

JWT Release Notes 2.3.3

[2018-03-20] Released [JIRA Workflow Toolbox 2.3.3](#)

New features

- New parser functions

Function	Returned value
dateTime (number year , number month , number dayOfMonth , number hourOfDay , number minute , timeZone time_zone) : number	This function is used for obtaining a date-time literal value from a set of numeric values representing a date-time timestamp. Example: dateTime(2018, 03, 25, 23, 15, LOCAL) returns 2018/03/25 23:15 .
daysInTheMonth (number t , timeZone time_zone) : number	Returns the number of days in the month of timestamp t in timezone time_zone . Example: daysInTheMonth(2016/02/28 00:00, LOCAL) returns 29 , taking into account that 2016 is a leap year.
lastDayOfTheMonth (number t , timeZone time_zone) : number	Returns the timestamp for the last day of the month of timestamp t in timezone time_zone . The timestamp returned is at 00:00 , i.e., just the beginning of the day. Example: lastDayOfTheMonth(2017/02/05 11:31, LOCAL) returns 2017/02/28 00:00 .
nextDayOfTheWeek (number t , number dayOfWeek , timeZone time_zone) : number	Returns the timestamp for the next day of the week represented by dayOfWeek since timestamp t in timezone time_zone . The timestamp returned is at 00:00 , i.e., just the beginning of the day. Example: nextDayOfTheWeek(2018/03/01 12:31, {SUNDAY}, LOCAL) returns 2018/03/04 00:00 , taking into account that 2018/03/01 is Thursday. Example: nextDayOfTheWeek(2018/03/01 12:31, {THURSDAY}, LOCAL) returns 2018/03/08 00:00 .
timeLogged (issue list issues) : number	Returns the sum of all the time logged in issues in milliseconds. Example: timeLogged(subtasks()) returns the sum of time logged in current issue's sub-tasks in milliseconds.
timeLogged (issue list issues , number datetime_ini , number datetime_end) : number	Returns the sum of all the time logged in issues in time interval defined by timestamps datetime_ini and datetime_end . If one or both parameters datetime_ini and datetime_end are null , then it's assumed that the time period hasn't low or high time limit respectively. Logged time is returned in milliseconds. Example: timeLogged(issuesUnderEpic(), datePart({00057}, LOCAL), addDays(datePart({00057}, LOCAL), 1, LOCAL)) returns the sum of time logged today in issues under current issue's Epic. Note that {00057} is field code for Current date and time .
timeLogged (issue list issues , string user) : number	Returns all the time logged in issues by user with username user . Logged time is returned in milliseconds. Argument user can contain a single user name (not be confused with user's full name), or a comma separated list of usernames, group names or project role names. Example: timeLogged(linkedIssues(), %{00003}) returns the sum of time logged by the assignee on linked issues. Note that %{00003} is field code for Assignee .
timeLogged (issue list issues , number datetime_ini , number datetime_end , string user) : number	Returns the sum of all the time logged in issues by user in time interval defined by timestamps datetime_ini and datetime_end . If one or both parameters datetime_ini and datetime_end are null , then it's assumed that the time interval hasn't low or high time limit respectively. Logged time is returned in milliseconds. Argument user can contain a single username (not be confused with user's full name), or a comma separated list of usernames, group names or project role names. Example: timeLogged(subtasks(), 2018/01/01, 2019/01/01, %{00003}) returns the sum of time logged by the assignee on subtasks during 2018. Note that %{00003} is field code for Assignee .

Improvements

- Maximum delay for delayed transition execution now is 60 s, instead of 30 s.

Bug fixes

- Each field update done by JWT post-functions appeared in issue history separately. Since 2.3.3 all the updates are shown in one unique register per issue transition, showing all field updates done by all the post-functions executed in the transition.
- [Issue #264](#) - Fixed bug with on functions **datePart()** and **timePart()** when used on the days where DST transition happens.
- Changed behavior of function **addMonths()** so that when adding 1 month to 30th April we will obtain 30th May instead of 31st May (which was the behavior up to version 2.3.2).
- Error in the way setting rules were rendered by "Syntax Check" button mainly on post-function [Set a field as a function of other fields](#) and also on some other features.
- Parser functions that admits a **comma separated list of usernames** didn't filter duplicated values. In this version we implement a filter to avoid duplications.
- When using JWT post-functions for setting fields in "Create Issue" transition in JSD, emails were not being sent to approvers.