Virtual Fields

Jira Workflow Toolbox provides a set of special fields called Virtual Fields, making accessible the most **interesting properties of issues, projects and users** to every feature in the plugin. Virtual fields may be read and written by the plugin the same way ordinary custom fields are.

Writable virtual fields

When a value is written into these fields, a feature or system field of the issue the field belongs to will be modified. These fields are also readable.

Field Name	Effect of Writing	Allowed String Values	Allowed Numeric Values	Examples
Summary	Sets issue **Summary**	text string trimmed to 255 characters	numbers are cast to string	-
Description	Sets issue **Description**	unlimited text string	numbers are cast to string	-
Assignee	Assigns the issue to a user	**user name** (not user's full name) or name of **project role**. You can also use string **unassigned** to make the issue become unassigned.	n/a	**Change assignee based on a custom field** **Assign issue based on the value of a Cascading Select custom field** **Assign issue to a specific user based on a specific custom field value** **Unassign an issue when assigned to project leader** **How can I set assignee depending on issue type?**
Reporter	Updates issue **Reporter** to a different user	**user name** (not user's full name) or name of **project role**	n/a	-
Due date	Sets issue's system field **Due Date**	date-time in format defined at **jira.date.time.picker.java.format**, or as in JQL (yyyy/MM/dd HH:mm, yyyy-MM-dd HH:mm, yyyy/MM/dd, yyyy-MM-dd, or relative to current time using "w" (weeks), "d" (days), "h" (hours) or "m" (minutes)	number of minutes relative to current time.	`2014-03-25` : date example `2015-01-31 14:35` : date with time example `- 4d 1h` : 4 days and 1 hour before current time. `2w, 3d, 4h, 5m` : 2 weeks, 3 days, 4 hours and 5 minutes after current time. **Set "Due date" with certain time offset from current date** **Set "Due date" to current date at issue creation if not initialized** **Setting Due Date 6 natural days (or work days) earlier than a Date Picker custom field**
Priority	Updates the issue **Priority**	**name of a Priority** (e.g., Blocker, Critical, Major,...)	(0 = highest priority, 1 = second highest,... n = lowest	**Changing issue priority depending on issue description** **Propagate highest priority from blocked issues to blocking issues** **Set a custom field "Urgency" depending on a combined value of issue's Priority and "Impact" custom field** **Setting priority for issues in certain status for longer than 24 hours**
Original estimate (minutes)	Sets original estimate	cast from string to number will be attempted	number of minutes	`30` : sets **Original estimate** to 30 minutes. **Original estimate (minutes)** can be increased or decreased using post-function **Mathematical and date-time expression calculator** with formulas like: `{00024} + 60` for increasing remaining estimate in 1 hour, or `{00024} - {10000}` for decreasing remaining estimate by the amount of minutes stored in custom field with code `{10000}`.

