MyRocks in MariaDB

Sergei Petrunia, MariaDB

Santa Clara, California | April 24th – 27th, 2017

What is MyRocks

- RocksDB + MySQL = MyRocks
- LSM architecture
- Better compression
- Better IO efficiency
- Write optimizations
  - Bulk load
  - No-read writes
- Used and verified at Facebook

Barriers to MyRocks' adoption

- Source repository at github.com/facebook/mysql-5.6
- No releases
  - grab the current tree
- No binaries / packages
- Very much "in-house" experience
  - Special branch of MySQL
  - Special way to compile
  - Special command to run tests

What is MariaDB

- Community-oriented variant of MySQL
- Releases, binaries and packages
- Default “MySQL” in many distributions
  - RedHat / CentOS
  - Fedora
  - SuSE
  - Debian
  - ...
- Lots of platforms
  - amd64, x86, Power8, Windows, ...

MariaDB accepts contributions

- Storage engines
  - Spider
  - TokuDB
  - Mroonga
  - ...
- Galera Cluster
- On-disk data encryption
- Compressed binary log
- ...

MariaDB ♥ new technologies

- Encryption
- Parallel replication
- Window functions
- Common Table Expressions
- Group Commit with binlog
- Virtual columns
- ...

MyRocks in MariaDB

+

- LSM-tree architecture
- Compression
- Storage efficiency

- Adoption
- Packaging
- Community
- MariaDB features

Moving away from InnoDB to MyRocks?

- **No!**
- InnoDB remains the default storage engine
- InnoDB in MariaDB 10.2 has new features
  - GIS
  - Persistent AUTO_INCREMENT
  - …
- InnoDB is a proven OLTP engine
- InnoDB has features not in MyRocks
  - Galera
  - Encryption
  - …

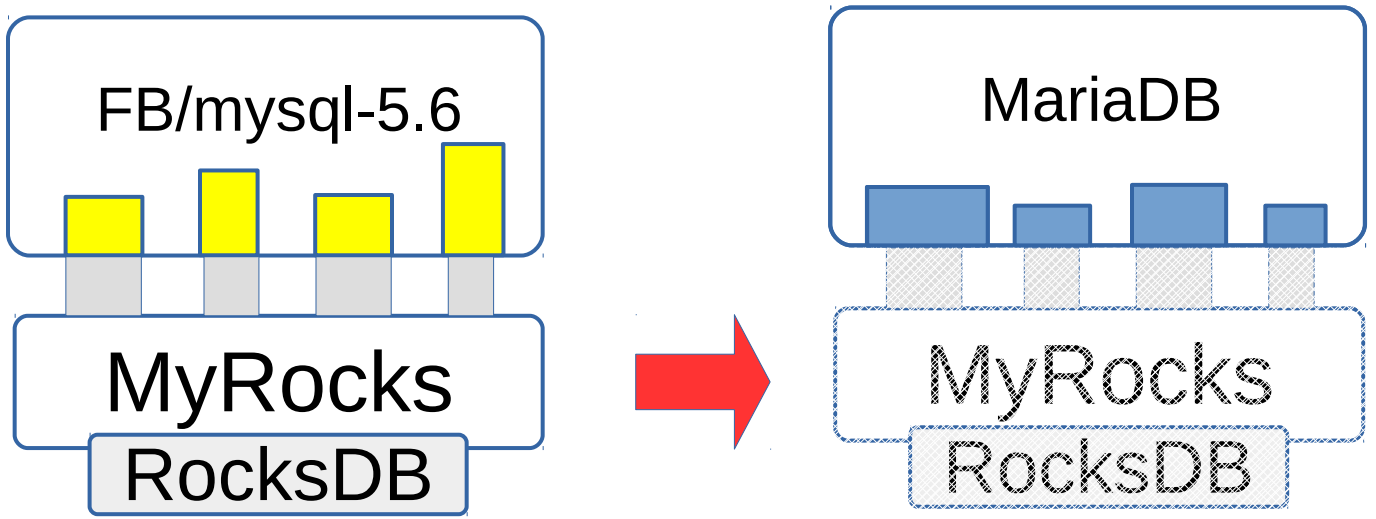
Putting MyRocks into MariaDB

Click to add text

PERCONA LIVE

Putting MyRocks into MariaDB

- MyRocks has interplay with SQL layer
- FB/5.6 [SQL layer] has extra features
- Need to un-couple from FB/5.6
- And couple with MariaDB's equivalents



Merging

- FB/MyRocks development continues
- Want to follow it with MariaDB
  - No plans to diverge
- Have a process to pull the latest MyRocks from facebook/mysql-5.6 tree
  - It’s manual work but not a lot of it
  - Similar to what we do with Galera/TokuDB/Spider

Other considerations

- Packages
  - Need to produce source/binary tarballs, debs, rpms, etc
  - Proper compression library dependencies
  - Don’t force MyRocks on all MariaDB users
- Builds
  - More platforms and architectures
- Documentation

