MySQL

New Features 5.6

FOSDEM MySQL and Friends Devroom February 5, 2012, ULB Brussels

Oli Sennhauser

Senior MySQL Consultant, FromDual



oli.sennhauser@fromdual.com

FromDual

- FromDual provides neutral and independent:
 - Consulting
 - Remote-DBA
 - Support for MySQL, Galera, Percona Server
 - Training
- Oracle Silver Partner (OPN)



More information about us: http://www.fromdual.com

Contents

- Milestone Release Model
- Partitioning
- InnoDB
- Optimizer
- Performance Schema (P_S)
- Replication
- Various

Milestone Release Model

- Between 5.4 and 5.5 MySQL introduced the new "Milestone Release Model"
- Dynamic Model for development in theory:
 - Starts and always is at least in beta quality
 - Milestone releases, with RC quality, every 3 6 months
 - Between Milestones new features allowed
 - GA releases every 12 18 months (5.5: Oct. 2010)
 - No more than 2 releases in active support!
- MySQL Lab releases

Milestone Release Model

- Look at the schedule:
 - 5.6.0+1 Milestone 4+5, not released
 - 5.6.2 no Milestone number, released April 2011
 - 5.6.3 + 5.6.4 (M6 + M7) both October 2011
 - 5.6.5 (M8) not released yet (February 2012)
- In practice:
 - → I cannot see much practical differences
- My guess: 5.6 GA at Collaborate in April 2012
 - → As a consequence: 5.0 and 5.1 EOL!
 - Who is still at 5.0 and 5.1?

NF 5.6 / Partitioning

Explicit Partition Selection

```
SELECT *
FROM sales PARTITION (p2010, p2011)
WHERE sum < 100;
```

- → Caution: implicit WHERE clause!
- Exchanging Partitions

```
ALTER TABLE sales_hist
EXCHANGE PARTITION p2011
WITH TABLE sales;
```

ETL, DWH

NF 5.6 / InnoDB

InnoDB INFORMATION SCHEMA tables

- INNODB_SYS_TABLESTATS ▼

 ◆ TABLE_ID BIGINT(21)

 ◆ NAME VARCHAR(193)

 ◆ STATS_INITIALIZED VARCHAR(193) ■ INNODB_BUFFER_PAGE ▼ INNODB_SYS_TABLES V POOL_ID BIGINT(21)
 POOL_SIZE BIGINT(21) INNODB BUFFER FOR NAME VARCHAR(193) SPACE BIGINT(21) • FREE_BUFFERS BIGINT(21 NUM ROWS BIGINT(21) NAME VARCHARISSS REE NAME VARCHARITS PAGE NUMBER BIGINT(2) DATABASE PAGES BIGINT(21) CHIST INDEX SIZE BIGINTI21 ON COLE BITOLD O PAGE TYPE VARCHARISA OUD DATABASE PAGES BIGIATION OTHER INDEX SIZE BIGINT(21) O FLUSH TYPE BIGINTIZE ■ INNODB_SYS_INDEXES ▼ PENDING_READS BIGINT(21) PENDING FLUSH LRU BIGINT(2) OLDEST MODIFICATION BIGINTI PENDING FLUSH LIST BIGINT(2) ACCESS TIME BIGINT(21) PAGES MADE YOUNG BIGINTIZ TABLE ID BIGINT(21) TABLE NAME VARCHA PAGES_MADE_YOUNG BIGINT(21)

 PAGES_MADE_YOUNG_RATE DOUBLE

 PAGES_MADE_NOT_YOUNG_RATE DOUBLE DATA_SIZE BIGINT(21) INNODB FT N FIELDS INT(11) - PAGE NO INTITUT O STIRSYSTEM VARIOUADINGS O NUMBER_PAGES_READ BIGINT(21) COMPRESSED_SIZE BIGINT(21) NUMBER PAGES CREATED BIGINT(21) ☐ INNODB_SYS_FOREIGN_COLS ▼ PAGE STATE VARCHARISA ANUMBER PAGES WRITTEN BIGINTIST ♦ ID VARCHAR(193)
 ♦ FOR_COL_NAME VARC IO DV VAROUADIRA DACES DEAD DATE DOUBLE AVG COUNT DOUBLE PAGES_READ_RATE DOUBLE
 PAGES_CREATE_RATE DOUBLE
 PAGES_WRITTEN_RATE DOUBLE NAME VARCHAR(193) COUNT RESET BIGINTI21 REF COL NAME VARCHAR(193) MAX COUNT RESET BIGIN → MAX_COUNT_RESET BIGINT(2: → MIN_COUNT_RESET BIGINT(2: → AVG_COUNT_RESET DOUBLE → TIME_ENABLED DATETIME → TIME_DISABLED DATETIME NUMBER PAGES GET BIGINT(21) O HIT BATE BIGINT(21) YOUNG MAKE PER THOUSAND GETS BIGINT/2 INNODB METRICS INNODB_SYS_FIELDS ▼

 INDEX_ID BIGINT(21)

