

- The union is `UNION ALL`, not `UNION` or `UNION DISTINCT`.
- There is no global `ORDER BY` clause.
- The union is not the top-level query block of an `{INSERT | REPLACE} ... SELECT ...` statement.

Internal Temporary Table Storage Engine

An internal temporary table can be held in memory and processed by the `TempTable` or `MEMORY` storage engine, or stored on disk by the `InnoDB` storage engine.

Storage Engine for In-Memory Internal Temporary Tables

The `internal_tmp_mem_storage_engine` session variable defines the storage engine for in-memory internal temporary tables. Permitted values are `TempTable` (the default) and `MEMORY`.

The `TempTable` storage engine provides efficient storage for `VARCHAR` and `VARBINARY` columns, and other binary large object types as of MySQL 8.0.13.

The `temptable_max_ram` variable defines the maximum amount of RAM that can be occupied by the `TempTable` storage engine before it starts allocating space from disk in the form memory-mapped temporary files or `InnoDB` on-disk internal temporary tables. The default `temptable_max_ram` setting is 1GiB. The `temptable_use_mmap` variable (introduced in MySQL 8.0.16; deprecated in MySQL 8.0.26) controls whether the `TempTable` storage engine uses memory-mapped files or `InnoDB` on-disk internal temporary tables when the `temptable_max_ram` limit is exceeded. The default setting is `temptable_use_mmap=ON`. The `temptable_max_mmap` variable, introduced in MySQL 8.0.23, defines the maximum amount of memory the `TempTable` storage engine is permitted to allocate from memory-mapped files before it starts storing internal temporary table data to `InnoDB` on-disk internal temporary tables. A `temptable_max_mmap=0` setting disables allocation from memory-mapped files, effectively disabling their use, regardless of the `temptable_use_mmap` setting.



Note

The `temptable_use_mmap` variable is deprecated as of MySQL 8.0.26; expect support for it to be removed in a future version of MySQL. Setting `temptable_max_mmap=0` is equivalent to setting `temptable_use_mmap=OFF`.

The `temptable_max_ram` setting does not account for the thread-local memory block allocated to each thread that uses the `TempTable` storage engine. The size of the thread-local memory block depends on the size of the thread's first memory allocation request. If the request is less than 1MB, which it is in most cases, the thread-local memory block size is 1MB. If the request is greater than 1MB, the thread-local memory block is approximately the same size as the initial memory request. The thread-local memory block is held in thread-local storage until thread exit.

Use of memory-mapped temporary files by the `TempTable` storage engine as an overflow mechanism for internal temporary tables is governed by these rules:

- Temporary files are created in the directory defined by the `tmpdir` variable.
- Temporary files are deleted immediately after they are created and opened, and therefore do not remain visible in the `tmpdir` directory. The space occupied by temporary files is held by the operating system while temporary files are open. The space is reclaimed when temporary files are closed by the `TempTable` storage engine, or when the `mysqld` process is shut down.
- Data is never moved between RAM and temporary files, within RAM, or between temporary files.