MariaDB: MySQL for the community

Kristian Nielsen

`knielsen@askmonty.org`

MariaDB Developer
Monty Program AB

Open Source Days 2010

Outline

Outline

Some history

Some of Michael "Monty" Widenius' creations. . .

- 1994: MySQL
- 2003: MaxDB (with SAP)
- 2009: MariaDB

Some of Michael "Monty" Widenius' creations. . .

- 1994: MySQL
- 2003: MaxDB (with SAP)
- 2009: MariaDB

- My
- Max
- Maria

Some history

Some of Michael "Monty" Widenius' creations...

- 1994: MySQL
- 2003: MaxDB (with SAP)
- 2009: MariaDB

- My
- Max
- Maria

"MySQL" is a trademark of Sun (now Oracle)

Apart from the name, MariaDB is best thought of as another version of MySQL

History and Background

- Sun buys MySQL in 2008
    - Around that time some core people start leaving
- MySQL development is closed
    - Poor opportunities for outsiders to participate
- Monty starts MariaDB February 2009
    - Community branch of MySQL
    - Open development for people outside of MySQL/Sun
- Monty Program AB currently employs around 7 core devs
    - Similar number of QA/web/sysadm/etc. people
    - Also people outside of Monty Program participate
- "Save the people, save the project"

MariaDB overview

- A branch of MySQL
  - Features
  - Bug fixes
  - Continually updated with latest MySQL development
- 100% compatible, plug-in replacement
- Stable release (MariaDB 5.1.42)
  - Packages: Debian, Ubuntu, Centos (as well as tarball and source)
- Infrastructure and processes for open development
- "MySQL 5.x ($x > 1$) as we would have done it"

History and Background

Monty Program AB

- Contributes to MariaDB development
- Offers NRE (features/bugfixes) for MariaDB and MySQL
    - `http://askmonty.org/wiki/index.php/Commercial_Offerings`
- Partnerships

Open Database Alliance (ODBA)

- Non-profit organization
- Members work together to promote Open Source database technology
    - Including MariaDB, MySQL, PostgreSQL, ...

Outline

MariaDB extra features

- XtraDB storage engine
- PBXT storage engine
- FederatedX storage engine.
- Slow query log extended statistics
- Microsecond precision in processlist
- Table elimination optimisation
- Maria storage engine
- Thread pool support
- utf8_croatian_ci, ucs2_croatian_ci collations
- Bug fixes

XtraDB storage engine

- Patched version of the InnoDB plugin
- By Percona, Inc (the people behind
  http://mysqlperformanceblog.com)
- It provides
  - Multi-core scalability improvements
  - I/O scalability improvements
  - More diagnostic information
  - Ability to save/pre-load buffer pool contents to reduce
    warm-up period at server restart
  - Fixes for index statistics collection
  - etc, etc...

XtraDB CPU and I/O scalability

Benchmark: tpcc-mysql, about 90GB of data.
4×4 cores, 32GB RAM, approximately 1000 MB/sec I/O
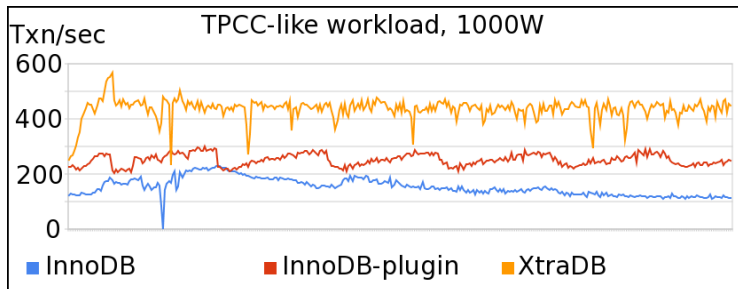
- Buffer pool mutex split
- Multiple I/O threads
- Better adaptive checkpointing

PBXT storage engine

- Developed by PrimeBase Technologies
  (`http://www.primebase.org`)
- Transactional
- ACID-compliant
- Multi-version concurrency control (MVCC)
- Use-cases similar to InnoDB
    - But using radically different internal algorithms
- Promising benchmarks
- Interesting with new approaches to utilise new types of hardware
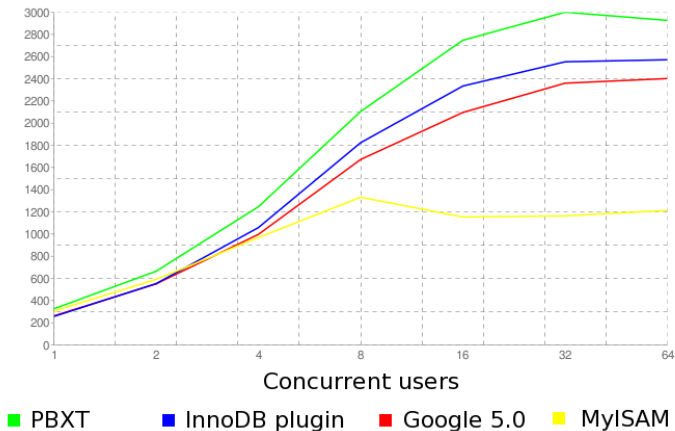- Synergy with the MySQL ecosystem

