

# Data Collection

If you want to collect data from the server, you need to select **Enable Data Collection**, so SQL DM for MySQL collects and stores various MySQL and OS metrics.

## Test Server

**CONFIG**   **TAGS**   **NOTIFICATIONS**   **ADVANCED**

- System Metrics
- Data Collection**
- Replication
- Galera
- MySQL Error Log
- MySQL Query Log
- Audit Log
- Sniffer
- Deadlock
- Monitors
- Real-Time
- Connection Settings

**Enable Data Collection**  
Monyog will collect various MySQL & OS counters

**COLLECTION INTERVAL**

5 Minute(s) ▾

**DATA RETENTION TIMEFRAME**

5 Day(s) ▾

Data collected before this timeframe is purged automatically.

**BASE TIME**

Tue Apr 03, 2018 00:00 **Reset**

Specify time from which you want to view data.

**APPLY THE SETTING TO**

Only this server ▾

**SAVE**

## Define the collection interval for every server as you want

You can also define the time interval between two successive retrievals of data. For production systems a setting between 2 and 10 minutes is a good place to start.

## Data retention time frame

SQL DM for MySQL is designed for storing large amounts of data for long periods of time. Data collected before the specified time frame is purged automatically. Time frame may be specified in seconds, minutes, hours and days for a particular server.

## Base time

For calculation of uptime-based counters the current value of each status variable is compared with either of those,

- server status variable 'uptime'
- server status variable 'uptime\_since\_flush\_status'
- SQL DM for MySQL 'base time' setting

If SQL DM for MySQL 'base time' setting is defined and server status variable `uptime_since_flush_status` is available then, `uptime_since_flush_status` is used, if it is not available then base time is used.

The reason for this implementation is that if `FLUSH STATUS` is executed with a MySQL server, the server status variables are reset to the same value as after a server restart. There is one important exception however and that is the 'uptime' status variable itself. This single status variable is not affected by `FLUSH STATUS`.

So, to get true uptime-based counters in SQL DM for MySQL with servers that do not support the `uptime_since_flush_status` variable you need to define a 'base time' in SQL DM for MySQL greater than or equal to the time where `FLUSH STATUS` was executed last time.

But also if `uptime` and/or `uptime_since_flush_status` is large ('old') you may use 'base time' setting to analyze uptime-based counters on an interval defined by you. For instance, if the server has been running for months you may choose to analyze uptime-base counters based on data collected from a specific time only as you have defined it.

Also, note that if the 'base time' is smaller than `uptime` (or `uptime_since_flush_status` if available), then 'base time' setting is ignored. Using a 'base time' larger than 'uptime' and/or `uptime_since_flush_status`, then base time is considered. If a base time is in future, then most recent collection time is considered (similar to Delta).