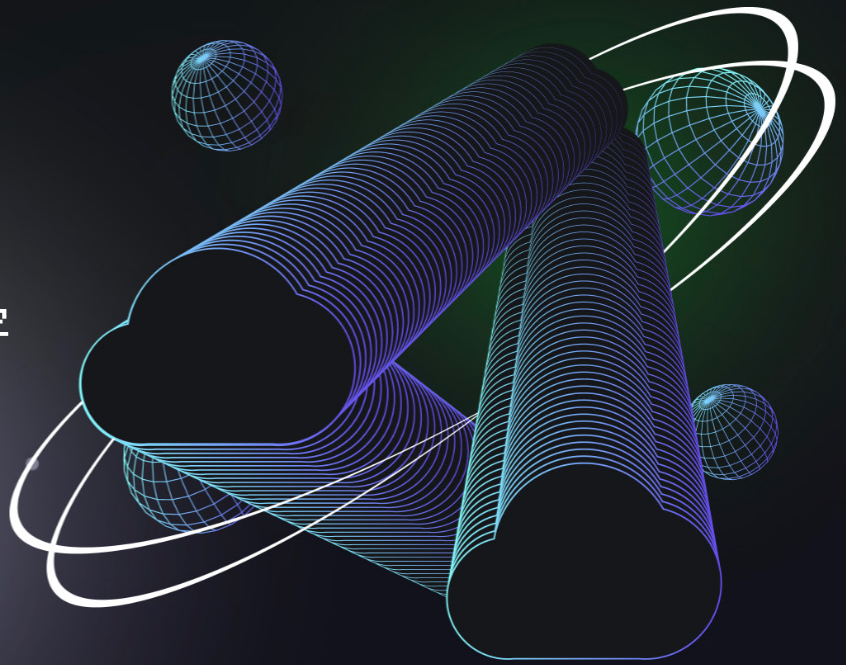
TiDB 黑客马拉松 | TiDB HACKATHON 2021 PingCAP TiDB

项目名称：TiDB Visual Plan

主讲人：张粲宇



团队介绍 - TiVP

PingCAP TiDB

队长：张粲宇 @yiwen92

队员：吴晓菊 @chrysan

李霞 @Tammyxia

陈元 @92hackers

*+_}—T.*i@(D#&*)@!B—*

“把复杂交给我们，把简单还给你”

