阿里云

2019阿里云峰会·上海
开发者大会
DEVELOPER CONFERENCE

冷香

阿里云智能高级技术专家

阿里云 2019阿里云峰会·上海 开发者大会

深度解析 AliSQL 8.0 特性和改进

赵建伟

高级技术专家
云智能-OLTP产品部-AliSQL 内核小组

Agenda

- Performance Insight & Diagnose
- New Feature
- Stability Improvement

Agenda

- Performance Insight & Diagnose
- New Feature
- Stability Improvement

Performance Insight

研发视角

Performance Schema

DBA和开发视角

- Transaction
  - Binlog Size
  - Lock Count
  - Elapsed Time
- Object
  - Read Write
  - Index Usage
- Statement
  - CPU / Elapsed Time
  - MDL / Trans Lock
  - IO Stats
  - Concurrency

Performance Insight

– Object Statistics

- Table statistics 是业务系统 scale 的数据支撑

```
mysql> select * from TABLE_STATISTICS where table_schema='test';
+--------------+------------+-----------+--------------+-----------------------+---------------+--------------+--------------+
| TABLE_SCHEMA | TABLE_NAME | ROWS_READ | ROWS_CHANGED | ROWS_CHANGED_X_INDEXES | ROWS_INSERTED | ROWS_DELETED | ROWS_UPDATED |
+--------------+------------+-----------+--------------+-----------------------+---------------+--------------+--------------+
| test         | sbtest2    |         0 |        12288 |                 36864 |         12288 |            0 |            0 |
| test         | sbtest1    |     12288 |            0 |                     0 |             0 |            0 |            0 |
+--------------+------------+-----------+--------------+-----------------------+---------------+--------------+--------------+
2 rows in set (0.01 sec)
```

- Index statistics 是业务系统优化 index 的数据支撑

```
mysql> select * from index_STATISTICS where table_schema='test' order by table_name;
+--------------+------------+------------+-----------+
| TABLE_SCHEMA | TABLE_NAME | INDEX_NAME | ROWS_READ |
+--------------+------------+------------+-----------+
| test         | sbtest1    | PRIMARY    |     12288 |
| test         | sbtest2    | ind_1      |     20480 |
| test         | sbtest2    | PRIMARY    |         1 |
+--------------+------------+------------+-----------+
3 rows in set (0.00 sec)
```

Performance Insight

– Statement Statistics

CPU	LOCK	Concurrency	IO
• ELAPSED_TIME • CPU_TIME	• SERVER_LOCK_TIME • TRANSACTION_LOCK_TIME	• MUTEX_SPINS • MUTEX_WAITS • RWLOCK_SPIN_WAITS • RWLOCK_SPIN_ROUNDS	• DATA_READS • DATA_WRITES • LOGICAL_READS • PHYSICAL_READS • PHYSICAL_ASYNC_READS

Performance Insight

– Statement Statistics

CPU Intensive

```
*************************** 3. row ***************************
           COUNT_STAR: 1
          SCHEMA_NAME: test
               DIGEST: 0c9159bc711f2221271e8bff7781a
          DIGEST_TEXT: SELECT `md5` ( REPEAT (...) )
         ELAPSED_TIME: 2268471
             CPU_TIME: 2269723
    SERVER_LOCK_TIME: 0
TRANSACTION_LOCK_TIME: 0
         MUTEX_SPINS: 0
         MUTEX_WAITS: 0
    RWLOCK_SPIN_WAITS: 0
   RWLOCK_SPIN_ROUNDS: 0
      RWLOCK_OS_WAITS: 0
         DATA_READS: 0
    DATA_READ_TIME: 0
        DATA_WRITES: 0
    DATA_WRITE_TIME: 0
        REDO_WRITES: 0
    REDO_WRITE_TIME: 0
       LOGICAL_READS: 0
      PHYSICAL_READS: 0
 PHYSICAL_ASYNC_READS: 0
3 rows in set (0.01 sec)
```