 NAME VARCHAR(193) O TIME_ELAPSED BIGINT(21) O TIME RESET DATETIME O STATUS VADOUAD/109 READ AHEAD RATE DOUBLE POS INT(11) READ AHEAD EVICTED RATE DOUBLE PAGE_NUMBER BIGINT(21) O LEU IO TOTAL BIGINT(21) PAGE_TYPE VARCHAR(64 □ LRU_IO_CURRENT BIGINT(21)
 □ UNCOMPRESS_TOTAL BIGINT(21)
 □ UNCOMPRESS_CURRENT BIGINT(21) FLUSH TYPE BIGINT(2) IS_HASHED VARCHAR(S)
 NEWEST_MODIFICATION BIGINT(21)
 OLDEST_MODIFICATION BIGINT(21) ■ INNODB_CMP ▼ ■ INNODB_CMP_RESET ▼ INNODB SYS trx_id VARCHAR(18) page_size INT(5)
 compress_ops INT(11)
 compress_ops INT(11) ACCESS TIME BIGINT(21) trx_started DATETIME TABLE NAME VARCHABITORAL trx requested lock id VARCHAR(8) INDEX NAME VARCHARI102 P lock_id VARCHAR(81) @ lock_trx_id VARCHAR(18 DATA_SIZE BIGINT(21)
 COMPRESSED_SIZE BIGINT(21) lock_mode VARCHAR(32 lock type VARCHAR(32) COMPRESSED VARCHAR(3) O IO FIX VAROHAR(64) o trx tables in use BIGINT(21) O IS OLD VARCHARIS o trx_tables_locked BIGINT(21) trx lock structs BIGINT(21) lock rec BIGINT(21) pages_free INT(11) lock_data VARCHAR(8 SELECT name, subsystem, count, comment o trx isolation level VARCHAR(16) trx_unique_checks INT(1)
 trx_foreign_key_checks INT(1)
 trx_last_foreign_key_error VARCHAR(2)
 trx_adaptive_hash_latched INT(1)
 trx_adaptive_hash_latched INT(1) FROM INFORMATION SCHEMA.innodb metrics WHER subsystem count trx rseq history len transaction Length of the TRX RSEG HISTORY list Current rollback segment size in pages trx rseg curent size
- Most of InnoDB Monitor is now obsolete!

NF 5.6 / InnoDB performance

- Page cleaner thread (before master thread)
- innodb purge threads can be set > 1
- Kernel mutex split → improved concurrency
- Concurrent read while creating secondary index
- Improved warm-up:
 - innodb_buffer_pool_dump_at_shutdown
- InnoDB REDO log size up to 512 Gbyte
- InnoDB threads scheduling better > 16 threads
- UNDO log → separate TS (random I/O → SSD!)
- Improved concurrency extending TS files innodb_file_per_table

NF 5.6 / InnoDB

- InnoDB and Optimizer:
 - Persistent Optimizer Statistics
 - Control of Statistics sampling (random dives)

- Deadlocks to the error log
 - innodb_print_all_deadlocks
- InnoDB page size can be: 16k, 8k, 4k
 - innodb_page_size

NF 5.6 / Optimizer

- ORDER BY on non indexed columns
 - → sort buffer avoid sort merge passes
- Multi-Range Read (MRR)
 - → optimize Range Scan on secondary indexes
- Index Condition Pushdown (ICP)
 - → WHERE is evaluated in the Storage Engine
- Query Execution Plan for DML Statements!
- Optimization of derived tables (FROM clause)
 - → Postponed materialization and index on derived table
- Batched Key Access (BKA) → improved JOIN performance
- Optimizer trace!

NF 5.6 / Performance Schema

- MySQL introduced with 5.5 the P_S
 - The idea measure everything!
 - Has some performance impact!
- New instrumentation for:
 - Table read and write (row-level accesses)
 - Stages and statements (stage = state)
 - Connections, Sockets
 - Table lock wait events
 - Table and index I/O wait events
- Filter by
 - thread
 - object

Sergei Petrunia at 15:00

Replication features of 2011

Globally Unique Server ID

```
cat $datadir/auto.cnf
[auto]
server-uuid=db731167-2b4c-11e1-928c-bcaec586ca65
```

- Delayed Replication
 - Before: mk-slave-delay (Maatkit)

```
CHANGE MASTER TO MASTER_DELAY = 42;
```

• Timestamp added to SHOW SLAVE STATUS:

```
SHOW SLAVE STATUS\G
...

Last_IO_Error_Timestamp: 120130 16:59:12
Last_SQL_Error_Timestamp:
```

- Row Image Control
 - RBR
 - Save: disk space, network resources, memory

```
binlog_row_image = {full | minimal | noblob}
```

- Crash safe Binary Logs
 - Event length + CRC32 checksum
 - Reading and writing on Master and Slave
 - Complete Events/Trx from/to binary log

- Slave Log Tables
 - master.info and relay-log.info into tables:
 - mysql.slave_*_info
 - MyISAM :(→ convert to InnoDB?

```
master-info-repository = TABLE
relay-log-info-repository = TABLE
```

Parallel Event Execution (multi-threaded slave)

Per schema

```
slave_parallel_workers = <n>
```

NF 5.6 / Various

- Fractional seconds
 - Up to microseconds (0.000001 s)
- GET DIAGNOSTICS
 - For Stored Programs
- Fulltext indexes on InnoDB tables!
- Pluggable authentication (Socket)
- Memcached Plug-in (still in Labs :()
- mysqlbinlog --read-from-remote-server --raw

Summary

- Bugs fixed: +400 bugs
- Clean-up!
- And many many more smaller features...
- Incompatible changes!
 - → Upgrade

Q & A

Questions?

Discussion?

We have some time for face-to-face talks...

