



Diagnosing Intermittent Performance Problems

Baron Schwartz

Today's Agenda

- Diagnosing intermittent MySQL problems
 - What kind of problems are we talking about?
 - Why are they hard to solve?
 - What approaches can solve them successfully?
 - What tools can help you do it more quickly?
 - How can you set up and use these tools?
 - How do you interpret the results?
 - Case Studies

Intermittent Problems

- Happen at random times
- Hard to observe in action
- No obvious reason

What Kinds Of Problems?

- In general, we see three kinds
 - Randomly slow query
 - Sudden error message
 - Server-wide “stalls”

What Kinds Of Problems?

- Real customer examples:
 - “My server seems to freeze for ten seconds to a minute at random times. Suddenly, everything clears up again. It seems to happen for no reason.”
 - “I get sporadic 'too many connections' errors. Increasing max_connections doesn't help. This is not related to my peak load.”

How Hard Can It Be?

- It's hard to troubleshoot when you can't see it.
 - “Our graphs show this happens for 1 to 3 minutes once or twice a week.”
- It's hard to get support when it's not reproducible.
 - “Our support staff thinks that we are imagining it.”
 - “We filed a bug, but it was closed because we can't create a test case.”

How Hard Can It Be?

- It can go on forever.
 - “We've been working on this for nearly 5 months.”

Why Does This Happen?

- More CPUs
- More memory
- More popularity
- Cloud computing

How Not To Do It

- DON'T try to use “tuning scripts”
- DON'T try to change server settings
- DON'T try rebooting everything
- DON'T do \$random_stab_in_the_dark
- DON'T try upgrading or replacing components

The Fruits of Trial-And-Error

- “I think this might be related to your networking. Can you try buying a new switch?”

The Fruits of Trial-And-Error

- “I think this might be related to your networking. Can you try buying a new switch?”
- “Oh, that didn't solve it? Hmmm..... let me think.”

The Fruits of Trial-And-Error

- “I think this might be related to your networking. Can you try buying a new switch?”
- “Oh, that didn't solve it? Hmmmm..... let me think.”
- “I saw someone else on the Internet with a problem like this. They said that switching from Debian to Red Hat fixed it. Can you try that?”

The Fruits of Trial-And-Error

- “It still happens? Oh wow. What version of Java are you using? Can you [upgrade|downgrade] that?”

The Fruits of Trial-And-Error

... Time passes...

- “Sorry, I really don't know. Well, this is a free forum, so at least this didn't cost you anything.”

Measure, Measure, Measure

- You cannot fix what you cannot measure.

How Do I Measure?

- Ideally, measure in three ways:
- Completely.
- Correctly timed.
- Correctly scoped.

What Should I Measure?

- Everything.
 - Yes, it's a lot of data.
 - Schwartz's Law: whatever you don't measure is what you needed to measure.

I Never See It Happen

- You need automatic tools watching for it.
- We've developed good tools for this.

Using Percona Toolkit

- Percona Toolkit = Maatkit + Aspersa
- The primary tools for this are:
 - pt-stalk: wait for something to happen, then gather tons of diagnostic data for a short time
 - pt-sift: look for needles in the pt-stalk haystack

Finding a Trigger

- Find a reliable way to detect the problem
- Getting this right is the foundation!
- Use this as a trigger for pt-stalk.

Example

```
$ mysqladmin ext -i1 | awk '/Queries/{q=$4-qp;qp=$4}/Threads_connected/{tc=$4}/Threads_running/{printf "%5d %5d %5d\n", q, tc, $4}'
```

798	136	7
767	134	9
828	134	7
683	134	7
784	135	7
614	134	7
108	134	24
187	134	31
179	134	28
1179	134	7
1151	134	7
1240	135	7
1000	135	7

Example

```
$ mysqladmin ext -i1 | awk '/Queries/{q=$4-qp;qp=$4}/Threads_connected/{tc=$4}/Threads_running/{printf "%5d %5d %5d\n", q, tc, $4}'
```

798	136	7
767	134	9
828	134	7
683	134	7
784	135	7
614	134	7
108	134	24
187	134	31
179	134	28
1179	134	7
1151	134	7
1240	135	7
1000	135	7

Configuring pt-stalk

- Set `--threshold=15`
- Set `--variable=Threads_running`
- Then run pt-stalk as root
- You may need to install and enable:
 - GDB for backtraces (wait analysis)
 - Oprofile for server profiling