MDL Block

```
*************************** 16. row ***************************
           COUNT_STAR: 1
          SCHEMA_NAME: test
               DIGEST: e685ac44a5a587ad64ef4041496df87df0
          DIGEST_TEXT: ALTER TABLE `t` ADD `col1` INTEGER
         ELAPSED_TIME: 7832686
             CPU_TIME: 65084
    SERVER_LOCK_TIME: 7827465
TRANSACTION_LOCK_TIME: 0
         MUTEX_SPINS: 0
         MUTEX_WAITS: 0
    RWLOCK_SPIN_WAITS: 0
   RWLOCK_SPIN_ROUNDS: 0
      RWLOCK_OS_WAITS: 0
         DATA_READS: 5
    DATA_READ_TIME: 152
        DATA_WRITES: 0
    DATA_WRITE_TIME: 0
        REDO_WRITES: 0
    REDO_WRITE_TIME: 0
       LOGICAL_READS: 380
      PHYSICAL_READS: 5
 PHYSICAL_ASYNC_READS: 0
```

Trans Block

```
*************************** 20. row ***************************
           COUNT_STAR: 2
          SCHEMA_NAME: test
               DIGEST: 6aafc183c822b96a2dc4ea149673e156f985356a
          DIGEST_TEXT: UPDATE `t` SET `col1` = ? WHERE `id` = ?
         ELAPSED_TIME: 16917306
             CPU_TIME: 4602
    SERVER_LOCK_TIME: 2082
TRANSACTION_LOCK_TIME: 16912736
         MUTEX_SPINS: 0
         MUTEX_WAITS: 0
    RWLOCK_SPIN_WAITS: 0
   RWLOCK_SPIN_ROUNDS: 0
      RWLOCK_OS_WAITS: 0
         DATA_READS: 0
    DATA_READ_TIME: 0
        DATA_WRITES: 0
    DATA_WRITE_TIME: 0
        REDO_WRITES: 0
    REDO_WRITE_TIME: 0
       LOGICAL_READS: 13
      PHYSICAL_READS: 0
 PHYSICAL_ASYNC_READS: 0
```

Performance Insight

– Statement Statistics

IO Intensive

```
*************************** 3. row ***************************
          COUNT_STAR: 1
         SCHEMA_NAME: test
              DIGEST: 936a61dc5894c1f57310a39b809bed324af3879f9663bafdb
         DIGEST_TEXT: INSERT INTO `t` ( `col1` ) SELECT `col1` FROM `t`
        ELAPSED_TIME: 4355092
            CPU_TIME: 4343601
     SERVER_LOCK_TIME: 593
TRANSACTION_LOCK_TIME: 0
         MUTEX_SPINS: 0
         MUTEX_WAITS: 0
    RWLOCK_SPIN_WAITS: 0
   RWLOCK_SPIN_ROUNDS: 0
      RWLOCK_OS_WAITS: 0
           DATA_READS: 0
       DATA_READ_TIME: 0
          DATA_WRITES: 15
     DATA_WRITE_TIME: 18541
          REDO_WRITES: 0
     REDO_WRITE_TIME: 0
       LOGICAL_READS: 51071
      PHYSICAL_READS: 0
 PHYSICAL_ASYNC_READS: 0
```

Concurrency

```
*************************** 3. row ***************************
          COUNT_STAR: 5672
         SCHEMA_NAME: test
              DIGEST: 02067fc1076d23a0fbb1d0652e79d7445ba4eff1fef5e0f991
         DIGEST_TEXT: UPDATE `sbtest1` SET `k` = `k` + ? WHERE `id` = ?
        ELAPSED_TIME: 36154956346
            CPU_TIME: 36571321
     SERVER_LOCK_TIME: 55829501
TRANSACTION_LOCK_TIME: 35288782973
         MUTEX_SPINS: 0
         MUTEX_WAITS: 0
    RWLOCK_SPIN_WAITS: 19381
   RWLOCK_SPIN_ROUNDS: 863751
      RWLOCK_OS_WAITS: 52013
           DATA_READS: 0
       DATA_READ_TIME: 0
          DATA_WRITES: 4
     DATA_WRITE_TIME: 6962
          REDO_WRITES: 0
     REDO_WRITE_TIME: 0
       LOGICAL_READS: 84043
      PHYSICAL_READS: 0
 PHYSICAL_ASYNC_READS: 0
3 rows in set (0.00 sec)
```

