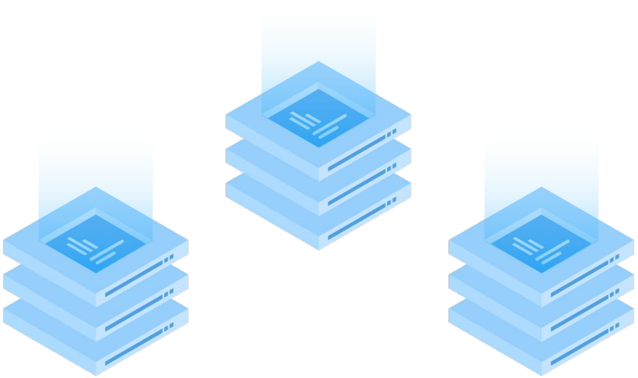
TiDB SQL 监控及典型的优化案例

qingli.miao

PingCAP TiDB



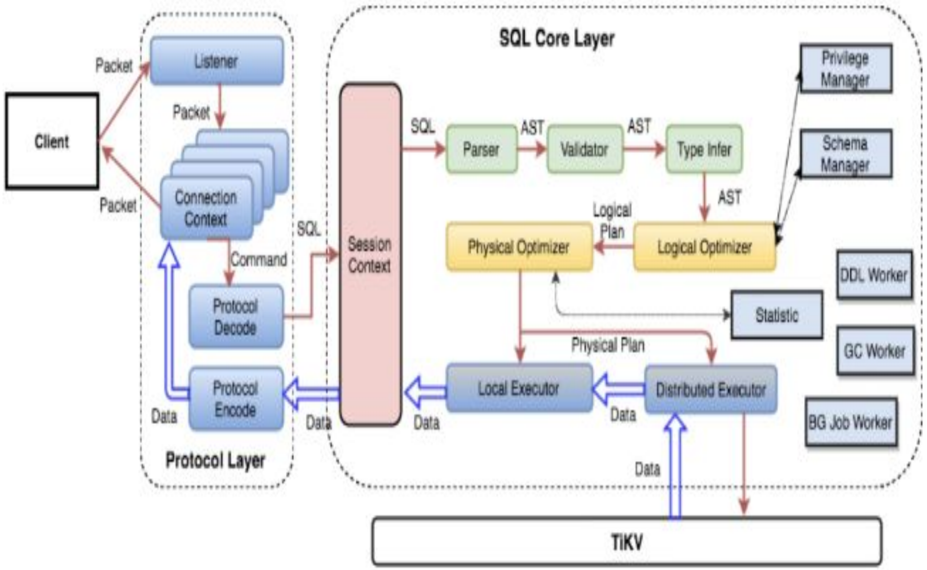
PingCAP PingCAP University

课程概要

- 课程背景：TiDB 作为通用分布式数据库产品，和其它数据库一样，也会存在 SQL 性能问题，轻则某个业务 SQL 执行慢，重则某个 SQL就能引发集群整体性能下降或不可用，本 课程从原理 + 实践的角度分析导致 slow-query 原因，并且针对常见的原因提供一些优化思路。
- 学习目标：
  - 了解 TiDB 读请求的执行流程。
  - 掌握基于 TiDB 慢查询的故障原因和定位思路。
  - 了解 TiDB 在分析场景下的参数优化。
- 受众：了解和使用过 TiDB 与 MySQL/MariaDB 的学员。
- 时长：1 课时（40 分钟）
- 关键知识点／大纲：
  - TiDB 读请求的执行流程。
  - slow query 慢查询定位及优化思路。
  - 优化案例。

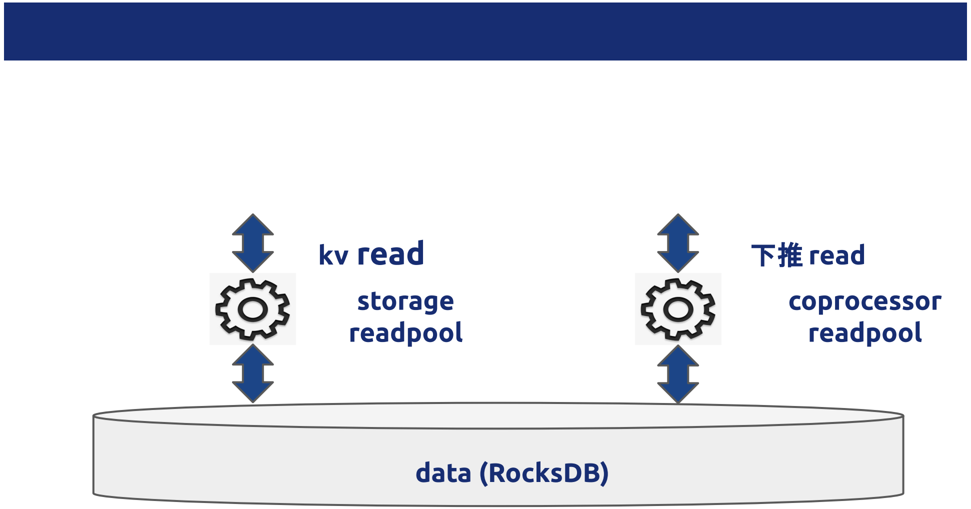
读请求执行流程

SQL 的一生



- 从客户端的 Socket 读取一条 SQL
- 获取一个 token
- 从 PD 获取 TSO（事务的时间戳）
- 使用 Parser 将 SQL parse 为 AST
- 将 AST compile 为执行计划
    - Logical Optimizer
    - Physical Optimizer
    - Execute Plan
- 根据对应的执行计划，最底层的 Executor 会根据这条 SQL 处理的 Key 范围构建出多个要下发到 TiKV 的请求，并通过 distsql 的 API 将这些请求分发到 TiKV
- TiKV 对结果做一些处理（包括 filter，limit 等）后，将中间结果集反馈给 TiDB
- TiDB 进行一些表关 联、聚合运算等最 终反回给 Client

SQL 的一生



TiKV 收到请求后，会将请求分为两类：

- storage read pool：负责主键或唯一索引点查。
- coprocessor：其余的请求。

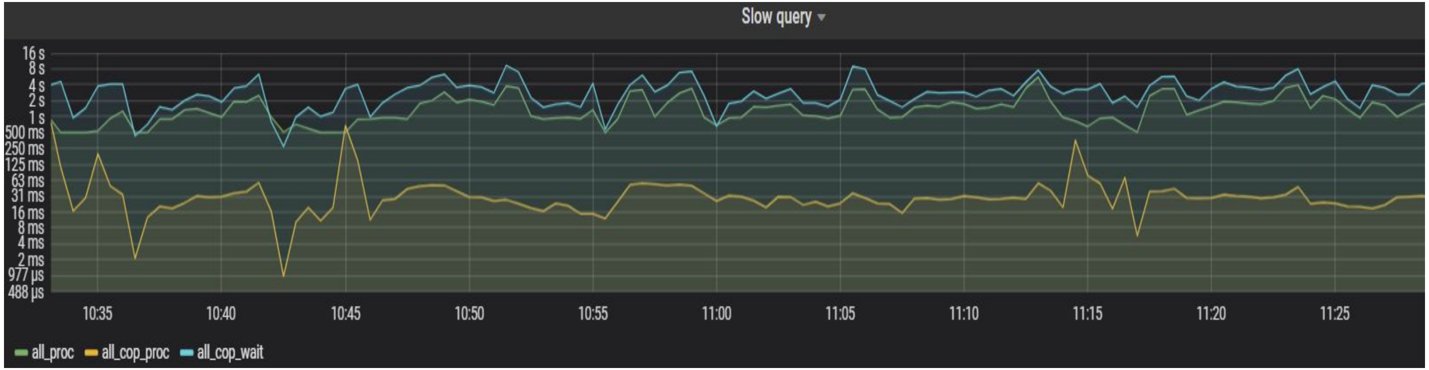
如何定位 slow query

slow query 产生的原因

- 按组件划分
  - TiDB
    - parse 慢
    - complie 慢
    - get token 慢
    - 执行计划不正确
  - PD
    - 获取 tso 慢
  - TiKV
    - 需要扫描大量的 key，耗时久
    - coprocessor cpu打满，资源等待
    - 读热点

