

Fractional Seconds

ClustrixDB supports storing fractional second precision for `TIMESTAMP`, `DATETIME`, and `TIME` with up to microsecond (6 digit) precision. ClustrixDB closely matches MySQL except for the following caveats.

Caveats for DATE/TIME

As with MySQL, ClustrixDB will interpret input values supplied to `DATE` and `TIME` functions and convert them to a standard format. However, unpredictable results may occur if values are provided in other formats.

ClustrixDB matches MySQL with the exception of:

- ClustrixDB does not support casting a `TIME` to a `DATETIME` and returns `NULL`.
- ClustrixDB does not support casting a `TIME` to a `DATE` or a `DATE` to a `TIME`.
- When converting from a `DATETIME` to a `DATE`, ClustrixDB will discard the the time portion. MySQL will take fractional seconds into account and round the time part.
- Casting negative value to `DATETIME` will return a zero date. MySQL returns `NULL`.
- When the input to `LAST_DAY` is a `DATETIME`, MySQL will round to the nearest `DATE` before computing `LAST_DAY` where ClustrixDB does not. For example, `LAST_DAY('2013-01-31 23:59:59.999999')` returns '2017-01-31' on ClustrixDB and '2017-02-01' on MySQL.
- If an invalid format string is supplied to `STR_TO_DATE`, ClustrixDB will return `NULL`. MySQL ignores extra characters at the end of format string.
- The results when using the `EXTRACT` with compound units (e.g. `DAY_SECOND`, `DAY_MICROSECOND`) may exclude some of the requested units or return incorrect results.
- Output from datetime functions (e.g. `NOW()`, `INTERVAL()`, `FROM_UNIXTIME()`, `SUBTIME()`) display microsecond precision by default.
- `UNIX_TIMESTAMP()` and `FROM_UNIXTIME` do not support input with fractional seconds.
- Fractional seconds cannot be replicated safely from ClustrixDB to MariaDB.

Clustrix recommends using RBR replication when utilizing fractional seconds.