Performance Diagnose

云产品深度依赖，可诊断性至关重要

InnoDB IO 表现

Slot: 10000
Interval: 2S
Durable: 5H

```
mysql> select * from IO_STATISTICS;
+---------------------+-----------+---------------+-------------------+---------------+------------+----------------+--------------------+-----------------+
| TIME                | DATA_READ | DATA_READ_TIME | DATA_READ_MAX_TIME | DATA_READ_BYTES | DATA_WRITE | DATA_WRITE_TIME | DATA_WRITE_MAX_TIME | DATA_WRITE_BYTES |
+---------------------+-----------+---------------+-------------------+---------------+------------+----------------+--------------------+-----------------+
| 2019-03-19 17:31:45 |         0 |             0 |                 0 |             0 |          0 |              0 |                  0 |               0 |
| 2019-03-19 17:31:49 |        69 |          1455 |                53 |       1130496 |         14 |           1122 |                258 |          589824 |
| 2019-03-19 17:31:51 |         0 |             0 |                 0 |             0 |         22 |           2242 |                246 |         1064960 |
| 2019-03-19 17:31:55 |         0 |             0 |                 0 |             0 |         36 |           3937 |                256 |         6176768 |
| 2019-03-19 17:31:57 |         0 |             0 |                 0 |             0 |         52 |           4451 |                285 |          851968 |
| 2019-03-19 17:31:59 |         0 |             0 |                 0 |             0 |         54 |           5852 |               1552 |          884736 |
| 2019-03-19 17:32:01 |         0 |             0 |                 0 |             0 |         66 |           5433 |                195 |         1245184 |
| 2019-03-19 17:32:03 |         0 |             0 |                 0 |             0 |         53 |           4348 |                321 |          868352 |
| 2019-03-19 17:32:05 |         0 |             0 |                 0 |             0 |         71 |           6751 |               1406 |         1163264 |
| 2019-03-19 17:32:07 |         0 |             0 |                 0 |             0 |        114 |         125630 |               3376 |        87588864 |
| 2019-03-19 17:32:11 |         0 |             0 |                 0 |             0 |         14 |           2409 |                677 |         4325376 |
| 2019-03-19 17:32:15 |         0 |             0 |                 0 |             0 |         13 |           2082 |                528 |         4325376 |
| 2019-03-19 17:32:21 |         0 |             0 |                 0 |             0 |         11 |            836 |                287 |         1835008 |
| 2019-03-19 17:32:25 |        37 |           841 |                32 |        606208 |         16 |           1767 |                597 |         3555328 |
| 2019-03-19 17:32:27 |         0 |             0 |                 0 |             0 |         22 |           1376 |                242 |         2441216 |
| 2019-03-19 17:32:29 |         0 |             0 |                 0 |             0 |         18 |           8109 |               1633 |         6307840 |
| 2019-03-19 17:32:31 |         0 |             0 |                 0 |             0 |         16 |           7842 |               1621 |         6291456 |
| 2019-03-19 17:32:33 |         0 |             0 |                 0 |             0 |         12 |            630 |                141 |         1310720 |
| 2019-03-19 17:32:35 |         0 |             0 |                 0 |             0 |         10 |            688 |                251 |         1376256 |
| 2019-03-19 17:32:37 |         0 |             0 |                 0 |             0 |         10 |            568 |                157 |          999424 |
| 2019-03-19 17:32:39 |         0 |             0 |                 0 |             0 |         21 |           1107 |                179 |         2310144 |
| 2019-03-19 17:32:43 |         0 |             0 |                 0 |             0 |         22 |           1240 |                213 |         2326528 |
| 2019-03-19 17:32:45 |         0 |             0 |                 0 |             0 |         11 |            524 |                168 |         1048576 |
| 2019-03-19 17:32:47 |         0 |             0 |                 0 |             0 |         17 |            694 |                211 |         1146880 |
| 2019-03-19 17:32:49 |         0 |             0 |                 0 |             0 |         18 |            632 |                179 |         1196032 |
| 2019-03-19 17:32:51 |         0 |             0 |                 0 |             0 |         15 |            686 |                199 |         1212416 |
| 2019-03-19 17:32:53 |         0 |             0 |                 0 |             0 |         29 |          13136 |               1719 |        10534912 |
| 2019-03-19 17:32:55 |         0 |             0 |                 0 |             0 |         19 |           3566 |               1449 |         3375104 |
| 2019-03-19 17:32:57 |         0 |             0 |                 0 |             0 |         14 |            249 |                 26 |          262144 |
| 2019-03-19 17:32:59 |         0 |             0 |                 0 |             0 |         17 |            824 |                162 |         1310720 |
```