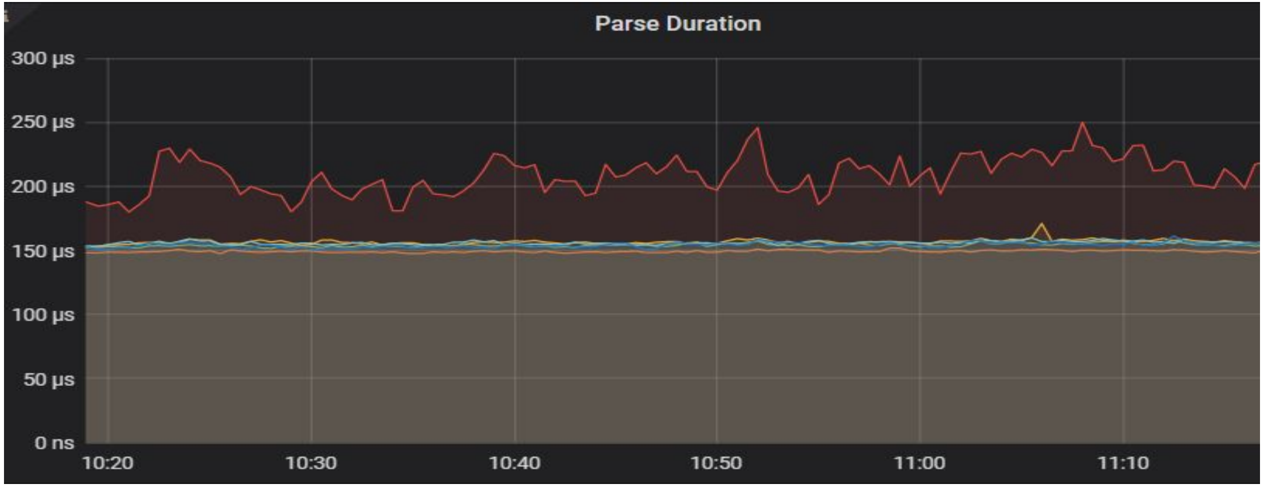
slow query 的获取渠道

- 集群监控上的 metrics 信息
- slow-query log，对标 Mysql 格式，支持市场上的 mysql 慢查询分析工具
- TiDB 的慢查询 SQL 内存表
- TiKV 节点日志 slow-query 反查



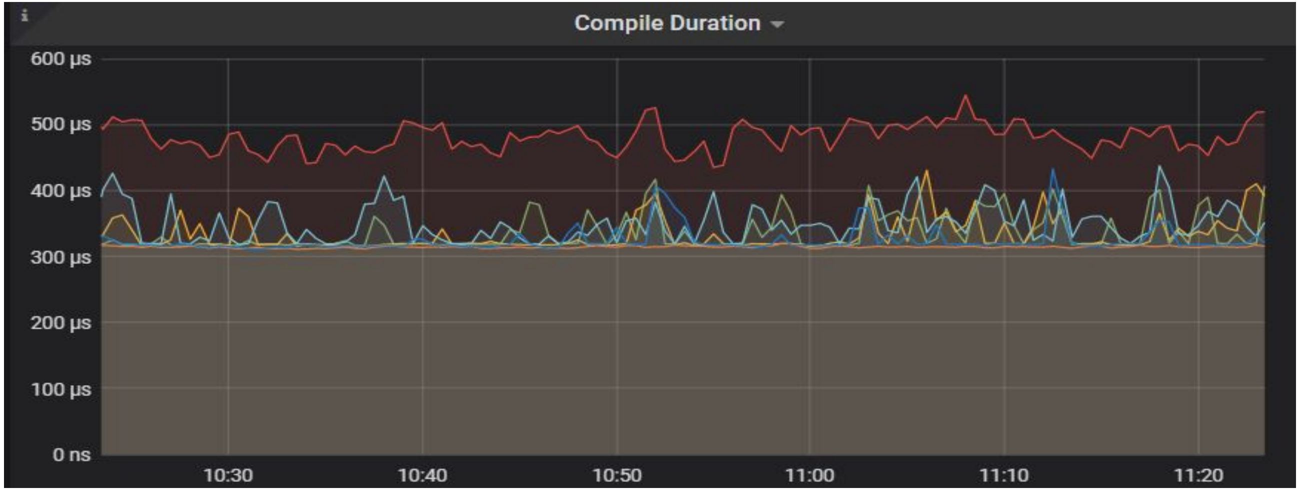
TiDB - Parse Metrics

- 位置 : TiDB -> Executor -> Parse Duration
- parse 慢可能原因 : TiDB 节点 CPU 压力大
- 图例 :



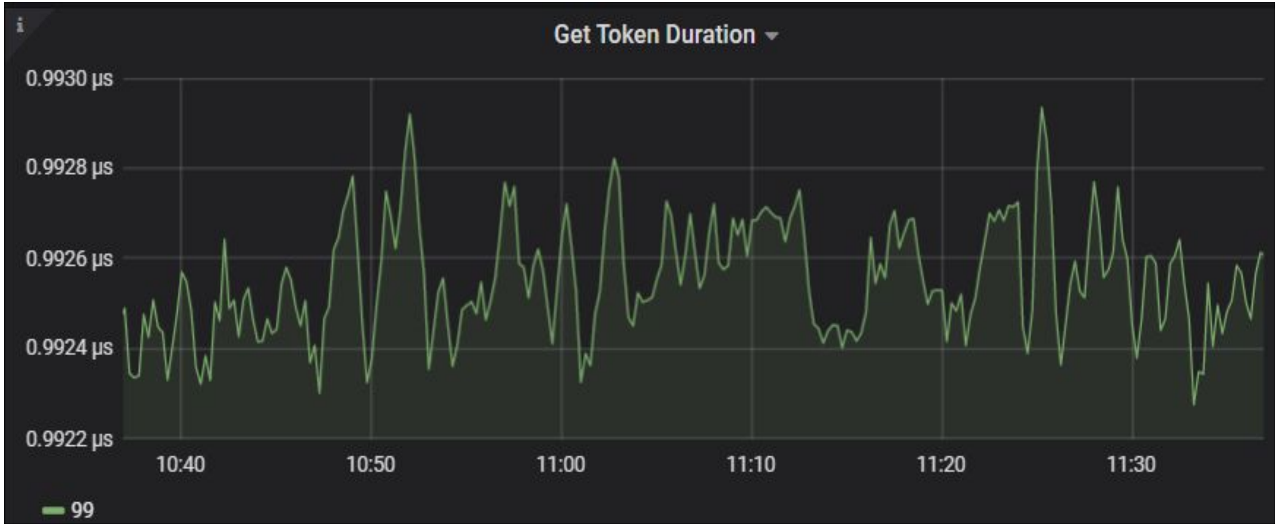
TiDB - Compile Metrics

- 位置 : TiDB -> Executor -> Compile Duration
- complie 慢的可能原因 :
  - TiDB 节点 cpu 压力大
  - in 子查询结果集多，跟参数 tidb_opt_insuqquery_unfold 有关（2.1）
- 图例 :



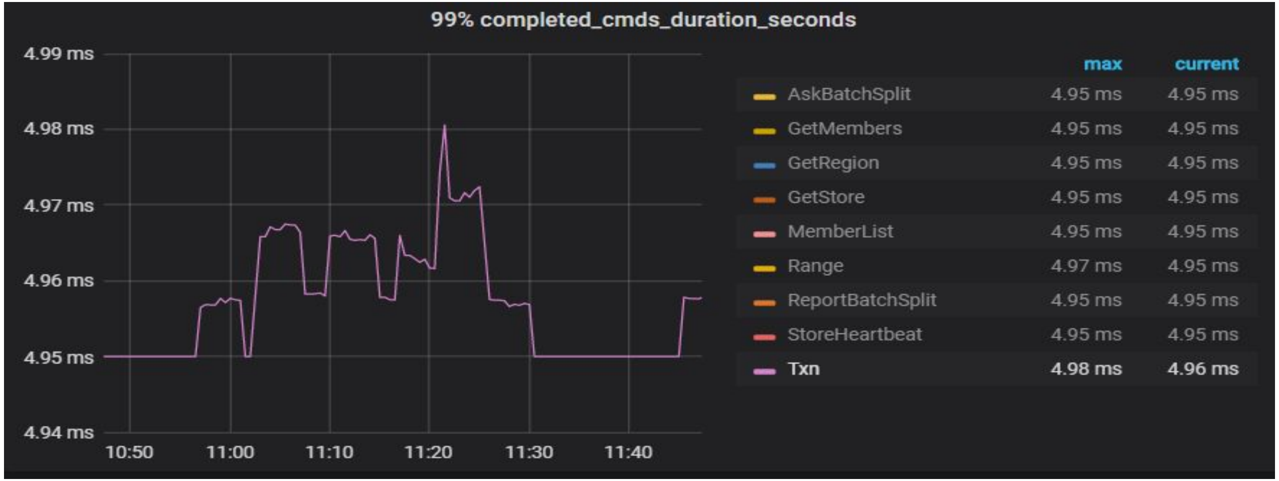
TiDB - Get Token Duration Metrics

- 位置：TiDB -> Server -> Get Token Duration
- 可能慢的原因：token 个数不足，需要 调整 token-limit
- 图例：