TiDB ☺

TiDB HACKATHON 2021

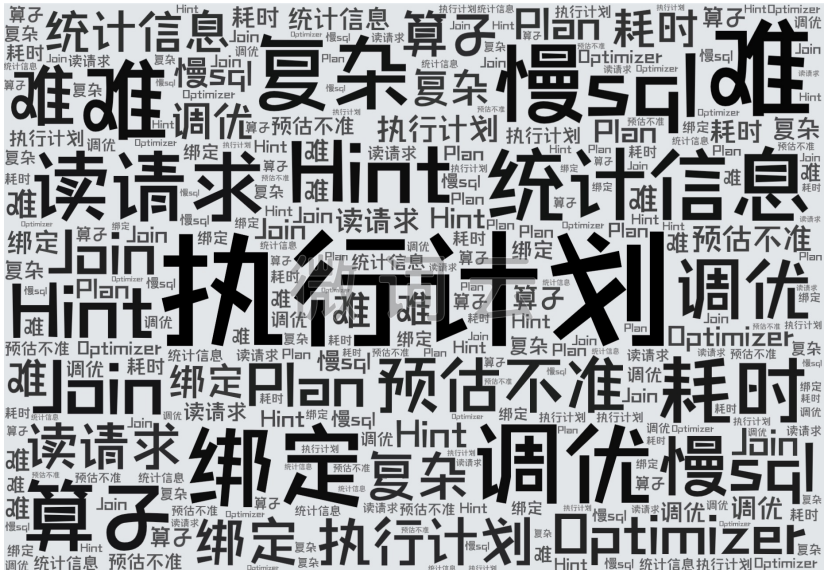
PingCAP TiDB

项目背景

TiDB HACKATHON 2021

为什么做这个项目？

Reason ①： TiDB SQL 调优的需求很大



Fact 1：慢 SQL 调优是 DBA **最头疼**的问题之一，甚至是同类竞对间的**胜负手**

Fact 2：TiDB Server 客户申请的原厂支持里，与**执行计划**（Plan）相关的问题过半

为什么做这个项目？

PingCAP TiDB

Reason ②：SQL，尤其慢 SQL 的执行计划难懂

```
mysql> explain select * from t join t1 on t.a = t1.b where t.a = 100 or t1.b = 200;
+-----------------------------------+---------+-----------+-------------------+-----------------------------------------------------------------------------------------------+
| id                                | estRows | task      | access object     | operator info                                                                                 |
+-----------------------------------+---------+-----------+-------------------+-----------------------------------------------------------------------------------------------+
| MergeJoin_9                       | 1.60    | root      |                   | inner join, left key:test.t.a, right key:test.t1.b, other cond:or(eq(test.t.a, 100), eq(test.t1.b, 200)) |
| ├─Projection_49(Build)            | 2.00    | root      |                   | test.t1.a, test.t1.b                                                                          |
| │ └─IndexLookUp_48                | 2.00    | root      |                   |                                                                                               |
| │   ├─Selection_47(Build)         | 2.00    | cop[tikv] |                   | not(isnull(test.t1.b))                                                                        |
| │   │ └─IndexRangeScan_45         | 2.00    | cop[tikv] | table:t1, index:b(b) | range:[100,100], [200,200], keep order:true                                                |
| │   └─TableRowIDScan_46(Probe)    | 2.00    | cop[tikv] | table:t1          | keep order:false                                                                              |
| └─Projection_44(Probe)            | 2.00    | root      |                   | test.t.a, test.t.b                                                                            |
|   └─IndexLookUp_43                | 2.00    | root      |                   |                                                                                               |
|     ├─Selection_42(Build)         | 2.00    | cop[tikv] |                   | not(isnull(test.t.a))                                                                         |
|     │ └─IndexRangeScan_40         | 2.00    | cop[tikv] | table:t, index:a(a) | range:[100,100], [200,200], keep order:true                                                |
|     └─TableRowIDScan_41(Probe)    | 2.00    | cop[tikv] | table:t           | keep order:false                                                                              |
+-----------------------------------+---------+-----------+-------------------+-----------------------------------------------------------------------------------------------+
11 rows in set (0.00 sec)
```

Fact 1：SQL 是一种声明性语言，观察执行计划是排查执行效率的**唯一手段**

Fact 2：慢 SQL 往往比较长，Explain 出来的执行计划**异常复杂**会**影响效率**

TiDB HACKATHON 2021

为什么做这个项目？


回复 张粲宇：比如这个执行计划是不是就很花时间去看？
是的，我基本是根据 exection info 中的耗时去找哪块耗时时间长
嗯嗯，你觉得这个执行计划大概需要花多长时间去理解和完全读懂？
甚至说就是很难完全读懂...
Support
有查询 sql 和表结构辅助的情况下会更容易弄明白执行计划在干嘛，如果完全看执行计划就比较费时间了
执行计划里面还是有一些东西不是那么清楚的吧
嗯嗯，首先这个执行计划是不是很大，即便像你是比较有经验的，通篇阅读下来应该也挺耗时间的吧？
这个倒不是主要问题
主要的问题是这玩意一行太长了，左右拖来拖去的
DBA
嗯嗯是的，explain 出来的信息你觉得简明清晰嘛
Dev
“明确显示耗时最长或代价最大的执行路径”这个explain不太明确 需要自己一行一行找
嗯嗯，如果你要理解这个执行计划，是否需要把这个按步骤，或者按图形进行个转义？
有工具最好啊。。没工具就在脑子里想想