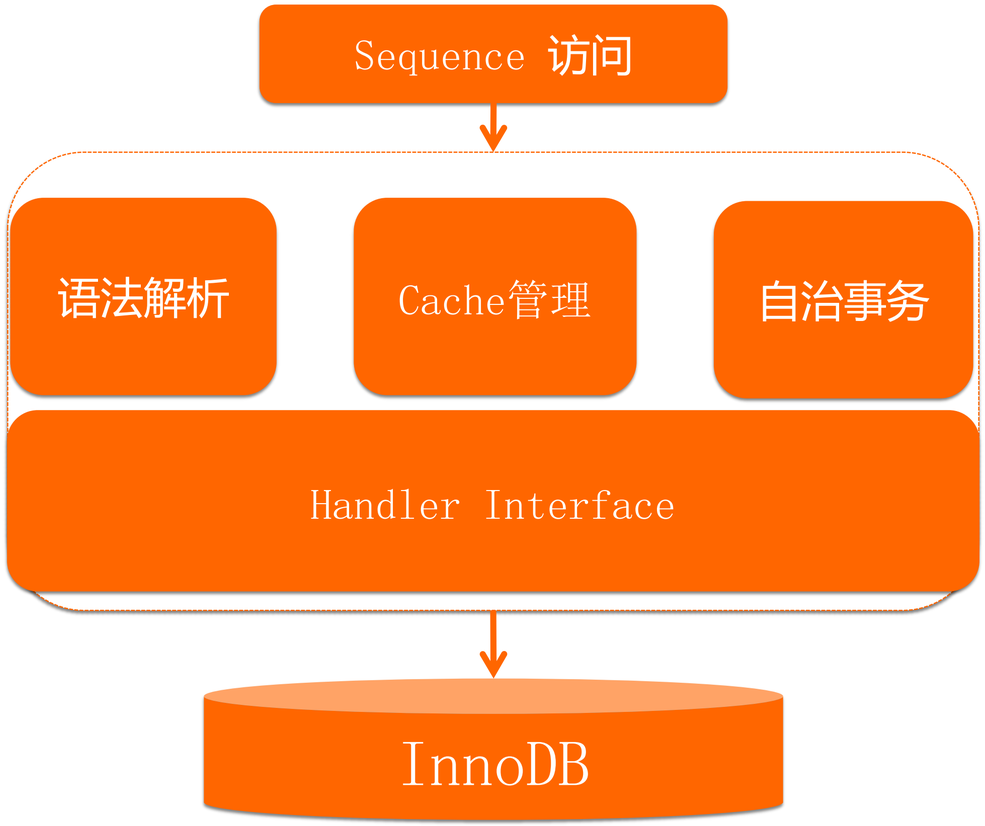
Agenda

- Performance Insight & Diagnose
- New Feature
- Stability Improvement

Sequence Engine

Sequence Syntax:

```
CREATE SEQUENCE [IF NOT EXISTS] schema.seq
          [START WITH <constant>]
          [MINVALUE <constant>]
          [MAXVALUE <constant>]
          [INCREMENT BY <constant>]
          [CACHE <constant> | NOCACHE]
          [CYCLE | NOCYCLE]
         ;
SELECT NEXTVAL(seq);
SELECT CURRVAL(seq);
```

