫描行数、返回行数三个指标上看，在EXPLAN语句中的type列反应了访问类型

重构查询

1. 将一个复杂查询分成多个简单查询
2. 切分查询，将大查询切分成小查询，每次只返回一小部分结果。如删除旧数据是，定期清理大量数据会耗尽系统资源阻塞很多重要查询，可以将大的DELETE切分成很多小的查询，一次删除一部分，暂停一会再做下一次删除
3. 分解关联查询，使用场景，如：当应用能够方便的缓存单个查询结果的时候、当可以将数据分到不同的服务器上的时候、当能够使用IN()代替关联查询的时候、当查询中使用同一个数据表的时候

查询执行的基础

**当希望MySQL能够以更高性能运行查询时，最好的方法就是弄清楚MySQL是如何优化和执行查询的**

1. 客户端发送一条查询给服务器
2. 服务器先检查查询缓存，如果命中缓存，则立刻返回缓存中结果，否则进入下个阶段
3. 服务器端进行SQL解析、预处理，再由优化器生成对应的执行计划。
4. MySQL根据优化器生成的执行计划，调用存储引擎API来执行查询。
5. 将结果返回给客户端。