MySQL线上常见故障剖析

俊达

各种故障

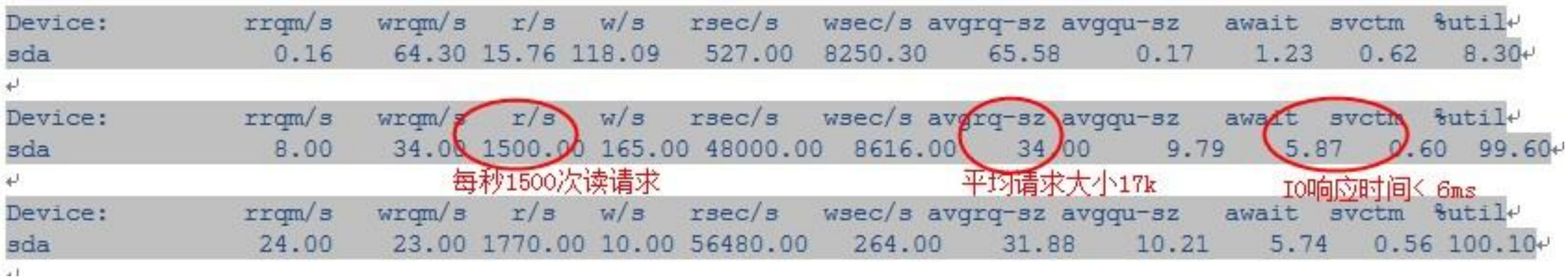
- 应用获取不到连接池
- 数据库响应慢
- SQL慢
- 服务器load高
- SWAP
- 表不见了
- MySQL crash
- 主机Hung
- …

观察你的系统

- **MySQL**
  - 活动进程(Process list)
  - 日志文件(slow log, alert log, general query log, binlog)
  - Status variables（com_select, com_insert,.etc）
  - InnoDB(物理读、逻辑读、innodb status)
  - 参数配置
  - Stack trace(plus source code)
- **SQL**
  - 执行计划，explain
- **OS**
  - 内存, SWAP, /proc/meminfo
  - CPU, load, ps
  - IO（磁盘、网络)
    - Iostat
- **Profile**
  - Oprofile
  - gprof

Case 1: XXX系统报连接池满

iostat



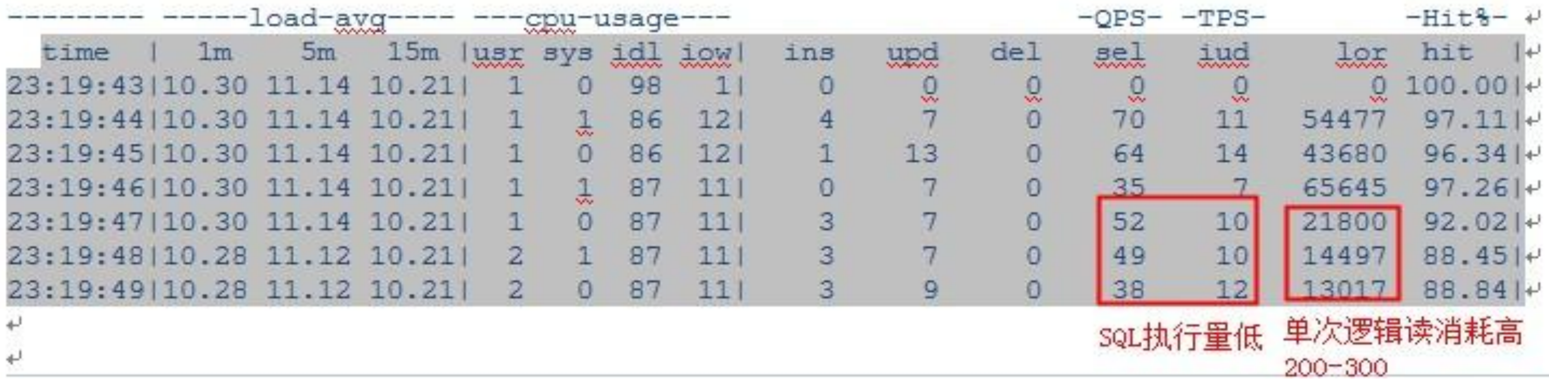
```
Device:         rrqm/s   wrqm/s   r/s   w/s   rsec/s   wsec/s avgrq-sz avgqu-sz   await  svctm  %util
sda               0.16    64.30 15.76 118.09   527.00  8250.30    65.58     0.17    1.23   0.62   8.30

Device:         rrqm/s   wrqm/s   r/s   w/s   rsec/s   wsec/s avgrq-sz avgqu-sz   await  svctm  %util
sda               8.00    34.00 1500.00 165.00 48000.00  8616.00    34.00     9.79    5.87   0.60  99.60

Device:         rrqm/s   wrqm/s   r/s   w/s   rsec/s   wsec/s avgrq-sz avgqu-sz   await  svctm  %util
sda              24.00    23.00 1770.00 10.00 56480.00   264.00    31.88    10.21    5.74   0.56 100.10
```

每秒1500次读请求

平均请求大小17k

IO响应时间< 6ms

orzdba



```
-------- -----load-avg---- ---cpu-usage---                                   -QPS- -TPS-         -Hit%-
  time   |  1m    5m   15m |usr sys idl iow|   ins    upd    del    sel    iud      lor    hit  |
23:19:43|10.30 11.14 10.21|  1   0  98   1|     0      0      0      0      0        0 100.00|
23:19:44|10.30 11.14 10.21|  1   1  86  12|     4      7      0     70     11    54477  97.11|
23:19:45|10.30 11.14 10.21|  1   0  86  12|     1     13      0     64     14    43680  96.34|
23:19:46|10.30 11.14 10.21|  1   1  87  11|     0      7      0     35      7    65645  97.26|
23:19:47|10.30 11.14 10.21|  1   0  87  11|     3      7      0     52     10    21800  92.02|
23:19:48|10.28 11.12 10.21|  2   1  87  11|     3      7      0     49     10    14497  88.45|
23:19:49|10.28 11.12 10.21|  2   0  87  11|     3      9      0     38     12    13017  88.84|
```

