JWT Release Notes 2.3.0

[2017-01-31] **Released Jira Workflow Toolbox 2.3.0**

New features

- Added new post-function **Transition issues**, which can be used as an alternative to classic issue transitioning method based on writing on virtual fields **Issue status**, **Issue status (delayed writing)**, **Execute transition** and **Execute transition (delayed execution)**.
- New parser functions:

Function	Returned value
epic() : issue list	Returns an issue list containing current issue's epic, in case current issue is directly under an epic (e.g., a **Story**). If current issue is a sub-task, then the epic of its parent issue is returned. If current issue is an epic itself, then current issue is returned.
epic(issue list **issues**) : issue list	Returns the list of epic issues under which those issues in argument **issues** are. If some of those issues are sub-tasks, then the epic of their parent is returned. Duplicated issues in argument **issues** are discarded. Output can contain duplicated issues. Example: **`epic(linkedIssues("is blocked by"))`** returns the list of epics of those issues which are blocking current issue.
epic(string **issue_keys**) : issue list	Returns the list of epic issues under which those issues with keys in **issue_keys** are. If some of those issues are sub-tasks, the epic of their parent is returned. Argument **issue_keys** is a comma separated list of issue keys. Duplicated issue keys in argument **issue_keys** are discarded. Output can contain duplicated issues. Example: **`epic("CRM-15, HD-21")`** returns the list of epics under which issues with keys **CRM-15** and **HD-21** are.
issuesUnderEpic() : issue list	Returns an issue list containing issues which are directly under current issue's epic (i.e., **stories** are included in the output, but their **sub-tasks** are not). Current issue's epic is obtained using the logic of function **`epic()`**. Current issue is included in the output, except if current issue is an epic itself.
issuesUnderEpic(issue list **issues**) : issue list	Returns an issue list containing issues which are directly under the epic of issues in argument **issues**. Duplicated issues are filtered from output. Example: **`issuesUnderEpic(linkedIssues("is blocked by"))`** returns the list of issues directly under epics of issues blocking current issue.
issuesUnderEpic(string **issue_keys**) : issue list	Returns an issue list containing issues which are directly under the epic of issues with keys in argument **issue_keys**. Argument **issue_keys** is a comma separated list of issue keys. Duplicated issues are filtered from output. Example: **`issuesUnderEpic("CRM-15, HD-21")`** returns the list of issues directly under epic of issues with keys **CRM-15** and **HD-21**.
siblingIssuesUnderEpic() : issue list	Returns an issue list containing issues which are directly under epic of current issue (i.e., **Stories** are included in the output, but their **sub-tasks** are not), excluding current issue. Current issue should be an issue directly under an epic, (i.e., it can't be a **sub-task** or an **epic**).
siblingIssuesUnderEpic(issue list **issues**) : issue list	Returns an issue list containing issues which are directly under the epic of issues in argument **issues**, excluding issues in argument **issues** from the output. Duplicated issues are filtered from output. Example: **`siblingIssuesUnderEpic(linkedIssues("is blocked by"))`** returns the list of issues directly under epics of issues blocking current issue, excluding from the output issues blocking current issue.
siblingIssuesUnderEpic(string **issue_keys**) : issue list	Returns an issue list containing issues which are directly under the epic of issues with keys in argument **issue_keys**, excluding from the output issues with keys in argument **issue_keys**. Argument **issue_keys** is a comma separated list of issue keys. Duplicated issues are filtered from output. Example: **`siblingIssuesUnderEpic("CRM-15, HD-21")`** returns the list of issues directly under epic of issues with keys **CRM-15** and **HD-21**, excluding from the output issues with keys **CRM-15** and **HD-21**.
startDates(string **versions**) : number list	Returns a number list with the start dates for versions in string **versions** for current issues project. Parameter **versions** is a comma separated list of version names. Example: **`startDates(%{00074})`** returns the list of start dates for fixed versions. Note that **%{00074}** is field code for **Fix version/s**.
startDates(string **versions**, string **projects**) : number list	Returns a number list with the start dates for versions in string **versions** for projects in parameter **projects**. Parameter **versions** is a comma separated list of version names. Parameter **projects** is a comma separated list of project keys or project names. Example: **`startDates(%{00077}, "CRM")`** returns the list of start dates for affected versions for project with key *"***CRM***"*. Note that **%{00077}** is field code for **Affects version/s**.
earliestUnreleasedUnarchivedVersion() : string list	Returns string with the name of the earliest unreleased version in current issue's project excluding archived ones.

earliestUnreleasedUnarchivedVersion(string **projects**) : string list	Returns string with the name of the earliest unreleased version excluding archived ones for projects in argument **projects**. Returned versions may either released or unreleased. Arguments **projects** is a comma separated list of **project keys** or **project names**.
latestReleasedUnarchivedVersion() : string list	Returns string with the name of the latest released version in current issue's project excluding archived ones.
latestReleasedUnarchivedVersion(string **projects**) : string list	Returns string with the name of the latest released version excluding archived ones for projects in argument **projects**. Returned versions may either released or unreleased. Arguments **projects** is a comma separated list of **project keys** or **project names**.

Improvements

- Added new parameters to post-function **Create issues and sub-tasks** for optionally inheriting **comments**, **attachments** and **issue links**.
- Added new parameter **Starting Date-Time** to post-function **Log work**.

Bug fixes

- **Issue #955** - Post-function **Format field value** was failing when formatting a version picker field in Jira 7.x
- Fixed problem with UI in post-function **Create issues and sub-tasks** related with JQL queries and Issue List expressions displaying.
- **Issue #965** - Bug on function `timeDifference(d1, d2, "schedule_name", timezone)` under certain scenarios.
- Fixed problem with license management.