

InnoDB Change Buffering

The change buffer has been disabled by default from MariaDB 10.5.15, MariaDB 10.6.7, MariaDB 10.7.3 and MariaDB 10.8.2 (MDEV-27734), was deprecated and ignored from MariaDB 10.9.0 (MDEV-27735), and was removed in MariaDB 11.0.0 (MDEV-29694).

Benchmarks attached to MDEV-19514 show that the change buffer sometimes reduces performance, and in the best case seem to bring a few per cent improvement to throughput. However, such improvement could come with a price: If the buffered changes are never merged (MDEV-19514, motivated by the reduction of random crashes and the removal of an innodb_force_recovery option that can inflict further corruption), then the InnoDB system tablespace can grow out of control (MDEV-21952).

INSERT, UPDATE and DELETE statements can be particularly heavy operations to perform, as all indexes need to be updated after each change. For this reason these changes are often buffered.

Pages are modified in the buffer pool, and not immediately on disk. After all the records that cover the changes to a data page have been written to the InnoDB redo log, the changed page may be written ("flushed") to a data file. Pages that have been modified in memory and not yet flushed are called dirty pages.

The Change Buffer is an optimization that allows some data to be modified even though the data page does not exist in the buffer pool. Instead of modifying the data in its final destination, we would insert a record into a special Change Buffer that resides in the system tablespace. When the page is read into the buffer pool for any reason, the buffered changes will be applied to it.

The Change Buffer only contains changes to secondary index leaf pages.

Before MariaDB 5.5, only inserted rows could be buffered, so this buffer was called Insert Buffer. The old name still appears in several places, for example in the output of SHOW ENGINE INNODB STATUS.

Inserts to UNIQUE secondary indexes cannot be buffered unless unique_checks=0 is used. This may sometimes allow duplicates to be inserted into the UNIQUE secondary index. Much of the time, the UNIQUE constraint would be checked because the change buffer could only be used if the index page is not located in the buffer pool.

When rows are deleted, a flag is set, thus rows are not immediately deleted. Delete-marked records may be purged after the transaction has been committed and any read views that were created before the commit have been closed. Delete-mark and purge buffering of any secondary indexes is allowed.

ROLLBACK never makes use of the change buffer; it would force a merge of any changes that were buffered during the execution of the transaction.

The Change Buffer is an optimization because:

- Some random-access page reads will be transformed into modifications of change buffer pages.
- A change buffer page can be modified several times in memory and be flushed to disk only once.
- Dirty pages are flushed together, so the number of IO operations is lower.

If the server crashes or is shut down, the Change Buffer might not be empty. The Change Buffer resides in the InnoDB system tablespace, covered by the write-ahead log, so they can be applied at server restart. A shutdown with innodb_fast_shutdown=0 will merge all buffered changes.

Starting with MariaDB 10.5, there no longer is a background task that would merge the change buffer to the secondary index pages. The changes would only be merged on demand.

The Change Buffer was removed in MariaDB 11.0 because it has been a prominent source of corruption bugs that have been extremely hard to reproduce.

The main server system variable here is innodb_change_buffering, which determines which form of change buffering, if any, to use.

The following settings are available:

- inserts
 - Only buffer insert operations
- deletes
 - Only buffer delete operations
- changes
 - Buffer both insert and delete operations
- purges
 - Buffer the actual physical deletes that occur in the background
- all
 - Buffer inserts, deletes and purges. Default setting from MariaDB 5.5 until MariaDB 10.5.14, MariaDB 10.6.6, MariaDB 10.7.2 and MariaDB 10.8.1.
- none
 - Don't buffer any operations. Default from MariaDB 10.5.15, MariaDB 10.6.7, MariaDB 10.7.3 and MariaDB 10.8.2.

Modifying the value of this variable only affects the buffering of new operations. The merging of already buffered changes is not affected.

The innodb_change_buffer_max_size system variable determines the maximum size of the change buffer, expressed as a percentage of the buffer pool.

See Also

- InnoDB Buffer Pool

Comments

No comments

Content reproduced on this site is the property of its respective owners, and this content is not reviewed in advance by MariaDB. The views, information and opinions expressed by this content do not necessarily represent those of MariaDB or any other party.

↑ InnoDB ↑

InnoDB Versions
InnoDB Limitations
InnoDB Troubleshooting
InnoDB System Variables
InnoDB Server Status Variables
AUTO_INCREMENT Handling in InnoDB
InnoDB Buffer Pool
InnoDB Change Buffering
InnoDB Doublewrite Buffer
InnoDB Tablespaces
InnoDB File Format
InnoDB Row Formats
InnoDB Strict Mode
InnoDB Redo Log
InnoDB Undo Log
InnoDB Page Flushing
InnoDB Purge
Information Schema
InnoDB Tables
InnoDB Online DDL
Binary Log Group Commit and InnoDB Flushing Performance
InnoDB Page

Products

MariaDB Platform

MariaDB SkySQL

Pricing

Download MariaDB

Services

Remote DBA

SkyDBA

Enterprise Architect

Technical Support

Migration Practice

Consulting

Training

Resources

Documentation

Developers

Blog

Support

OpenWorks

Customer Stories

Events

MariaDB Roadshow

About

Contact

Leadership

Partners

Newsroom

Investors

Careers

Trust Center

Vulnerability Reporting

Contact

Subscribe to our newsletter!

YOUR EMAIL ADDRESS

-- Please Select Country --

Add Me