Considerations from MariaDB side

Click to add text

Getting into a MariaDB release

- MariaDB 10.0:
  - Alpha: 12 Nov 2012
  - Stable: 31 Mar 2014
- MariaDB 10.1 (Stable)
  - Alpha: 30 Jun 2014
  - Stable: 17 Oct 2015
- **MariaDB 10.2 (RC)**
  - **Alpha: 18 Apr 2016**
  - **RC: 17 Feb 2017**
  - **Stable: soon**
- **MariaDB 10.3**
  - Alpha: 17 Apr 2017

Plugin maturity

- Plugins declare their maturity
  - unknown, experimental, alpha, beta, gamma, stable
- mysqld --plugin-maturity=level
  - will not load plugins less mature than level.
- This is how MyRocks gets into MariaDB 10.2
  - A plugin with maturity=alpha
  - Will work to increase maturity= value :-)

Current status of MyRocks in MariaDB

Click to add text

PERCONA LIVE

Current status

- MariaDB 10.2.5 RC, Apr 5, 2017
- **Includes an ALPHA version of MyRocks plugin!**
- It’s a loadable plugin (ha_rocksdb.so)
- Packages
  - Bintar, deb, rpm, win64 zip + MSI
  - deb/rpm have MyRocks .so and tools in a separate package
  - Available on recent versions of distros only
    - RocksDB requires a recent compiler.

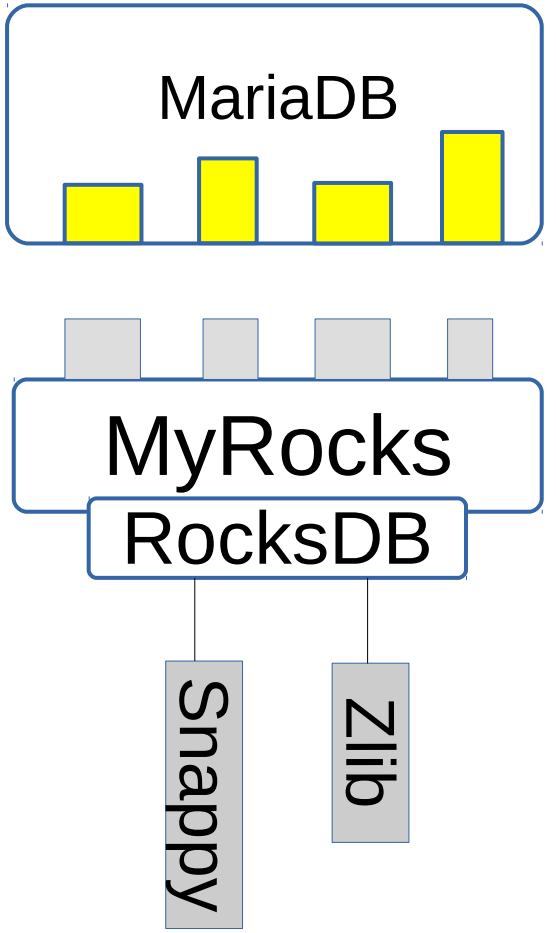
Compression libraries support

- Good compression is one RocksDB's advantages.
- RocksDB supports: Snappy, Zlib, Bzip, LZ4, LZ4HC, ZStandard

- MariaDB's Bintar package
  - Statically links with zlib and Snappy
- MariaDB's deb/rpm packages
  - Have dependencies on libz and libsnappy packages
  - Plan to support Zstandard where it is packaged
- MariaDB's Windows packages (zip + MSI)
  - Zlib (and lz4?)

Linking with RocksDB library

- Debian has a package for RocksDB
- MyRocks is tied to RocksDB@revno
  - Git submodule
- No compatibility with other versions
- Always compiling RocksDB together with MyRocks
- And statically linking