PingCAP TiDB

为什么做这个项目？

Talk is cheap, show you the shit !


```
| Projection_40                      | 454128.34    | 0      | root     |                                                   | time:669.2µs, loops:1,
                                     | Column#129, Column#130, Column#136, Column#137, Column#138, merchant.mht_trade_request.trans_id, merchant.mht_trade_request.sub_trans_code, Column#128, merchant.mht_trad
e_request.asset_acct_id, merchant.mht_ta_user_account.trade_acct, merchant.mht_product_ta_info.ta_no, merchant.mht_user_account.customer_name, merchant.mht_trade_request.product_id, Column#139, Column#155, Column#133, Col
umn#272, Column#273, Column#274, Column#275, Column#276, Column#151, Column#152, Column#149
                                     | 169.5 KB   | N/A     |
| └HashAgg_41                        | 454128.34    | 0      | root     |                                                   | time:553.4µs, loops:1,
, concurrency:5, task_num:0, tot_wait:2.491273ms, tot_exec:0s, tot_time:2.496748ms, max:504.972µs, p95:504.972µs}, final_worker:{wall_time:554.785µs, concurrency:5, task_num:0, tot_wait:2.598492ms, tot_exec:6.35µs, tot_ti
0.906µs}                             | group by:Column#334, Column#335, Column#336, Column#337, Column#338, Column#339, Column#340, Column#341, Column#342, Column#343, Column#344, Column#345, Column#346, Colu
mn#350, Column#351, Column#352, Column#353, funcs:group_concat(Column#306 separator ",")->Column#269, funcs:group_concat(Column#307 separator ",")->Column#270, funcs:group_concat(Column#308 separator ",")->Column#271, fun
",")->Column#272, funcs:group_concat(Column#310 separator ",")->Column#273, funcs:group_concat(Column#311 separator ",")->Column#274, funcs:group_concat(Column#312 separator ",")->Column#275, funcs:sum(Column#313)->Colum
olumn#128, funcs:firstrow(Column#315)->Column#129, funcs:firstrow(Column#316)->Column#130, funcs:firstrow(Column#317)->Column#133, funcs:firstrow(Column#318)->Column#136, funcs:firstrow(Column#319)->Column#137, funcs:firs
firstrow(Column#321)->Column#139, funcs:firstrow(Column#322)->Column#149, funcs:firstrow(Column#323)->Column#151, funcs:firstrow(Column#324)->Column#152, funcs:firstrow(Column#325)->Column#155, funcs:firstrow(Column#326)-
, funcs:firstrow(Column#327)->merchant.mht_trade_request.sub_trans_code, funcs:firstrow(Column#328)->merchant.mht_trade_request.user_id, funcs:firstrow(Column#329)->merchant.mht_trade_request.asset_acct_id, funcs:firstrow
uest.product_id, funcs:firstrow(Column#331)->merchant.mht_user_account.customer_name, funcs:firstrow(Column#332)->merchant.mht_product_ta_info.ta_no, funcs:firstrow(Column#333)->merchant.mht_ta_user_account.trade_acct | 2
|   └Projection_197                  | 454128.34    | 0      | root     |                                                   | time:490.9µs, loops:1,
                                     | Column#153, Column#154, Column#142, Column#143, Column#140, Column#141, cast(Column#134, var_string(20))->Column#312, cast(Column#134, decimal(37,0) BINARY)->Column#313,
 Column#133, Column#136, Column#137, Column#138, Column#139, Column#149, Column#151, Column#152, Column#155, merchant.mht_trade_request.trans_id, merchant.mht_trade_request.sub_trans_code, merchant.mht_trade_request.user_
_acct_id, merchant.mht_trade_request.product_id, merchant.mht_user_account.customer_name, merchant.mht_product_ta_info.ta_no, merchant.mht_ta_user_account.trade_acct, Column#128, Column#129, Column#136, merchant.mht_trade
e_request.sub_trans_code, Column#155, merchant.mht_trade_request.user_id, merchant.mht_trade_request.asset_acct_id, merchant.mht_user_account.customer_name, merchant.mht_trade_request.product_id, Column#139, Column#130, C
ta_user_account.trade_acct, merchant.mht_product_ta_info.ta_no, Column#133, Column#151, Column#152, Column#149 | 217.6 KB  | N/A    |
|     └Projection_42                 | 454128.34    | 0      | root     |                                                   | time:315.7µs, loops:1,
                                     | Column#128, Column#129, Column#130, Column#133, Column#134, Column#136, Column#137, Column#138, Column#139, Column#140, Column#141, Column#142, Column#143, Column#149, C
olumn#154, Column#155, merchant.mht_trade_request.trans_id, merchant.mht_trade_request.sub_trans_code, merchant.mht_trade_request.user_id, merchant.mht_trade_request.asset_acct_id, merchant.mht_trade_request.product_id, m
me, merchant.mht_product_ta_info.ta_no, merchant.mht_ta_user_account.trade_acct
                                     | 153.1 KB   | N/A     |
|       └HashJoin_43                 | 454128.34    | 0      | root     |                                                   | time:202.1µs, loops:1,
ch:1.43ms, build:0s}
                                     | inner join, equal:[eq(merchant.mht_trade_request.request_id, Column#129)]

                                     | 0 Bytes    | 0 Bytes |
|         ├Union_112(Build)          | 5954.12      | 0      | root     |                                                   | time:1.3ms, loops:1
                                     |

                                     | N/A        | N/A     |
|         │ ├Projection_113          | 2977.06      | 0      | root     |                                                   | time:438.4µs, loops:1,
                                     | merchant.mht_trade_cash_out_info.cash_out_id, merchant.mht_trade_cash_out_info.request_id, merchant.mht_trade_cash_out_info.step_no, merchant.mht_trade_cash_out_info.cas
ut_info.cash_amt, merchant.mht_trade_cash_out_info.cash_out_date, merchant.mht_trade_cash_out_info.real_cash_out_date, merchant.mht_trade_cash_out_info.advance_natural_days, merchant.mht_trade_cash_out_info.cash_out_statu
inst_account_no, merchant.mht_trade_cash_out_info.pay_decision_id, merchant.mht_trade_cash_out_info.pay_success_date, merchant.mht_trade_cash_out_info.pay_success_time, merchant.mht_trade_cash_out_info.advance_type, merch
ime, merchant.mht_trade_cash_out_info.modify_time, merchant.mht_trade_request.card_idx_no, merchant.mht_trade_request.inst_id, merchant.mht_trade_cash_out_info.cash_type
                                     | 104.0 KB   | N/A     |
|         │ │ └IndexJoin_118         | 2977.06      | 0      | root     |                                                   | time:373.2µs, loops:1,
```