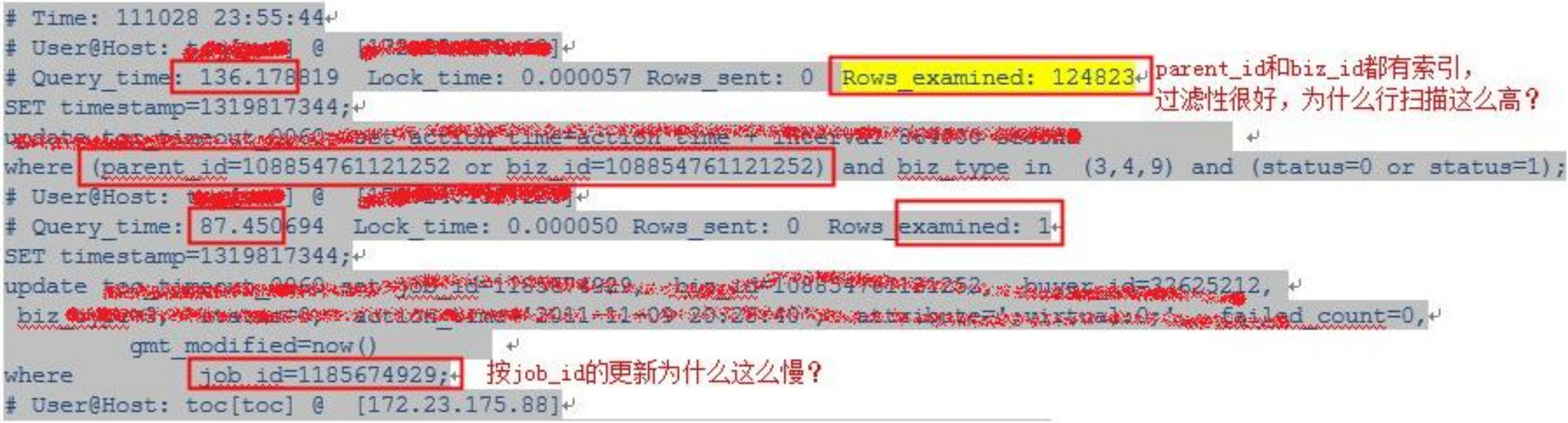
SQL执行量低

单次逻辑读消耗高

200-300

slowlog

- **What’s in slow log?**



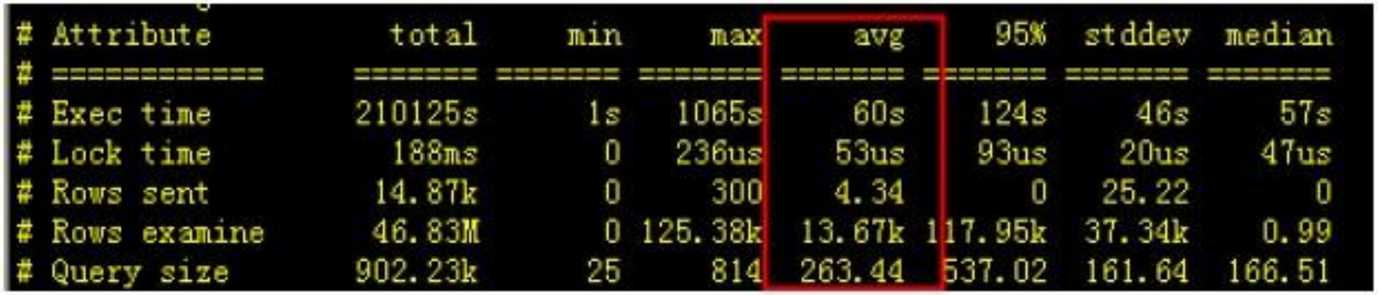
```
# Time: 111028 23:55:44
# User@Host: [redacted] @  [redacted]
# Query_time: 136.178819  Lock_time: 0.000057 Rows_sent: 0  Rows_examined: 124823
SET timestamp=1319817344;
[redacted]
where (parent_id=108854761121252 or biz_id=108854761121252) and biz_type in  (3,4,9) and (status=0 or status=1);
# User@Host: [redacted] @  [redacted]
# Query_time: 87.450694  Lock_time: 0.000050 Rows_sent: 0  Rows_examined: 1
SET timestamp=1319817344;
update [redacted]625212,
 biz [redacted]_count=0,
        gmt_modified=now()
where          job_id=1185674929;
# User@Host: toc[toc] @  [172.23.175.88]
```

