MySQL8第108期-性能优化之数据库结构1

原创 维宝儿 运维一线 2025年03月18日 10:00 河南



其实设计数据库的时候,如果方案整明白了,性能真能翻倍提升!打个比方,就像搭积木——结构搭得越合理,不光硬盘能少占地方(省空间),查数据还能像刷短视频一样快。具体操作时得注意几个坑:比如重复数据别堆太多(像手机里存了100张一样的照片),查数据别卡成PPT,更新数据别转圈圈,还有每个字段的类型也要选准(比如存电话号码就别用数字类型,否则开头带0的号码就完犊子)。

// 将字段很多的表分解成多个表 //

对于字段较多的表,如果有些字段的使用频率很低,可以将这些字段分离 出来,形成新表。因为当一个表的数据量很大时,会由于使用频率低的字段的 存在而变慢。

假设会员表存储会员登录认证信息,该表中有很多字段,如id、姓名、密码、地址、电话、个人描述字段。其中,地址、电话、个人描述等字段并不常用。可以将这些不常用字段分解出另外一个表。将这个表取名叫members_detail。表中有member_id、address、telephone、description等字段。其中,member_id是会员编号,address字段存储地址信息,telephone字段存储电话信息,description字段存储会员个人描述信息。这样就把会员表分成两个表,分别为members表和members_detail表。

创建这两个表的 SQL 语句如下:

```
CREATE TABLE members(
```

ID int(11) NOT NULL AUTO_INCREMENT,
username varchar(255) DEFAULT NULL,
password varchar(255) DEFAULT NULL,
last_login_time datetime DEFAULT NULL,
last_login_ip varchar(255) DEFAULT NULL,
PRIMARY KEY (Id)
);
CREATE TABLE members_detail(
member_id int(11) NOT NULL DEFAULT 0,
address varchar(255) DEFAULT NULL,
telephone varchar(16) DEFAULT NULL,
description text

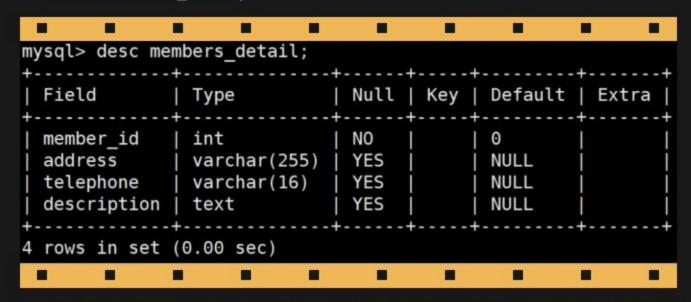
```
mysql> CREATE TABLE members(
          ID int(11) NOT NULL AUTO_INCREMENT,
    ^
          username varchar(255) DEFAULT NULL,
    ^
           password varchar(255) DEFAULT NULL,
    -
          last_login_time datetime DEFAULT NULL,
    ^
          last_login_ip varchar(255) DEFAULT NULL,
    ^
           PRIMARY KEY (Id)
    -
    -> );
Query OK, 0 rows affected, 1 warning (0.03 sec)
mysql> CREATE TABLE members_detail(
          member_id int(11) NOT NULL DEFAULT 0,
    ^
           address varchar(255) DEFAULT NULL,
    -
           telephone varchar(16) DEFAULT NULL,
    ^
          description text
    ^
    -> );
Query OK, 0 rows affected, 1 warning (0.02 sec)
```

这两个表的结构如下:

desc members;

				•			•		
mysql> desc members;									
Fiel	d	Typ				Default		a	İ
last last	name word _login_tim _login_ip	var e dat var	rchar(255) rchar(255) tetime rchar(255)		PRI 	NULL NULL NULL NULL NULL		_incremen	t
	in set (0								

desc members_detail;



如果需要查询会员的详细信息,可以用会员的id 来查询。如果需要将会员的基本信息和详细信息同时显示,可以将 members 表和 members_detail 表进行联合查询,查询语句如下:

SELECT * FROM members LEFT JOIN members_detail ON members.id=members_detail.member_id;



现在表里还没数据,所以查询结果为空。

通过这种分解,可以提高表的查询效率。对于字段很多且有些字段使用不 频繁的表,可以通过这种分解的方式来优化数据库的性能。

(以上内容参考书籍《MySQL8从入门到精通》,作者王英英,如有需要请自行购买,如有侵权,请联系删除。)