TiDB HACKATHON 2021

为什么做这个项目？

当前 Explain 只管疯狂输出…
完全不顾使用者的感受 > <

随着 TiDB 深入到更复杂的场景，业务 SQL 也愈发复杂 T.T

HTAP 引入了更复杂 AP 查询，让原本并不富裕的生活雪上加霜…

当所有这些因素结合到一起时 @@

PingCAP TiDB

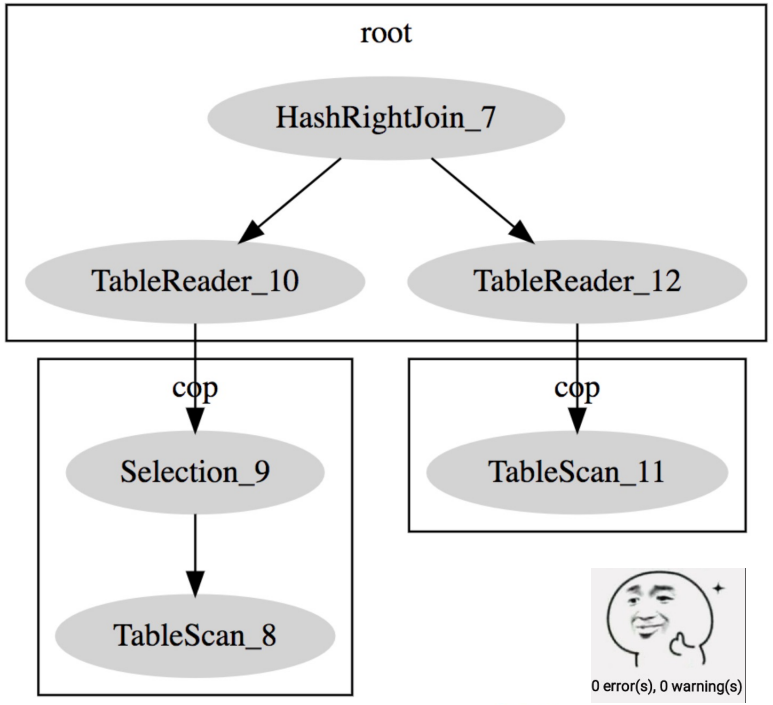
项目解决了什么问题？

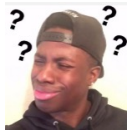
TiDB HACKATHON 2021

项目解决了什么问题？

PingCAP TiDB

假如能有个图像化的显示...Ummm





项目解决了什么问题？

PingCAP TiDB

用户

用户

用户

内核

TiDB HACKATHON 2021

项目解决了什么问题？

100% 解决 DBA 头疼难题

降低使用门槛，易用性++

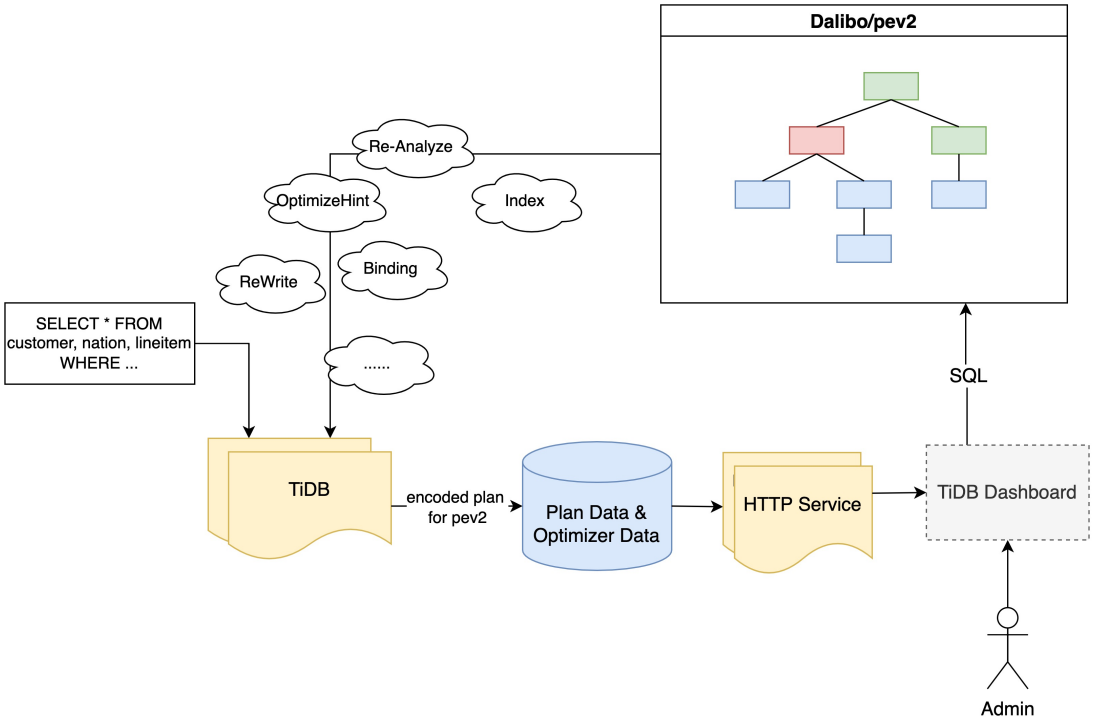
复杂问题排查效率提升 >666%

PingCAP TiDB

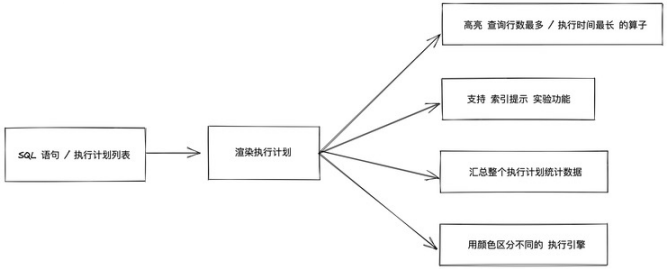
Hackathon 上实现了哪些内容？

TiDB HACKATHON 2021

Hackathon 上实现了哪些内容？



前端技术实现：



TiDB HACKATHON 2021

Hackathon 上实现了哪些内容？

PingCAP TiDB

1. 将可读性差的执行计划做可视化的**呈现和交互**
2. 对庞杂数据库侧疯狂输出信息做**分类梳理**
3. 对问题**关键高亮**提示，**辅助问题定位**，加速处理效率

TiDB HACKATHON 2021