Remaining estimate (minutes)	Sets remaining estimate	cast from string to number will be attempted	number of minutes	`30` : sets **Remaining estimate** to 30 minutes. **Remaining estimate (minutes)** can be increased or decreased using post-function **Mathematical and date-time expression calculator** with formulas like: `{00024} + 60` for increasing remaining estimate in 1 hour, or `{00024} - {10000}` for decreasing remaining estimate by the amount of minutes stored in custom field with code `{10000}`.
Total time spent (minutes)	Sets time spent.	cast from string to number will be attempted	number of minutes	`30` : sets **Time spent** to 30 minutes. **Total time spent (minutes)** can be increased or decreased using post-function **Mathematical and date-time expression calculator** with formulas like: `{00024} + 60` for increasing remaining estimate in 1 hour, or `{00024} - {10000}` for decreasing remaining estimate by the a mount of minutes stored in custom field with code `{10000}`. • **Sum subtask's Time Spent (work logs) and then add it to a certain linked issue** • **Sum "Time Spent" in all subtasks of issues linked with issue link types "LinkA", "LinkB", "LinkC", etc** • **Setting "Total time spent = Current Date-Time - Date-Time of Last Update"**
Add to time spent (minutes) (since version 2.1.20)	Adds a number of minutes to current value of field "**Total time spent (minutes)**"	cast from string to number will be attempted	number of minutes	`60` : increases time spent in 1 hour. `- {10000}` : decreases time spent by the amount of minutes stored in custom field with code **{10000}**.
Components	Sets issue **Components**	Comma separated list of **component names**. Prefixes + and – can be used to add or remove (check or uncheck) single values or set of values.	n/a	`+ Web Site, User Management` : adds the components **Web Site** and **User Management** to current issue, provided they exist in the project. `- Library A, Component B` : removes the components **Library A** and **Component B** from current issue, provided they exist in the project.
Fixed versions	Sets **Fixed versions**	Comma separated list of **version names**. Prefixes + and – can be used to add or remove (check or uncheck) single values or set of values.	numbers are cast to string	`+ 1.0, 1.1` : adds versions **1.0** and **1.1** to field **Fixed versions**. `- 2.0, 2.2` : removes versions **2.0** and **2.2** from field **Fixed versions**.
Affected versions	Sets **Affected versions**	Comma separated list of **version names**. Prefixes + and – can be used to add or remove (check or uncheck) single values or set of values.	numbers are cast to string	`+ 1.0, 1.1` : adds versions **1.0** and **1.1** to field **Affected versions**. `- 2.0, 2.2` : removes versions **2.0** and **2.2** from field **Affected versions**.
Environment	Sets issue **Environment**	unlimited text string	numbers are cast to string	-
Issue status	Will make the issue progress through the workflow to the written status, provided there is transition in the workflow from current status to the written status. It requieres that all conditions and validations in the transition are satisfied.	**name of a Status** (e.g., Open, In Progress, Resolved,...)	n/a	`Resolved :` will search for a transition from current issue's status to **Resolved** status. If conditions are satisfied, it will try to execute the transition. Then if validators are satisfied in that transition are satisfied, it will be executed, resulting in moving issue to **Resolved** status. Every post-function in the executed transition is also executed. • **Automatically close resolved sub-tasks when parent issue is closed** • **Automatically Cancel subtasks when Parent issue is cancelled** • **Automatically reopen Parent issue when one of its Subtasks is reopened**
Issue status (delayed writing)	Same as virtual field **Issue status**, with the only difference that actual writing is carried out after transition has finished in current issue. It's very useful when you need to make progress linked issues or subtasks blocked by "**Sub-Task Blocking Condition**" or "**Condition/Validation on linked issues**" due to current issue's status.	**name of a Status** (e.g., Open, In Progress, Resolved,...)	n/a	• **Make linked issues, sub-tasks and JQL selected issues progress through its workflows** • **Make parent issue progress through its workflow**
Execute transition (since version 2.1.20)	Execute the transition whose name is written into this field, provided conditions and validators are matched, and the transition is available for current status. This field is an alternative to "**Issue status**".	**name of a transition** (e.g., Open Issue, Start Progress, Resolve Issue, Close Issue,...)	n/a	• **Close parent issue when all sub-tasks are closed** • **Transition sub-tasks when parent is transitioned**

