Background InnoDB I/O Threads". For general I/O tuning advice, see Section 8.5.8, "Optimizing InnoDB Disk I/O".

Note

On Linux systems, running multiple MySQL servers (typically more than 12) with default settings for `innodb_read_io_threads`, `innodb_write_io_threads`, and the Linux `aio-max-nr` setting can exceed system limits. Ideally, increase the `aio-max-nr` setting; as a workaround, you might reduce the settings for one or both of the MySQL variables.

- `innodb_read_only`

Command-Line Format	`--innodb-read-only[={OFF|ON}]`
System Variable	`innodb_read_only`
Scope	Global
Dynamic	No
`SET_VAR` Hint Applies	No
Type	Boolean
Default Value	`OFF`

Starts `InnoDB` in read-only mode. For distributing database applications or data sets on read-only media. Can also be used in data warehouses to share the same data directory between multiple instances. For more information, see Section 15.8.2, "Configuring InnoDB for Read-Only Operation".

Previously, enabling the `innodb_read_only` system variable prevented creating and dropping tables only for the `InnoDB` storage engine. As of MySQL 8.0, enabling `innodb_read_only` prevents these operations for all storage engines. Table creation and drop operations for any storage engine modify data dictionary tables in the `mysql` system database, but those tables use the `InnoDB` storage engine and cannot be modified when `innodb_read_only` is enabled. The same principle applies to other table operations that require modifying data dictionary tables. Examples:

- If the `innodb_read_only` system variable is enabled, `ANALYZE TABLE` may fail because it cannot update statistics tables in the data dictionary, which use `InnoDB`. For `ANALYZE TABLE` operations that update the key distribution, failure may occur even if the operation updates the table itself (for example, if it is a `MyISAM` table). To obtain the updated distribution statistics, set `information_schema_stats_expiry=0`.
- `ALTER TABLE` *`tbl_name`* `ENGINE=`*`engine_name`* fails because it updates the storage engine designation, which is stored in the data dictionary.

In addition, other tables in the `mysql` system database use the `InnoDB` storage engine in MySQL 8.0. Making those tables read only results in restrictions on operations that modify them. Examples:

- Account-management statements such as `CREATE USER` and `GRANT` fail because the grant tables use `InnoDB`.
- The `INSTALL PLUGIN` and `UNINSTALL PLUGIN` plugin-management statements fail because the `mysql.plugin` system table uses `InnoDB`.
- The `CREATE FUNCTION` and `DROP FUNCTION` loadable function-management statements fail because the `mysql.func` system table uses `InnoDB`.