parent_id和biz_id都有索引，
过滤性很好，为什么行扫描这么高？

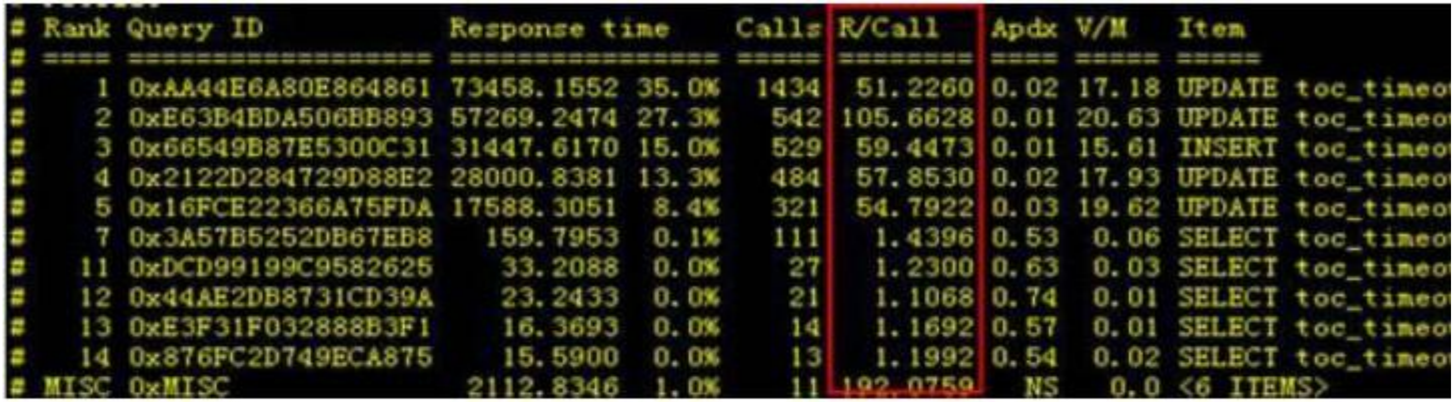
按job_id的更新为什么这么慢？

Mk-query-digest

- **mk-query-digest 全面分析slow log**



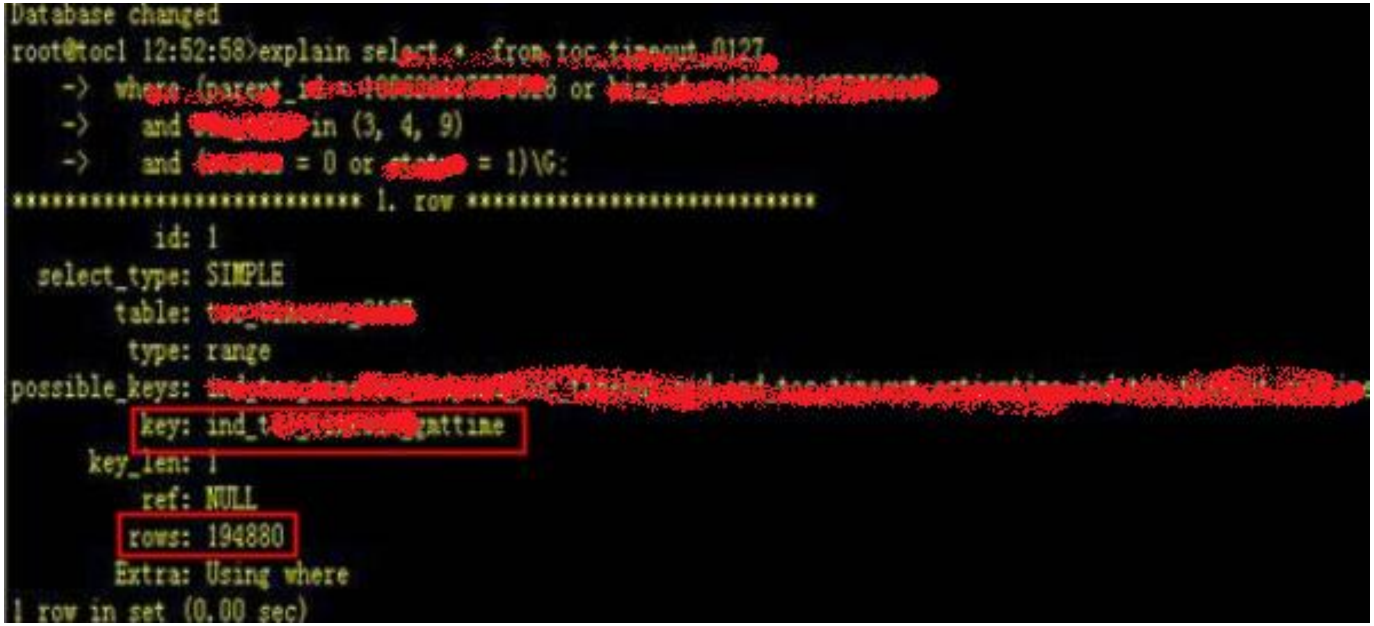
```
# Attribute          total     min     max     avg     95%  stddev  median
# ============     ======= ======= ======= ======= ======= ======= =======
# Exec time        210125s      1s   1065s     60s    124s     46s     57s
# Lock time          188ms       0   236us    53us    93us    20us    47us
# Rows sent         14.87k       0     300    4.34       0   25.22       0
# Rows examine      46.83M       0 125.38k  13.67k 117.95k  37.34k    0.99
# Query size       902.23k      25     814  263.44  537.02  161.64  166.51
```



```
# Rank Query ID           Response time      Calls R/Call   Apdx V/M   Item
# ==== ================== ================== ===== ======== ==== ===== =====
#    1 0xAA44E6A80E864861 73458.1552 35.0%  1434  51.2260 0.02 17.18 UPDATE toc_timeo
#    2 0xE63B4BDA506BB893 57269.2474 27.3%   542 105.6628 0.01 20.63 UPDATE toc_timeo
#    3 0x66549B87E5300C31 31447.6170 15.0%   529  59.4473 0.01 15.61 INSERT toc_timeo
#    4 0x2122D284729D88E2 28000.8381 13.3%   484  57.8530 0.02 17.93 UPDATE toc_timeo
#    5 0x16FCE22366A75FDA 17588.3051  8.4%   321  54.7922 0.03 19.62 UPDATE toc_timeo
#    7 0x3A57B5252DB67EB8   159.7953  0.1%   111   1.4396 0.53  0.06 SELECT toc_timeo
#   11 0xDCD99199C9582625    33.2088  0.0%    27   1.2300 0.63  0.03 SELECT toc_timeo
#   12 0x44AE2DB8731CD39A    23.2433  0.0%    21   1.1068 0.74  0.01 SELECT toc_timeo
#   13 0xE3F31F032888B3F1    16.3693  0.0%    14   1.1692 0.57  0.01 SELECT toc_timeo
#   14 0x876FC2D749ECA875    15.5900  0.0%    13   1.1992 0.54  0.02 SELECT toc_timeo
# MISC 0xMISC              2112.8346  1.0%    11 192.0759   NS   0.0 <6 ITEMS>
```

explain

- **查看执行计划**
  - 选择了不好的索引



```
Database changed
root@tocl 12:52:58>explain sel[redacted]
    ->  wh[redacted] or [redacted]
    ->     and [redacted]in (3, 4, 9)
    ->     and [redacted] = 0 or [redacted] = 1)\G;
*************************** 1. row ***************************
           id: 1
  select_type: SIMPLE
        table: [redacted]
         type: range
possible_keys: [redacted]
          key: ind_t[redacted]gmttime
      key_len: 1
          ref: NULL
         rows: 194880
        Extra: Using where
1 row in set (0.00 sec)
```

哪些SQL在执行

- **Slow log**
  - Set global long_query_time=0
- **General log**
- **Binlog**
  - For DML, mysqlbinlog binlog解析
- **Processlist**
  - If some query is really slow
- **Tcpdump**
  - Tcpdump + mk-query-digest

