MySQL Conference & Expo

Presented by MySQL Conference & Expo O'REILLY

Monitoring MySQL

Kristian Köhntopp
Consultant
MySQL AB
<kris@mysql.com>

I am...

- Kristian Köhntopp <kris@mysql.com>
- Principal Consultant for MySQL
- Architecture, Scaleout and Performance Tuning
- In former lives:

  Chief Security Guy for a large web hoster, University Teacher for Security Management, Security Consultant, Performance Consultant, Developer for Web Applications, Developer of some PHP-Extensions, PHPLIB Developer, Linux HOWTO-Author

- Unix 1987, Online 1988, online 1992, PHP + MySQL since 1997

You are

- Job...

  DBA, Developer, General IT, IT Management

- Using version...

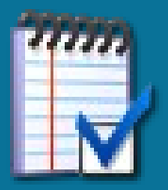
  3.23, 4.0, 4.1, 5.0, 5.1

- Using MySQL for...

  Webapps, Enterprise, Embedded, ...

- Number of servers...

  1, <3, <10, <25, <100, 100 or more



Agenda: What kind of Monitoring?

- Goals and Deliverables of Monitoring
  - Functional checks
  - Compliance
  - Capacity planning
  - Debugging
  - Auditing
- Metering what?
- The Toolbox

What kind of Monitoring?

- Functional checks & incident detection (Are we still online?)
- Compliance (Are we fulfilling our contracts?)
- Capacity planning (monitoring for long term trends)
- For debug (individual query tracking)
- For auditing (Inescapable, unforgeable high level change + access record)

Different Monitoring, different requirements

- Each kind of monitoring has different
  - Purpose and goal
  - May require different metering points
  - Has a different deliverable
  - Has different HA requirements

Functional checks

- Purpose: Find problems early, signal them and kick off incident handling.
- Metering point: “High level availability test”
- Deliverable: Ticket to Helpdesk -> Operating -> Incident Management
- HA Requirement: Minutes, high

Compliance

- Purpose: Reporting availability to document SLA fulfillment
- Metering point: “high level availability tests”
- Deliverable: weekly/monthly report to IT management/customer
- HA Requirement: days, low

Capacity planning

- Purpose: Detect potential problems in advance, planning invest
- Metering points: detailed metering of variables in all subsystem
- Deliverable: weekly/monthly report to IT Management, general Management
- HA Requirement: days/low

For debug

- Purpose: Trace individual queries for a developer audience
- Metering points: detailed metering of variables while processing a single query
- Deliverable: individual report to single developer
- HA Requirement: N/A

For Audit

- Purpose: Detect tampering, alteration and access, create accountability records for changes and access
- Metering points: ideally, “functional gates”, high level event records with application semantics

  additional requirements: inescapeable, unforgeable event records
- Deliverable: daily/weekly report to IT Management
- HA Requirement: N/A (must not affect regular operations)

Agenda: Metering what?

- Goals and Deliverables of Monitoring
- Metering what?
  - Data sources at OS level
  - Data sources in MySQL
  - Derived data sources
- The Toolbox

Data sources at OS level

- Availability
  - internally: presence of PID file, presence of process
    - test -f linux.pid is not good enough
    - “kill -0 $(cat linux.pid)” is better than “ps axuwww | grep mysql[d]”
  - externally: ping check, trivial query
    - Set timeouts for the trivial query according to SLA

Data sources at OS level

- process size in memory
- buffer cache size (“free -m”)
- swap check!
  - vm.swappiness = 0
- “iostat -x 1 3” output
  - In general, databases are limited by seek/sec, not MB/sec
- Network I/O quality
  - Smokeping? (Cluster!)

Data sources in MySQL

- Internal data sources:
  - SHOW /*!50000 GLOBAL */ STATUS;
  - SHOW /*!50000 GLOBAL */ VARIABLES;
  - SHOW FULL PROCESSLIST;
  - SHOW TABLE STATUS; SHOW OPEN TABLES;
  - SHOW SLAVE STATUS; SHOW MASTER STATUS;
  - SHOW ENGINE INNODB STATUS; SHOW GLOBAL STATUS LIKE 'inno%';
  - SHOW MUTEX STATUS;

SHOW GLOBAL STATUS: General

- qps: questions/uptime
  - Order of magnitude: 10s, 100s, 1000s
- com_% Counters
  - Read/Writes:
    (select + qcache_hits) / (insert+update+delete+replace)
  - Transactions:
    #commit, rollback/commit, writes/commit
  - SET-Commands:
    JDBC?