Looking At The Data

```
Terminal - baron@ginger:~/collected
2011_07_21_11_08_20-opentables1      2011_07_21_11_40_41-diskstats
2011_07_21_11_08_20-opentables2      2011_07_21_11_40_41-hostname
2011_07_21_11_08_20-output           2011_07_21_11_40_41-innodbstatus1
2011_07_21_11_08_20-pmap             2011_07_21_11_40_41-innodbstatus2
2011_07_21_11_08_20-processlist1     2011_07_21_11_40_41-interrupts
2011_07_21_11_08_20-processlist2     2011_07_21_11_40_41-iostat
2011_07_21_11_08_20-procstat         2011_07_21_11_40_41-iostat-overall
2011_07_21_11_08_20-procvmstat       2011_07_21_11_40_41-log_error
2011_07_21_11_08_20-ps               2011_07_21_11_40_41-lsof
2011_07_21_11_08_20-slabinfo         2011_07_21_11_40_41-meminfo
2011_07_21_11_08_20-stacktrace       2011_07_21_11_40_41-mpstat
2011_07_21_11_08_20-sysctl           2011_07_21_11_40_41-mpstat-overall
2011_07_21_11_08_20-top              2011_07_21_11_40_41-mutex-status1
2011_07_21_11_08_20-trigger          2011_07_21_11_40_41-mutex-status2
2011_07_21_11_08_20-variables        2011_07_21_11_40_41-mysqldadmin
2011_07_21_11_08_20-vmstat           2011_07_21_11_40_41-netstat_s
2011_07_21_11_08_20-vmstat-overall   2011_07_21_11_40_41-opentables1
2011_07_21_11_10_31-df               2011_07_21_11_40_41-opentables2
2011_07_21_11_10_31-diskstats        2011_07_21_11_40_41-output
2011_07_21_11_10_31-hostname         2011_07_21_11_40_41-pmap
2011_07_21_11_10_31-innodbstatus1    2011_07_21_11_40_41-processlist1
2011_07_21_11_10_31-innodbstatus2    2011_07_21_11_40_41-processlist2
2011_07_21_11_10_31-interrupts       2011_07_21_11_40_41-procstat
2011_07_21_11_10_31-iostat          2011_07_21_11_40_41-procvmstat
```


Using pt-sift

```
Terminal - baron@ginger:~/collected
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$
[baron@ginger collected]$ pt-sift .

2011_07_21_10_14_58 2011_07_21_10_16_49 2011_07_21_10_19_16
2011_07_21_10_22_02 2011_07_21_10_31_55 2011_07_21_10_40_36
2011_07_21_10_44_47 2011_07_21_10_54_48 2011_07_21_10_57_36
2011_07_21_11_00_11 2011_07_21_11_01_54 2011_07_21_11_04_05
2011_07_21_11_06_13 2011_07_21_11_08_20 2011_07_21_11_10_31
2011_07_21_11_12_40 2011_07_21_11_14_51 2011_07_21_11_16_59
2011_07_21_11_19_09 2011_07_21_11_21_16 2011_07_21_11_23_27
2011_07_21_11_25_35 2011_07_21_11_27_44 2011_07_21_11_29_35
2011_07_21_11_32_03 2011_07_21_11_34_12 2011_07_21_11_36_23
2011_07_21_11_38_30 2011_07_21_11_40_41

Select a timestamp from the list [2011_07_21_11_40_41]
```

Case Study

```
Terminal - baron@ginger:~/collected
--vmstat--
 r b swpd  free   buff   cache si so bi   bo    in    cs us sy id wa st
42 0 15840 760604 142084 16869724 0 0 36   381    0    0 7  1 92 0 0
 1 0 15840 742016 142284 16911564 0 0 73 13053 31329 56995 11 3 86 0 0
wa 0% . . . . .
--innodb--
  txns: 10xACTIVE (1s) 310xnot (0s)
  0 queries inside InnoDB, 0 queries in queue
  Main thread: flushing buffer pool pages, pending reads 1, writes 6, flush 0
  Log: lsn = 1777474532609, chkp = 1776282083670, chkp age = 1192448939
  Threads are waiting at:
  Threads are waiting on:
--processlist--
  State
  494
    19 Sending data
    5 Has sent all binlog to slave; waiting for binlog to be updated
    3 Reading from net
    2 freeing items
  Command
  496 Sleep
    25 Query
    6 Connect
    5 Binlog Dump
```