



南开大学
Nankai University

查询缓存

数据库系统上机

计算机学院&网络空间安全学院 乜鹏

<https://dbis.nankai.edu.cn/2019/0417/c12139a128118/page.htm>

声明：上机课程的内容偏向举例，通俗化，一些术语的准确定义请查看理论课程。

01

查询缓存是什么



00:00:00 00:00:00 00:00:00

投票 最多可选3项



是真的吗



南开大学
Nankai University



输入的每一个 SQL 语句都会被服务器完整执行?

A

是

B

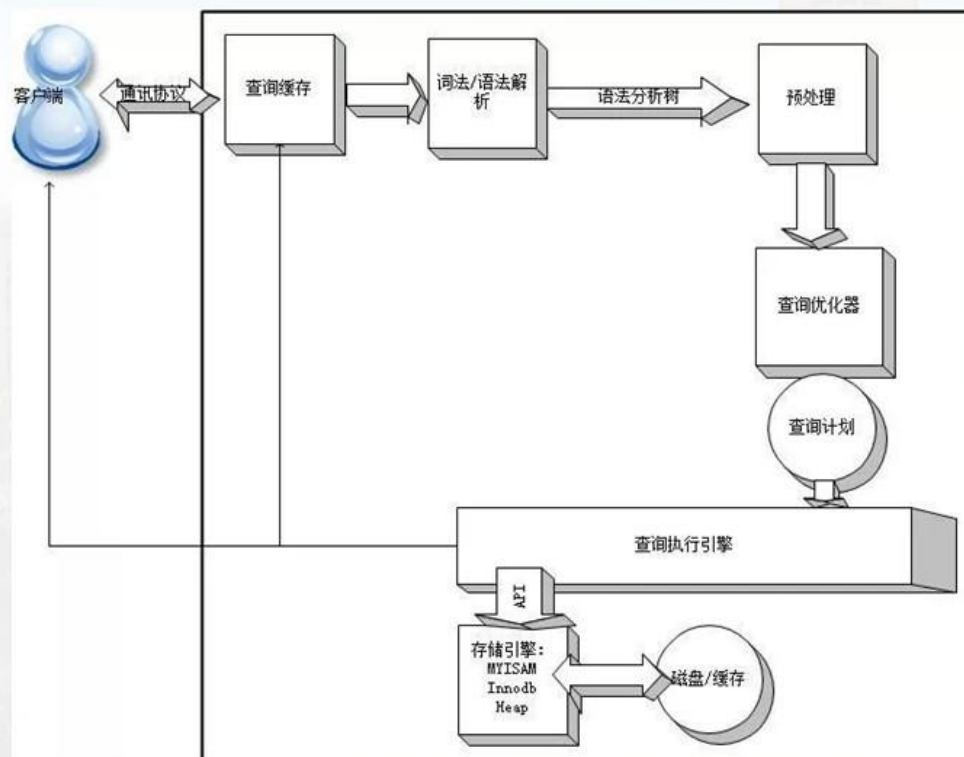
不是

C

不清楚

允公允能 日新月异

NANKAI UNIVERSITY



MySQL 查询缓存是 MySQL 中比较独特的一个缓存区域，用来缓存特定 Select 语句的整个结果集信息，且共享给所有客户端。

如果运行相同的SQL，服务器直接从缓存中取到结果，而不需要再去解析和执行SQL。如果表更改了，那么使用这个表的所有缓存查询将不再有效，查询缓存中值相关条目被清空。



查询缓存开启前后对比



南开大学
Nankai University

```
mysql> select * from emp where empno = 1234567;
+-----+-----+-----+-----+-----+-----+-----+-----+
| empno | ename | job      | ngr | hiredate |          |          |          |          |
| password |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1234567 | KUtijX | SALESMAN | 1 | 2015-04-12 | 2000.00 | 400.00 | 345 | a9cedc94b4e995ec3afd89b3a4f36982 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.54 sec)

mysql> select * from emp where empno = 1234567;
+-----+-----+-----+-----+-----+-----+-----+-----+
| empno | ename | job      | ngr | hiredate |          |          |          |          |
| password |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1234567 | KUtijX | SALESMAN | 1 | 2015-04-12 | 2000.00 | 400.00 | 345 | a9cedc94b4e995ec3afd89b3a4f36982 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.56 sec)

mysql> select * from emp where empno = 1234567;
+-----+-----+-----+-----+-----+-----+-----+-----+
| empno | ename | job      | ngr | hiredate |          |          |          |          |
| password |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1234567 | KUtijX | SALESMAN | 1 | 2015-04-12 | 2000.00 | 400.00 | 345 | a9cedc94b4e995ec3afd89b3a4f36982 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

允公允能 日新月异

NANKAI UNIVERSITY

02

查询缓存怎么用



02:00 00:00 00:00



查看查询缓存的状态



南开大学
Nankai University