SHOW GLOBAL STATUS: General

- Table Cache
  - (opened_tables/sec)
  - (table_cache_size – open_tables)
- Thread Cache
  - (threads_created/sec)
  - (thread_cache_size – threads_cached)
- Connections
  - max_connections – max_used_connections
  - max_connections - threads_connected

SHOW GLOBAL STATUS: Query Cache

- Hit-Ratio:
  - qcache_hits*100 / (qcache_hits + com_select)
- Hits vs. Inserts
- Lowmem Prunes:
  - qcache_lowmem_prunes / uptime
  - qcache_lowmem_prunes per second
- Increase query cache size: less prunes, higher hit ratio
  - Sometimes it is better to delay writes or to split tables instead

SHOW GLOBAL STATUS: MyISAM

- Key Cache Hit Ratio:
  - 100- (key_reads *100 / key_read_requests)

    99.3% - 99.9% target

- MyISAM Lock Contention
  - table_locks_waited * 100 / table_locks_immediate
  - <1% good, 1% warning, >3% you are currently dying
  - distinctly nonlinear behavior

SHOW GLOBAL STATUS: Innodb

- Page Cache Usage:
  - Innodb_buffer_pool_pages_free *100 / Innodb_buffer_pool_pages_total
- Cache Hit Ratio:
  - 100 - (Innodb_buffer_pool_reads * 100 / Innodb_buffer_pool_read_requests)
  - target: 96%-99%
- Cache Monitoring: Innodb_buffer_pool_wait_free must not count up!
- Log-Monitoring: Innodb_log_waits must not count up!

SHOW GLOBAL STATUS: Temp tables

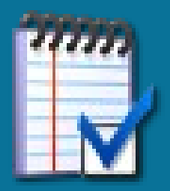
- Temp tables per second:
  - created_tmp_tables
- Temp tables to Disk:
  - created_disk_tmp_tables * 100 / created_tmp_tables
- Additional hints:
  - What kind of filesystem is tmpdir pointing to?
  - Are we selecting BLOB/TEXT types?
  - tmp_table_size and max_heap_table_size must match

SHOW SLAVE STATUS: Replication

- Functionality:
  - Slave_IO_running: YES, Slave_SQL_running: YES
- Lag:
  - Seconds_behind_master
- Rate:
  - Read_Master_Log_Pos/sec, Exec_Master_Log_Pos/sec
- Binlog Cache:
  - binlog_cache_disk_use * 100 / binlog_cache_use

SHOW GLOBAL STATUS: Slow Queries

- Slow Queries in general:
  - Slow_queries/sec
- Counting evil queries:
  - select_full_join / sec
  - select_full_join / com_select



Agenda: The Toolbox

- Goals and Deliverables of Monitoring
- Metering what?
  -
- The Toolbox
  - Availability: Nagios family, SNMP trap based tools
  - Trend and load monitoring: SNMP bridges, Cacti, Munin, mysqlar

Tools: Nagios

- Nagios
- MySQL Tools at http://www.nagiosexchange.org/Databases.57.0.html
- In C: http://blog.koehntopp.de/archives/1553-guid.html
- Many checks are
  - written in Shell or Perl (slow and large)
  - not compliant with Nagios Plugin Development Guidelines
  - in general they work as advertised
- Nagios also covers compliance (That is not a good idea!)

Tools: SNMP

- Free SNMP tools are rare
- Exporting Variables, Status and Slave Status to SNMP:
  - Perl Coprocess a la (broken) http://mysqldump.azundris.com/archives/63-guid.html
- SNMP is broken in many interesting ways, but
  - alternatives do not really exist
- I have a longer rant available on this, if you are interested.

Tools: Cacti

- Oh! So shiny!
- Creating templates is a pain!
- Ready-made templates from
  - http://faemalia.net/mysqlUtils/
- Can Cacti plot derives variables?
  - Even those dependent on two measurement values?

