JWT Release Notes 2.1.23

[2014-12-11] **Released Jira Workflow Toolbox 2.1.23**

New features

- New condition "**Transition is triggered by Jira Workflow Toolbox post-function**". This condition is intended to hide transitions that are thought to be triggered exclusively by post-functions, i.e. by writing into fields "*Issue status*", "*Issue status (delayed writing)*", "*Transition issue*" and "*Transit ion issue (delayed transitions)*".
- New conditional operator **? :** (similar to the one in C and JAVA). This operator maybe very useful for composing conditional texts.
- New functions to access the **historic values of fields**:

FUNCTION	RETURNED VALUE
previousValue(% {nnnnn}) : *string*	Returns a string with the previous value of a field for current issue. It will return *null* if field was previously uninitialized.
previousValue({nnnnn}) : number	Returns a number with the previous value of a **numeric** or **date field** for current issue. It will return *null* if field was previously uninitialized.
previousValue(% {nnnnn.i}) : string	Returns a string with the previous value of a **cascading** or **multi-cascading select field** for current issue **at level i** (with root level = 0). It will return *null* if field was previously uninitialized.
fieldHistory(% {nnnnn}) : string list	Returns a list of strings with all the values that a field ever had in the past for current issue. Uninitialized field statuses are represented by *empty strings*.
fieldHistory({nnnnn}) : number list	Returns a list of numbers with all the values that a **numeric** or **date field** ever had in the past for current issue. Uninitialized field statuses are not represented.
fieldHistory(% {nnnnn.i}) : string list	Returns a list of strings with all the values that a **cascading** or **multi-cascading select field** ever had in the past **for level i** (with root level = 0) in current issue. Uninitialized field statuses are represented by empty strings.

Supported fields are: *Summary*, *Description*, *Assignee*, *Reporter*, *Due date*, *Issue status*, *Priority*, *Resolution*, *Environment*, *Fixed versions*, *Affected versions*, *Labels*, *Components*, *Security level* and all the *custom fields*.

Improvements

- Issue #91 - Raised log level for certain messages, in order to reduce verbosity of the plugin in the server's log file.
- UI improvement: Parser help and lists of field codes only shown or hidden by means of buttons.
- UI improvement: Added field code injectors in features "*Create issue link*", "*Log work*", " Read Field From Issues Returned By JQL Query", and "*W rite field on issues returned by JQL query*".
- Added option "**Evaluate all the setting rules, not stoping at first match.**" to post-function *Set field as a function of other fields*". This options is available only for **multi-valued**" target fields, and is intended to be used with prefixes **+** and **-** to add and remove values by means of a set of rules.
- Renamed virtual field "**Last comment's visibility**" to "**Last comment's visibility restrictions**", as a more suitable name for this field.
- Field "**Last comment's visibility restrictions**" now supports *Service Desk's* **Internal** visibility restriction.
- Now writing on "*Cascading Select*" and "*Multi-Cascading Select*" is case unsensitive.
- Improvements in functions "*availableItems({nnnnn}) : string list*" and "*numberOfAvailableItems({nnnnn}) : number*" for supporting the following field types: "*Status*", "*Resolution*", "*Priority*", "*User Picker*", "*Multi-User Picker*", "*Group Picker*" and "*Multi-Group Picker*".

Bug fixes

- Issue #69 - Support for custom fields with IDs longer than 5 digits.
- Issue #80 - Minor bug with Multi-Cascading Select fields
- Bug on function *fieldValue()* - Function *fieldValue()* was retrieving values entered in current's issue transition screen as values of every issue read by the function.
- Bug related with virtual field "Issue Status"
- Bug when writing into virtual field "*Issue Status*" the name issue's current status, and the only transitions available for current user are global transitions.
- Uninitialized field codes were throwing exceptions when used in advanced parsed texts. Now they simply return and empty string.
- Post-function "*Create issue link*" failing in transition "*Create Issue*", when using a JQL query for issue selection and issue is created by Email.
- Fixed bug when trying to clean a cascading select field by writing an empty string into it.
- Parser didn't recognize empty list literal **[ ]**.
- Now, field codes inserted in JQL queries are not auto-double-quoted, since it was cause of errors in some cases.