

公告

wiki和教程：www.pythontav.com

免费教学视频：[B站](#)：[凸头统治地球](#)

高级专题教程：[网易云课堂](#)：[武沛齐](#)

聊技术，加武Sir微信



武沛齐

扫一扫上面的二维码图案，加我微信

Python技术交流群：737658057

软件测试开发交流群：721023555

昵称：武沛齐
园龄：8年1个月
粉丝：9974
关注：44
[+加关注](#)

我的标签

- Python(17)
- ASP.NET MVC(15)
- python之路(7)
- Tornado源码分析(5)
- 每天一道Python面试题(5)
- crm项目(4)
- 面试都在问什么？(2)
- Python开源组件 - Tyrion(1)
- Python面试315题(1)
- Python企业面试题讲解(1)

积分与排名

积分 - 426642
排名 - 785

随笔分类

- JavaScript(1)
- MVC(15)
- Python(17)
- 面试都在问什么系列？【图】(2)
- 其他(37)

随笔 - 140 文章 - 164 评论 - 893

索引补充

1、索引

索引是表的目录，在查找内容之前可以先在目录中查找索引位置，以此快速定位查询数据。对于索引，会保存在额外的文件中。

2、索引种类

- 普通索引：仅加速查询
- 唯一索引：加速查询 + 列值唯一（可以有null）
- 主键索引：加速查询 + 列值唯一 + 表中只有一个（不可以有null）
- 组合索引：多列值组成一个索引，
专门用于组合搜索，其效率大于索引合并
- 全文索引：对文本的内容进行分词，进行搜索

索引合并，使用多个单列索引组合搜索
覆盖索引，select的数据列只用从索引中就能够取得，不必读取数据行，换句话说查询列要被所建的索引覆盖

3、相关命令

```
1 - 查看表结构
2 desc 表名
3
4 - 查看生成表的SQL
5 show create table 表名
6
7 - 查看索引
8 show index from 表名
9
10 - 查看执行时间
11 set profiling = 1;
12 SQL...
13 show profiles;
```

4、使用索引和不使用索引

```
1 由于索引是专门用于加速搜索而生，所以加上索引之后，查询效率会快到飞起来。
2
3 # 有索引
4 mysql> select * from tb1 where name = 'wupeiqi-888';
5 +-----+-----+-----+-----+
6 | nid | name          | email                | radom
7 +-----+-----+-----+-----+
8 | 889 | wupeiqli-888 | wupeiqli888@live.com | 5312269e76a16a90b8a8301d5314204
9 +-----+-----+-----+-----+
10 1 row in set (0.00 sec)
11
12 # 无索引
13 mysql> select * from tb1 where email = 'wupeiqli888@live.com';
14 +-----+-----+-----+-----+
15 | nid | name          | email                | radom
16 +-----+-----+-----+-----+
17 | 889 | wupeiqli-888 | wupeiqli888@live.com | 5312269e76a16a90b8a8301d5314204
18 +-----+-----+-----+-----+
19 1 row in set (1.23 sec)
```

企业面试题及答案(1)
请求响应(6)
设计模式(9)
微软C#(34)

随笔档案

- 2020年6月(1)
- 2020年5月(1)
- 2019年11月(1)
- 2019年10月(1)
- 2019年9月(4)
- 2018年12月(1)
- 2018年8月(1)
- 2018年5月(2)
- 2018年4月(1)
- 2017年8月(1)
- 2017年5月(1)
- 2017年3月(1)
- 2016年10月(1)
- 2016年7月(1)
- 2015年10月(1)
- 2015年8月(1)
- 2015年7月(1)
- 2015年6月(2)
- 2015年4月(2)
- 2014年3月(3)
- 2014年1月(3)
- 2013年12月(2)
- 2013年11月(2)
- 2013年10月(7)
- 2013年8月(17)
- 2013年7月(1)
- 2013年6月(14)
- 2013年5月(23)
- 2013年4月(3)
- 2013年3月(13)
- 2013年2月(1)
- 2012年11月(26)

相册

git(14)

最新评论

1. Re:Python开发【第十九篇】：Python操作MySQL
执行本网页中第一段SQL语句，其中执行语句改成： cursor.execute('insert into part(caption) values("AB") ') 报错
unable to reso...
--serene1979

