Maria Engine

MySQL conference, 2008-04-16

Michael Widenius

MySQL Fellow

Project Goals

1. To create an ACID and multi-version concurrency Control (MVCC) transactional storage engine that can function as the default non-transactional and the default transactional storage engine for MySQL.

2. To be a MyISAM replacement. This is possible because Maria can also be run in non-transactional mode and Maria supports same row formats and features as MyISAM

3. To create a Storage Engine for MySQL which is good for data warehousing (DW) purposes

Why Falcon AND Maria?

- Maria and Falcon have architecture differences (feature/capability/performance -wise)
  - With Falcon+Maria we should be able to cover more scenarios and requirements that are important for MySQL users
- Maria to also target Data Warehousing - a growing market where MyISAM (the "core" of Maria) already has proven some success
- Maria can be used standalone outside of MySQL

- => Differences between the two engines are Good = Users have options to choose from based on their needs
  - Exact target market for engines will be more clear as the engine starts to be more deployed

Project Team and allocation

- Patrik Backman   Project Manager
- Michael "Monty" Widenius   Technical Lead
- Sanja Byelkin   Developer
- Sergei Golubchik   Developer
- Guilhem Bichot   Developer
- Jani Tolonen   QA

We are going to to add 2-3 full time developers/QA persons to the team shortly

Project Plan (as seen by Maria team)

- ## MySQL 5.1 + Maria release
  - ### Maria 1.0,"Crash Safe", released in January 2008
  - ### Maria 1.5, "Concurrent insert/select", April-May 2008
    - Maria 1.5 to be merged as part of formal MySQL 6.0 release
- ## MySQL 6.0 + Maria Release
  - "Maria 2.0 Transactional and ACID" alpha, by Q4/08
  - "Maria 3.0 High Concurrency & Online Backup" alpha, by Q1/09
  - **"Maria 3.0 High Concurrency & Online Backup" GA, by Q2/09**
- ## MySQL 6.1 + Maria release
  - "Maria 4.0 Data Warehousing" alpha, by Q3/2009
    - Bitmap & XDB indexes

Note: These are Maria releases, not official MySQL server releases!

In-depth look into Maria

- What to expect of Maria 1.0 / 1.5
- Technical goals
- How Maria 1.5 compares to MyISAM
- Resources for understanding more about Maria

Technical goals of Maria 1/2

- Multi-version concurrency Control (MVCC) and ACID
- MyISAM replacement (non transactional table support)
- Used for internal temporary tables in MySQL
- All indexes should have equal speed (clustered index is not on current road map).
- Allow 'any' length transactions to work (Having long running transactions will just cause more logs space to be used)
- Allow log shipping; that is, you can do incremental backups of Maria tables just by copying the Maria logs.
- Can be used as standalone library

Technical goals of Maria 2/2

- Allow copying of Maria tables between different Maria servers (under some well-defined constraints)
- Better blob handling (than that in MyISAM at least):
  - No memory copying or extra memory used for blobs on insert/update.
  - Blobs allocated in big sequential blocks - Less fragmentation over time
  - Blobs are stored so that Maria can easily be extended to have access to any part of a blob with a single fetch in the future.
- Instant SELECT COUNT(*) and table checksums
- Efficient storage on disk (that is, low row data overhead, low page data overhead, and little lost space on pages).
- Small footprint, to make MySQL + Maria suitable for desktop and embedded applications.

Maria 1.5 - Comparison to MyISAM 1/2

- Maria 1.0 is basically a crash safe, non transactional version of MyISAM with a new cacheable row format (Released)
- Maria 1.5 is a crash safe, non transactional version of MyISAM with higher insert/select concurrency (Coming soon).