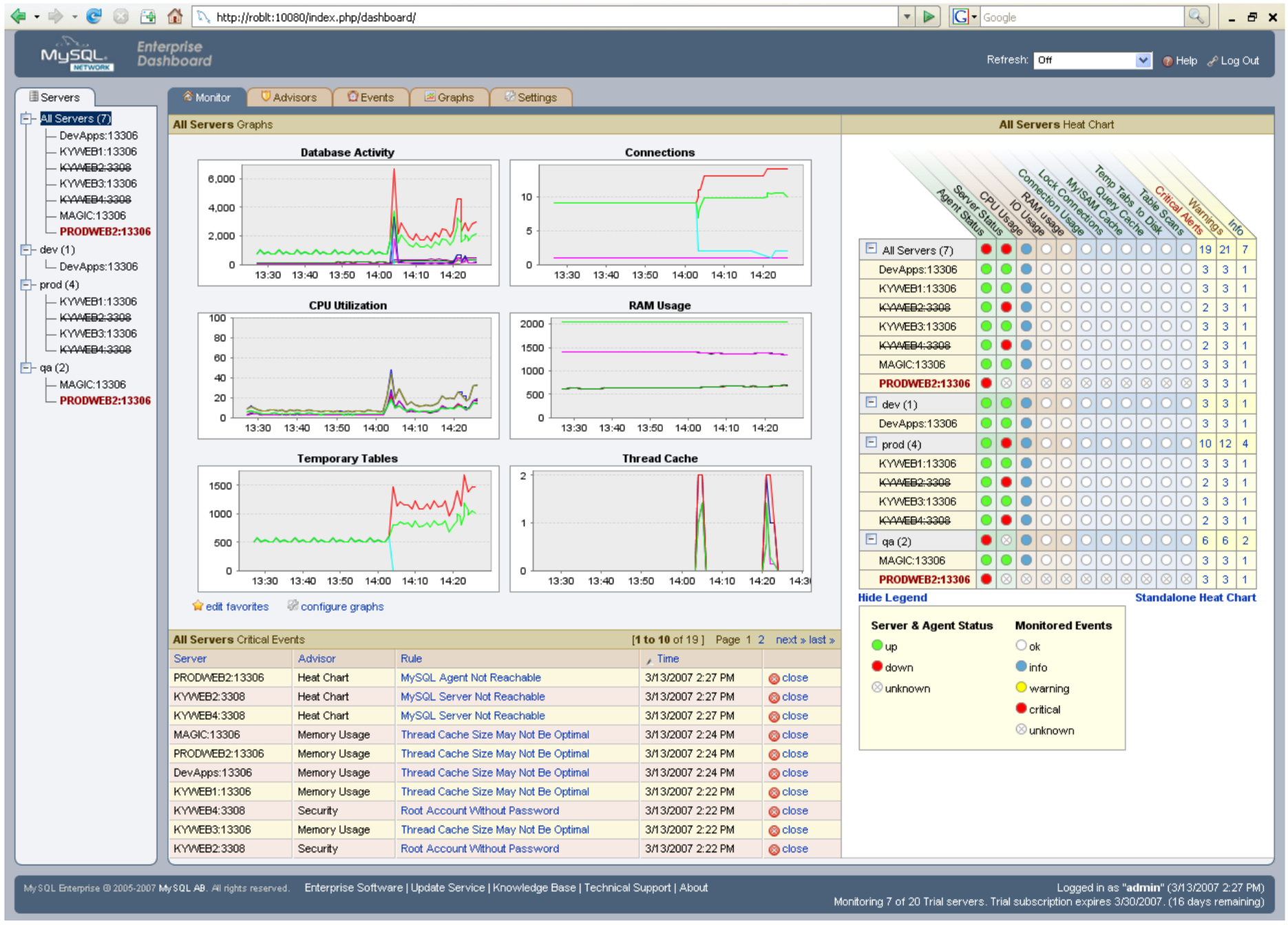
Tools: Munin, mysqlar

- Munin:
  - Even more shiny!
  - Very good OS level support
  - MySQL support integrated, but limited
    - go to http://munin.projects.linpro.no/
- mysqlar:
  - Limited metering points
  - Rules and advice in a sidebar
  - http://gert.sos.be/en/

Console: mytop, innotop

- mytop - http://jeremy.zawodny.com/mysql/mytop/
  - Permanent “SHOW FULL PROCESSLIST”
  - Killing long running queries, Point-and-click-EXPLAIN
  - qps monitor
  - Does not know InnoDB (too old)
- innotop - http://sourceforge.net/projects/innotop/
  - Permanent “SHOW STATUS LIKE 'inno%'” prettyfied
  - active development, but does not replace mytop

The tool I am no longer allowed to call “Merlin”



The tool I am no longer allowed to call “Merlin”

- C-language agents on MySQL server machines
- reporting back to data collection instance
  - Alerting
  - Advisories link to Knowledge Base and support
- Scaleable server and GUI
  - Logging into local MySQL
  - Server groups, “Heat chart”
- Part of Enterprise: http://mysql.com/products/enterprise/