Case 2: 很多MySQL线程都卡住了

- **Processlist**

```
Id: 1842782 User: provide Host: 192.168.0.1:59068 db: provide Command: Query
Time: 2326
State: Waiting for table
 Info: update table_xxxx set sold=sold+1, money=money+39800, Gmt_create=now() where xxxx_id=1
        and day='2011-10-07 00:00:00

Id: 1657130 User: provide Host: 192.168.0.2:40093 db: provide Command: Query
Time: 184551
State: Sending data
Info: select xxxx_id, sum(sold) as sold from  table_xxxx where xxxx_id in (select xxxx_id from table_xxxx
        where Gmt_create >= '2011-10-05 08:59:00') group by xxxx_id

1044    system user              Connect 27406   Flushing tables FLUSH TABLES
```

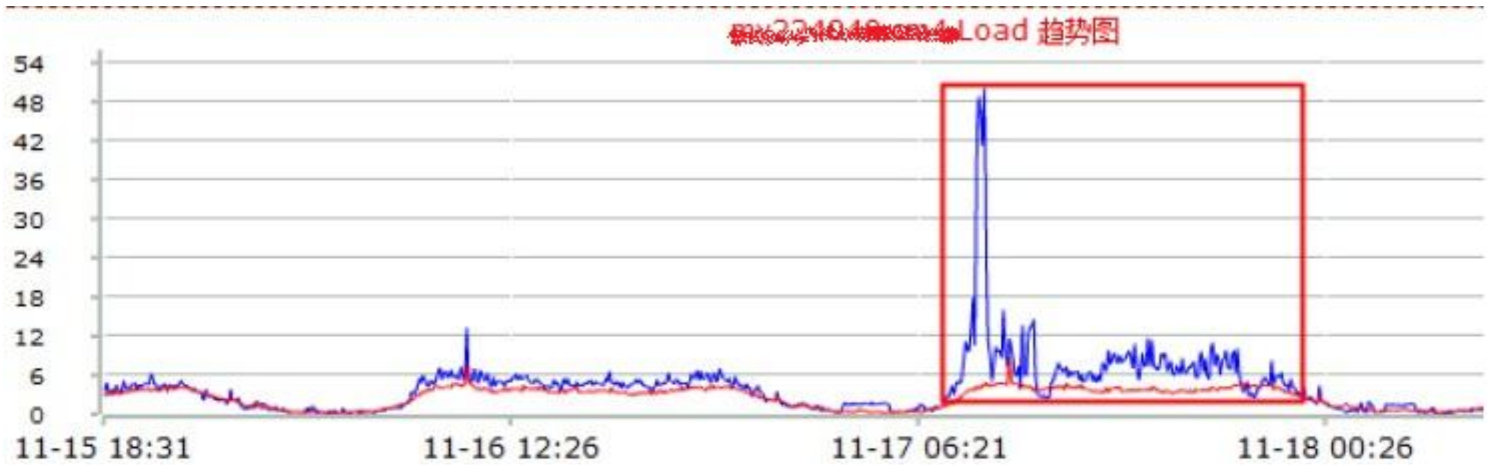
- **Pstack**

```
#0 0x0000003b4380ab99 in pthread_cond_wait@@GLIBC_2.3.2 ()
#1 0x00000000005aac4a in wait_for_refresh ()
#2 0x00000000005b2857 in open_table ()
#3 0x00000000005b312f in open_tables ()
#4 0x00000000005b3440 in open_and_lock_tables ()
#5 0x00000000005817a4 in mysql_execute_command ()
#6 0x0000000000586516 in mysql_parse ()
#7 0x0000000000586a65 in dispatch_command ()
#8 0x0000000000588923 in handle_one_connection ()
#9 0x0000003b438064a7 in start_thread () from /lib64/libpthread.so.0
```

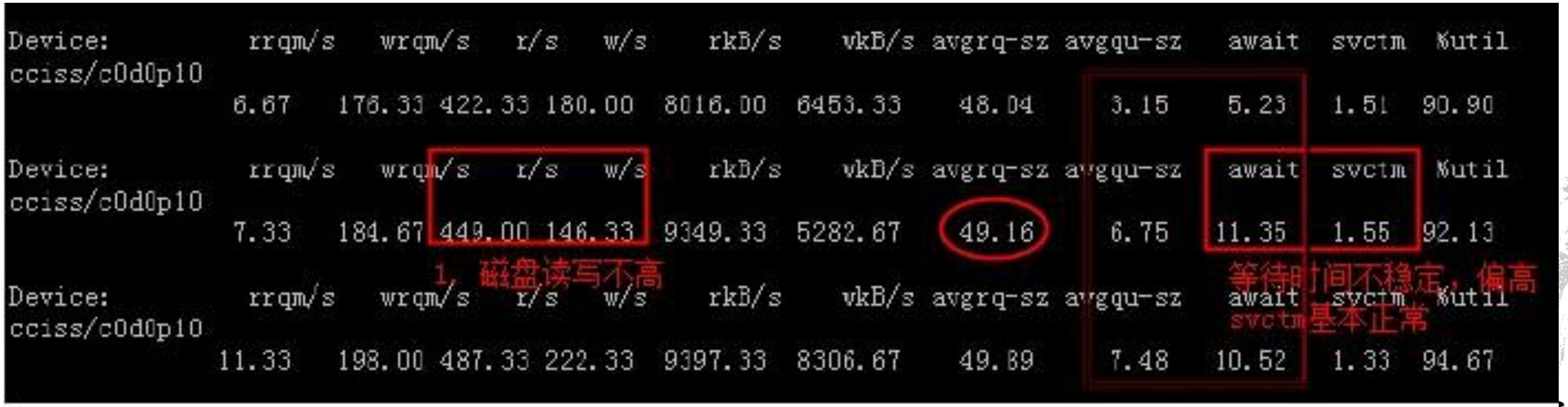
- **Processlist分析**
  - 谁是因，谁是果？
- **System user execute flush tables**
  - System user是谁, mysql主从复制（io thread, sql thread)
  - Binlog
- **谁最先执行了flush tables**
  - 人工执行？
  - App？ 没有权限
  - 定时任务，备份
    - Xtrabackup 会执行flush tables with read lock, 不记录到binlog
    - Mysqldump理论上不会执行flush tables，但如果有bug呢（ http://bugs.mysql.com/bug.php?id=35157 ）