PD - tso

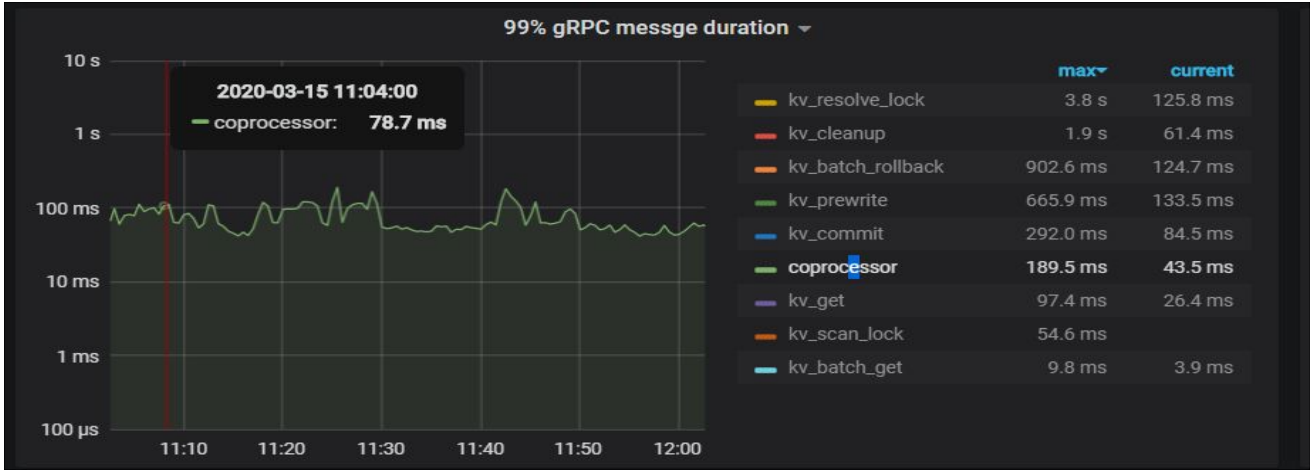
- 位置：PD -> Grpc -> 99% completed_cmd_duration_seconds -> txn
- 可能慢的原因：
  - PD Leader 切换
  - PD Leader 节点异常，包括 cpu、磁盘等。
- 图例：



	max	current
AskBatchSplit	4.95 ms	4.95 ms
GetMembers	4.95 ms	4.95 ms
GetRegion	4.95 ms	4.95 ms
GetStore	4.95 ms	4.95 ms
MemberList	4.95 ms	4.95 ms
Range	4.97 ms	4.95 ms
ReportBatchSplit	4.95 ms	4.95 ms
StoreHeartbeat	4.95 ms	4.95 ms
Txn	4.98 ms	4.96 ms

TiKV - Grpc Duration Metrics

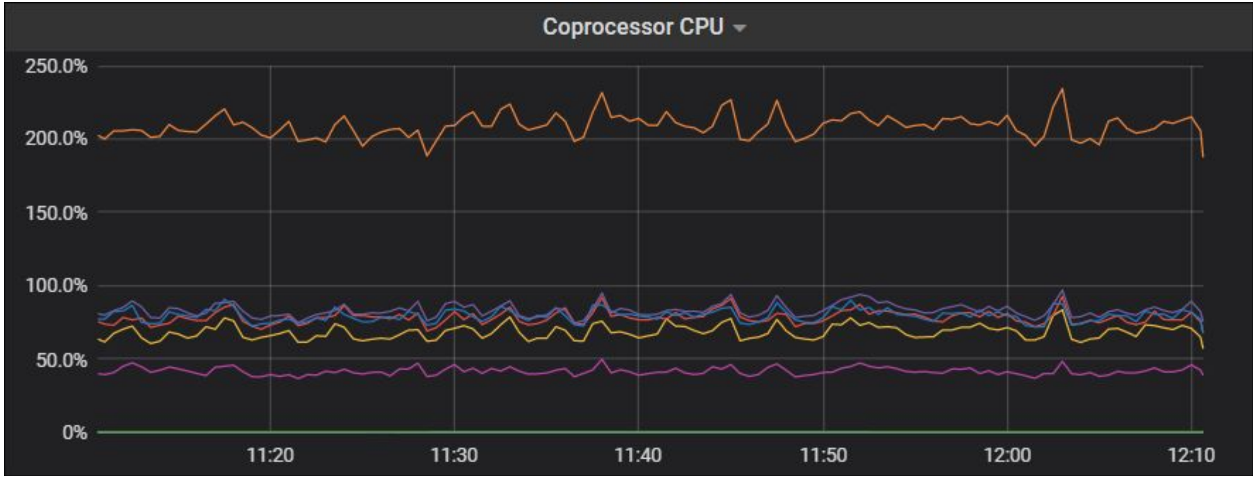
- 位置 : TiKV -> Grpc -> Coprocessor
- 可能慢的原因 :
  - ○ 需要扫描大量的 key
  - ○ Coprocessor cpu 打满
- 图例 :



	max▾	current
▬ kv_resolve_lock	3.8 s	125.8 ms
▬ kv_cleanup	1.9 s	61.4 ms
▬ kv_batch_rollback	902.6 ms	124.7 ms
▬ kv_prewrite	665.9 ms	133.5 ms
▬ kv_commit	292.0 ms	84.5 ms
▬ coprocessor	189.5 ms	43.5 ms
▬ kv_get	97.4 ms	26.4 ms
▬ kv_scan_lock	54.6 ms	
▬ kv_batch_get	9.8 ms	3.9 ms

TiKV - Coprocessor Cpu

- 位置：TiKV -> Thread Cpu -> Coprocessor Cpu
- 可能慢的原因：
  - 扫描大量的 key，将 cpu 资源占满
  - 读热点，造成 cpu 资源等待
- 图例：

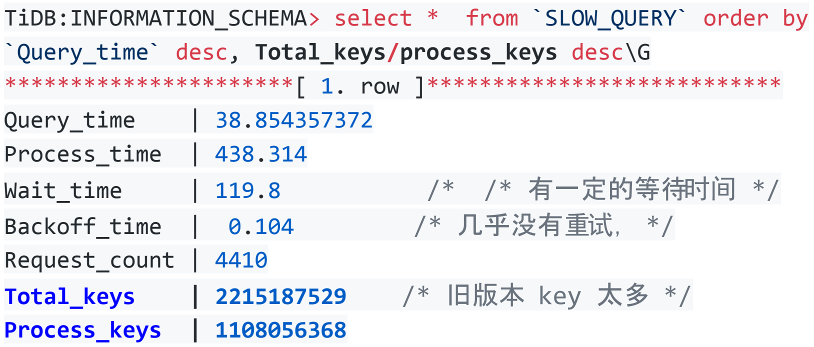


慢查询排查技能 - slow query log

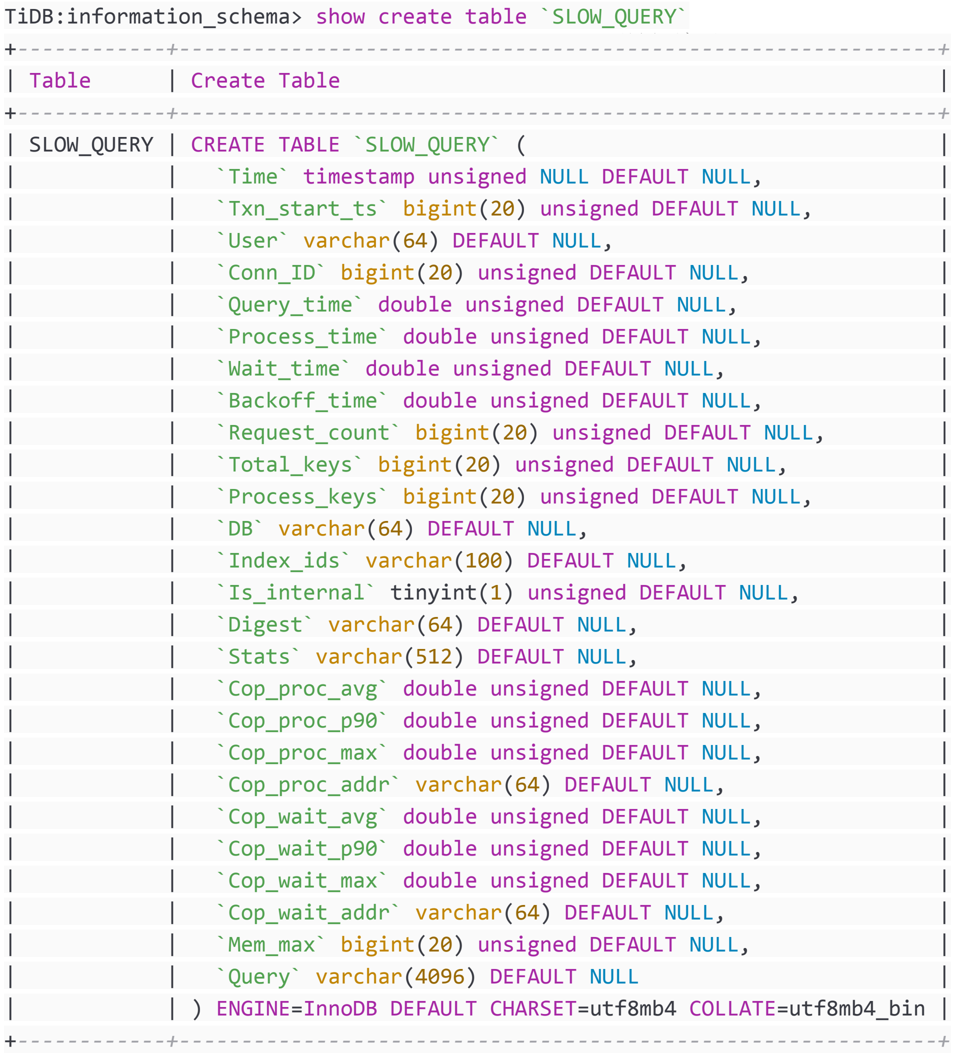
```
# Time: 2019-04-25-15:19:33.26029 +0800
# Txn_start_ts: 407942403346923524
# User: root@127.0.0.1
# Conn_ID: 1
# Query_time: 2.632671582
# Process_time: 0.079 Wait_time: 0.009 Backoff_time: 0.1 Request_count: 8 Total_keys: 20008 Process_keys: 20000
# DB: test
# Index_ids: [1]
# Is_internal: false
# Digest: edb16a8f28d9c48790925fd1c868fdae3feb49bc58481dda7df228625a5ba6e1
# Stats: t_wide:407941920305971202,t_slim:pseudo
# Cop_proc_avg: 0.009875 Cop_proc_p90: 0.018 Cop_proc_max: 0.018 Cop_proc_addr: 127.0.0.1:22160
# Cop_wait_avg: 0.001125 Cop_wait_p90: 0.002 Cop_wait_max: 0.002 Cop_wait_addr: 127.0.0.1:24160
# Mem_max: 195349
select count(1) from t_slim, t_wide where t_slim.c0>t_wide.c0 and t_slim.c1>t_wide.c1 and t_wide.c0 > 5000;
```