PBXT storage engine

Mark Callaghan benchmarked PBXT

`http://mysqlha.blogspot.com/2009/03/pbxt-is-fast-no-kidding.html`

Throughput

Concurrent users

■ PBXT   ■ InnoDB plugin   ■ Google 5.0   ■ MyISAM

FederatedX storage engine

- By Patrick Galbraith
- Replacement for MySQL FEDERATED storage engine
    - This appears to be no longer maintained by MySQL
- Bug fixes
- Future plans
    - Other databases (ODBC . . .)
    - Condition pushdown support

FederatedX storage engine

```
CREATE TABLE t1 (continent, country);
CREATE TABLE rt2 (country VARCHAR(100),
                  city VARCHAR(100)) ENGINE=federated
     CONNECTION="mysql://user:pass@host/db/t2";
SELECT t1.continent, rt2.city
  FROM t1 JOIN rt2 ON (t1.country = rt2.country);

    +-----------+----------------+
    | continent | city           |
    +-----------+----------------+
    | africa    | ouagadougou    |
    | africa    | bobo-dioulasso |
    | africa    | accra          |
    | asia      | beijing        |
    | asia      | shanghai       |
    +-----------+----------------+

DELETE t1, rt2
  FROM t1 JOIN rt2 ON (t1.country = rt2.country)
 WHERE t1.continent = "asia";
```

Extended statistics in slow query log

MySQL

```
# User@Host: root[root] @ localhost []
# Query_time: 3.480293  Lock_time: 0.000754 ...
use test;
SET timestamp=...;
select count(*) from one_k A,one_k B,ten C where...
```

MariaDB

```
# User@Host: root[root] @ localhost []
# Thread_id: 1  Schema: test  QC_hit: No
# Query_time: 4.605642  Lock_time: 0.000964  ...
# Full_scan: Yes  Full_join: Yes  Tmp_table: No ...
# Filesort: No  Filesort_on_disk: No  Merge_passes: 0
SET timestamp=...;
select count(*) from one_k A,one_k B,ten C where...
```

Extended statistics in slow query log

Configure slow log in `my.cnf`

```
slow_query_log=/path/to/slow.log
log_slow_verbosity=Query_plan
log_slow_filter=name,name,...
log_slow_rate_limit=n
```

'name's:

- `admin`
- `filesort`, `filesort_on_disk`,
- `full_join`,
- `full_scan`
- `query_cache`, `query_cache_miss`,
- `tmp_table` `tmp_table_on_disk`

Microsecond precision in processlist

- Based on `microsec_process.patch` by Percona
- Displays milliseconds with fractions in processlist
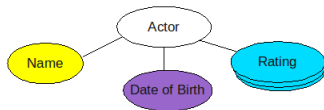   - Useful for analyzing load of small queries

MySQL

```
MySQL > select * from information_schema.processlist;
+----+...+---------+------+--------------+-------...-+
| ID |   | COMMAND | TIME | STATE        | INFO      |
+----+...+---------+------+--------------+-------...-+
| 2  |   | Query   |    2 | Sending data | select    |
```

MariaDB

```
MariaDB > select * from information_schema.processlist;
+----+...+---------+------+-----------+-------...-+---------+
| ID |   | COMMAND | TIME | STATE     | INFO      | TIME_MS |
+----+...+---------+------+-----------+-------...-+---------+
| 2  |   | Query   |    0 | executing | select    |   1.363 |
```

Table elimination


```
create view actors as select * from
select
    ac_anchor.AC_ID, AC_Name , AC_birthdate , AC_rating
from
    ac_anchor
    left join ac_name on ac_anchor.AC_ID=ac_name.AC_ID
    left join ac_dob on ac_anchor.AC_ID=ac_dob.AC_ID
    left join ac_rating on (ac_anchor.AC_ID=ac_rating.AC_ID and
                            ac_rating.AC_fromdate =
                              (select max(sub.AC_fromdate)
                               from ac_rating sub
                               where sub.AC_ID=ac_rating.AC_ID))

select AC_rating from actors where AC_name='Gary Oldman'
```