Case 3: 服务器load高

- **调查问题**
  - SQL层面未见明显异常
  - 业务没有变动，没有发布
  - 调用量没有明显变化

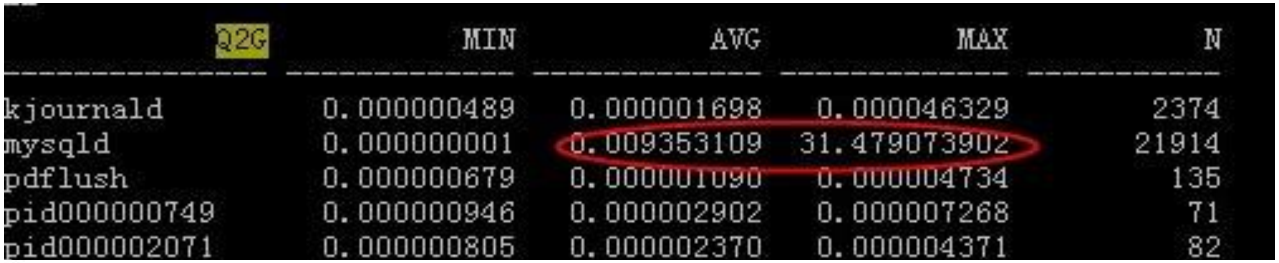


- **Iostat**
  - r/s, w/s
  - await, svctm
  - avgrq-sz

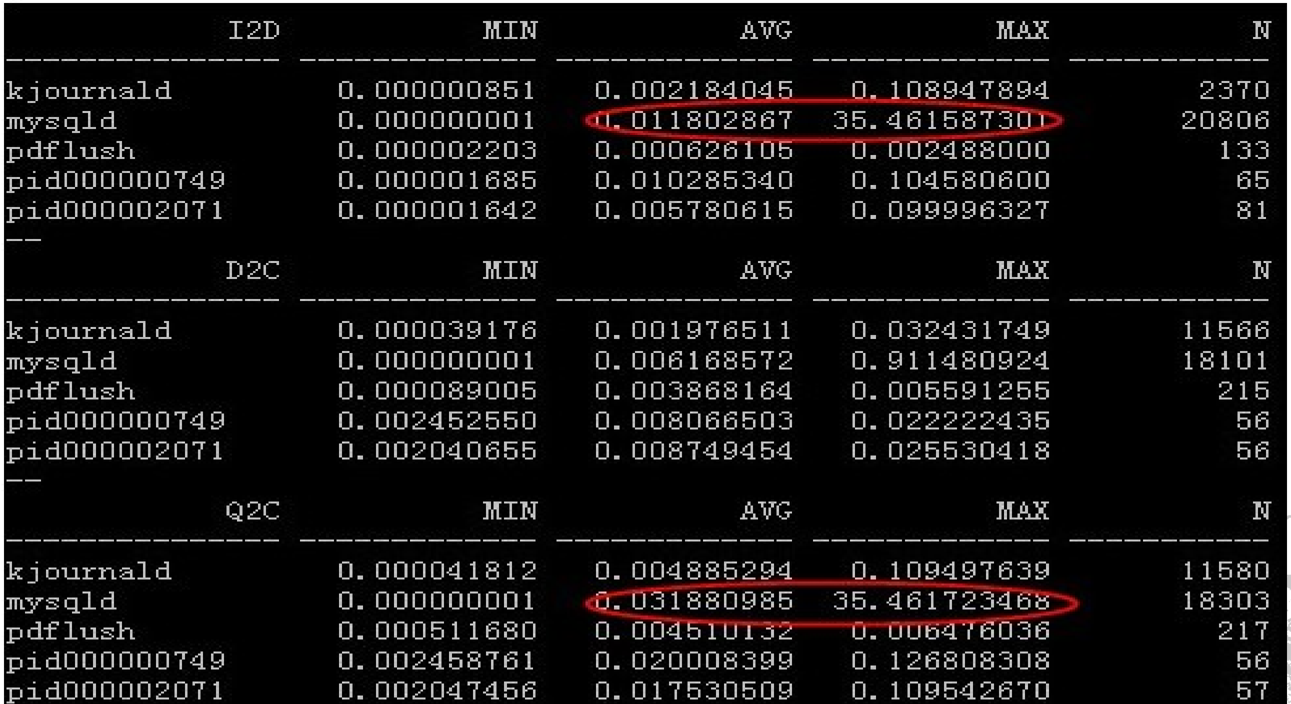


```
Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                  6.67   176.33  422.33  180.00  8016.00  6453.33    48.04     3.15    5.23   1.51  90.90

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                  7.33   184.67  449.00  146.33  9349.33  5282.67    49.16     6.75   11.35   1.55  92.13

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                 11.33   198.00  487.33  222.33  9397.33  8306.67    49.89     7.48   10.52   1.33  94.67
```

- **Blktrace, btt**

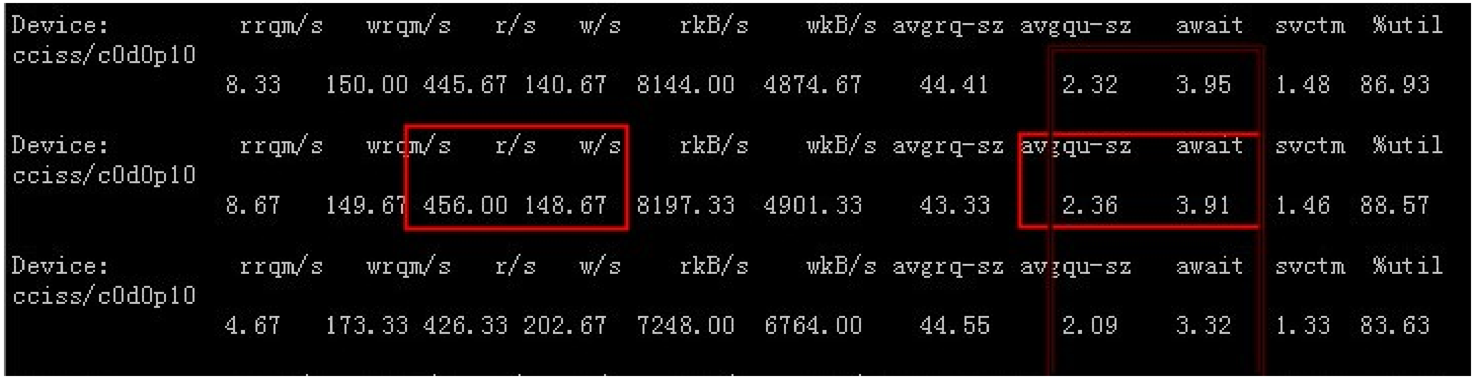


```
            Q2G            MIN            AVG            MAX           N
--------------- -------------- -------------- -------------- -----------
kjournald         0.000000489    0.000001698    0.000046329        2374
mysqld            0.000000001    0.009353109   31.479073902       21914
pdflush           0.000000679    0.000001090    0.000004734         135
pid000000749      0.000000946    0.000002902    0.000007268          71
pid000002071      0.000000805    0.000002370    0.000004371          82
```