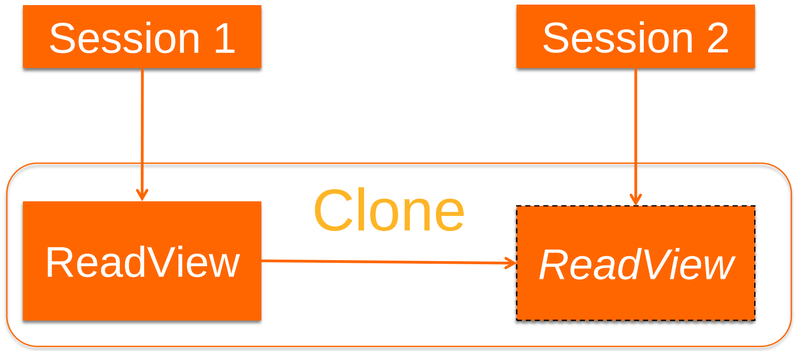


Customized ReadView

- 自定义 ReadView（Cross-session consistent）
- Syntax

```
EXPORT CONSISTENT SNAPSHOT LOCAL
RELEASE CONSISTENT SNAPSHOT '$snap_id'
START TRANSACTION WITH CONSISTENT SNAPSHOT '$snap_id'
```

- Proxy 将可以跨 session 做并行计算



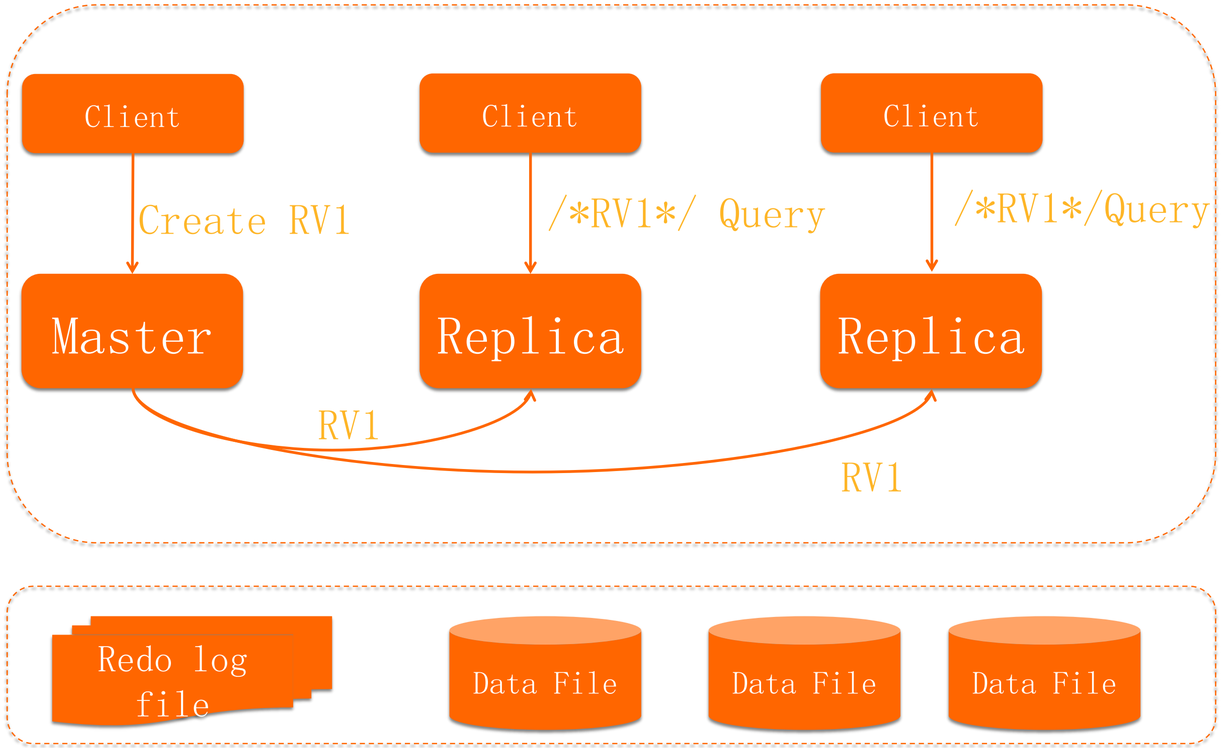
Global ReadView (PolarDB)

