# 背景

MySQL本身没有常规意义上的执行计划，一般情况就是通过JOIN和QEP\_TAB这两个结构组成。QEP\_TAB 的全称是Query Execution Plan Table，这个“Table“可以是物理表、内存表、常量表、子查询的结果表等等。作为整个单独JOIN执行计划载体之前还承担着整个执行路径的调用和流转，但是从8.0.20后，全面的生成了独立的Iterator执行器引擎模式。在8.0.22中，又引入了AccessPath概念，真正的生成了独立的执行计划，从而进一步做到了优化过程到树型执行计划，最后到Iterator载体在执行引擎中的执行。

MySQL原始的Join都是依赖于QEP\_TAB列表，因为原来MySQL并不支持其他形态的Join结构，只支持左深树，那很容易直接使用数组来表示就可以了。优化器在生成执行计划只需要在QEP\_TAB上增加JOIN的属性op\_type，就可以递归去使用不同的Join方法和表访问方式了。

// Operation between the previous QEP\_TAB and this one.

enum enum\_op\_type {

// Regular nested loop.

OT\_NONE,

// Aggregate (GROUP BY).

OT\_AGGREGATE,

// Various temporary table operations, used at the end of the join.

OT\_MATERIALIZE,

OT\_AGGREGATE\_THEN\_MATERIALIZE,

OT\_AGGREGATE\_INTO\_TMP\_TABLE,

OT\_WINDOWING\_FUNCTION,

// Block-nested loop (rewritten to hash join).

OT\_BNL,

// Batch key access.

OT\_BKA

} op\_type = OT\_NONE;

# 概述

参考：

Access Path Selection in a Relational Database Management System：

<https://zhuanlan.zhihu.com/p/364303913?utm_id=0>

MySQL8.0优化器：<https://zhuanlan.zhihu.com/p/623046544>

MySQL中的HyperGraph优化器：

<http://mysql.taobao.org/monthly/2022/06/04/>

MySQL 8.0.23 Hypergraph Join Optimizer代码详解：

<http://mysql.taobao.org/monthly/2021/02/03/>

# 原理

# 源码