Note: The primary goal is to make Maria of good quality
- Performance is not a primary goal for Maria 1.0-2.0, - to be addressed in Maria 3.0 after we know the basic works

Maria 1.5 - Comparison to MyISAM 2/2

- Maria supports all aspects of MyISAM, except as noted on following slides. This includes
  - External and internal check/repair/compressing of rows
  - Different row formats (fixed size, dynamic, compressed + page)
  - Fast count(*) and checksum table
  - Different index compress formats
  - maria_check etc.
  - After a normal shutdown (or flush tables + maria_chk –zerofill) one can copy Maria files between servers.

Advantages of Maria, Compared to MyISAM 1/2

- Data, index and statistics are crash safe
  - On crash, things will rollback to state of the start of statement or last LOCK TABLES commands.
- Maria can replay everything from the log.
  - Including create/drop/rename/truncate tables
  - One can thus make a backup of Maria by just copying the log.
- LOAD INDEX can skip index blocks for not wanted indexes
- Maria has unit tests of most parts

Advantages of Maria, Compared to MyISAM 2/2

- Supports both crash safe (soon to be transactional) and not transactional tables.
  - Not transactional tables are not logged and rows uses less space (identical space & performance as MyISAM tables)
  - CREATE TABLE foo (...) TRANSACTIONAL=0|1
- Supports all MyISAM row formats + PAGE format where data is stored in pages. PAGE format is:
  - The only crashsafe/transactional row format for Maria
  - Versioned, which allows you to have any number of concurrent insert & selects on the same tables (Maria 1.5)
  - Cached by page cache and should give a notable speed improvement on systems which bad data caching

Differences between Maria and MyISAM

- Maria uses BIG (1G by default) log files

- Maria has a log control file (maria_log_control) and log files (maria_log.???????).
  - The log files can be automatically purged when not needed or purged on demand (after backup)

- Maria uses by default 8K pages (MyISAM 1K).
  - Maria should be faster on static size indexes but slower on variable length keys until we add a directory to index pages (Maria 2.0)

Disadvantages of Maria compared to MyISAM, that *will be fixed soon*

- Repair with many threads (maria_chk –parallel-recover) not yet implemented for Maria's PAGE format

- Maria doesn't support INSERT DELAYED (yet)

- Fulltext and R-tree (geographical) indexes are not yet crash safe

Differences to MyISAM that are not likely to be fixed

- No external locking
  - MyISAM has external locking, but is not much used
- Maria has one page size for both index and data (defined when Maria is used first time)
  - MyISAM supports different page sizes per index
- Maria has only one page cache
- Index number + checksums requires 5 extra byte per page
- Maria doesn't support RAID (disabled in MyISAM too)
- Minimum data file size for BLOCK format is 16K (with 8K pages)
- Storage of very small rows (< 25 bytes) are not efficient for PAGE format.
  - Maria data pages in block format has an overhead of 10 byte/page and 5 byte/rows. Transaction and multiple concurrent writer support will use an extra overhead of 7 bytes for new rows, 14 bytes for deleted rows and 0 bytes for old compacted rows.

Todo before release of 1.5 (April/May)

- Make indexes & statistics versioned (Data is already versioned)
- Make fulltext (boolean mode) and R-tree index crashsafe
- Parallel-recovery (rolling feature)