- 全局 ReadView（Cross-Node consistent）

- Syntax

```
EXPORT CONSISTENT SNAPSHOT CLUSTER
RELEASE CONSISTENT SNAPSHOT '$snap_id'
START TRANSACTION WITH CONSISTENT SNAPSHOT '$snap_id'
```

- Proxy 将可以跨节点做并行计算



Agenda

- Performance Insight & Diagnose
- New Feature
- Stability Improvement

Statement Concurrency control

1. CCL 规则设计

- SQL command

根据 statement 的类型，例如 ‘SELECT’, ’UPDATE’, ’INSERT’, ’DELETE’;

- Object

根据 statement 操作的对象进行控制， 例如 TABLE，VIEW;

- keywords

根据 statement 语句的关键字进行控制;

2. CCL 接口设计

DBMS_CCL.add_ccl_rule();

DBMS_CCL.del_ccl_rule();

DBMS_CCL.show_ccl_rule();

DBMS_CCL.flush_ccl_rule();

```
mysql> show processlist;
+-----+-----------------+-----------------+------+---------+------+-----------------------------+-----------------------------------+
| Id  | User            | Host            | db   | Command | Time | State                       | Info                              |
+-----+-----------------+-----------------+------+---------+------+-----------------------------+-----------------------------------+
|  72 | root            | localhost:33601 | NULL | Query   |    0 | starting                    | show processlist                  |
| 171 | u1              | localhost:60120 | test | Query   |    2 | Concurrency control waitting | SELECT pad FROM sbtest3 WHERE id=51 |
| 172 | u1              | localhost:60128 | test | Query   |    5 | Concurrency control waitting | SELECT pad FROM sbtest4 WHERE id=35 |
| 174 | u1              | localhost:60385 | test | Query   |    4 | Concurrency control waitting | SELECT pad FROM sbtest3 WHERE id=54 |
| 178 | u1              | localhost:60136 | test | Query   |   12 | Concurrency control waitting | SELECT pad FROM sbtest4 WHERE id=51 |
| 179 | u1              | localhost:60149 | test | Query   |    5 | Concurrency control waitting | SELECT pad FROM sbtest2 WHERE id=51 |
| 182 | u1              | localhost:60124 | test | Query   |    1 | Concurrency control waitting | SELECT pad FROM sbtest4 WHERE id=51 |
| 183 | u1              | localhost:60371 | test | Query   |    5 | User sleep                  | SELECT pad FROM sbtest2 WHERE id=51 |
| 184 | u1              | localhost:60133 | test | Query   |    4 | Concurrency control waitting | SELECT pad FROM sbtest3 WHERE id=51 |
| 190 | u1              | localhost:60406 | test | Query   |    5 | Concurrency control waitting | SELECT pad FROM sbtest3 WHERE id=51 |
| 191 | u1              | localhost:60402 | test | Query   |    1 | Concurrency control waitting | SELECT pad FROM sbtest4 WHERE id=51 |
| 192 | u1              | localhost:60131 | test | Query   |    2 | User sleep                  | SELECT pad FROM sbtest1 WHERE id=51 |

......
```

