

深入浅出 OceanBase | 易用性专题

《SQL 性能优化实战》

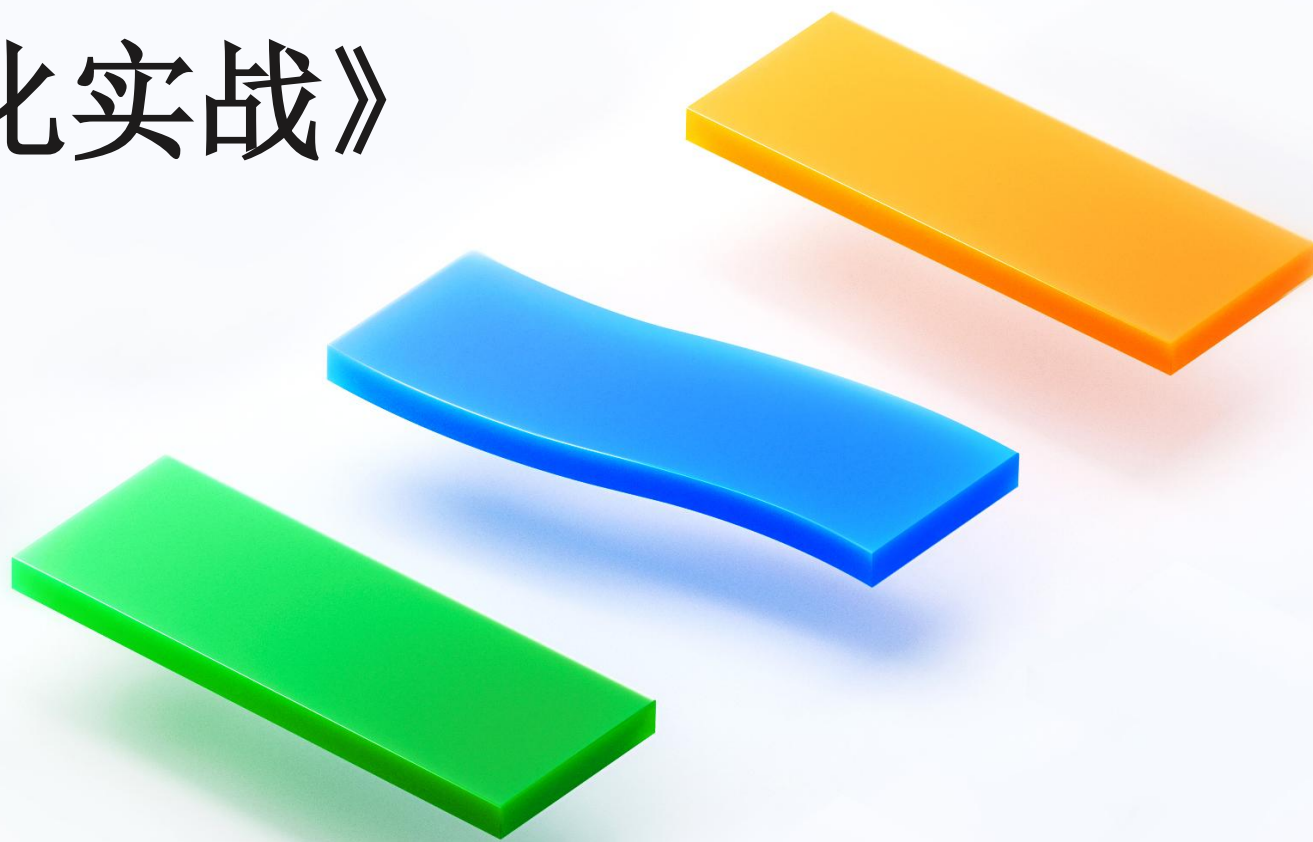
何文超

上海爱可生：DBA

OceanBase博客：何文超

CSDN：雅俗数据库

2025/08/14

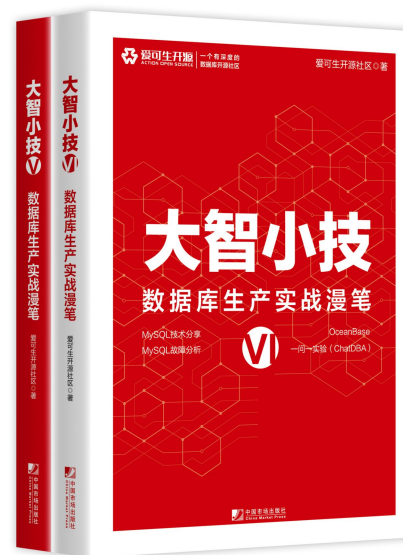


你将收获



- ① OceanBase 慢查询需要采集那些信息?
- ② 如何精准区分 trace_id、sql_id、plan_id?
- ③ 同类型 SQL 缘何冒出多个 sql_id?
- ④ 3.X vs 4.X: OceanBase 手动收集统计信息

社区介绍



技术交流：钉钉 [33254054](#)