Execute transition (delayed execution) (since version 2.1.20)	Same as virtual field "**Execute transition**", but execution is done once current transition has finished. This field is an alternative to "**Issue status (delayed writing)**"	**name of a transition** (e.g., Open Issue, Start Progress, Resolve Issue, Close Issue,...)	n/a	• **Close parent issue when all sub-tasks are closed**
Resolution	Sets issue **Resolution**	**name of issue Resolution** (e.g., Fixed, Won't Fix, Duplicate, Incomplete,...)	n/a	-
Labels	Sets **Labels**	Comma separated list of **labels**. If label doesn't exist it is created. Prefixes + and – can be used to add or remove (check or uncheck) single values or set of values.	numbers are cast to string	`+ jira, plugin` : adds labels "**jira**" and "**plugin**" to the issue. `- scrum, web` : removes labels "**scrum**" and "**web**" from the issue.
New labels (since version 2.2.1)	Adds new labels to the issue	Comma separated list of **labels**. If label doesn't exist it is created.	numbers are cast to string	-
Attachments (only new attachments will be added)	Adds new attachments coming from another issue keeping current attachments. Rejects duplicated attachments. Can be set by post-functions **Write field on linked issues or sub-tasks** and **Update issue fields**	Comma or blank separated list of **issue keys** whose attachments will be copied to current issue.	n/a	`CMR-1` : adds to current issue the attachments in issue **CRM-1** except if attachments are already in current issue. `CRM-2, HR-34, HR-50` : adds to current issue the attachments in issues **CRM-2**, **HR-34** and **HR-50** except if attachments are already in current issue.
Attachments (current attachments will be replaced)	Replaces current's issue attachments with those coming from another issues. This field can be set by post-functions **Write field on linked issues or sub-tasks** and **Write Field On Issues Returned By JQL Query**	Comma or blank separated list of **issue keys** whose attachments will be copied to current issue.	n/a	`CMR-1` : copy attachments in **CRM-1** to current issue, replacing the existing ones. `CRM-2, HR-34, HR-50` : copy attachments in **CRM-2**, **HR-34** and **HR-50** to current issue, replacing the existing ones.
Security level	Sets issue **Security Level**	**name of a Security Level**	n/a	`Public` : sets security level to **Public**, provided this security level exists in the Security Scheme of the project. `Reserved` : sets security level to **Reserved**, provided this security level exists in the Security Scheme of the project. • **Setting Security Level depending on Reporter or Creator**
Last comment	Updates the last comment added to the issue if it exists	unlimited text string Optionally parameter **visibility** can be added for setting comment visibility. To do it add : `{visibility=visibility_value}` at the end of the text. This parameter admits the same values as "**Last comment's visibility restriction**". (**See example**) Since version 2.2	numbers are cast to string	**WARNING:** When writing into this field, please add your writer post-function AFTER "**Add a comment to an issue if one is entered during a transition.**" post-function, since otherwise updated comment will be the one before last, instead of the last one. **Example**
Last comment's visibility restriction Formerly called "Last comment's visibility"	Updates the visibility of the last comment in the issue if it exists. Can be used to set the visibility of a comment introduced with virtual field **New comment**	**name of a *Project Role*** or **name of a *Group***. In Jira Service Desk it also admits values **public** and **internal** (without doble quotes), or alternatively **jsd_public** and **jsd_internal** to avoid name collision with existing project roles and user groups. Since version 2.2	n/a	`Developers` : for setting visibility to project role or group with name **Developers**. `jsd_internal` : for setting JSD internal visibility. `jira-developers` : for setting visibility to "**jira-developers**" user group.