Hackathon 上实现了哪些内容？

PingCAP TiDB

1. RFC： https://github.com/Hackathon-2022-TiVP/TiVP/blob/main/README.md
2. 项目地址： https://github.com/Hackathon-2022-TiVP

TiDB HACKATHON 2021

PingCAP TiDB

测试结果

TiDB HACKATHON 2021

PingCAP TiDB

未来展望

TiDB HACKATHON 2021

未来展望

PingCAP TiDB

1. 功能
    - 更准确的 Insights：Cost、Time、Stats、...
    - 初级调优：一键更新统计信息、Plan Diff
    - Optimizer Trace 诊断可视化：打开黑盒，看见内核逻辑

2. 集成
    - 与 Dashboard 集成：开箱即用
    - 与 Clinic 集成：执行计划 JSON 的导入与导出

Explore The Sky

PingCAP TiDB

1. What-If 启发式调优
智能 – Index Advisor, Hint Advisor, Partition Advisor, Data Temperature Advisor…

2. Not Only for TiDB
扩展 – 定义通用接口协议，支持 TiDB 外其他数据库如 MySQL, PostgreSQL 等，支持多种执行计划的展示、对比和优化

PingCAP TiDB

Thanks

@ Dongyu，Baoling, Ruoxi, Wangfan
@ Eason, Darui, Shenghui, Yuanyuan
@ 初赛评委，双呆

TiDB HACKATHON 2021