```
show variables like 'query_cache%';
```

```
MariaDB [(none)]> show variables like 'query_cache%';
```

Variable_name	Value
query_cache_limit	1048576
query_cache_min_res_unit	4096
query_cache_size	1048576
query_cache_strip_comments	OFF
query_cache_type	OFF
query_cache_wlock_invalidate	OFF

```
6 rows in set (0.001 sec)
```

```
MariaDB [(none)]>
```

单条缓存限制

缓存存放单位

缓冲区总大小

query_cache_type:

0 / OFF: 不缓存结果;

1 / ON: 默认缓存所有有效的查询结果 (除了指定SQL_NO_CACHE的select语句);

2 / DEMAND: 只缓存指定SQL_CACHE的select语句。



开启查询缓存并设置大小



```
MariaDB [(none)]> show variables like 'query_cache%';
```

Variable_name	Value
query_cache_limit	1048576
query_cache_min_res_unit	4096
query_cache_size	1048576
query_cache_strip_comments	OFF
query_cache_type	OFF
query_cache_wlock_invalidate	OFF

```
6 rows in set (0.001 sec)
```

query_cache_wlock_invalidate

如果某个表被锁住，是否返回缓存中的数据，默认关闭，也是建议的。

一般情况，当客户端对MyISAM表进行WRITE锁定时，如果查询结果位于查询缓存中，则其它客户端未被锁定，可以对该表进行查询。将该变量设置为1，则可以对表进行WRITE锁定，使查询缓存内所有对该表进行的查询变得非法。这样当锁定生效时，可以强制其它试图访问表的客户端来等待。



开启查询缓存并设置大小



南开大学
Nankai University

```
mysql> set global query_cache_type = 1;  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> set global query_cache_size = 1024*1024*32;  
Query OK, 0 rows affected (0.00 sec)
```

1. 当 my.cnf 中, `query_cache_type = OFF`, 启动 mysql 服务后, 在 mysql 命令行中使用上面语句开启查询缓存, 会报错;
2. `query_cache_size` 的单位是字节, 且数值必须是 1024 的整数倍;
3. 如果 `query_cache_size=0`, 那即便你设置了 `query_cache_type = ON`, 查询缓存仍然是无法工作的。



查询缓存不生效的情况



1. 指定 SQL_NO_CACHE
2. Select 语句必须完全一致，字母
3. Select 包含动态语句 (now())、
4. 对应表的数据或者结构被更改，

```
mysql> select *, now() from emp where empno = 1234567;
+-----+-----+-----+-----+-----+-----+-----+-----+
| empno | ename | job   | mgr | hiredate | sal   | comm | deptno | e
password | now() |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1234567 | KUTiJX | SALESMAN | 1 | 2015-04-12 | 2000.00 | 400.00 | 345 | a
9cedc94b4e995ec3afd89b3a4f36982 | 2015-04-13 12:03:09 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.54 sec)

mysql> select *, now() from emp where empno = 1234567;
+-----+-----+-----+-----+-----+-----+-----+-----+
| empno | ename | job   | mgr | hiredate | sal   | comm | deptno | e
password | now() |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1234567 | KUTiJX | SALESMAN | 1 | 2015-04-12 | 2000.00 | 400.00 | 345 | a
9cedc94b4e995ec3afd89b3a4f36982 | 2015-04-13 12:03:11 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.52 sec)
```



查询缓存的缺点



南开大学
Nankai University

- 查询语句的 hash 计算和 hash 查找带来的资源消耗。如果将 `query_cache_type` 设置为 1（也就是 ON），那么 MySQL 会对每条接收到的 SELECT 类型的查询进行 hash 计算，然后查找这个查询的缓存结果是否存在。
- Query Cache 的失效问题。如果表的变更比较频繁，则会造成 Query Cache 的失效率非常高。表的变更不仅仅指表中的数据发生变化，还包括表结构或者索引的任何变化。
- 查询语句不同，但查询结果相同的查询都会被缓存，这样便会造成内存资源的过度消耗。查询语句的字符大小写、空格或者注释的不同，Query Cache 都会认为是不同的查询（因为他们的 hash 值会不同）。
- 相关系统变量设置不合理会造成大量的内存碎片，这样便会导致 Query Cache 频繁清理内存。



- 一个实际运行的论坛数据库的状态参数:
- QCache_hit 5280438
- QCache_insert 8008948
- Qcache_not_cache 95372
- select 8104159
- 可以看到, 数据库一共往Query Cache中写入了约800W次缓存, 但是实际命中的只有约500W次。也就是说, 每一个缓存的使用率约为0.66次。



- 碎片率
- 利用率
- 命中率