New comment	Inserts a new comment into the issue. It doesn't trigger any event for email notifications.	unlimited text string Optionally parameter **visibility** can be added for setting comment visibility. To do it add : `{visibility=visibility_value}` at the end of the text. This parameter admits the same values as "**Last comment's visibility restriction**". (**See example**) Since version 2.2	numbers are cast to string	This is a comment only visible to developers group.: {visibility=jira-developers} This is a comment only visible to Admin project role.: {visibility=Administrators} This is a comment only visible to Jira Service Desk agents and collaborators.: {visibility=jsd_internal} This is a comment also visible to Jira Service Desk customers.: {visibility=jsd_public} • **Create a comment in subtasks** • **Creating a Jira Service Desk internal comment** • **Creating a JSD's internal comment on linked issues**
New comment (sends email notifications) (since version 2.2.22)	Inserts a new comment into the issue, triggering an event for **sending email notifications**.	unlimited text string Optionally parameter **visibility** can be added for setting comment visibility. To do it add : `{visibility=visibility_value}` at the end of the text. This parameter admits the same values as "**Last comment's visibility restriction**". (**See example**) Since version 2.2	numbers are cast to string	-
Watchers	Sets watchers of an issue. Can be used to add or remove watchers to linked issues, sub-tasks or issues returned by JQL, but requires to use operator '+' and '-'. Also can be used to copy watchers between issues.	comma separated list of **user names**, **group names** or **project roles names**. Prefixes + and – can be used to add or remove single users or set of users.	n/a	`albert, richard, john` : sets 3 users as watchers replacing current ones. `+ Developers, Administrators` : adds every user in project roles or groups **Developers** and **Administrators** to current watchers. `- Testers, Administrators, + albert, richard` : removes users in project roles or groups **Testers** and **Administrators** and adds users **albert** and **richard** as watchers. • **Automatically become watcher of every issue blocking an issue assigned to you** • **Add watchers depending on the value of a custom field**
New watchers	Adds new watchers to an issue. Can be used to add new watchers to linked issues, sub-tasks or issues returned by JQL more easily that with virtual field *Watchers*, since you don't need to use operator *'+'*.	comma separated list of **user names**, **group names** or **project roles names**.	n/a	`albert, richard, john` : adds 3 watchers keeping current watchers untouched. • **Add watcher depending on security level** • **Automatically become watcher of every issue blocking an issue assigned to you**
Ephemeral string X	There are 5 ephemeral string fields. These fields are used to **hold temporary text-string values** returned by a post-function, which are used as input by another post-function. **Once the transition execution has ended, all ephemeral fields are cleared**, i.e., these fields can't be used to pass values from one transition to another.	unlimited text string values	numbers are cast to string	-
Ephemeral number X	There are 5 ephemeral number fields. These fields are used to **hold temporary numeric and date-time values** returned by a post-function, which are used as input by another post-function. **Once the transition execution has ended, all ephemeral fields are cleared**, i.e., these fields can't be used to pass values from one transition to another.	String containing base 10 representation of a number, with dot character (i.e., .) as separator of integer and fractional parts	integers or real numbers, and also any Date-Time field (Date Picker, Date-Time Picker, Due Date, Created, etc.) which are stored as the number of milliseconds elapsed since January 1, 1970, 00:00:00 GMT	• **Example of usage of an ephemeral number field to hold a date value**

These fields are set by any of the following post-functions:

- **Copy parsed text to a field**: to set a field using a literal string or the value of another field, i.e., can be used to copy one field into another field.
- **Write field on linked issues or sub-tasks**
- **Update issue fields**
- **Set a custom field "Urgency" depending on a combined value of issue's Priority and "Impact" custom field**: used to update the value of a field depending on the value of other fields.
- **Mathematical and date-time expression calculator**: this is the better way to set **Number**, **Date** or **Date-Time** fields. Also can be used to set system's fields **Priority**, **Original estimate**, **Remaining estimate** and **Time spent**.
- **Create issues and sub-tasks**: fields are set on created issues.

Values returned when reading virtual fields

Field name	Type	Value returned when cast to string
Summary	Text string	Issue summary.
Description	Text string	Issue description.
Assignee	User	The name of the user who has the issue assigned.
Assignee's full name	Text string	Name and surname of the user who currently has the issue assigned
Assignee's email	Text string	Email address of the user who currently has the issue assigned
Reporter	User	The name of the user who reported the issue. This field is editable, while **Creator** field isn't.
Reporter's full name	Text string	Name and surname of the user who created the issue
Reporter's email	Text string	Email address of the user who created the issue
Creator (since version 2.1.20)	User	User name of the user who actually creates the issue in JIRA. This field, unlike **Reporter**, can't be edited.
Creator's full name (since version 2.1.20)	Text string	Name and surname of the user who actually created the issue in JIRA
Creator's email (since version 2.1.20)	Text string	Email address of the user who actually created the issue in JIRA
Current user	User	Name of the user who is executing the transition.
Current user's full name	Text string	Name and surname of the user who is executing the transition.
Current user's email	Text string	Email address of the user who leads the project the issue belongs to.