Statement Outline

1. Outline 规则设计

– Optimizer Hint

根据作用域（query block）和 hint 对象，

分为：Global level hint，Table/Index level hint，Join order hint等等

```
call dbms_outln.show_outline();
+------------+----------------------------------------------------------------------+-----------+-------+------+-------------------------------------
| SCHEMA     | DIGEST                                                               | TYPE      | SCOPE | POS  | HINT
+------------+----------------------------------------------------------------------+-----------+-------+------+-------------------------------------
| outline_db | 36bebc61fce7e32b93926aec3fdd790dad5d895107e2d8d3848d1c60b74bcde6 | OPTIMIZER |       |    1 | /*+ SET_VAR(foreign_key_checks=OFF) *
| outline_db | 36bebc61fce7e32b93926aec3fdd790dad5d895107e2d8d3848d1c60b74bcde6 | OPTIMIZER |       |    1 | /*+ MAX_EXECUTION_TIME(1000) */
| outline_db | d4dcef634a4a664518e5fb8a21c6ce9b79fccb44b773e86431eb67840975b649 | OPTIMIZER |       |    1 | /*+ BNL(t1,t2) */
| outline_db | 5a726a609b6fbfb76bb8f9d2a24af913a2b9d07f015f2ee1f6f2d12dfad72e6f | OPTIMIZER |       |    2 |  /*+ QB_NAME(subq1) */
| outline_db | 5a726a609b6fbfb76bb8f9d2a24af913a2b9d07f015f2ee1f6f2d12dfad72e6f | OPTIMIZER |       |    1 | /*+ SEMIJOIN(@subq1 MATERIALIZATION,
| outline_db | b4369611be7ab2d27c85897632576a04bc08f50b928a1d735b62d0a140628c4c | USE INDEX |       |    1 | ind_1
| outline_db | 33c71541754093f78a1f2108795cfb45f8b15ec5d6bff76884f4461fb7f33419 | USE INDEX |       |    2 | ind_2
+------------+----------------------------------------------------------------------+-----------+-------+------+-------------------------------------
in set (0.00 sec)
```

– DBMS_OUTLN.show_outline(); 展示内存中可用的所有 outline 及命中情况

– DBMS_OUTLN.del_outline(); 删除内存和持久化表中的 outline

– DBMS_OUTLN.flush_outline(); 刷新所有的 outline，从 mysql.outline 表中重新

load

Async Purge InnoDB Data File

- Big Table drop 的成本

- 单机文件系统 EXT4
  - Page Cache 回收
  - Meta 信息 flush
  - Journal 日志写入

```
-     12.74%     rm  [jbd2]                [k] jbd2_journal_invalidatepage
   - jbd2_journal_invalidatepage
      - 99.54% ext4_invalidatepage
           do_invalidatepage
           truncate_inode_page
           truncate_inode_pages_range
           truncate_inode_pages
           ext4_delete_inode
           generic_delete_inode
           generic_drop_inode
           iput
           do_unlinkat
           sys_unlinkat
           system_call
           unlinkat
           0x400000
+      8.08%     rm  [kernel.kallsyms]  [k] __mem_cgroup_uncharge_common
+      7.02%     rm  [kernel.kallsyms]  [k] free_pcppages_bulk
+      5.53%     rm  [kernel.kallsyms]  [k] list_del
+      3.91%     rm  [kernel.kallsyms]  [k] find_get_pages
+      3.84%     rm  [kernel.kallsyms]  [k] free_hot_cold_page
+      3.65%     rm  [kernel.kallsyms]  [k] truncate_inode_pages_range
+      3.33%     rm  [kernel.kallsyms]  [k] put_page
+      3.22%     rm  [kernel.kallsyms]  [k] drop_buffers
+      3.11%     rm  [kernel.kallsyms]  [k] release_pages
+      3.05%     rm  [kernel.kallsyms]  [k] kmem_cache_free
+      2.70%     rm  [kernel.kallsyms]  [k] cancel_dirty_page
+      2.64%     rm  [kernel.kallsyms]  [k] truncate_inode_page
+      2.36%     rm  [kernel.kallsyms]  [k] unlock_buffer
+      2.35%     rm  [kernel.kallsyms]  [k] radix_tree_delete
+      2.29%     rm  [kernel.kallsyms]  [k] _spin_lock
+      2.28%     rm  [kernel.kallsyms]  [k] bit_spin_lock
```