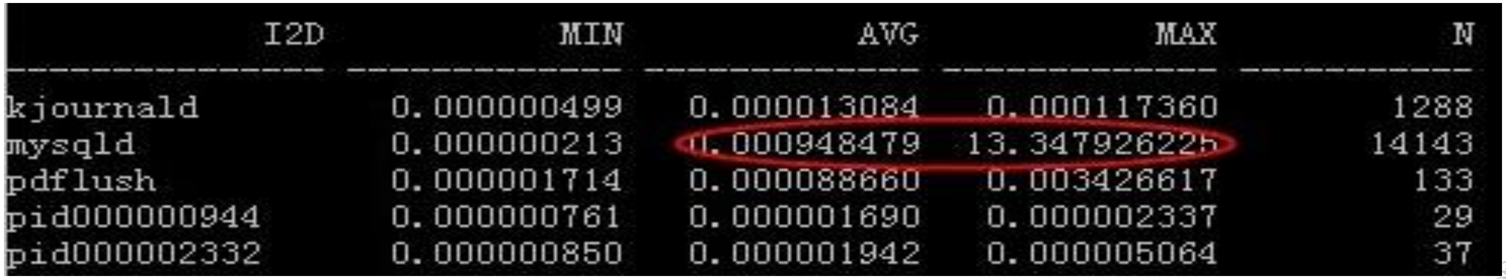


```
            I2D            MIN            AVG            MAX           N
--------------- -------------- -------------- -------------- -----------
kjournald         0.000000851    0.002184045    0.108947894        2370
mysqld            0.000000001    0.011802867   35.461587301       20806
pdflush           0.000002203    0.000626105    0.002488000         133
pid000000749      0.000001685    0.010285340    0.104580600          65
pid000002071      0.000001642    0.005780615    0.099996327          81
---
            D2C            MIN            AVG            MAX           N
--------------- -------------- -------------- -------------- -----------
kjournald         0.000039176    0.001976511    0.032431749       11566
mysqld            0.000000001    0.006168572    0.911480924       18101
pdflush           0.000089005    0.003868164    0.005591255         215
pid000000749      0.002452550    0.008066503    0.022222435          56
pid000002071      0.002040655    0.008749454    0.025530418          56
---
            Q2C            MIN            AVG            MAX           N
--------------- -------------- -------------- -------------- -----------
kjournald         0.000041812    0.004885294    0.109497639       11580
mysqld            0.000000001    0.031880985   35.461723468       18303
pdflush           0.000511680    0.004510132    0.006476036         217
pid000000749      0.002458761    0.020008399    0.126808308          56
pid000002071      0.002047456    0.017530509    0.109542670          57
```

- **IO调度算法**
  - cfq -> deadline



```
Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                  8.33   150.00  445.67  140.67  8144.00  4874.67    44.41     2.32    3.95   1.48  86.93

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                  8.67   149.67  456.00  148.67  8197.33  4901.33    43.33     2.36    3.91   1.46  88.57

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
cciss/c0d0p10
                  4.67   173.33  426.33  202.67  7248.00  6764.00    44.55     2.09    3.32   1.33  83.63
```



```
            I2D              MIN            AVG            MAX           N
--------------- -------------- -------------- -------------- ------------
kjournald         0.000000499    0.000013084    0.000117360         1288
mysqld            0.000000213    0.000948479   13.347926225        14143
pdflush           0.000001714    0.000088660    0.003426617          133
pid000000944      0.000000761    0.000001690    0.000002337           29
pid000002332      0.000000850    0.000001942    0.000005064           37
```

Case 4: DDL lost table

- **alert.log大量报错**
  - 持续10几分钟后，Table lost。
- **几百个进程都block在”opening tables”，这些表都不是DDL的那个表**

丢表时的alert.log

```
110803  2:15:02  InnoDB: Warning: problems renaming 'feel_23/#sql-2635_23da0d' to
    'feel_23/feed_send_1476', 24998 iterations
InnoDB: Warning: tablespace './feel_23/#sql-2635_23da0d.ibd' has i/o ops stopped for a long
    time 24998
110803  2:15:02  InnoDB: Warning: problems renaming 'feel_xx/#sql-2635_23da0d' to
    'feel_xx/feed_send_xxxx', 24999 iterations
InnoDB: Warning: tablespace './feel_23/#sql-2635_23da0d.ibd' has i/o ops stopped for a long
    time 24999
110803  2:15:02  InnoDB: Warning: problems renaming 'feel_xx/#sql-2635_23da0d' to
    'feel_xx/feed_send_xxxx', 25000 iterations
InnoDB: Warning: tablespace './feel_23/#sql-2635_23da0d.ibd' has i/o ops stopped for a long
    time 25000
110803  2:15:02  InnoDB: Warning: problems renaming 'feel_xx/#sql-2635_23da0d' to
    'feel_xx/feed_send_xxxx', 25001 iterations
110803  2:15:02 [ERROR] Cannot find or open table feel_23/feed_send_1476 from
the internal data dictionary of InnoDB though the .frm file for the
table exists. Maybe you have deleted and recreated InnoDB data
files but have forgotten to delete the corresponding .frm files
of InnoDB tables, or you have moved .frm files to another database?
or, the table contains indexes that this version of the engine
doesn't support.
See http://dev.mysql.com/doc/refman/5.1/en/innodb-troubleshooting.html
how you can resolve the problem.
```

Pstack - master thread

```
#0  0x000000364aacced2 in select () from /lib64/libc.so.6
#1  0x00002aaab2e595fb in os_thread_sleep ()
#2  0x00002aaab2e18838 in fil_mutex_enter_and_prepare_for_io ()
#3  0x00002aaab2e18aa5 in fil_io ()
#4  0x00002aaab2df5b63 in buf_flush_buffered_writes ()
#5  0x00002aaab2df6048 in buf_flush_batch ()
#6  0x00002aaab2ea13d8 in srv_master_thread ()
#7  0x000000364b6064a7 in start_thread () from /lib64/libpthread.so.0
#8  0x000000364aad3c2d in clone () from /lib64/libc.so.6
```