如何诊断慢 SQL ？

① OceanBase 慢查询需要采集那些信息？ | 全局审计表

获取 SQL 执行的基础信息

3. x `GV$SQL_AUDIT`

4. x `GV$OB_SQL_AUDIT`

① OceanBase 慢查询需要采集那些信息？ | 定位性能瓶颈

条件	问题定位
ELAPSED_TIME 远大于 EXECUTE_TIME NET_WAIT_TIME/QUEUE_TIME 高	非数据库阶段耗时多，可能在应用 / 网络层
EXECUTE_TIME 占比高，CPU_TIME/IO_TIME 高	数据库侧瓶颈，需优化计划或索引

① OceanBase 慢查询需要采集那些信息？ | 执行计划类型

计划类型	是否高效	主要优化方向
本地执行计划	☒ 高效	保持局部性，避免退化
远程执行计划	☐ 中等	改进路由、使用 ODP 亲和性
分布式执行计划	☐ 较低	合理设计表结构、表组、并行执行

① OceanBase 慢查询需要采集那些信息？ | 获取信息

查看最优执行计划

```
explain extended
```

查看实际走的执行计划

```
gv$plan_cache_plan_stat
```

```
gv$plan_cache_plan_explain
```


① OceanBase 慢查询需要采集那些信息？ | 视图对比

对比维度	gv\$plan_cache_plan_stat	gv\$plan_cache_plan_explain
功能描述	记录计划缓存中所有 plan 的具体信息及每个计划总的执行统计信息， 每个 plan 在该视图中一条记录	展示某条 SQL 在计划缓存中的物理执行计划的详细信息。
数据内容	包含计划缓存中每个执行计划的统计信息，例如：参数化信息、执行次数、内存使用情况、命中率等。	包含某条 SQL 的物理执行计划的详细信息，例如：算子名称、表名、预估行数、代价、算子属性等。
查询条件	可直接根据 SQL ID 或其他字段筛选。	必须指定以下字段值：IP、PORT、TENANT_ID、PLAN_ID

① OceanBase 慢查询需要采集那些信息？ | 获取信息

查看执行画像

```
gv$sql_plan_monitor
```

作用：展示数据库执行 SQL 时的详细过程，包括操作符的执行顺序、数据的处理流程等；

有助于定位慢 SQL 主要的性能瓶颈点

获取表结构、数据量等基础信息

关联字段不是主键且未加索引

关联字段隐式转化

字段区分度

大表 or 小表

小表一定适合做驱动表吗？

① OceanBase 慢查询需要采集那些信息？ | 小表不适合作为驱动表

- 无索引或索引不佳
- 数据分布不均匀
- 连接条件复杂

大表一定不适合做驱动表吗？

① OceanBase 慢查询需要采集那些信息？ | 大表作为驱动表

- 存在强过滤条件
- 使用哈希连接且数据分布均匀
- 大表的索引设计合理
- 基于成本的优化器选择

hash join使用场景？

① OceanBase 慢查询需要采集那些信息？ | hash join

- 大表
- 存在有效的过滤条件，过滤后数据量很小

② 如何精准区分 trace_id、sql_id、plan_id?

ID 类型	定义与核心特性	关键说明
trace_id	OceanBase 数据库内部的 SQL 级别 ID	故障排查：用于标识一条 SQL 的执行过程。 一个 trace_id 可能对应多条 SQL 语句。
sql_id	用于唯一确定一条具体 SQL 的标识符	一个 sql_id 只对应一条 SQL。 常用 sql_id 绑定执行计划。
plan_id	标识 SQL 执行计划的编号	由 plan_cache 模块进行管理。 plan_id 需与 tenant_id 共同决定一个执行计划。

同类 SQL, 多个 sql_id?

③ 同类型 SQL 缘何冒出多个 sql_id? | 快速参数化

快速参数化

```
SELECT * FROM t1 WHERE c1 = 5 AND c2 = 'oceanbase'
```

参数化后变为:

```
SELECT * FROM t1 WHERE c1 = ? AND c2 = ?
```

③ 同类型 SQL 缘何冒出多个 sql_id? | 不适用场景

- in 的元素个数不定
- 大小写和空格
- 所使用的 ORDER BY / GROUP BY 后面常量不同

③ 同类型 SQL 缘何冒出多个 sql_id? | 不适用场景

- 定义不同的字符串格式
- 函数输入参数不同，并且函数计算结果不同
- 函数输入参数中，带有隐含信息并最终影响执行计划的常量

④ 3. X vs 4. x: OceanBase 手动收集统计信息

4.1 不同版本在命令支持上的差异

版本	MySQL 模式	Oracle 模式
3. X	仅支持 ANALYZE 命令方式	支持 DBMS_STATS 包和 ANALYZE 命令
4. X	支持 DBMS_STATS 包和 ANALYZE 命令	支持 DBMS_STATS 包和 ANALYZE 命令

4.2 Oracle 模式

推荐使用 DBMS_STATS 系统包进行手动收集统计信息。

4.3 MySQL 模式

3. x 版本 推荐使用 ANALYZE 命令进行手动收集统计信息。

④ 3. X vs. 4. x: OceanBase 手动收集统计信息

改进点	V3.x 及之前	V4.0 及之后
统计信息收集时机	每日合并期间	解耦，独立收集
统计信息准确性	增量合并导致不准确	全量扫描，更准确
是否支持直方图	不支持	支持
执行计划稳定性	易受合并影响	更稳定
管理灵活性	依赖合并	支持手动 + 自动 + 窗口配置

Thank you!

