Using SLA data stored in database

> ⊘ *See our new document.*

Time to SLA app provides various REST APIs to retrieve SLA data stored in app. Sometimes customers want to obtain those information from database mostly to be used with business information tools. This article explains reasons behind limited data access to SLA data in database and how to interpret those limited information.

> ⊘ **Important Notes and Disclaimer**
>
> - Information stored in database by Time to SLA plugin is never meant to be used externally. We do not guarantee the integrity of the information you have collected by those instructions here. Use them at your own risk.
> - Database tables stores vital information for Jira, Time to SLA and many other plugins. Manually working on it might cause irreversible damage or bad application performance due to corrupted table statistics etc.
> - When possible use REST services instead. If something missing from REST services please raise a ticket and we will do our best to provide related services.

AO_C5D949_TTS_ISSUE_SLA : SLA Data on issues

This table stores intermediate information required to speed-up the calculation of updated SLA values on an issue. Each line represents an SLA instance on a single issue.

Column	Type	Explanation	Reference
ID	Number	ID of this SLA instance. Note that this ID is subject to change at any time.	
SLA_ID	Number	ID of the SLA definition this SLA instance belongs to.	AO_C5D949_TTS_SLA.ID
ISSUE_ID	Number	ID of the issue this SLA instance belongs to.	jiraissue.id
ORIGIN_DATE	Date	Exact instant this SLA instance started to count.	
EXPECTED_TARGET_DATE	Date	Deadline for this SLA instance.	
ACTUAL_TARGET_DATE	Date	Actual instant that this SLA has reached its target condition.	
RESET_DATE	Date	Last reset instant for SLA	
INDICATOR *	String	Current state of SLA (EXCEED, STILL or SUCCESS).	
PAUSED	Boolean	Whether or not SLA is paused	
WORKING_DURATION *	Integer	Elapsed time from the start of SLA in milliseconds.	
PAUSED_DURATION *	Integer	Time spent on paused state (when SLA in progress) in milliseconds	
FORMATTED_WORKING_DURATION	String	Formatted working duration, filled only when SLA is completed	
FORMATTED_PAUSED_DURATION	String	Formatted paused duration, filled only when SLA is completed	

* Limitation applies for information stored on those columns please see following:

- Data on this table only updated during issue transitions.
    - This is why you might observe inconsistencies between SLA information on issue view. Because SLA is a an active metric for example remaining duration for an active SLA changes with each second.
    - SLAs are considered as breached once their deadline has been passed yet this is not persisted on database until they go through an issue transition after breaching their deadline.
    - One exception for that is paused SLAs. Target dates of paused SLAs will be always kept updated. (This is done by a scheduled task so it is normal to observe 2-3 minutes inaccuracies.)
- Completed SLAs (SLAs that reached their target status) always shows updated information since there is no actively running clock on them.
    - So if you see FINISHED column true you can be sure that all values in other columns are will be consistent with actual SLA state.
- EXPECTED_TARGET_DATE is a good reference to check whether or not an SLA is breached. It does not change by time (except when an SLA is paused in that case target date shifts further in time and this is properly reflected on database entries with help of a scheduled task.)

Here are some queries to differentiate different SLA states:

Met SLAs

```
"INDICATOR" = 'MET'
```

Progress SLAs (Including paused ones)

```
("INDICATOR" = 'STILL' AND "EXPECTED_TARGET_DATE" > now()) OR "PAUSED" is true
```

Progress SLAs (Excluding paused ones)

```
("INDICATOR" = 'STILL' AND "EXPECTED_TARGET_DATE" > now()) OR "PAUSED" is false
```

ℹ️ SLA Indicator custom field displays paused SLAs as **PROGRESS**

Exceeded SLAs

```
("INDICATOR" = 'EXCEED' OR ("INDICATOR" = 'STILL' AND "EXPECTED_TARGET_DATE" < now())) AND "PAUSED" is false
```