- Red color is related with **TiDB**
- Blue color is related with **TiKV/Coprocessor**

慢查询排查技能 - 内存表



```
TiDB:INFORMATION_SCHEMA> select *  from `SLOW_QUERY` order by
`Query_time` desc, Total_keys/process_keys desc\G
***********************[ 1. row ]****************************
Query_time    | 38.854357372
Process_time  | 438.314
Wait_time     | 119.8          /*  /* 有一定的等待时间 */
Backoff_time  |  0.104        /* 几乎没有重试，  */
Request_count | 4410
Total_keys    | 2215187529    /* 旧版本 key 太多 */
Process_keys  | 1108056368
```



```
TiDB:information_schema> show create table `SLOW_QUERY`
+------------+------------------------------------------------------------+
| Table      | Create Table                                               |
+------------+------------------------------------------------------------+
| SLOW_QUERY | CREATE TABLE `SLOW_QUERY` (                                |
|            |   `Time` timestamp unsigned NULL DEFAULT NULL,             |
|            |   `Txn_start_ts` bigint(20) unsigned DEFAULT NULL,         |
|            |   `User` varchar(64) DEFAULT NULL,                         |
|            |   `Conn_ID` bigint(20) unsigned DEFAULT NULL,              |
|            |   `Query_time` double unsigned DEFAULT NULL,               |
|            |   `Process_time` double unsigned DEFAULT NULL,             |
|            |   `Wait_time` double unsigned DEFAULT NULL,                |
|            |   `Backoff_time` double unsigned DEFAULT NULL,             |
|            |   `Request_count` bigint(20) unsigned DEFAULT NULL,        |
|            |   `Total_keys` bigint(20) unsigned DEFAULT NULL,           |
|            |   `Process_keys` bigint(20) unsigned DEFAULT NULL,         |
|            |   `DB` varchar(64) DEFAULT NULL,                           |
|            |   `Index_ids` varchar(100) DEFAULT NULL,                   |
|            |   `Is_internal` tinyint(1) unsigned DEFAULT NULL,          |
|            |   `Digest` varchar(64) DEFAULT NULL,                       |
|            |   `Stats` varchar(512) DEFAULT NULL,                       |
|            |   `Cop_proc_avg` double unsigned DEFAULT NULL,             |
|            |   `Cop_proc_p90` double unsigned DEFAULT NULL,             |
|            |   `Cop_proc_max` double unsigned DEFAULT NULL,             |
|            |   `Cop_proc_addr` varchar(64) DEFAULT NULL,                |
|            |   `Cop_wait_avg` double unsigned DEFAULT NULL,             |
|            |   `Cop_wait_p90` double unsigned DEFAULT NULL,             |
|            |   `Cop_wait_max` double unsigned DEFAULT NULL,             |
|            |   `Cop_wait_addr` varchar(64) DEFAULT NULL,                |
|            |   `Mem_max` bigint(20) unsigned DEFAULT NULL,              |
|            |   `Query` varchar(4096) DEFAULT NULL                       |
|            | ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_bin |
+------------+------------------------------------------------------------+
```

PingCAP

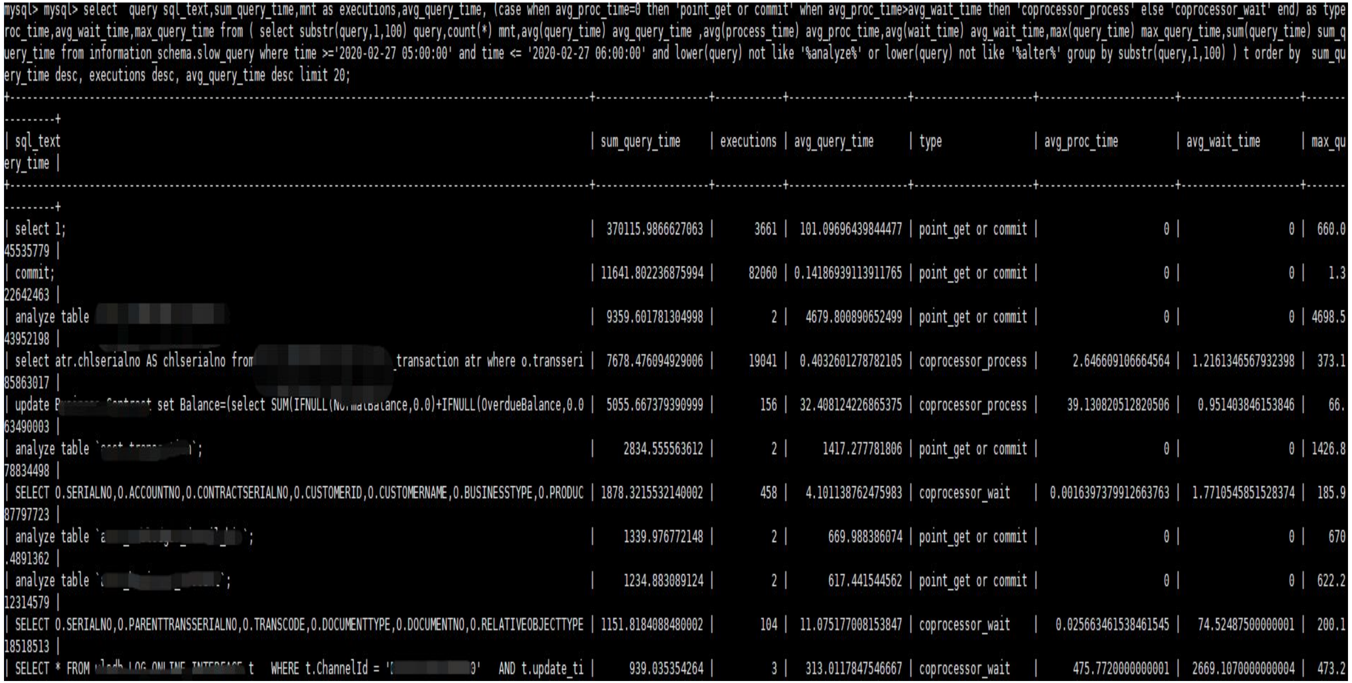
PingCAP
University

slow log 相关参数

- slow-threshold
  - 参数含义：输出慢 SQL 的耗时阈值，单位 ms，静态参数，需要重启 TiDB SERVER 生效。
  - 建议值：在 OLTP 系统，建议设置 50 ms 左右，OLTP 系统 SQL 一般有以下特点：SQL 短小、单次运行时间短、执行次数多；OLAP 系统可以适当放大点。
- query-log-max-len
  - 参数含义：日志记录的 SQL 长度，超过这个长度会截断输出，单位为字符，静态参数，需要重启 TiDB SERVER 生效。。
  - 建议值：4096
- tidb_slow_query_file
  - 参数含义：慢查询日志的文件名，默认值为 tidb_slow.log，可以存储 300 MB 内容，动态参数，可以通过会话变量生效。
  - 建议值：tidb_slow.log 300MB 以后就会重新生成一个文件，所以如果要定位某一段时间的 slow log，需要设置该环境变量。TiDB 通过 session 变量 tidb_slow_query_file 控制查询 INFORMATION_SCHEMA.SLOW_QUERY 时要读取和解析的文件，可通过修改 session 变量的值来查询其他慢查询日志文件的内容。