Current date and time	Date and Time	Current date and time of JIRA server's clock. When cast to string format defined at **jira.date.time.picker.java.format** is used.
Date and time of creation	Date and time	Date and time when issue has been created. When cast to string format defined at **jira.date.time.picker.java.format** is used.
Date and time of last update	Date and time	Date and time of the most recent issue update. When cast to string format defined at **jira.date.time.picker.java.format** is used.

Date and time of latest status change (since version 2.1.22)	Date and time	Date and time of the most recent status change in the issue. When cast to string format defined at **jira.date.time.picker.java.format** is used.
Due date	Date	Field that can store a date with no time part, and is used to schedule issues. When cast to string format defined at **jira.date.time.picker.java.forma**
Date and time of resolution	Date and time	Date and time of the most recent update of field **Resolution**. When cast to string format defined at **jira.date.time.picker.java.format** is used.
Priority	Priority	Name of the priority in **Default language** configured in the JIRA instance
Original estimate (minutes)	Number	Time originally estimated for the issue expressed in minutes. It's a real number, so it may have a non-zero fractional part.
Remaining estimate (minutes)	Number	Remaining time estimated for the issue expressed in minutes. It's a real number, so it may have a non-zero fractional part.

Total time spent (minutes)	Number	Work time currently spent at the issue expressed in minutes. It's a real number, so it may have a non-zero fractional part.
Work logged in transition (minutes) (since version 2.1.20)	Number	Number of minutes logged in current transition screen.
Date and time of work logged in transition (since version 2.1.34)	Date and time	Date and time introduced for the beginning of the work logged in current transition screen.
Components	Components	Comma separated list of component names
Components leaders	Multi user	Comma separated list of user names
Fixed versions	Versions	Comma separated list of fixed versions
Fixed versions with details	Text string	A text with a line for each fixed versions. Each line contains the following information separated by characters '#': name of version, description, relea
Number of fixed versions	Number	Number of fixed versions in current issue.
Affected versions	Versions	Comma separated list of fixed versions
Affected versions with details	Text string	A text with a line for each affected versions. Each line contains the following information separated by characters '#': name of version, description, re
Number of affected versions	Number	Number of affected versions in current issue.
Number of votes received	Number	Number of votes received by the issue.

Keys of subtasks	Text string	Comma separated list of issue keys.
Number of subtasks	Number	Number of subtasks of current issue.
Keys of linked issues	Text string	Comma separated list of issue keys.
Number of linked issues	Number	Number of issues linked to current issue.
Environment	String	Field to describe the environment where the issue has occurred.
Attachments	Attachments	Comma separated list of file names of all files attached to the issue.
Attachments with details	Text string	Comma separated list of file names with its mime types and sizes in Kbytes, for every file attached to the issue.
Number of attachments	Number	Number of files attached to current issue.
Transition's attachments	Attachments	Comma separated list of file names of all files attached to the issue in current transition's screen.
Transition's attachments with details	Text string	Comma separated list of file names with its mime types and sizes in Kbytes, for every file attached in current transition's screen.
Number of transition's attachments	Number	Number of files attacher to current issue in transition screen. It's useful to enforce attachments in transition screen, by means of **Boolean condition terms**.

Recent attachments (since version 2.2.20)	Attachments	Comma separated list of file names of all files attached to current issue in the last minute (60 seconds).
Recent attachments with details (since version 2.2.20)	Text string	Comma separated list of file names with its mime types and sizes in Kbytes, for every file attached to current issue in the last minute (60 seconds).
Labels	Labels	Blank separated list with all the labels of the issue.
Number of labels	Number	Number of labels in current issue.
Issue key	Text string	Issue key
Issue type	Issue type	Name of issue's **Issue Type** in Jira's default language
Issue status	Issue status	Name of current status of the issue in Jira's default language.

