

A.16 MySQL 5.7 FAQ: InnoDB Change Buffer

A.16.1. What types of operations modify secondary indexes and result in change buffering?

A.16.2. What is the benefit of the InnoDB change buffer?

A.16.3. Does the change buffer support other types of indexes?

A.16.4. How much space does InnoDB use for the change buffer?

A.16.5. How do I determine the current size of the change buffer?

A.16.6. When does change buffer merging occur?

A.16.7. When is the change buffer flushed?

A.16.8. When should the change buffer be used?

A.16.9. When should the change buffer not be used?

A.16.10. Where can I find additional information about the change buffer?

A.16.1.	What types of operations modify secondary indexes and result in change buffering?
	INSERT, UPDATE, and DELETE operations can modify secondary indexes. If an affected index page is not in the buffer pool, the changes can be buffered in the change buffer.
A.16.2.	What is the benefit of the InnoDB change buffer?
	Buffering secondary index changes when secondary index pages are not in the buffer pool avoids expensive random access I/O operations that would be required to immediately read in affected index pages from disk. Buffered changes can be applied later, in batches, as pages are read into the buffer pool by other read operations.
A.16.3.	Does the change buffer support other types of indexes?
	No. The change buffer only supports secondary indexes. Clustered indexes, full-text indexes, and spatial indexes are not supported. Full-text indexes have their own caching mechanism.
A.16.4.	How much space does InnoDB use for the change buffer?

	<p>Prior to the introduction of the <code>innodb_change_buffer_max_size</code> configuration option in MySQL 5.6, the maximum size of the on-disk change buffer in the system tablespace was 1/3 of the InnoDB buffer pool size.</p> <p>In MySQL 5.6 and later, the <code>innodb_change_buffer_max_size</code> configuration option defines the maximum size of the change buffer as a percentage of the total buffer pool size. By default, <code>innodb_change_buffer_max_size</code> is set to 25. The maximum setting is 50.</p> <p>InnoDB does not buffer an operation if it would cause the on-disk change buffer to exceed the defined limit.</p> <p>Change buffer pages are not required to persist in the buffer pool and may be evicted by LRU operations.</p>
A.16.5.	How do I determine the current size of the change buffer?
	<p>The current size of the change buffer is reported by <code>SHOW ENGINE INNODB STATUS \G</code>, under the <code>INSERT BUFFER AND ADAPTIVE HASH INDEX</code> heading. For example:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre>----- INSERT BUFFER AND ADAPTIVE HASH INDEX ----- Ibuf: size 1, free list len 0, seg size 2, 0 merges</pre> </div> <p>Relevant data points include:</p> <ul style="list-style-type: none"> • <code>size</code>: The number of pages used within the change buffer. Change buffer size is equal to <code>seg size - (1 + free list len)</code>. The <code>1 +</code> value represents the change buffer header page. • <code>seg size</code>: The size of the change buffer, in pages. <p>For information about monitoring change buffer status, see Section 14.5.2, “Change Buffer”.</p>
A.16.6.	When does change buffer merging occur?
	<ul style="list-style-type: none"> • When a page is read into the buffer pool, buffered changes are merged upon completion of the read, before the page is made available.

	<ul style="list-style-type: none"> • Change buffer merging is performed as a background task. The <code>innodb_io_capacity</code> parameter sets an upper limit on the I/O activity performed by InnoDB background tasks such as merging data from the change buffer. • A change buffer merge is performed during crash recovery. Changes are applied from the change buffer (in the system tablespace) to leaf pages of secondary indexes as index pages are read into the buffer pool. • The change buffer is fully durable and can survive a system crash. Upon restart, change buffer merge operations resume as part of normal operations. • A full merge of the change buffer can be forced as part of a slow server shutdown using <code>--innodb-fast-shutdown=0</code>.
A.16.7.	When is the change buffer flushed?
	Updated pages are flushed by the same flushing mechanism that flushes the other pages that occupy the buffer pool.
A.16.8.	When should the change buffer be used?
	The change buffer is a feature designed to reduce random I/O to secondary indexes as indexes grow larger and no longer fit in the InnoDB buffer pool. Generally, the change buffer should be used when the entire data set does not fit into the buffer pool, when there is substantial DML activity that modifies secondary index pages, or when there are lots of secondary indexes that are regularly changed by DML activity.
A.16.9.	When should the change buffer not be used?
	You might consider disabling the change buffer if the entire data set fits within the InnoDB buffer pool, if you have relatively few secondary indexes, or if you are using solid-state storage, where random reads are about as fast as sequential reads. Before making configuration changes, it is recommended that you run tests using a representative workload to determine if disabling the change buffer provides any benefit.
A.16.10.	Where can I find additional information about the change buffer?
	See Section 14.5.2, “Change Buffer”.