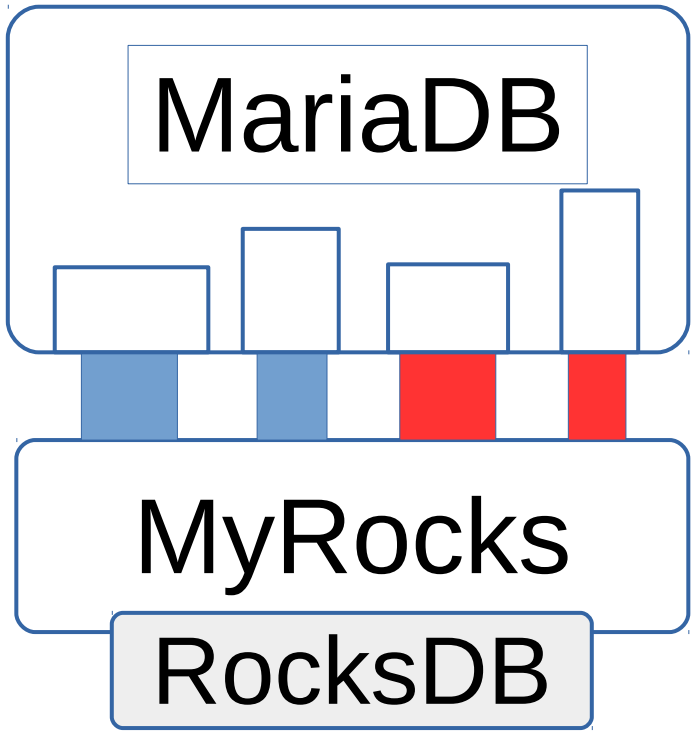


diff -u mysql/myrocks mariadb/myrocks

3K line diff for the code
- CMake build changes
- MariaDB's differences in Extended Keys
- MariaDB's differences in Index Condition Pushdown
- MariaDB calls ha_statistic_increment(...) above the storage engine
- Implement prepare_*_scan(), Bloom Filter works for ORDER BY DESC
- Storage Engine API function signature changes
  - s/MySQL/MariaDB/ in names of structs, constants, etc
- #ifdef-away code related to replication and binlog
- #ifdef-away extra diagnostics like SHOW ENGINE ROCKSDB TRANSACTION STATUS
- ...

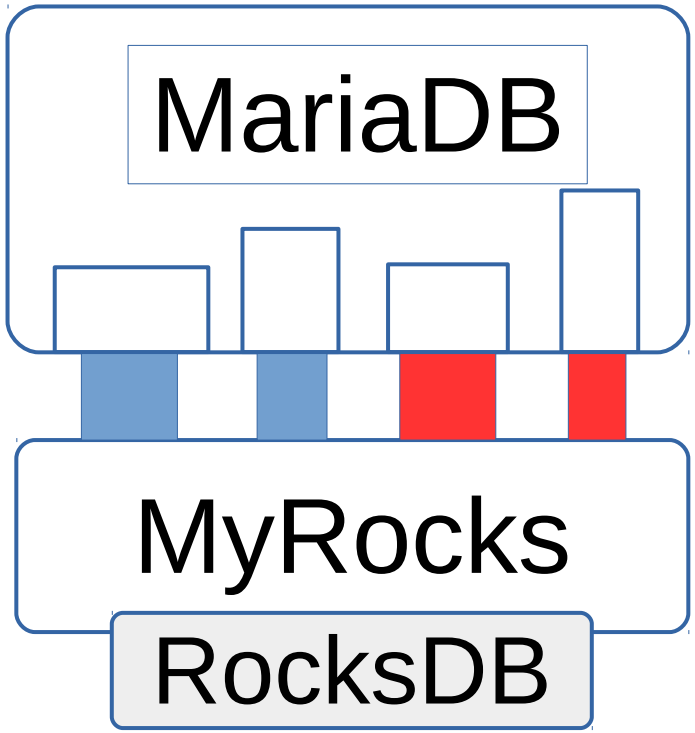
8K lines diff for tests:
- MariaDB has different default values for SQL columns
- EXPLAIN output is slightly different
- Handler_xxx counters counted a bit differently
- MTR explicitly logs connection open/switch/etc
- mysql-test/suite/$MYROCKS_TEST → storage/rocksdb/mysql-test/$MYROCKS_TEST
- ...

Is it ready for use?



- The components are stable
  - (MyRocks + RocksDB) are run in production @ Facebook
  - RocksDB is also used elsewhere
  - MyRocks not much. yet.
- Connections with MariaDB
  - Some are stable
  - Some are [nearly] missing

Is it ready for use?



- Already working
  - SQL features
  - Query optimizer features
  - Bloom filter use support
  - ...
- Not yet working
  - Work with binary log
  - Work with replication
    - [Parallel] Slave
  - Some diagnostic info

Future work

Click to add text

Further plans

- Finish non-working core features
  - Work with binary log
  - Work with replication
    - [Parallel] Slave
  - Some diagnostic info
- Package myrocks_hotbackup
  - Works but is not in the packages
- Make MyRocks work with advanced MariaDB features.

Gap Lock detector

- Differences in transaction isolation
  - InnoDB locks gaps between rows
  - MyRocks doesn't support gap locks
  - A feature to catch gap lock queries before migration
- Global, SQL level variables
  - gap_lock_raise_error, gap_lock_write_log, ...
- Getting pushback
  - “MyRocks should not put its stuff into SQL layer”
- Will try to resolve this.

MDEV-12179: Per-engine mysql.gtid_slave_pos tables

- mysql.gtid_slave_pos
  - Stores current slave position
  - Is an InnoDB table: slave position is restored on recovery
  - Will require cross-engine XA if using multiple engines
- mysql_gtid_slave_pos_${engine_name}
  - Will store slave position for each engine
  - Recovery will pick the biggest position
  - => Efficient crash-safe slave when using multiple engines.

MDEV-12179: Per-engine mysql.gtid_slave_pos tables

- A patch is available from Kristian Nielsen
- MariaDB devs are reviewing it.

Conclusions

Click to add text

Conclusions

- MyRocks is available in MariaDB 10.2 as an ALPHA-maturity plugin
- A lot of features work
  - Packages, binaries
  - The storage engine and generic SQL use
- Some features are in progress
  - Interplay between storage engine and binlog/replication
- MariaDB will work to make the plugin mature
- Will continue to merge from the upstream.

Thanks!

Rate My Session