

# 一刻钟 MySQL优化

叶金荣@知数堂

2017-12-16

# 只有一刻钟

## 哪些优化原则最要紧？

# 几条核心原则

- 选择对的存储引擎
- 表结构设计不留坑
- 及早发现性能瓶颈

# 存储引擎

- 首选InnoDB
- TokuDB、MyRocks
  - 海量日志、采集数据
  - 需要压缩的压缩数据
- ColumnStore
  - 基于MariaDB的OLAP应用

# 表结构设计

- InnoDB表一定要有主键
- 主键最好是INT/BIGINT，并呈单调递增特性
- 要注意字符集/校验集一致性，避免造成类型隐式转换
- 将TEXT/BLOB等大对象列独立存放新表中
- 多用INT，少用CHAR

# 表结构设计

- show table status

```

yejr@imysql.com [test]>show table status like 'yejr'\G
***** 1. row *****
      Name: yejr
      Engine: InnoDB
      Version: 10
      Row_format: Dynamic
      Rows: 797626
      Avg_row_length: 70
      Data_length: 56164352
Max_data_length: 0
      Index_length: 13123584
      Data_free: 279969792
      Auto_increment: 893210
      Create_time: 2017-12-02 20:57:36
      Update_time: NULL
      Check_time: NULL
      Collation: utf8_general_ci
      Checksum: NULL
      Create_options:
      Comment:
  
```

- 查看表宽度
- 计算碎片率

# 发现性能瓶颈

- `select * from information_schema.innodb_trx`

```
mysql> select * from information_schema.innodb_trx\G
***** 1. row *****
      trx_id: 150458138
      trx_state: RUNNING
      trx_started: 2017-12-10 16:17:18
      trx_weight: 1038029
      trx_mysql_thread_id: 48620
...
      trx_tables_in_use: 0
      trx_tables_locked: 1
      trx_lock_structs: 14029
      trx_lock_memory_bytes: 1597648
      trx_rows_locked: 1038028
      trx_rows_modified: 1024000
...

---TRANSACTION 150458138, ACTIVE 329 sec
14029 lock struct(s), heap size 1597648, 1038028 row lock(s), undo log entries 1024000
MySQL thread id 48620, OS thread handle 139902140466944, query id 3967568190 localhost root starting
```

- 找到锁定/修改最多rows的事务
- 找到活跃时间最久的的事务

# 发现性能瓶颈

- slow query log

```
# User@Host: root[root] @ localhost [] Id: 504
# Schema: test Last_errno: 0 Killed: 0
# Query_time: 0.018041 Lock_time: 0.000189 Rows_sent: 1 Rows_examined: 102400 Rows_affected: 0
# Bytes_sent: 68 Tmp_tables: 0 Tmp_disk_tables: 0 Tmp_table_sizes: 0
# QC_Hit: No Full_scan: Yes Full_join: No Tmp_table: No Tmp_table_on_disk: No
# Filesort: No Filesort_on_disk: No Merge_passes: 0
# InnoDB_IO_r_ops: 0 InnoDB_IO_r_bytes: 0 InnoDB_IO_r_wait: 0.000000
# InnoDB_rec_lock_wait: 0.000000 InnoDB_queue_wait: 0.000000
# InnoDB_pages_distinct: 175
SET timestamp=1512654670;
select count(*) from t1;
```

- 关注加粗指标
- 尤其是 InnoDB\_pages\_distinct



**最重要秘籍**  
**多看我的公众号**

# 提问的艺术

- 1、有问题时，将相关信息整理清楚、完善，发群，at我，涉密问题才单独发给我，一次没人回答多发重几次
- 2、若是咨询SQL效率问题，提供
  - 1、表DDL以及索引统计信息，用 `SHOW CREATE TABLE / SHOW INDEX FROM TABLE` 查看；
  - 2、SQL执行计划，用 `EXPLAIN` 解析（含完整SQL）；
  - 3、最好还能提供SQL执行时的PROFILE / STATUS结果；
- 3、若是咨询性能上的问题，提供
  - 1、（负载较高时）运行top后的截屏；
  - 2、（负载较高时）运行`vmstat -S m 1 50`后的截屏；
  - 3、（负载较高时）`PROCESSLIST`根据耗时排序后的截屏；
  - 4、MySQL `global variables` 清单；
  - 5、MySQL `global status` 清单；
  - 6、服务器硬件配置信息。
- 4、有报错时，请提供详细错误信息
- 5、招聘需求，直接联系助教