用 SQL 从多个维度查询 slow log 表

```
select
query sql_text,sum_query_time,mnt as
executions,avg_query_time,
(case when avg_proc_time=0 then 'point_get or
commit' when avg_proc_time>avg_wait_time then
'coprocessor_process' else 'coprocessor_wait' end)
as type,
avg_proc_time,avg_wait_time,max_query_time
from (
select substr(query,1,100) query,count(*)
mnt,avg(query_time) avg_query_time
,avg(process_time) avg_proc_time,avg(wait_time)
avg_wait_time,max(query_time)
max_query_time,sum(query_time) sum_query_time
from information_schema.slow_query
where time >='2020-02-27 05:00:00'
and time <= '2020-02-27 06:00:00'
and lower(query) not like '%analyze%'
or lower(query) not like '%alter%'
group by substr(query,1,100)
) t
order by
sum_query_time desc,
executions desc,
avg_query_time desc
limit 20;
```



```
mysql> mysql> select  query sql_text,sum_query_time,mnt as executions,avg_query_time, (case when avg_proc_time=0 then 'point_get or commit' when avg_proc_time>avg_wait_time then 'coprocessor_process' else 'coprocessor_wait' end) as type
roc_time,avg_wait_time,max_query_time from ( select substr(query,1,100) query,count(*) mnt,avg(query_time) avg_query_time ,avg(process_time) avg_proc_time,avg(wait_time) avg_wait_time,max(query_time) max_query_time,sum(query_time) sum_q
uery_time from information_schema.slow_query where time >='2020-02-27 05:00:00' and time <= '2020-02-27 06:00:00' and lower(query) not like '%analyze%' or lower(query) not like '%alter%' group by substr(query,1,100) ) t order by  sum_qu
ery_time desc, executions desc, avg_query_time desc limit 20;
+------------------------------------------------------------------------------------------------------+-------------------+------------+---------------------+---------------------+----------------------+--------------------+-------
---------+
| sql_text                                                                                             | sum_query_time    | executions | avg_query_time      | type                | avg_proc_time        | avg_wait_time      | max_qu
ery_time |
+------------------------------------------------------------------------------------------------------+-------------------+------------+---------------------+---------------------+----------------------+--------------------+-------
---------+
| select 1;                                                                                            |  370115.9866627063 |       3661 |  101.09696439844477 | point_get or commit |                    0 |                  0 |  660.0
45535779 |
| commit;                                                                                              | 11641.802236875994 |      82060 | 0.14186939113911765 | point_get or commit |                    0 |                  0 |    1.3
22642463 |
| analyze table [redacted]                                                                             |  9359.601781304998 |          2 |   4679.800890652499 | point_get or commit |                    0 |                  0 | 4698.5
43952198 |
| select atr.chlserialno AS chlserialno from [redacted]_transaction atr where o.transseri              |  7678.476094929006 |      19041 |  0.4032601278782105 | coprocessor_process |    2.646609106664564 | 1.2161346567932398 |  373.1
85863017 |
| update B[redacted] set Balance=(select SUM(IFNULL(N[redacted]alance,0.0)+IFNULL(OverdueBalance,0.0   |  5055.667379390999 |        156 |  32.408124226865375 | coprocessor_process |   39.130820512820506 |  0.951403846153846 |    66.
63490003 |
| analyze table `[redacted]`;                                                                          |     2834.555563612 |          2 |      1417.277781806 | point_get or commit |                    0 |                  0 | 1426.8
78834498 |
| SELECT O.SERIALNO,O.ACCOUNTNO,O.CONTRACTSERIALNO,O.CUSTOMERID,O.CUSTOMERNAME,O.BUSINESSTYPE,O.PRODUC | 1878.3215532140002 |        458 |   4.101138762475983 | coprocessor_wait    | 0.0016397379912663763 | 1.7710545851528374 |  185.9
87797723 |
| analyze table `a[redacted]`;                                                                         |     1339.976772148 |          2 |       669.988386074 | point_get or commit |                    0 |                  0 |    670
.4891362 |
| analyze table `[redacted]`;                                                                          |     1234.883089124 |          2 |       617.441544562 | point_get or commit |                    0 |                  0 |  622.2
12314579 |
| SELECT O.SERIALNO,O.PARENTTRANSSERIALNO,O.TRANSCODE,O.DOCUMENTTYPE,O.DOCUMENTNO,O.RELATIVEOBJECTTYPE | 1151.8184088480002 |        104 |  11.075177008153847 | coprocessor_wait    |  0.025663461538461545 |  74.52487500000001 |  200.1
18518513 |
| SELECT * FROM [redacted] t   WHERE t.ChannelId = '[redacted]0'   AND t.update_ti                     |      939.035354264 |          3 |   313.0117847546667 | coprocessor_wait    |     475.7720000000001 | 2669.1070000000004 |  473.2
```

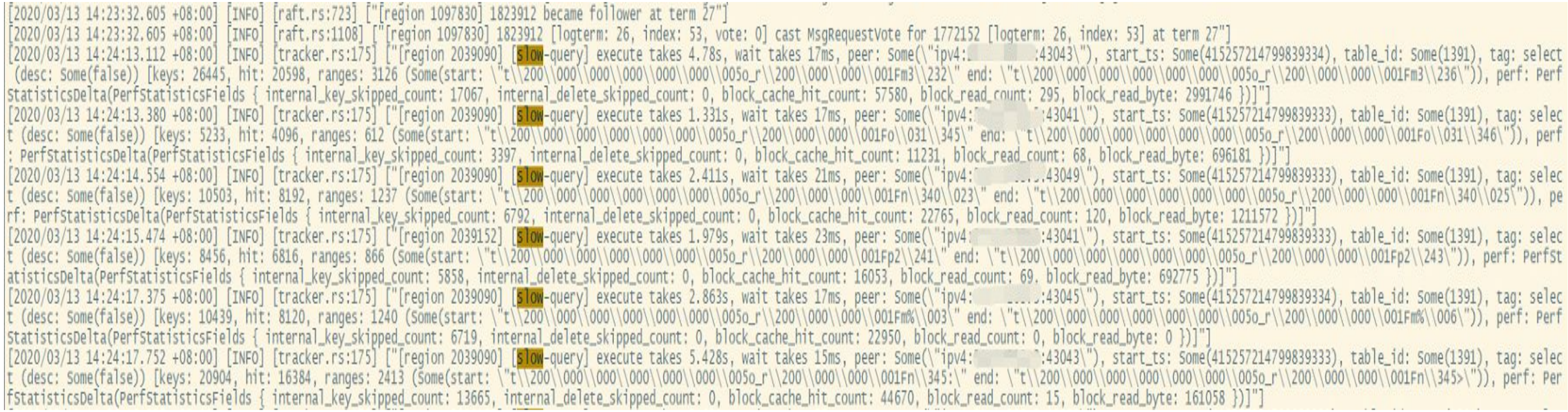
PingCAP

PingCAP University

TiKV 日志反查 slow log

在某个 SQL 执行导致的 TiKV cpu 异常高的场合下，使用 TiKV 日志可以快速的定位问题 SQL。
TiKV 日志中记录了 slow-query 的几个关键信息：

- ipv4：发出请求的 TiDB 地址，通过该地址可以确定 SQL 在哪个 TiDB 节点执行。
- start_ts：事务的 start_ts，通过 start_ts 反查 slow log，快速定位问题 SQL。
- table_id：查询的那张表，可以通过 information_schema.tables 的 tidb_table_id 反查表名



```
[2020/03/13 14:23:32.605 +08:00] [INFO] [raft.rs:723] ["[region 1097830] 1823912 became follower at term 27"]
[2020/03/13 14:23:32.605 +08:00] [INFO] [raft.rs:1108] ["[region 1097830] 1823912 [logterm: 26, index: 53, vote: 0] cast MsgRequestVote for 1772152 [logterm: 26, index: 53] at term 27"]
[2020/03/13 14:24:13.112 +08:00] [INFO] [tracker.rs:175] ["[region 2039090] [slow-query] execute takes 4.78s, wait takes 17ms, peer: Some(\"ipv4:[redacted]:43043\"), start_ts: Some(415257214799839334), table_id: Some(1391), tag: select
 (desc: Some(false)) [keys: 26445, hit: 20598, ranges: 3126 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fm3\\232\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fm3\\236\")), perf: Perf
StatisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 17067, internal_delete_skipped_count: 0, block_cache_hit_count: 57580, block_read_count: 295, block_read_byte: 2991746 })]"]
[2020/03/13 14:24:13.380 +08:00] [INFO] [tracker.rs:175] ["[region 2039090] [slow-query] execute takes 1.331s, wait takes 17ms, peer: Some(\"ipv4:[redacted]:43041\"), start_ts: Some(415257214799839333), table_id: Some(1391), tag: selec
t (desc: Some(false)) [keys: 5233, hit: 4096, ranges: 612 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fo\\031\\345\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fo\\031\\346\")), perf
: PerfStatisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 3397, internal_delete_skipped_count: 0, block_cache_hit_count: 11231, block_read_count: 68, block_read_byte: 696181 })]"]
[2020/03/13 14:24:14.554 +08:00] [INFO] [tracker.rs:175] ["[region 2039090] [slow-query] execute takes 2.411s, wait takes 21ms, peer: Some(\"ipv4:[redacted].43049\"), start_ts: Some(415257214799839333), table_id: Some(1391), tag: selec
t (desc: Some(false)) [keys: 10503, hit: 8192, ranges: 1237 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fn\\340\\023\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fn\\340\\025\")), pe
rf: PerfStatisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 6792, internal_delete_skipped_count: 0, block_cache_hit_count: 22765, block_read_count: 120, block_read_byte: 1211572 })]"]
[2020/03/13 14:24:15.474 +08:00] [INFO] [tracker.rs:175] ["[region 2039152] [slow-query] execute takes 1.979s, wait takes 23ms, peer: Some(\"ipv4[redacted]:43041\"), start_ts: Some(415257214799839333), table_id: Some(1391), tag: selec
t (desc: Some(false)) [keys: 8456, hit: 6816, ranges: 866 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fp2\\241\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fp2\\243\")), perf: PerfSt
atisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 5858, internal_delete_skipped_count: 0, block_cache_hit_count: 16053, block_read_count: 69. block_read_byte: 692775 })]"]
[2020/03/13 14:24:17.375 +08:00] [INFO] [tracker.rs:175] ["[region 2039090] [slow-query] execute takes 2.863s, wait takes 17ms, peer: Some(\"ipv4:[redacted]:43045\"), start_ts: Some(415257214799839334), table_id: Some(1391), tag: selec
t (desc: Some(false)) [keys: 10439, hit: 8120, ranges: 1240 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fm%\\003\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fm%\\006\")), perf: Perf
StatisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 6719, internal_delete_skipped_count: 0, block_cache_hit_count: 22950, block_read_count: 0, block_read_byte: 0 })]"]
[2020/03/13 14:24:17.752 +08:00] [INFO] [tracker.rs:175] ["[region 2039090] [slow-query] execute takes 5.428s, wait takes 15ms, peer: Some(\"ipv4:[redacted]:43043\"), start_ts: Some(415257214799839333), table_id: Some(1391), tag: selec
t (desc: Some(false)) [keys: 20904, hit: 16384, ranges: 2413 (Some(start: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fn\\345:\" end: \"t\\200\\000\\000\\000\\000\\000\\005o_r\\200\\000\\000\\001Fn\\345>\")), perf: Per
fStatisticsDelta(PerfStatisticsFields { internal_key_skipped_count: 13665, internal_delete_skipped_count: 0, block_cache_hit_count: 44670, block_read_count: 15, block_read_byte: 161058 })]"]
```

