

metadata may be extracted from tablespace files if the data dictionary becomes unavailable. SDI extraction is performed using the `ibd2sdi` tool. SDI data is stored in `JSON` format.

The inclusion of SDI data in tablespace files increases tablespace file size. An SDI record requires a single index page, which is 16KB in size by default. However, SDI data is compressed when it is stored to reduce the storage footprint.

- The `InnoDB` storage engine now supports atomic DDL, which ensures that DDL operations are either fully committed or rolled back, even if the server halts during the operation. For more information, see [Section 13.1.1, “Atomic Data Definition Statement Support”](#).
- Tablespace files can be moved or restored to a new location while the server is offline using the `innodb_directories` option. For more information, see [Section 15.6.3.6, “Moving Tablespace Files While the Server is Offline”](#).
- The following redo logging optimizations were implemented:
 - User threads can now write concurrently to the log buffer without synchronizing writes.
 - User threads can now add dirty pages to the flush list in a relaxed order.
 - A dedicated log thread is now responsible for writing the log buffer to the system buffers, flushing system buffers to disk, notifying user threads about written and flushed redo, maintaining the lag required for the relaxed flush list order, and write checkpoints.
 - System variables were added for configuring the use of spin delay by user threads waiting for flushed redo:
 - `innodb_log_wait_for_flush_spin_hwm`: Defines the maximum average log flush time beyond which user threads no longer spin while waiting for flushed redo.
 - `innodb_log_spin_cpu_abs_lwm`: Defines the minimum amount of CPU usage below which user threads no longer spin while waiting for flushed redo.
 - `innodb_log_spin_cpu_pct_hwm`: Defines the maximum amount of CPU usage above which user threads no longer spin while waiting for flushed redo.
 - The `innodb_log_buffer_size` variable is now dynamic, which permits resizing of the log buffer while the server is running.

For more information, see [Section 8.5.4, “Optimizing InnoDB Redo Logging”](#).

- As of MySQL 8.0.12, undo logging is supported for small updates to large object (LOB) data, which improves performance of LOB updates that are 100 bytes in size or less. Previously, LOB updates were a minimum of one LOB page in size, which is less than optimal for updates that might only