Pstack – alter table

```
#0  0x000000364aacced2 in select () from /lib64/libc.so.6
#1  0x00002aaab2e595fb in os_thread_sleep ()
#2  0x00002aaab2e1a3e2 in fil_rename_tablespace ()
#3  0x00002aaab2e0672b in dict_table_rename_in_cache ()
#4  0x00002aaab2e86af5 in row_rename_table_for_mysql ()
#5  0x00002aaab2e316db in ha_innodb::rename_table ()
#6  0x00000000006bea6c in mysql_rename_table ()
#7  0x00000000006c77ff in mysql_alter_table ()
#8  0x00000000005c6a8e in mysql_execute_command ()
#9  0x00000000005cd371 in mysql_parse ()
#10 0x00000000005cd773 in dispatch_command ()
#11 0x00000000005cea04 in do_command ()
#12 0x00000000005bf0d7 in handle_one_connection ()
```

```
retry:    (fil_rename_tablespace)
   ... Some lines ommitted here(print warning,...)
   space->stop_ios = TRUE;
   if (node->n_pending > 0 || node->n_pending_flushes > 0) {
          /* There are pending i/o's or flushes, sleep for a while and
          retry */
          mutex_exit(&fil_system->mutex);
          os_thread_sleep(20000);
          goto retry;
    } else if (node->modification_counter > node->flush_counter) {
          /* Flush the space */
          mutex_exit(&fil_system->mutex);
          os_thread_sleep(20000);
          fil_flush(id);
          goto retry;
    } else if (node->open) {
          /* Close the file */
          fil_node_close_file(node, fil_system);
    }
    space->stop_ios = FALSE;
    mutex_exit(&fil_system->mutex);
```

```
retry:(fil_mutex_enter_and_prepare_for_io)
    mutex_enter(&fil_system->mutex);
    if (space_id == 0 || space_id >= SRV_LOG_SPACE_FIRST_ID) {
           return;
    }
    if (fil_system->n_open < fil_system->max_n_open) {
           return;
    }
    space = fil_space_get_by_id(space_id);
    if (space != NULL && space->stop_ios) {
           /* We are going to do a rename file and want to stop new i/o's
           for a while */
           if (count2 > 20000) {
                      fputs("InnoDB: Warning: tablespace ", stderr);
                      ut_print_filename(stderr, space->name);
                      fprintf(stderr,
                                " has i/o ops stopped for a long time %lu\n",
                                (ulong) count2);
           }
           mutex_exit(&fil_system->mutex);
           os_thread_sleep(20000);
           count2++;
           goto retry;
      }
```

```
mysql_alter_table() {
    error= copy_data_between_tables(table, new_table,
                    ...
     VOID(pthread_mutex_lock(&LOCK_open));

    if (mysql_rename_table(old_db_type, db, table_name, db, old_name,
                         FN_TO_IS_TMP)){
          error=1;
          VOID(quick_rm_table(new_db_type, new_db, tmp_name, FN_IS_TMP));
    }  else if (mysql_rename_table(new_db_type, new_db, tmp_name, new_db,
                              new_alias, FN_FROM_IS_TMP) ||
           ((new_name != table_name || new_db != db) &&
           (need_copy_table != ALTER_TABLE_METADATA_ONLY ||
            mysql_rename_table(save_old_db_type, db, table_name, new_db,
                               new_alias, NO_FRM_RENAME)) &&
           Table_triggers_list::change_table_name(thd, db, table_name,
                                                  new_db, new_alias))) {
    error=1;
          VOID(quick_rm_table(new_db_type,new_db,new_alias, 0));
          VOID(quick_rm_table(new_db_type, new_db, tmp_name, FN_IS_TMP));
          VOID(mysql_rename_table(old_db_type, db, old_name, db, alias,
                            FN_FROM_IS_TMP));
    }
     VOID(pthread_mutex_unlock(&LOCK_open));
}
```

Case 5: MyISAM

- **SQL**

```
select count(*) as num
from vid_film left join vid_class_map
on vid_film.id=vid_class_map.v_id
and vid_class_map.s_type=1 and vid_class_map.class_id=1;
+--------+
| num |
+--------+
| 13536 |
```

Orzdba

- **Qps不到50，cpu耗尽(user 25%, sys 75%)**

```
-----load-avg---- ---cpu-usage--- ---swap--- -QPS- -TPS- -Hit%-
time | 1m 5m 15m |usr sys idl iow| si so| ins upd del sel iud lor
hit |
15:39:20|11.09 5.59 5.07| 26 74 0 0| 0 0| 0 0 0 4 0 0 100.00|
15:39:22|11.09 5.59 5.07| 25 75 0 0| 0 0| 0 0 0 47 0 0 100.00|
15:39:24|14.20 6.33 5.31| 25 74 0 0| 0 0| 0 0 0 8 0 0 100.00|
15:39:25|14.20 6.33 5.31| 26 74 0 0| 0 0| 0 0 0 44 0 0 100.00|
15:39:27|14.20 6.33 5.31| 26 74 0 0| 0 0| 0 0 0 8 0 0 100.00|
15:39:28|14.20 6.33 5.31| 26 74 0 0| 0 0| 0 0 0 36 0 0 100.00|
```