Other MariaDB features

- Maria storage engine
  - Currently: MyISAM with buffer pool and crash recovery.
  - Roadmap: Transactional, MVCC
  - Currently on lower priority, as MySQL and InnoDB are now both owned by Oracle
- Thread pool
- utf8_croatian_ci, ucs2_croatian_ci collations
- Bug fixes
  - Especially related to test suite failures and community discussions

Compatibility with MySQL

Client libraries	Yes
Client-server protocol	Yes
Command line tool names, syntax, etc	Yes[*]
SQL dialect	Yes
Replication master-slave	Yes (if both have the used features)
Data files (start one server on datadir from another)	MySQL->MariaDB: Yes Backwards: as long as both servers support used features[†]

---

[*]MariaDB by default use the same port/socket/binary names as MySQL and you can't install both side-by-side. This is similar to using two different versions of MySQL simultaneously.

[†]e.g. can't just go back if you used PBXT or utf8_croatian_ci.

Roadmap: MariaDB 5.2

- Already pushed:
    - Virtual columns (based on contribution by Andrey Zhakov)
    - Pluggable authentication
    - userstatsv2 patch (Percona)
    - mysqlbinlog —rewrite-database (with support for RBR)
- Expected updates:
    - Upstream components: xtradb, pbxt, federatedx.
    - Partitioned MyISAM key cache
    - Better observability for Row-Based Replication

Roadmap: MariaDB 5.3

- Depends on what will be funded or contributed
- Features we're working on right now:
    - Batched Key Access
        - Backport from MySQL 6.0, fixing known bugs
    - Subquery optimisations
        - Backport from MySQL 6.0
        - Backport from MySQL 6.x
        - Additional optimiser improvements

Outline

Community development

- Much of the MariaDB featureset is actually "available" already...
- But not *really* available:
  - Not included in official MySQL (not even a beta)
  - Not available from distribution repositories
  - Not available in a single source package
- MariaDB makes the existing community development more *available*

But MariaDB is not just a distribution of existing development...

Open development model

We now have the infrastructure to work on the MySQL code:

- Web page, mailing list, bzr repository, package mirrors, . . .
- Packaging/releases
- Merging with upstream MySQL
- Continuous integration tests (using Buildbot)
- Proven processes for integrating code
    - And (hopefully) for inclusion in a timely release
- Forums (IRC, mailing lists) for discussions, code reviews
    - And developers with knowledge of server internals
- So given sufficient skill, everyone can contribute on an equal footing
    - Taken for granted in eg. Linux kernel development
    - Not true for MySQL development, despite good intentions

Open development model: examples

- Can work with storage engine developers (XtraDB, PBXT)
    - Extend storage engine API
    - Extend engines to implement the new API
- Can work with distro packagers
    - Integrate changes needed by distributions
    - Integrate package build in Buildbot, to save packagers from having to discover and fix future breakages
- "Scratching an itch"
    - Going beyond new storage engines

To me, this is the primary motivation for MariaDB

- But still need to prove it works in practice

How to get involved

- Website
    - `http://askmonty.org/`
- IRC channel #maria on FreeNode
- Mailing list
    - `maria-developers@lists.launchpad.net`
- Bug reports
    - `https://bugs.launchpad.net/maria`
- Documentation
    - The MySQL manual is not free!
    - Knowledge base project starting, for wiki-like documentation
- Patches welcome!

Licencing

- MySQL and MariaDB are GPL (v2)
- MySQL additionally dual licenced
  - Sun Contributor Agreement (SCA) signs over shared copyright
- MariaDB wants to be able to co-operate with MySQL
  - MCA (SCA variant)
  - BSD (3-clause)
  - Separate modules (plugin/storage engine) can be GPL-only
- Idea is to be the community version of MySQL (similar to Fedora/RHEL)
- Waiting to hear Oracle's plans

Why use MariaDB?

- Want to use or test some of the new features or storage engines (or bug fixes)
  - Benefit from the integration
- Want to participate
  - Development
  - Bug-fixing, platform-tuning
  - Testing
  - Documentation
  - Infrastructure
  - . . .
- Want to buy or fund NRE (available for MySQL also)
- Want to support an open development model for the MySQL codebase

Why *not* use MariaDB?

- Wait and see
- Happy with MySQL 5.1 (or MySQL 5.0, or 4.1, or . . .)
- More confidence in the MySQL team at Oracle

Outline

Conclusion

- There is now a well-established community-developed branch of MySQL: MariaDB
- MariaDB is available now to test or use
- Will be interesting to see how (and if?) Oracle wants to cooperate

Slides:

```
http://knielsen-hq.org/maria/osd2010.pdf
```

Contact:

```
knielsen@askmonty.org
```