使用方法总结

- 通过监控的相关 metrics 可以获取到 TiDB 集群的整体运行情况，比如某些 TiKV 节点 CPU 高不高等。对于定位慢查询来说可以有一个初步的认知，用 metrics 来判断那些时间段存在问题，可能存在那些问题。
- slow query log 及 slow log 内存表，可以从整体上观察 SQL 运行情况，比如 SQL 执行花了多长时间，执行了多少次，哪些 SQL 最耗时，哪些 SQL 占用资源最多等等。
- 但是 slow log 内存表还存在一定的局限性，因为每个 TiDB 有自己的一个 slow log 且 slow log file 到达 300 mb 后还会切换，所以使用起来还是有一定的局限性。
- TiKV 日志能够快速定位问题 SQL，对于只执行一次或几次的大 SQL 来说，非常容易定位问题。但是如果是单次执行快但是执行频率高的 SQL 来说，查看 TiKV 日志还是相对比较麻烦。
- 综上所述，结合的使用三个工具才能比较快速定位问题 SQL。

AP 场景加速

相关并发参数

tidb_distsql_scan_concurrency

这个变量用来设置 scan 操作的并发度,对于 AP 类应用，最大值建议不要超过所有 TiKV 节，的 CPU 核数

tidb_index_serial_scan_concurrency

这个变量用来设置顺序 scan 操作的并发度

tidb_index_lookup_concurrency

这个变量用来设置 index lookup 操作的并发度

tidb_index_lookup_join_concurrency

这个变量用来设置 index lookup join 算法的并发度

tidb_hash_join_concurrency

这个变量用来设置 hash join 算法的并发度

SQL 优化案例

TiDB SQL 优化需要注意的点

- where + col1 条件 or col2 条件，在 TiDB 中无法使用多个索引即 MySQL 的 index_merge 功能，会导致全表扫描，可以转换为 from xxx where col1 union from xxx where col2，这样就可以同时使用 col1 索引和 col2 索引。注意：需要注意本身 结果集是否有重复数据，若有重复数据 union 和 or 无法等价转换。
- index join 的被驱动表，不能有函数处理的过滤条件，类似 where year(xxx)='2019'，尽量不要在过滤条件增加对列的函数处理。（现在还不支持函数索引）
- 其余与通用数据库 SQL 一样，需要注意尽量减少 not in，not exits 使用，尽量用表关联。
- 亮点：TiDB 支持 null 及 like 'abc%' 使用索引，!=xxx 在 TiDB 也能转换为索引范围扫描，可以转换为 [-inf,xxx),(xxx,+inf]，在某些特定场景比较有意义。

Union + 函数改写案例

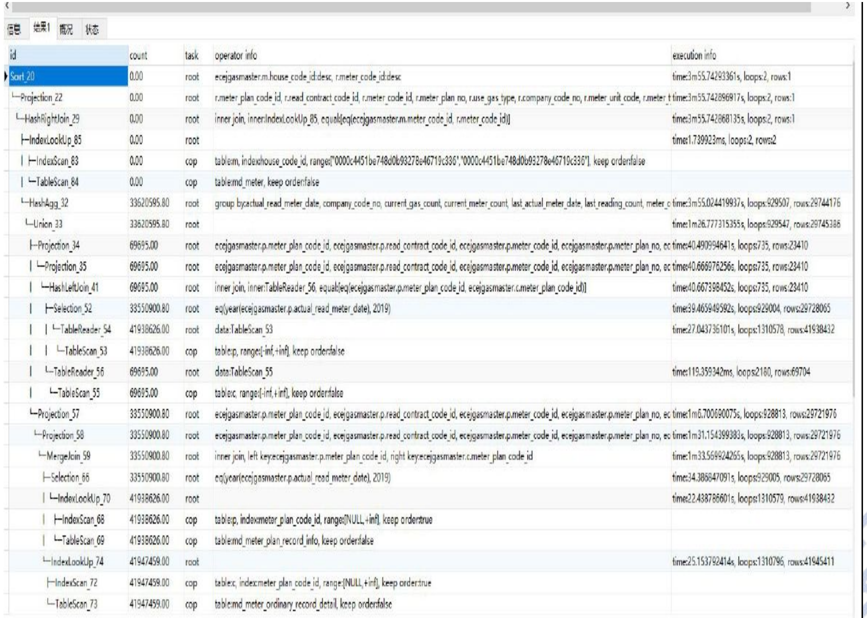
sql 文本：

```
select r.*,m.real_steel_grade,m.house_code_id,m.read_meter_type from [redacted] m inner join
      (
      select
p.meter_plan_code_id,read_contract_code_id,meter_code_id,meter_plan_no,use_gas_type,company_code_no,meter_unit_code,meter_type,meter_read_reason,

meter_reading_emp_name,meter_plan_status,plan_meter_reading_date,last_actual_meter_date,actual_read_meter_date,meter_round_count,meter_residual_count,meter_used_count
      ,0.0000 last_reading_count,0.0000 current_meter_count,0.0000 current_gas_count,0.00 payable_money
      from [redacted] p inner join [redacted] c on p.meter_plan_code_id =
c.meter_plan_code_id
      union
      select
p.meter_plan_code_id,read_contract_code_id,meter_code_id,meter_plan_no,use_gas_type,company_code_no,meter_unit_code,meter_type,meter_read_reason,

meter_reading_emp_name,meter_plan_status,plan_meter_reading_date,last_actual_meter_date,actual_read_meter_date,
      0.0000 meter_round_count, 0.0000 meter_residual_count, 0.0000 meter_used_count,
      last_reading_count,current_meter_count,current_gas_count,payable_money
      from [redacted] p inner join [redacted] c on p.meter_plan_code_id =
c.meter_plan_code_id
      ) r
       on m.meter_code_id=r.meter_code_id where  m.house_code_id in
     ('[redacted]')
        and year(actual_read_meter_date) = '2019'
      order by house_code_id desc, meter_code_id desc
```

