MariaDB®

# InnoDB in 10.2 and beyond

## Some recent InnoDB changes in MariaDB

# Short history of InnoDB and Marko Mäkelä

1995 Created by Heikki Tuuri (who shortly worked at Solid)

2002 InnoDB Included in MySQL 3.23

2003 Marko Mäkelä joins as the first full-time employee

2005 Heikki Tuuri sells Innobase Oy to Oracle Corporation

2010 Oracle acquires Sun Microsystems (which had acquired MySQL)

2016 Marko Mäkelä joins MariaDB as Lead Developer InnoDB

# New InnoDB features in MariaDB 10.2

- MDEV-6076 Persistent AUTO_INCREMENT
  - AliSQL idea: Repurpose PAGE_MAX_TRX_ID on the clustered index root page
  - Update atomically in the DML mini-transaction (before transaction commit)
  - Differences from MySQL 8.0.0:
    - No redo log format change
    - No dependency added to data dictionary; preserved on export/import
- MDEV-11824 Allow ROW_FORMAT=DYNAMIC in the system tablespace
- MDEV-12289 Keep 128 persistent rollback segments for compatibility and performance
  - Do not reserve 32 persistent rollback segments for temporary tables.
  - Unlike MySQL 5.7, allow an upgrade without prior innodb_fast_shutdown=0.

# New InnoDB features in MariaDB 10.2

- MDEV-12219 Discard temporary undo logs at transaction commit
  - There is no purge or MVCC for TEMPORARY TABLE
- Cleaned up startup and shutdown (no memory leaks after failed startup)
  - Some refactoring of the undo log, purge, and transaction system code
  - Removed some race conditions around events and thread status variables
  - Added regression tests
  - MDEV-11915 Detect InnoDB system tablespace size mismatch early
  - MDEV-11947 InnoDB purge workers fail to shut down
  - MDEV-11985 Make innodb_read_only shutdown more robust
- MDEV-11520 (5.5+) Properly use posix_fallocate() for extending files
  - 10.1+ page_compression: Do not physically preallocate sparse files

# Upcoming InnoDB features in MariaDB 10.3

- MDEV-10139 Support for SEQUENCE objects
  - Implemented as a no-rollback single-index table (containing a single record)
  - No-rollback tables could find other uses, such as undo log storage (MDEV-11657)
  - (2 days of InnoDB work; many days of SQL layer work by Monty)
- MDEV-11369 Instant ADD COLUMN
  - (will start implementation soon)
  - After 10.3, MDEV-11424 would allow any ALTER TABLE that does not involve data conversions that could fail, or building any indexes
  - Will be compatible with old data files

# InnoDB Crash Recovery Changes

- MDEV-11027 (10.0+) Better progress reporting for InnoDB crash recovery
- MDEV-11556 (10.1+) InnoDB redo log apply fails to adjust data file sizes
  - Redo log scan tracks FSP_SIZE changes; files are extended when first opened
  - This bug was always worked around by InnoDB Hot Backup (innobackup) and later MySQL Enterprise Backup (MEB), and also Percona XtraBackup.
    - They would silently extend the file on seemingly out-of-bounds page number.
- MDEV-11690 Remove UNIV_HOTBACKUP (used by MEB only)
- MDEV-11782 Redefine the innodb_encrypt_log format
  - Implement proper upgrade when using redo log encryption.
  - Always rebuild the redo log when disabling or enabling encryption.
- MDEV-11814 Refuse innodb_read_only startup if crash recovery is needed
- MDEV-12061 Allow innodb_log_files_in_group=1
- MDEV-12103 Reduce the time of looking for MLOG_CHECKPOINT

# InnoDB performance improvements (w/ experimental patches for MySQL 5.7.17)

- MDEV-12121 (10.2.5) Introduce build option WITH_INNODB_AHI=OFF
  - Could benefit users of innodb_adaptive_hash_index=0.
  - The InnoDB adaptive hash index can improve performance for some workloads
  - It could also hurt performance. And it incurs an overhead for DDL operations.
- MDEV-12288 (10.3?) Reset DB_TRX_ID when the history is removed, to speed up MVCC
  - Changes the undo log format!
  - Changes the B-tree format (by allowing “null pointers” in DB_TRX_ID)
  - Would pave the road for MDEV-11658 (faster IMPORT of .ibd files)
  - Needs some work to allow upgrade from earlier versions
- MDEV-11585 Hard-code the shared InnoDB temporary tablespace ID
  - MySQL 5.7 would reassign the built-in temporary tablespace ID on every startup.
  - Simpler, faster code and easier troubleshooting with a compile-time constant.

# Background: InnoDB System Columns, MVCC and Implicit Locking

- Clustered index record format:
  - (PRIMARY KEY columns, DB_TRX_ID, DB_ROLL_PTR, other columns)
  - (PRIMARY KEY columns, child page number) in internal nodes
  - If there is no PRIMARY KEY, then a hidden DB_ROW_ID will be assigned
- MDEV-12288 (10.3?) updates DB_TRX_ID to “null” on purge, to avoid unnecessarily looking up very old transactions (committed&purged long ago)
- Secondary index record format:
  - (secondary key columns, PK columns, [child page number])
  - Delete-marked record or “too recent” PAGE_MAX_TRX_ID means that lock checks and MVCC must look up the matching PRIMARY KEY record version (if it exists)
  - Worst-case, need PK lookup for every row in a SK page!
  - Lookups can be reduced by Index Condition Pushdown (MySQL 5.6/MariaDB 10)
  - A per-record DB_TRX_ID,DB_ROLL_PTR in secondary keys could help

# Future InnoDB ideas

- MDEV-11633 Make the InnoDB system tablespace optional
  - No single point of failure; no opaque cannot-shrink-without-starting-over file
  - MDEV-11634 Improve or remove the InnoDB change buffer
  - MDEV-11655 Transactional data dictionary
    - Crash-safe DDL using a low-overhead, user-friendly, standards-based serialized representation (SQL snippets)
    - Keep *.frm files as a ‘backup’ and for easy copying of files across instances
  - MDEV-11657 Store InnoDB undo logs in a persistent table
    - Enables privileged access to undo log records; could be useful in disaster recovery
  - MDEV-11658 Simpler, faster IMPORT of InnoDB tables
  - MDEV-11659 Move the InnoDB doublewrite buffer to flat files
- MDEV-11424 Instant ALTER TABLE of failure-free record format changes
- Hopefully some of this will be in the successor of MariaDB 10.3

# Oracle InnoDB Features Removed from 10.2

- Oracle version of encryption, page compression
  - MariaDB 10.1 already shipped something similar
- MDEV-11816 Disallow CREATE TEMPORARY TABLE…COMPRESSED
- MDEV-12050 Remove unused InnoDB Memcached hooks
  - MySQL 5.6+ ships a patched snapshot of an abandoned Memcached dev branch
  - InnoDB Memcached was never enabled in MariaDB (dead code in 10.0, 10.1)
- CREATE TABLESPACE; CREATE TABLE…TABLESPACE=…
  - Tablespaces are not proper 'native' objects (even in MySQL, no import/export)
  - MDEV-11426 Remove InnoDB INFORMATION_SCHEMA.FILES implementation
- MDEV-11487 Revert InnoDB internal temporary tables from WL#7682
  - In MariaDB, query execution uses MEMORY and MyISAM tables
  - Try SELECT COUNT(*)…GROUP BY…: InnoDB cannot update the count in-place, but would allocate COUNT(*) rows in the temporary table!

# Questions?

Thank you for your attention!
marko.makela@mariadb.com