2. Re:python 面向对象（进阶篇）
class Foo:

pass

这个类不是由,object.new(cls)创建的吗

5、正确使用索引

数据库表中添加索引后确实会让查询速度起飞，但前提必须是正确的使用索引来查询，如果以错误的方式使用，则即使建立索引也不会奏效。
即使建立索引，索引也不会生效：

```
1 - like '%xx'
2     select * from tb1 where name like '%cn';
3 - 使用函数
4     select * from tb1 where reverse(name) = 'wupeiqi';
5 - or
6     select * from tb1 where nid = 1 or email = 'seven@live.com';
7     特别的：当or条件中有未建立索引的列才失效，以下会走索引
8         select * from tb1 where nid = 1 or name = 'seven';
9         select * from tb1 where nid = 1 or email = 'seven@live.com' and
10
11 - 类型不一致
12     如果是字符串类型，传入条件是必须用引号引起来，不然...
13     select * from tb1 where name = 999;
14 - !=
15     select * from tb1 where name != 'alex'
16     特别的：如果是主键，则还是会走索引
17     select * from tb1 where nid != 123
18 - >
19     select * from tb1 where name > 'alex'
20     特别的：如果是主键或索引是整数类型，则还是会走索引
21     select * from tb1 where nid > 123
22     select * from tb1 where num > 123
23 - order by
24     select email from tb1 order by name desc;
25     当根据索引排序时候，选择的映射如果不是索引，则不走索引
26     特别的：如果对主键排序，则还是走索引：
27     select * from tb1 order by nid desc;
28 - 组合索引最左前缀
29     如果组合索引为：(name,email)
30     name and email      -- 使用索引
31     name                 -- 使用索引
32     email                -- 不使用索引
```

6、其他注意事项

- 避免使用select *
- count(1)或count(列) 代替 count(*)
- 创建表时尽量用 char 代替 varchar
- 表的字段顺序固定长度的字段优先
- 组合索引代替多个单列索引（经常使用多个条件查询时）
- 尽量使用短索引
- 使用连接（JOIN）来代替子查询(Sub-Queries)
- 连表时注意条件类型需一致
- 索引散列值（重复少）不适合建索引，例：性别不适合

7、limit分页

无论是否有索引，limit分页是一个值得关注的问题

每页显示10条:
当前 118 120, 125

倒序:

大

小

980 970 7 6 6 5 54 43 32

21 19 98

--bajie_new

3. Re:MySQL练习题参考答案

14、查询和“002”号的同学学习的课程完全相同的其他同学学号和姓名； select sid,sname from (select sid,sname from student where sid i...

--BobAylIn

4. Re:MySQL练习题参考答案

13、查询至少学过学号为“001”同学所有课的其他同学学号和姓名； 需要考虑这种情况，001同学选了1、2、4，002同学选了1、2、3。这种情况，两位同学选课有交集，count(选课数)也一样，然是...

--BobAylIn

5. Re:人生没有白走的路， 每一步都算数 后续来了吗

--小鱼仔。

下一页:

```
select
*
from
tb1
where
nid < (select nid from (select nid from tb1 where nid < 当前页最小值 o
order by
nid desc
limit 10;

select
*
from
tb1
where
nid < (select nid from (select nid from tb1 where nid < 970 order b
order by
nid desc
limit 10;
```

上一页:

```
select
*
from
tb1
where
nid < (select nid from (select nid from tb1 where nid > 当前页最大值 o
order by
nid desc
limit 10;

select
*
from
tb1
where
nid < (select nid from (select nid from tb1 where nid > 980 order by
order by
nid desc
limit 10;
```



8、执行计划

explain + 查询SQL - 用于显示SQL执行信息参数，根据参考信息可以进行SQL优化

```
mysql> explain select * from tb2;
+----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | type | possible_keys | key | key_len | ref |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE | tb2 | ALL | NULL | NULL | NULL | NULL |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```



id

查询顺序标识

如: mysql> explain select * from (select nid,name from tbl where

+-----+-----+-----+-----+-----+-----+					
id	select_type	table	type	possible_keys	key
+-----+-----+-----+-----+-----+-----+					
1	PRIMARY	<derived2>	ALL	NULL	NULL
2	DERIVED	tbl	range	PRIMARY	PRIMAR
+-----+-----+-----+-----+-----+-----+					

特别的: 如果使用union连接气值可能为null

select_type

查询类型

SIMPLE	简单查询
PRIMARY	最外层查询
SUBQUERY	映射为子查询
DERIVED	子查询
UNION	联合
UNION RESULT	使用联合的结果
...	

table

正在访问的表名

type

查询时的访问方式, 性能: all < index < range < index_merge < ref_or_null

ALL	全表扫描, 对于数据表从头到尾找一遍 select * from tbl; 特别的: 如果有limit限制, 则找到之后就不在继续向下扫描 select * from tbl where email = 'seven@li select * from tbl where email = 'seven@li 虽然上述两个语句都会进行全表扫描, 第二句使用了lim
-----	--

INDEX	全索引扫描, 对索引从头到尾找一遍 select nid from tbl;
-------	---

RANGE	对索引列进行范围查找 select * from tbl where name < 'alex'; PS: between and in > >= < <= 操作 注意: != 和 > 符号
-------	---

INDEX_MERGE	合并索引, 使用多个单列索引搜索 select * from tbl where name = 'alex' or nid in
-------------	---

REF	根据索引查找一个或多个值 select * from tbl where name = 'seven';
-----	---

EQ_REF	连接时使用primary key 或 unique类型 select tb2.nid,tbl.name from tb2 left join tbl o
--------	---

CONST	常量 表最多有一个匹配行, 因为仅有一行, 在这行的列值可被优化器剩余 select nid from tbl where nid = 2 ;
-------	---

SYSTEM	系统 表仅有一行 (=系统表)。这是const联接类型的一个特例。 select * from (select nid from tbl where nid = 1
--------	--

possible_keys

可能使用的索引



更多参见:

<http://www.cnblogs.com/xiaoboluo768/p/5400990.html>

http://dev.mysql.com/doc/refman/5.7/en/explain-output.html#jointype_system

9、慢日志查询

a、配置MySQL自动记录慢日志

<code>slow_query_log = OFF</code>	是否开启慢日志记录
<code>long_query_time = 2</code>	时间限制，超过此时间，则记录
<code>slow_query_log_file = /usr/slow.log</code>	日志文件
<code>log_queries_not_using_indexes = OFF</code>	为使用索引的搜索是否记录

注：查看当前配置信息：