执行计划：



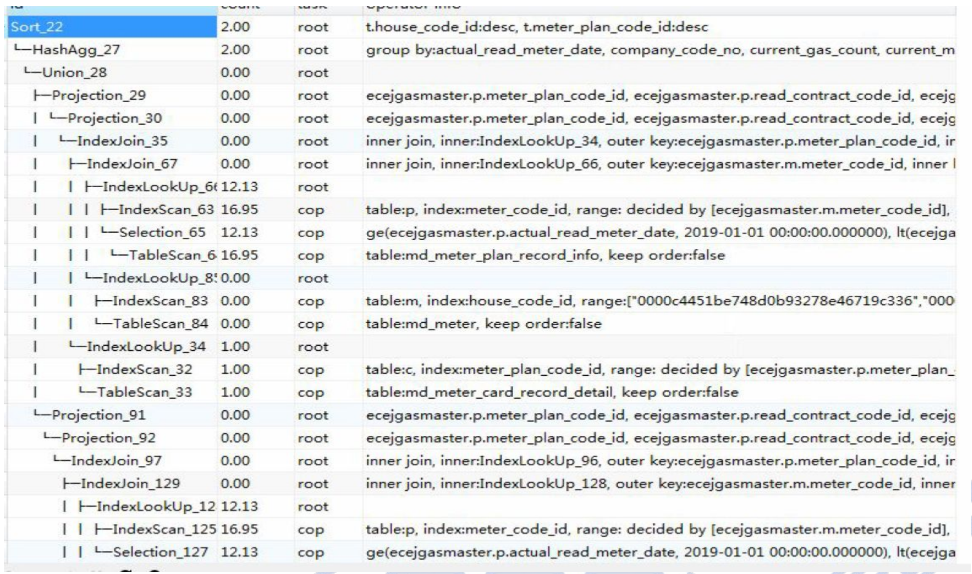
id	count	task	operator info	execution info
Sort_20	0.00	root	ecejgasmaster.m.house_code_id:desc, r.meter_code_id:desc	time:3m55.74293361s, loops:2, rows:1
└─Projection_22	0.00	root	r.meter_plan_code_id, r.read_contract_code_id, r.meter_code_id, r.meter_plan_no, r.use_gas_type, r.company_code_no, r.meter_unit_code, r.meter_t	time:3m55.742896917s, loops:2, rows:1
└─HashRightJoin_29	0.00	root	inner join, inner:IndexLookUp_85, equal:[eq(ecejgasmaster.m.meter_code_id, r.meter_code_id)]	time:3m55.742868135s, loops:2, rows:1
├─IndexLookUp_85	0.00	root		time:1.739923ms, loops:2, rows:2
│ ├─IndexScan_83	0.00	cop	table:m, index:house_code_id, range:["0000c4451be748d0b93278e46719c336","0000c4451be748d0b93278e46719c336"], keep order:false	
│ └─TableScan_84	0.00	cop	table:md_meter, keep order:false	
└─HashAgg_32	33620595.80	root	group by:actual_read_meter_date, company_code_no, current_gas_count, current_meter_count, last_actual_meter_date, last_reading_count, meter_c	time:3m55.024419937s, loops:929507, rows:29744176
└─Union_33	33620595.80	root		time:1m26.777315355s, loops:929547, rows:29745386
├─Projection_34	69695.00	root	ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejgasmaster.p.meter_code_id, ecejgasmaster.p.meter_plan_no, ec	time:40.490994641s, loops:735, rows:23410
│ └─Projection_35	69695.00	root	ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejgasmaster.p.meter_code_id, ecejgasmaster.p.meter_plan_no, ec	time:40.666976256s, loops:735, rows:23410
│ └─HashLeftJoin_41	69695.00	root	inner join, inner:TableReader_56, equal:[eq(ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.c.meter_plan_code_id)]	time:40.667398452s, loops:735, rows:23410
│ ├─Selection_52	33550900.80	root	eq(year(ecejgasmaster.p.actual_read_meter_date), 2019)	time:39.465949592s, loops:929004, rows:29728065
│ │ └─TableReader_54	41938626.00	root	data:TableScan_53	time:27.043736101s, loops:1310578, rows:41938432
│ │ └─TableScan_53	41938626.00	cop	table:p, range:[-inf,+inf], keep order:false	
│ └─TableReader_56	69695.00	root	data:TableScan_55	time:119.359342ms, loops:2180, rows:69704
│ └─TableScan_55	69695.00	cop	table:c, range:[-inf,+inf], keep order:false	
└─Projection_57	33550900.80	root	ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejgasmaster.p.meter_code_id, ecejgasmaster.p.meter_plan_no, ec	time:1m6.700690075s, loops:928813, rows:29721976
└─Projection_58	33550900.80	root	ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejgasmaster.p.meter_code_id, ecejgasmaster.p.meter_plan_no, ec	time:1m31.154399383s, loops:928813, rows:29721976
└─MergeJoin_59	33550900.80	root	inner join, left key:ecejgasmaster.p.meter_plan_code_id, right key:ecejgasmaster.c.meter_plan_code_id	time:1m33.569924265s, loops:928813, rows:29721976
├─Selection_66	33550900.80	root	eq(year(ecejgasmaster.p.actual_read_meter_date), 2019)	time:34.386847091s, loops:929005, rows:29728065
│ └─IndexLookUp_70	41938626.00	root		time:22.438786601s, loops:1310579, rows:41938432
│ ├─IndexScan_68	41938626.00	cop	table:p, index:meter_plan_code_id, range:[NULL,+inf], keep order:true	
│ └─TableScan_69	41938626.00	cop	table:md_meter_plan_record_info, keep order:false	
└─IndexLookUp_74	41947459.00	root		time:25.153792414s, loops:1310796, rows:41945411
├─IndexScan_72	41947459.00	cop	table:c, index:meter_plan_code_id, range:[NULL,+inf], keep order:true	
└─TableScan_73	41947459.00	cop	table:md_meter_ordinary_record_detail, keep order:false	

Union + 函数改写案例

改写后sql 文本：

```
select t.* from(
select xxx,m.read_meter_type
from  p inner join m on m.meter_code_id=p.meter_code_id
inner join md_meter_card_record_detail c on p.meter_plan_code_id
= c.meter_plan_code_id where
m.house_code_id in ('xxx')
union
select xxx,m.read_meter_type
from p inner join m on m.meter_code_id=p.meter_code_id
inner join md_meter_ordinary_record_detail c on
p.meter_plan_code_id = c.meter_plan_code_id
where
m.house_code_id in (xxxx)
)t
where
t.actual_read_meter_date >= '2019-01-01 00:00:00'
and t.actual_read_meter_date < '2020-01-01 00:00:00'
order by
t.house_code_id desc,
t.meter_plan_code_id desc
```

改写后执行计划：



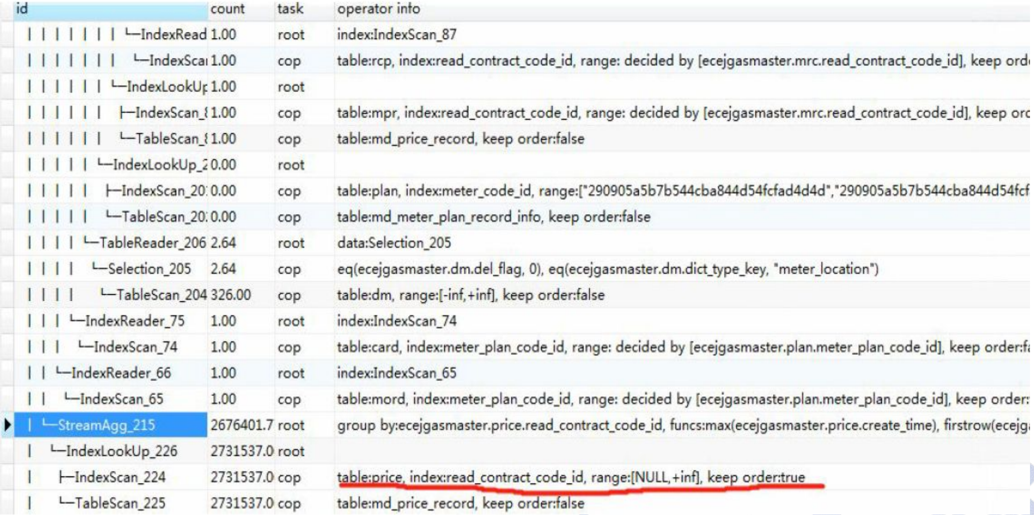
```
id                       count  task  operator info
Sort_22                  2.00   root  t.house_code_id:desc, t.meter_plan_code_id:desc
└─HashAgg_27             2.00   root  group by:actual_read_meter_date, company_code_no, current_gas_count, current_m
  └─Union_28             0.00   root
    ├─Projection_29      0.00   root  ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejg
    │ └─Projection_30    0.00   root  ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejg
    │   └─IndexJoin_35   0.00   root  inner join, inner:IndexLookUp_34, outer key:ecejgasmaster.p.meter_plan_code_id, ir
    │     ├─IndexJoin_67 0.00   root  inner join, inner:IndexLookUp_66, outer key:ecejgasmaster.m.meter_code_id, inner l
    │     │ ├─IndexLookUp_6 12.13 root
    │     │ │ ├─IndexScan_63 16.95 cop table:p, index:meter_code_id, range: decided by [ecejgasmaster.m.meter_code_id],
    │     │ │ └─Selection_65 12.13 cop ge(ecejgasmaster.p.actual_read_meter_date, 2019-01-01 00:00:00.000000), lt(ecejga
    │     │ │   └─TableScan_6 16.95 cop table:md_meter_plan_record_info, keep order:false
    │     │ └─IndexLookUp_8 0.00 root
    │     │   ├─IndexScan_83 0.00 cop table:m, index:house_code_id, range:["0000c4451be748d0b93278e46719c336","000
    │     │   └─TableScan_84 0.00 cop table:md_meter, keep order:false
    │     └─IndexLookUp_34 1.00 root
    │       ├─IndexScan_32 1.00 cop  table:c, index:meter_plan_code_id, range: decided by [ecejgasmaster.p.meter_plan_
    │       └─TableScan_33 1.00 cop  table:md_meter_card_record_detail, keep order:false
    └─Projection_91      0.00   root  ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejg
      └─Projection_92    0.00   root  ecejgasmaster.p.meter_plan_code_id, ecejgasmaster.p.read_contract_code_id, ecejg
        └─IndexJoin_97   0.00   root  inner join, inner:IndexLookUp_96, outer key:ecejgasmaster.p.meter_plan_code_id, ir
          ├─IndexJoin_129 0.00  root  inner join, inner:IndexLookUp_128, outer key:ecejgasmaster.m.meter_code_id, inner
          │ ├─IndexLookUp_12 12.13 root
          │ │ ├─IndexScan_125 16.95 cop table:p, index:meter_code_id, range: decided by [ecejgasmaster.m.meter_code_id],
          │ │ └─Selection_127 12.13 cop ge(ecejgasmaster.p.actual_read_meter_date, 2019-01-01 00:00:00.000000), lt(ecejga
```