XDB indexes (for Maria 4.0)
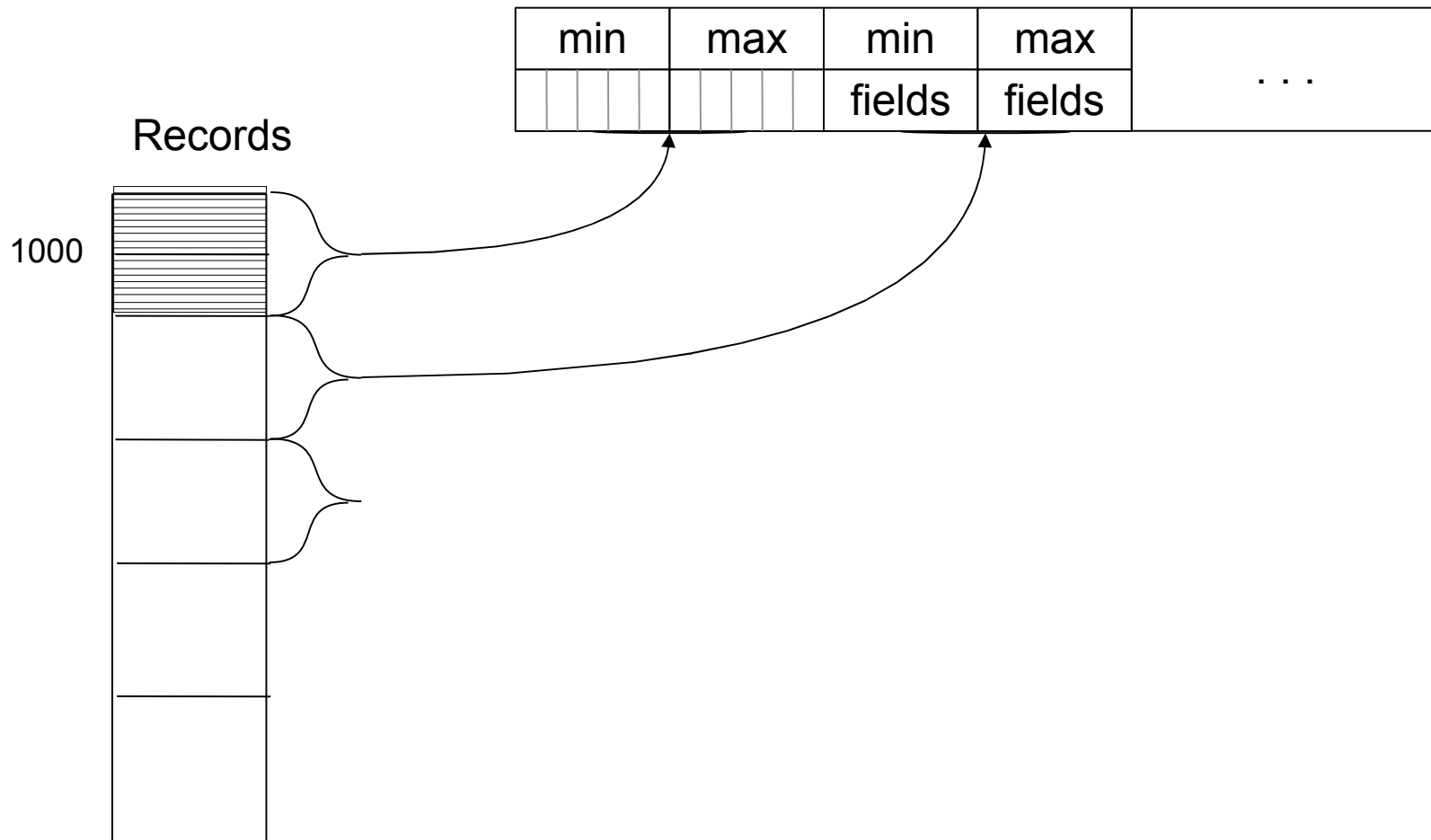
Maria - resources

More information/Questions/Feedback:

- Maria roadmap, algorithms and internals are described in detail at http://forge.mysql.com/worklog/task.php?id=3871

- End user documentation about Maria can be found at http://dev.mysql.com/doc/refman/5.1/en/storage-engines.html

- Monty's blog at: http://monty-says.blogspot.com/ will have all important updates to the Maria project.

- You can follow and participate in all Maria development discussions: maria@lists.mysql.com or at the Maria forum at: http://forums.mysql.com, section Maria

- You can report bugs and check bugs in Maria in the MySQL bugs system at http://bugs.mysql.com/

Maria - Live Demo!

=> Welcome to BOF, at 19.00 (7.00 PM for those who don't understand international time format)