```
show variables like '%query%'
```

修改当前配置：

```
set global 变量名 = 值
```

b、查看MySQL慢日志

```
mysqldumpslow -s at -a /usr/local/var/mysql/MacBook-Pro-3-slow.log
```

```

1  ""
2  --verbose  版本
3  --debug    调试
4  --help     帮助
5
6  -v         版本
7  -d         调试模式
8  -s ORDER   排序方式
9
10             what to sort by (al, at, ar, c, l, r, t), 'at' is default
11             al: average lock time
12             ar: average rows sent
13             at: average query time
14             c: count
15             l: lock time
16             r: rows sent
17             t: query time

```

```
17 -r          反转顺序，默认文件倒序拍。reverse the sort order (largest last i
18 -t NUM      显示前N条just show the top n queries
19 -a          不要将SQL中数字转换成N，字符串转换成S。don't abstract all numbers
20 -n NUM      abstract numbers with at least n digits within names
21 -g PATTERN  正则匹配；grep: only consider stmts that include this string
22 -h HOSTNAME mysql机器名或者IP；hostname of db server for *-slow.log filenam
23             default is '*', i.e. match all
24 -i NAME      name of server instance (if using mysql.server startup script)
25 -l          总时间中不减去锁定时间；don't subtract lock time from total time
26 """"
```



作者: 武沛齐
出处: <http://www.cnblogs.com/wupeiqi/>
本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章
页面明显位置给出原文连接。

好文要顶

关注我

收藏该文



武沛齐
关注 - 44
粉丝 - 9974
[+加关注](#)

5 0

posted @ 2016-07-29 07:50 武沛齐 阅读(8290) 评论(1) 编辑 收藏

评论列表

- #1楼 2019-01-03 01:40 扫驴 回复 引用
- 武老师，今天看了您的关于分页的讲解，有一点疑惑：
- 上一页 192 193 [196] 197 198 199 下一页
- 比如说访客现在在196页，要访问199页，您的方法是一大长串比较复杂的sql语句 select * from userinfo3 where id in (select id from (select id from userinfo3 where id > max_id limit 30) as N order by N.id desc limit 10);您的方法是基于当前页面的最大id，计算出199页面的所有id。
- 但是，为什么不能直接这样呢？
select * from xx where id>(select id from xx limit 1990,1) limit 10
就是先计算出199页的最小id，select id from xx limit 1990,1 ;然后直接查找。
这样我感觉也没什么性能上的落后啊。
- 支持(0) 反对(1)

发表评论

编辑

预览

B

支持 Markdown

提交评论

退出

订阅评论

[Ctrl+Enter快捷键提交]

- 【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库
- 【推荐】了不起的开发者，挡不住的华为，园子里的品牌专区
- 【推荐】《Flutter in action》开放下载! 闲鱼Flutter企业级实践精选

相关博文:

- [mysql -- 索引补充](#)
- [mysql补充索引](#)
- [mysql索引补充](#)
- [索引补充](#)
- [索引补充](#)
- » [更多推荐...](#)

最新 IT 新闻:

- [马斯克的Neuralink 是先知的指引还是无知的妄想?](#)
- [2020年度国家“杰出青年”公布，21人计算机领域贡献突出](#)
- [腾讯不想把半条命给合作伙伴了?](#)
- [听小米员工讲述他们所亲历的小米10年](#)
- [一边退场一边上市 新造车进入“季后赛”](#)
- » [更多新闻...](#)