相关子查询优化

sql 文本：

……
LEFT JOIN xxx mord ON mord.meter_plan_code_id = plan.meter_plan_code_id -WHERE c.customer_code_id = 'xxxx' AND h.house_code_id = 'xxxx' AND m.meter_code_id= 'xxxxx'
AND mpr.create_time = (
SELECT MAX(create_time) FROM xxx price WHERE price.read_contract_code_id = mrc.read_contract_code_id
)
AND plan.meter_plan_code_id = (
SELECT mpri.meter_plan_code_id FROM xxxxmpri WHERE mpri.meter_code_id = m.meter_code_id ORDER BY create_time DESC LIMIT 1

执行计划：



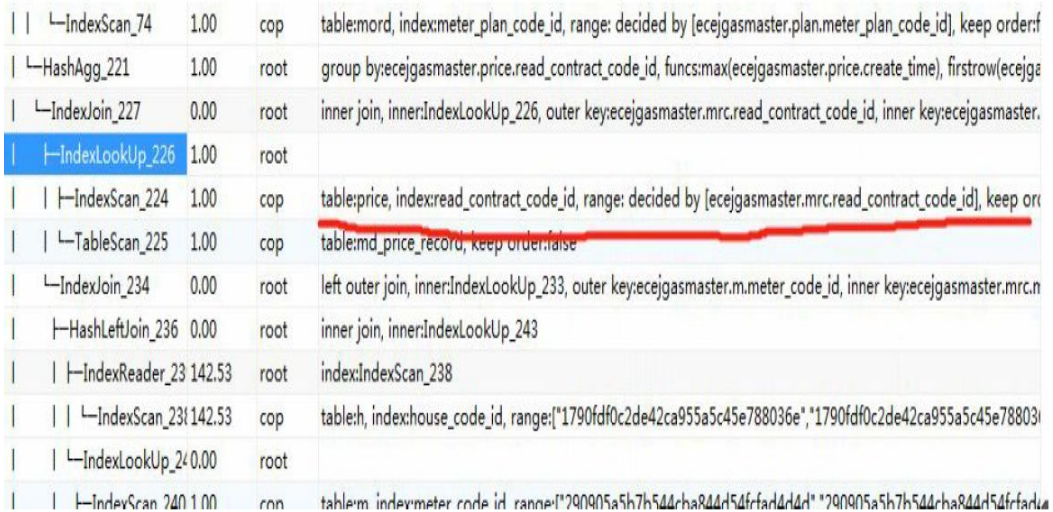
id	count	task	operator info
│ │ │ │ │ │ │ └─IndexRead	1.00	root	index:IndexScan_87
│ │ │ │ │ │ │ └─IndexScai	1.00	cop	table:rcp, index:read_contract_code_id, range: decided by [ecejgasmaster.mrc.read_contract_code_id], keep ord
│ │ │ │ │ │ └─IndexLookUp	1.00	root	
│ │ │ │ │ │ ├─IndexScan_	1.00	cop	table:mpr, index:read_contract_code_id, range: decided by [ecejgasmaster.mrc.read_contract_code_id], keep orc
│ │ │ │ │ │ └─TableScan_	1.00	cop	table:md_price_record, keep order:false
│ │ │ │ │ └─IndexLookUp_2	0.00	root	
│ │ │ │ │ ├─IndexScan_20	0.00	cop	table:plan, index:meter_code_id, range:["290905a5b7b544cba844d54fcfad4d4d","290905a5b7b544cba844d54fcf
│ │ │ │ │ └─TableScan_20	0.00	cop	table:md_meter_plan_record_info, keep order:false
│ │ │ │ └─TableReader_206	2.64	root	data:Selection_205
│ │ │ │ └─Selection_205	2.64	cop	eq(ecejgasmaster.dm.del_flag, 0), eq(ecejgasmaster.dm.dict_type_key, "meter_location")
│ │ │ │ └─TableScan_204	326.00	cop	table:dm, range:[-inf,+inf], keep order:false
│ │ │ └─IndexReader_75	1.00	root	index:IndexScan_74
│ │ │ └─IndexScan_74	1.00	cop	table:card, index:meter_plan_code_id, range: decided by [ecejgasmaster.plan.meter_plan_code_id], keep order:f
│ │ └─IndexReader_66	1.00	root	index:IndexScan_65
│ │ └─IndexScan_65	1.00	cop	table:mord, index:meter_plan_code_id, range: decided by [ecejgasmaster.plan.meter_plan_code_id], keep order:
│ └─StreamAgg_215	2676401.7	root	group by:ecejgasmaster.price.read_contract_code_id, funcs:max(ecejgasmaster.price.create_time), firstrow(ecejg
│ └─IndexLookUp_226	2731537.0	root	
│ ├─IndexScan_224	2731537.0	cop	table:price, index:read_contract_code_id, range:[NULL,+inf], keep order:true
│ └─TableScan_225	2731537.0	cop	table:md_price_record, keep order:false

相关子查询优化

改写后 sql 文本：

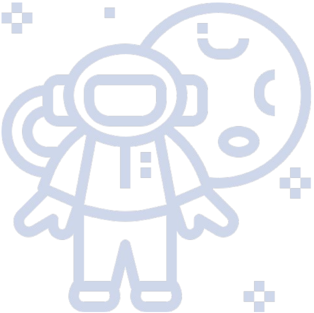
```
......
LEFT JOIN xxx mord ON mord.meter_plan_code_id =
plan.meter_plan_code_id
inner join(SELECT MAX(create_time) as
max_create_time,read_contract_code_id FROM (
SELECT price.create_time,price.read_contract_code_id FROM xxxx price
JOIN(SELECT mrc.read_contract_code_id from xxxx AS h left JOIN xxxx
AS m ON m.house_code_id = h.house_code_id left JOIN xxx AS mrc ON
mrc.meter_code_id = m.meter_code_id where
h.house_code_id = 'xxxx'
AND m.meter_code_id = 'xxx')
tmp_t WHERE price.read_contract_code_id =
tmp_t.read_contract_code_id) price
GROUP BY price.read_contract_code_id) price
ON mpr.create_time = price.max_create_time
and price.read_contract_code_id = mrc.read_contract_code_id
WHERE
c.customer_code_id = 'xxx'
AND h.house_code_id = 'xxxx'
AND m.meter_code_id = 'xxxx'
```

改写后的执行计划：



```
| |   └─IndexScan_74      1.00    cop    table:mord, index:meter_plan_code_id, range: decided by [ecejgasmaster.plan.meter_plan_code_id], keep order:f
| └─HashAgg_221           1.00    root   group by:ecejgasmaster.price.read_contract_code_id, funcs:max(ecejgasmaster.price.create_time), firstrow(ecejga
|   └─IndexJoin_227       0.00    root   inner join, inner:IndexLookUp_226, outer key:ecejgasmaster.mrc.read_contract_code_id, inner key:ecejgasmaster.
|     ├─IndexLookUp_226   1.00    root
|     | ├─IndexScan_224   1.00    cop    table:price, index:read_contract_code_id, range: decided by [ecejgasmaster.mrc.read_contract_code_id], keep or
|     | └─TableScan_225   1.00    cop    table:md_price_record, keep order:false
|     └─IndexJoin_234     0.00    root   left outer join, inner:IndexLookUp_233, outer key:ecejgasmaster.m.meter_code_id, inner key:ecejgasmaster.mrc.m
|       ├─HashLeftJoin_236 0.00   root   inner join, inner:IndexLookUp_243
|       | ├─IndexReader_23 142.53 root   index:IndexScan_238
|       | | └─IndexScan_23 142.53 cop    table:h, index:house_code_id, range:["1790fdf0c2de42ca955a5c45e788036e","1790fdf0c2de42ca955a5c45e78803
|       | └─IndexLookUp_240.00    root
|       |   ├─IndexScan_240 1.00  cop    table:m, index:meter_code_id, range:["290905a5b7b544cba844d54fcfad4d4d","290905a5b7b544cba844d54fcfad4
```

谢谢！



课程作业

- 问题1:slow query 内存表 (   ) 字段代表着 SQL 实际的查询时间。
  A. query_time   B.process_time   C.wait_time   D.execute_time

- 问题2:点查请求在 TiKV 中是通过 (   ) 执行的 。
  A. coprocessor   B.key range   C.storage read pool   D.parse

- 问题3:在 TiDB中 以下哪个条件无法走索引？(   )
  A.like ‘abc%’   B.!=   C.in   D.or

- 问题4:Get Token duration 是指？( )
  A. SQL 解析时间   B. 获取 PD tso 时间 C.获取 token 的时间 D.生成执行计划时间

- 问题5:某几个 TiKV 节点在 10:00 - 11:00 CPU 使用率在 90%以上，如何定位是哪个或那些 SQL 影响到了？