Issue status category (since version 2.1.20)	Status category	Name of the category in Jira's default language.
Previous issue status (since version 2.1.20)	Issue status	Name of the status the issue was just before current one. The name is shown in Jira's default language. This field will only return the last different st status, use the parser function **previousValue(%{00016})**
Previous issue status category (since version 2.1.20)	Status category	Name of the category of the status the issue was just before current one. The name is shown in Jira's default language. The category will be read fr last different status and return the category.
Current transition (since version 2.2.26)	Text string	The **name of current transition**. In versions previous to **2.2.43**, when used in workflow **conditions**, this field behaves like virtual field **Available transitions**. When used in post-functions, you should ensure that your post-function is inserted **before** JIRA system post-function **"Set issue status to the linke** otherwise an empty string is returned.
Available transitions (since version 2.2.43)	Text string	A comma separated list with the **names of transitions** with origin in current issue status.
Target status (since version 2.2.26)	Text string	The **name of the target status of current transition**. In versions previous to **2.2.42**, when used in workflow **conditions**, this field behaves like virtual field **Available target statuses**.
Available target statuses (since version 2.2.42)	Text string	A comma separated list with the **names of target status** of those transitions with origin in current issue status.
Resolution	Resolution	The name of the resolution currently set at the issue, or empty string if **Resolution** is not set. The name of the resolution is shown in Jira's default la
Security level	Security level	Name of the security level the issue currently has.
Last comment	Text string	Last comment entered in the issue.

Last comment's visibility restriction	Text string	Name of a **group** or **project role** the visibility of the last comment entered in the issue is restricted to. If no restriction is applied to the comment, this special **Internal** visibility restriction, string `"Internal"` is returned. This field can be used to check visibility restriction of field entered in transition v When using a post-function for setting this field in order to set visibility restrictions over the comment entered in the transition screen, always insert t **comment to an issue if one is entered during a transition.**", like in this example:
Last commenter (since version 2.1.33)	User	Name of the user who entered the last comment in the issue.
Transition's comment	Text string	Comment entered in transition screen. This field can be used to make comment mandatory in transition screen, or to validate the content of commer
Watchers	Multi-user	Comma separated list of user names of current watchers of the issue.
Project key	Text string	**Project Key** of the project the issue belongs to.
Project id	Text string	**Project id** from the current project the issue belongs to.
Project name	Text string	Name of the project the issue belongs to.
Project description	Text string	Field **Description** of the project the issue belongs to.
Project URL	Text string	Field **URL** of the project belongs to.
Project category	Text string	Name of the **Category** of the project the issue belongs to.
Project leader	User	Name of the user who leads the project the issue belongs to.
Project leader's full name	Text string	Name and surname of the user who leads the project the issue belongs to.
Project leader's email	Text string	Email address of the user who leads the project the issue belongs to.
Rest of issues in the project	Text string	Comma separated list of issue keys in the project except current issue's key.
Workflow scheme	Text string	The name of the workflow scheme of the project current issue belongs to.

JIRA base URL (since version 2.2.12)	Text string	Returns the base URL of current JIRA instance.
Custom er Reques t Type Name (since version 2.2.28)	Text string	The **name** of the **Customer Request Type**.
Tempo Accoun t Key (since version 2.2.28)	Text string	The **key** of Tempo Timesheets' **Account** field.
Tempo Accoun t Name (since version 2.2.28)	Text string	The **name** of Tempo Timesheets' **Account** field.
Sprint ID (since version 2.2.29)	Text string	The internal ID for the sprint of current issue. This value can be used for setting field **Sprint** in other issues.
Sprint Start Date (since version 2.2.40)	Date	Sprint's start date.
Sprint End Date (since version 2.2.40)	Date	Sprint's end date.
Sprint Comple te Day (since version 2.9.3)	Date	Sprint's completion date.

For every virtual and custom field in a sub-task, there is also another virtual field representing the **same field in its parent issue**, making it very easy to read and write parent's issue fields.