Async Purge InnoDB Data File

– DDL Atomic

数据库和文件系统一致性保证（日志补偿机制）

CREATE TABLE

1. 开启事务
2. 修改DD
3. 开启事务
4. 插入DDL log
5. 提交事务
6. 删除DDL log
7. 创建表空间和文件
8. 提交 DD 事务

如果DD 事务失败或者crash，
Replay DDL log 清理文件

DROP TABLE

1. 开启事务
2. 修改DD
3. 插入DDL log
4. 提交事务
5. 开启事务
6. Replay DDL log
   删除表空间和文件
7. 删除 DDL log
8. 提交事务

如果 DD 事务失败，Do nothing；
如果 crash，启动 DDL log recovery 清理文件

Async Purge InnoDB Data File

DROP TABLE

1. 开启事务
2. 修改DD
3. 插入DDL log
4. 提交事务
5. Replay DDL log
6. 开启事务
7. 插入 Purge DDL log
8. 提交事务
9. Rename Data File
10. 插入队列
11. 删除 DDL log
12. 提交事务

InnoDB File Purge Thread

1. 从队列中取一个文件
2. ftruncate 文件
3. 如果文件 < threshold：
   unlink文件；删除 Purge log
   如果文件 > threshold：
   回到步骤1；

如果 crash，DDL log recovery 保证atomic

Multi-Queue Thread Pool

Thread Scheduling

one-thread-per-connection

- CPU 时间片公平调度
- 线程切换开销线性增长
- 无业务识别能力

Priority Thread Pool

- 线程切换开销稳定
- 业务识别，事务优先
- 无 SQL 复杂度判断

Multi-Queue Thread Pool

- 线程切换开销稳定
- 针对不同的SQL，识别事务，复制查询，短平快 SQL 等建立多队列，提升稳定和吞吐

TP ： **在大规模连接和复杂混合** SQL 模型下，保持MySQL 持续稳定吞吐能力

Implicit Primary Key

- 增加一个 implicit column 和 key
- Slave SQL apply 索引选择优先 implicit key

```
mysql> show create table t\G
*************************** 1. row ***************************
       Table: t
Create Table: CREATE TABLE `t` (
  `id` int(11) DEFAULT NULL,
  `__#alibaba_rds_row_id#__` bigint(20) NOT NULL AUTO_INCREMENT COMMENT
  KEY `__#alibaba_rds_row_id#__` (`__#alibaba_rds_row_id#__`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
1 row in set (0.00 sec)
```

背景：

- 分区表 Constraint 需要带分区键，所以 PK->Key
- 大量的 NULL 导致 UK 并不是 SQL apply 的优先选择

Transaction management

Transaction Isolation comparison

MySQL 各种阻塞频发

MySQL		Oracle
READ UNCOMMITTED		
READ COMMITTED	VS	READ COMMITTED（Default）
REPEATABLE READ（Default）		
SERIALIZABLE		SERIALIZABLE

- Non-transaction First Query

  RR 级别以下的第一条 select 语句不启动事务

- Kill idle transaction

  事务空闲超时：Set kill_idle_transaction_timeout=xxx (seconds)

阿里云开发者社区

扫码加入社群
与志同道合的码友一起
Code Up

阿里云数据库微信公众号

谢谢！