vmstat

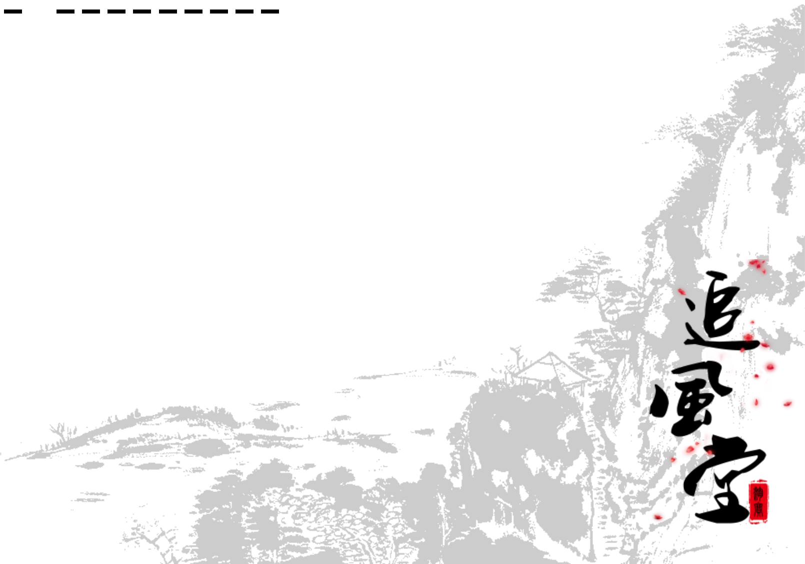
- **CS每秒25万次**

```
procs -----------memory---------- ---swap-- -----io---- --system-- -----
cpu------
r b swpd free buff cache si so bi bo in cs us sy id wa st
49 0 256 2139532 1623780 19942492 0 0 0 68 1027 256677 25 75 0 0 0
48 0 256 2139532 1623784 19942492 0 0 0 16 1007 251610 25 75 0 0 0
47 0 256 2139548 1623784 19942500 0 0 0 0 1003 251741 25 75 0 0 0
33 0 256 2139548 1623784 19942504 0 0 0 0 1003 251605 25 75 0 0 0
49 0 256 2139424 1623792 19942500 0 0 0 36 1007 251313 25 74 0 0 0
50 0 256 2139424 1623792 19942508 0 0 0 52 1006 251436 25 74 0 0 0
```

strace mysqld

- **大量futex调用**

```
% time seconds usecs/call calls errors syscall
------ ----------- ----------- --------- ---------
84.93 53.100058 145 365451 127938 futex
14.65 9.159573 277563 33 select
0.16 0.099231 149 668 read
0.13 0.084391 126 668 668 sched_setparam
0.13 0.078847 159 496 write
------ ----------- ----------- --------- ---------
100.00 62.522100 367316 128606 total
```



Oprofile global

```
CPU: CPU with timer interrupt, speed 0 MHz (estimated)
Profiling through timer interrupt
samples % linenr info app name symbol name
2200675 40.3334 processor_idle.c:222 vmlinux acpi_processor_idle
1131206 20.7324 futex.c:1973 vmlinux do_futex
893066 16.3679 futex.c:727 vmlinux futex_wake
489614 8.9735 (no location information) libpthread-2.5.so pthread_mutex_lock
244503 4.4812 thread_info.h:63 vmlinux acpi_safe_halt
66355 1.2161 sched.c:2056 vmlinux thread_return
66173 1.2128 futex.c:603 vmlinux wake_futex
50622 0.9278 (no location information) libpthread-2.5.so pthread_mutex_unlock
39782 0.7291 (no location information) libpthread-2.5.so __lll_lock_wait
27363 0.5015 mf_keycache.c:1666 mysqld find_key_block
25902 0.4747 my_handler.c:124 mysqld ha_key_cmp
25624 0.4696 mf_keycache.c:2542 mysqld key_cache_read
22179 0.4065 bmove512.c:40 mysqld bmove512
21092 0.3866 mf_keycache.c:1349 mysqld unreg_request
14228 0.2608 mi_search.c:184 mysqld _mi_bin_search
11051 0.2025 futex.c:2020 vmlinux sys_futex
```

Oprofile mysqld

```
CPU: CPU with timer interrupt, speed 0 MHz (estimated)
Profiling through timer interrupt
samples % linenr info symbol name
27363 14.8909 mf_keycache.c:1666 find_key_block
25902 14.0958 my_handler.c:124 ha_key_cmp
25624 13.9445 mf_keycache.c:2542 key_cache_read
22179 12.0697 bmove512.c:40 bmove512
21092 11.4782 mf_keycache.c:1349 unreg_request
14228 7.7428 mi_search.c:184 _mi_bin_search
4927 2.6813 mi_search.c:64 _mi_search
4069 2.2143 mi_key.c:220 _mi_pack_key
4046 2.2018 sql_select.cc:11896 join_read_key(st_join_table*)
3116 1.6957 sql_select.cc:11345 sub_select(JOIN*, st_join_table*, bool)
2995 1.6299 sql_select.cc:11413 evaluate_join_record(JOIN*, st_join_table*, int)
2748 1.4955 mi_rkey.c:26 mi_rkey
2674 1.4552 mi_key.c:468 _mi_read_key_record
2170 1.1809 mi_page.c:25 _mi_fetch_keypage
1648 0.8968 item_sum.cc:1193 Item_sum_count::add()
1588 0.8642 mi_rnext.c:28 mi_rnext
1564 0.8511 sql_select.cc:12360 end_send_group(JOIN*, st_join_table*, bool)
```

pstack

典型的调用栈

```
#0 0x000000321220d174 in __lll_lock_wait () from /lib64/libpthread.so.0
#1 0x0000003212208b00 in _L_lock_1233 () from /lib64/libpthread.so.0
#2 0x0000003212208a83 in pthread_mutex_lock () from /lib64/libpthread.so.0
#3 0x00000000008889db in key_cache_read ()
#4 0x000000000083fe28 in _mi_fetch_keypage ()
#5 0x000000000083efa1 in _mi_search ()
#6 0x000000000083f14d in _mi_search ()
#7 0x000000000083f692 in _mi_search_next ()
#8 0x000000000083bbf0 in mi_rnext ()
#9 0x00000000008332b5 in ha_myisam::index_next ()
#10 0x00000000006393eb in join_read_next ()
#11 0x00000000006243ba in sub_select ()
#12 0x0000000000636e2d in do_select ()
#13 0x00000000006442f7 in JOIN::exec ()
#14 0x0000000000645ff8 in mysql_select ()
#15 0x00000000006469a7 in handle_select ()
```

Summary

- **了解系统各个部件的工作原理**
  - MySQL和InnoDB
  - SQL和索引
  - OS 进程、线程、内存
  - IO、文件操作
  - 硬件的能力（响应时间、吞度量）
- **数据采集和观测**
  - 熟悉工具
  - 收集各个部件的数据，数据的采集和保存
  - 汇总，对比，分析
  - 重视证据
- **问题的解决**
  - 应用层优化（设计、SQL优化、索引调整)
  - 参数调整（MySQL, OS）
  - patch

谢谢大家